Course Title	**Systems Analysis and Design**				
Course Code	**MIS202**				
Course Type	**Compulsory**				
Level	BSc/1st Cycle				
Year / Semester	2ND/3RD				
Teacher's Name	Ioannis Filippopoulos				
ECTS	7.5	Lectures / week	3 hours	Laboratories / week	-
Course Purpose and Objectives	The main objectives of the course are to: • Address different types of organizational needs which may undertake some information technology-based solution. • Introduce the various aspects of feasibility and their use in the determination of project feasibility. • Examine several development methodologies which may be used to manage the software development process. Such methodologies include: Structured Systems Analysis and Design Methodology (SSADM) and the Systems Development Life Cycle (SDLC); agile and iterative methodologies including Prototyping, Rapid Application Development and other agile software development approaches; Object-Oriented Analysis and Design using UML and other methodologies. • Examine a variety of information gathering techniques and their potential use. • Cover formal project management techniques and team dynamics. • Identify, evaluate (Cost vs. Benefit analysis), and suggest different systems alternatives. • Briefly discuss security, validation, and privacy issues relating to data maintenance and accessibility. • Address the objectives for effective design (input, output, database, data entry procedures). Introduce human-computer interaction and incorporate its principles in the software design. • Discuss successful information system implementation by addressing training requirements and possibilities, physical conversion strategies, and the need for evaluation				
Learning Outcomes	After completion of the course students are expected to be able to: 1. Understand the types of organizational needs that can be addressed using information technology-based solutions.				

	2. Initiate, specify, and prioritize information systems projects by the determination of various aspects of feasibility for these projects. 3. Understand and compare different systems development methodologies. 4. Use at least one specific methodology for analyzing an organizational situation (a problem or opportunity), modeling it using a formal technique, and specifying requirements for a system that enables a productive change in the way the organization operates. Within the context of this methodology, students will learn to write clear and concise requirements' documents and convert them into technical specifications. 5. Communicate effectively with various organizational stakeholders to collect information using a variety of techniques and to convey proposed solution characteristics to them. 6. Manage information systems projects using formal project management methods. 7. Identify various systems acquisition alternatives, including the use of packaged systems and outsourced design and development resources. 8. Compare various acquisition alternatives systematically including performing a cost and benefit evaluation of the alternatives. 9. Incorporate principles of security and user experience from the beginning of the systems development process. 10. Design high-level logical system characteristics (user interface design, design of data and information requirements). 11. Propose a user-training program and a conversion strategy for the successful implementation of the information system. 12. Use CASE tools effectively to complete different tasks of the software development process.

	2. Initiate, specify, and prioritize information systems projects by the determination of various aspects of feasibility for these projects. 3. Understand and compare different systems development methodologies. 4. Use at least one specific methodology for analyzing an organizational situation (a problem or opportunity), modeling it using a formal technique, and specifying requirements for a system that enables a productive change in the way the organization operates. Within the context of this methodology, students will learn to write clear and concise requirements' documents and convert them into technical specifications. 5. Communicate effectively with various organizational stakeholders to collect information using a variety of techniques and to convey proposed solution characteristics to them. 6. Manage information systems projects using formal project management methods. 7. Identify various systems acquisition alternatives, including the use of packaged systems and outsourced design and development resources. 8. Compare various acquisition alternatives systematically including performing a cost and benefit evaluation of the alternatives. 9. Incorporate principles of security and user experience from the beginning of the systems development process. 10. Design high-level logical system characteristics (user interface design, design of data and information requirements). 11. Propose a user-training program and a conversion strategy for the successful implementation of the information system. 12. Use CASE tools effectively to complete different tasks of the software development process.		
Prerequisites		Required	-
Course Content	1. Identification of opportunities for IT-enabled organizational change. Organizations and their information needs. Types of information systems. Systems modeling. 2. Software development methodologies: a) Structured Systems Analysis and Design (SSADM) and the Systems Development Life Cycle (SDLC) b) Agile and iterative systems development as in Prototyping, Rapid Application Development, other agile development methodologies. c) Object-Oriented Systems Analysis and Design (OOAD); the Unified Modeling Language (UML). 3. The use of CASE tools. 4. Project management including initiation, selection, prioritization. Establishing project feasibility. Planning and managing team members and activities.		

	5. Gather information requirements using different data collection techniques 6. Process modeling using data flow diagrams 7. Analysis and specification of system requirements. Writing process specifications. Data dictionaries. 8. Implementation alternatives; Packaged vs. Custom-made software, Outsourcing vs. Inhouse development. Cost and benefit evaluation of alternatives. 9. Design effective output and input. Understand design objectives. 10. System data requirements. Design the database. Security issues. 11. Human-computer interaction principles. Design the user interface; effective dialog and user feedback. 12. User experience, training issues. 13. Physical conversion strategies for implementation. 14. Security and disaster recovery. 15. New system evaluation.
Teaching Methodology	Lectures, Exercises, CASE tools demonstrations, Project workshop sessions
Bibliography	**Essential Reading** K. Kendall, J. Kendall,(2013),*Systems Analysis and Design, 9/e,*Pearson, ISBN : 9780273787105
Assessment	Participation/Attendance, Project, Assignments, Final Exam
Language	English