KING FAHD UNIVERSITY OF PETROLEUM & MINERALS

COLLEGE OF COMPUTER SCIENCES AND ENGINEERING

DEPARTMENT OF INFORMATION & COMPUTER SCIENCE

Master of Science in
Software Engineering

Student Guide

Version 1.4
August 2019

Revision History

Date	Version	Description	Author
2014-04-19	1.2	Clarified the elective courses.	M. Alshayeb
2014-09-02	1.3	SWE 588 added to the list of SWE electives	Jameleddine Hassine
2019-08-25	1.4	Removed SWE 588 and added SWE 670 and SWE 671 & Unify all ICS graduate guides	M. Alshayeb

Table of Content

1. Introduction

The Department of Information and Computer Science offers a Master of Science in Software Engineering (MS.SWE) degree, which was approved in April 2010. The program is in compliance with the international standards and recommendations.

2. ICS Department Vision and Mission

The vision of the ICS department is to be a **regional leader that is recognized worldwide in education, research and professional development in the areas of Computer Science and Software Engineering.** The mission of the Department of Information and Computer Science is to:

1. Provide high quality undergraduate and graduate educational programs in the fields of Computer Science and Software Engineering.
2. Contribute significantly to the research and the discovery of new knowledge and methods in computing.
3. Offer expertise, resources, and services to the community.
4. Keep its faculty members current by providing opportunities for professional development

3. Program Educational Objectives and Outcomes

3.1 Program Educational Objectives

In line of the vision and mission of the ICS Department, the objectives of the MS.SWE Program are to:

1. Have the necessary core skills based on the core body of knowledge in software engineering so that the individual can formulate research models, select the best solution to solve real world problem,
2. Exhibit leadership in the Software Engineering.
3. Have strong foundation for further research and discovery leading to a Ph.D. degree.

3.2 Program Learning Outcomes

Upon completion of the program, a graduate of the MS.SWE program will be able to:

1. Apply proper theoretical practical knowledge of software requirements engineering and software systems design. This includes feasibility analysis, negotiation, and good communication with stakeholders.
2. Self-learn new models, techniques, and technologies as they emerge.
3. Analyze the current significant software technology; articulate its strengths and weaknesses, and improvements.
4. Recognize the relationships between core body of knowledge in software engineering and other related engineering disciplines (e.g. systems and computer engineering) and to be able to apply software engineering techniques to solve problems in related engineering disciplines.
5. Reconcile conflicts in software project objectives, finding acceptable compromises within limitations of cost, time, and organization’s core business.
6. Carry out literature review, develop research proposal, and conduct research in specific topics related to software engineering core areas/develop an approach to analyze and solve specific software engineering problem.

4. Program Requirements

The Master of Science in Software Engineering (MS.SWE) is offered with thesis and requires thirty (30) credit hours that include twenty-four (24) credit hours of course work (i.e. 8 courses) and six (6) credit hours of thesis work. Nine (9) credit hours are Software Engineering core courses. The program has another fifteen (15) credit hours out of which six (6) credit hours are to be taken from software engineering elective courses, three (3) credit hours of elective from computer science courses, and six (6) credit hours of free elective courses from graduate courses (e.g. Computer Science (ICS), Security & Information Assurance (SEC), Computer Engineering (COE), Systems Engineering (SE), Electrical Engineering (EE), or Mathematics (MATH) courses) approved by the ICS department.

4.1 Course Requirements

4.1.1 Core Courses

- Nine (9) credit hours (i.e. 3 courses) of core Software Engineering course:
  - SWE 515 Software Requirements Engineering
  - SWE 516 Software Design
  - SWE 526 Software Testing and Quality Assurance
- 0-credit hours Seminar Course
  - SWE 599: Seminar
- Six (6) credit hours thesis work
  - SWE 610: Master Thesis

4.1.2 SWE Elective Courses

Six (6) credit hours (i.e. 2 courses) from any of the Software Engineering elective courses:

SWE 531	Secure Software
SWE 532	Web Applications Security
SWE 536	Software Architecture
SWE 539	Software Metrics
SWE 566	Software Agents
SWE 585	Empirical Software Engineering
SWE 587	Software Project Management
SWE 595	Special Topics in Software Engineering
SWE 606	Independent Research
SWE 634	Software Reuse
SWE 638	Software Maintenance and Re-Engineering
SWE 670	Formal Methods and Models in Software Engineering
SWE 671	Global Software Engineering

4.1.3 Free Elective Courses

MS.SWE students are expected to take three (3) credit hours of elective from Computer Science (ICS) courses, and six (6) credit hours of elective courses from graduate courses (e.g. Computer Science (ICS), Security & Information Assurance (SEC), Computer Engineering (COE), Systems Engineering (SE), Electrical Engineering (EE), or Mathematics (MATH) courses) approved by the ICS department.

*: Software Engineering courses cannot be counted towards the fulfillment of the free elective courses' requirement.

4.2 Degree Plan

The degree plan for the MS.SWE is shown in Table 1.

Table 1: MS.SWE Degree Plan

Course No.	Title	LT	LB	CR	
First Semester					
SWE 515	Software Requirements Engineering	3	0	3	
SWE 516	Software Design	3	0	3	
SWE 5XX	SWE Elective Course I	3	0	3	
		9	0	9	**9**
Second Semester					
SWE 526	Software Testing and Quality Assurance	3	0	3	
ICS 5XX	Computer science elective	3	0	3	
YYY 5XX	Free Elective	3	0	3	
SWE 599	Graduate Seminar	1	0	0	
		10	0	9	**9**
Third Semester					
SWE 5XX	SWE Elective Course II	3	0	3	
YYY 5XX	Free Elective	3	0	3	
SWE 610	MS Thesis	0	0	3	
		6	0	9	**9**
Fourth Semester					
SWE 610	MS Thesis	0	0	3	**3**
		Total			**30**

5. Admission Requirements

The applicant should have the equivalent degree of an undergraduate software engineering of King Fahd University of Petroleum and Minerals. In general, applicants with a four-year degree in related fields in science and engineering (e.g. computer science, computer engineering, systems engineering, electrical engineering, information technology, etc.) may be considered for admission. However, an applicant lacking an adequate undergraduate training may be admitted if recommended by the department's graduate committee and the chairman, with the understanding that the course work taken to remove the deficiency in the undergraduate training may not be credited towards the degree."

In addition to the general university admission requirements set by the KFUPM Deanship of Graduate Studies stated in section 5.1, the department also sets other admission requirements relevant to the program stated in sections 5.2.

5.1 General University Admission Requirements for MS degree

The minimum requirements for possible admission as a regular graduate student to pursue a Master program in engineering or science are as follows:

1. A four-year Bachelor's (B.S.) Degree in engineering or science from a recognized institution with a major in the proposed field or evidence of suitable background for entering the proposed field.
2. A Grade-Point Average (GPA) of 3.00 or higher on a scale of 4.00 or equivalent, and a GPA of 3.00 in the subject of the major field. Official transcripts and degree certificates are required for final admission.

3. Completion of TOEFL with a minimum score of 520 (PBT), 190 (CBT) or 68 (IBT). The TOEFL score must be sent directly to the Deanship of Graduate Studies (KFUPM code is 0868). IELTS is also acceptable (min 6.0).
4. Acceptable General Graduate Record Examination (GRE), which should also be reported directly (KFUPM code is 0868). (Minimum required GRE score is Quantitative: 700, Analytical: 4.0 and Verbal: 300)
5. At least three letters of recommendation from the faculty who taught the applicant university-level courses. [to be submitted through the online recommendation system]
6. Satisfactorily meeting any additional departmental or university admission requirements.

Note (1): Satisfying the minimum admission requirements does not guarantee admission into the program as final admission is subject to an evaluation of the whole application, and the capacity of individual programs.

Note (2): The general university admission requirements can be found at the deanship of graduate studies website:
http://www.kfupm.edu.sa/deanships/dgs/Pages/en/Admission-Requirements.aspx

5.2 Admission Requirements for MS SWE degree

The priority for the enrollment in MS.SWE program is for applicants who hold BS in software engineering. Applicants who hold BS in other related IT disciplines should have a satisfactory background in the following core areas of software engineering and computer science such as: software requirements engineering, software design and architecture, software testing, project management, database systems, operating systems, and design and analysis of algorithms.

Unsatisfactory background in any of these areas is considered a deficiency. Provisional admission may be granted to qualified students such students must take the appropriate deficiency course(s) at KFUPM with a grade of B or better before a change of status to regular graduate student.

6. Courses Description

SWE 505 Principles of Software Engineering (3-0-3)

Software Requirements: Modern SRS for Enterprise Application, Software Process: Personal and Team Software Process, Traditional Software Processes and Agile Processes. Software Design: Architecture Tradeoff Analysis and patterns. Software Project Management: project initiation, planning, executing, monitoring control and closing. Software Engineering Measurement and Analysis, Software Quality Assurance: 6-Sigma. Software Integration: Enterprise Application Integration, COTS Integration.

Note: ***This course is for non SWE students, it cannot be taken for credit toward a MS degree by SWE graduate students.***

SWE 515 Software Requirements Engineering (3-0-3)

The course gives state of the art and state of the practice in software requirements engineering. In-depth research-oriented study of methods, tools, notations, and validation techniques for the analysis, specification, prototyping, and maintenance of software requirements. Topics include study of object-oriented requirements modeling, using state of the art modeling techniques such as the Unified Modeling Language (UML). The course work includes a project investigating or applying approaches to requirements engineering.

Note: SWE 515 cannot be taken for credit with ICS 512

SWE 516 Software Design (3-0-3)

Concepts and methods for the architectural design of large-scale software systems. Fundamental design concepts and design notations are introduced. Several design methods are presented and compared. In-depth research-oriented study of object-oriented analysis and design modeling using state of the art modeling techniques such as Unified Modeling Language (UML). Students participate in a group project on object-oriented software design.

Note: SWE 516 cannot be taken for credit with ICS 513

SWE 526 Software Validation, Verification, and Quality Assurance (3-0-3)

In-depth research-oriented study of verification and validation throughout the development lifecycle. Techniques for validation and verification, quality assurance at the requirements and design phases, software testing at the unit, module, subsystem, and system levels. Automatic and manual techniques for generating and validating test data. Testing process: static vs. dynamic analysis, functional testing, inspections, and reliability assessment.

Note: SWE 526 cannot be taken for credit with ICS 514

SWE 531 Secure Software (3-0-3)

Software security development lifecycle including security requirements analysis, design, coding, review, and testing. Construction of secure and safe C/Unix programs. Vulnerabilities in C source code. Stack and heap buffer overflows. Overview of secure web application development with consideration for SQL injection, cookies, and forceful browsing. 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.

SWE 532 Web Applications Security (3-0-3)

Study of contemporary web application vulnerabilities, based on the OWASP (Open Web Application Security Project). Study of exploitation techniques for server and client web applications, and techniques that lead to web defacement and server penetration. Auditing and scanning web applications and servers for security weaknesses and vulnerabilities. Contemporary attack scenarios exploiting web vulnerabilities such as cross-site scripting, SQL injection, cookies, and forceful browsing. Content-based attacks and effective countermeasures. Secure programming for the following technologies: .NET, ASP.NET, ActiveX, JAVA, Secure Sockets, and XML, and a study of web security protocols such as SSL and HTTPS.

SWE 536 Software Architecture (3-0-3)

Advanced principles, methods and best practices in building software architecture and the architecture design process are discussed. Architectural styles and patterns are presented and compared. Software architecture analysis and evaluation methods such as ATAM and CBAM, tradeoffs among conflicting constraints in building high quality architecture are also discussed. Architecture documentation is also presented.

SWE 539 Software metrics (3-0-3)

Software metrics history and current practice, basics of measurement theory for software metrics, framework for software measurement, product, application, and process metrics. The course includes introduction to foundations of measurement theory, models of software engineering measurement, software products metrics, software process metrics and measuring management.

SWE 566 Software Agents (3-0-3)

Agent-based programming; elements of distributed artificial intelligence; beliefs, desires and intentions; component based technology; languages for agent implementations; interface agents; information sharing and coordination; KIF; collaboration; communication; ontologies; KQML; autonomy; adaptability; security issues; mobility; standards; agent design issues and frameworks; applications in telecommunications.

Prerequisite: Consent of Instructor

SWE 585 Empirical software engineering (3-0-3)

The course discusses how empirical studies are carried out in software engineering. The distinction between analytical techniques and empirical techniques is reviewed. Other topics include empirical studies required in software engineering, kinds of problems that can be solved empirically, methods used to control variables and eliminate bias in empirical studies, and analysis and presentation of empirical data for decision making.

SWE 587 Software Project Management (3-0-3)

Lifecycle and process models; process metrics; planning for a software project; mechanisms for monitoring and controlling schedule, budget, quality, and productivity; and leadership, motivation, and team building. Topics cover quantitative models of the software lifecycle, process improvement techniques, cost-effectiveness analysis in software engineering, multiplegoal decision analysis, uncertainty and risk analysis, software cost estimation, software engineering metrics; and quantitative lifecycle management techniques.

Note: SWE 587 cannot be taken for credit with ICS 515

SWE 595 Special Topics in Software Engineering (3-0-3)

Advanced topics selected from current journals of software engineering that deal with theoretical development or applications in the field. Topic include: Reusable Software Architectures, Software Engineering, Experimentation, Concurrent Software Systems,

Software Metrics, Web Engineering or Formal Methods and Models in Software Engineering, etc.
Prerequisite: Consent of Instructor

SWE 599 Seminar (1-0-0)
Graduate students are required to attend the seminars given by faculty members, visiting scholars, and fellow graduate students. Additionally, each student must give at least presentation on a timely research topic. Among other things, this course is designed to give the student an overview of research, research methodology, journals and professional societies. Graded on a Pass or Fail basis
Prerequisite: Graduate standing

SWE 606 Independent Research (3-0-3)
This course is intended to allow the student to conduct research in advanced problems in his MS research area. The faculty offering the course should submit a research plan to be approved by the Graduate Program Committee at the academic department. The student is expected to deliver a public seminar and a report on his research outcomes at the end of the course. Graded on a Pass or Fail basis
Prerequisite: Prior arrangement with an instructor

SWE 610 Master Thesis (0-0-6)
The student has to undertake and complete a research topic under the supervision of a faculty member in order to probe in depth a specific problem in Computer Science.
Prerequisite: SWE 599 or Consent of Instructor.

SWE 634 Software Reuse (3-0-3)
In-depth research based study of the concepts and engineering principles of software reuse with a focus on component-based reuse, domain analysis and modeling, service-oriented architectures; quality aspects of reuse, economic models of reuse; and reuse of non-code artifacts.
Prerequisite: Consent of Instructor

SWE 638 Software Maintenance & Re-Engineering (3-0-3)
Software evolution and reengineering approaches and abstraction techniques to extract specifications and design from existing code are discussed. Major maintenance activities are presented including estimating maintenance costs, managing change and predicting maintainability with software quality metrics. Organizational issues relative to product maintenance are discussed. Principles of reverse engineering techniques are also presented.
Prerequisite: Consent of Instructor.

SWE 670 Formal Methods and Models in Software Engineering 3-0-3
In-depth advanced formal mechanisms for specifying, validating, and verifying software systems. Program verification. Formal specification via algebraic specifications and abstract model specifications, including initial specification and refinement toward implementation. Integration of formal methods with existing programming languages, and the application of formal methods to requirements analysis, testing, safety analysis, and object-oriented approaches. Model-driven architectures. Formal methods using the Object Constraint Language (OCL).
Prerequisites: Consent of Instructor

SWE 671 Global Software Engineering 3-0-3

Topics include: Essentials of global software engineering, Software engineering outsourcing (Onshore outsourcing, Nearshore Outsourcing, Offshore out-sourcing), Outsourcing models (Simple Dyadic Outsourcing, Multi-Vendors Outsourcing, Co-Sourcing and Complex Outsourcing), Global software project management concepts, tools, and techniques, Managing virtual teams, Cross-cultural collaboration, Global project leadership, Measuring organizations readiness for global software development, Software quality in global software development (CMMI, ISO 9001:2000), Global software engineering challenges, Professional practices for global software engineering (Intellectual Property Rights, Group working, conflict and negotiations management, Presentations, writing and referencing)

Prerequisites: Consent of Instructor