

School of Computer Science
Software and Societal Systems Department

Doctoral Student Handbook

Degree Programs Covered by This Handbook:
Software Engineering
Software Engineering Joint Portugal Program

Table of Contents

SECTION 1: Welcome & Introduction.....	3
SECTION 2: Program Vision, Mission, and Values.....	4
2.1: Vision.....	4
2.2: Mission.....	4
2.3: Values.....	4
2.4: Research Outcomes.....	5
2.5: PhD Community.....	6
SECTION 3: Degrees Offered.....	7
SECTION 4: Departmental Personnel.....	7
SECTION 5: Departmental Resources.....	9
SECTION 6: Advising.....	10
6.1: Role of an Advisor and Advisor Assignments.....	10
6.2: Advisor/Advisee Collaboration.....	11
6.3: Review of Academic Conflicts.....	11
SECTION 7: Doctoral Degree Requirements.....	12
7.1: Residency Requirements.....	12
7.2: Expected Timeline Estimates.....	12
7.3: Registration Process.....	13
7.4: Required Units for Degree Attainment.....	14
7.5 Core Courses.....	14
7.6 STAR Courses.....	15
7.7: Electives.....	18
7.8: Software Engineering Research Seminar.....	18
7.9: Written Communications Skills.....	19
7.10: Speaking Proficiency.....	20
7.11: Practicum Requirement.....	21
7.12: Specific Requirements for the CMU Portugal Program.....	22
7.13: Course Exemptions.....	24
7.14: Protocol for Evaluation of Transfer Credit.....	25
7.15: Protocol for Evaluation of Course Waiver.....	25
7.16: Teaching Requirements/Opportunities.....	26
7.17: Research Requirements/Opportunities.....	29
7.18: Internships.....	29
SECTION 8: Dissertation Preparation & Requirements.....	30
8.1: PhD Criteria for Advancement to Candidacy.....	30
8.2: Ph.D. Dissertation.....	31

8.3: Thesis Committee.....	32
8.4: Proposal.....	32
8.5: Proposal Scheduling.....	33
8.6: PhD Defense.....	34
8.6: Defense Scheduling.....	36
SECTION 9: Department Policies & Protocols.....	37
9.1: Petition Procedures- for adding a course to current STAR list.....	37
9.2: Department Policy for Withdrawing from a Course.....	38
9.3: Master's Degree en route to a PhD Process.....	39
9.4: Time Away from Academic Responsibilities.....	39
SECTION 10: Grading & Evaluation.....	40
10.1: Grading Scale/System.....	40
10.2: Department Policy on Pass/Fail, Satisfactory/Unsatisfactory.....	40
10.3: Independent Study/Directed Research.....	40
10.4: Doctoral Student Review (DSR).....	41
10.5: Satisfactory Academic Standing.....	42
SECTION 11: Funding & Financial Support.....	43
11.1: Health Insurance Requirement.....	43
11.2: Additional Sources of Internal & External Financial Support.....	43
11.3: Department Policy on Outside Employment.....	43
SECTION 12: Graduate Student Handbook.....	44

SECTION 1: Welcome & Introduction

Welcome to the Software Engineering Ph.D. Program!

Software Engineering is the branch of computer science that creates practical, cost-effective solutions to computing and information processing problems, preferentially by applying scientific knowledge, developing software systems in the service of mankind.¹ Carnegie Mellon's Software Engineering Ph.D. program produces graduates who are well-prepared for faculty positions in software engineering, for research positions in industrial laboratories, and for leadership positions in development within the software field. The Ph.D. degree is a certification by the faculty that the student has a broad education in Software Engineering and has performed a substantial piece of original research in the area.

This document is an informal description of the Software Engineering Ph.D. program; herein "we" refers to all the faculty and staff involved in the Ph.D. program. Currently, the Department head is Nicolas Christin nicolasc@andrew.cmu.edu, the Director of Software Engineering Ph.D. program is Christian Kästner ckaestne@andrew.cmu.edu, and the Ph.D. Program Managers are Connie Herold cherold@andrew.cmu.edu, and Alisha Roudebush aroudebu@andrew.cmu.edu.

To complete the Ph.D. degree, we require that each student:

- Contribute to scientific knowledge in software engineering by engaging in *directed research*. This is the central element of the Ph.D. program, and students should spend at least half their time on research from the first semester onward.
- Develop a broad foundation in software engineering and specific intellectual skills by passing 84 university units worth of *graduate courses*, with certain distribution requirements
- Acquire and demonstrate teaching skills by serving as a *teaching assistant* at least twice
- Acquire and demonstrate oral and written *communication skills* by writing about research and participating in S3D's Software Research Seminar
- Demonstrate, through an issue-focused oral presentation and written *practicum report*, an understanding of software engineering that is grounded in practice.
- Write and orally defend a *thesis*, a significant piece of original research related to software engineering. To support effective planning and ensure that the student's thesis topic is viable, we require them to present a *thesis proposal* in advance.

¹ Mary Shaw (editor). *Software Engineering for the 21st Century: A Basis for Rethinking the Curriculum*. CMU-ISRI-05-108. Carnegie Mellon University, 2005

SECTION 2: Program Vision, Mission, and Values

2.1: Vision

The Software Engineering program at Carnegie Mellon University will have a transformative impact on society through continual innovation in education, research, creativity, and entrepreneurship.

2.2: Mission

To create a transformative educational experience for students focused on deep disciplinary knowledge; problem solving; leadership, communication, and interpersonal skills; and personal health and well-being.

To cultivate a transformative university community committed to (a) attracting and retaining diverse, world-class talent; (b) creating a collaborative environment open to the free exchange of ideas, where research, creativity, innovation, and entrepreneurship can flourish; and (c) ensuring individuals can achieve their full potential.

To impact society in a transformative way — regionally, nationally, and globally — by engaging with partners outside the traditional borders of the university campus.

2.3: Values

A distinguishing characteristic of our program is that we encourage and expect students to engage in research from their first day. The program is also noted for the unique background of its student community. Many of our students bring significant prior experience in industry that we value, and which they can leverage in their research—while other students may not have this experience themselves but benefit from those who do. Our community in the Software and Societal Systems department is also part of the larger community in the School of Computer Science and in the university at large. To help students fulfill the requirements of our program, we provide these educational opportunities:

- An active research environment, with experienced and dedicated faculty advisors
- Many graduate courses covering various topics within software engineering and related areas in computer science.

The entire faculty meet twice a year to evaluate each student's progress. A student demonstrates progress by passing courses, doing directed research, teaching, fulfilling the skills and practicum requirements, and doing thesis work. Because each student's path is different, the order in which students complete the various aspects of the program will vary, though direct research progress is expected every semester. A timeline showing how a typical student in our program might go through the program elements is available in

2.4: Research Outcomes

A Ph.D. is a research degree. Accordingly, its desired outcomes focus primarily on capabilities in research and education, together with expectations of capabilities in the subject area of software design and development; we therefore assume some level of the subject area capability in our entering students.

This differs from undergraduate and professional master's programs, which focus on proficiency in software design and development. In particular, we expect that the following will be outcomes of the Ph.D. program:

- **Ability to do independent research.** SE Ph.D. 's must have the ability to carry out independent research – to select significant practical problems, solve them in creative ways, evaluate them critically, demonstrate the validity of the solution, and gather the resources to carry out the work. This is the absolutely essential capability of a Ph.D.
- **Skill in several research methods.** SE Ph.D. 's will have broad knowledge with the research methods of the field, empirical and formal (symbolic) methods, together with the ability to evaluate the application of a research method and to select the appropriate research method for a specific research project.
- **Depth of knowledge in chosen specialty.** SE Ph.D.'s are deeply knowledgeable in their specialties. Within their specialties they can evaluate and critique material. They exercise this knowledge both within the discipline and in the public realm.
- **Broad general knowledge of SE.** SE Ph.D.'s are broadly knowledgeable in their field. They have software design and development skills, and they are familiar with issues in computer science at large. They exercise this knowledge both within the discipline and in the public realm, and they seek relevant knowledge from other fields.
- **Ability to teach a range of software courses.** SE Ph.D.'s will be technical leaders. As such, they will be able to organize a body of knowledge so it can be taught to others and should be able to plan presentations and other activities to teach that material. This requires communication with non-experts as well as experts.
- **Communication skills.** SE Ph.D. 's will be able to communicate effectively about technical material both within and outside of their specialization, both to other researchers and also to policy makers and the public.
- **Deep understanding of practical software issues.** SE Ph.D.'s will ground their research in a deep understanding of software engineering practice. In addition to basic software design and development skills, they will have an in-depth understanding, drawn from personal observation, of practical software engineering issues. These include the implications of development at scale, the gnarly engineering tradeoffs and conflicts that arise in practice, and the tangle of technical, business, and often policy issues that are imposed by project context.
- **Broad, mature, multidisciplinary perspective.** SE Ph.D.'s will be prepared for

interdisciplinary collaboration and professional leadership. This depends on their ability to view SE critically; to understand how software-intensive systems interact with larger issues in society, business, socio-economic impacts, and public policy; and to appreciate the perspective of both collaborators and competitors.

2.5: PhD Community

We are committed to a strong sense of community within the S3D as well as the School of Computer Science as a whole. Our community is one of the reasons many students choose to come here. We foster community spirit through close working relationships between students and advisors, among faculty, and among students. Many working relationships turn into friendships for life.

In practice, our students, faculty, and staff volunteer their time, energy, intellect, talent, and other skills to do many of the things that keep our environment running smoothly. These efforts include organizing seminars, serving on departmental committees, grading for graduate courses, planning, and running social activities, giving tours, and hosting visitors. Our Software Engineering Ph.D. students have an impressive record of volunteer leadership not just within the program, but also at SCS and university levels.

Curriculum Representative

An important part of our culture is that students have a voice and a vote in decisions about the Ph.D. community. In general, decisions regarding Ph.D. program policies are made by a Software Engineering Faculty Meetings. The committee is composed of software engineering faculty as well as two Ph.D. student representatives who serve staggered 2-year terms. Decisions in the committee are nearly always made by consensus, but the student representatives are voting members of the committee when consensus is unclear.

Mutual Respect

An essential aspect of our culture is mutual respect among students, faculty, and staff that are highly diverse, not only in terms of professional and research interests, but also in terms of gender, national origin, religion, sexual orientation, and other demographic characteristics. Words or actions that express discrimination, disrespect, intimidation, or harassment based on race, color, national origin, birth sex, gender identity, handicap or disability, age, sexual orientation, religion, creed, ancestry, belief, veteran status, or genetic information are not acceptable within our community. Any violations of these standards should be brought to the Ph.D. program director, Christian Kästner, Ph.D. Coordinator, Connie Herold, the ombudsperson for the program or the Office of Student Affairs.

Incidents of Discrimination or Harassment

Our community places great value on mutual respect. However, if you witness an incident of discrimination based on gender, race, religion, or similar characteristics; it is important to know what to do about it.

If you feel comfortable and safe responding to the discriminatory behavior in the moment, we encourage you to do that; such responses, especially from bystanders, reinforce our

community standards and can start an educational dialog that leads to reconciliation and prevent similar incidents from occurring in the future.

We encourage you to talk with the Ph.D. program director, any faculty member, the student ombudsperson(s), the Ph.D. program coordinator, or someone in the Office of Student Affairs about any incidents you witness or hear about secondhand. Faculty members, student ombudspersons, and the Ph.D. program coordinator are responsible for forwarding such incidents to the Ph.D. program director (or, when the director has a conflict, to the faculty ombudsperson) for follow-up.

There are a wide range of behaviors that may be concerning; we can help you talk through them and then investigate to come to an understanding of whether there is a violation of community standards. If there is a violation, we will take it seriously and will take action according to the severity of the incident.

One-time incidents that are not very severe may be handled by a conversation with the individual engaged in the problematic behavior. Here, the primary goal is education about our community expectations, a discussion of how to ameliorate the damage caused by the discriminatory incident, and an understanding of the consequences if violations of community standards were to continue.

Individual or repeated incidents that are severe enough to have a significant negative impact on an individual or on our community will be reported to [the Office of Student Affairs or the Office of Title IX](#) Initiatives, which have a processor dealing with them. These more serious incidents, when perpetrated by a student, will also be discussed at Black Friday, where consequences may be imposed such as required amelioration actions, an N-1 letter, or termination from the program.

SECTION 3: Degrees Offered

Software Engineering Ph.D. (Software Engineering Ph.D. Degree)

Software Engineering Dual Degree with Portugal Ph.D. (Software Engineering Ph.D. Degree)

SECTION 4: Departmental Personnel

Software and Societal Systems Department-S3D

Name/Title	Role/Interactions	Office	Contact Info
Nicolas Christin Department Head	The Department head is a key member of university leadership who helps shape the department's future. They work with faculty, students, and staff to establish a long-term vision for the department that aligns with the university's mission. They also serve as a liaison between the department and other units of the institution.	TCS 448	nicolasc@andrew.cmu.edu
Victoria Poprocky Administrative Assistant to Nicolas Christin, Department Head Monika De Reno, Deputy Director	Point of contact for Nicolas Christian and Monika De Reno.	TCS 433	poprocky@andrew.cmu.edu
Monika DeReno Deputy Director	Providing guidance in areas such as student learning and resource management.	TCS 449	monikade@andrew.cmu.edu
Christian Kästner Director, of Software Engineering PhD Program	The director controls the vision of the program, provides high-level guidance to students and faculty, is involved in student evaluations and in resolving any program-level disputes.	TCS 345	kaestner@cs.cmu.edu
Connie Herold Programs Manager	The Program Manager oversees all aspects of the PhD Programs administration, including admissions, personal or professional conflicts, course and faculty concerns, and program management.	TCS 449	cherold@andrew.cmu.edu
Alisha Roudebush Academic Coordinator	Coordinates the PhD Programs. Helps with admission questions, registration, commencement, orientation and general program information.	TCS 431	aroudebu@andrew.cmu.edu
S3D SE Faculty	A list of SE PhD Faculty contact information may be found online .	TCS Hall	S3D SE Faculty
S3D IT Team	S3D IT services	TCS 432	S3D IT Services Team
S3D Video Service Team	Video Technicians for S3D	TCS 246	S3D IT Services Team
Paul Stockhausen Senior Manager, SCS Building Facilities	Handles issues regarding facilities, furniture, electrical issues, and office/room access to space and maintenance.	GHC 4107	building@cs.cmu.edu SCS Building Facilities

SECTION 5: Departmental Resources

Office Assignments

It is Department policy that no student in TCS will receive less than 40 square feet of space. For any questions regarding office assignments or office space please contact [Connie Herold](#), [Alisha Roudebush](#).

Mailboxes

TCS Mailboxes are located on the Fourth Floor in room 430.
CMU main [Postal Service](#) is located in Cohon University Center (CUC)

Copy Machines

Copy Machines are available on the third and fourth floor. To create a print account please refer to [SCS Computing Facilities](#).

Keys

General access keys are distributed to each student at orientation.
Building Access Cards are distributed at orientation. Many of the buildings around campus have access control readers on their exterior doors in order to ensure security after hours and on weekends. By default and as members of the campus community, all cardholders, including students, faculty, staff, and sponsored individuals, have access to these buildings:

- University Center
- Cyert Hall
- Newell-Simon Hall
- 311 S. Craig street
- Wean Hall
- Porter Hall
- Baker Hall
- Hamburg Hall
- West Wing Computer Cluster

For questions or issues regarding building access, contact [The HUB](#).

Purchasing and Reimbursement Procedures and Policies

The university has detailed and strict policies relating to the purchase of goods, services, equipment, etc. There are also reimbursement policies, along with tax exempt considerations.

Please reach out to your Advisor's Administrative staff for reimbursements information.

Department Office/Building Security, Repairs and Services

For department office/ building repairs and services please contact Stefan Hadricky stefanh@andrew.cmu.edu.

For security concerns please contact [CMU Police](#).

Department and/or College Graduate Student Organizations/Advisory Committee/Graduate Representatives

[Graduate Student Assembly](#) (GSA) is the branch of CMU's student government that represents graduate students. Please contact Software Engineering student [Andy Hammer](#) with any questions regarding GSA.

SECTION 6: Advising

6.1: Role of an Advisor and Advisor Assignments

Except during their first month in the program, each student has a faculty advisor charged with guiding the education and monitoring the progress of the student through the program. This personal student-advisor relationship ensures that every student receives the necessary faculty mentoring. Throughout the program, the advisor is responsible for guiding the student's research and education. Early in the program, the advisor guides the student along some research initiatives and helps with strategic planning for courses and other educational activities. Later, the advisor helps to focus the student's research interests towards a thesis topic. Toward the end of the program, the advisor chairs the student's thesis committee, and helps to select the other members of the committee. The advisor also provides the student with career advice.

How are advisors initially chosen? When students first arrive at CMU, we provide an orientation known as the Immigration Course, in which students learn about the environment at CMU and meet the faculty. Each faculty member provides an introduction to their research. Students are expected to identify faculty with related research interests and set up meetings with those faculty in order to discuss a potential advising relationship. After about a month at CMU, students are matched with faculty advisors through what we call the "handshake" process. Students list faculty preferences and faculty list student preferences; the SE Ph.D. Program Director then matches each student with a faculty member, taking into consideration each of their preferences and other factors.

There is flexibility in the kind of relationship a student has with their advisor. While the advisor is a student's primary source of guidance, many students interact closely with

faculty other than their formal advisor, for example as part of a research collaboration. Some students have two co-advisors.

Occasionally evolving research interests and other factors motivate changes in advising relationships. It is OK for students to request a change in advisors (including switching advisors, adding a co-advisor, or dropping a co-advisor). Such changes are approved by the SE Ph.D. Program Director with agreement from all new (co-)advisors. Students are encouraged to gracefully tie up any loose ends in the previous advising relationships and research project.

Any non-courtesy Tenure Track or Research Track faculty member in SCS may serve as a sole advisor or co-advisor. In addition, faculty in other tracks, or in other schools, can serve as advisors with permission of the SE Ph.D. Program Director.

6.2: Advisor/Advisee Collaboration

As described in Section 6.1, the advisor guides the student in their research and mentors them toward the program goals. Ideally, the student and their advisor set goals jointly and document them in a [Individual Development Plan \(IDP\)](#). In addition to regular advising meetings, students report progress and receive an evaluation through the semi-annual Doctoral Student Review (DSR) process described in Section 10.7.

6.3: Review of Academic Conflicts

Faculty and Student Ombudspersons

If a student feels the need to reach out to someone other than their direct advisor, or the program managers, or is not sure what avenue to pursue or is appropriate for addressing his or her problem, the student can turn to the Ph.D. program's student or faculty ombudspersons.

Currently, the faculty ombudsperson is [Mayank Goel](#). The faculty ombudsperson's roles and responsibilities are:

- To meet with students and listen to their problems
- To give advice, perhaps suggesting someone else to talk to or suggesting the next step to take
- To take action on any issues where the program director may be conflicted
- To keep conversations confidential

SECTION 7: Doctoral Degree Requirements

7.1: Residency Requirements

The Ph.D. program is a full-time resident program. Ph.D. students must register as a full-time student for a minimum of 2 academic years in total. Full-time students must be resident in Pittsburgh, or with the approval of their advisor and the program head, at a collaborative site.

7.2: Expected Timeline Estimates

The following table indicates estimates for approximately when students should have finished each requirement. Overall, we expect students to complete the program within 5-6 years, depending on background and dissertation research.

These figures are meant to be suggestive, not prescriptive, and can be modified for any student by agreement with the student's advisor. We present them so all faculty and students can develop a shared image of a typical path through the program.

Component	Intensity	Completion time
Practicum	¼ time	By end of year 2
Writing skills	variable	By end of year 2
Speaking skills	SSSG	By end of year 3
Course requirements	each ¼ time	By end of year 4
Thesis proposal	½ time	By end of year 4
Teaching	½ time	By end of year 5
Thesis	full time	By end of year 5 or 6

Students are expected to be working on research every semester with intensity at least 1/2 time throughout their time at CMU. In addition, it is expected that students volunteer within

the department and school throughout their time at CMU.

Outcome to Requirement Mapping

The following table provides a correspondence between the program outcomes and the program requirements. In the table, a filled-in box indicates that the activity on the left is a principal contributor to the outcome above, whereas an outlined box indicates that the activity is an auxiliary contributor to the outcome. Naturally, for individual students, other activities might contribute as well.

Activity (Program Requirement)	Independent Research	Research Methods	Depth in Area	Broad SE Knowledge	Teaching SW	Comm Skills	Practical Development	Mature Perspective
Dir research	■	■	■	□		□	□	□
Thesis	■	■	■			□		
Course: Core				■	■			
Course: SYM		■	□	■	□			
Course: ENG		□	□	■	□		□	
Course: BEH		■	□	■	□			
Course: SOC			□	■				■
Electives			■	■	□			
SSSG		□		■		■		
TAing					■	□		
Comm skill	□				□	■		
Practicum							■	□

7.3: Registration Process

For registration information please refer to the [HUB](#).

Summer registration will be done by the program coordinator and will be based on the student's response to the Summer Plans form that will be sent in early spring. The deadline for summer plans is May 1st.

Software Engineering courses you must register for Fall and going forward.

17-808	Software Engineering Research	12 Units	Fall – First year only
17-997	Graduate Reading & Research-Section A	Variable	Every Semester

XX-7XX	Optional: Any Star or Elective course on a topic that interests you.	6 or 12 Units	
--------	--	---------------	--

You must carry 48 units every semester. You will adjust your Reading & Research Course (17-997) to meet these units. For example, in your first semester:

17-808 – Software Engineering Research (12 Units)
 XX-7xx – Star course or elective if you choose to take on this semester (12 Units)
 Total of 24 Units

You will then register for your Reading & Research course for (24 Units) = Total of 48 Units

7.4: Required Units for Degree Attainment

The purpose of completing graduate courses at Carnegie Mellon University is to attain a broad understanding of software engineering and closely related fields, a core set of research skills, and a deep understanding of topics that lead into the student’s thesis research.

Our requirement is that students **complete 84 university** units, which is the equivalent of 7 standard 12-unit courses.

Our core research course, 17-808, provides an understanding of the Software Engineering field, including important ideas and the major research strategies in use. Certain courses are designated as ‘star’ courses because they provide a solid foundation in some areas. By taking a star course in each of the four categories, students acquire breadth through exposure to fundamental knowledge, concepts, and skills in software engineering. Through the equivalent of two elective courses, students typically choose to gain more depth in the student’s particular area of research.

Some students use electives to gain more breadth by specialized exposure to an area outside of the student’s core research area.

7.5 Core Courses

Software Engineering Research 17-808

The Software Engineering Research Course requirement is fulfilled by taking the 12-unit course 17-808: *Software Engineering Research*, typically in their first semester in the software engineering Ph.D. program. Our core research course, 17-808, provides an understanding of the Software Engineering field, including important ideas and the major research strategies in use.

This course is taught jointly by the software engineering faculty and is designed to prepare Ph.D. students to do research in software engineering. It introduces important ideas across the breadth of software engineering and the major research strategies of the field. Students will become familiar with the structure of the field; they will learn the seminal ideas and developments that led to current research questions; they will learn to critique research papers to evaluate their claims and evidence; and they will also become familiar with the current software engineering research themes at CMU.

7.6 Star Courses

Four Areas:

Each student must pass one-star course from each of four categories:

- SYM: Symbolic mathematical modeling and analysis
- BEH: Human-focused empirical research
- ENG: Design and engineering of software systems
- SOC: The interaction of software with larger issues in society, business, or public policy.

These categories are chosen to ensure that students acquire breadth through exposure to fundamental knowledge, concepts, and skills in software engineering. Each category captures a particular set of knowledge and skills that every software engineering student should possess. At the same time, the choice of courses within the categories gives students the flexibility to customize their course selection to their individual needs.

Star courses are identified as especially appropriate to provide grounding in a topical area or set of research skills within Software Engineering. The criteria for a star course include:

- They provide a broad introduction to a topic or skills relevant to software engineering research, while being deep enough to be appropriate for Ph.D. studies.
- They assume only an undergraduate background in the relevant area.
- They use multiple forms of evaluation, such as assignments, exams, projects, or term papers.

Approved Star Courses

The more detailed descriptions of the star course categories, together with the rationale for their selection and the currently approved courses in each category, are listed below. For requesting the approval of additional courses as star courses see [9.1: Petition Procedures](#) for adding a course to the current STAR list.

SYM

A course whose primary focus is on symbolic mathematical modeling and analysis techniques that are applicable to software artifacts. Students taking a SYM course should

engage in symbolic research methods that might include discrete models, proofs, state space exploration, or other software-relevant mathematical topics. Symbolic mathematical techniques are useful in many areas of software engineering research, and more broadly, many students have found them helpful in writing careful definitions and precisely distinguishing among related concepts. The courses currently approved in this category are:

- 10-701 Machine Learning (Ph.D.-level)
- 11-711 Advanced NLP
- 10-715 Advanced Introduction to Machine Learning
- 10-725 Convex Optimization
- 11-727 Computational Semantics for NLP
- 15-812 Programming Language Semantics
- 15-814 Types and Programming Languages
- 15-816 Advanced Topics in Logic: Automated Reasoning and Satisfiability
- 17-714 Formal Methods AND 17-724 Advanced Formal Methods†
- 17-819 Program Analysis
- 80-610 Formal Logic

ENG

A course with a primary focus on software systems design and engineering. Courses in this category must include (A) significant engagement with software design, (B) consideration of software artifacts at a significant scale and complexity, and (C) exposure to the tradeoffs (such as cost/benefit) at the core of the engineering discipline. The courses currently approved in this category are:

- 05-631 Software Structures for User Interfaces
- 05-830 Advanced User Interface Software
- 10-714 Deep Learning Systems
- 15-640 Distributed Systems
- 15-712 Advanced Operating Systems and Distributed Systems
- 15-745 Optimizing Compilers for Modern Architectures
- 15-793: Secure Computer Systems
- 15-821 Mobile and Pervasive Computing
- 15-829 Programmable Networks
- 17-626 Requirements for Information Systems†
- 17-700 Data Science and Machine Learning at Scale
- 17-712 Fantastic Bugs and How to Find Them
- 17-745 Machine Learning in Production
- 17-766 Software Engineering for Startups
- 17-722 User Focused Sensing Systems
- 17-770 Virtual Machines and Managed Runtimes
- 17-882 Software Architectures†

- 18-730 Introduction to Computer Security
- 18-732 Secure Software Systems
- 18-749 Building Reliable Distributed Systems

SOC

A course with a primary focus on how technology interacts with larger issues in society, business, or public policy. This requirement is intended to create breadth in the curriculum, pushing students out of a focus on the software system itself (which is what ENG does). Suitable courses are commonly found outside of the school of computer science, but can also be found within the school, for example HCI or robotics courses strongly rooted in psychology. At least half of the course content, including some evaluations, should go beyond technical implementation and algorithmic concerns. The courses currently approved in this category are:

- 05-618 Human-AI Interaction
- 05-813 Human Factors†
- 05-820 Social Web
- 05-899 Fairness, Accountability, Transparency, & Ethics (FATE) in Sociotechnical Systems ‡ **
- 08-631/17-631 Information Security and Privacy
- 08-805/17-735 Privacy Engineering
- 10-713 Machine Learning, Ethics and Society
- 16-735 Ethics and Robotics
- 16-867 Principles of Human-Robot Interaction
- 17-668 Network Analysis
- 17-731/18-734 Foundations of Privacy
- 17-733 Privacy Policy, Law and Technology
- 19-758 Special Topics: Organizational Theory for Engineers
- 90-802 Information Security: Comparison of US and European Policies
- 90-880 Strategy and Management of Technological Innovation

BEH

A course that is primarily concerned with behavioral science research methods and research study designs. The course must cover one or more human-focused empirical research methods for data collection and data analysis in depth. These methods may involve working with subjects directly or inferring information about subjects based on artifacts they have left behind, as in mining software repositories research. Common examples include case studies, interviews, surveys, human subjects' experiments, or various forms of statistical analysis. The course must also discuss higher-level study design issues that expose students to the inherent strengths and weaknesses of different research methods applied in different contexts for answering different types of research questions. Finally, the course must require students to plan and prototype a sample project using at least one of these research methods in some depth.

- 05-748 Research Methods for the Learning Sciences**
- 05-816 Applied Research Methods
- 05-823 E-Learning Design Principles and Methods
- 05-899 Human Aspects of Software Development ‡ **
- 17-734 Usable Privacy and Security
- 17-803 Empirical Methods
- 36-743 Statistical Methods for the Behavioral and Social Sciences **

** Not offered regularly

† These courses are approved even though they are 6 or 9 units only under the condition that they are complemented with an independent study for the remaining 6 or 3 units. The independent study is to be arranged with the course instructor and will typically focus on a research project related to the course content.

‡ HCI recycles their course numbers for special topics courses, making sure that the title matches.

7.7: Electives

24 Units Required

Students must take 24 university units worth of elective courses. In general, elective graduate courses must be relevant to the software engineering degree and must be Ph.D. level (University course numbering 700 and above); exceptions to the latter rule may be made with a note from the student's advisor to the SE Ph.D. Program Administrator.

Through the equivalent of two elective courses students may use electives to gain additional depth of knowledge in the student's research area, e.g., to complement their directed research or to prepare them for choosing a thesis topic. Students may also use electives to gain additional breadth of knowledge in an area outside of the students research area.

We strongly advise students to choose electives in consultation with their advisors. The student and their advisor are both responsible for making sure that through these 24 elective units the student gains new knowledge, perhaps to fill gaps or prepare for thesis research. Students are free to take more than the required number of elective units. The following is a sample of past electives taken.

- 05-830 Advanced User Interface Software
- 05-899D Human Aspects of Software Development
- 15-816 Modal Logic
- 15-819 Advanced Topics in Programming Language

7.8: Software Engineering Research Seminar

All SE Ph.D. students are expected to actively participate in the weekly Software Engineering Research Seminar (SSSG) each semester. This seminar is an opportunity to learn about Software Engineering research in the department and to gain presentation skills. Each student is expected to present twice per year in the seminar.

If the student presents in another academic venue, the requirement is reduced to one talk per year. The outside presentation can come in many forms and should be research-related or teaching-related, such as (a) presenting research at an external workshop, seminar, or conference, (b) giving a guest lecture in a course, and (c) a practice talk given to a research group. Presentations given as part of required coursework or required TAing duties do not qualify. It is expected that the outside presentation is non trivially different from the SSSG presentation, at a minimum a substantially revised version of a practice talk.

Typically, at least one SSSG presentation per year should be related to the student's recent, current, or future research.

If during the doctoral review meeting, it is determined that a student has not met these requirements, the student will be asked to schedule additional presentations in SSSG to make up for the difference in the next year.

A student who has a course conflict with SSSG should get approval from their advisor and the SE Ph.D. Program Director; approval is routinely granted but is expected to be rare (typically once or twice in a student's time in the program).

This seminar is typically held on Mondays between 3:30-5:00 during the semester in TCS 358.

7.9: Written Communications Skills

To be a well-rounded software engineer, each student should have not just basic knowledge, but also the abilities:

- To communicate technical ideas clearly in writing
- To communicate technical ideas clearly orally

We also expect students to be able to program, but there is no formal checkpoint to certify programming skills. It is left up to the advisor and student to make sure the student has the necessary skills.

Writing Proficiency: To satisfy the writing proficiency requirement, each student must write a scholarly document, as either its sole author or its primary author (if co-authored), that is at least the quality of a Carnegie Mellon technical report. This document must be a scholarly paper with references to the literature that could be submitted for peer review. It may be a technical report, a paper published at or in preparation for a conference or

journal, a document written to satisfy a course requirement (e.g., a course project report), or a comprehensive survey paper (e.g., suitable for submission to ACM Computing Surveys). There is no requirement on scope or length of the paper, as long as it is a fully formed work that presents scientific results with the structure one would expect in a conference or survey paper. It is OK if the paper includes edits by other authors, as long as the student wrote the majority of the text and was involved in the editing process.

Annotated bibliographies, user manuals, and reference manuals do not qualify because they do not require the same kind of explication, organization, and summarization skills needed to write a conference- or journal-like publication. Similarly, opinion pieces, proposals, or descriptions of work in progress do not qualify. The paper may not be a practicum document, the thesis proposal, or the thesis.

The writing requirement is evaluated by at least **two faculty** members and **one S3D** Ph.D. student who has fulfilled a writing requirement. Any non-courtesy Tenure Track or Research Track faculty member in SCS may be an evaluator. In addition, faculty in other tracks, or in other schools, can serve as evaluators with permission of the SE Ph.D. Program Director, which is typically granted for faculty active in research in a relevant area. One of the reviewers must be a faculty member of S3D, and none of them should be a co-author of the paper being reviewed.

These evaluators must read the document and provide written feedback using the [Writing Evaluator Form](#).

Students are responsible for asking the appropriate faculty members and Ph.D. student to help them with satisfying their writing requirement. We expect students to be able to satisfy this requirement within their first three years, and prior to their thesis proposal.

Ph.D. students are welcome to enroll in the undergraduate communications course, required of undergraduate computer science majors, to enhance their writing skills; however, taking it is not sufficient to satisfy the written communication skills requirement.

7.10: Speaking Proficiency

At the student evaluation meeting held each semester, the faculty makes a judgment to pass students who have demonstrated high speaking proficiency through the presentation(s) they have given in the Software Engineering Research Seminar and other venues. We expect a standard of proficiency typical of good presentations at an academic conference, or of a respected instructor giving course lectures.

Students who have passed the proficiency requirement still benefit from honing their speaking skills, and so they are expected to continue to attend the Seminar, including active participation and twice-yearly presentations as described in the Software Engineering Research Seminar course requirement (above).

7.11: Practicum Requirement

An integral part of S3D's software engineering research program is ongoing interaction with industrial-strength software development in a real (not just realistic) setting. Many students already have industrial experience. Those whose prior experience is insufficient may be required to acquire such experience, typically by participating in one or more industrial internships while in the program.

The purpose of the practicum, therefore, is to ground academic study in practice by careful reflection or scientific study of direct experience in software development. Each student will complete one practicum. A practicum may take one of the following forms:

1. An issue-focused reflection and analysis of a practical software engineering experience of the author.
2. An empirical study of (some aspects of) the software development process.

A practicum of the first type is similar to an experience report, such as those published in the ICSE Software Engineering in Practice track but is not expected to attain the level of polish or broad applicability that might be expected for formal publication. It is not merely a report of the author's experience. Rather, it is a critical reflection on that experience, focused on a well-defined issue or a related set of issues. The practicum should be grounded in experience and careful observation, and possibly data as well. At the same time, it should draw out substantive lessons that might be applied to other similar situations. As an informed reflection, it should be framed in the fundamental concepts of the software engineering literature, which might explain or contradict the student's practical experience. It is not always necessary to have sufficient data for statistical validity, but in all cases the narrative should be clear about the strength of the evidence.

The second category of practicum is in the form of a scientific paper, potentially publishable at a peer-reviewed conference or workshop in the area of empirical software engineering.

Practicum Format and Evaluation

The practicum can be presented (a) orally as a talk in the Software Engineering Research Seminar (SSSG) or (b) in the written form of a report. In either format, the practicum shall be self-contained and suitable for an audience of entering software engineering PhD students or advanced undergraduates. The practicum shall be presented in a scientific style: it should have clear definitions, careful distinctions between observations and interpretation, and appropriate comparisons to the scientific literature. A typical written report would be 7-10 pages in a normal technical report format. Students may choose to develop their practicum as part of taking the course 17-415 Software Engineering Reflection. We expect a typical student will be able to prepare and practice a practicum talk or write and revise a practicum report in approximately a week of work (40 hours).

The practicum talk or report must be approved by two faculty members. Any faculty in the S3D are eligible; other faculty may be approved by the SE Ph.D. Program Director. If the

practicum was written as part of taking 17-415 Software Engineering Reflection, instructor approval is sufficient without input from another faculty member. For oral presentations, the talk faculty evaluators will provide immediate feedback based on the provided rubric, for written reports the expected cycle for feedback is approximately 3 weeks.

Guidance and a rubric for evaluating the practicum is available together with past practicum reports on the internal Google drive for the SE Ph.D. Program.

Practicum and Confidentiality

As with any report on practical experience, practicum papers may be sensitive. Practicum reports must be available to members of the Carnegie Mellon Software Engineering Community without restriction; however, they need not be public beyond the scope of that community. It is the student's responsibility to ensure compliance with any NDAs the student may have signed. As with other papers, it is acceptable to shield the identity of individuals and organizations, as well as details of data about the experience (for example, by removing units from graphs).

Prior practicum reports are available on the password-protected internal SE Ph.D. Program web page, along with informal pragmatic advice on writing the practicum document.

7.12: Specific Requirements for the CMU Portugal Program

Portugal Curriculum

The Software and Societal System Department (S3D) offers a dual degree Ph.D. in Software Engineering in cooperation with several Portuguese universities. The regulations are essentially the same as given in this document, except that some requirements can be fulfilled in an affiliated program in Portugal.

The Dual Degree Program is based on a matching of requirements of both the Carnegie Mellon and the partner institutions PhD Programs. Each student will be co-advised by a S3D-SE/Carnegie Mellon faculty member and a faculty member of the Portuguese Institution.

The student is expected to spend two years at S3D/Carnegie Mellon. In this period, the student shall take star courses (as specified by Carnegie Mellon/S3D SE regulations) and start working on research.

The second period, whose duration is to be determined by the co-advisors but is expected to last typically three years, will be hosted by the partner institution in Portugal, where the student is expected to continue with the research work. The research work will take place in the context of collaborative research activities between partner institutions and Carnegie Mellon.

The remaining time until completion of the thesis shall be divided between research stays at SE-S3D/Carnegie Mellon and the Portuguese Institution. The schedule should be

flexible to fit the needs of the doctoral research project.

By the end of the third or the fourth year the student must present the thesis proposal formally at S3D SE/Carnegie Mellon or at the partner institution in Portugal. The Portuguese co-advisor should attend the oral presentation and discussion of the Thesis Proposal at S3D SE/Carnegie Mellon or via videoconference.

The public oral thesis defense will take place either at Carnegie Mellon or at the Portuguese Institution, according to the rules of the SE PhD program. The thesis evaluation committee must include both the Carnegie Mellon and the Portuguese co-advisors, members of the advisory committee, and external reviewers

Courses

The requirements for courses, skills, and teaching for the dual degree program are the same as for the software engineering degree program, with the following exceptions:

1. Students may petition the program for approval of one class taken in Portugal as part of the Ph.D. program to be counted for a Star category, and any number of classes taken in Portugal as part of the Ph.D. program to be counted as electives. Those courses must fit our requirements for Star courses and electives as described in this handbook. Students should follow the petition process for approval of CMU courses described in this handbook to request approval.
2. Students may complete the practicum and writing requirement in Portugal with the completion of CMU paperwork.
3. The evaluation of speaking skills will typically consider talks in SSSG but may also consider outside talks
4. One of the two TA requirements can be completed in Portugal, if the instructor fills out the CMU paperwork.

Note that the “Course Waiver Policy for Prior Courses Work” applies equally for dual degree students, many of whom may have previously completed a Master’s degree. In requests, students should clearly indicate whether they intend to waive or approve a course requirement based on a course taken in Portugal.

Portugal students will receive a Software Engineering Degree from Carnegie Mellon for Software Engineering

7.13: Course Waiver

This program does not accept transfer credit of courses taken outside of CMU. However, students may request to waive up to two course requirements of the PhD based on

equivalent graduate level work.

Students may request to waive up to two course requirements based on equivalent graduate-level courses they have already taken, or based on industry experience they have acquired, prior to entering the Ph.D. program. The Software Engineering Research course cannot be waived, and no more than 12 units of star courses can be waived.

To apply for a waiver, a student must submit a petition to the Ph.D. Program Coordinator **within their first year in** the program. A separate petition must be submitted for each course to be waived. The prior course need not be equivalent in content to one of the approved courses in the same category but rather should be equivalent in substance: a student who has done excellent work in an intellectually rigorous graduate course on a computer science topic that we happen not to teach may be granted a waiver if it matches the intent of the corresponding category.

The petitioner must make a case for how prior courses are equivalent in substance, submitting a self-contained justification, a syllabus, and a transcript (translated if necessary). As needed, the petitioner can provide additional support for the case by providing slide excerpts, reading lists, homework assignments, work products, or other supporting artifacts. For star courses, the justification must argue why the course(s) match the intent and expectations of the appropriate star requirement, similar to requests for approval for a new star course (described above). To waive a course based on comparable experience, the student should document how the experience demonstrates their systematic knowledge of the relevant material.

Typically, courses may be waived based on rigorous graduate courses taken as part of a master's degree. For example, a master's level psychology course might be appropriate to fulfill the BEH star requirement, and a similar economics course might fulfill the SOC requirement. Students who have taken additional Master-level courses during their undergraduate degree that did not count toward their undergraduate graduation requirements may request waivers based on those. In rare cases, unusual advanced undergraduate coursework or research experience may be sufficient to waive a course, if the material covered is equivalent in substance to the star courses in question.

In order to apply for the course waiver please fill out the [SE Course Waiver form](#) with all supporting documents and send it to the Program Director Christian Kästner or Connie Herold and Alisha Roudebush.

7.14: Protocol for Evaluation of Transfer Credit

The Software Engineering program does not accept transfer credit.

7.15: Protocol for Evaluation of Course Waiver

Students may request to waive up to two course requirements for the PhD degree based on equivalent graduate level courses they have already taken as described in Section 7.13.

All grades for course waivers will be recorded as part of the internal Student Record as with CMU courses taken while enrolled in the PhD program. Any course waiver will not show up in the students official student transcript.

These petitions will be considered by the faculty, typically delegated to a faculty member with expertise in the corresponding star area. The outcome of the petition process will be decided based on provided information on the prior course and the petitioner's performance in it. The faculty may ask appropriate instructors for assistance in this decision or reject the petition with a request for resubmission if insufficient information is provided. If appropriate, the faculty may stipulate conditions on the waiver, such as preparing a supplementary project that exposes the student to software engineering research in the area of the course and is roughly equivalent in scope to a final project in a typical PhD. level course.

7.16: Teaching Requirements/Opportunities

The ability to teach is an important skill for all scientists, not only for those who plan to teach after completing their degrees. Teaching skills include the ability to communicate technical material ranging from elementary to advanced, and to communicate technical material to audiences ranging from general to specialized. Thus, we expect students to develop and exercise teaching skills as part of their graduate education.

Students have ample opportunities to present advanced material while working on research projects, by participating in research seminars and by giving practice conference talks. To gain experience in presenting material at an introductory or intermediate level, we require that all graduate students help teach two courses. The norm is for students to teach one course focused on introductory material in computer science or software engineering, and one course focused on mastery of material (typically an advanced undergraduate or master's course). Teaching assistants typically spend 15-20 hours per week.

In the rare event that students desire to fulfill their teaching requirement using the same course twice, then they are expected to accept additional responsibility. The additional responsibility is intended to afford the student an education benefit that is at least equivalent to TAing two different courses. Thus, students should expect to supplement their second TA'ship by assuming the teaching apprentice or co-instructor role in the second course installment, as defined below:

- Teaching Assistant (TA) is the norm and generally consists of assisting with grading, holding office hours and tutoring, teaching recitations if these exist, and developing

exam questions.

- Teaching Apprentices fulfill all the TA responsibilities, in addition to some appropriate combination of assisting with lecture design for two or more lectures, managing teaching staff, if there are staff, course module redesign, and homework design.
- Co-instructors are an official designation assigned by the registrar and individuals in this role will share responsibility for the entire course with a second co-instructor, who is usually a faculty member. This may include lead responsibility for one- to two-thirds of the lectures and a commensurate portion of designing and evaluating homework, in-class assignments, and exams.

The table below summarizes the key differences between teaching assistants, teaching apprentices and co-instructors: the columns correspond to roles, checkmarks indicate that a teaching activity is often expected, question marks indicate the activity may arise and dashes indicate the activity is often absent in the role. The roles for teaching apprentice and co-instructor should be individualized to a level of effort and teaching activities that are appropriate for the student and the course. While the responsibilities are monotonically non-decreasing from left to right, the teaching apprenticeship is not a prerequisite to become a co-instructor. Finally, the differences among roles were developed assuming that the course is conducted in a lecture-style, however, the expectation for increasing responsibility for each role can, and should, be adapted to courses taught in any other style.

Teaching Activity	Teaching Assistant	Teaching Apprentice	Co-Instructor
Grading	Typical	Typical	Typical
Lecturing	Maybe	2 or more	1/3-2/3 lectures
Managing staff	No	If there are staff	Typical
Module redesign	Maybe	Typical	Typical
Major course revision	No	Maybe	Maybe
Office hours, tutoring	Typical	Typical	Typical
Recitations	Typical	Typical	Maybe

Design/refine homework	Typical	Typical	Maybe
Design exams	Contribute	Typical	Typical
Assign grades	No	Contribute	Typical

Key: Typical = typical activity in this role, Maybe = activity may arise for this role, No = activity is often absent for this role

SE PhD students who hope to perform any of the above roles should contact the S3D TA coordinator (currently **Michael Hilton**) in the semester before the TA-ship would begin to get approval for TAing. There are typically two cases:

- Case 1: The student is looking for a course to TA. The coordinator will normally have a list of courses that are looking for TAs and the student should discuss possible options with the coordinator.
- Case 2: The student has some idea of the course to TA, perhaps having already talked to the instructor. In this case the student should send an email to the coordinator requesting to be assigned as a TA for that course. The request should include: (a) the course number and title, and whether it is primarily an undergrad, master's, or PhD course, (b) whether TAing has been discussed with the instructor, and (c) what role the student would have as a TA in the course (e.g., as part of a team, as the sole TA, etc.).

After serving as a teaching assistant, in order to receive credit for the teaching requirement, the student must have the faculty fill out the [teaching assistant evaluation form](#).

Students are encouraged to teach more than twice. At the semiannual student review meeting the faculty give special recognition to those who do an outstanding job as a TA and to those who teach beyond the required load. The School of Computer Science and the CMU Eberly Center offer teaching workshops which we encourage students to take advantage of. S3D also regularly offers a CS Pedagogy course (15-890).

Graduate students are required to have a certain level of fluency in English before they can instruct in Pennsylvania, as required by the English Fluency in Higher Education Act of 1990. Through this Act, all institutions of higher education in the state are required to evaluate and certify the English fluency of all instructional personnel, including teaching assistants and interns. The full university policy can be reviewed [here](#).

The fluency of all instructional personnel will be rated by Language Support in the Student Academic Success Center to determine at what level of responsibility the student can TA. In addition to administering the International Teaching Assistant (ITA) Test (a mandatory screening test for any non-native speaker of English), Language Support in the Student

Academic Success Center helps teaching assistants who are non-native English speakers develop fluency and cultural understanding to teach successfully at Carnegie Mellon. Visit the [Student Academic Success Center](#) website for additional information.

7.17: Research Requirements/Opportunities

The Software Engineering Ph.D. is first and foremost a research degree, and carrying out direct research is the most important activity for students in the program. We expect students to spend at least half their time on research throughout the program. Accordingly, active students (i.e., those who are not on LOA or ABS status, are not Dual Degree Portugal students while in Portugal, or are not taking a summer vacation semester) must enroll in approximately 36-48 units of Graduate Reading and Research each semester.

At each semiannual graduate student review meeting, the faculty assess the student's previous semester's research progress and the student's next semester's research plans to ensure that the student is making satisfactory progress. The evaluation of a student's progress in directed research often depends on the student having produced some tangible result; examples include the implementation of pieces of a software system, a written report on research explorations, an annotated bibliography in a major area, or, as part of the preparation for doing research, a passing grade in a graduate course (beyond the 84 required units).

Advisors are responsible for supervising this portion of the Ph.D. program, with regular input from other faculty provided at the semiannual student review as well as in more informal settings.

Resources and Regulations Governing Research at Carnegie Mellon

- [Office of Sponsored Programs](#)
- [Office of Research Integrity & Compliance](#)
- [Intellectual Property Policy](#)
- [Policy on Restricted Research](#)
- [Human Subjects in Research Policy](#)

7.18: Internships

Summer Internship Opportunities

Practice-oriented summer internships are particularly important for students who have had little or no prior full-time experience in the software industry. Faculty can provide help in finding suitable summer employment.

Different advisors may have different default assumptions about internships and summer support. Students should talk to their advisors about usual expectations and specific summer plans well in advance.

Students who participate in relevant software engineering internships may remain an active student by taking the 17-998 Section I, Practicum Internship course.

International Students/Summer Internship

Ph.D. students are eligible for [CPT](#) based internship course if the following conditions are met:

1. You have completed two consecutive semesters in the U.S. maintaining F-1 Status
2. You will enroll in 17-998 - SE Internship Course (Section I). This will count towards your research units that will fulfill your degree requirements
3. You have not yet completed your coursework requirements (there are different eligibility criteria if you have completed your coursework requirements)
4. You have not yet reached the maximum number of off campus employment semester experiences (max of four). Please note an extension into the next semester with the same company will not count as an extra semester in your total of 4 allowable internships during your Phd. in Software Engineering.

Ph.D. students are eligible for [CPT](#) if they have completed their coursework requirements and:

1. The internship is essential for your dissertation research/data collection
2. The research needed is proprietary and only accessible if you are an employee of a specific employer.
3. The employer provides an email or letter explaining they are aware your employment is for your dissertation, and your employment is necessary to access the research/data needed. They also must provide a brief summary of how your employment is integral for your dissertation
4. Your academic advisor affirms on the OIE CPT advisor form that this internship is needed for research/data collection for our dissertation.
5. You have not yet reached the maximum number of off campus employment semester experiences allowed of 4.

During any semester, students studying via an internship experience may substitute up to 36 units of Practicum Internship for these research units with permission of the advisor or Program Director. Note that students typically substitute practicum units for research no more than 4 times during their PhD degree program.

SECTION 8: Dissertation Preparation & Requirements

8.1: PhD Criteria for Advancement to Candidacy

After the acceptance of a student's thesis proposal by the thesis committee, and after the student has satisfied all other requirements except for the dissertation and its oral defense, the student is regarded as having "all but dissertation" (ABD) status.

An ABD candidate may choose to continue as a regular student In Residence, or, if the residency requirement above is fulfilled; they may choose to be In Absentia (ABS).

ABS - Off Campus: Students who leave CMU but plan to continue working on the thesis will be classified as ABS. These students should not require substantial use of university resources, but are permitted use of the libraries and consultation with faculty or students as necessary. While the candidate is ABS, they are required to pay the university technology fee each semester. No formal enrollment or payment of tuition is required, with the exception of the academic semester in which the degree requirements are completed. A candidate who is ABS is required to enroll for a minimum of five units during the academic semester in which the degree requirements are completed. For students defending remotely without returning to campus a Dissertation Completion Fee is charged. Charges for these units are the responsibility of the candidate.

Since an ABS candidate will not be certified by the University as a "student" for immigration purposes, students who are in the United States on a student visa and who become ABD should not choose to become ABS.

ABD - On Campus: Students who are self-supporting and are in ABD status may remain on campus to complete the thesis. They must register and pay for a minimum of five units each semester. However, students who receive a stipend based on their status as a graduate student and paid by or administered by the university will be required to register for a minimum of 36 research units. Nearly every ABD student in S3D falls into this latter category.

The ABD Status Agreement Form can be found at:
<https://www.cmu.edu/hub/docs/abd-status-agree.pdf>

Students are advised to read CMU's Doctoral Student Status Policy (<https://www.cmu.edu/policies/student-and-student-life/doctoral-student-status.html>) and talk to the program administrator to fully understand the financial implications of deciding to switch to In Absentia status, before making any decisions.

8.2: Ph.D. Dissertation

The thesis must describe a significant piece of original research work and must describe it well. It is on this basis that the Software and Societal Systems department certifies the qualifications of the new Ph.D.

Furthermore, it is the most important basis on which the scientific community judges the initial achievement and potential of that individual.

8.3: Thesis Committee

The student's advisor chairs the thesis committee. All other committee members, including the external member, should be agreed upon before the thesis proposal presentation. Members of the student's committee must accept the responsibility of meeting with the student regularly to ensure that the research is progressing in the right direction.

The Thesis Committee must consist of at least one Software and Societal Systems department faculty member, two other members of the SCS faculty and/or other faculty approved by the Department Director, and an external committee member. All thesis committees are subject to departmental approval.

8.4: Proposal

The student submits a written proposal to the faculty. The student also orally presents the thesis proposal to interested faculty and students in a public colloquium. The Thesis Proposal must describe a significant piece of original research. It is evidence of proficiency, high attainment, and ability to do research in software engineering

Guidelines:

- Explain the basic idea of the thesis topic (e.g., the problem to be solved and the approach to solving it)
- Argue why that topic is interesting (e.g., what contributions to the field would be made in carrying out the proposed work)
- State what kind of results are expected
- Argue that these results are obtainable within a reasonable amount of time
- Demonstrate that the student is qualified to perform the proposed work, including an understanding of the area and its literature

The main purpose of the thesis proposal is to convince the faculty that the chosen thesis topic is significant, and that the student's approach has a reasonable chance of success. A thesis proposal gives the faculty the opportunity to pass such judgment at the start of the work and not at the end. We want to minimize the chance that a thesis will be turned down when almost completed. We expect students to present their thesis proposals as early as

possible, not halfway through writing the thesis. A thesis proposal should be short, about 15–20 pages, and the oral presentation should take about 40 minutes, not including questions.

A thesis proposal should **not** be:

- A dry run for the thesis
- A summary or abstract of the thesis
- The first chapter or part of the thesis
- A technical report
- A survey of the field
- An annotated bibliography

Any included list of references or bibliography should serve the purpose of supporting the assessment of the state of the art and the student's personal qualifications.

To provide ample notice to the public, at least one week in advance of the oral presentation, students should provide the SE Ph.D. Program Administrator with an electronic copy or link to the thesis proposal, an electronic copy of the proposal's abstract, and a list of the thesis committee members, including the external member. The committee should also be consulted about the readiness of the proposal before announcing the oral presentation. The SE Ph.D. Program Administrator posts the public announcement of the thesis proposal presentation.

At least **three** thesis committee members (including the Chair) must be physically present for the thesis proposal, and the thesis proposal must be held at Carnegie Mellon. The only exception is for students in Ph.D. programs offered jointly with other universities, in which case the thesis proposal may be held remotely, with one Carnegie Mellon member physically present, and the proposal session broadcast to a Carnegie Mellon room open to the public.

After the presentation and approval of the proposal, the Committee Chair will send the Program Administrator a "Pass form".

When the student has completed all of the program requirements and passes their Proposal, they will complete the Doctoral Candidate Contractual Agreement form provided by the graduate programs administrator.

8.5: Proposal Scheduling

All Thesis Proposals must be given during the Academic Calendar year.

Thesis Proposals should be scheduled only during academic periods – not during holidays, weekends, etc. and should be scheduled in normal business hours. The department head must approve exceptions. Please schedule for 2 hours.

Students should coordinate a date with their committee members and finalize the date as early as possible to allow time for travel arrangements, final review, and comments by committee prior to proposal date.

Students need to coordinate with their advisor's assistant to secure a room for their proposal. The room scheduled must be available to the public and must accommodate a reasonably large number of people (25-30).

Once you schedule your defense, please reach out to our IT team to let them know the date and if you will need any help in setting up for zoom, etc. Please note they will not automatically be there to set up unless you request help.

If Proposal has a remote component such as (Zoom), the student is responsible for setting up the zoom link, and completing the "Consent to Publicly Livestream a Presentation".

Thesis Proposal will be announced to the public

Please send the following information to Alisha Roudebush aroudebu@andrew.cmu.edu **ten business** days prior to your proposal date:

1. Your name as it should appear on your diploma
2. Thesis Title
3. Date/Time/Location of proposal presentation
4. Thesis Committee Members, names, titles, affiliations, and email addresses
5. Thesis Abstract – Less than 350 words describing the thesis
6. Pointer to Thesis for any outside faculty who wish to review (Website address, etc.)
7. Print and post 10 Posters to the SCS Community (TCS/Wean/ Gates)
8. Send Alisha a PDF of Poster: [Thesis Poster Template](#)
9. Zoom link, if applicable

Policies:

- [University Policy for Doctoral Student Status](#),
- [University-level ABD status](#) & [frequently asked questions](#)

8.6: PhD Defense

The student's thesis committee decides whether to accept the thesis based on its content and the outcome of the thesis defense, which is a public presentation describing the contributions of the thesis.

At least **two** weeks in advance of the oral presentation, students must provide the SE Ph.D. Program Administrator with an electronic copy of the abstract and a list of all thesis committee members. The SE Ph.D. Program Administrator posts the public announcement of the thesis defense.

Thesis Defenses must be given during the academic year.

Committee members: At least **three** thesis committee members (including the Chair) must be physically present for the thesis proposal, and the thesis proposal must be held at Carnegie Mellon. The only exception is for students in Ph.D. programs offered jointly with other universities, in which case the thesis proposal may be held remotely, with one Carnegie Mellon member physically present, and the proposal session broadcast to a Carnegie Mellon room open to the public.

Before the thesis defense, the entire thesis committee is expected to have read the entire thesis, to have given comments to the candidate, and to have given approval for scheduling the public defense. This means that a copy of the complete thesis document should be provided to the whole thesis committee a minimum six weeks in advance of any proposed date for the defense.

Significant deviations from this rule must be approved by the SE Ph.D. Program Director. Committee members should meet briefly before the thesis presentation to discuss any issues.

The presentation by the candidate is normally about 45 minutes, followed by a question-and-answer period which may be as long as needed. Plan to schedule your defense for 3 hours.

The thesis committee chair (advisor) determines who may ask questions and in what order and brings the discussion to a close at the appropriate time. The question-and-answer period is followed by a closed-door session attended by only the members of the thesis committee and any interested faculty members. The options of the committee are:

- To approve without corrections
- To approve subject to minor changes, to be approved later by the thesis chair only
- To require a resubmission after major changes and reapproval of the entire committee
- Not to approve the thesis

All members of the committee are required to sign a Final Oral Examination card, indicating that the student has passed the thesis oral examination. In addition, the thesis committee chair, the Department Head, and the Dean sign a final certification sheet when the student submits the final version of the thesis.

The SE Ph.D. Program Administrator maintains a checklist of procedures for scheduling the thesis oral presentation and completing the other requirements for graduation. The SE Ph.D. Program Administrator certifies fulfillment of requirements for graduation only when the final version of the thesis has been approved by the thesis committee, the Institute Head, and the Dean. Students are not allowed to participate in commencement exercises unless final certification has been made, so the Ph.D. defense should be scheduled a few weeks in advance of graduation to allow time for possible revisions and certification.

If the final copy of the thesis is not submitted within one year of the thesis defense, the

faculty may require a second defense before making a final certification.

Your degree title will appear on your diploma as: **Ph.D. in Software Engineering.**

8.6: Defense Scheduling

Thesis Defense should be scheduled only during academic periods – not during holidays, weekends, etc. and should be scheduled in normal business hours. The department head must approve exceptions. Please schedule for 3 hours.

Once you schedule your defense, please reach out to our IT team to let them know the date and if you will need any help in setting up for zoom, etc. Please note they will not automatically be there to set up unless you request help.

Students should coordinate a date with their committee members and finalize the date as early as possible to allow time for travel arrangements, final review, and comments by committee prior to defense date.

Students need to coordinate with their advisor's assistant to secure a room for their defense if on campus. The room scheduled must be available to the public and must accommodate a reasonably large number of people (25-30).

(1). If you are a full-time student currently registered for 48 units, there are no other financial responsibilities for you.

(2). If you are returning from ABS only to defend on campus, you will need to fill out a Return from ABS form, register and pay for [5 units of tuition](#). Please talk with your advisor to see if they will cover the on-campus cost. You can find the Tuition and Fee Effects under the “In Absentia Student Status Including the Final Semester Fees” at the link below:

<https://www.cmu.edu/policies/student-and-student-life/doctoral-student-status.html>

(3) Remote Defense - If your Defense is remote only (via Zoom), the student is responsible for setting up the zoom link, and completing the [“Consent to Publicly Livestream a Presentation”](#).

While an All But Dissertation student is In Absentia, no tuition will be assessed. The student will, however, [be responsible for all applicable fees](#). Please talk to your advisor to see if they will pay the remote defense fee.

Preparing your Thesis Document:

Please contact Catherine Copetas (copetas@cs.cmu.edu) for the following information:

1. Latex Template (See below for example of title page)
2. Request a TR number

3. Check for proper title page format (verify with Catherine prior to final submission)- See example at the end of the directions.

Thesis Defense will be announced to the public by the Program Administrator. Please send the following information to Alisha Roudebush no less than **10 business days prior to your defense date**:

1. Your name as it should appear on your diploma
2. Thesis Title
3. Date/Time/Location of defense presentation
4. Thesis Committee Members, names, titles, affiliations, and email addresses
5. Thesis Abstract – Less than 350 words describing the thesis
6. Pointer to Thesis for any outside faculty who wish to review (Website address, etc.)
7. Print and post 10 Posters to the SCS Community (TCS/Gates)
8. Send Alisha a PDF of Poster: [Defense Poster Template](#)
9. Zoom link, if applicable

SECTION 9: Department Policies & Protocols

9.1: Petition Procedures- for adding a course to current STAR list

The faculty have selected an initial set of approved courses in each category. These are subject to review from time to time to ensure that, if the course content changes, it remains consistent with the purpose of that STAR.

SE Ph.D. students may request that the faculty approve an additional course in one of the STAR categories. In general, if the request is approved, the course will be added to the list for other students to take for STAR credit. When a request is student-initiated, it is the student's responsibility to make a case supporting STAR status. Students should submit a request to the SE Ph.D. Program Director and the SE Ph.D. Program Administrator using the [STAR request form](#).

Given sufficient information, requests received by the faculty should generally receive a response within 2 weeks if the request is made during a regular academic semester. STAR credit should generally be requested at least **2 weeks before the end of the semester before taking a course**, and preferably 2 weeks before the beginning of the registration period. This ensures students can register for a course before it fills up and avoid spending time on a course that is not in the end approved.

Courses will not, in general, be approved in two categories, but instead will be approved in the category that best fits the primary emphasis of the course (if any). If any exception to

this principle is made, the student must choose which category to apply the course to and find a different course with which to fulfill the other requirement.

There is a precedent for approving an independent study for STAR credit, in the rare case where an appropriate independent study suits the student's needs better than any available course. The approval process is the same in this case, but the proposal submitted by the student should identify who would advise the independent study, what the output of the study will be and how it should be evaluated, the match to the appropriate STAR category, appropriateness of the course for Ph.D. study in terms of depth and engagement with research and should indicate advisor approval.

Course curricula may evolve over time, due to the advancing state of knowledge, the changing background and needs of students, or the strengths that a new instructor brings to bear on a course. Therefore, the faculty may re-examine STAR courses from time to time to verify the course continues to fulfill the requirements for a STAR. If it does not, STAR status may be withdrawn for future offerings of the course.

9.2: Department Policy for Withdrawing from a Course

Students must follow the University deadline for withdrawing from a course.
<https://www.cmu.edu/hub/registrar/course-changes/index.html>

Students taking undergraduate and master's level courses must follow the procedures and deadlines for adding, dropping, or withdrawing from courses as identified on the academic calendar. Information can be found at [Hub](#).

There is a separate calendar for doctoral level courses.

9.3: Master's Degree en route to a PhD Process

Ph.D. students may wish to have their progress in the program recognized by receiving a Master of Science in Software Engineering degree upon completion of an appropriate number of Ph.D. program requirements. These requirements serve to characterize the student's preparedness to develop a doctoral thesis proposal in the program.

Upon completing the Master of Science in Software Engineering degree, students should be able to:

- Demonstrate breadth of knowledge across three foundational areas in software engineering.
- Demonstrate the ability to identify, read, and understand relevant research literature, and to design a research study using an established research method.

The above learning objectives can be realized by the following course plan:

- Complete the Software Engineering Research Course (17-808)
- Participate in the Software Engineering Research Seminar (SSSG) each semester, unless excused by the Program Director due to a course conflict.
- Complete four additional courses, covering at least three of the four-STAR areas
- Complete 96 units of supervised research (17-997)
- Either complete an additional 24 units of research or coursework, or else serve as a teaching assistant for one semester
- Fulfill the writing requirement

The learning objectives and sample course plan are equivalent to a two-year master's program with no thesis option. All courses used to qualify for the master's degree must be taken at CMU, and no master's degree will be granted to a student who has previously received a master's degree from the CMU School of Computer Science. The degree is granted upon written request by the student to the SE Ph.D. Program Administrator once they have completed sufficient requirements to demonstrate the learning objectives.

Students may request this during the Doctoral Student Review Process that is held every Fall and Spring, by requesting it on their Student Statement.

9.4: Time Away from Academic Responsibilities

Research is a year-round activity, and students should expect to continue working on their research even when classes are out of session. Nevertheless, it is important for students' health and quality of life to take time off occasionally throughout the year. Students should meet with their advisor to discuss their advisor's expectations regarding vacation time (a.k.a. paid time off). In addition to university holidays (see the university calendar for days the university is closed), their advisor is expected to approve, at a minimum, 10 days of vacation per year. The seasonal pattern of every research group is different, so it is impossible to state general rules regarding timing of vacation. With the exception of an unexpected emergency situation, students should coordinate time off with sufficient advance notice to minimize impact to individual or research group progress. In the unusual circumstance of a conflict regarding vacation between a student and advisor, the department can mediate.

University Expectations Link: [University expectation](#).

SECTION 10: Grading & Evaluation

10.1: Grading Scale/System

Dept./College Grading Scale/System: A student's progress in the Ph.D. program is measured along multiple dimensions. One of these dimensions is a student's performance

in courses, and our expectation is that Ph.D. students earn a B- or better. Grades of C+ or below do not count towards program requirements.

Grades are just one dimension of student performance, and in fact are largely irrelevant for students who complete the program. To encourage students to place their primary focus on research over coursework, our general program policy is to record grades as pass/fail in the student's official transcript. A form to request pass/fail grading is available on the university HUB website. While we encourage all instructors to follow our general policy for SE Ph.D. students in their courses, instructors have the discretion to make their courses letter graded only, which overrides the general policy above for those specific courses.

SE Ph.D. students may register for graduate or undergraduate courses in other departments. However, where possible, they should register for these courses with pass/fail grading.

10.2: Department Policy on Pass/Fail, Satisfactory/Unsatisfactory

If a student requests a Pass/Fail as a grade, they must receive a B- or better in order to receive credit for the course.

10.3: Independent Study/Directed Research

If you would like credit for an Independent Study and count for credit towards your Ph.D. degree courses in Software Engineering, you must fill out the SE Ph.D Independent Study Prospectus. You will need your advisors approval, and Ph.D. directors approval. Please reach out to Connie Herold, Programs Manager for the form. Students will register under **17-806**.

Students must request credit prior to the start of the semester of the Independent Study course by completing an Independent Study form. Please reach out to Connie or Alisha for the form.

10.4: Doctoral Student Review (DSR)

Evaluation and feedback on a student's progress are important both to the student and to the faculty. Students need information on their overall progress to make long-range plans. The faculty need to make evaluations to advise students, to make support decisions, and to write recommendations to potential employers.

The faculty meet at the end of each semester to make a formal evaluation of each student in the Ph.D. program. This meeting is called the "Doctoral Student Review" meeting. The purpose of having all the faculty meet together to discuss all the students is to ensure

uniformity and consistency in evaluation across all the different areas, by all the different advisors, throughout the years of the SE Ph.D. program as it inevitably changes.

The faculty measure each student's progress against the goal of completing the Ph.D. program in a reasonable period of time. The evaluation considers all components of the program using indicators and information sources described below. Requirements need not be fulfilled in any particular order, but each student must show reasonable progress each semester toward satisfying the full set of requirements. Because the critical path to completing the Ph.D. is research, making early and regular research progress is the most important consideration. Through a Doctoral Student Review letter, the faculty inform students of the results of this evaluation, which may include specific recommendations for future work or requirements that must be met for continued participation in the program.

Components and Indicators

In their evaluation, the faculty consider the following components, though naturally only some of these components will be applicable in any given semester; they are not equally important at every stage of a student's career, and each student will progress through the requirements as suits his or her individual needs:

- *Directed research*: Evaluated by research supervisor and other collaborating faculty.
- *Courses taken*: Evaluated by the course instructor—brief prose evaluation/summary grade.
- *Teaching*: Evaluated by the course instructor and two different teaching evaluation forms (One filled out by the course instructor and the other filled out by students, where appropriate).
- *Skills*: Writing and speaking, by the relevant faculty and forms.
- *Thesis*: Status summarized by the thesis advisor and comment by members of the thesis committee.
- *Other*: Lectures given, papers written, etc. Evaluated by cognizant faculty.

The faculty's primary sources of information about the student are the student's advisor and the student's statement. The advisor is responsible for assembling the above information and presenting it at the faculty meeting. The student should make sure the advisor is informed about participation in activities and research progress made during the semester. Each student is asked to submit a summary of this information to the advisor at the end of each semester—the Student Statement for Doctoral Review at <https://gsaudit.cs.cmu.edu>. This statement is used as student input to the evaluation process and as factual information on activities and becomes part of the internal student record. It is strongly recommended that the student and advisor meet prior to the faculty meeting to review the information provided in this statement.

Outcomes and Recommendations

Based on the above information, the faculty decide whether a student is making satisfactory progress in the Ph.D. program. If so, the faculty usually suggest goals for the student to achieve over the next semester. If not, the faculty make more rigid demands of

the student; these may be long-term (e.g., finish your thesis within 1-1/2 years) or short-term (e.g., select and complete one or more specific courses next semester; prepare a thesis proposal by next Doctoral Review meeting).

Ultimately, permission to continue in the Ph.D. program is contingent on whether or not the student continues to make satisfactory progress toward the degree. If a student is not making satisfactory progress, the faculty may choose to dismiss the student from the program.

The faculty also decides whether financial support (including tuition and stipend) should be continued for each student if the student does not make satisfactory progress. Termination of support does not always mean termination from the program.

If you wish to appeal any decisions please follow the Graduate Student Appeal and Grievance Procedures below:

[Summary of Graduate Student Appeal and Grievance Procedures](#)

10.5: Satisfactory Academic Standing

Students are evaluated at the end of every semester through the Doctoral Review Process described in [Section 10.4: Doctoral Student Review-\(DSR\)](#). If a student is not making satisfactory progress, the faculty may choose to dismiss the student from the program. Typically, when students are first informed that their progress is not satisfactory through a DSR evaluation, the faculty provides concrete requirements that must be met in the following semester to be considered satisfactory progress in the next evaluation.

[See Summary of Graduate Student Appeal and Grievance Procedures](#)

SECTION 11: Funding & Financial Support

11.1: Health Insurance Requirement

Student (SHIP) Health Insurance Coverage: If you elect to enroll in Carnegie Mellon University's Student Health Insurance Plan (SHIP), the University will cover 100% percent of the premium cost for your individual coverage under SHIP. While you will have the opportunity to purchase partner, spouse or dependent coverage under the SHIP plan, the University's support will be 100% percent of the individual coverage amount. Please note

that if you wish to elect the required health insurance coverage under an alternate plan, you will not be eligible for the University support referenced here.

<https://www.cmu.edu/health-services/student-insurance/plans.html>

<https://www.cmu.edu/sfs/billing/payments/monthly-plan/index.html>

The HUB's website (<https://www.cmu.edu/hub/new-grad/enrollment-finances.html>) also has information specific to health insurance coverage and waivers for doctoral students.

11.2: Additional Sources of Internal & External Financial Support

We encourage students to seek their own external funding since often the award is prestigious (e.g., NSF or Hertz) or the source provides an opportunity to make professional connections (e.g., an industrial fellowship).

If a student receives an external fellowship/scholarship, they must notify the SC Ph.D. Program Administrator. The Institute supplements the stipends of students with an outside fellowship to meet the stipends of students with internal funding, plus a bonus: in a year when a student brings in a fellowship worth \$X, that student gets 1% of X added to their stipend each month, for a total of 12% of X if the student *takes a stipend all 12 months*.

The department also pays a dependency allowance that is 10% of the S3D monthly base stipend per eligible dependent provided that your spouse or qualifying domestic partner earns less than 15% of the stipend amount.

11.3: Department Policy on Outside Employment

International students must contact the Office of International Education regarding their ability to hold employment.

Working (i.e., doing anything for pay) either within or externally to the university, beyond your responsibilities as a teaching assistant or research assistant, is a privilege, not a right.

Work is permitted under exceptional circumstances, and with the written approval of the student's advisor (or with the written approval of all advisors, in case of multiple advisors) and of the Software Engineering program director, the following is permissible:

- Outside consulting will be limited in time, and should consist of no more than **8 (eight) hours of consulting per week**.
- The purpose of the consulting must be closely related to the research area of the student and be clearly aligned with the student's thesis goals.
- The consulting project must be self-contained. In particular, it cannot use any Carnegie Mellon University intellectual property.

- Likewise, the project cannot make any use of university facilities (e.g., buildings), or equipment (e.g., laptops, or computing resources, including networking and email).
- Approval from the advisor(s) and the program director must be renewed every semester.

Students are responsible for understanding the tax implications and are encouraged to consult with a tax advisor if needed. International students should consult with OIE, as outside employment is usually forbidden.

Students who are not on a leave of absence must abide by all Carnegie Mellon rules and policies while performing these activities, including, but not limited to, rules governing potential conflicts of interest and ownership of intellectual property.

We require that students limit employment to follow university and government rules, but the more important principle is maintaining adequate focus and creative energy for the research that is at the core of the Ph.D. degree.

SECTION 12: Graduate Student Handbook

Graduate Student Handbook Suite.

<https://www.cmu.edu/graduate/resources/index.html>
