THE UNIVERSITY OF ARIZONA

NEW ACADEMIC PROGRAM – MAJOR
Preliminary Proposal Form

I. **Program Details**

a. **Name (and Degree Type) of Proposed Academic Programs:** MS Software Engineering

i. **Emphases (if applicable):** None

b. **Academic Unit(s)/College(s):** College of Engineering:
2303 - Electrical and Computer Engineering (50%) and
2302 - Systems and Industrial Engineering (50%)

c. **Campus/Location(s):** Main Campus (Tucson) and UArizona Online

d. **First Admission Term:** Fall 2023

e. **Primary Contact and Email:** Sharon ONeal *sharononeal@arizona.edu*

II. **Executive Summary:**

The future looks very promising for the software engineering discipline and profession. The global market of software engineering, alone, will be worth $64 billion by 2025, and it is a vital part of a bigger industry. The overall software market size was $567 million globally in 2020 and is projected to reach $1,021 million by 2025, projected to almost doubling in size within 5 years. Some of the factors behind this growth include increased automation in multiple industries, the demand for cloud-based solutions, the internet of things and an increase of devices which can be used in daily life for convenience. In response to this projected growth, we are proposing the creation of MS Software Engineering degree programs at the University of Arizona, which is aligned with several UArizona strategic pillars.

UArizona launched its BS Software Engineering in Fall 2021, developing a strong pipeline of incoming students to the MS program. As with the BS SFWE program, the proposed program will be co-hosted in the Electrical & Computer Engineering (ECE) and Systems and Industrial Engineering (SIE) departments. The ECE and SIE departments already offer a rich array of courses that can be used as electives within the proposed programs.

The MS SFWE programs will serve local, state, and national increasing needs in engineering computing talent related to economic development and national security and are aligned with Arizona's New Economic Initiative. The program will also support and enable the University of Arizona's growth goals / initiatives to increase student enrollments, research opportunities, and collaborations with our faculty and external entities.

III. **Brief Program Description:**

The MS Software Engineering curriculum applies software engineering fundamentals to develop and produce computing-based products/solutions. The MS SFWE program is grounded in solid engineering practices and principles governed by the IEEE Software Engineering Body of Knowledge (IEEE SWEBOK). IEEE SWEBOK recommends specific skills that all software engineers should possess regardless of their programming languages and platforms. Our curriculum focuses on the application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software.

The MS SFWE program has a firm engineering foundation that encompasses discovery-based education utilizing an experiential learning approach. As a part of the curriculum, students will complete projects in areas that emphasize software engineering, communication, teamwork, critical thinking, and engineering professionalism. Students will also conduct novel research in many diverse software engineering related areas. The program's flexibility allows students to design their course of study / research from a diverse pool of courses and research opportunities in software and computer engineering domains such as web and mobile applications, embedded systems, cybersecurity, machine learning, systems, and other interdisciplinary areas.

Program Rationale:

Two of the College of Engineering's 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 MS degree program plays a critical role in both pillars. The students graduating with a MS in Software Engineering degree will be better positioned to develop the skills and mindsets to be leaders in the areas of software development, computing, machine learning, ever-increasing automation and connectivity, human and intelligent systems, data science, and network sciences.

By offering competitive, relevant, and experiential-based learning Software Engineering programs 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 of Engineering and University at large. Hence, we will recruit faculty that can significantly impact software engineering and computing areas of research and education. As we have realized with recruitment for faculty in the undergraduate program in Software Engineering, having a graduate level program is also essential to recruit world-class faculty to the program. These faculty will pursue research grants to advance the state-of-the-art in applied software engineering and integrate their research into the curricula. 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 graduate degrees. 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.

IV. Projected Enrollment for the First Five Years:

The planned start date for the MS program in Software Engineering is Fall 2023. The projected enrollment in the program is shown in the table below *(note that it was extended out to a 5-year projection)*. The basis for these projections was derived by comparing enrollments at other Arizona and AAU universities that have a similar Software Engineering graduate programs.

Degree	Year 1 (2023 / 2024)	Year 2 (2024 / 2025)	Year 3 (2025 / 2026)	Year 4 (2026 / 2027)	Year 5 (2027 / 2028)

MS	20	50	100	150	200

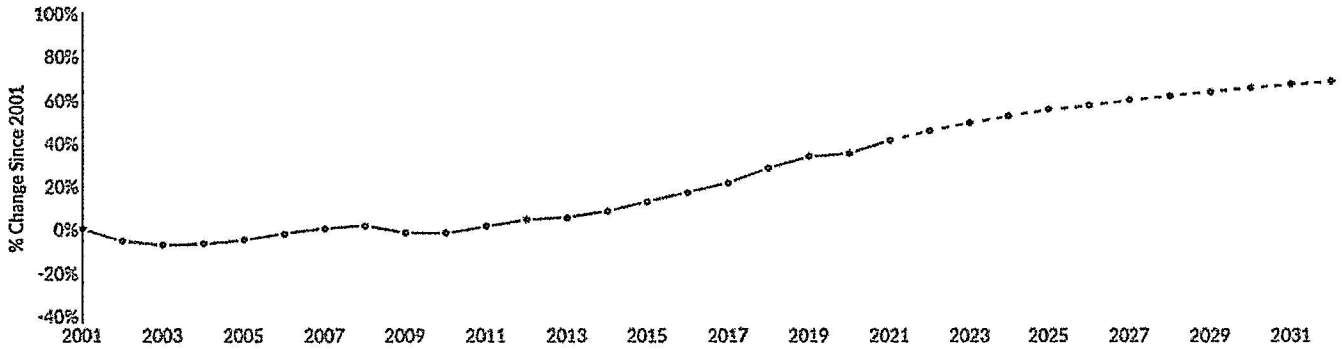
V. **Evidence of Market Demand:**

The market demand for those trained in engineering computing /software engineering disciplines (CIP code 14.093 - Computer Software Engineering) is projected to have significant growth in both the near-term and long-term future. As shown in the diagram below, sourced from Lightcast Q3 2022 data[1], the number of regional[2] jobs in the workforce with a MS or other professional degree is expected to see a 11.3% increase over 5 years, between 2021 – 2026 (and continues to increase at a similar rate through at least 2031).

[1] Lightcast Q3 2022 Data Set, www.economicmodeling.com

[2] Regional jobs include the states located in the southwest region of the United States

Regional Trends



Region	2021 Jobs	2026 Jobs	Change	% Change
• Region	3,074,978	3,421,626	346,648	11.3%

Thus, the new SFWE MS degree program will serve both local, state, and national needs related to employment, economic development, and national security. Indeed, these degree programs are among the most important in support of the ongoing fourth industrial revolution and in close alignment with Arizona's New Economy Initiative[3].

The full marketing and analysis report can be found at the following link:
https://arizona.box.com/s/g2sm18hc6gwxi5th7fch49vnau4etsrb

[3] World Economic Forum. https://www.weforum.org/agenda/2016/01/the-fourth-industrial-revolution-what-it-means-and-how-to-respond/

VI. **Similar Programs Offered at Arizona Public Universities:**

University	Program	College
University of Arizona	BS Software Engineering	College of Engineering
University of Arizona	MS Computer Science PhD Computer Science	College of Science
Arizona State University	MS Software Engineering	School of Computing and Augmented Intelligence, IRA A Fulton Schools of Engineering
Arizona State University	MS Computer Science PhD Computer Science	School of Computing and Augmented Intelligence, IRA A Fulton Schools of Engineering
Northern Arizona University	MS Computer Science PhD Information and Computing	School of Informatics, Computing, and Cyber Systems

VII. **Resources**

a. The table below summarizes the projected new resources required to offer the program:

Resources	Quantity
Faculty	5
Staff	2
Other (TAs, Graders, LAs) *(Semester hires over 5 years)*	34 TAs *(total # semester hires over 5 years, averaging 3.5 TAs/year)* 0 Graders 0 LAs
Equipment	None
Facilities	Office and lab space *(for new faculty)*

b. Estimate total expected cost: $5,796,299 ***(extrapolated over 5 years)***

c. Estimate total expected revenue of the program: $5,791,074 ***(extrapolated over 5 years)***

VIII. **Required Signatures**

a. ~~**Graduate Studies Committee Chair**~~ **/ Main Proposer:**

i. **Signature:** _____

ii. **Name and Title:** ~~Dr Vignesh Subbian, Chair SIE Graduate Studies Committee~~ Director, Software Engineering

iii. **Date:** 10/21/2022

b. **Managing Unit/Department Head:**

i. **Signature:** Ricardo Valerdi

ii. **Name and Title:** Dr Ricardo Valerdi, Systems and Industrial Engineering (SIE) Dept Head

iii. **Date:** 10/21/2022

c. **Managing Unit/Department Head:**

i. **Signature:** _____

ii. **Name and Title:** Dr Michael Wu, Electrical and Computer Engineering (ECE) Dept Head

iii. **Date:** 10/21/2022

d. **College Dean/Associate Dean:**

i. **Signature:** _____

ii. **Name and Title:** Dr David Hahn, Dean College of Engineering

iii. **Date:** 10/21/2022

THE UNIVERSITY OF ARIZONA

ACADEMIC ADMINISTRATION
Administration Building, 402
1401 E. University Blvd.
PO Box 210066
Tucson, AZ 85721-0066

To: David Hahn, Dean, College of Engineering
Ricardo Valerdi, Department Head, Systems and Industrial Engineering (SIE), College of Engineering
Michael Wu, Department Head, Electrical and Computer Engineering (ECE), College of Engineering
Sharon O'Neal, Director, Software Engineering Program, College of Engineering

From: Greg Heileman, PhD, Vice Provost for Undergraduate Education

Date: November 28, 2022

Subject: Approval of Preliminary Proposal for MS and PhD in Software Engineering

---

Thank you for submitting the preliminary proposal for a MS and PhD in Software Engineering. The proposed academic programs should provide an excellent educational opportunity and useful degrees for students. We believe your ideas are sufficiently well developed that it now makes sense to advance through the stages of the formal academic program approval process.

Please proceed to the development of a full proposal, and do not hesitate to reach out the Curricular Affairs Office for assistance with this process.

CC: Liesl Folks, Senior Vice President for Academic Affairs and Provost
Liz Sandoval, Director, Curricular Affairs

THE UNIVERSITY OF ARIZONA

New Academic Program Workflow Form

General

Proposed Name: Software Engineering

Transaction Nbr: 00000000000160

Plan Type: Major

Academic Career: Graduate

Degree Offered: Master of Science

Do you want to offer a minor? N

Anticipated 1st Admission Term: Fall 2023

Details

Department(s):

ENGR

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

Campus(es):

MAIN

LOCATION	DESCRIPTION
TUCSON	Tucson

ONLN

LOCATION	DESCRIPTION
ONLN	Online

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

Plan admission types:

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

Non Degree Certificate (UCRT only): Y

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 Master of Science Software Engineering

Transcript: Y Master of Science Software Engineering

Conditions for Admission/Declaration for this Major:

Bachelor's degree from an institution recognized by the UA. Students who do not have a degree equivalent to the UA Bachelor of Science degree in a computing related program may be admitted into the graduate SFWE program but may be required to complete additional graduate level courses prior to enrolling in specific graduate courses. This policy also applies to students in both the MS-thesis and the MS non-thesis option.

- Grade-point average of 3.0 in the overall undergraduate degree or in their last 60 units of the undergraduate degree.
- Applicants whose native language is not English are required by the Graduate College to take an English proficiency test (see Graduate College admissions requirements for list of approved tests).
- Students in the MS non-thesis option are expected to be self-supported or supported by external fellowships or industry.

Requirements for Accreditation:

N/A

Program Comparisons

University Appropriateness

The University of Arizona College of Engineering currently does not offer a graduate 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 market analysis), it is likely that students are selecting other universities since UArizona does not offer graduate degrees in Software Engineering. It is believed that offering an innovative Software Engineering graduate 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 ENGR to diversify their degree offerings by offering this new degree. This will attract more students to the University of Arizona.

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

By offering competitive, relevant, and experiential-based learning Software Engineering programs 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 of Engineering and University at large. Hence, we will recruit faculty that can significantly impact software engineering and computing areas of research and education. As we have realized with recruitment for faculty in the undergraduate program in Software Engineering, having a graduate level program is also essential to recruit world-class faculty to the program. These faculty will pursue research grants to advance the state-of-the-art in applied software engineering and integrate their research into the curricula. 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
3	Computer Science	MS	250	University of Arizona / Main	N
5	Software Engineering	MS	148	ASU / Main / Online	N
6	Computer Science	MS	1245	ASU / Main / Online	N
7	Computer Science	MS	100	NAU / Main	N

Peer Comparison

See attached file for a detailed analysis of the proposed Software Engineering graduate degree compared to the Software Engineering degree programs at Arizona State University and Carnegie Mellon University.

Faculty & Resources

Faculty

Current Faculty:

INSTR ID	NAME	DEPT	RANK	DEGREE	FCLTY/%
02600592	Kenneth Head	2302	Professor	Doctor of Philosophy	.25
03308095	Jerzy Rozenblit	2303	Professor	Doctor of Philosophy	.25
06606459	Michael Marefat	2303	Professor	Doctor of Philosophy	.25
10608029	Salim Hariri	2303	Professor	Doctor of Philosophy	.25
16600630	Loukas Lazos	2303	Professor	Doctor of Philosophy	.25
17109846	Jian Liu	2302	Assoc. Prof	Doctor of Philosophy	.25
22053534	Ricardo Valerdi	2302	Professor	Doctor of Philosophy	.20
22063694	Samuel Peffers	2302	Assoc. Prof. Pract.	Doctor of Philosophy	.25
22067654	Ravi Tandon	2303	Professor	Doctor of Philosophy	.25
22067655	Tosiron Adegbija	2303	Professor	Doctor of Philosophy	.25
22071369	Vignesh Subbian	2302	Assoc. Prof	Doctor of Philosophy	.25
22072066	Sharon O'Neal	2302	Prof. Pract.	Master of Science	.50
22092594	Fredrick	2302	Adj. Lect.	Master of	.25

INSTR ID	NAME	DEPT	RANK	DEGREE	FCLTY/%
	Steiner			Science	
22094595	Mohammad Abu matar	2302	Assoc. Prof. Pract.	Doctor of Philosophy	.50
22095228	Abhijit Mahalanobis	2303	Assoc. Prof	Doctor of Philosophy	.25
22095632	Soheil Salehi Mobarakeh	2303	Assit. Prof	Doctor of Philosophy	.25
23190064	Pratik Satam	2302	Assit. Prof	Doctor of Philosophy	.50

Additional Faculty:

In the first 3 years of the program, it is projected that the following Tenure Track faculty will be required (total of 3):

2023/2024 - 1
2024/2025 - 1
2025/2026 - 1

Additionally, in the first 3 years of the program, it is projected that the following Career Track faculty will be required (total of 2):

2023/2024 - 1
2024/2025 - 1
2025/2026 - 0

Current Student & Faculty FTE

DEPARTMENT	UGRD HEAD COUNT	GRAD HEAD COUNT	FACULTY FTE
2302	325	210	27.35
2303	465	305	48.00

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	325	400	475	230	260	310	30.00	32.00	33.00
2303	525	595	765	310	325	355	55.00	60.00	64.00

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.

New laboratory equipment needed for Tenure Track faculty is included in estimated start-up packages and will vary depending on the nature of the research for acquired new faculty members.

Additional Facilities Required & Anticipated:

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.

Office and laboratory space will be required for new faculty.

Other Support

Other Support Currently Available:

The College of ENGR and the SIE and ECE Departments are currently well structured and to be able to accommodate the new program, including IT support. Additional staff will be required and described below.

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:

Graduate Advisor: 1

Finance Support: 1

Other staff support HR, Communications, IT: 1

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

Comments During Approval Process

11/30/2022 3:55 PM

WILLIAMSCINDY

Comments
Removed Distance Campus.

1/9/2023 10:29 PM

RVALERDI

Comments
Approved.

1/12/2023 3:24 PM

ESANDMAR

Comments
Approved.

THE UNIVERSITY OF ARIZONA

ACADEMIC PROGRAM – ADDITIONAL INFORMATION FORM

To be used once the preliminary proposal has been approved.

I. MAJOR REQUIREMENTS -

Master of Science (MS) – Software Engineering

Total units required to complete the degree	30
Pre-admissions expectations (i.e., academic training to be completed prior to admission)	Bachelor's degree from a regionally accredited USA university or equivalent international university recognized by the UA. Students who do not have a degree equivalent to the UA Bachelor of Science degree in a computing related program may be admitted into the graduate SFWE program but may be required to complete additional graduate-level pre-requisite courses prior to enrolling in some graduate courses. This policy also applies to students in both the MS-thesis and the MS non-thesis option. • Grade-point average of 3.0 in the overall undergraduate degree or meet Graduate College minimum admissions requirements. • Applicants whose native language is not English are required by the Graduate College to take an English proficiency test. A description of acceptable tests can be found at: https://grad.arizona.edu/admissions/requirements/international-applicants#english-proficiency • Students in the MS non-thesis option are expected to be self-supported or supported by external fellowships or industry.
Major requirements. List all major requirements including core and electives. If applicable, list the emphasis requirements for each proposed emphasis*. 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	Required *Core* Courses (12 units required): • SFWE 507 (3) - Foundations of Software Engineering (NEW) And 3 courses of additional Core: • SFWE 502 (3) - Software DevSecOps • SFWE 503 (3) - Software Project Management • SFWE 504 (3) - Software Requirements Analysis and Test (NEW) • SFWE 505 (3) - Software Architecture and Design (NEW)

department head(s) for courses not owned by your department.	• SFWE 506 (3) – Distributed Computing (NEW) Students that have a BS in Software Engineering and have demonstrated a given Core course's competencies ***must*** consult/work with their faculty advisor to substitute the course with one of the SFWE 5xx level courses listed in the **Technical Computing Electives** below. **Complete one of the following options:** ***Emphasis 1: Thesis Option:*** Complete 12 units from the **Technical Computing Electives** listed below or in a closely related computing field (must be approved by graduate studies committee*).* ○ Any SFWE core courses not used to meet the 12-units of *Core* coursework requirement can also be used as technical electives ○ SFWE 501 (3) - SW Assurance ○ SFWE 508 (3) –Data Mining (NEW) ○ SFWE 509 (3) – Cloud Computing Principles and Practices (NEW) ○ SFWE 510 (3) – Cloud Native Software Engineering (NEW) ○ SFWE 511 (3) – Software for Industrial Control Systems (NEW) ○ SFWE 512 (3) –Robotics (NEW) ○ SFWE 513 (3) – Software Engineering Research Methods (NEW) ○ CSE 501 (3) – Operating System Design ○ ECE 503 (3) - Probability and Random Processes for Engineering Applications ○ ECE 509 (3) –Cybersecurity Concept, Theory, Practice ○ ECE 513 (3) –Web Development and the IoT ○ ECE 523 (3) –Engineering Applications of Machine Learning and Data Analytics ○ ECE 562 (3) –Computer Architecture and Design ○ ECE 576A (3) - Engineering of Computer Based Systems ○ ECE 576B (3) - Embedded System Design and Optimization

- ECE 579 (3) –Principles of Artificial Intelligence
- SIE 533 (3) –Fundamentals of Data Science for Engineers
- SIE 558 (3) –Model Based Systems Engineering
- SIE 577 (3) – Introduction to Biomedical Informatics
- *Other courses may be added at the discretion of the faculty advisor and GSC, or as additional new SFWE courses not listed in section **III. New Courses Needed** are developed.*

Also complete 6 units of thesis (SFWE 910)

Emphasis 2: Non-Thesis Option:

Complete 18 units from the **Technical Computing Electives** list below or in a closely related computing field (must be approved by graduate studies committee*).*

- Any SFWE core courses not used to meet the 12-units of *Core* coursework requirement can also be used as technical electives
- SFWE 501 (3) - SW Assurance
- SFWE 508 (3) –Data Mining (NEW)
- SFWE 509 (3) – Cloud Computing Principles and Practices (NEW)
- SFWE 510 (3) – Cloud Native Software Engineering (NEW)
- SFWE 511 (3) – Software for Industrial Control Systems (NEW)
- SFWE 512 (3) –Robotics (NEW)
- SFWE 513 (3) – Software Engineering Research Methods (NEW)
- CSE 501 (3) – Operating System Design
- ECE 503 (3) - Probability and Random Processes for Engineering Applications
- ECE 509 (3) –Cybersecurity Concept, Theory, Practice
- ECE 513 (3) –Web Development and the IoT
- ECE 523 (3) –Engineering Applications of Machine Learning and Data Analytics
- ECE 562 (3) –Computer Architecture and Design
- ECE 576A (3) - Engineering of Computer Based Systems

	○ ECE 576B (3) - Embedded System Design and Optimization ○ ECE 579 (3) –Principles of Artificial Intelligence ○ SIE 533 (3) –Fundamentals of Data Science for Engineers ○ SIE 558 (3) –Model Based Systems Engineering ○ SIE 577 (3) – Introduction to Biomedical Informatics ○ *Other courses may be added at the discretion of the faculty advisor and GSC, or as additional new SFWE courses not listed in section **III. New Courses Needed** are developed.*
Research methods, data analysis, and methodology requirements (Yes/No). If yes, provide description.	If the student selects the Thesis Option, they are required to complete 6 units of research and thesis development in an area related to Software Engineering. It is also recommended (but not required) that one of their technical electives be SFWE 513 – Software Engineering Research Methods.
Internship, practicum, applied course requirements (Yes/No). If yes, provide description.	None
Master thesis or dissertation required (Yes/No). If yes, provide description.	If the student selects the *Thesis Option*, they are required to complete 6 units of research in an area related to software engineering. The thesis will capture the results of the research that was completed under the guidance of a faculty member. The thesis paper should demonstrate the students core knowledge, technical skills, and the ability to articulate and synthesize the findings of a project they were engaged in over multiple semesters. Students must pass an oral defense of their thesis.
Additional requirements (provide description)	None
Minor options (as relevant)	No required minor options for the MS degree.

*Emphases are officially recognized sub-specializations within the discipline. ABOR Policy 2-221 c. Academic Degree Programs Sub specializations 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.

II. CURRENT COURSES–using the table below, list all existing courses included in the proposed major. You can find information to complete the table using the UA course catalog or UAnalytics (Catalog and Schedule Dashboard> "Printable Course Descriptions by Department" On

Demand Report; right side of screen).

Course prefix and number (include cross-listings)	Units	Title	Pre-requisites	Modes of delivery (online, in-person, hybrid)	Typically Offered (F, W, Sp, Su)	Dept signed party to proposal? (Yes/No)
ECE 509	3	Cybersecurity Concept, Theory, Practice	ECE 578	In-person	Sp	Yes
ECE 513	3	Web Development and the IoT	ECE 275 (or equivalent)	In-person	F	Yes
ECE 503	3	Probability and Random Processes for Engineering Applications	Undergraduate course in Probability	In-person	F	Yes
ECE 523	3	Engineering Applications of Machine Learning and Data Analytics	ECE 503 or equivalent	In-person	Sp	Yes
ECE 562	3	Computer Architecture and Design	ECE 369A (or consent of instructor)	In-person	Sp	Yes
ECE 576A	3	Engineering of Computer Based Systems	ECE 579	In-person	F	Yes
ECE 576B	3	Embedded System Design and Optimization	ECE 576A	In-person	Sp	Yes
ECE 579	3	Principles of Artificial Intelligence	ECE 373 (or equivalent)	In-person	Sp	Yes
SIE 533	3	Fundamentals of Data Science for Engineers	SIE 530 or SIE 500A (or consent of instructor)	In-person, online	F	Yes

SIE 558	3	Model Based Systems Engineering	SIE 554A (or consent of instructor)	In-person, online	F	Yes
SIE 577	3	Introduction to Biomedical Informatics	None	In-person, online	F	Yes
SFWE 501	3	Software Assurance (co-convened with SFWE 401) (updated to include graduate level requirements)	CSE 201, ECE 275 or consent of instructor	In-person, online	Sp	Yes
SFWE 502	3	Software DevSecOps (co-convened with SFWE 402) (updated to include graduate level requirements)	CSE 201, ECE 275 or consent of instructor	In-person, online	F	Yes
SFWE 503	3	Software Project Management (co-convened with SFWE 403) (updated to include graduate level requirements)	Consent of instructor	In-person, online	F	Yes

III. **NEW COURSES NEEDED** – using the table below, list any new courses that must be created for the proposed program. If the specific course number is undetermined, please provide level (i.e., CHEM 4XX). Add rows as needed.

Course prefix and number (include cross-listings)	Units	Title	Pre-requisites	Modes of delivery (online, in-person, hybrid)	Status*	Anticipated first term offered	Typically Offered (F, W, Sp, Su)	Dept signed party to proposal? (Yes/No)	Faculty members available to teach the courses
SFWE 507	3	Foundations of Software Engineering	ECE 275 (or equivalent)	In-person, online	D	Fall 2023	F, Sp	Yes	Dr Mohammad Abu Matar
SFWE 504	3	Software Requirements Analysis and Test	SFWE 507 (recommended) (or consent of instructor)	In-person, online	D	Spring 2024	Sp,	Yes	Sharon ONeal
SFWE 505	3	Software Architecture and Design	SFWE 507 (recommended) (or consent of instructor)	In-person, online	D	Fall 2024	F	Yes	Dr Mohammad Abu Matar
SFWE 506	3	Distributed Computing	SFWE 507 (recommended) (or consent of instructor)	In-person, online	D	Spring 2025	Sp	Yes	Sharon ONeal
SFWE 508	3	Data Mining	ECE 275 (or equivalent)	In-person, online	D	Spring 2025	F	Yes	TBR (New Faculty)
SFWE 509	3	Cloud Computing Principles and Practices	SFWE 507 (recommended) (or consent of instructor)	In-person, online	D	Spring 2024	F	Yes	Dr Mohammad Abu Matar

SFWE 510	3	Cloud Native Software Engineering	SFWE 507 (recommended) (or consent of instructor)	In-person, online	D	Fall 2024	Sp	Yes	TBR (New Faculty)
SFWE 511	3	Software for Industrial Control Systems	ECE 275 (or equivalent)	In-person, online	D	Fall 2023	F	Yes	Dr Pratik Satam
SFWE 512	3	Robotics	ECE 275 (or equivalent)	In-person, online	D	Fall 2025	Sp	Yes	TBR (New Faculty)
SFWE 513	3	Software Engineering Research Methods	None	In-person, online	D	Fall 2024	F	Yes	Dr Pratik Satam
SFWE 910	6	Thesis	Thesis Option	In-person, online	D	Fall 2024	F, Sp	Yes	Varying

*In development (D); submitted for approval (S); approved (A)

Click or tap here to enter text.

IV. **FACULTY INFORMATION-** complete the table below. If UA Vitae link is not provided/available, add CVs to a Box folder and provide that link. UA Vitae profiles can be found in the UA directory/phonebook. Add rows as needed. **NOTE: full proposals are distributed campus-wide, posted on committee agendas and should be considered "publicly visible". Contact Office of Curricular Affairs if you have concerns about CV information being "publicly visible".**

Faculty Member	Involvement	UA Vitae link or Box folder link
Sharon ONeal	Teach SFWE 503, SFWE 504 and SFWE 506	Sharon L ONeal | UA Profiles (arizona.edu)
Dr Mohammad Abu Matar	Teach SFWE 507 and SFWE 509 (Technical Elective)	https://arizona.box.com/s/7trdpezytljufs2b1bn8fzgawtq8k46g
Dr Pratik Satam	Teach SFWE 501 and SFWE 511 (Technical Electives) and conduct/collaborate in SFWE related research	https://arizona.box.com/s/k41xa25zolvk5qfw0qj5lni0d4a4e3ao

Dr Ravi Tandon	Teach ECE 503 (Technical Elective) and conduct/collaborate in SFWE related research	https://profiles.arizona.edu/person/tandonr
Dr Tosiron Adegbija	Teach ECE 562 (Technical Elective) and conduct/collaborate in SFWE related research	Tosiron Adegbija | UA Profiles (arizona.edu)
Dr Abhijit Mahalanobis	Teach ECE 523 (Technical Elective) and conduct/collaborate in SFWE related research	https://arizona.box.com/s/eximdtrp92tutik04yrfclbyyzj1zidh
Dr Loukas Lazos	Teach ECE 578 (Technical Elective and conduct/collaborate in SFWE related research	Loukas Lazos | UA Profiles (arizona.edu)
Dr Salim Hariri	Teach ECE 509 (Technical Elective) and conduct/collaborate in SFWE related research	Salim A Hariri | UA Profiles (arizona.edu)
Dr Soheil Salehi Mobarakeh	Teach ECE 513 (Technical Elective) and conduct/collaborate in SFWE related research	Soheil Salehi | UA Profiles (arizona.edu)
Dr Michael Marefat	Teach ECE 579 (Technical Elective) and conduct/collaborate in SFWE related research	Michael M. Marefat | UA Profiles (arizona.edu)
Dr Jerzy Rozenblit	Teach ECE 576 A and 576B Conduct/collaborate in CSE related research	https://profiles.arizona.edu/person/jerzyr
Dr Jian Liu	Teach SIE 533 (Technical Elective) and conduct/collaborate in SFWE related research	Jian Liu | UA Profiles (arizona.edu)
Dr Vignesh Subbian	Teach SIE 578 (Technical Elective) and conduct/collaborate in SFWE related research	Vignesh Subbian | UA Profiles (arizona.edu)
Frederick Steiner	Teach SIE 558 (Technical Elective)	Rick Steiner | UA Profiles (arizona.edu)
Dr Ricardo Valerdi	Conduct/collaborate in SFWE related research	https://profiles.arizona.edu/person/rvalerdi

V. **GRADUATION PLAN –** provide a sample degree plan, based on your program that includes all requirements to graduate with this major and takes into consideration course offerings and sequencing.

The table below represents a sample MS SFWE degree plan for students completing the MS degree program in 4 semesters. Because of the flexibility in this degree program, the degree plan of an individual student may differ from what is shown. Each student will develop a tailored degree plan with the SFWE graduate advisor and faculty advisor. The student's degree plan will then be approved by the student's faculty advisor and/or the Graduate Studies Committee (GSC).

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
SFWE 507	3	SFWE 502,503, 504,505,506	3	SFWE 502,503, 504,505,506	3	Technical Elective	3
Technical Elective	3	Technical Elective	3	Technical Elective	3	Technical Elective/SFWE 910 (Thesis Option)	3
SFWE 502,503, 504,505,506	3			Technical Elective/ SFWE 910 (Thesis Option)	3		
Total	9	Total	6	Total	9	Total	6

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
N/A		N/A		N/A		N/A	
Total	-	Total	-	Total	-	Total	-

An alternate sample MS SFWE degree plan for graduate students wanting to complete the MS degree in 3 semesters (9-12-9) is captured in the table below:

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
SFWE 507	3	SFWE 502,503, 504,505,506	3	Technical Elective	3	N/A	N/A
Technical Elective	3	Technical Elective	3	Technical Elective	3		
SFWE 502,503, 504,505,506	3	SFWE 502,503, 504,505,506	3	Technical Elective/ SFWE 910 (Thesis Option)	3		
		Technical Elective/ SFWE 910 (Thesis Option)	3				
Total	9	Total	12	Total	9	Total	N/A

VI. **Curriculum Map and Assessment Map** - Complete this table as a summary of your learning outcomes and assessment plan, using these examples as a model. If you need assistance completing this table and/or the Curriculum Map, please contact the Office of Instruction and Assessment. Attach your Curriculum Map here.

Program: MS Software Engineering

Learning Outcome #1: Demonstrate broad knowledge in the student's field in software engineering.
Concepts: Students will study diverse topics in software engineering including software engineering fundamentals, software development and security operations (SW DevSecOps), software requirements analysis and test, software architecture and design, distributed computing. They also have the option to select from a broad range of software engineering and computing related technical electives. Courses may include computing topics such as data mining, cloud computing, software for industrial control systems, robotics and a variety of other computing topics that vary based on the electives the student opts to take.
Competencies: Students will demonstrate the ability to design, develop, test, integrate and evaluate varied software applications/products/systems in diverse computing and engineering domains. Students opting for the *Thesis-Option* to satisfy their course requirements may also apply their acquired knowledge in these areas to conduct original and novel research in state-of the-art

and advanced software engineering principles, processes, and methodologies to meet the requirements/needs of diverse engineering applications.

Assessment Methods: For every new 5xx / 6xx SFWE course, a rubric will be created that identifies criteria/source of evidence, assessment measures, and an achievement level rating for specified course performance indicators used to measure this outcome. For each course that contributes to this outcome, specific student artifacts for a given course will be evaluated and assessed.
The sources of evidence can include class assignments, exams, projects, papers / reports and other forms of student work.
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 program/course. For existing courses (predominately technical electives), the evidence used to measure the effectiveness of the student outcome have been defined and will be followed. The rubric achievement levels will include: "Exemplary", "Satisfactory", "Developing", and "Unsatisfactory".

At the end of every semester, a team comprised of the course instructor and the SIE/ECE Graduate Studies Committee (GSC)s, will score the rubric using the assessment measures 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 as appropriate. 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.

Additionally, students are asked to complete a survey near the end of their MS program to self-evaluate how well they feel they met the Student Learning Outcomes for the MS program. In addition to the survey, the student is required to compose a brief description of how a project or projects completed during the course of their program serves to demonstrate their ability to analyze, design, and implement a SFWE-related system. The written description and survey are used to gather program-level assessment data.

A rubric is used to evaluate a student's written description. The completed survey and written description are returned to the SFWE Graduate Advisor, who archives this data. At the end of the spring semester, the surveys and written descriptions are provided to the SFWE Program Director and the appropriate Engineering department heads, who ensure the written descriptions are evaluated. The results of the surveys and descriptions will be organized and formatted over the summer. During the following academic year, the Graduate Studies Committee reviews the direct measures (written descriptions) and the indirect measures (student surveys) and then make recommendations for changes in response to the findings. Once each year, the summarized data, and proposed changes, if any, are reviewed and discussed by the SFWE faculty.

Measures: Rubrics will be used for the specific graded student projects in a given 5xx course that clearly evaluates the learning objectives and outcomes of the assignment and/or projects that students are asked to complete. All rubrics are developed by faculty members with expertise in software engineering domains. Faculty may consult with instructional designers as appropriate to ensure the course learning outcomes are measurable and contribute to the overall program learning outcomes. Exams are also used to assess and measure student

learning. Throughout the degree program, the student's core course cumulative GPA will be used as a global measure of the overall student software engineering knowledge. Student surveys and program written description are also collected and evaluated to determine the overall program outcomes as students exit the program.
Learning Outcome #2 (Non-Thesis Option): Analyze the requirements, quality attributes and implementation of a software engineering system or subsystem.
Concepts: Students will derive, develop and analyze the requirements and quality attributes for a software product / system / subsystem. Additionally, students will also develop test plans and procedures that verify/validate the correct implementation of the requirements of the specified product. Students will also evaluate whether a software implementation meets its associated quality attributes and requirements as specified.
Competencies: Using industry best practices, students will learn to elicit, analyze, specify and validate functional and non-functional software requirements. Students will derive, develop and analyze the software requirements and quality attributes for a specific software product / system / subsystem. Students will also develop software test plans and procedures to validate that the implemented product meets its requirements. Students will also evaluate the implemented software to verify that it meets specified quality attributes and requirements.
Assessment Methods: This outcome will be assessed predominately through implementation of a semester project that meets a given or derived set of software requirements for a software / computing product. Midterm and final exams that test the students comprehension of the software engineering concepts taught and competencies acquired will also be used. Rubrics will be created that identify criteria/source of evidence, assessment measures, and an achievement level rating for specified course performance indicators used to measure this outcome. For each course that contributes to this outcome, specific student project artifacts for a given course will be evaluated and assessed. 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 program/course. For existing courses (predominately technical electives), the evidence used to measure the effectiveness of the student outcome have been defined and will be followed. The rubric achievement levels will include: "Exemplary", "Satisfactory", "Developing", and "Unsatisfactory". At the end of every semester, a team comprised of the course instructor and the SIE/ECE Graduate Studies Committee (GSC)s, will score the rubric using the assessment measures 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 as appropriate. 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.
Measures: Instructor grading of course exams and software projects for each course that contributes to this outcome. Rubrics will be used to measure the specific project evidence for each course. Descriptions that clearly measure the given evidence are used to assign

the students achievement level. The achievement levels for the rubrics include: "Exemplary", Satisfactory", "Developing", and "Unsatisfactory".
Learning Outcome #2 (Thesis Option): Critically analyze published research results in the student's area of study.
Concepts: *Thesis-Option* students will learn various research techniques to interpret methods used and results from software related research papers, journals, presentations, and/or conferences. Throughout the program, students will have the opportunity to attend several seminars presented by a diverse group of researchers / scholars and faculty from a broad spectrum of software and computing related fields/areas. They will learn techniques used to critically read published research papers/journals, explore writing techniques used in technical/academic works, learn to develop evidence-based arguments, and draw conclusions from the sources being reviewed. They will also be provided numerous resources and learn to develop strategies for acquiring and using technical references from a variety of sources.
Competencies: Students will demonstrate their ability to read and interpret various forms of software research information, papers, conference proceedings, and data collected to support research. Students will also demonstrate their understanding of techniques used to write technical papers and journal articles. Students will also be able to analyze and explain research approaches taken and results included in published software engineering research papers, journals and conference proceedings.
Assessment Methods: This outcome will be assessed in the student's software engineering related research project and written thesis under the guidance of a faculty advisor. The thesis is reviewed by an examining committee consisting of at least 3 faculty (2 of which must be Graduate Faculty) from the SIE and ECE departments and chaired by the faculty advisor prior to the student conducting an oral presentation/defense of their thesis. The outcome will also be assessed in specific research-oriented assignments from SFWE 513 coursework.
Measures: This outcome will be measured by instructor grading of research related coursework in SFWE 513. It will also be assessed by the student's written thesis and oral defense using the collective results of a Program Assessment Survey completed by all thesis committee members.
Learning Outcome #3 (Non-Thesis Option): Design and implement a system in a software engineering area based on given specifications.
Concepts: Students will use industry best practices, methods, and tools in architecting, modeling, and designing software systems. Students also investigate and evaluate the importance of developing a sound software architecture and design as part of the evolution of the software implementation of a computing system/product. Working in teams on larger-scale semester projects, students implement their designs and show the traceability between requirements, software design, and the resulting code implementation for a specified system/product.
Competencies: Using industry best practices, students will architect/design specified software products for diverse applications including mobile, cloud, embedded, or other computing applications. Students will collaborate with other students and software advisors to complete software projects using modern modeling tools and methodologies (i.e. UML or equivalent). Students will also implement/develop their software designs for a variety of computing-based products. Students will present/review their designs and implementations in a professional setting with other software engineering students, faculty and industry professionals.

Assessment Methods: This outcome will be assessed predominately through implementation of a semester project that meets a given specification for a software / computing-based product. Midterm and final exams that test the students comprehension of the software engineering concepts taught and competencies acquired will also be used. Rubrics will be created that identify criteria/source of evidence, assessment measures, and an achievement level rating for specified course performance indicators used to measure this outcome. For each course that contributes to this outcome, specific student project artifacts for a given course will be evaluated and assessed. 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 program/course. For existing courses (predominately technical electives), the evidence used to measure the effectiveness of the student outcome have been defined and will be followed. The rubric achievement levels will include: "Exemplary", "Satisfactory", "Developing", and "Unsatisfactory". At the end of every semester, a team comprised of the course instructor and the SIE/ECE Graduate Studies Committee (GSC)s, will score the rubric using the assessment measures 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 as appropriate. 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.
Measures: Instructor grading of course exams and computing projects for each course that contributes to this outcome. Rubrics will be used to measure the specific project evidence for each course. Descriptions that clearly measure the given evidence are used to assign the students achievement level. The achievement levels for the rubrics include: "Exemplary", Satisfactory", "Developing", and "Unsatisfactory".
Learning Outcome #3 (Thesis Option): Conduct original research on a significant software engineering problem.
Concepts: *Thesis-Option* students will utilize their acquired knowledge and new research strategies to conduct novel research in a software engineering field of specialization. Students will meet regularly with their faculty advisor and others within the ECE / SIE departments to receive guidance and coaching in a variety of research areas. Students will summarize all aspects of their research and their findings in a written thesis that will be defended orally before a faculty thesis committee.
Competencies: Students will demonstrate their ability to articulate all aspects of their research in a software engineering specialization area, describe and defend the significance of their research, describe methodologies used in conducting the research, and summarize their overall findings resulting from said research.
Assessment Methods: A thesis committee will assess the originality, merit, and contributions of the candidate's research. The written thesis and oral defense is facilitated by a faculty committee consisting of at least 3 faculty (2 of which must be graduate faculty) from the SIE and ECE departments and chaired by the faculty advisor. All members of the thesis committee will be asked to complete a Program Assessment Survey, and the results will be culminated together to form an overall assessment of the student's thesis.
Measures: Evaluation of the student's final written thesis. The thesis will be evaluated by a faculty led committee that assesses the originality, merit, and contributions of the candidate's research. This includes their ability to (a) identify and critically evaluate relevant

literature, (b) formulate and solve original problems using software engineering theory and methods, and (c) interpret and communicate research ideas, data and findings.
Learning Outcome #4: Communicate and defend (written and oral) results of projects or research to peers and broader engineering audiences.
Concepts: Students will utilize their acquired software engineering skills and knowledge to communicate effectively in both written and oral mediums. This may be accomplished in a variety of methods including presenting the results of software / computing related projects to peers, faculty, and potentially industry experts. Additionally, students that opt for the *Thesis-Option* will prepare a written thesis that demonstrates all aspects of their research including the significance of their work, a detailed review of relevant literature, methodologies employed and/or developed, significant findings from the work, a critical discussion of the findings, limitations, and the impact, and potential for future research. *Thesis* students will also be required to present their research findings in an oral defense of their project / research.
Competencies: Students will demonstrate their ability to articulate all aspects of the product development and/or research in a software engineering specialization area. Students opting for the course work only option of the degree program, will describe the relevant design or implementation details of projects that they have developed/implemented, describe methodologies used in developing the software products, and demonstrate designs and/or working products. Students opting for the *Thesis-Option* of the degree program will also describe and defend the significance of their projects / research in a thesis, describe the methodologies used in conducting the software related research, and summarize their overall findings resulting from said research in written and oral mediums.
Assessment Methods: For all SFWE students, specific project rubrics will be used for the specific graded student project for a given course that clearly evaluates the learning objectives and outcomes of the projects that students are asked to complete. All rubrics are developed by faculty members with expertise in associated software engineering domains. Faculty may consult with instructional designers as appropriate to ensure the course learning outcomes are measurable and contribute to the overall program learning outcomes. Additionally, for *Thesis-Option* students, a thesis committee will assess the originality, merit, and contributions of the candidate's research. The written thesis and oral defense is facilitated by a faculty committee appointed consisting of at a minimum 3 faculty from the SIE and/or ECE departments and chaired by the faculty advisor.
Measures: For all students, rubrics will be used for specific graded student projects in a given 5xx course that clearly evaluates the learning objectives and outcomes of the assignment and/or projects that students are asked to complete. All rubrics are developed by faculty members with expertise in software engineering domains. Faculty may consult with instructional designers as appropriate to ensure the course learning outcomes are measurable and contribute to the overall program learning outcomes. Exams are also used to assess and measure student learning. Throughout the degree program, the student's core course cumulative GPA will be used as a global measure of the overall student software engineering knowledge. For *Thesis-Option* students, evaluation of the student's final written and oral thesis. The thesis will be evaluated by a faculty led committee that assesses the originality, merit, and contributions of the candidate's research. This includes their ability to (a) identify and

critically evaluate relevant literature, (b) formulate and solve original problems using software engineering theory and methods, and (c) interpret and communicate research ideas and findings.

The Taskstream Curriculum Map is shown below. Note that the assessment plan includes only new SFWE courses that are part of this program. Existing technical elective courses used to fulfill the degree requirements are not included. The assessment plans for existing courses will be followed per the home department's assessment plan for each respective existing course.

MS Software Engineering

Courses and Activities Mapped to MS Software Engineering

	Outcome					
	SLO 1 Demonstrate broad knowledge in his/her field in software engineering.	SLO 2: Non-Thesis Option Analyze the requirements, quality attributes, and implementation of a software engineering system or subsystem.	SLO 2:Thesis Option Critically analyze published research results in his/her area of study.	SLO 3: Non-Thesis Option Design and implement a system in a software engineering area based on given specifications.	SLO 3: Thesis Option Conduct original research on a significant software engineering problem.	SLO 4 Communicate and defend (written and oral) results of projects or research to peers and broader engineering audiences.
Courses and Learning Activities						
SFWE 507 Foundations of Software Engineering	IPA	I		I		IPA
SFWE 502 Software DevSecOps	P/A			P/A		P/A
SFWE 503 Software Project Management	P/A					IPA
SFWE 504 Software Requirements Analysis and Test		P/A		P/A		P/A
SFWE 505 Software Architecture and Design	P/A			P/A		P/A
SFWE 506 Distributed Computing		P/A		P/A		P/A
SFWE 508 Data Mining	P/A	P/A				P/A
SFWE 509 Cloud Computing Principles and Practices	P/A	P/A		P/A		P/A
SFWE 510 Cloud Native Software Engineering	P/A	P/A		P/A		P/A
SFWE 511 Software for Industrial Control Systems		P/A		P/A		P/A
SFWE 512 Robotics		P/A		P/A		
SFWE 513 Software Engineering Research Methods			IPA		IPA	P/A
SFWE 910 Thesis		P/A	IPA		IPA	P/A
Exit Survey Exit survey (Indirect)	A	A	A	A	A	A

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

VII. PROGRAM ASSESSMENT PLAN- using the table below, provide a schedule for program evaluation 1) while students are in the program and 2) after completion of the major. Add rows as needed. Delete EXAMPLE rows.

MS Thesis Students

The Software Engineering (SFWE) MS Thesis Option program requires a student complete an oral defense of their thesis. The student also completes a survey near the end of their MS program to self-evaluate how well they feel they met the Student Learning Outcomes for the MS SFWE program. The thesis defense and survey are used to gather program-level assessment data.

A rubric to evaluate a student's thesis work is developed by the SFWE faculty. Each faculty committee member completes the rubric at the end of the oral thesis defense. These completed rubrics are returned to the SFWE Graduate Advisor, who archives this data. At the end of the Spring semester, the completed rubrics and surveys are provided to the SFWE Program Director and appropriate Engineering department heads, who will organize and format the results during the summer. During the following academic year, the Graduate Studies Committee (GSC) reviews the direct measures (thesis rubrics) and the indirect measures (student surveys) and makes recommendations for changes in response to the findings. Once each year, the summarized data, and proposed changes, if any, are reviewed and discussed by the SFWE faculty.

MS Non-Thesis Students

The Software Engineering (SFWE) MS Non-Thesis Option program requires a student complete 30 units of graduate coursework. This coursework has limitations as described in the SFWE Graduate Handbook. The student completes a survey near the end of their MS program to self-evaluate how well they feel they met the Student Learning Outcomes for the MS SFWE program. In addition to the survey, the student is required to compose a brief description of how a project or projects completed during the course of their program serves to demonstrate their ability to analyze, design, and implement a SFWE-related system. The written description and survey are used to gather program-level assessment data.

- A rubric is used to evaluate a student's written description. The completed survey and written description are returned to the SFWE Graduate Advisor, who archives this data. At the end of the spring semester, the surveys and written descriptions are provided to the SFWE Program Director and the appropriate Engineering department heads, who ensure the written descriptions are evaluated. The results of the surveys and descriptions will be organized and formatted over the summer. During the following academic year, the Graduate Studies Committee reviews the direct measures (written descriptions) and the indirect measures (student surveys) and then make recommendations for changes in response to the findings. Once each year, the summarized data, and proposed changes, if any, Any SFWE core courses not used to meet the 12-units of *Core* coursework requirement can also be used as technical electives
- SFWE 501 (3) - SW Assurance (co-convened with SFWE 401)

- SFWE 508 (3) –Data Mining (NEW)
- SFWE 509 (3) – Cloud Computing Principles and Practices (NEW)
- SFWE 510 (3) – Cloud Native Software Engineering (NEW)
- SFWE 511 (3) – Software for Industrial Control Systems (NEW)
- SFWE 512 (3) –Robotics (NEW)
- SFWE 513 (3) – Software Engineering Research Methods (NEW)
- CSE 501 (3) – Operating System Design
- ECE 503 (3) - Probability and Random Processes for Engineering Applications
- ECE 509 (3) –Cybersecurity Concept, Theory, Practice
- ECE 513 (3) –Web Development and the IoT
- ECE 523 (3) –Engineering Applications of Machine Learning and Data Analytics
- ECE 562 (3) –Computer Architecture and Design
- ECE 576A (3) - Engineering of Computer Based Systems
- ECE 576B (3) - Embedded System Design and Optimization
- ECE 579 (3) –Principles of Artificial Intelligence
- SIE 533 (3) –Fundamentals of Data Science for Engineers
- SIE 558 (3) –Model Based Systems Engineering
- SIE 577 (3) – Introduction to Biomedical Informatics

Other courses may be added at the discretion of the faculty advisor and GSC, or as additional new SFWE courses not listed in section ***III. New Courses Needed*** *are developed.*

Assessment Measure	Source(s) of Evidence	Data Collection Point(s)
Rubrics for all new courses used to assess each student outcome that identifies criteria, measure of assessment, and an achievement level rating *(i.e., Exemplary, Satisfactory, Developing, Unsatisfactory).*	Specifically targeted: • Class assignments • Exams • Course Projects • Course Reports • Other forms of student work tailored to any specific course)	End of each semester the specific courses are taught.
MS final written thesis *(Thesis-Option only)*	Written thesis	Written and evaluated at the conclusion of the student's thesis efforts to assess the

		merit and contributions of the student's research and findings.
Thesis oral presentation / defense (*Thesis-Option* only)	Oral presentation / defense of the student's thesis	At the completion of the thesis, the student will present their work to a faculty thesis committee and answer any general questions related to their work
Graduation exit survey (used for indirect measures of outcomes).	Student survey	At student graduation
Academic Program Review	APR reviewer evaluation responses	Minimum of every 7 years
Data from continuous improvement implementation efforts (recommended by the Graduate Studies Committee (GSC))	Assessment data	At the end of an academic year

VIII. **ANTICIPATED STUDENT ENROLLMENT**-complete the table below. What concrete evidence/data was used to arrive at the numbers?

5-YEAR PROJECTED ANNUAL ENROLLMENT					
	1st Year	2nd Year	3rd Year	4th Year	5th Year
Number of Students					
MS Degree	20	50	100	150	200

Data/evidence used to determine projected enrollment numbers:

Several regional and/or AAU universities with MS and PhD Software Engineering programs were canvassed for program enrollment. The table that follows shows the total enrollment in some of these programs[1]:

University	MS Software Engineering Student Enrollment (2020)
Arizona State University	148
Drexel University	57
University of Michigan - Dearborn	51
Stevens Institute of Technology	Not Available *(Note: 27 MS degrees were awarded in 2019 and 59 MS degrees were awarded in 2020-2021*
Carnegie Mellon	Not Available *(Note: 177 MS degrees were awarded in 2019 - 2020*[2]
University of California - Irvine	95

[1] Enrollments derived from https://shinyapps.asee.org/apps/Profiles/

[2] Carnegie Mellon University MS in Software Engineering - College Factual

IX. **ANTICIPATED DEGREES AWARDED**- complete the table below, beginning with the first year in which degrees will be awarded. How did you arrive at these numbers? Take into consideration departmental retention rates. Use National Center for Education Statistics College Navigator to find program completion information of peer institutions offering the same or a similar program.

PROJECTED DEGREES AWARDED ANNUALLY					
	1st Year	2nd Year	3rd Year	4th Year	5th Year
Number of Degrees	0	16	50	75	100

Data/evidence used to determine number of anticipated degrees awarded annually:

The estimates in the table above are based on the projected total enrollments over the first 5 years of the program, which includes potential student transfers into the program in the earlier years of the program. Students are expected to graduate as soon as the second year of the program.

Additionally, an analysis of other university graduation rates in similar programs was performed as defined in the National Center for Education Statistics[3] (also included numbers from ASEE[4]). In particular, the universities considered are shown in the table below:

University	Number of MS Awards Conferred in either 2019-2020 or 2020-2021	ASEE Number of MS Awards Conferred in 2019 and 2020
Arizona State University	105	192
Drexel University	34	20
University of Michigan - Dearborn	0	0
Stevens Institute of Technology	59	27 (no data available in 2020)
Carnegie Mellon	140	63
University of California - Irvine	48	34

[3] https://nces.ed.gov/collegenavigator/
[4] https://shinyapps.asee.org/apps/Profiles/

X. **PROGRAM DEVELOPMENT TIMELINE-** describe plans and timelines for 1) marketing the major and 2) student recruitment activities.

The graduate level programs in Software Engineering utilize several existing courses within the College of Engineering, including:

- Electrical and Computer Engineering (ECE) Department
- Systems and Industrial Engineering (SIE) Dept
- Software Engineering (SFWE) program (which is co-owned by ECE and SIE)

The table below shows the preliminary plan for the new MS Software Engineering course development required for the program. We will work closely with UArizona's University Center for Assessment, Teaching and Technology (UCATT) and UA Online to execute the plan shown in the table below.

Course Number / Name	Planned Development Timeframe	First Semester Offered
SFWE 507 - Foundations of Software Engineering	Fall 2023	Spring 2024
SFWE 504 - Software Requirements Analysis and Test	Fall 2023	Spring 2024
SFWE 505 - Software Architecture and Design	Spring 2024	Fall 2024
SFWE 506 - Distributed Computing	Fall 2024	Spring 2025
SFWE 508 - Data Mining	Fall 2024	Spring 2025
SFWE 509 - Cloud Computing Principles and Practices	Fall 2023	Spring 2024
SFWE 510 - Cloud Native Software Engineering	Spring 2024	Fall 2024
SFWE 511 - Software for Industrial Control Systems	Spring 2024	Fall 2024
SFWE 512 - Robotics	Spring 2025	Fall 2025
SFWE 513 – Research Methods	Spring 2024	Fall 2024
SFWE 910 - Thesis	Spring 2024	Fall 2024

We will also work closely with the recruitment and marketing teams (MarCom) within the College of Engineering to market the program as soon as ABOR approves the degree program. Additionally, we will also work closely with Arizona Online to market the program through their marketing channels.

IX. **Program Fees and Differential Tuition (PFDT) Request** – For implementation of fees, you must work with University Fees. The annual deadline is December 1. For any questions, please contact the University Fees Program Manager.

None planned for this program.

Appendix B. Emphasis Print Information-if applicable, complete the table below to indicate if proposed emphases should be printed on transcript and diploma. Add rows as needed. Note: emphases are displayed on transcript and diplomas as " _____ Emphasis".

Emphasis	Print on transcript	Print on diploma
MS Software Engineering – Thesis Option	No	No
MS Software Engineering – Non-Thesis Option	No	No

THE UNIVERSITY OF ARIZONA

ACADEMIC PROGRAM – ADDITIONAL INFORMATION FORM

To be used once the preliminary proposal has been approved.

Appendix C. ABOR Form

Request to Establish New Academic Program in Arizona

Please complete all fields. Boxes may be expanded to accommodate longer responses. Clarifying field descriptions can be found below. Should you have any questions or concerns, please email Helen Baxendale, Director of Academic Affairs and Policy at helen.baxendale@azregents.edu

University: *University of Arizona*

Name of Proposed Academic Program: Master of Science Software Engineering (SFWE)
Academic Department: **College of Engineering:** 2303 - Electrical and Computer Engineering (50%) 2302 - Systems and Industrial Engineering (50%)
Geographic Site: MS SFWE – Main Campus (Tucson) and UArizona Online
Instructional Modality: Primary modalities will be: In-person Online / ONLN (Note: there may be iCourses and/or hybrid courses offered to complement the In-person and Online modalities.)
Total Credit Hours: 30 units

Proposed Inception Term: Fall 2023

Brief Program Description:

The MS Software Engineering curriculum applies software engineering fundamentals to develop and produce computing-based products/solutions. The MS SFWE program is grounded in solid engineering practices and principles governed by the IEEE Software Engineering Body of Knowledge (IEEE SWEBOK). IEEE SWEBOK recommends specific skills that all software engineers should possess regardless of their programming languages and platforms. Our curriculum focuses on the application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software.

The MS Software Engineering program has a firm engineering foundation that encompasses discovery-based education utilizing an experiential learning approach. As a part of the curriculum, students will complete projects in areas that emphasize software engineering, communication, teamwork, critical thinking, and engineering professionalism. If the student opts to take the Thesis Option of the program, they will also conduct novel research in many diverse software engineering related areas. The program's flexibility allows students to design their course of study / research from a diverse pool of courses and research opportunities in software and computer engineering domains such as web and mobile applications, embedded systems, cybersecurity, machine learning, systems, and other interdisciplinary areas.

Learning Outcomes and Assessment Plan:

The learning outcomes are described in the table below:

Learning Outcome #1: Demonstrate broad knowledge in the student's field in software engineering.
Concepts: Students will study diverse topics in software engineering including software engineering fundamentals, software development and security operations (SW DevSecOps), software requirements analysis and test, software architecture and design, distributed computing. They also have the option to select from a broad range of software engineering and computing related technical electives. Courses may include computing topics such as data mining, cloud computing, software for industrial control systems, robotics and a variety of other computing topics that vary based on the electives the student opts to take.
Competencies: Students will demonstrate the ability to design, develop, test, integrate and evaluate varied software applications/products/systems in diverse computing and engineering domains. Students opting for the *Thesis-Option* to satisfy their course requirements may also apply their acquired knowledge in these areas to conduct original and novel research in state-of the-art and advanced software engineering principles, processes, and methodologies to meet the requirements/needs of diverse engineering applications.

Assessment Methods: For every new 5xx / 6xx SFWE course, a rubric will be created that identifies criteria/source of evidence, assessment measures, and an achievement level rating for specified course performance indicators used to measure this outcome. For each course that contributes to this outcome, specific student artifacts for a given course will be evaluated and assessed.

The sources of evidence can include class assignments, exams, projects, papers / reports and other forms of student work.

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 program/course. For existing courses (predominately technical electives), the evidence used to measure the effectiveness of the student outcome have been defined and will be followed. The rubric achievement levels will include: "Exemplary", "Satisfactory", "Developing", and "Unsatisfactory".

At the end of every semester, a team comprised of the course instructor and the SIE/ECE Graduate Studies Committee (GSC)s, will score the rubric using the assessment measures 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 as appropriate. 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.

Additionally, students are asked to complete a survey near the end of their MS program to self-evaluate how well they feel they met the Student Learning Outcomes for the MS program. In addition to the survey, the student is required to compose a brief description of how a project or projects completed during the course of their program serves to demonstrate their ability to analyze, design, and implement a SFWE-related system. The written description and survey are used to gather program-level assessment data.

A rubric is used to evaluate a student's written description. The completed survey and written description are returned to the SFWE Graduate Advisor, who archives this data. At the end of the spring semester, the surveys and written descriptions are provided to the SFWE Program Director and the appropriate Engineering department heads, who ensure the written descriptions are evaluated. The results of the surveys and descriptions will be organized and formatted over the summer. During the following academic year, the Graduate Studies Committee reviews the direct measures (written descriptions) and the indirect measures (student surveys) and then make recommendations for changes in response to the findings. Once each year, the summarized data, and proposed changes, if any, are reviewed and discussed by the SFWE faculty.

Measures: Rubrics will be used for the specific graded student projects in a given 5xx course that clearly evaluates the learning objectives and outcomes of the assignment and/or projects that students are asked to complete. All rubrics are developed by faculty members with expertise in software engineering domains. Faculty may consult with instructional designers as appropriate to ensure the course learning outcomes are measurable and contribute to the overall program learning outcomes. Exams are also used to assess and measure student learning. Throughout the degree program, the student's core course cumulative GPA will be used as a global measure of the overall student software engineering knowledge. Student surveys and program written description are also collected and evaluated to determine the overall program outcomes as students exit the program.
Learning Outcome #2 (Non-Thesis Option): Analyze the requirements, quality attributes and implementation of a software engineering system or subsystem.
Concepts: Students will derive, develop and analyze the requirements and quality attributes for a software product / system / subsystem. Additionally, students will also develop test plans and procedures that verify/validate the correct implementation of the requirements of the specified product. Students will also evaluate whether a software implementation meets its associated quality attributes and requirements as specified.
Competencies: Using industry best practices, students will learn to elicit, analyze, specify and validate functional and non-functional software requirements. Students will derive, develop and analyze the software requirements and quality attributes for a specific software product / system / subsystem. Students will also develop software test plans and procedures to validate that the implemented product meets its requirements. Students will also evaluate the implemented software to verify that it meets specified quality attributes and requirements.
Assessment Methods: This outcome will be assessed predominately through implementation of a semester project that meets a given or derived set of software requirements for a software / computing product. Midterm and final exams that test the students comprehension of the software engineering concepts taught and competencies acquired will also be used. Rubrics will be created that identify criteria/source of evidence, assessment measures, and an achievement level rating for specified course performance indicators used to measure this outcome. For each course that contributes to this outcome, specific student project artifacts for a given course will be evaluated and assessed. 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 program/course. For existing courses (predominately technical electives), the evidence used to measure the effectiveness of the student outcome have been defined and will be followed. The rubric achievement levels will include: "Exemplary", "Satisfactory", "Developing", and "Unsatisfactory". At the end of every semester, a team comprised of the course instructor and the SIE/ECE Graduate Studies Committee (GSC)s, will score the rubric using the assessment measures 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 as appropriate. 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.
Measures: Instructor grading of course exams and software projects for each course that contributes to this outcome. Rubrics will be used to measure the specific project evidence for each course. Descriptions that clearly measure the given evidence are used to assign the students achievement level. The achievement levels for the rubrics include: "Exemplary", Satisfactory", "Developing", and "Unsatisfactory".
Learning Outcome #2 (Thesis Option): Critically analyze published research results in the student's area of study.
Concepts: *Thesis-Option* students will learn various research techniques to interpret methods used and results from software related research papers, journals, presentations, and/or conferences. Throughout the program, students will have the opportunity to attend several seminars presented by a diverse group of researchers / scholars and faculty from a broad spectrum of software and computing related fields/areas. They will learn techniques used to critically read published research papers/journals, explore writing techniques used in technical/academic works, learn to develop evidence-based arguments, and draw conclusions from the sources being reviewed. They will also be provided numerous resources and learn to develop strategies for acquiring and using technical references from a variety of sources.
Competencies: Students will demonstrate their ability to read and interpret various forms of software research information, papers, conference proceedings, and data collected to support research. Students will also demonstrate their understanding of techniques used to write technical papers and journal articles. Students will also be able to analyze and explain research approaches taken and results included in published software engineering research papers, journals and conference proceedings.
Assessment Methods: This outcome will be assessed in the student's software engineering related research project and written thesis under the guidance of a faculty advisor. The thesis is reviewed by an examining committee consisting of at least 3 faculty (2 of which must be Graduate Faculty) from the SIE and ECE departments and chaired by the faculty advisor prior to the student conducting an oral presentation/defense of their thesis. The outcome will also be assessed in specific research-oriented assignments from SFWE 513 coursework.
Measures: This outcome will be measured by instructor grading of research related coursework in SFWE 513. It will also be assessed by the student's written thesis and oral defense using the collective results of a Program Assessment Survey completed by all thesis committee members.
Learning Outcome #3 (Non-Thesis Option): Design and implement a system in a software engineering area based on given specifications.
Concepts: Students will use industry best practices, methods, and tools in architecting, modeling, and designing software systems. Students also investigate and evaluate the importance of developing a sound software architecture and design as part of the evolution of the software implementation of a computing system/product.

	Working in teams on larger-scale semester projects, students implement their designs and show the traceability between requirements, software design, and the resulting code implementation for a specified system/product.
	Competencies: Using industry best practices, students will architect/design specified software products for diverse applications including mobile, cloud, embedded, or other computing applications. Students will collaborate with other students and software advisors to complete software projects using modern modeling tools and methodologies (i.e. UML or equivalent). Students will also implement/develop their software designs for a variety of computing-based products. Students will present/review their designs and implementations in a professional setting with other software engineering students, faculty and industry professionals.
	Assessment Methods: This outcome will be assessed predominately through implementation of a semester project that meets a given specification for a software / computing-based product. Midterm and final exams that test the students comprehension of the software engineering concepts taught and competencies acquired will also be used. Rubrics will be created that identify criteria/source of evidence, assessment measures, and an achievement level rating for specified course performance indicators used to measure this outcome. For each course that contributes to this outcome, specific student project artifacts for a given course will be evaluated and assessed. 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 program/course. For existing courses (predominately technical electives), the evidence used to measure the effectiveness of the student outcome have been defined and will be followed. The rubric achievement levels will include: “Exemplary”, “Satisfactory”, “Developing”, and “Unsatisfactory”. At the end of every semester, a team comprised of the course instructor and the SIE/ECE Graduate Studies Committee (GSC)s, will score the rubric using the assessment measures 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 as appropriate. 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.
	Measures: Instructor grading of course exams and computing projects for each course that contributes to this outcome. Rubrics will be used to measure the specific project evidence for each course. Descriptions that clearly measure the given evidence are used to assign the students achievement level. The achievement levels for the rubrics include: “Exemplary”, Satisfactory”, “Developing”, and “Unsatisfactory”.
Learning Outcome #3 (Thesis Option): Conduct original research on a significant software engineering problem.	
	Concepts: *Thesis-Option* students will utilize their acquired knowledge and new research strategies to conduct novel research in a software engineering field of specialization. Students will meet regularly with their faculty advisor and others within the ECE / SIE departments to receive guidance and coaching in a variety of research areas. Students will

	summarize all aspects of their research and their findings in a written thesis that will be defended orally before a faculty thesis committee.
	Competencies: Students will demonstrate their ability to articulate all aspects of their research in a software engineering specialization area, describe and defend the significance of their research, describe methodologies used in conducting the research, and summarize their overall findings resulting from said research.
	Assessment Methods: A thesis committee will assess the originality, merit, and contributions of the candidate's research. The written thesis and oral defense is facilitated by a faculty committee consisting of at least 3 faculty (2 of which must be graduate faculty) from the SIE and ECE departments and chaired by the faculty advisor. All members of the thesis committee will be asked to complete a Program Assessment Survey, and the results will be culminated together to form an overall assessment of the student's thesis.
	Measures: Evaluation of the student's final written thesis. The thesis will be evaluated by a faculty led committee that assesses the originality, merit, and contributions of the candidate's research. This includes their ability to (a) identify and critically evaluate relevant literature, (b) formulate and solve original problems using software engineering theory and methods, and (c) interpret and communicate research ideas, data and findings.
Learning Outcome #4: Communicate and defend (written and oral) results of projects or research to peers and broader engineering audiences.	
	Concepts: Students will utilize their acquired software engineering skills and knowledge to communicate effectively in both written and oral mediums. This may be accomplished in a variety of methods including presenting the results of software / computing related projects to peers, faculty, and potentially industry experts. Additionally, students that opt for the *Thesis-Option* will prepare a written thesis that demonstrates all aspects of their research including the significance of their work, a detailed review of relevant literature, methodologies employed and/or developed, significant findings from the work, a critical discussion of the findings, limitations, and the impact, and potential for future research. *Thesis* students will also be required to present their research findings in an oral defense of their project / research.
	Competencies: Students will demonstrate their ability to articulate all aspects of the product development and/or research in a software engineering specialization area. Students opting for the course work only option of the degree program, will describe the relevant design or implementation details of projects that they have developed/implemented, describe methodologies used in developing the software products, and demonstrate designs and/or working products. Students opting for the *Thesis-Option* of the degree program will also describe and defend the significance of their projects / research in a thesis, describe the methodologies used in conducting the software related research, and summarize their overall findings resulting from said research in written and oral mediums.
	Assessment Methods: For all SFWE students, specific project rubrics will be used for the specific graded student project for a given course that clearly evaluates the learning objectives and outcomes of the projects that students

are asked to complete. All rubrics are developed by faculty members with expertise in associated software engineering domains. Faculty may consult with instructional designers as appropriate to ensure the course learning outcomes are measurable and contribute to the overall program learning outcomes. Additionally, for *Thesis-Option* students, a thesis committee will assess the originality, merit, and contributions of the candidate's research. The written thesis and oral defense is facilitated by a faculty committee appointed consisting of at a minimum 3 faculty from the SIE and/or ECE departments and chaired by the faculty advisor.
Measures: For all students, rubrics will be used for specific graded student projects in a given 5xx course that clearly evaluates the learning objectives and outcomes of the assignment and/or projects that students are asked to complete. All rubrics are developed by faculty members with expertise in software engineering domains. Faculty may consult with instructional designers as appropriate to ensure the course learning outcomes are measurable and contribute to the overall program learning outcomes. Exams are also used to assess and measure student learning. Throughout the degree program, the student's core course cumulative GPA will be used as a global measure of the overall student software engineering knowledge. For *Thesis-Option* students, evaluation of the student's final written and oral thesis. The thesis will be evaluated by a faculty led committee that assesses the originality, merit, and contributions of the candidate's research. This includes their ability to (a) identify and critically evaluate relevant literature, (b) formulate and solve original problems using software engineering theory and methods, and (c) interpret and communicate research ideas and findings.

The Taskstream Curriculum Map is shown below. Note that the assessment plan includes only new SFWE courses that are part of this program. Existing technical elective courses used to fulfill the degree requirements are not included. The assessment plans for existing courses will be followed per the home department's assessment plan for each respective existing course.

MS Software Engineering

Courses and Activities Mapped to MS Software Engineering

	Outcome					
	SLO 1 Demonstrate broad knowledge in his/her field in software engineering.	SLO 2: Non-Thesis Option Analyze the requirements, quality attributes, and implementation of a software engineering system or subsystem.	SLO 2:Thesis Option Critically analyze published research results in his/her area of study.	SLO 3: Non-Thesis Option Design and implement a system in a software engineering area based on given specifications.	SLO 3: Thesis Option Conduct original research on a significant software engineering problem.	SLO 4 Communicate and defend (written and oral) results of projects or research to peers and broader engineering audiences.
Courses and Learning Activities						
SFWE 507 Foundations of Software Engineering	IPA	I		I		IPA
SFWE 502 Software DevSecOps	P/A			P/A		P/A
SFWE 503 Software Project Management	P/A					IPA
SFWE 504 Software Requirements Analysis and Test		P/A		P/A		P/A
SFWE 505 Software Architecture and Design	P/A			P/A		P/A
SFWE 506 Distributed Computing		P/A		P/A		P/A
SFWE 508 Data Mining	P/A	P/A				P/A
SFWE 509 Cloud Computing Principles and Practices	P/A	P/A		P/A		P/A
SFWE 510 Cloud Native Software Engineering	P/A	P/A		P/A		P/A
SFWE 511 Software for Industrial Control Systems		P/A		P/A		P/A
SFWE 512 Robotics		P/A		P/A		
SFWE 513 Software Engineering Research Methods			IPA		IPA	P/A
SFWE 910 Thesis		P/A	IPA		IPA	P/A
Exit Survey Exit survey (Indirect)	A	A	A	A	A	A

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

MS Thesis Students

The Software Engineering (SFWE) MS Thesis Option program requires a student complete an oral defense of their thesis. The student also completes a survey near the end of their MS program to self-evaluate how well they feel they met the Student Learning Outcomes for the MS SFWE program. The thesis defense and survey are used to gather program-level assessment data.

A rubric to evaluate a student's thesis work is developed by the SFWE faculty. Each faculty committee member completes the rubric at the end of the oral thesis defense. These completed rubrics are returned to the SFWE Graduate Advisor, who archives this data. At the end of the Spring semester, the completed rubrics and surveys are provided to the SFWE Program Director and appropriate Engineering department heads, who will organize and format the results during the summer. During the following academic year, the Graduate Studies Committee (GSC) reviews the direct measures (thesis rubrics) and the indirect measures (student surveys) and makes recommendations for changes in response to the findings. Once each year, the summarized data, and proposed changes, if any, are reviewed and discussed by the SFWE graduate faculty.

MS Non-Thesis Students

The Software Engineering (SFWE) MS Non-Thesis Option program requires a student complete 30 units of graduate coursework. This coursework has limitations as described in the SFWE Graduate Handbook. The student completes a survey near the end of their MS program to self-evaluate how well they feel they met the Student Learning Outcomes for the MS SFWE program. In addition to the survey, the student is required to compose a brief description of how a project or projects completed during the course of their program serves to demonstrate their ability to analyze, design, and implement a SFWE-related system. The written description and survey are used to gather program-level assessment data.

A rubric is used to evaluate a student's written description. The completed survey and written description are returned to the SFWE Graduate Advisor, who archives this data. At the end of the spring semester, the surveys and written descriptions are provided to the SFWE Program Director and the appropriate Engineering department heads, who ensure the written descriptions are evaluated. The results of the surveys and descriptions will be organized and formatted over the summer. During the following academic year, the Graduate Studies Committee reviews the direct measures (written descriptions) and the indirect measures (student surveys) and then make recommendations for changes in response to the findings. Once each year, the summarized data, and proposed changes, if any, are reviewed and discussed by the SFWE faculty.

Assessment Measure	Source(s) of Evidence	Data Collection Point(s)
Rubrics for all new courses used to assess each student outcome that identifies criteria, measure of assessment, and an achievement level rating *(i.e., Exemplary, Satisfactory, Developing, Unsatisfactory).*	Specifically targeted: • Class assignments • Exams • Course Projects • Course Reports • Other forms of student work tailored to any specific course)	End of each semester the specific courses are taught.
MS final written thesis *(Thesis-Option only)*	Written thesis	Written and evaluated at the conclusion of the student's thesis efforts to assess the merit and contributions of the student's research and findings.
Thesis oral presentation / defense (*Thesis-Option* only)	Oral presentation / defense of the student's thesis	At the completion of the thesis, the student will present their work to a faculty thesis committee and answer any general questions related to their work
Graduation exit survey (used for indirect measures of outcomes).	Student survey	At student graduation
Academic Program Review	APR reviewer evaluation responses	Minimum of every 7 years
Data from continuous improvement implementation efforts (recommended by the Graduate Studies Committee (GSC))	Assessment data	At the end of an academic year

Projected Enrollment for the First Three Years:

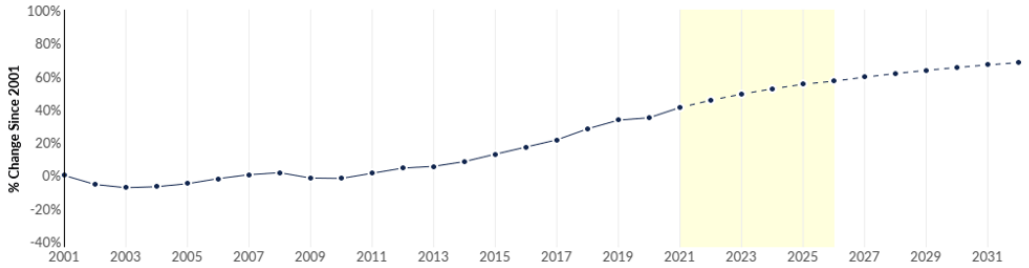
Please provide anticipated enrollment numbers for each of the first three years of the proposed program.

Degree	Year 1 (2023 / 2024)	Year 2 (2024 / 2025)	Year 3 (2025 / 2026)
MS	20	50	100

Evidence of Market Demand:

The market demand for those trained in engineering computing /software engineering disciplines (CIP code 14.093 - Computer Software Engineering) is projected to have significant growth in both the near-term and long-term future. As shown in the diagram below, sourced from Lightcast Q3 2022 data[5], the number of regional[6] jobs in the workforce with a MS or PhD or other professional degree is expected to see a 11.3% increase over 5 years, between 2021 - 2026.

Regional Trends



Region	2021 Jobs	2026 Jobs	Change	% Change
• Region	3,074,978	3,421,626	346,648	11.3%

Thus, the new SFWE MS and PhD degree programs will serve both local, state, and national needs related to employment, economic development, and national security. Indeed, these degree programs are among the most important in support of the ongoing fourth industrial revolution and in close alignment with Arizona's New Economy Initiative[7].

[5] Lightcast Q3 2022 Data Set, www.economicmodeling.com

[6] Regional jobs include the states located in the southwest region of the United States

[7] World Economic Forum. https://www.weforum.org/agenda/2016/01/the-fourth-industrial-revolution-what-it-means-and-how-to-respond/

The full marketing and analysis report can be found at the following link:
https://arizona.box.com/s/g2sm18hc6gwxi5th7fch49vnau4etsrb

Similar Programs Offered at Arizona Public Universities:

University	Program	College
University of Arizona	BS Software Engineering	College of Engineering
University of Arizona	MS Computer Science	College of Science
Arizona State University	MS Software Engineering	School of Computing and Augmented Intelligence, IRA A Fulton Schools of Engineering
Arizona State University	MS Computer Science	School of Computing and Augmented Intelligence, IRA A Fulton Schools of Engineering
Northern Arizona University	MS Computer Science	School of Informatics, Computing, and Cyber Systems

FOR CURRICULAR AFFAIRS USE ONLY

Objection(s) Raised by Another Arizona Public University? YES NO

Has another Arizona public university lodged a written objection to the proposed program with the proposing university and the Board of Regents within seven days of receiving notice of the proposed program?

If Yes, Response to Objections:

Please provide details of how the proposing university has addressed the objection. If the objection remains unresolved, please explain why it is in the best interests of the university system and the state that the Board override it.

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

Resources	Quantity
Faculty	5
Staff	2

Other (TAs/RAs, Graders, LAs) *(Semester hires over 5 years)*	34 Teaching Assistants or Research Assistants (TAs / RAs) *(total # semester TA hires over 5 years, averaging 3.5 TAs or RAs /year)* 0 Graders 0 LAs
Equipment	None
Facilities	Office and lab space *(for new faculty)*

Plan to Request Program Fee/Differentiated Tuition? No

Estimated Amount: N/A

Program Fee Justification:

Note: The fee setting process requires additional steps and forms that need to be completed. Please work with your University Fees office to complete a fee request.

N/A

Specialized Accreditation? No

Accreditor: N/A

Graduate Major Peer Comparison Chart-select two peers for completing the comparison chart from (in order of priority) ABOR-approved institutions, AAU members, and/or other relevant institutions recognized in the field. The comparison chart will be used to identify typically required coursework, themes, and experiences for majors within the discipline. The comparison programs are not required to have the same degree type and/or major name as the proposed UA program. Information for the proposed UA program must be consistent throughout the proposal documents.

Program name, emphasis (sub-plan) name (if applicable), degree, and institution	**Proposed UA Program:** **MS Software Engineering**	**Peer 1:** **MS Software Engineering Arizona State University (ASU)** ASU MS SW Engr	**Peer 2:** **MS Software Engineering Carnegie Mellon University** CMU MS SW Engr
Current # of enrolled students		148[1]	Not available *(117 MS degrees awards in 2019-2020)*[2]
Major Description. Includes the purpose, nature, and highlights of the curriculum, faculty expertise, emphases (sub-plans; if any), etc.	The MS Software Engineering curriculum applies software engineering fundamentals to develop and produce computing-based products/solutions. The MS SFWE program is grounded in solid engineering practices and principles governed by the IEEE Software Engineering Body of Knowledge (IEEE SWEBOK). IEEE SWEBOK recommends specific skills that all software engineers should possess regardless of their programming languages and platforms. Our curriculum focuses on the application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software. The program has a firm engineering foundation that encompasses discovery-based education utilizing an experiential learning approach. As a part of the curriculum, students will complete projects	The Master of Science in Software Engineering degree program focuses on developing advanced knowledge and abilities in the design and application of software. Students will learn to apply engineering principles to software development, including design methodologies, operation principles and maintenance and testing approaches. The program involves the application of engineering principles to software development including design methodologies, operation principles, and maintenance and testing approaches. The MS in software engineering degree program is offered on the Polytechnic campus. The MS Software Engineering program builds upon the Bachelor of Science in Software Engineering program and is	The Master of Science in Software Engineering is a unique program offered exclusively at CMU's Silicon Valley campus. It emphasizes a rigorous foundation in the core disciplines of software engineering. The program offers students fundamental knowledge, skills, and first-hand experience in software engineering by balancing theory and practice, engaging students in active learning, and encouraging collaboration on projects drawn from real-world contexts. Our students enter the program with a strong foundation in computer science. They leave the program with a deep knowledge of software engineering.

[1] Enrollments derived from https://shinyapps.asee.org/apps/Profiles/

[2] Carnegie Mellon University MS in Software Engineering - College Factual

	in areas that emphasize software engineering, communication, teamwork, critical thinking, and engineering professionalism. The program's flexibility allows students to design their course of study / research from a diverse pool of courses and research opportunities in software and computer engineering domains such as web and mobile applications, embedded systems, cybersecurity, machine learning, systems, and other interdisciplinary areas.	aimed at developing professional skills in this discipline as well as providing opportunities for students to engage in and develop research abilities.	
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 Project Management / Leadership	• Software developer • Software Engineer for variety of application areas: - Web - Mobile - Embedded systems - Avionics - Robotics - Other software related fields • Software Quality Assurance • Software Project Management / Leadership	• Software developer • Software Engineer for variety of application areas: - Web - Mobile - Embedded systems - Avionics - Robotics - Other software related fields • Software Quality Assurance • Software Project Management / Leadership
Total units required to complete the degree	30	30	97 *(Note: Most CMU courses are 12 units)*
Pre-admission expectations (i.e. academic training to be completed prior to admission)	Bachelor's degree from an institution recognized by the UA. Students who do not have a degree equivalent to the UA Bachelor of Science degree in a computing related program may be admitted into the graduate SFWE program but may be required to complete additional graduate-level pre-requisite courses prior to enrolling in some graduate courses. • Grade-point average greater than 3.0 overall (or in the last 60 units). • Applicants whose native language is not English are required by the Graduate College	• Minimum GPA = 3.0 in last 60 credit hours of undergraduate degree program • Must have at least Calculus 1 & 2 • Deficiency courses differ depending on undergraduate degree (cannot have more than 2 deficiencies) • GRE scores of 146 verbal, 155 quantitative, and 3.0 analytical writing • English proficiency – TOEFL score of 575	• Qualified students that have earned an undergraduate degree in a relevant field. • GRE optional • Minimum DET composite score of 105 for international students

	to take an English proficiency test. A description of acceptable tests can be found on the Graduate College admissions website. • Students in the MS non-thesis option are expected to be self-supported or supported by external fellowships or industry.		
Major requirements. List all major requirements including core and electives. If applicable, list the emphasis requirements. 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.	**Complete 12 units of core coursework (select 4 from the following):** • SFWE 501 (3) Foundations of Software Engineering • SFWE 502 (3) Software DevSecOps • SFWE 503 (3) Software Project Management • SFWE 504 (3) Software Requirements Analysis and Test • SFWE 505 (3) Software Architecture and Design • SFWE 506 – Distributed and Parallel Processing **Complete one of the following options:** ***Thesis Option:*** • Complete 12 units pre-approved technical computing electives or in a closely related computing field (must be approved by graduate studies committee*).* • Complete 6 units of thesis (SFWE 910) ***Non-Thesis Option:*** • Complete 18 units pre-approved technical computing electives or in a closely related computing field (must be approved by graduate studies committee*).*	***Required Core Coursework (9 units):*** • SER 501 (3) Advanced Data Structures and Algorithm Analysis • SER 502 (3) Emerging Languages and Programming Paradigms • SER 515 (3) Foundations of Software Engineering ***Software Engineering Elective Courses (6 units):*** • SER516 (3) Software Agility • SER574 (3) Advanced Software Design • CSE563 (3) Software Requirements and Specification • CSE564 (3) Software Design • CSE565 (3) Software Verification, Validation, and Testing • CSE566 (3) Software Project, Process, and Quality Management ***General Degree (Free Electives) (9-12 units in approved coursework)*** ***Culminating Experience (3 units)*** • Capstone Option: SER 517 Software Factory I (3) or • Thesis Option: SER 599 Thesis (6) over two semesters	**Core Requirements** *(each class is 12 units except for the 1-unit Introduction to Graduate Studies)* ***60 units of Core Coursework:*** • 18-652 Foundations of Software Engineering ***(Required)*** ***(Select 4 additional Core from the following):*** • 18-653 Software Architecture and Design • 18-654 Software Verification and Testing • 18-657 Decision Analysis and Engineering Economics for SW Engr • 18-658 Software Requirements and Interaction Design • 18-659 Software Engineering Methods • 18-664 Software Refactoring • 18-668 Data Science for Software Engineering ***1 Unit:*** • Introduction to Graduate Studies ***36 Units of Elective Courses***
Research methods, data analysis, and methodology	If the student selects the Thesis Option, they are required to complete 6 units of research and	If the student selects the Thesis Option, they are required to complete 6 units of research	Not required

requirements (Yes/No). If yes, provide description.	thesis development in an area related to Software Engineering.	and thesis development in an area related to Software Engineering.	
Internship, practicum, applied course requirements (Yes/No). If yes, provide description.	None	Required if the student selects the Capstone Option as noted above.	Students may take up to 27 project units. 12 project units can count towards their core requirements and up to 15 project units can count toward their elective requirements.
Master thesis or dissertation required (Yes/No). If yes, provide description.	If the student selects the *Thesis Option*, they are required to complete 6 units of research in an area related to software engineering. The thesis will capture the results of the research that was completed under the guidance of a faculty member. The thesis paper should demonstrate the students core knowledge, technical skills, and the ability to articulate and synthesize the findings of a project they were engaged in over multiple semesters. Students must pass an oral defense of their thesis.	If the student selects the *Thesis Option*, they are required to complete 6 units of research in an area related to software engineering. The thesis will capture the results of the research that was completed under the guidance of a faculty member. The thesis paper should demonstrate the students core knowledge, technical skills, and the ability to articulate and synthesize the findings of a project they were engaged in over multiple semesters. Students must pass an oral defense of their thesis.	No
Additional requirements (provide description)	Students must maintain a GPA of 3.0 or better.	The ASU Graduate College requires that all graduate students achieve a grade of B or better in the culminating experience; students not achieving this grade will need to repeat the culminating experience.	• 85 units must be from the College of Engineering

*Note: comparison of additional relevant programs may be requested.

THE UNIVERSITY OF ARIZONA

BUDGET PROJECTION FORM

Name of Proposed Program or Unit: Software Engineering MS

Budget Contact Person:	Projected 1st Year 2023- 2024	2nd Year 2024 - 2025	3rd Year 2025- 2026	4th Year 2026- 2027	5th Year 2027- 2028
METRICS					
Net increase in annual college enrollment UG					
Net increase in college SCH UG					
Net increase in annual college enrollment Grad On campus	12	30	50	80	100
Net increase in college SCH Grad On campus - 18 units/yr	216	540	900	1,440	1,800
Net increase in annual college enrollment Grad Online	8	20	50	70	100
Net increase in college SCH Grad Online - 18 units/yr	144	360	900	1,260	1,800
Number of enrollments being charged a Program Fee					
New Sponsored Activity (MTDC)					
Number of Faculty FTE					
Sharon: Doubled the enrollment in MS program per Larry Head recommendation					
FUNDING SOURCES					
Continuing Sources					
UG AIB Revenue - enrollment					
UG AIB On Campus Degree					
UG SCH					
Grad AIB Revenue SCH On campus- Used average of 368 per SCH	79,488	198,720	331,200	529,920	662,400
Grad AIB Revenue enrollment On campus-Used average of 1009 per enrollment	12,108	30,270	50,450	80,720	100,900
Grad AIB Revenue SCH Online-Used average of 675 per SCH	97,200	243,000	607,500	850,500	1,215,000
Grad AIB Revenue enrollment Online-Used average of 1109 per enrollment	8,872	22,180	55,450	77,630	110,900
Program Fee Revenue (net of revenue sharing)					
F and A AIB Revenues		106,666	160,000	106,666	53,333
Reallocation from existing College funds (attach description)					
Other Items (attach description)					
Total Continuing	**$ 197,668**	**$ 600,836**	**$ 1,204,600**	**$ 1,645,436**	**$ 2,142,533**
One-time Sources					
College fund balances					
Institutional Strategic Investment					
Gift Funding					
Other Items (attach description)					
Total One-time	**$ -**	**$ -**	**$ -**	**$ -**	**$ -**
TOTAL SOURCES	**$ 197,668**	**$ 600,836**	**$ 1,204,600**	**$ 1,645,436**	**$ 2,142,533**
EXPENDITURE ITEMS					
Continuing Expenditures					
Faculty	130,000	263,250	399,831	409,827	420,073
Other Personnel	140,000	283,500	290,588	297,852	305,298
Employee Related Expense	86,130	174,413	220,244	225,750	231,393
Graduate Assistantships	40,160	60,240	80,320	80,320	80,320
Other Graduate Aid	29,917	44,875	59,834	59,834	59,834
Operations (materials, supplies, phones, etc.) -Faculty recruitment, student travel, s	27,500	27,500	27,500	20,000	20,000
Additional Space Cost					
Other Items (attach description)					
Total Continuing	**$ 453,707**	**$ 853,778**	**$ 1,078,316**	**$ 1,093,582**	**$ 1,116,918**
One-time Expenditures					
Construction or Renovation					
Start-up Equipment	133,333	266,666	399,999	266,666	133,333
Replace Equipment					
Library Resources					
Other Items (attach description)					
Total One-time	**$ 133,333**	**$ 266,666**	**$ 399,999**	**$ 266,666**	**$ 133,333**
TOTAL EXPENDITURES	**$ 587,040**	**$ 1,120,444**	**$ 1,478,315**	**$ 1,360,248**	**$ 1,250,251**
Net Projected Fiscal Effect	**$ (389,372)**	**$ (519,608)**	**$ (273,715)**	**$ 285,188**	**$ 892,282**

THE UNIVERSITY OF ARIZONA
Online, Distance & Continuing Education

October 2022

Prof. Sharon ONeal
Professor and Director, Software Engineering
College of Engineering
University of Arizona

Re: MS and PhD Software Engineering online and distance degree programs – Letter of support

Dear Prof. ONeal,

On behalf of the University of Arizona's Online, Distance and Continuing Education (ODCE) division, I am pleased to offer this letter of support for your proposal to offer master's and doctorate degrees in Software Engineering to online and distance students. Increased access to this critical field will provide our students, many of whom are adults with family and job responsibilities and cannot travel to Tucson, a great opportunity to achieve their educational goals.

As the university's in-house full-service enterprise for online and distance education we look forward to collaborating with you. Here is a snapshot of our services and support:

- Online curriculum planning and program development;
- Online course design;
- Marketing, student recruitment, and enrollment management; and,
- Student success coaching for increased retention and graduation.

Our ODCE team will work with you in efforts to help increase diversity, accessibility, and degree completion for students enrolled in these innovative STEM programs.

If you require more information about our support for this proposal, please feel free to contact Caleb Simmons (calebsimmons@arizona.edu), executive director for online education; and/or, Carla Holloway (carlaholloway@arizona.edu), executive director for distance education.

Sincerely,

Craig Wilson, JD, PhD
Vice Provost, Online, Distance and Continuing Education
Professor of Practice, College of Education

THE UNIVERSITY OF ARIZONA

NEW ACADEMIC PROGRAM – MAJOR
Supplemental Info Form

NOTE: This is being added to the proposal post Graduate Programs Executive Review Committee (GPERC) viewed and commented on the document. Below are their questions and the responses given by the proposing department/college.

1. *An internship or practicum requirement would strengthen a software engineering program. I have hired a good number of student SE (from CS, IS, and Eller) for my research projects and found most of them lack practice in real software development settings.*

I did speak to a few faculty members about the recommendation to include an internship or practicum requirement and they felt that was something we could consider in the future, but not make it a requirement for the curriculum at this point. There are several reasons for this – one being that we are still building our relationships with industry and making this a requirement at this point is something we do not feel comfortable with. Certainly, as the program grows and we establish ourselves, it could be an "option" that we incorporate at a later time.

2. *They proposed a new course 'Data Mining'. As far as I know, there is a DM course in MIS and another one in INFO. Not sure how the SE version of data mining is different from the existing ones.*

I looked at MIS 545 Data Mining for Business Intelligence. This course covers data mining for business intelligence. Data mining refers to extracting or "mining" knowledge from large amounts of data. It consists of several techniques that aim at discovering rich and interesting patterns that can bring value or "business intelligence" to organizations.

The INFO 523 Data Mining and Discovery introduces students to the concepts and techniques of data mining for knowledge discovery. It includes methods developed in the fields of statistics, large-scale data analytics, machine learning, pattern recognition, database technology and artificial intelligence for automatic or semi-automatic analysis of large quantities of data to extract previously unknown interesting patterns.

The SFWE 508 Data Mining course will cover basic principles and strategies used in data mining but will focus on how that data can be used in the development (implementation and test) of sophisticated/complex software products.

The focus areas are very different between the 3 courses.