UNIVERSITETI I EVROPËS JUGLINDORE
УНИВЕРЗИТЕТ НА ЈУГОИСТОЧНА ЕВРОПА
SOUTH EAST EUROPEAN UNIVERSITY

Study program	Computer Sciences
Faculty	Contemporary Sciences and Technologies
Study Cycle	Second Cycle (Postgraduate)
ECTS	60
Code	SE-60
Title	Master in Computer Sciences / Module: Software Engineering
Accreditation archive number [60]	
Accreditation archive number []	
Decision for starting of the program	
Accreditation date	16.02.2023

Description of the program

Changes in the field of computer sciences and their application are very dynamic. The main challenge of the research and studies in this area is developing new advanced systems and technologies that will provide solutions in the area of information and communication technologies. Information and communication technologies have become the largest, the most important and the most developed sectors that are rapidly expanding in the European Union and the global market. In addition, the emergence of new markets for the software and telecommunications sector in Southeast Europe has led to increased demand for highly qualified and specialized professionals in this field. Graduate students can work as professional software engineers or as software architects in the development of software companies or in IT departments of various different enterprises. The high level of professional skills will enable graduate students to become successful leaders in the software industry.

Career

The program will supply students with the necessary knowledge and skills so that they can contribute to all aspects of the software development process, including planning, collaboration, specifications, design, development, delivery and maintenance of software products. In addition, students will also acquire general skills, such as analytical and critical thinking, teamwork including multicultural environments, planning and organization. After finishing this program, the graduates will have career opportunities in a variety of industries, mainly fulfilling the needs for designing computer systems, developing software for mobile and Web applications, working as database engineers, managers of software projects and processes, etc. depending on the track the students will choose within this study program. The last semester of studies includes master thesis writing, enabling program graduates to continue their studies towards a doctoral degree in computer sciences.

Learning outcomes

Knowledge and understanding

Ability to develop and implement original and creative IT ideas to ensure the quality and design and managing applications related to telecommunications applications areas such as security and quality assurance;

Ability to apply IT skills and knowledge and demonstrate specialized competencies in computer sciences and information technologies in order to organize and connect telecommunications processes like a structure that is managed and monitored both in terms of data flow and in terms of creating user interfaces;

Having knowledge and understanding of areas such as computer sciences and engineering (programming, web technologies, databases, networks, computer and information systems and multimedia);

Having knowledge of one or more areas of the telecommunications industry that can upgrade students to expert s in the application of knowledge in a given area;

Applying knowledge and understanding

Ability to critically, independently and creatively solve problems in new and unfamiliar environments with no previous experience in telecommunications;

Planning, management and evaluation of independent research in the field of telecommunications as well as development and implementation of appropriate tools for testing, simulation and implementation;

Creativity and originality in the interpretation of the knowledge in informatics to solve problems related to the objectives of the industrial production area of telecommunications;

Making judgement

Ability for creative integration and synthesis of knowledge from several areas in the telecommunicationsfield, and administration processes and systems using IT tools designed and created for a specific issue.

Creating educational processes using computer tools and techniques;

Ability to deal with complex situations associated with specific processes resulting in real-time telecoms space;

Ability to identify appropriate specialized instances and make sound judgments in situations of lack of complete information or data based on personal, social and ethical principles and responsibilities associated with the application of knowledge and understanding;

Communication skills

Ability to share findings and proposals with rational argument and reliance both with professionals and with unskilled people, clearly and unambiguously;

Taking considerable responsibility in shared outcomes, running, and initiating activities, etc.

Learning skills

Ability to take responsibility for continued private study in specialized areas of business and information within the networked economy;

Ability to take responsibility for further professional development and training;

Able to identify personal needs and directions for individual and autonomous study and to perform it in self-directed and autonomous manner in the common business and informatics areas.

Able to take responsibility for ongoing individual and group learning in specialized business and informatics fields within the networked economy, including defining learning objectives for medium and longer terms.

List of courses

Semester 1

- [MCS-303] [6.0 ECTS] **Research Methodology**
- [CM191] [6.0 ECTS] **Advanced Algorithms and Data Structures**
- [CM192] [6.0 ECTS] **Formal Methods in Computer Science**
- [6.0 ECTS] **Professional Elective (from the module)***
- [6.0 ECTS] **Professional Elective (from the module)***

Semester 2

- [CST-THESIS-120] [30.0 ECTS] **Master Thesis**

Description of courses

Core courses

- **Research Methodology**
  The purpose of this course is to provide students with knowledge and understanding of different scientific theories and methodologies. Initially the student will be introduced to the conceptual, theoretical definitions and examples of all existing methods of research, hypothesis, direct and indirect variables, validation of the results, the conclusions BIAS and scientific qualitative and quantitative methodologies, "ground research" methodology and other methodological approaches. In each chapter the student will work on practical assignments. After completing the course the student will be able to explain thoroughly and understand the importance of basic scientific concepts, effectively search and find information-relevant literature, identify, describe and formulate scientific problems, make a careful choice of alternative research approaches, thoroughly described, compare and explain the advantages and disadvantages of different scientific methods for collecting quantitative and qualitative data, apply basic scientific methods to analyse quantitative and qualitative data, understand different frameworks for building theory and review and evaluate scientific publications.

- **Advanced Algorithms and Data Structures**
  This course builds on previous knowledge in the area of algorithms and data structures. The goal of the course is to acquaint students with efficient advanced algorithms and adequate data structures that are used to organize, search and optimize data. It also includes the theoretical efficiency of algorithms and its practical determination in order to be able to compare different algorithms. During the course, students will be introduced to several well-known algorithms, particularly search and optimization in complex nonlinear structures such as trees and graphs.

- **Formal Methods in Computer Science**
  This course gives students a comprehensive introduction to formal methods and their application in software specification and verification. It covers some fundamentals in formal methods, including set theory, functions, finite state machines, predicate and temporal logics, and model checking. The course will give students examples of real-world application of these formal techniques.

- **Master Thesis**
  This module enables students to transfer their skills and knowledge to research and carry out more complex tasks related to their master thesis. The module is designed to be fully practical and students to acquire the necessary knowledge and skills to approach writing the thesis. The module has unique return result-to enable students to write the master thesis with minimal difficulties, and with maximum efficiency. The course aims to improve research techniques and style of writing the paper, taking into account the prevention of the usage of illegal means, such as plagiarism and infringement of copyright, which are prohibited by the Statute of SEEU.

Elective courses

- **Requirements Engineering**
  Requirements Engineering (RE) plays a fundamental role within the systems development process. The goal of this course is to introduce the concepts, methods and techniques needed in the eliciting, documenting, negotiating, validating, and managing requirements for complex information systems. The course explains how requirements engineering fits into a broader systems development process and provides an understanding of the main challenges

in requirements engineering nowadays. During this course, students will learn how to identify stakeholders and their influence on system requirements, how to specify functional system requirements using different modelling methods, how to identify and classify non-functional system requirements, how to analyse, write, and validate requirements, how to establish traceability among the documented requirements, and how to manage changing requirements.

- **Usability Testing**
  This course aims to enable students with knowledge of the theory and practice in usability testing, with the goal of creating usable application interfaces and better UX (user experience). The course deals with the process and evaluation strategy for the user interface. Through concrete examples of user interfaces, students should understand the principles and be able to plan a test, collect data, analyse and interpret the results for better user interface.

- **Human - Computer Interaction**
  This course aims to enable students with knowledge of the theory and practice in software development related to the communication between humans and computers, with the goal of creating usable application interfaces. The course deals with the psycho-motor aspects that influence the way people communicate with machines. Through concrete examples of user interfaces, students should understand the principles and be able to apply them while designing their applications.

- **Containerized Architecture**
  This course aims to enable students with knowledge of the theory and practice in deploying applications in containerized architecture. The course deals with the process to package software and its dependencies in an isolated unit, called a container, which can run consistently in any environment. Through concrete examples of distributed applications, students should understand containers, create containerized applications and deploy their solutions to the cloud.

- **Software Architectures**
  Software Architecture is an important technical concept in modern large-scale software development that serves several purposes: planning development resources; analysing the problem domain from multiple perspectives using the architectural views; and abstracting big amounts of information to obtain a useful overview of the system. In addition, the architecture and the decisions on how to construct architecture are a key component to plan and achieve a certain level of quality in a system, which thereby determines how successful the system will be. In the course the students are expected to obtain detailed knowledge about software architectures and software quality and the effects software architectures and architectural decisions have on the qualities of the software product under development. The students will acquire understanding about how to: i) document software architectures of existing systems; ii) construct software architectures based on modern methods and ideas such as architecture styles and patterns and tactics, by considering the intended product, the surrounding technology and the developing organization; iii) discuss the suitability of a software architecture for a specific software product in a specific context using facts as basis.

- **Object-Oriented Design and Patterns**
  This course aims to teach a rigorous approach to object-oriented design and programming, with an emphasis on abstraction, modularity, and code reuse as applied to the building and understanding of large-scale systems. In addition to object-oriented concepts, it covers object-oriented modelling using UML, best design practices, design patterns, and their application to real world problem solving and modelling of applications.

- **Web Services and Applications**
  This course introduces students to software architectures designed to enable computer programs to communicate over a network using open standard Internet protocols. They will learn how to design and develop SOAP and RESTful web services using various frameworks and tools. Moreover, they will learn about advanced concepts like security, versioning, error handling and serverless APIs. After completing the course, students will have enough knowledge to design, develop and deploy highly adaptable and scalable real-world applications using services.