Brief course description- Course Plan Development and Updating Procedures\ Department Software Engineering	QF01/0409-3.0E

Faculty	Science & IT	Academic Department	Software Engineering	Number of the course plan (2016-2017)
Number of Major requirement courses	9			

This form is just for the major requirement courses

Course number	Credit hours	**Title of the course**	Prerequisite-co-requisite
0104714	3	Advanced Software Testing	-

This course will examine advanced software testing techniques. In particular, the important phases of testing will be reviewed, emphasizing on the significance of each phase when testing different types of software.
Students will learn the state of the art in testing technology for object-oriented, component-based, concurrent, distributed, graphical-user interface, and web software. In addition, closely related concepts such as model checking and program analysis will also be studied. Emerging concepts such as test-case prioritization and their impact on testing will be examined. Students will gain hands-on testing/analysis experience by proposing new solutions to open research problems in the field of software testing and experimentally demonstrating the strengths/weaknesses of their solutions.

Course number	Credit hours	**Title of the course**	Prerequisite-co-requisite
0104716	3	Advanced project management	-

This course describes the key aspects of a software project. It begins with the job description of a software manager and then addresses those topics germane to successful software development management, including organizing the software development team; interfacing with other engineering organizations (systems engineering, quality assurance, configuration management, and test engineering); assessing development standards; selecting the best approach and tailoring the process model; estimating software cost and schedule; planning and documenting the plan; staffing the effort; managing software cost and schedule during development; risk engineering; and continuous process improvement. Personnel management topics, including performance evaluations, merit planning, skills building, and team building, are also covered. This course introduces software engineers aspiring to become technical team leaders or software project managers to the responsibilities of these roles. For those software engineers who have advanced to a software development leadership position, this course offers formal training in software project management**.**

Course number	Credit hours	**Title of the course**	Prerequisite-co-requisite
0104713	3	Advanced Software development	-

This course gives students experience designing, implementing, testing, and debugging large programs. Students will also get advanced Java programming experience; covering topics such as inheritance, multithreading, networking, database programming, and web development. Additionally, this course provides students an opportunity to share the potential and limitations of research works on the software development with their colleagues.

Course number	Credit hours	**Title of the course**	Prerequisite-co-requisite
0104712	3	Advanced Software Architecture and Design	-

This course describes architectural patterns for various architectures, such as broker, discovery, and transaction patterns for service-oriented architectures, and addresses software quality attributes

including maintainability, modifiability, testability, traceability, scalability, reusability, performance, availability, and security. Complete case studies illustrate design issues for different software architectures are also discussed in this course

Course number	Credit hours	**Title of the course**	Prerequisite-co-requisite
0104711	3	Advanced Software Requirements	-

The focus of this course is how to find and collect requirements from relevant sources both at the start and during a software development project. Different methods for this as well as different underlying principles and formats for documenting and maintaining requirements are covered. Important is also that we consider different contexts such as market-driven and agile development. Issues in requirements engineering are not only technical but fundamentally depend on process, organizational, business and strategical considerations.

Course number	Credit hours	**Title of the course**	Prerequisite-co-requisite
0102721	3	Advanced Algorithms	-

This course aims to introduce the concepts of algorithm design and analysis. Its topics cover the following concepts: Solving summations and recurrences. Efficiency and complexity analysis. Tree terminology and algorithms. Binary trees. Hashing methods and solving collision in hashing. Heaps and heap sort. Insertion sort, merge sort and quicksort. Graph terminology, representation and algorithms. Algorithms of Prim, Kreskas, Dijkstra and Floyd. Breadth-first and depth-first search. The greedy, divide-and-conquer, and dynamic programming techniques.

Course number	Credit hours	**Title of the course**	Prerequisite-co-requisite
0104718	3	Research methodology	-

The module aims to develop the student knowledge and competence of the research process, and the application of research methods in the area of software engineering. The topics you will cover include research methods, strategies and paradigms, as well as supporting skills and professional issues relating to a career in software engineering. The student will undertake a literature review, critical reading of research papers and the writing and presentation of a research proposal.

Course number	Credit hours	**Title of the course**	Prerequisite-co-requisite
0102741	3	**Advanced database**	-

This course aims to introduce Data Mining Concepts, Advanced Topics in data Mining Case studies in Data Mining, Hashing and Indexing, Query Processing and Optimization Concurrency Control. Recovery System.

Course number	Credit hours	**Title of the course**	Prerequisite-co-requisite
0104715	3	Advance software quality	-

Advance software quality course descriptions a step by step description of software quality and software reliability engineering process. It includes introduction to software quality, prediction and measurement of software size and cost, software reliability engineering process, defining necessary reliability, developing operational profiles, decision making based on the test results, techniques to improve and

predict software reliability, application of quality concept to agile and incremental software development processes. The focus is on the reliability of object-oriented software systems. A workshop (project) is designed to reinforce the presented material. In the workshop, the students will actually go through the estimation and evaluation of quality of a realistic software project.