Applications

Applications for Software Engineering are considered for every quarter. All application materials should be sent to Graduate Admissions by the stated deadline for that quarter. Late applicants can only be considered as non-matriculating students and on a space available basis

Application Deadlines:

Quarter	Domestic	International
Fall:	July 20	April 1
Winter:	November 20	September 1
Spring:	February 20	December 1

The Application procedure is:

- Obtain a complete application packet from Graduate Admissions or our website or apply online through Graduate Admissions website
- Have two official transcripts from each institution you have attended mailed directly to Graduate Admissions.
- Complete the software engineering applicant information form.
- Distribute the provided recommendation forms and envelopes to two individuals who will serve as professional references or use the online reference function.
- Complete a professional autobiographical statement describing the relevance of your experiences to the MSE program and your career objectives in completing the program.
- Request that official GRE scores be sent directly to Graduate Admissions.

Send to Graduate Admissions at Seattle University. Parts of the application may be sent as available. Test scores and tran scripts typically can be sent after filing an application.

Degree Requirements

Minimum requirements for the degree are 47 graduate credits. These include 29 hours of required courses, nine hours of elective courses, and nine hours of software engineering project. Required courses in computing (SEGR 5010 and 503) may be waived for students passing a waiver exam and replaced by additional electives. A maximum of 12 credits taken in non-matriculated status may be applied to this program. All degree requirements must be completed within six years after course work has begun.

Required Core Courses

29 Credits Including:

- SEGR 5010 - Object-Oriented Concepts
- SEGR 5030 - Data Structures and Algorithms
- SEGR 5110 - Software Requirements
- SEGR 52200 - Software Architecture and Design I
- SEGR 5220 - Human-Computer Interaction
- SEGR 51200 - Software Construction
- SEGR 5250 - Data Modeling
- SEGR 5120 - Software Project Planning
- SEGR 5240 - Software Architecture and Design II
- SEGR 5800 - Ethics and Professional Issues in Computing

Elective Courses

9 credits from the following:

- SEGR 5150 - Software Economics
- SEGR 5210 - Software Testing
- SEGR 5160 - Software Project Management
- SEGR 5180 - Software Quality Assurance
- SEGR 5410 - Applied Formal Methods
- SEGR 5220 - Distributed Computing
- SEGR 5530 - Embedded Systems
- SEGR 5610 - Artificial Intelligence
- SEGR 5710 - Software Security

Note: In addition to the above courses, students may take a maximum of three credits of Special Topics (SEGR 5910, 592, 593), Software Engineering Seminar (SEGR 594), Independent Study (SEGR 5960, 597, 598), or Internship (SEGR 5950) for elective credit, without department approval. Students may take up to six credits of graduate courses from other programs with department approval. Approved MBA courses include such topics as leadership, entrepreneurship, high tech marketing, and management of change. Approved CPSC courses may also be taken to fulfill the elective requirement

Required Project Courses

- SEGR 5850 Software Engineering Project 1.....3 credits
- SEGR 5860 Software Engineering Project 2.....3 credits
- SEGR 5870 Software Engineering Project 3.....3 credits

Note: Satisfactory performance in the three-quarter software engineering project sequence (SEGR 5850 , 586 , and 587) is required of all MSE students. Students are grouped into teams that complete a significant software project. Students who are not full-time are expected to take only the project sequence in their final year. Students are encouraged not to begin the software engineering project sequence unless SEGR 5870 will be their final course in the program. All participants in the project sequence will be required to sign a Seattle University intellectual property (IP) and computer project room agreement, and may be required to sign an IP agreement with the project sponsor.

Typical Program Schedules

Three-year plan with SEGR 501 and SEGR 503 required

	Fall	Winter	Spring	Summer
Year 1	SEGR 501 SEGR 520	SEGR 503 SEGR 521	SEGR 525 SEGR 531	SEGR 580
Year 2	SEGR 524 Elective	SEGR 526 Elective	SEGR 523 Elective	
Year 3	SEGR 585	SEGR 586	SEGR 587	

Three-year plan with SEGR 501 and SEGR 503 waived

	Fall	Winter	Spring	Summer
Year 1	SEGR 520 Elective	SEGR 521 Elective	SEGR 525 SEGR 531	SEGR 580
Year 2	SEGR 524 Elective	SEGR 526 Elective	SEGR 523 Elective	
Year 3	SEGR 585	SEGR 586	SEGR 587	

Two-year plan for fulltime students

	Fall	Winter	Spring	Summer
Year 1	SEGR 501 SEGR 520 Elective	SEGR 503 SEGR 521 SEGR 526	SEGR 525 SEGR 531 SEGR 523	SEGR 580
Year 2	SEGR 585 SEGR 524 Elective	SEGR 586 Elective Elective	SEGR 587 SEGR 523 Elective	

*International students may need to take additional elective credits to maintain their student visa status.

Faculty

Richard LeBlanc, PhD, University of Wisconsin, Chair
Roshanak Roshandel, PhD, University of Southern California
Adair Dingle, PhD, University of Texas at Dallas
Eric Larson, PhD, University of Michigan
Lin Li, PhD, Clemson University
Ben Tribelhorn, PhD, Oregon State University
Yingwu Zhu, PhD, University of Cincinnati
Jeff Gilles, MS, University of Iowa

Our Adjunct faculty are working industry professionals drawn from many of the corporations in the Seattle area, such as Microsoft and Boeing.

Seattle University

Seattle University was founded in 1891 by Jesuit priests. As one of 28 Jesuit colleges in the United states, we are dedicated to teaching and educating for leadership and service. While we have grown to be the largest independent institution in the Pacific Northwest, we still enjoy a small-college atmosphere that balances academic challenges with personal, individual attention. We offer a diverse and quality curriculum to students of every religion and culture, and our faculty members are committed to a strong student-teacher relationship.

For Additional Information

- **MSE Program:**

Department of Computer Science/Software Engineering,
Seattle University,
901 12th Ave, P.O. Box 222000, EGRN 526,
Seattle, WA 98122-1090
206-296-5428, Fax 206-296-5518
Email: mse@seattleu.edu
Web: www.seattleu.edu/scieng/mse

- **Graduate Admissions:**

Seattle University
901 12th Ave, P.O. Box 222000
Seattle, WA 98122.
206-296-2000, Fax 206-296-5656.
Email: grad-admissions@seattleu.edu.
Web: www.seattleu.edu

"Seattle University does not discriminate on the basis of race, color, religion, sex, national origin, age, disability, marital status, sexual or political orientation, or status as a Vietnam-era or special disabled veteran in the administration of any of its education or admission policies, scholarship and loan programs, athletics, and other school-administered policies and programs, or in its employment policies and practices. All University policies, practices and procedures are administered in a manner consistent with Seattle University's Catholic and Jesuit identity and character. Inquiries about the non-discrimination policy may be directed to the University's EEO Officer and Title IX coordinator, University Services Building 107, (206) 296-5870."

Course Descriptions

SEGR 5010 Object-Oriented Development
Introduction to principles and techniques of object-oriented development. In-depth understanding of object-oriented concepts, including abstraction, encapsulation, inheritance, subtypes, subclasses, polymorphism and support for re-use.

SEGR 5030 Data Structures & Algorithms
Basic strategies of algorithm design: top-down design, divide and conquer, average and worst-case complexity, asymptotic costs, simple recurrence relations. Choice of appropriate data structures such as arrays, stacks, queues, trees, heaps, graphs, hash tables, etc. Applications to sorting and searching. Introduction to discrete optimization algorithm: dynamic programming, greedy algorithms. Prerequisite: SEGR 5010.

SEGR 5110 Software Requirements
Definition of types of requirements; elicitation processes; analysis techniques; documentation methods; validation methods and security issues.

SEGR 5230 Software Architecture & Design I
Fundamentals of software architecture design, modeling, and analysis. Impact of architectural design choices on quality. Architectural styles, architectural evolution, domain specific software engineering, and software product families.

SEGR 5220 Human-Computer Interaction
Relationship of user interface design to human-computer interaction. Interface quality and methods of evaluation; dimensions of interface variability; dialogue genre, tools and techniques; user-centered design, task analysis and implementation.

SEGR 5260 Software Construction
High-quality life cycle practices; key programming decisions; design practices; classes; defensive programming; using variables and types; collaborative work; developer testing; re-factoring, code-tuning and system considerations; layout; style and documentation. Prerequisite: SEGR 5240 & 5250.

SEGR 5250 Data Modeling
Data design and modeling for transactional and analytical systems; the use of relational and object databases; database security issues; database performance issues; and data access. Pre or co-equisites: SEGR 5030.

SEGR 5120 Software Project Planning
Initiation and scope definition; software project planning process; deliverables; effort, schedule and cost estimation; resource allocation; risk management; quality management; and management; project planning and estimating tools.

SEGR 5150 Software Economics
Business decision-making process; time value of money; comparing proposals; for-profit decisions; not-for-profit decisions; estimation; risk and uncertainty; buy vs. build decisions, outsourcing costs and return. Prerequisite: SEGR 5110 and SEGR 5120.

SEGR 5240 Software Architecture & Design II
Fundamentals of detailed design and architectural refinement using model consistency, traceability, architectural patterns, and model driven development. Integration with exisiting systems. Architecture and design recovery. Prerequisite: SEGR 5220.

SEGR 5210 Software Testing
Test levels (unit, integration and system); test objectives testing techniques, measures and process (planning, test case generation and defect tracking). Students may not get credit for both SEGR 5210 and CPSC 5210. Prerequisite: SEGR 5030.

SEGR 5160 Software Project Management
Management of software projects (implementation of plans, supplier and subcontractor contract management, implementation of measurement process, monitoring and controlling process, and reporting); review and evaluation (determining satisfaction of requirements, reviewing and evaluating performance); project closure; software engineering measurement; process improvement. Prerequisite: SEGR 5120.

SEGR 5180 Software Quality Assurance
Software engineering process (implementation and change, definition and measurement); software quality (fundamentals, process and practical considerations); Capability Maturity Model (CMM) for software. Evaluate current projects. Prerequisite: SEGR 5110 and SEGR 5120.

SEGR 5410 Applied Formal Methods
Formal techniques for building reliable systems. Use of abstractions for concisely and precisely defining system behavior. Formal logic and proof techniques for verifying the correctness of programs. Hierarchies of abstractions, state transition models, Petri Nets, communicating processes. Operational and definitional specification languages. Prerequisite: SEGR 5030.

SEGR 5220 Distributed Computing
Design and analysis of distributed systems, distributed objects, middleware, quality of service in distributed systems and emerging topics in distributed computing. Students may not get credit for both SEGR 5220 and CPSC 5220. Prerequisite: SEGR 5030.

SEGR 5530 Embedded Systems
Methods, techniques, and tools for design, analysis, and development of embedded systems; real-time concerns; performance, distribution, dynamism, and mobility concerns. Prerequisite: SEGR 5030.

SEGR 5610 Artificial Intelligence
Survey of artificial intelligence as it applies to software engineering. Acquisition and representation of knowledge. Search strategies. Selected applications, such as natural language processing, image recognition, planning, neural nets, and expert systems. Students may not get credit for both SEGR 5610 and CPSC 5610. Prerequisite: SEGR 5030.

SEGR 5800 Ethics & Professional Issues in Computing
Examination of the role and impact of information and communication technology in society, with emphasis on ethical, professional, and public policy issues. Prerequisite: Graduate standing in the MSE program.

SEGR 5710 Software Security
Introduction to software dependability, software security vs. security systems, security concerns in various phases of software development life cycle, risk management framework and threat modeling, security analysis (worms, viruses, physical leaks, root kits, Trojans, etc.), common exploits, legal and ethical issues and emerging topics in software security. Students may not get credit for both SEGR 5710 and CPSC 5710.

SEGR 5910 Special Topics
May include such topics as artificial intelligence, data privacy, embedded systems, ethics of computing, programming languages, real-time systems, soft-ware metrics, and other software engineering topics. A maximum of three credits of 5910 courses may be taken toward the MSE degree without department approval. Prerequisite: instructor permission.

SEGR 5950 Internship
By permission only. Supervised practical training combined with academic studies in which students apply and develop their software engineering knowledge and skills working for a business or non-profit institution. Students are required to conduct related academic studies under the supervision of a faculty advisor. A maximum of three credits of 59x courses may be taken toward the MSE degree without department approval.

SEGR 5960 Independent Study
Independent research and in-depth study of topics under the supervision of a faculty advisor. A maximum of three credits of 59x courses may be taken toward the MSE degree without department approval. Prerequisite: instructor permission.

Required Project Courses

SEGR 5850, 5860, 5870 Software Engineering Project 1,2 &3
A three-quarter sequence in which students are grouped into teams, and each team completes a year-long software project. The sequence begins in the fall and ends spring quarter. Prerequisite: completion of core courses and departmental approval. All participants in the Project sequence will be required to sign a Seattle University intellectual property (IP) and computer project room agreement, and may be required to sign an IP agreement with the Project Sponsor.

COMPUTER SCIENCE AND SOFTWARE ENGINEERING

SEATTLE UNIVERSITY

Master of Software Engineering

Why Graduate Study in Software Engineering?

The software industry requires highly skilled people to develop and maintain the computer programs that support our society: programs that do everything from managing financial systems to controlling high-performance aircraft.

Graduate study in software engineering offers the software professional the mix of managerial and technical education necessary to understand and apply advanced software engineering principles vital to industry.

Students enter the master of software engineering (MSE) program with a variety of interests and experiences, but they share a common goal: to improve their mastery of software development and maintenance. By studying a comprehensive range of topics, MSE students acquire the practical and theoretical skills to confidently address difficult software issues. Our graduates enjoy the increased financial rewards, heightened responsibilities, and enhanced job challenges that accompany an MSE degree.

Our Students

The typical MSE student is a software professional with at least two years of software development or maintenance experience. Many MSE students have a traditional computer science education; other students come to the program from diverse backgrounds, including various engineering disciplines, mathematics, the physical and biological sciences, and business. This combination of experience and diversity promotes a cross-pollination of issues and ideas throughout the program.

Our Tradition

Seattle University has long been a leader in software engineering education. It established its pioneering MSE program in 1979, and awarded the world's first MSE degree in 1982.
Our graduate faculty have extensive industrial experience as well as strong academic credentials. Our curriculum has been shaped by close contacts with local industry, ensuring that courses are relevant to the workplace and that projects are tailored to the professional needs of our students.

Our Program

The MSE program at Seattle University is designed for working professionals. All classes are typically offered in the evenings. The courses are aligned with the Guide to the Software Engineering Body of Knowledge, (SWEBOK), a project of the IEEE Computer Society Professional Practices Committee. The program builds on the computing experience of its students by providing course work on a variety of software engineering and computer science topics, with an emphasis on team work and disciplined approach to problem solving. We offer a balanced core curriculum of technical and managerial courses, and a choice of elective streams to address areas of personal interest. Principles and techniques learned throughout the program are integrated into a year-long software project as the capstone experience.

Admission Requirements

- A four-year Bachelor's degree, in computer science or a related discipline, earned at a regionally accredited school. Completed Application for Graduate Admission and non-refundable application fee (waived for Seattle University Alumni
- Minimum GPA of 3.00 calculated with coursework from all post-secondary educational institutions attended in the last 90 quarter/60 semester credits of your bachelor's degree and any post-baccalaureate coursework.
- Current resume reflecting at least two years of professional ex perience in software development or maintenance.
- Completed Software Engineering Applicant Information form noting working knowledge of at least one programming language, such as C++, Java, or C#.
- Professional autobiography.
- Two professional recommendations using the MSE recommendation forms. At least one of the recommendations should be prepared by a supervisor.
- Official Graduate Record Exam (GRE) General Test scores from within the past five years. The Subject Test in Computer Science is optional.
- If English is not applicant's native language, official TOEFL paper-based score of 580 (92 internet-based or 237 computer-based) required. Equivalent MELAB and IELTS scores also accepted to determine English Proficiency.