



National University of Sciences & Technology (NUST)
School of Electrical Engineering and Computer Science
(SEECs)
Department of Software Engineering

SE311 - Software Requirements Engineering

Course Code:	SE-311	Semester:	4 th
Credit Hours:	3+0	Prerequisite Codes:	SE200, CS212;
Instructor:	Dr. Rafia Mumtaz	Class:	BESE-9A
Office:	SEECs A-108	Telephone:	+92-51-9085 2161
Lecture Days:	Mon, Wed	E-mail:	rafia.mumtaz@seecs.edu.pk
Class Room:		Consulting Hours:	Friday 10:00am-12:00 pm
Lab Engineer:	N/A	Lab Engineer Email:	N/A
Knowledge Group:	Software Engineering	Updates on LMS:	Before/After every lecture

Course Description:

Nowadays, software systems pervade in every facet of our everyday life. Such systems are developed to meet our needs by solving the real world problems. To realize a software system, requirements engineering plays a critical role not only in bridging the gap between the actual needs and system realization but act as a catalyst for the whole development life cycle. During requirements engineering needs, desires, and intentions of different stakeholders are taken into account and provides methods and techniques for the analyst to refine high level needs into meaningful requirements for the new system. This course introduces the activities, concepts, and techniques needed in eliciting, analyzing, documenting, validating, and managing requirements for software systems. Various types of requirement engineering processes and techniques, along with their impact and usage will be explained with emphasis. The course also covers introduction of advance requirement engineering techniques.

Course Objectives:

- 1- To study various software development models and phases of software development life cycle.
- 2- To understand the role and scope of requirements engineering and know how to apply appropriate methods, techniques and tools to elicit, analyze, document, validate and manage requirements.
- 3- To be able to define a system (or system capability) that satisfies the requirements specification.

Course Learning Outcomes (CLOs):

At the end of the course the student will be able to:	PLO	BT-level
1. Understand the functional and non-functional requirements.	PLO1	C-2
2. Able to elicit and specify the requirements	PLO2	C-4
3. Apply requirement engineering process to manage requirements	PLO2	C-3
4. Evaluate functional and non-functional requirements	PLO4	C-6



National University of Sciences & Technology (NUST)
School of Electrical Engineering and Computer Science
(SEECs)
Department of Software Engineering

Grading Criteria
Assessments/CLOs
Quizzes: 10%
Assignments: 10%
OHT-1: 15%
OHT-2: 15%
Project:10%
End Semester Exam:40%
Total : 100 %

Books:	
Text Book:	<ol style="list-style-type: none">1. Requirements Engineering: Fundamentals, Principles, and Techniques By Klaus Pohl, 1st Edition, ISBN 978-1-937538-77-4 Publication year:20152. Requirement Engineering: Processes and Techniques By Gerald Kotonya and Ian Sommerville ISBN: 978-0-471-9708-2, Published: Sep 1998
Reference Books:	<ol style="list-style-type: none">1. Requirements Engineering Handbook, By Young, Ralph, Artech House, Publication year: 20042. Ian Sommerville Software Engineering, 10th ed. Addison-Wesley, 2016

Pre-Requisites:	
Existing	SE-200: Software Engineering, SE210; Software Design and Architecture, CS212: Object Oriented Programming
Proposed	SE-200: Software Engineering, SE210; Software Design and Architecture, CS220: Database Systems



National University of Sciences & Technology (NUST)
School of Electrical Engineering and Computer Science
(SEecs)
Department of Software Engineering

Lecture Breakdown:	
Week	Topic
Week 1	Introduction to Requirements Engineering <ul style="list-style-type: none"> Overview of Requirement Engineering Process Overview of the Requirements Types (Functional vs Non-functional)
Week 2	System/Functional Requirements <ul style="list-style-type: none"> Definition and Classification of System Requirements Elicitation of System Requirements Impact of System Requirements on Software Development
Week 3	Quality/Non-Functional Requirements <ul style="list-style-type: none"> Definition and Classification of Quality Requirements Elicitation of Quality Requirements Impact of Quality Requirements on Software Development
Week 4	Software Requirement Specification I <ul style="list-style-type: none"> Requirement Specification in Requirement Engineering Process Analysing System and Quality Requirements Prioritising System and Quality Requirements
Week 5	Software Requirement Specification II <ul style="list-style-type: none"> Introduction to Software Requirement Specification (SRS) Document Documenting Requirements with SRS Documenting Requirements – Formal vs Informal Specifications
Week 6	OHT-1
Week 07	Twin Peaks of Requirements and Design/Architecture <ul style="list-style-type: none"> Introduction to Twin Peak Model Dependencies in Twin Peak Model
Week 8	Architecturally Significant Requirements (ASRs) <ul style="list-style-type: none"> Introduction to ASRs Types of ASRs Elicitation of ASRs
Week 9	Requirements Analysis and Validation <ul style="list-style-type: none"> Requirement Analysis Techniques Requirements Verification and Validation Visual Modeling of Requirements (using UML) Static and Behavior models
Week 10	Requirement Management <ul style="list-style-type: none"> Requirements Documentation and Management Tool Support for Requirement Management
Week 11	Requirements Evolution <ul style="list-style-type: none"> Requirement Change and Evolution Process for Requirement Evolution Impacts of Requirement Evolution
Week 12	OHT-2
Week 13	Requirements Traceability Management <ul style="list-style-type: none"> Traceability of Requirement Forward and Backward Traceability
Week 14	Goal Oriented Requirements Engineering (GORE) I <ul style="list-style-type: none"> Introduction to GORE GORE Process Eliciting and Managing Goal-oriented Requirements
Week 15	Goal Oriented Requirements Engineering (GORE) II <ul style="list-style-type: none"> GORE modeling practices Case study with GORE
Week 16	Advanced Topics (RE for Adaptive Systems, Systems of Systems, Cloud Computing, and Mobile Computing)
Week 17	Project Presentations, demo and viva
Week 18	ESE



National University of Sciences & Technology (NUST)
School of Electrical Engineering and Computer Science
(SEECs)
Department of Software Engineering

Grading Policy:

Quiz Policy: The quizzes will be unannounced and normally last for ten minutes. The question framed is to test the concepts involved in last few lectures. Number of quizzes that will be used for evaluation is at the instructor's discretion.

Assignment Policy: In order to develop comprehensive understanding of the subject, assignments will be given. Late assignments will not be accepted / graded. All assignments will count towards the total (No 'best-of' policy). The students are advised to do the assignment themselves. Copying of assignments is highly discouraged and violations will be dealt with severely by referring any occurrences to the disciplinary committee. The questions in the assignment are meant to be challenging to give students confidence about the subject matter and enable them to prepare for the exams.

Plagiarism: SEECs maintains a zero tolerance policy towards plagiarism. While collaboration in this course is highly encouraged, you must ensure that you do not claim other people's work/ ideas as your own. Plagiarism occurs when the words, ideas, assertions, theories, figures, images, programming codes of others are presented as your own work. You must cite and acknowledge all sources of information in your assignments.



National University of Sciences & Technology (NUST)
School of Electrical Engineering and Computer Science
(SEECS)
Department of Software Engineering

Program Learning Outcomes (PLOs)

Program outcomes are the narrower statements that describe what students are expected to know and be able to do by the time of graduation. These relate to the knowledge, skills and attitude that the students acquire while progressing through the program. The program must demonstrate that by the time of graduation the students have attained a certain set of knowledge, skills and behavioral traits, at least to some acceptable minimum level. Specifically, it is to be demonstrated that the students have acquired the following graduate attributes:

- (i) **Engineering Knowledge:** An ability to apply knowledge of mathematics, science, engineering fundamentals and an engineering specialization to the solution of complex engineering problems.
- (ii) **Problem Analysis:** An ability to identify, formulate, research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences and engineering sciences.
- (iii) **Design/Development of Solutions:** An ability to design solutions for complex engineering problems and design systems, components or processes that meet specified needs with appropriate consideration for public health and safety, cultural, societal, and environmental considerations.
- (iv) **Investigation:** An ability to investigate complex engineering problems in a methodical way including literature survey, design and conduct of experiments, analysis and interpretation of experimental data, and synthesis of information to derive valid conclusions.
- (v) **Modern Tool Usage:** An ability to create, select and apply appropriate techniques, resources, and modern engineering and IT tools, including prediction and modeling, to complex engineering activities, with an understanding of the limitations.
- (vi) **The Engineer and Society:** An ability to apply reasoning informed by contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to professional engineering practice and solution to complex engineering problems.
- (vii) **Environment and Sustainability:** An ability to understand the impact of professional engineering solutions in societal and environmental contexts and demonstrate knowledge of and need for sustainable development.
- (viii) **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of engineering practice.
- (ix) **Individual and Team Work:** An ability to work effectively, as an individual or in a team, on multifaceted and /or multidisciplinary settings.
- (x) **Communication:** An ability to communicate effectively, orally as well as in writing, on complex engineering activities with the engineering community and with society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
- (xi) **Project Management:** An ability to demonstrate management skills and apply engineering principles to one's own work, as a member and/or leader in a team, to manage projects in a multidisciplinary environment.
- (xii) **Lifelong Learning:** An ability to recognize importance of, and pursue lifelong learning in the broader context of innovation and technological developments.