



## Jadara University

### Software Engineering (SE) Courses Description

Course name And Description	Credits	Prerequisites
<p><b><u>Introduction to Software engineering (503201)</u></b></p> <p>Basic Concepts: Software product, Software crisis, software engineering, software process, software process model, methodologies, methods, tools, artefacts; Software Process (I): process models, iterative process; Software Process (II): software process activities (specification, design and implementation, validation/verification, evolution); Software Requirement Engineering (I): Functional/Non Functional requirements, user requirements, system requirement, requirement document; Software Requirement Engineering (II): Software requirement, elicitation and analysis, basics on Use cases, UML notation; Software Prototyping; System Models (I): Context models, Behavioural models; System Models (II): Data Models, Objects Models; Architectural Design: system structuring, control models, modular decomposition; Object Oriented Design, UML notation; User interface design: user interface design principles, user interaction, information presentation; Verification and Validation: planning, software inspections, automated static analysis; Software Testing : defect testing, integration testing; Software Change: program evolution dynamics, software maintenance; Software Cost estimation</p>	3	Programming Language I (501121)
<p><b><u>Computer Human Interaction (503221)</u></b></p> <p>Introduction. Human Aspects: Introducing a range of established and emerging theories, conceptual frameworks and methods of the human aspects of HCI; Knowledge representation and organisation, mental models, the utility of mental models in HCI, verbal metaphors, virtual interface metaphors, classification of interface metaphors for applications, conceptual models; Technology Aspects: Introducing a range of input and output devices and interaction styles, and discussing some higher level system design issues; Design Practice: Discussing the most popular design and evaluation methods and design support tools that are available to make HCI design user-centred, including principles and methods for user centred design, requirement gathering, task analysis and structured HCI design. Screen Design: An advanced topic that covers a theoretical model to support screen design. Hypertext, Multimedia and the World Wide Web: Covering major research issues in multimedia and the Web. GUI programming: Introduction to GUI, Java review exercises, GUI Components - Swing, Event processing, Mouse Events, Keyboard Events, Window Events.</p>	3	Introduction to Software engineering (503201)

<p align="center"><b><u>Software Design Methodologies (503321)</u></b></p> <p>Basic Concepts of Design; Software Design Quality; Design Strategies and Processes; Design Methodologies; Software Architecture; Description of Software Architectures; Software Architectural Styles; Design in Different Styles; Case Study 1 Extraction of keyword vector from text; Design Spaces; Case Study 2 Design space for user interface architecture; Design by Applying Design Patterns; Case Study 3 Document editor; Analysis and Evaluation of Architectural Designs; Case Study 4 - Analysis of designs of extraction of keyword vector;</p>	3	Introduction to Software engineering (503201)
<p align="center"><b><u>Object Oriented Software engineering (503331)</u></b></p> <p>Knowledge and practice in the object-oriented analysis and design activities of software engineering. Topics covered: unified process model, object oriented analysis models, object oriented design models, changing the design to code. The course will be associated with hands-on experience through lab sessions on one of the available CASE tools.</p>	3	Software Design Methodologies (503321) Object Oriented Programming (501333)
<p align="center"><b><u>Project Management (503401)</u></b></p> <p>The module addresses issues involving the creation, development, and maintenance of software projects.</p>	3	Systems Analysis and Design (501361) Software development (503323)
<p align="center"><b><u>Software Testing(503422)</u></b></p> <p>Role of verification and validation (V&amp;V) in the system life cycle;. Techniques and tools; Quality assessment, testing, inspection, proof-of-correctness and relevant V&amp;V standards; The student will be assigned chapters in the required tests to read and use in the projects and assignments.</p>	3	Object Oriented Software engineering (503331)
<p align="center"><b><u>Software re-engineering (503425)</u></b></p> <p>This module will focus on enabling software maintenance through reengineering; computer-aided techniques to recover information from pre-existing systems; Refactoring, migration, Program transformation, Data reverse engineering, Object Oriented Reengineering.</p>	3	Software development (503323)
<p align="center"><b><u>System analysis and design (501381)</u></b></p> <p>Introduction to System Development Life Cycle (SDLC) covering: feasibility studies, requirements collection</p>	3	Database Systems (501341)
<p align="center"><b><u>Software engineering case tool (503466)</u></b></p> <p>Hands on practice on using CASE tools for software development life cycle. Classification of CASE tools. Comparative study and research on existing CASE tools.</p>	3	Software Design Methodologies (503321)
<p align="center"><b><u>Software Documentation (503431)</u></b></p> <p>Introduction to the documentation activity during the software development process. Detailed documentation guidelines including: organizational policy and document templates. Introduction to IEEE Software Engineering Standards-based templates for documentation. Professional techniques for designing, producing, and packaging the documentation. Integrating user-friendly tutorials and on-line help</p>	3	Software Design Methodologies (503321)

menus within any documentation project. Hands-on experience in writing software documentations.		
<p align="center"><b><u>Web Engineering (503332)</u></b></p> <p>The course addresses the concepts, methods, technologies, and techniques of developing Web sites that collect, organize and expose information resources. Topics covered include requirements engineering for Web applications, design methods and technologies, interface design, usability of web applications, accessibility, testing, metrics, operation and maintenance of Web applications, security, and project management. Specific technologies covered in this course include client-side (HTML, JavaScript, and CSS) and server-side (PHP).</p>	3	Database Systems (501341)
<p align="center"><b><u>Software development (503323)</u></b></p> <p>Rigorous introduction to the technological and managerial discipline concerned with the design and implementation of large software systems. Techniques for software specification, design, verification, and validation. Formal methods for proving the correctness of programs. Student teams design, implement, and present a substantial software project.</p>	3	Object Oriented Software engineering (503331)
<p align="center"><b><u>Software security (503361)</u></b></p> <p>This course examines best practices for designing secure systems, with particular emphasize on software engineering. We review various criteria for designing secure systems and apply those principles to real systems. Students will be exposed to various techniques for analyzing system properties and for verifying program correctness, and will be expected to use that knowledge in examining existing protocols. Topics to be covered include the limits of techniques for software protection, such as code obfuscation, tamper-proofing and water-marking, analysis of software based attacks (and defenses), timing attacks and leakage of information, type safety, and capability systems. A course project is required.</p>	3	Algorithms (501292)
<p align="center"><b><u>Software modeling(503335)</u></b></p> <p>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.</p>	3	Software Design Methodologies (503321)
<p align="center"><b><u>Distributed Software Engineering(503432)</u></b></p> <p>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.</p>	3	Object Oriented Software engineering (503331)
<p align="center"><b><u>Introduction to Decision Support systems (503312)</u></b></p> <p>This course focuses on the use and application of information systems to support the decision-making process. Knowledge-based systems, neural networks, expert systems, electronic meeting systems, group systems and web-based systems are discussed as a basis for designing and developing highly effective decision support</p>	3	Systems Analysis and Design (501361)

systems. Data models, interactive processes, knowledge-based approaches and integration with database systems are also described. Theoretical concepts are applied to real-world applications.		
<b><u>Practical Training (Software Engineering) (503496)</u></b>	3	Pass in (99) Academic Hours
<b><u>Special Topics in Software Engineering (502491)</u></b> Advanced issues developments in the field of Software Engineering: This class will be centered on extensive paper reading and discussion. Students will develop a term project with research content. Advanced Topics must be approved by SE Department.	3	permission of the SE Department
<b><u>Graduation Project (503499)</u></b> In this course Students will develop a project related to any field of Software Engineering.	3	Pass in (99) Academic Hours