

Jordan University of Science & Technology

Faculty of

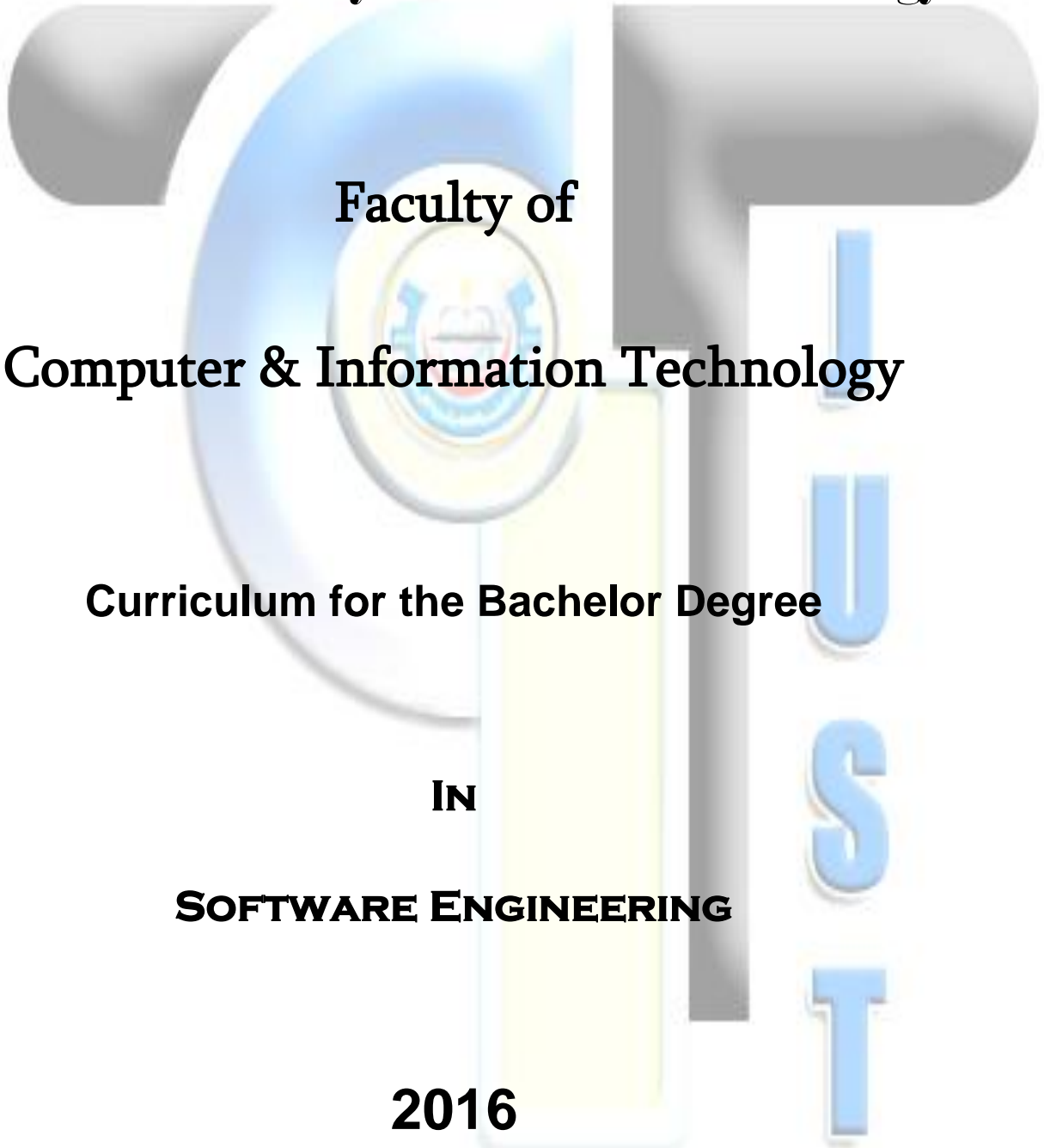
Computer & Information Technology

Curriculum for the Bachelor Degree

IN

SOFTWARE ENGINEERING

2016



Important Contacts

Jordan University of Science and Technology

P.O. Box: 3030 Irbid 22110 Jordan

Tel: (962)-2-7201000

Fax: (962)-2-7095141

E-mail: prsd@just.edu.jo

Faculty of Computer and Information Technology

Tel: (962)-2-7201000 Ext. 20000

Fax: (962)-2-7201077

E-mail: cit@just.edu.jo

Department of Software Engineering

Tel: (962)-2-7201000 Ext: 26953

Fax: (962)-2-7201077

E-mail: se@just.edu.jo

Deanship of Students affairs

Tel: (962)-2-7201000 Ext: 22543

Fax: (962)-2-7201043

E-mail:

studentaffairs@just.edu.jo

Admission and registration unit

Tel: (962)-2-7201000 Ext: 27163

Fax: (962)-2-7201027

E-mail: register@just.edu.jo

International Students Office

Tel: (962)-2-7201000 Ext: 23040

Fax: (962)-2-7201025

E-mail: iso@just.edu.jo

Vision

Making positive contribution to the future of information technology at the local, regional, and international levels and contribution to the knowledge based economy in Jordan.

Mission

Our primary mission is to prepare Software Engineering graduates who can define, design, develop, and deliver high quality software systems using cutting-edge teaching and research technologies as well as collaborations with industry to keep our students at the edge of industrial technological need.

Key values:

Creativity: Creating new ways in understanding and building software systems.

Engagement: Focusing on real-world information technology problems in which our students can design and implement.

Professionalism and Ethics: Keeping the highest standards of ethics and professionalism while delivering software products and services.

Partnership: Building relationships with other departments within the same university as well as other universities and companies in the region.

Software Engineering Program Educational Objectives:

1. Have strong and solid foundation in mathematics, science, and the theories and best practices of the software engineering discipline.
2. Demonstrate critical thinking, problem solving, and analytical skills in building and maintaining high quality software systems in the most cost effective manner.
3. Demonstrate leadership and creativity in the IT discipline.
4. Are proficient in oral and written communication, and effective in teamwork with the highest levels of ethical standards and social responsibilities.
5. Engage in lifelong learning, advance their knowledge, and have skills and ability to pursue graduate studies and do research in software engineering and related interdisciplinary areas.

Outcomes

The software engineering program adopts the IET learning outcomes. IET learning outcomes are categorized under five main learning outcomes:

1. Science & Mathematics:

- SM1p: Knowledge and understanding of scientific principles and methodology necessary to underpin their education in their engineering discipline, to enable appreciation of its scientific and engineering context, and to support their understanding of relevant historical, current and future developments and technologies.
- SM2p: Knowledge and understanding of mathematical and statistical methods necessary to underpin their education in their engineering discipline and to enable them to apply mathematical and statistical methods, tools and notations proficiently in the analysis and solution of engineering problems.
- SM3p: Ability to apply and integrate knowledge and understanding of other engineering disciplines to support study of their own engineering discipline.

2. Engineering Analysis:

- EA1p: Understanding of engineering principles and the ability to apply them to analyse key engineering processes.
- EA2p: Ability to identify, classify and describe the performance of systems and components through the use of analytical methods and modelling techniques.
- EA3p: Ability to apply quantitative and computational methods in order to solve engineering problems and to implement appropriate action.
- EA4p: Understanding of, and the ability to apply, an integrated or systems approach to solving engineering problems.

3. Design:

- D1p: Understand and evaluate business, customer and user needs, including considerations such as the wider engineering context, public perception and Aesthetics.
- D2p: Investigate and define the problem, identifying any constraints including environmental and sustainability limitations; ethical, health, safety, security and risk issues; intellectual property; codes of practice and standards.
- D3p: Work with information that may be incomplete or uncertain and quantify the effect of this on the design.
- D4p: Apply advanced problem-solving skills, technical knowledge and understanding, to establish rigorous and creative solutions that are fit for purpose for all aspects of the problem including production, operation, maintenance and disposal.
- D5p: Plan and manage the design process, including cost drivers, and evaluate Outcomes.
- D6p: Communicate their work to technical and non-technical audiences.

4. Economic, legal, social, ethical and environmental context:

- ET1p: Understanding of the need for a high level of professional and ethical conduct in engineering and a knowledge of professional codes of conduct.

- ET2p: Knowledge and understanding of the commercial, economic and social context of engineering processes.
- ET3p: Knowledge and understanding of management techniques, including project management that may be used to achieve engineering objectives.
- ET4p: Understanding of the requirement for engineering activities to promote sustainable development and ability to apply quantitative techniques where appropriate.
- ET5p: Awareness of relevant legal requirements governing engineering activities, including personnel, health & safety, contracts, intellectual property rights, product safety and liability issues.
- ET6p: Knowledge and understanding of risk issues, including health & safety, environmental and commercial risk, and of risk assessment and risk management techniques.

5. Engineering Practice:

- EP1p: Understanding of contexts in which engineering knowledge can be applied (e.g. operations and management, application and development of technology, etc.).
- EP2p: Knowledge of characteristics of particular materials, equipment, processes, or products.
- EP3p: Ability to apply relevant practical and laboratory skills.
- EP4p: Understanding of the use of technical literature and other information sources.
- EP5p: Knowledge of relevant legal and contractual issues.
- EP6p: Understanding of appropriate codes of practice and industry standards.
- EP7p: Awareness of quality issues and their application to continuous improvement.
- EP8p: Ability to work with technical uncertainty.
- EP9p: Understanding of, and the ability to work in, different roles within an engineering team.

Course Numbering Convention:

Digit	Meaning	Explanation	
Hundreds	Course Level	1	First year
		2	Second year
		3	Third year
		4	Forth year
Tens	Course Subject	0	Software Engineering Basics
		1	Programming Languages
		2	Modeling, Analysis and Design
		3	Development and Testing
		4	Management and Quality
		5	Computer Organization and Architecture
		6	Artificial Intelligence
		7	Systems Programming
		8	Miscellaneous
		9	Special Topics and Training
Ones	Course Sequence	Course sequence number within Subject area	



Study Plan for the Software Engineering Program 2016-2017

The Bachelor Degree in Software Engineering is awarded at Jordan University of Science & Technology after the successful completion of 132 credit hours distributed as follows:-

Table 1: Distribution of credit hours

Requirements	Mandatory	Elective	Total
University Requirements	16	9	25
Faculty Requirements	27	-	27
Departmental Requirements	71	9	80
Total	114	18	132

and after fulfilling the terms and conditions for awarding the bachelor's degree at Jordan University of Science & Technology number (1) issued in 1987 (amended).

1. University Requirements (25 CHs):

1-a) University Mandatory Courses (16 CHs) as shown in table 2:

Table 2: University compulsory courses

Course Number	Course Title	Credit Hours	Weekly Hours	
			Lecture	Lab
MS 100 ⁽¹⁾	Military Science	3	3	0
ARB 101	Arabic Language	3	3	0
HSS 110 ⁽²⁾	Social Responsibility	3	3	1
ENG 112 ⁽³⁾	English Language II	3	3	0
HSS 100 ⁽⁴⁾	Entrepreneurship and Innovation	2	2	0
HSS129	General Skills	2	2	0

NOTE: Non-Arabic speaking students register for the following course instead of (ARB 101), HSS 110:

Courses for non-Arabic Speaking Students

Course Number	Course Title	Credit Hours	Lecture	Lab
ARB 101A	Fundamentals of Arabic language <i>(for Non-Arabs)</i>	3	3	0
HSS 110 A	لم نستطيع الحصول على الاسم باللغة الانجليزية			

¹- The grade for this course is a Pass/Fail grade. Non-Jordanian students can take a substitute course from the elective courses, and the grade for the elective course goes into the calculation of the grade point average.

²- The student who passes the computer skills exam with 50% or more, or has ICDL certificate or Cambridge certificate will be exempted from taking this course.

³- Prerequisite: (ENG 111) or pass the prelim English exam with no less than 80%.

⁴- The grade for this course is a Pass/Fail grade.

1-b) University Elective Courses (9 CHs): the student selects one course from each of the three groups in table (3).

Table (3): University Elective Courses

Course No.	Course title	Credit hours	Theoretical	Lab
Group 1: Humanities Field				
ARB 200	Appreciation of literary texts	3	3	0
HSS 115	Islam and recent problems	3	3	0
HSS 116	Economic system in Islam	3	3	0
HSS 121	Principles of sociology	3	3	0
HSS 126	Principles of psychology	3	3	0
HSS 127	Educational technology	3	3	0
HSS 131	Islamic civilization	3	3	0
HSS 132	The history of the city of Jerusalem	3	3	0
HSS 133	Civilization and recent cultures	3	3	0
HSS 135	Islamic culture	3	3	0
HSS 137	Human rights	3	3	0
HSS 161	Contemporary problems	3	3	0
HSS 211	Sociology (in English)	3	3	0
HSS 213	Individual and society	3	3	0
HSS 221	Introduction to psychology (<i>in English</i>)	3	3	0
HSS 231	History of science in Islam	3	3	0
Group 2: Science and Agriculture Field				
ES 103	Environment protection (<i>for non environmental sciences students</i>)	3	3	0
ME 102	Introduction to renewable energy	3	3	0
ME 211	Fundamentals of automobile engineering (<i>for non-ME students</i>)	3	3	0
CHE 191	Introduction to the Nanotechnology	3	3	0
NF 177	Food preservation (<i>in English</i>)	3	3	0
AP 200	Farm animal products and production (<i>for non-agriculture & veterinary students</i>)	3	3	0
PP 200	Home gardens (<i>for non-agriculture students</i>)	3	3	0
PP 201	Bee keeping (<i>for non-agriculture students</i>)	3	3	0
NR 200	Natural resources and man (<i>for non-agriculture students</i>)	3	3	0
NR 207	Plant earth problems & solutions (<i>for non-agriculture students</i>)	3	3	0
NE 200	Principles in Nuclear Energy and its Peaceful Applications (<i>for non-nuclear students</i>)	3	3	0
Group 3: Health Field				
NUR100	Health promotion	3	3	0
NUR109	Family Health	3	3	0
ADS 100	Oral and dental health (<i>for non-dentistry and non-dentistry sciences students</i>)	3	3	0
PH 104	Community health and nutrition (<i>for non-medicine, non-nursing, and non-midwifery students</i>)	3	3	0
VM 211	Animal health (<i>not for veterinary medical and agriculture students</i>)	3	3	0
VM 212	Pet Animal Care	3	3	0
VM 213	Animal Behaviour and Welfare	3	3	0
VM 214	Animal Products and Public Health (<i>for non veterinary & agriculture students</i>)	3	3	0
PT 100	Wellness and life styles (<i>for non-physical therapy students</i>)	3	3	0
O.T 100	Disability and the Society (<i>for non Rehabilitation Science students</i>)	3	3	0

2. Mandatory Faculty Requirements (27 CHs):

Table (4):

Course Number	Course Title	Credit Hours	Weekly Hours		Prerequisite
			Lecture	Lab	
MATH 101	Calculus I	3	3	0	-
MATH 102	Calculus II	3	3	0	MATH 101
MATH 241	Discrete Mathematics	3	3	0	-
CS 101	Introduction to Programming	3	3	0	CIS 99
CS 102	Programming Lab	1	0	3	CS 101 or Concurrent
CS 112	Introduction to Object-Oriented Programming	3	3	0	Passing CS 102
CS 113	Object-Oriented Programming Lab	1	0	3	CS 112 or concurrent
CS 211	Data Structures	3	3	0	MATH 241 + Passing CS 112
CIS 200	Professional & Ethical Issues in Computing	1	1	0	-
CIS 201	Introduction to Web Design	1	0	3	CS 113
CIS 202	Communication Skills	2	2	0	CIS 200 or Concurrent
CIS 221	Fundamentals of Database Systems	3	3	0	CS 211

3. Department Requirements (80 CHs) distributed as follow:

3-a) Mandatory Departmental Courses (71 CHs) shown in table 5.

3-b) Elective Department Courses (9 CHs) (*) shown in table 6. Student must select at least 6 CHs from Software Engineering Department.

Table (5): Mandatory Department Courses

Course Number	Course Title	Credits Hours	Weekly Hours		Prerequisite
			Lecture	Lab	
CPE 236	Digital Logic Design	3	3	0	CS 113
MATH 233	Probability & Statistics (for CS Students)	3	3	0	MATH 102
CPE 200	Numerical Analysis	2	2	0	MATH 102
CS 284	Analysis and Design of Algorithms	3	3	0	CS 211
CS 318	Human-Computer Interaction	3	3	0	SE 310
CS 375	Principles of Modern Operating Systems	3	3	0	CS 284
CIS 341	Web Applications Development	3	3	0	CIS 201
CIS 441	Business Data Communication	3	3	0	CS 211 + CPE 236
SE 210	Java Programming	3	3	0	Passing CS 113
SE 220	Software Modeling	3	3	0	CS 112
SE 230	Fundamentals of Software Engineering	3	3	0	CS 211
SE 310	Visual Programming	3	3	0	CS 211 + Passing SE 210
SE 320	System Analysis and Design	3	3	0	SE 230 + CIS 221
SE 321	Software Requirements Engineering	3	3	0	SE 220 + SE 320
SE 323	Software Documentation	2	2	0	SE 230
SE 324	Software Architecture & Design	3	3	0	SE 320
SE 326	Software Engineering Lab	1	0	3	SE 310
SE 371	Client/Server Programming	3	Online		SE 310
SE 390	Practical Training	3	0	0	Completion of (90 CHs)
SE 430	Software Testing	3	3	0	SE 320
SE 431	Software Security	3	3	0	SE 324 + CPE 200
SE 432	Software Engineering for Web Applications	3	3	0	CIS 341 + SE 371
SE 440	Project Management	3	Online		SE 320
SE 491	Graduation Project I	3	0	0	Completion of (90 CHs)
SE 492	Graduation Project II	3	0	0	SE 491

(*) Students who are trained in academy or professional training programs in the Faculty of Computer and Information Technology with at least 150 training hours and pass the corresponding international certification exam are exempted from 3 CHs.

Table (6): Elective Department Courses

Course Number	Course Title	Credits	Weekly Hours		Prerequisite
			Lecture	Lab	
CS 485	Multimedia Systems and Networking	3	3	0	CS 375 + CIS 441
CS 486	Simulation and Modeling	3	3	0	MATH 233 + CS 211
CIS 421	Database Applications	3	3	0	CIS 221
CIS 476	Cloud Computing	3	3	0	CIS 441
SE 401	Software Engineering Economics	3	3	0	SE 230
SE 410	Mobile Applications Development	3	3	0	SE 210
SE 412	Selected Programming Languages	3	3	0	SE 210
SE 411	Component-Based Software Development	3	3	0	SE 230
SE 417	Software Engineering Tools	3	3	0	SE 230
SE 420	Formal Methods in Software Engineering	3	3	0	MATH 241 + SE 321
SE 441	Software Quality Assurance	3	3	0	SE 324
SE 442	Software Maintenance and Evolution	3	3	0	SE 230
SE 472	Embedded Software Engineering	3	3	0	SE 324
SE 473	Large Scale Systems Design	3	3	0	SE 324
SE 493	Special Topics in Software Engineering 1	1	1	0	Department Approval
SE 494	Special Topics in Software Engineering 2	2	2	0	Department Approval
SE 495	Special Topics in Software Engineering 3	3	3	0	Department Approval
SE 496	Special Topics in Software Requirements Engineering	3	3	0	SE 321
SE 497	Special Topics in Software Design	3	3	0	SE 324
SE 498	Special Topics in Software Testing & Debugging	3	3	0	SE 430
SE 499	Special Topics in Software Security	3	3	0	SE 431
-	Courses from other departments in the faculty (400 level and above)	-	-	-	Department Approval

Recommended Study Plan/ Software Engineering

1st Year

First Semester			
Course Number	Course Name	Credits	Prerequisite
MATH 101	Calculus I	3	-
HSS 110	Social Responsibility	3	-
CS 101	Introduction to Programming	3	Passing CIS 99
CS 102	Programming Lab	1	CS 101 or concurrent
ENG 112	English Language II	3	Passing ENG 099
ARB 101	Arabic Language	3	-
Total Hours		16	
Second Semester			
Course Number	Course Name	Credits	Prerequisite
MATH 102	Calculus II	3	MATH 101
CS 112	Introduction to Object-Oriented Programming	3	Passing CS 102
CS 113	Object -Oriented Programming Lab	1	CS 112 or concurrent
MATH 241	Discrete Mathematics	3	-
HSS 100	Entrepreneurship and Innovation	2	-
HSS 129	General Skills	2	-
MS 100	Military Science	3	-
Total Hours		17	

2nd Year

First Semester			
Course Number	Course Name	Credits	Prerequisite
CS 211	Data Structures	3	Passing CS 112+MATH 241
CIS 200	Professional & Ethical Issues in Computing	1	-
CIS 201	Introduction to Web Design	1	CS 113
SE 210	Java Programming	3	Passing CS 113
SE 220	Software Modeling	3	CS 112
CPE 236	Digital Logic Design	3	CS 113
-	University Elective	3	-
Total Hours		17	
Second Semester			
Course Number	Course Name	Credits	Prerequisite
MATH 233	Probability & Statistics <i>(for CS Students)</i>	3	MATH 102
CS 284	Analysis and Design of Algorithms	3	CS 211
CIS 221	Fundamentals of Database Systems	3	CS 211
SE 230	Fundamentals of Software Engineering	3	CS 211
CIS 202	Communication Skills	2	CIS 200 or Concurrent
-	University Elective	3	-
Total Hours		17	

Recommended Study Plan/ Software Engineering

3rd Year

First Semester			
Course Number	Course Name	Credits	Prerequisite
CIS 341	Web Applications Development	3	CIS 201
CIS 441	Business Data Communication	3	CS 211
SE 310	Visual Programming	3	CS 211 + Passing SE 210
SE 320	System Analysis and Design	3	SE 230 + CIS 221
CPE 200	Numerical Analysis	2	MATH 102
SE 323	Software Documentation	2	SE 230
Total Hours		16	
Second Semester			
Course Number	Course Name	Credits	Prerequisite
CS 375	Principles of Modern Operating Systems	3	CS 284
SE 321	Software Requirement Engineering	3	SE 220 + SE 320
SE 324	Software Architecture & Design	3	SE 320
CS 318	Human Computer Interaction	3	SE 310
SE 371	Client/Server Programming	3	SE 310
SE 326	Software Engineering Lab	1	SE 310
Total Hours		16	
Summer Semester			
Course Number	Course Name	Credits	Prerequisite
SE 390	Practical Training	3	Completion of 90 CHs
Total Hours		3	

4th Year

First Semester			
Course Number	Course Name	Credits	Prerequisite
SE 430	Software Testing	3	SE 320
SE 431	Software Security	3	SE 324 + CPE 200
SE 491(*)	Graduation Project I	3	Completion of 90 CHs
-	Department Elective	3	-
-	Department Elective	3	-
Total Hours		15	
Second Semester			
Course Number	Course Name	Credits	Prerequisite
SE 440	Project Management	3	SE 320
SE 492	Graduation Project II	3	SE 491
SE 432	Software Engineering for Web Applications	3	CIS 341 + SE 371
-	Department Elective	3	-
-	University Elective	3	-
Total Hours		15	

(*) Students must register "SE 491" before the graduation semester (Semester 1 or Semester 2).

Software Engineering Department

Course Description (2016-2017)

At least 30% of the total grade shall be allotted for practical work including projects, practical sessions, assignments, and case studies.

SE 210: Java Programming (3C, 3H, 0L)

Prerequisite: Passing CS 113

Covers structured programming in Java object-oriented programming, inheritance, interfaces, polymorphism, over loading, Graphical User Interfaces (GUI), I/O streams, exceptions, multithreading, layout manager and applications including client-server.

SE 220: Software Modeling (3C, 3H, 0L)

Prerequisite: CS 112

Introduction to the concepts of object-oriented software modeling (techniques and methodologies). A general modeling language (e.g., UML), structure modeling, behavior modeling, domain modeling, architecture modeling, model checking, limitations of modeling, validation of models, comparison of different approaches considering their advantages and disadvantages. An internal laboratory is included.

SE 230: Fundamentals of Software Engineering (3C, 3H, 0L)

Prerequisite: CS 211

Develop a strong foundation in software engineering with this exploration of key development processes. Topics include process definition and maturity; the system development life cycle; software life cycles and methodologies; requirements analysis and specification; architectural design, integration and testing.

SE 310: Visual Programming (3C, 3H, 0L)

Prerequisite: Passing SE 210 + CS 211

Fundamentals programming, new programming features such as, LINQ, Auto implemental properties, and implicitly typed variables. Object oriented programming concepts such as inheritance interfaces. GUI Graphics and Multimedia such as, WPF, XAML, and WINFORMS. WEB/DATABASE concepts such as, XML, LINO To XML, Database, SQL, LINO To SQL.

SE 320: System Analysis and Design (3C, 3H, 0L)

Prerequisite: SE 230 + CIS 221

An introduction to the system development life cycle. Emphasis on strategies and techniques of systems planning, analysis and design, documentation, implementation and evaluation. Students are expected to carry out group projects using the system development life cycle.

SE 321: Software Requirements Engineering (3C, 3H, 0L)

Prerequisite: SE 320 + SE 220

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 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 modeling using UML notation. Students participate in a group project on software requirements, specification, and object-oriented software design.

SE 324: Software Architecture & Design (3C, 3H, 0L)

Prerequisite: SE 320

An overview of principles of object-oriented design through design patterns. A discussion of the design pattern alternatives. Software architecture frameworks. Software Architecture analysis and validation. An introduction to interface design.

SE 323: Software Documentation (2C, 2H, 0L)

Prerequisite: SE 230

An overview of writing methods and practices that software engineers use to create software documentation. The software documentation process. Documenting for the programmer. Documenting system tests. Online documentation. Types of online documentation. User documentation. Types of user manuals. Style and layout. Tutorials. System documentation. Types of system documentation.

SE 326: Software Engineering Lab (1C, 0H, 3L)

Prerequisite: SE 310

In this lab students should build a complete software system including documentation, requirements specification, risk analysis, design models, testing plans, and user manual. Student should learn how to work as a team.

SE 371: Client/Server Programming (3C, online, 0L)

Prerequisite: SE 310

This course covers several aspects for client-server systems, including: client-server models, transaction processing, communications, programming, security, middleware, developments.

SE 390: Practical Training (3C, 3H, 0L)

Prerequisite: Completing of 90 CHs

Students will train in companies, factories, governmental agencies, private institutions, etc., in a pre-approved computer-related activity for a period of eight weeks under the supervision of a faculty member.

SE 401: Software Engineering Economics (3C, 3H, 0L)

Prerequisite: SE 230

The course covers quantitative models of software lifecycle, cost-effectiveness analysis in software engineering, multiple-goal decision analysis, uncertainty and risk analysis, software cost estimation, software engineering metrics; and quantitative lifecycle management techniques.

SE 410: Mobile Application Development (3C, 3H, 0L)

Prerequisite: SE 210

The course covers the design, development and deployment of mobile applications using a modern programming language.

SE 411: Component-Based Software Development (3C, 3H, 0L)

Prerequisite: SE 230

Introduces concepts and foundations of software component and component-based software development. Detailed study of engineering principles of modeling, designing, implementing, testing, and deploying component-based software architectures. Also explores state-of-the-art component technologies.

SE 412: Selected Programming Language (3C, 3H, 0L)

Prerequisite: SE 210

The course covers one of the emerging programming languages.

SE 417: Software Engineering Tools (3C, 3H, 0L)

Prerequisite: SE 230

Developing models of systems & software; Verifying model properties; fixing wrong models; Generating systems from models—statically and dynamically; Extending generated parts – statically and dynamically; Composing parts (in sensible ways) – statically & dynamically; Adaptive, dynamic, service-oriented architectures; Automated testing techniques – proactive & reactive; Practical, realistic applications using these techniques.

SE 420: Formal Methods in Software Engineering (3C, 3H, 0L)

Prerequisite: SE 321 + MATH 241

This course introduces the use of formal mathematical notation and reasoning in the software development process. These methods have applications in requirements specification, design and verification. Course topics include mathematical foundations, predicates, preconditions and post conditions, alternative notations, types of formal models, and the strengths and limitations of formal methods.

SE 430: Software Testing (3C, 3H, 0L)

Prerequisite: SE 320

Concepts and techniques for testing and modifying software in evolving environments. Topics include software testing at the unit, module, subsystem, and system levels; automatic and manual techniques for generating test data; testing concurrent and distributed software; designing and implementing software to increase maintainability and reuse.

SE 431: Software Security (3C, 3H, 0L)

Prerequisite: SE 324 + CPE 200

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 to integrate analysis and risk management throughout the software life cycle.

SE 432: Software Engineering for Web Applications (3C, 3H, 0L)

Prerequisite: CIS 341 + SE 371

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 client-server programming, component-based software development, middleware, and reusable components.

SE 440: Project Management (3C, online, 0L)

Prerequisite: SE 320

Introduce basic concepts such as the definition of a project, the nature of the project team, and the role of the project manager. Examines techniques related to project formation, acceptance, and funding. Covers the project life cycle, work breakdown structure (WBS), Gantt charts, network diagrams, scheduling techniques, and resource allocation/estimation including return on investment, cost/benefit analysis and earned value. Emphasizes the role of contingency management in planning.

SE 441: Software Quality Assurance (3C, 3H, 0L)

Prerequisite: SE 324

This course covers a broad range of topics related to software quality assurance (SQA). The course will explore combined application of a variety of SQA components, including: SQA activities typically performed by external participants; extension of SQA activities to project schedules and budget control; SQA implementation issues, SQA risk management considerations; and costs associated with SQA.

SE 442: Software Maintenance and Evolution (3C, 3H, 0L)

Prerequisite: SE 230

Introduction to Software Evolution, Maintenance and Reengineering. Reverse Engineering: Program Analysis, Architecture Recovery, Software Complexity and Maintenance Metrics, Program Visualization. Forward Engineering: Refactoring, Code Transformation, Web-enabling. Software Reengineering Strategies and Management.

SE 472: Embedded Software Engineering (3C, 3H, 0L)

Prerequisite: SE 324

Architecture of embedded systems and explore the difference between embedded design and traditional software engineering. The special demands on embedded systems including real-time programming, portability, low power usage, and miniaturization approach. The course introduces models and architectures, and covers such topics as specification, system partitioning, design quality, and developing synthesizable models.

SE 473: Large Scale Systems Design (3C, 3H, 0L)

Prerequisite: SE 324

The course describes the lifecycle of computer systems, and explains how they may be specified, designed, and implemented. Systems analysis is presented as a way of gathering and structuring information such that the required specification corresponds closely to the users' requirements. Systems design is then presented as a technique for transforming the specification to a form in which it can be implemented. The course considers thoroughly the activities which are required and the tools which are available to manage commercial software development projects. Real case studies will be used to illustrate many of the tools and techniques introduced.

SE 491: Graduation Project I (3C, 3H, 0L)

Prerequisite: Completing of 90 C.H

Provides the senior student with the opportunity to undertake a substantial graduation project under the supervision of a faculty member. At least two weeks prior to registration, an interested student must submit to the department chair a written request for permission to select a project. The request is to include a preliminary description of the proposed project and the name of the supervising faculty member. During this course, the student is expected to specify and design the proposed system or software.

SE 492: Graduation Project II (3C, 3H, 0L)

Prerequisite: SE 491

This is a continuation of SE 491, where the student implements, tests and presents the proposed system or software to a 3-member faculty committee that includes the project's supervisor. A written report is to be submitted to the department and committee.

SE 493: Special Topics in Software Engineering I (1C, 1H, 0L)

Prerequisite: Department approval

This course should cover state-of-the-art problems and solutions in software engineering.

SE 494: Special Topics in Software Engineering II (2C, 2H, 0L)

Prerequisite: Department approval

This course should cover state-of-the-art problems and solutions in software engineering.

SE 495: Special Topics in Software Engineering III (3C, 3H, 0L)

Prerequisite: Department approval

This course should cover state-of-the-art problems and solutions in software engineering.

SE 496: Special Topics in Software Requirements Engineering (3C, 3H, 0L)

Prerequisite: SE 321

This course should cover state-of-the-art problems and solutions in Software Requirements Engineering.

SE 497: Special Topics in Software Design (3C, 3H, 0L)

Prerequisite: SE 324

This course should cover state-of-the-art problems and solutions in Software Design.

SE 498: Special Topics in Software Testing and Debugging (3C, 3H, 0L)

Prerequisite: SE 430

This course should cover state-of-the-art problems and solutions in Testing and Debugging.

SE 499: Special Topics in Software Security (3C, 3H, 0L)

Prerequisite: SE 431

This course should cover state-of-the-art problems and solutions in Software Security.