SOFTWARE ENGINEERING (SWE)

200 Level Courses

SWE 205: Software Usability Analysis and Design. 3 credits.
Principles of user interface design. Concepts for objectively and quantitatively assessing the usability of software user interfaces. Outcomes include knowledge of quantitative engineering principles for designing usable software interfaces and an understanding that usability is more important than efficiency for almost all modern software projects, and often the primary factor that leads to product success. Major topics include cognitive models for human perceptions and needs, which are used as a basis for analytical and critical thinking about user interfaces; specific engineering principles for designing usable menus, forms, command languages, web sites, graphical user interfaces and web-based user interfaces. Assessments will include written analytical evaluations of existing user interfaces, exams, and HTML-based design projects. Offered by Computer Science. Limited to two attempts.

Registration Restrictions:
Required Prerequisite: ENGH 101C.
C Requires minimum grade of C.

Students with the terminated from VSE major attribute may not enroll.

Schedule Type: Lecture

300 Level Courses

SWE 301: Internship Preparation. 0 credits.
Preparation for Internship Educational Experience. Intended for, but not limited to, students planning internships in the Applied Computer Science Software Engineering Program. Internship employment opportunities. Basic interview skills. Techniques for applying academic knowledge to practical software development. Techniques for extracting knowledge from practical experience. Peer presentation from students who have completed internships. Offered by Computer Science. Limited to two attempts.

Recommended Prerequisite: Limited to ACS or CS majors with junior standing or permission of instructor.

Registration Restrictions:
Students with the terminated from VSE major attribute may not enroll.

Schedule Type: Lecture

SWE 321: Software Engineering. 3 credits.
An introduction to concepts, methods, and tools for the creation of large-scale software systems. Methods, tools, notations, and validation techniques to analyze, specify, prototype, and maintain software requirements. Introduction to object-oriented requirements modeling, including use of case modeling, static modeling, and dynamic modeling using the Unified Modeling Language (UML) notation. Concepts and methods for the design of large-scale software systems. Fundamental design concepts and design notations are introduced. A study of object-oriented analysis and design modeling using the UML notation. Students participate in a group project on software requirements, specification, and object-oriented software design. Offered by Computer Science. Limited to two attempts. Equivalent to CS 321.

Registration Restrictions:
Required Prerequisites: ((ENGH 302C) or ((HNRS 110C) and (HNRS 122C, 130C, 131C, 230C or 240C)) and (CS 310C)).
C Requires minimum grade of C.

Enrollment is limited to students with a major in Applied Computer Science, Computer Science or Systems Engineering.
Enrollment is limited to students with a minor in Computer Science or Software Engineering.
Enrollment limited to students in a Bachelor of Science or Post-Baccalaureate Certificate degrees.

Students with the terminated from VSE major attribute may not enroll.

Schedule Type: Lecture

SWE 332: Object-Oriented Software Design and Implementation. 3 credits.
In-depth study of software design and implementation using a modern, object-oriented language with support for graphical user interfaces and complex data structures. Topics covered are specifications, design patterns, and abstraction techniques, including typing, access control, inheritance, and polymorphism. Students will learn the proper engineering use of techniques such as information hiding, classes, objects, inheritance, exception handling, event-based systems, and concurrency. Offered by Computer Science. Limited to two attempts. Equivalent to CS 332.

Registration Restrictions:
Required Prerequisite: (CS 310C).
C Requires minimum grade of C.

Students with the terminated from VSE major attribute may not enroll.

Schedule Type: Lecture

400 Level Courses

SWE 401: Internship Reflection. 1 credit.
Reflection on Internship Educational Experience. Intended for, but not limited to, students completing internships in the Applied Computer Science Software Engineering Program. Analysis of techniques for applying academic knowledge to practical software development. Analysis of techniques for extracting knowledge from practical experience. Student presentations summarizing internships relating them to academic program goals. Offered by Computer Science. Limited to two attempts.

Recommended Prerequisite: Completion of internship.

Registration Restrictions:
Required Prerequisite: SWE 301C.
C Requires minimum grade of C.

Students with the terminated from VSE major attribute may not enroll.

Schedule Type: Lecture

SWE 432: Design and Implementation of Software for the Web. 3 credits.
Teaches how to develop software for web applications. Covers client-server computing, theories of usable graphical user interfaces, and models for web-based information retrieval and processing. Goals are to understand how to design usable software interfaces and implement them on web, learn how to build software that accepts information from users across web and returns data to user, and understand how to interact with database engines to store and retrieve information. Specific
Software Engineering (SWE)

Topics are HTML, CGI programming, Java, Java applets, Javascripts, and Java servlets. Offered by Computer Science. Limited to two attempts.

Registration Restrictions:
Required Prerequisites: (MATH 125\textsuperscript{C}) and (CS 310\textsuperscript{C}).
\textsuperscript{C} Requires minimum grade of C.

Students with the terminated from VSE major attribute may not enroll.

Schedule Type: Lecture

SWE 437: Software Testing and Maintenance. 3 credits.
Concepts and techniques for testing and modifying software in evolving environments. Topics include software testing at the unit, module, subsystem, and system levels; developer testing; automatic and manual techniques for generating test data; testing concurrent and distributed software; designing and implementing software to increase maintainability and reuse; evaluating software for change; and validating software changes. Offered by Computer Science. Limited to two attempts.

Registration Restrictions:
Required Prerequisites: (MATH 125\textsuperscript{C} and CS 310\textsuperscript{C}).
\textsuperscript{C} Requires minimum grade of C.

Students with the terminated from VSE major attribute may not enroll.

Schedule Type: Lecture

SWE 443: Software Architectures. 3 credits.
Teaches how to design, understand, and evaluate software systems at an architectural level of abstraction. By end of course, students will be able to recognize major architectural styles in existing software systems, describe a system's architecture accurately, generate architectural alternatives to address a problem and choose from among them, design a medium-size software system that satisfies a specification of requirements, use existing tools to expedite software design, and evaluate the suitability of a given architecture in meeting a set of system requirements. Offered by Computer Science. Limited to two attempts.

Registration Restrictions:
Required Prerequisites: CS 321\textsuperscript{C}, 421\textsuperscript{C}, SWE 321\textsuperscript{C} or 421\textsuperscript{C}.
\textsuperscript{C} Requires minimum grade of C.

Students with the terminated from VSE major attribute may not enroll.

Schedule Type: Lecture

500 Level Courses

SWE 510: Object-Oriented Programming in Java. 3 credits.
Introduces students to programming in the Java language. Topics include problem-solving methods and algorithm development, program structures, abstract data types, simple data and file structures and program development in a modular, object-oriented manner. Introductory use of O0 language features, including data hiding, inheritance, polymorphism, and exception handling. Goals include design and development of Java classes and class type hierarchies. An introduction to Java servlets and applets is included. Emphasis on program development is reinforced through several programming projects. Notes: Credit cannot be applied to a graduate degree in the Volgenau School or the BS degree in computer science. Offered by Computer Science. May not be repeated for credit.

Recommended Prerequisite: SWE foundation courses or equivalent.

Registration Restrictions:
Recommended Prerequisite: Undergraduate courses or equivalent knowledge in programming in a high-level language.

Schedule Type: Lecture

SWE 619: Object-Oriented Software Specification and Construction. 3 credits.
In-depth study of software construction using modern, object-oriented language with support for graphical user interfaces and complex data structures. Specifications, design patterns, and abstraction techniques, including procedural, data, iteration, type, and polymorphic. Information hiding, classes, objects, and inheritance. Exception handling, event-based systems, and concurrency. Offered by Computer Science. May not be repeated for credit.

Recommended Prerequisite: SWE foundation courses or equivalent.

Schedule Type: Lecture

600 Level Courses

SWE 620: Software Requirements Analysis and Specification. 3 credits.
In-depth study of object-oriented requirements modeling, including use case modeling, static modeling and dynamic modeling with Unified Modeling Language (UML) notation. Students participate in group project on software requirements and specification using modern method. Offered by Computer Science. May not be repeated for credit.

Recommended Prerequisite: SWE foundation courses or equivalent.

Registration Restrictions:
Recommended Prerequisite: Undergraduate courses or equivalent knowledge in programming in a high-level language.

Schedule Type: Lecture

SWE 621: Software Modeling and Architectural Design. 3 credits.
Concepts and methods for the architectural design of large-scale software systems are presented. Fundamental design concepts and
design notations are introduced. Concepts of requirements analysis and specification are taught. Several design methods are presented and compared. In-depth study of object-oriented analysis and design modeling using the Unified Modeling Language (UML) notation. Students participate in a group project on software requirements, analysis, and design modeling. Offered by Computer Science. May not be repeated for credit.

**Recommended Prerequisite:** SWE foundation courses or equivalent.

**Registration Restrictions:**
Enrollment limited to students with a class of Advanced to Candidacy, Graduate, Non Degree or Senior Plus.

Enrollment is limited to Graduate, Non-Degree or Undergraduate level students.

Students in a Non-Degree Undergraduate degree may not enroll.

Enrollment limited to students in the College of Science, Schar School of Policy and Gov or Volgenau School of Engineering colleges.

**Schedule Type:** Lecture

**SWE 622: Distributed Software Engineering.** 3 credits.
Hands-on introduction to techniques and programming interfaces for distributed software engineering. Networking protocols at several layers. Construction of distributed and concurrent software using network protocol services. Applications of Internet and web-based software. Offered by Computer Science. May not be repeated for credit.

**Recommended Prerequisite:** SWE foundation courses or equivalent.

**Registration Restrictions:**
Enrollment limited to students with a class of Advanced to Candidacy, Graduate, Non Degree or Senior Plus.

Enrollment is limited to Graduate, Non-Degree or Undergraduate level students.

Students in a Non-Degree Undergraduate degree may not enroll.

Enrollment limited to students in the College of Science, Schar School of Policy and Gov or Volgenau School of Engineering colleges.

**Schedule Type:** Lecture

**SWE 625: Software Project Management.** 3 credits.
Lifecycle and process models; process metrics; planning for a software project; mechanisms for monitoring and controlling schedule, budget, quality, and productivity; and leadership, motivation, and team building. Offered by Computer Science. May not be repeated for credit.

**Recommended Prerequisite:** SWE foundation courses or equivalent.

**Registration Restrictions:**
Enrollment limited to students with a class of Advanced to Candidacy, Graduate, Non Degree or Senior Plus.

Enrollment is limited to Graduate, Non-Degree or Undergraduate level students.

Students in a Non-Degree Undergraduate degree may not enroll.

Enrollment limited to students in the College of Science, Schar School of Policy and Gov or Volgenau School of Engineering colleges.

**Schedule Type:** Lecture

**SWE 626: Software Project Laboratory.** 3 credits.
Covers requirements analysis, design, implementation, and management of software development project. Students work in teams to develop or modify software product, applying sound principles of software engineering. Uses both industrial and academic standards to assess quality of work products. Offered by Computer Science. May not be repeated for credit.

**Recommended Prerequisite:** SWE 619,620, and 621; or permission of instructor.

**Registration Restrictions:**
Enrollment limited to students with a class of Advanced to Candidacy, Graduate, Non Degree or Senior Plus.

Enrollment is limited to Graduate, Non-Degree or Undergraduate level students.

Students in a Non-Degree Undergraduate degree may not enroll.

Enrollment limited to students in the College of Science, Schar School of Policy and Gov or Volgenau School of Engineering colleges.

**Schedule Type:** Lecture

**SWE 631: Software Design Patterns.** 3 credits.

**Recommended Prerequisite:** SWE 621.

**Registration Restrictions:**
Enrollment limited to students with a class of Advanced to Candidacy, Graduate, Non Degree or Senior Plus.

Enrollment is limited to Graduate, Non-Degree or Undergraduate level students.

Students in a Non-Degree Undergraduate degree may not enroll.

Enrollment limited to students in the College of Science, Schar School of Policy and Gov or Volgenau School of Engineering colleges.

**Schedule Type:** Lecture

**SWE 632: User Interface Design and Development.** 3 credits.
Principles of user interface design, development, and programming. Includes user psychology and cognitive science, menu system design, command language design, icon and window design, graphical user interfaces, web-based user interfaces. Offered by Computer Science. May not be repeated for credit.

**Recommended Prerequisite:** SWE 619, or CS 540 and 571, or permission of instructor.

**Registration Restrictions:**
Enrollment limited to students with a class of Advanced to Candidacy, Graduate, Non Degree or Senior Plus.
Enrollment is limited to Graduate, Non-Degree or Undergraduate level students.

Students in a Non-Degree Undergraduate degree may not enroll.

Enrollment limited to students in the College of Science, Schar School of Policy and Gov or Volgenau School of Engineering colleges.

**Schedule Type:** Lecture

**SWE 637: Software Testing.** 3 credits.

Students learn to test software effectively. Programmers learn practical ways to design high quality tests during all phases of software development. Students learn the theory behind criteria-based test design and to apply that theory in practice. Topics include test design, test automation, test coverage criteria, and how to test software in cutting-edge software development environments. Offered by Computer Science. May not be repeated for credit.

**Recommended Prerequisite:** SWE 619, or permission of instructor.

**Registration Restrictions:**
Enrollment limited to students with a class of Advanced to Candidacy, Graduate, Non Degree or Senior Plus.

Enrollment is limited to Graduate, Non-Degree or Undergraduate level students.

Students in a Non-Degree Undergraduate degree may not enroll.

Enrollment limited to students in the College of Science, Schar School of Policy and Gov or Volgenau School of Engineering colleges.

**Schedule Type:** Lecture

**SWE 642: Software Engineering for the World Wide Web.** 3 credits.

Detailed study of engineering methods and technologies for building highly interactive web sites for e-commerce and other web-based applications. Presents engineering principles for building web sites that exhibit high reliability, usability, security, availability, scalability, and maintainability. Teaches methods such as clientserver programming, component-based software development, middleware, and reusable components. Offered by Computer Science. May not be repeated for credit.

**Recommended Prerequisite:** SWE 619 or CS 540 and CS 571, or permission of instructor.

**Registration Restrictions:**
Enrollment limited to students with a class of Advanced to Candidacy, Graduate, Non Degree or Senior Plus.

Enrollment is limited to Graduate, Non-Degree or Undergraduate level students.

Students in a Non-Degree Undergraduate degree may not enroll.

Enrollment limited to students in the College of Science, Schar School of Policy and Gov or Volgenau School of Engineering colleges.

**Schedule Type:** Lecture

**SWE 645: Component-Based Software Development.** 3 credits.

Introduces concepts and foundations of software component and component-based software. Detailed study of engineering principles of modeling, designing, implementing, testing, and deploying component-based software. Also explores state-of-the-art component technologies. Offered by Computer Science. May not be repeated for credit.

**Recommended Prerequisite:** SWE 619 or CS 540 and CS 571, or permission of instructor.

**Registration Restrictions:**
Enrollment limited to students with a class of Advanced to Candidacy, Graduate, Non Degree or Senior Plus.

Enrollment is limited to Graduate, Non-Degree or Undergraduate level students.

Students in a Non-Degree Undergraduate degree may not enroll.

Enrollment limited to students in the College of Science, Schar School of Policy and Gov or Volgenau School of Engineering colleges.

**Schedule Type:** Lecture

**SWE 681: Secure Software Design and Programming.** 3 credits.

Theory and practice of software security, focusing in particular on some common software security risks, including buffer overflows, race conditions and random number generation, and on identification of potential threats and vulnerabilities early in design cycle. Emphasizes methodologies and tools for identifying and eliminating security vulnerabilities, techniques to prove absence of vulnerabilities, ways to avoid security holes in new software, and essential guidelines for building secure software: how to design software with security in mind from the ground up and integrate analysis and risk management throughout the software life cycle. Offered by Computer Science. May not be repeated for credit. Equivalent to ISA 681, SWE 781.

**Recommended Prerequisite:** SWE 619.

**Registration Restrictions:**
Enrollment limited to students with a class of Advanced to Candidacy, Graduate, Non Degree or Senior Plus.

Enrollment is limited to Graduate, Non-Degree or Undergraduate level students.

Students in a Non-Degree Undergraduate degree may not enroll.

Enrollment limited to students in the College of Science, Schar School of Policy and Gov or Volgenau School of Engineering colleges.

**Schedule Type:** Lecture

**SWE 699: Special Topics in Software Engineering.** 3 credits.

Special topics not occurring in regular SWE sequence. Notes: May be repeated for credit when semester topic is different. Offered by Computer Science. May be repeated within the term.

**Recommended Prerequisite:** SWE 619 or CS 540 and CS 571, or permission of instructor.

**Registration Restrictions:**
Enrollment limited to students with a class of Advanced to Candidacy, Graduate, Non Degree or Senior Plus.

Enrollment is limited to Graduate, Non-Degree or Undergraduate level students.

Students in a Non-Degree Undergraduate degree may not enroll.

Enrollment limited to students in the College of Science, Schar School of Policy and Gov or Volgenau School of Engineering colleges.
Schedule Type: Lecture

700 Level Courses

SWE 721: Reusable Software Architectures. 3 credits.
Investigates software concepts that promote reuse of software architectures. Studies influence of object technology on software design and reuse. Investigates domain modeling methods, which model the application domain as a software product family from which target systems can be configured. Covers reusable software patterns including architecture patterns and design patterns, software components, and object-oriented frameworks. Offered by Computer Science. May not be repeated for credit.

Recommended Prerequisite: SWE 621.

Registration Restrictions:
Enrollment is limited to Graduate or Non-Degree level students.
Students in a Non-Degree Undergraduate degree may not enroll.
Enrollment limited to students in the College of Science, Schar School of Policy and Gov or Volgenau School of Engineering colleges.

Schedule Type: Lecture

SWE 722: Service Oriented Architecture. 3 credits.
This course covers the state-of-the-art approaches to building dependable Service-Oriented Architecture (SOA) software systems. A variety of relevant topics are covered, including SOA design principles, implementation platforms and standards, quality of service contracts, runtime management of service providers, and coordination and composition of services. The course includes a final project to exercise the concepts covered in class. Offered by Computer Science. May not be repeated for credit.

Recommended Prerequisite: SWE 622 or instructor's permission.

Registration Restrictions:
Enrollment is limited to Graduate or Non-Degree level students.
Students in a Non-Degree Undergraduate degree may not enroll.
Enrollment limited to students in the College of Science, Schar School of Policy and Gov or Volgenau School of Engineering colleges.

Schedule Type: Lecture

SWE 724: Program Analysis for Software Testing. 3 credits.
Different methods for analyzing software, primarily for purpose of testing. Analysis techniques, algorithms, tools, and applications. Goals are to explore current research issues, learn how to build analysis tools, and understand how these techniques can be applied to software-related activities such as maintenance, reuse and optimization. Offered by Computer Science. May not be repeated for credit. Equivalent to SWE 824.

Recommended Prerequisite: A compiler class (e.g. CS 540) OR a testing class (e.g. SWE 637) or permission of the instructor.

Schedule Type: Lecture

SWE 727: Quality of Service for Software Architectures. 3 credits.
Builds on acquired skills for modeling architectures, and focuses on the relationship between architectural patterns and qualities of service (QoS). By the end of the course, students will be able to elicit the QoS preferences of stakeholder; recognize major architectural styles and the QoS tradeoffs that each presents; design for and reconcile competing QoS requirements; and evaluate a given architecture with respect to a set of QoS requirements. Offered by Computer Science. May not be repeated for credit.

Recommended Prerequisite: SWE 621 or permission of instructor.

Registration Restrictions:
Enrollment is limited to Graduate or Non-Degree level students.
Students in a Non-Degree Undergraduate degree may not enroll.
Enrollment limited to students in the College of Science, Schar School of Policy and Gov or Volgenau School of Engineering colleges.

Schedule Type: Lecture

SWE 737: Advanced Software Testing. 3 credits.
Cutting edge concepts and techniques in software testing. An in-depth study of existing approaches to testing software as well as development of new approaches. Applications of existing concepts and techniques to new technologies. Advanced MS students learn in-depth knowledge for how to apply testing in difficult and challenging real-world scenarios. PhD students learn current research trends, both theoretical and practical. Offered by Computer Science. May not be repeated for credit.

Recommended Prerequisite: SWE 637.

Registration Restrictions:
Enrollment is limited to Graduate or Non-Degree level students.
Students in a Non-Degree Undergraduate degree may not enroll.
Enrollment limited to students in the College of Science, Schar School of Policy and Gov or Volgenau School of Engineering colleges.

Schedule Type: Lecture

SWE 760: Software Analysis and Design of Real-Time Systems. 3 credits.
Background for students who want to conduct research in software engineering of real-time systems. Provides understanding of key real-time software system analysis, design concepts and methods, and how they are used in developing large-scale, real-time software systems. Also explores potential impact of emerging technologies. Includes term project in design and analysis of complex, real-time software system. Offered by Computer Science. May not be repeated for credit. Equivalent to SWE 860.

Recommended Prerequisite: SWE 621.

Registration Restrictions:
Enrollment is limited to Graduate or Non-Degree level students.
Students in a Non-Degree Undergraduate degree may not enroll.
Enrollment limited to students in the College of Science, Schar School of Policy and Gov or Volgenau School of Engineering colleges.

Schedule Type: Lecture

SWE 763: Software Engineering Experimentation. 3 credits.
Detailed study of scientific process, particularly using experimental method. Examines how empirical studies are carried out in software engineering. Reviews distinction between analytical techniques and empirical techniques. Other topics include experimentation required in software engineering, problems that can be solved using experimentation, methods used to control variables and eliminate bias
in experimentation, and analysis and presentation of empirical data for decision making. Offered by Computer Science. May not be repeated for credit.

**Recommended Prerequisite:** SWE 621, or permission of instructor.

**Registration Restrictions:**
Enrollment is limited to Graduate or Non-Degree level students.

Students in a Non-Degree Undergraduate degree may **not** enroll.

Enrollment limited to students in the College of Science, Schar School of Policy and Gov or Volgenau School of Engineering colleges.

**Schedule Type:** Lecture

SWE 795: **Advanced Topics in Software Engineering.** 3 credits.
Advanced topics not occurring in existing courses. Topics normally assume knowledge in one or more existing MS SWE courses. Notes: Repeatable within degree for credit when subject differs. Offered by Computer Science. May be repeated within the degree for a maximum 6 credits.

**Recommended Prerequisite:** 12 credits applicable toward MS.

**Registration Restrictions:**
Enrollment is limited to Graduate or Non-Degree level students.

Students in a Non-Degree Undergraduate degree may **not** enroll.

**Schedule Type:** Research

SWE 821: **Software Engineering Seminar.** 3 credits.
Study of application of software engineering principles, design methods, and support tools through real-life problems extracted from faculty and industry projects. Notes: May be repeated with change in topic. Offered by Computer Science. May be repeated within the degree for a maximum 6 credits.

**Recommended Prerequisite:** SWE 621

**Registration Restrictions:**
Enrollment is limited to Graduate level students.

Enrollment limited to students in the College of Science, Schar School of Policy and Gov or Volgenau School of Engineering colleges.

**Schedule Type:** Seminar

SWE 825: **Special Topics in Web-Based Software.** 3 credits.
Advanced topics in specifying, designing, modeling, developing, deploying, testing and maintaining software written as web applications and web services. May be repeated with change in topic. Offered by Computer Science. May be repeated within the degree for a maximum 6 credits.

**Recommended Prerequisite:** SWE 642 Software Engineering for the World Wide Web.

**Registration Restrictions:**
Enrollment is limited to Graduate level students.

Enrollment limited to students in the College of Science, Schar School of Policy and Gov or Volgenau School of Engineering colleges.

**Schedule Type:** Lecture