Software Engineering

**SE 5354 (CE 5354, CS 5354)** Software Engineering (3 semester hours) Formal specification and program verification. Software life-cycle models and their stages. System and software requirements engineering; user-interface design. Software architecture, design, and analysis. Software testing, validation, and quality assurance. Corequisite: **CS 5343** (CS 5343 can be taken before or at the same time as CE/CS/SE 5354) (3-0) S

**SE 5v81** Special Topics in Computer Science (1-9 semester hours) Selected topics in Computer Science. (May be repeated to a maximum of 9 credit hours.) ([1-9]-0) S

**SE 6301** Special Topics in Software Engineering (3 semester hours) Topics will vary. (May be repeated for credit) (3-0) S

**SE 6354 (CE 6354, CS 6354)** Advanced Software Engineering (3 semester hours) This course covers advanced theoretical concepts in software engineering and provides an extensive hands-on experience in dealing with various issues of software development. It involves a semester-long group software development project spanning software project planning and management, analysis of requirements, construction of software architecture and design, implementation, and quality assessment. The course will introduce formal specification, component-based software engineering, and software maintenance and evolution. Prerequisites: CE/CS/SE 5354 (or equivalent) and knowledge of Java. (3-0) S

**SE 6356 (CS 6356, SYSM 6308)** Software Maintenance, Evolution, and Re-Engineering (3 semester hours) Principles and techniques of software maintenance. Impact of software development process on software justifiability, maintainability, evolvability, and planning of release cycles. Use of very high-level languages and dependencies for forward engineering and reverse engineering. Achievements, pitfalls, and trends in software reuse, reverse engineering, and re-engineering. Prerequisite: CE/CS/SE 5354. (3-0) Y

**SE 6357** Software Quality Assurance and Metrics (3 semester hours) Concepts of the pervasive system attributes: reliability, efficiency, maintainability, reusability, etc. Software complexity and measures. Software process measures, product measures and resource measure. Validation of software measures. Software measures and measurement theory. Measuring, monitoring and controlling reliability. Supporting tools. Prerequisite: CE/CS/SE 5354. (3-0) Y

**SE 6359 (CS 6359)** Object-Oriented Analysis and Design (3 semester hours) Analysis and practice of modern tools and concepts that can help produce software that is tolerant of change. Consideration of the primary tools of encapsulation and inheritance. Construction of software-ICs which show the parallel with hardware construction. Prerequisites: CE/CS/SE 5354 and either CS 3335 or CS 5336. (3-0) S

**SE 6360 (CS 6360)** Database Design (3 semester hours) Methods, principles, and concepts that are relevant to the practice of database software design. Database system architecture; conceptual database models; relational and object-oriented databases; database system implementation; query processing and optimization; transaction processing concepts, concurrency, and recovery; security. Prerequisite: CS 5343. (3-0) S

**SE 6361 (CS 6361, SYSM 6309)** Advanced Requirements Engineering (3 semester hours) System and

https://catalog.utdallas.edu/2013/graduate/courses/se
software requirements engineering. Identification, elicitation, modeling, analysis, specification, management, and evolution of functional and non-functional requirements. Strengths and weaknesses of different techniques, tools, and object-oriented methodologies. Interactions and trade-offs among hardware, software, and organization. System and sub-system integration with software and organization as components of complex, composite systems. Transition from requirements to design. Critical issues in requirements engineering. Prerequisite: CE/CS/SE 5354. (3-0) S

**SE 6362 (CS 6362)** Advanced Software Architecture and Design (3 semester hours) Concepts and methodologies for the development, evolution, and reuse of software architecture and design, with an emphasis on object-orientation. Identification, analysis, and synthesis of system data, process, communication, and control components. Decomposition, assignment, and composition of functionality to design elements and connectors. Use of non-functional requirements for analyzing trade-offs and selecting among design alternatives. Transition from requirements to software architecture, design, and to implementation. State of the practice and art. Prerequisite: CE/CS/SE 5354. (3-0) S

**SE 6367 (CS 6367, SE 6367, SYSM 6310)** Software Testing, Validation and Verification (3 semester hours) Fundamental concepts of software testing. Functional testing. GUI based testing tools. Control flow based test adequacy criteria. Data flow based test adequacy criteria. White box based testing tools. Mutation testing and testing tools. Relationship between test adequacy criteria. Finite state machine based testing. Static and dynamic program slicing for testing and debugging. Software reliability. Formal verification of program correctness. Prerequisite: CE/CS/SE 5354 or consent of instructor. (3-0) Y

**SE 6387 (CS 6387)** Advanced Software Engineering Project (3 semester hours) This course is intended to provide experience in a group project that requires advanced technical solutions, such as distributed multi-tier architectures, component-based technologies, automated software engineering, etc., for developing applications, such as web-based systems, knowledge-based systems, real-time systems, etc. The students will develop and maintain requirements, architecture and detailed design, implementation, and testing and their traceability relationships. Best practices in software engineering will be applied. Prerequisites: CS/SE 6361 or SYSM 6309, and CS/SE 6362. Corequisite: CE/CS/SE 6367 or SYSM 6310. (3-0) S

**SE 6388 (CS 6388)** Software Project Planning and Management (3 semester hours) Techniques and disciplines for successful management of software projects. Project planning and contracts. Advanced cost estimation models. Risk management process and activities. Advanced scheduling techniques. Definition, management, and optimization of software engineering processes. Statistical process control. Software configuration management. Capability Maturity Model Integration (CMMI). Prerequisite: CE/CS/SE 5354. (3-0) Y

**SE 6389 (CS 6389)** Formal Methods and Programming Methodology (3 semester hours) Formal techniques for building highly 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. Applications to reliability-critical, safety-critical, and mission-critical systems, ranging from commercial computer communication systems to strategic command control systems. Prerequisite: CE/CS/SE 5354. (3-0) Y

**SE 6v81** Independent Study in Software Engineering (1-9 semester hours) Topics vary from semester to semester. May be repeated for credit as topics vary. ([1-9]-0) S
**SE 7301 (CS 7301)** Recent Advances in Computing (3 semester hours) Advanced topics and publications will be selected from the theory, design, and implementation issues in computing. May be repeated for credit as topics vary. Prerequisite: Consent of the instructor. (3-0) Y

**SE 8v02** Topics in Software Engineering (1-6 semester hours) (May be repeated to a maximum of 9 hours.) ([1-6]-0) S

**SE 8v07** Research (1-9 semester hours) Open to students with advanced standing subject to approval of the graduate adviser. May be repeated for credit (9 hours maximum). ([1-9]-0) S

**SE 8v98** Thesis (3-9 semester hours) (May be repeated for credit.) ([3-9]-0) S

**SE 8v99** Dissertation (1-9 semester hours) (May be repeated for credit.) ([1-9]-0) S