

Degree Program: B.E Software Engineering

Degree Program:
B.E (Software Engineering)

MUET Vision

To become world class educational and research institute and contribute effectively towards building up indigenous and technological capabilities for sustainable socio-economic development.

MUET Mission

To equip our undergraduate, postgraduate and doctoral students with advance knowledge through collaborative opportunities emerged from linkages with academia, industry and government.

Department Vision

To become the center of excellence and the aspiration in the discipline of software engineering by producing the highly skilled professionals, who with their analytical capabilities and proficiencies apply the technical knowledge for the socio-economic development.

Department Mission

To provide technically sound ambiance of learning and realizing the frequently changing traits of software industry to pursue sustainable socio-economic growth with the sense of ethics, professionalism and leadership to serve community and humanity at large.

Program Education Objectives (PEOs):

A Software Engineering Graduate:

1. Performs his/her professional role in the Software Industry and related fields ^[PLO 1,2,3,4,5].
2. Adheres to professional responsibilities in multicultural environment with continual improvement ^[PLO 6, 7, 8, 12].
3. Works effectively as a team lead or team member in challenging ventures ^[PLO 9, 11].
4. Conveys technical and managerial information efficiently in oral and written forms ^[PLO 1, 9, 10, 12].

PEOs Mapping with University's and Department's Vision & Mission

	PEO 1	PEO 2	PEO 3	PEO 4
MUET vision	✓	✓	✓	✓
MUET Mission	✓	✓	✓	✓
Department Vision	✓	✓	✓	✓
Department Mission	✓	✓	✓	✓

Evaluation Sheets and Performance Indicators for PEOs

A minimum attainment level for each PEO has been defined along with its method of measurement. The key performance indicators for the attainment levels are given in table below.

Program Educational Objectives (PEOs)	Key Performance Indicators (KPIs)	Measurement Tool
Engineering Knowledge A graduate who performs his/her professional role based on acquired fundamental engineering knowledge and analytical skills [PLO 1,2,3,4,5]	50% of the Alumni survey obtained from graduates (3-5 years after graduation) is satisfactory. 50% of the employers' survey is satisfactory.	Alumni Surveys, Employer Surveys
Ethical and Societal Responsibilities A graduate who adheres to professional responsibilities with societal aspects and ethical conduct in multicultural environment with continual improvement [PLO 6, 7, 8, 12]	50% of the Alumni survey obtained from graduates (3-5 years after graduation) is satisfactory. 50% of the employers' survey is satisfactory.	Alumni Surveys, Employer Surveys
Leadership Qualities A graduate who works effectively as a team lead or team member in challenging ventures [PLO 9, 11]	50% of the Alumni survey obtained from graduates (3-5 years after graduation) is satisfactory. 50% of the employers' survey is satisfactory.	Alumni Surveys, Employer Surveys
Communication Skills A graduate who conveys technical and managerial information efficiently in oral and written communication [PLO 1, 9, 10, 12]	50% of the Alumni survey obtained from graduates (3-5 years after graduation) is satisfactory. 50% of the employers' survey is satisfactory.	Alumni Surveys, Employer Surveys

Program Learning Objectives (PLOs):

The twelve graduate attributes provided by the PEC as per Manual of Accreditation 2014 have been adopted by the Department of Software Engineering MUET, Jamshoro as the Program Learning Outcomes (PLOs) for its Bachelor's in Software Engineering Program.

Sr.#	PLO	Description
1.	Engineering Knowledge	An ability to apply knowledge of mathematics, science, Engineering fundamentals and an engineering specialization to the solution of complex engineering problems.
2.	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.
3.	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.
4.	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.
5.	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.
6.	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.
7.	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.
8.	Ethics	Apply ethical principles and commit to professional ethics and responsibilities and norms of engineering practice.
9.	Individual and Teamwork	An ability to work effectively, as an individual or in a team, on multifaceted and /or multidisciplinary settings.
10.	Communication	An ability to communicate effectively, orally as well as in writing, on complex engineering activities with the engineering
11.	Project Management	An ability to demonstrate management skills and apply engineering principles to one's own work, as a member and/or
12.	Lifelong Learning	An ability to recognize importance of and pursue lifelong learning in the broader context of innovation and technological

Mapping of PLOs to PEOs

The twelve PLOs, defined for the Software engineering program, are mapped to the three PEOs. Mapping of the PLOs to PEOs is given in table below.

Program Learning Outcomes		PEO1	PEO2	PEO3	PEO 4
PLO1	Engineering Knowledge	✓			✓
PLO2	Problem Analysis	✓			
PLO3	Design / Development of Solutions	✓			
PLO4	Investigation	✓			
PLO5	Modern Tool Usage	✓			
PLO6	The Engineer and Society		✓		
PLO7	Environment and Sustainability		✓		
PLO8	Ethics		✓		
PLO9	Individual and Teamwork			✓	✓
PLO10	Communication				✓
PLO11	Project Management			✓	
PLO12	Lifelong Learning		✓		✓

Course Scheme B.E Software Engineering

Subject Category	Sr. #	Course Code	Name of Subject	Marks	Credit Hours		
					Theory	Practical	Total
First Semester							
Maths	1	MTH108	Applied Calculus	100	3	0	3
Computing Core	2	SW112	Programming Fundamentals	150	3	1	4
General	3	SW113	Introduction to Info. & Comm. Technologies	100	2	1	3
General	4	ENG111	Functional English	100	3	0	3
Maths	5	EL119	Applied Physics	100	3	0	3
Total				550	14	2	16
Second Semester							
Computing Core	6	SW121	Object Oriented Programming	150	3	1	4
General	7	SW123	Professional Practices	100	3	0	3
Maths	8	MTH112	Linear Algebra & Analytical Geometry	100	3	0	3
Computing Core	9	SW125	Introduction to Software Engineering	100	3	0	3
General	10	PS106	Pakistan studies	50	2	0	2
General	11	IS111 / SS104	Islamic Studies / Ethics	50	2	0	2
Total				550	16	1	17
Third Semester							
Computing Core	12	SW212	Data Structures & Algorithms	150	3	1	4
Computing Core	13	SW215	Database Systems	150	3	1	4
Software Core	14	SW216	Software Requirements engineering	100	3	0	3
Uni Elective	15	SW211	Software Economics & Management	100	3	0	3
Software Support	16	SW217	Operations Research	100	3	0	3
Total				600	15	2	17

Fourth Semester							
Computing Core	17	SW225	Operating Systems	150	3	1	4
Computing Core	18	SW226	Computer Networks	150	3	1	4
Software Core	19	SW227	Software design & architecture	100	2	1	3
Univ Elective	20	SW228	Data Warehousing	100	3	0	3
Uni Elective	21	ENT121	Introduction to Entrepreneurship and creativity	100	3	0	3
Total				600	14	3	17
Fifth Semester							
Software Core	22	SW315	Software Construction and Development	100	2	1	3
Maths	23	MTH317	Statistics & Probability	100	3	0	3
Computing Core	24	SW316	Information Security	100	3	0	3
Software Core	25	SW317	Human computer Interaction	100	3	0	3
Software elective	26	SW318	Agent based Intelligent Systems	100	3	0	3
General	27	ENG311	Communication and presentation Skills	100	3	0	3
Total				600	17	1	18
Sixth Semester							
Software Core	28	SW322	Software Project Management	100	3	0	3
Computing Core	29	SW325	Discrete Structures	100	3	0	3
General	30	ENG319	Technical & business Writing	100	3	0	3
Software Elective	31	SW326	Data Science and Analytics	150	3	1	4
Software Elective	32	SW327	Mobile Application Development	150	3	1	4
Total				600	15	2	17

Seventh Semester							
Software Core	33	SW415	Software re-engineering	100	3	0	3
Software Elective	34	SW416	Multimedia Communication	150	3	1	4
Software Core	35	SW417	Web Engineering	150	3	1	4
Software Support	36	SW418	Formal Methods in Software Engineering	100	3	0	3
Computing Core	37	SW498	Thesis/Project - I	100	0	3	3
Total				600	12	5	17
Eighth Semester							
Software Support	38	SW424	Simulation & Modeling	100	3	0	3
Software Elective	39	SW425	Cloud Computing	150	3	1	4
Software Core	40	SW426	Software Quality Engineering	150	3	1	4
Computing Core	41	SW499	Thesis/Project - II	100	0	3	3
Total				500	9	5	14
Grand Total				4600	112	21	133

HEC Guidelines wise statistics of Course Scheme

Subject Wise Statistics of Proposed Scheme with respect to HEC Guidelines					
Subject Category	Proposed Scheme	HEC Guidelines	Course category	Proposed Scheme	HEC Guidelines
General	19	19	Common Courses (Non Core)	~60%	60 - 63%
Mathematics & Science Foundation	12	12			
Computing-Core	39	39			
University Electives	9	12			
SE Core	26	24	SE Courses (Core)	~40%	35 – 37 %
Software Electives	19	15			
Software Supporting	9	9			
Total Credits	133	130 – 136	Theory Credits		112
			Practical Credits		21
Total Marks			4600		
Even Practicals Credits		11	Odd Practical Credits		10
Total courses from HEC Guidelines			37		
Total courses Not from HEC Guidelines			3		

Mapping of courses to PLOs of B.E Software Engineering

B.E Software Engineering Courses				Engineering Knowledge	Problem Analysis	Design / Development of Solutions	Investigation	Modern Tool Usage	The Engineer and Society	Environment and Sustainability	Ethics	Individual and Teamwork	Communication	Project Management	Lifelong Learning
S. No		Course Code	Course Title	PLO 1	PLO 2	PLO 3	PLO 4	PLO 5	PLO 6	PLO 7	PLO 8	PLO 9	PLO 10	PLO 11	PLO 12
1.	1 st Semester	MTH108	Applied Calculus		✓										
2.		SW112	Programming Fundamentals	✓				✓							
3.		SW113	Introduction to Info. & Comm. Technologies	✓				✓							
4.		ENG111	Functional English		✓								✓		✓
5.		EL119	Applied Physics	✓											
6.	2 nd Semester	SW121	Object Oriented Programming	✓				✓							
7.		SW123	Professional Practices	✓					✓		✓				✓
8.		MTH112	Linear Algebra & Analytical Geometry		✓										
9.		SW124	Introduction to Software Engineering	✓											
10.		PS106	Pakistan studies						✓	✓					✓
11.		IS111 / SS104	Islamic Studies / Ethics						✓		✓				✓
12.	3 rd Semester	SW212	Data Structures & Algorithms	✓	✓			✓							
13.		SW215	Database Systems	✓				✓							
14.		SW216	Software Requirements engineering		✓	✓	✓								
15.		SW211	Software Economics & Management	✓	✓					✓				✓	
16.		SW217	Operations Research	✓	✓	✓									
17.	4 th Semester	SW225	Operating Systems	✓				✓							
18.		SW226	Computer Networks	✓	✓	✓		✓							
19.		SW227	Software design & architecture		✓	✓	✓								
20.		SW228	Data Warehousing	✓	✓	✓	✓								
21.		ENT121	Introduction to Entrepreneurship and creativity						✓	✓	✓	✓	✓		✓

	B.E Software Engineering Courses			Engineering Knowledge	Problem Analysis	Design / Development of Solutions	Investigation	Modern Tool Usage	The Engineer and Society	Environment and Sustainability	Ethics	Individual and Teamwork	Communication	Project Management	Lifelong Learning
S. No		Course Code	Course Title	PLO 1	PLO 2	PLO 3	PLO 4	PLO 5	PLO 6	PLO 7	PLO 8	PLO 9	PLO 10	PLO 11	PLO 12
22.	5 th Semester	SW315	Software Construction and Development		✓	✓		✓							
23.		MTH317	Statistics & Probability		✓										
24.		SW316	Information Security	✓	✓										
25.		SW317	Human computer Interaction	✓	✓										
26.		SW318	Agent based Intelligent Systems	✓	✓	✓									
27.		ENG311	Communication and presentation Skills										✓	✓	
28.	6 th Semester	SW322	Software Project Management	✓	✓		✓		✓			✓		✓	
29.		SW325	Discrete Structures	✓	✓										
30.		ENG319	Technical & business Writing		✓									✓	
31.		SW326	Data Science and Analytics	✓	✓	✓	✓	✓							
32.		SW327	Mobile Application Development	✓	✓	✓		✓							
33.	7 th Semester	SW415	Software Re-Engineering	✓	✓	✓	✓								
34.		SW416	Multimedia Communication	✓	✓			✓				✓			
35.		SW417	Web Engineering	✓		✓		✓				✓			
36.		SW418	Formal Methods in Software Engineering	✓	✓	✓									
37.		SW498	*Thesis/Project - I	✓	✓	✓	✓	✓			✓	✓	✓		
38.	8 th Semester	SW424	Simulation & Modelling	✓		✓	✓								
39.		SW425	Cloud Computing	✓	✓	✓		✓							
40.		SW426	Software Quality Engineering	✓	✓			✓							
41.		SW499	Thesis/Project - II					✓	✓	✓	✓	✓	✓	✓	✓
TOTAL				27	26	15	8	16	6	4	4	6	5	3	6



Mehran University of Engineering and Technology, Jamshoro
Department of Software Engineering



Title of Subject	:	<u>Programming Fundamentals (SW-112)</u>
Discipline	:	Software Engineering (1 st Semester)
Effective	:	19 Batch & onwards
Pre-requisite	:	None
Assessment	:	Theory: 20% Sessional, 80% Written Semester Examination (20% Mid, 60% Final) Practical: 40% Sessional, 60% Final Examination
Credit Hours	:	03 + 01
Minimum Contact Hours:		45 + 45
		Marks : 100 + 50

Course Learning Outcomes (CLOs):

Upon successful completion of the course, the student will be able to:

CLO	Description	Taxonomy Level	PLO
1	Explain and implement concepts of functions, structures, pointers in C++	C3	1
2	Use arrays and pointers to solve complex problems using arrays and pointers	C3	1
3	Code document, test and implement a well-structured, robust computer program using C++ programming language.	P3	5

PROGRAM LEARNING OUTCOMES (PLOs):

The course is designed so that students will achieve the following PLOs:

1	Engineering Knowledge	<input checked="" type="checkbox"/>	7	Environment and Sustainability	<input type="checkbox"/>
2	Problem Analysis	<input type="checkbox"/>	8	Ethics	<input type="checkbox"/>
3	Design/Development of Solutions	<input type="checkbox"/>	9	Individual and Team Work	<input type="checkbox"/>
4	Investigation	<input type="checkbox"/>	10	Communication	<input type="checkbox"/>
5	Modern Tool Usage	<input checked="" type="checkbox"/>	11	Project Management	<input type="checkbox"/>
6	The Engineer and Society	<input type="checkbox"/>	12	Lifelong Learning	<input type="checkbox"/>

Course outline:

- Introduction to Programming and languages
- Algorithms, Flowcharts and pseudocode
- Overview of programming (C, C+, Python)
- Writing, compiling and debugging
- Coding style
- Statements
- Variables and datatypes
- Operators and expressions
- Selection
- Relational operators
- Conditional Statements
- Conditional operators
- Switch, break, continue
- Logical operators
- Modular programming
- Structures in functions and Arrays
- File pointers

- Error handling
- Revision
- Project Demos

Practical Work to be carried out:

1. Basics of C++ Programming and IDE Environment
2. Variables and Constants
3. Data Types and Expressions
4. LOOPS
5. Decision making statements
6. Structures.
7. Functions.
8. Pointers.
9. Arrays
10. Objects and Classes.
11. Inheritance and Polymorphism
12. Strings.
13. Operator overloading.
14. Streams and Files.
15. Case Study/ Semester Project

Recommended Books:

- Peter Norton, Introduction to computers, Latest Edition.
- Robert Lafore, Object Oriented Programming in Turbo C++, Latest Edition.
- Deitel and Deitel, C++ How to Program, Prentice Hall Publications, Latest Edition
- Paul J. Deitel and Harvey M. Deitel, C: How to Program, Prentice Hall, latest edition.
- The C Programming Language by Kernighan and Ritchie, latest edition.
- Code Complete by Steve McConnell, latest edition.
- The Art of Computer Programming (TAOCP) by Donald E. Knuth, latest edition

Approval:

Board of Studies:

Resolution No. 02

Dated: 29-08-2019

Board of Faculty:

Resolution No. 01

Dated: 07-10-2019

Academic Council:

Resolution No. 96.10

Dated: 07-10-2019



Title of Subject	:	OBJECT ORIENTED PROGRAMMING (SW-121)
Discipline	:	Software Engineering (2 nd Semester)
Effective	:	19 Batch & onwards
Pre-requisite	:	Programming Fundamentals
Assessment	:	Theory: 20% Sessional, 20% Mid examination, 60% Final examination. Practical: 40% Sessional, 60% Final examination
Credit Hours	:	3 + 1
Minimum Contact Hours:		45+45
		Marks : 100 + 50

Course Learning Outcomes (CLOs):

Upon successful completion of the course, the student will be able to:

CLO	Description	Taxonomy Level	PLO
1	Explain principles of object oriented paradigm, class libraries, exception handling.	C2	1
2	Use class instances & their relationships to build object oriented solutions	C3	1
3	Implement OOP concepts to develop piece of code (Programs)	P3	5

PROGRAM LEARNING OUTCOMES (PLOs):

The course is designed so that students will achieve the following PLOs:

1	Engineering Knowledge:	<input checked="" type="checkbox"/>	7	Environment and Sustainability:	<input type="checkbox"/>
2	Problem Analysis:	<input type="checkbox"/>	8	Ethics:	<input type="checkbox"/>
3	Design/Development of Solutions:	<input type="checkbox"/>	9	Individual and Team Work:	<input type="checkbox"/>
4	Investigation:	<input type="checkbox"/>	10	Communication:	<input type="checkbox"/>
5	Modern Tool Usage:	<input checked="" type="checkbox"/>	11	Project Management:	<input type="checkbox"/>
6	The Engineer and Society:	<input type="checkbox"/>	12	Lifelong Learning:	<input type="checkbox"/>

Course outline:

• **OOP CONCEPTS AND INTRODUCTION TO JAVA**

Objects, Classes, Messages, OOP principles, Encapsulation, Inheritance, Polymorphism, What is Java, History of Java, Features of Java, Class library, Java Environment, Byte code, JVM, Java Applications & Applets, Comments, White Spaces, Identifiers, Separators, Keywords, Reserved words, Variables, Data Types, Operators, Mathematical functions and constants, Expressions, Escape Sequences, Type Conversion and Casting

• **INTRODUCTION TO APPLETS**

How to write applets, HTML code for applets, applet viewer, paint() Method, drawstring, Option Pane, Applet Life cycle

• **JAVA CONTROL STATEMENTS**

Java Selection statements (if, if-else, switch), Loop Statements (while, do-while, for), nested loops, break and continue statements

- **ARRAYS AND STRINGS**

Array Variables, Defining arrays, Accessing Elements, Initializing arrays, Array Length, Multidimensional Arrays, Array of characters, Strings and common Operations on Strings

- **INTRODUCING CLASSES INHERITANCE**

General Form of a Class, Variables and Methods in Class Definition, Defining Methods, Declaring Objects, Returning a value, Parameter List, Constructors, Parameterized Constructors, this Keyword, Method overloading, Types of Arguments, Objects as parameters, Returning Objects, Access Specifies, Understanding static, Nested and Inner Classes, finalize method, Inheritance Basics, Member Access and Inheritance, Using Super, Multilevel Hierarchy, Method Overriding, Dynamic Method Dispatch, Abstract Classes

- **PACKAGES AND INTERFACES**

Understanding Packages, Defining a package, Packages and Directory Structure, Understanding CLASSPATH, Standard Packages, Importing Packages, Interfaces, Implementing Interfaces, Extending Interfaces, Multiple Interfaces, Anonymous Classes

- **EXCEPTION HANDLING**

Exception Handling Basics, Exception Types, Uncaught Exceptions, Using Try and Catch, Multiple Catch Clauses, Nested Try Statements, throw statement, throws clause, finally Block, Built-in Exceptions

- **MULTITHREADINGINTRODUCING THE AWT AND JAVA GUI**

Understanding threads, Class Thread, Thread Life Cycle, Priority, Synchronization, Runnable Interface, AWT classes, Windows Fundamentals, Graphics Context, Color and Font Controls, Drawing Lines, Rectangles, Ovals etc, Labels, Buttons, Text Fields, Checkboxes etc, Event Handling, Layout Managers, Menu Bars, Dialog Boxes

- **IMAGES, SOUND AND ANIMATION**

Displaying Images, Image Observers, Media Tracker, Double Buffering, Playing Sound Clips, Sound API

Practical Work to be carried out:

1. Installation of Java, Basic Program Structure, Identifiers and Primitive data types
2. Demonstrating various operators in Java
3. Working with iterative structures
4. Working with conditional structures
5. Implementation of arrays
6. Implementing the concepts of encapsulation
7. Implementing the concepts of inheritance
8. Implementing the concepts of Polymorphism
9. Exception and Error Handling.
10. Demonstrating Threads and Multi Threaded Programming logics
11. Understanding I/O Fundamentals
12. Building GUI components
13. Applying various Layouts managers
14. Java Event handling.
15. Java Database Connectivity

Recommended Books:



Mehran University of Engineering and Technology, Jamshoro
Department of Software Engineering

- H. M. Deitel, P.J. Deitel, “Java How To Program”, Latest Edition , Prentice Hall
- Ivor Horton, “Beginning Java 2”, Wrox Corp, Latest Edition.
- Patrick Naughton, Herbert Schildt, ‘Java 2 : The Complete Reference’, Latest Edition.,Osborne/McGrawHill
- Joseph L. Weber, “:Special Edition Using Java 2 Platform”, Prentice Hall India, Latest Edition.

Approval:

Board of Studies:
Board of Faculty:
Academic Council:

Resolution No. 02
Resolution No. 01
Resolution No. 96.10

Dated: 29-08-2019
Dated: 07-10-2019
Dated: 07-10-2019



Mehran University of Engineering and Technology, Jamshoro
Department of Software Engineering



Title of Subject	:	<u>Software Requirement Engineering (SW216)</u>	
Discipline	:	Software Engineering (3 rd Semester)	
Effective	:	18 Batch & onwards	
Pre-requisite	:	Introduction to Software Engineering	
Assessment	:	Theory: 20% Sessional, 80% Written Semester Examination (20% Mid, 60% Final)	
Credit Hours	:	03 + 0	Marks : 100
Minimum Contact Hours:		45	

COURSE LEARNING OUTCOMES:

Upon successful completion of the course, the student will be able to:

CLOs	Description	Taxonomy level	PLO
1	Examine requirements engineering process and analyze software for developments of cost effective and efficient technical solutions.	C4	2
2	Design Software Requirement Specification (SRS) document using clear and unambiguous requirements.	C5	3
3	Interpret requirements of real world problems.	C6	4

PROGRAM LEARNING OUTCOMES (PLOs):

The course is designed so that students will achieve the following PLOs:

1	Engineering Knowledge:	<input type="checkbox"/>	7	Environment and Sustainability:	<input type="checkbox"/>
2	Problem Analysis:	<input checked="" type="checkbox"/>	8	Ethics:	<input type="checkbox"/>
3	Design/Development of Solutions:	<input checked="" type="checkbox"/>	9	Individual and Team Work:	<input type="checkbox"/>
4	Investigation:	<input checked="" type="checkbox"/>	10	Communication:	<input type="checkbox"/>
5	Modern Tool Usage:	<input type="checkbox"/>	11	Project Management:	<input type="checkbox"/>
6	The Engineer and Society:	<input type="checkbox"/>	12	Lifelong Learning:	<input type="checkbox"/>

Course outline:

Introduction to Requirements Engineering, Software Requirements, classification of requirements, Requirements process, Levels/layers of requirements, Requirement characteristics, Analyzing quality requirements, Software requirements in the context of systems engineering, Requirement evolution, requirement traceability, requirement prioritization, trade-off analysis, risk analysis and impact analysis, Requirement management, interaction between requirement and architecture, Requirement elicitation, elicitation sources and techniques, Requirement specification and documentation, specification sources and techniques, Requirements validation and techniques, Management of Requirements, Introduction to Management, Requirements Management Problems , Managing Requirements in an Acquisition Organization, Supplier, Organizations, Product Organizations

Books Recommended:

1. Wiegers K. & Beatty J., Software Requirements, Microsoft Press, Latest edition.
2. Elizabeth Hull, Ken Jackson and Jeremy Dick, Requirements Engineering, Springer-Verlag London Limited, Latest edition.
3. Chumturi M., Requirements Engineering and Management for Software Development Projects, Springer New York, Latest Edition.

Approval:

Board of Studies:
Board of Faculty:
Academic Council:

Resolution No. 02
Resolution No. 01
Resolution No. 96.10

Dated: 29-08-2019
Dated: 07-10-2019
Dated: 07-10-2019



Mehran University of Engineering and Technology, Jamshoro
Department of Software Engineering



Title of Subject	:	<u>Data Structures & Algorithms (SW-212)</u>
Discipline	:	Software Engineering (3 rd Semester)
Effective	:	18 Batch & onwards
Pre-requisite	:	Programming Fundamentals
Assessment	:	Theory: 20% Sessional, 80% Written Semester Examination (20% Mid, 60% Final) Practical: 40% Sessional, 60% Final Examination
Credit Hours	:	03 + 01
Minimum Contact Hours:		45 + 45
		Marks : 100 +50

COURSE LEARNING OUTCOMES:

Upon successful completion of the course, the student will be able to:

CLOs	Description	Taxonomy level	PLO
1	Explain various data structures such as arrays, lists, trees, graphs etc.	C3	1
2	Construct algorithms associated with each data structure.	C4	2
3	Demonstrate the implementation of algorithms programmatically.	P4	5

PROGRAM LEARNING OUTCOMES (PLOs):

The course is designed so that students will achieve the following PLOs:

1	Engineering Knowledge:	<input checked="" type="checkbox"/>	7	Environment and Sustainability:	<input type="checkbox"/>
2	Problem Analysis:	<input checked="" type="checkbox"/>	8	Ethics:	<input type="checkbox"/>
3	Design/Development of Solutions:	<input type="checkbox"/>	9	Individual and Team Work:	<input type="checkbox"/>
4	Investigation:	<input type="checkbox"/>	10	Communication:	<input type="checkbox"/>
5	Modern Tool Usage:	<input checked="" type="checkbox"/>	11	Project Management:	<input type="checkbox"/>
6	The Engineer and Society:	<input type="checkbox"/>	12	Lifelong Learning:	<input type="checkbox"/>

Course outline:

- Fundamentals of data structures: An overview of computer programming,
- Data types, abstract data types, C/C++ background,
- Review of pointers, Pointer arithmetic, Pointer indirections
- Computational complexity of algorithms and their time-space analysis:
- Running time calculations, asymptotic notations for algorithmic complexity analysis.
- Lists Data Structure: Simple arrays, Linked lists, Linear search vs binary search
- Lists Data Structure: Double linked lists, Circular linked lists.
- Stacks & Queues: Sequential/array implementation of stacks and queues,
- Linked list implementation of stacks and queues.
- Arithmetic expressions: Polish notation, Recursion: Recursive implementation of stacks and queues.
- Sorting: Bubble sort, Insertion sort, Selection sort.
- Sorting: Merge sort, Quick sort, Counting Sort & Radix sort, Heap sort (tentative).
- Trees: Data structure definition and generic implementation, Tree traversals and its application, Binary tree, binary search tree, Expression trees, AVL trees.
- Huffman coding, B-Tree.
- Graphs: Adjacency matrix implementation, Linked list implementation
- Graphs Search: Depth-first traversal of graphs, Breadth-first traversal of graphs, Shortest distance algorithms

- Hashing and searching: Hashing techniques, Implementation of Hashing techniques
- Priority Queues: Binary Heap and its applications

Books Recommended:

1. Robert Lafore, "Data structures and Algorithm analysis in java", Latest Edition
2. Weiss Mark Allen, "Data structures & Algorithms Analysis", Latest Edition
3. Herbert Schildt, "The Complete Reference Java2", Latest Edition
4. Semour Lipschutz, "Data Structure", Schaum's outline series, Latest Edition
5. Data Structures and Algorithm Analysis in C by Mark Weiss. Addison Wesley; ISBN: 0-201-49840-5, latest edition.
6. Data Structures and Algorithm Analysis in C++ by Mark Weiss. Addison Wesley; ISBN 0321-44146-X, latest edition
7. Introduction to Algorithms, Thomas H. Cormen et al, latest edition

Practical Work to be carried out:

- 1 To become familiar with conditional and control structures in Java.
- 2 Implementation of Objects and classes.
- 3 Implementation of Arrays in Java
- 4 Implementation of Linear Search Algorithms.
- 5 Implementation of Binary Search Algorithms.
- 6 Implementation of Linked list.
- 7 Implementation of Queues.
- 8 Implementation of Stacks.
- 9 Implementation of Merge sort & Heap sort.
- 10 Implementation of Bubble sort, Quick Sort and insertion sort.
- 11 Implementation of Recursion.
- 12 Implementation of Trees.
- 13 Implementation of Graph.
- 14 Implementation of Binary tree traversals.
- 15 Case study

Approval:

Board of Studies:
Board of Faculty:
Academic Council:

Resolution No. 02
Resolution No. 01
Resolution No. 96.10

Dated: 29-08-2019
Dated: 07-10-2019
Dated: 07-10-2019



Mehran University of Engineering and Technology, Jamshoro
Department of Software Engineering



Title of Subject	:	<u>Software Economics & Management (SW - 211)</u>
Discipline	:	Software Engineering (3 rd Semester)
Effective	:	18 Batch & onwards
Pre-requisite	:	Introduction to Software Engineering
Assessment	:	Theory: 20% Sessional, 80% Written Semester Examination (20% Mid, 60% Final)
Credit Hours	:	03 + 0
Minimum Contact Hours:		45
		Marks : 100

COURSE LEARNING OUTCOMES:

Upon successful completion of the course, the student will be able to:

CLOs	Description	Taxonomy level	PLO
1	Explain fundamentals of software economics engineering by considering the basic but important terminologies.	C3	1
2	Examine cost estimation methods by giving careful attention towards economics lifecycle and uncertainty.	C4	2,7,11

PROGRAM LEARNING OUTCOMES (PLOs):

The course is designed so that students will achieve the following PLOs:

1	Engineering Knowledge:	<input checked="" type="checkbox"/>	7	Environment and Sustainability:	<input checked="" type="checkbox"/>
2	Problem Analysis:	<input checked="" type="checkbox"/>	8	Ethics:	<input type="checkbox"/>
3	Design/Development of Solutions:	<input type="checkbox"/>	9	Individual and Team Work:	<input type="checkbox"/>
4	Investigation:	<input type="checkbox"/>	10	Communication:	<input type="checkbox"/>
5	Modern Tool Usage:	<input type="checkbox"/>	11	Project Management:	<input checked="" type="checkbox"/>
6	The Engineer and Society:	<input type="checkbox"/>	12	Lifelong Learning:	<input type="checkbox"/>

Course outline:

- **Engineering Economics**
 - Role of engineers in business
 - Economic decisions v/s design decisions
 - Large scale engineering projects and types of strategic economic decisions
 - Fundamental principles of engineering economics
- **Interest Rate and Economic Equivalence**
 - Interest: The Cost of Money
 - Economic Equivalence
 - Development of Formulas for Equivalence Calculation
 - Unconventional Equivalence Calculations
- **Understanding Money and Its Management**
 - Nominal and Effective Interest Rates
 - Equivalence Calculations with Effective Interest Rates and with Continuous
- **Payments**
 - Changing Interest Rates
 - Debt Management
 - Investing in Financial Assets
- **Present-Worth Analysis**
 - Project Cash Flows
 - Initial Project Screening Methods: payback Screening and Discounted Cash
- **Flow Analysis**
 - Variations of Present-Worth Analysis

- Comparing Mutually Exclusive Alternatives
- **Annual Equivalent-Worth Analysis**
 - Annual Equivalent-Worth Criterion
 - Capital Costs versus Operating Costs
 - Applying Annual-Worth Analysis
 - Life-Cycle Cost Analysis
 - Design Economics
- **Rate-of-Return Analysis**
 - Rate of Return and Methods of Finding It
 - Internal Rate-of-Return Criterion
 - Mutually Exclusive Alternatives
- **Cost Concepts Relevant to Decision Making**
 - General Cost Terms; Classifying Costs for Financial Statements
 - Cost Classifications for Predicting Cost Behavior
 - Future Costs for Business Decisions
 - Estimating Profit from Production
- **Depreciation and Corporate Taxes**
 - Asset Depreciation: Economic versus Accounting
 - Book and Tax Depreciation Methods (MACRS)
 - Depletion
 - Income Tax Rate to be used in Economic Analysis
 - The Need for cash Flow in Engineering Economic Analysis
- **Developing Project Cash Flows**
 - Cost-Benefit Estimation for Engineering Projects
 - Developing Cash Flow Statements
- **Project Risk and Uncertainty**
 - Origins of Project Risk
 - Methods of Describing Project Risk: Sensitivity, Break-Even and Scenario Analysis
- **Special Topics in Engineering Economics**
 - Replacement Decisions
 - Capital Budgeting Decisions
 - Economic Analysis in the Service Sector

Books Recommended:

1. Software Engineering Economics and Declining Budgets by Pamela T. Geriner, Thomas R. Gulledge, William P. Hutzler, Springer Verlag, (Latest Edition)
2. Estimating Software Costs: Bringing Realism to Estimating, Capers Jones, McGrawHill Osborne Media; (Latest Edition)
3. Software Cost Estimation and Sizing Methods, Issues, and Guidelines, Shari Lawrence Pfleeger, Rand Publishing, (Latest Edition)
4. Contemporary Engineering Economics by Chan S. Park, latest edition, Pearson, ISBN: 9780134105598
5. Engineering Economic Analysis by Donald G. Newnan, Jerome P. Lavelle, Ted G. Eschenbach, latest edition, Oxford University Press, ISBN: 978- 0199339273
6. Engineering Economy by Leland T. Blank and Anthony Tarquin, latest edition

Approval:

Board of Studies:
Board of Faculty:
Academic Council:

Resolution No. 02
Resolution No. 01
Resolution No. 96.10

Dated: 29-08-2019
Dated: 07-10-2019
Dated: 07-10-2019



Mehran University of Engineering and Technology, Jamshoro
Department of Software Engineering



Title of Subject	:	<u>Operating Systems (SW-225)</u>
Discipline	:	Software Engineering (4 th Semester)
Effective	:	18 Batch & onwards
Pre-requisite	:	Introduction to Information & Communication Technologies
Assessment	:	Theory: 20% Sessional, 80% Written Semester Examination (20% Mid, 60% Final) Practical: 40% Sessional, 60% Final Examination
Credit Hours	:	03 + 01
Minimum Contact Hours:		45 + 45
		Marks: 100 + 50

COURSE LEARNING OUTCOMES:

Upon successful completion of the course, the student will be able to:

CLOs	Description	Taxonomy level	PLO
1	Describe the core concepts of operating system ,memory ,process and file management.	C2	1
2	Determine the applications and principles on which the core functions of the operating systems are built on.	C3	1
3	Execute modern operating systems using learned knowledge and skills.	P4	5

PROGRAM LEARNING OUTCOMES (PLOs):

The course is designed so that students will achieve the following PLOs:

1	Engineering Knowledge:	<input checked="" type="checkbox"/>	7	Environment and Sustainability:	<input type="checkbox"/>
2	Problem Analysis:	<input type="checkbox"/>	8	Ethics:	<input type="checkbox"/>
3	Design/Development of Solutions:	<input type="checkbox"/>	9	Individual and Team Work:	<input type="checkbox"/>
4	Investigation:	<input type="checkbox"/>	10	Communication:	<input type="checkbox"/>
5	Modern Tool Usage:	<input checked="" type="checkbox"/>	11	Project Management:	<input type="checkbox"/>
6	The Engineer and Society:	<input type="checkbox"/>	12	Lifelong Learning:	<input type="checkbox"/>

Course outline:

- Basic Elements: Evolution of the Microprocessor, Instruction Execution.
- Interrupts: Interrupts and the Instruction Cycle, Interrupt Processing, Multiple Interrupts.
- Memory: The Memory Hierarchy, Cache Memory, Direct Memory Access.
- Operating System: Objectives and Functions, The Evolution of Operating Systems, Developments Leading to Modern Operating Systems.
- Virtual Machines: Virtual Machine Architecture.
- Multiprocessor and Multicore Organization: OS Design Considerations for Multiprocessor and Multicore.
- Processes: Process Definition, Process States, Process Description, Process Control, OS Execution.
- Processes and Threads: Types of Threads, Multicore and Multithreading.
- Principles of Concurrency.
- Mutual Exclusion, Hardware Support, Semaphores.
- Multi-Process Synchronization: Producer/Consumer Problem.
- Message Passing: Readers/Writers Problem.
- Deadlocks: Principles of Deadlock, Deadlock Prevention, Detection and Avoidance, An

Integrated Deadlock Strategy.

- Memory Management: Memory Partitioning, Paging, Segmentation, Virtual Memory, Hardware and Control Structures for Virtual Memory.
- Fetch Policy, Placement Policy, Replacement Policy, Resident Set Management, Cleaning Policy, Load Control.

Practical work to be carried out

1. Installation Procedure of Windows
2. To become familiar with Command Prompt
3. Working with Batch Files and Windows Power Shell
4. Getting familiar with Control Panel and Task Scheduler
5. Working with Windows File Protection and User Privileges
6. Becoming familiar with Network Manager and Windows File Sharing
7. Installing a Desktop Sharing Environment using SSH
8. Creating Ghost Installations and working with Utility Software
9. Installing a Linux Environment/Setting up a Virtual Environment for Linux
10. Getting familiar with Ubuntu GUI
11. Familiarization with some of the most frequently used Linux Commands
12. Introduction to Shell Scripting
13. Simulating CPU Scheduling Algorithms
14. Simulating Memory Management Techniques
15. Case Studies

RECOMMENDED BOOKS:

1. Operating Systems: Internals and Design Principles William Stallings, Latest Edition
2. Operating Systems: D. R. Choffnes, Harvey Deitel, and Paul Deite Latest Edition
3. Incorporating Unix and Windows Colin Ritchie, Latest Edition
4. Operating System Concepts, Abraham Silberschatz, Latest Edition
5. Silberschatz, P. B. Galvin and G. Gagne, "Operating System Concepts, Internals & Design Principles", latest Ed., Wiley.
6. William Stallings "Operating Systems: Internals & Design Principles", latest Ed., Prentice Hall.
7. S. Tanenbaum and A. S. Woodhull "Modern Operating Systems", latest Ed., Prentice Hall.

Approval:

Board of Studies:
Board of Faculty:
Academic Council:

Resolution No. 02
Resolution No. 01
Resolution No. 96.10

Dated: 29-08-2019
Dated: 07-10-2019
Dated: 07-10-2019



Mehran University of Engineering and Technology, Jamshoro
Department of Software Engineering



Title of Subject	:	<u>Computer Networks (SW226)</u>
Discipline	:	Software Engineering (4 th Semester)
Effective	:	18 Batch & onwards
Pre-requisite	:	None
Assessment	:	Theory: 20% Sessional, 80% Written Semester Examination (20% Mid, 60% Final) Practical: 40% Sessional, 60% Final Examination
Credit Hours	:	03 + 01
Minimum Contact Hours:		45 + 45
		Marks: 100 +50

COURSE LEARNING OUTCOMES:

Upon successful completion of the course, the student will be able to:

CLOs	Description	Taxonomy level	PLO
1	Describe basics of networks, internetworking devices, layered network architectures and routing protocols.	C2	1
2	Analyze features, services and operations of various network, transport and application layer protocols of communication stack.	C4	2
3	Solve network layer logical addressing problems efficiently	P3	3
4	Configure and troubleshoot network devices. Analyze network traffic and apply techniques related to the design and analysis of computer networks using simulation tools.	P3	5

PROGRAM LEARNING OUTCOMES (PLOs):

The course is designed so that students will achieve the following PLOs:

1	Engineering Knowledge:	<input checked="" type="checkbox"/>	7	Environment and Sustainability:	<input type="checkbox"/>
2	Problem Analysis:	<input checked="" type="checkbox"/>	8	Ethics:	<input type="checkbox"/>
3	Design/Development of Solutions:	<input checked="" type="checkbox"/>	9	Individual and Team Work:	<input type="checkbox"/>
4	Investigation:	<input type="checkbox"/>	10	Communication:	<input type="checkbox"/>
5	Modern Tool Usage:	<input checked="" type="checkbox"/>	11	Project Management:	<input type="checkbox"/>
6	The Engineer and Society:	<input type="checkbox"/>	12	Lifelong Learning:	<input type="checkbox"/>

Course outline:

- Internet: Definitions and protocols.
- Network core: packet/circuit switching, Internet structure.
- Principles of network applications: Web and HTTP.
- Electronic Mail: SMTP, POP3, IMAP.
- DNS: P2P applications.
- Transport-layer services: Multiplexing and demultiplexing.
- Connectionless transport: UDP.
- Principles of reliable data transfer.
- Connection-oriented transport: TCP Flow control and connection management.
- Principles of congestion control.
- TCP congestion control.
- Network Layer: Virtual circuit and datagram networks, routers.
- IP: Internet Protocol: Datagram format, IPv4 addressing, ICMP, IPv6.

- Routing algorithms: Link state, Distance Vector, Hierarchical routing, Routing in the Internet, RIP, OSPF.
- Link Layer: Error Correction and Detection Techniques, Carrier Sense Multiple Access.

Practical Work to be carried out:

- 1 To work with with cable specification, installation and troubleshooting.
- 2 To design a peer to peer network.
- 3 To become familiar with use of Ping, Traceroute and tracert
- 4 To become familiar with the basic configuration of a switch using packet tracer software
- 5 Configure a network topology using packet tracer software
- 6 To become familiar with command line interface (CLI) and examining router interfaces, setting passwords and saving configuration files of routers.
- 7 To configure the STATIC routes between two routers and to configure the DEFAULT routes between two routers to allow data transfer without the use of dynamic routing protocols.
- 8 To configure the distance vector routing protocol on two routers.
- 9 To configure the RIP dynamic routing protocol on two routers.
- 10 To become familiar with subnetting concepts.
- 11 Solving network layer logical addressing problems efficiently
- 12 To configure the IGRP dynamic routing protocol on two routers.
- 13 To configure the EIGRP dynamic routing protocol on two routers.
- 14 To become familiar with network security.
- 15 Basic Access Control Lists.

Recommended Books:

1. Computer Networks and Internets with Internet Applications, Comer, Douglas E., Prentice-Hall Inc, Latest Edition.
2. CCNA INTRO self-study guide, CISCO, Latest Edition.
3. Data Communication and Networking, BEHROZ A. FOROUZAN. Latest Edition.
4. Data and computer communication, Williams Stallings, Latest Edition.
5. Computer networks with internet technology, William Stallings, Latest Edition.
6. Johnson, Allan. LAN Switching and Wireless, CCNA Exploration Labs and Study Guide (Lab Companion). Cisco Press, latest edition.
7. Rullan, John. Accessing the WAN, CCNA Exploration Labs and Study Guide (Lab Companion). Cisco Press, latest edition.
8. Kreigh, Jack. "CCIE Routing and Switching Lab Exam Prep." Latest edition

Approval:

Board of Studies:

Resolution No. 02

Dated: 29-08-2019

Board of Faculty:

Resolution No. 01

Dated: 07-10-2019

Academic Council:

Resolution No. 96.10

Dated: 07-10-2019



Mehran University of Engineering and Technology, Jamshoro
Department of Software Engineering



Title of Subject	:	<u>Software Design and Architecture (SW-227)</u>
Discipline	:	Software Engineering (4 th Semester)
Effective	:	18 Batch & onwards
Pre-requisite	:	Introduction to Software Engineering, Software Requirement Engineering
Assessment	:	Theory: 20% Sessional, 80% Written Semester Examination (20% Mid, 60% Final) Practical: 40% Sessional, 60% Final Examination
Credit Hours	:	02 + 01
Minimum Contact Hours:		30 + 45
		Marks: 50+ 50

COURSE LEARNING OUTCOMES:

Upon successful completion of the course, the student will be able to:

CLOs	Description	Taxonomy level	PLO
1	Examine software architecture, differentiate architecture styles, methods for best use of architecture	C4	2
2	Formulate design patterns to generate code for different applications	C5	3
3	Evaluate the UML models of designed softwares and compare them	C6	4

PROGRAM LEARNING OUTCOMES (PLOs):

The course is designed so that students will achieve the following PLOs:

1	Engineering Knowledge:	<input type="checkbox"/>	7	Environment and Sustainability:	<input type="checkbox"/>
2	Problem Analysis:	<input checked="" type="checkbox"/>	8	Ethics:	<input type="checkbox"/>
3	Design/Development of Solutions:	<input checked="" type="checkbox"/>	9	Individual and Team Work:	<input type="checkbox"/>
4	Investigation:	<input checked="" type="checkbox"/>	10	Communication:	<input type="checkbox"/>
5	Modern Tool Usage:	<input type="checkbox"/>	11	Project Management:	<input type="checkbox"/>
6	The Engineer and Society:	<input type="checkbox"/>	12	Lifelong Learning:	<input type="checkbox"/>

Course outline:

- Software Design Concepts, Design principles
- System design and software architecture
- Architectural design issues
- Software Architecture, Architectural Structures & Styles-,
- Architectural Patterns, Interactive systems with MVC architecture
- Middleware and service-oriented architecture
- Component based design and development
- Model driven development
- Object-Oriented Design with UML, Exploring inheritance and object composition
- Data design, Persistent layer design
- Functional Design
- User interface design
- Web applications design
- Mobile application design

Practical Work to be carried out:

1. Introduction to UML and to become familiar with use case diagrams

2. Class Diagram
3. Interaction diagrams
4. State Transition Diagram
5. Sequence diagram
6. To implement Singleton Pattern
7. To implement Abstract Factory Pattern
8. To implement Adapter Pattern
9. To implement Decorator Pattern
10. To implement Observer Pattern
11. To implement Iterator Pattern
12. CRC Modeling
13. Procedural design (Pseudo code)
14. Procedural design (Decision table)
15. Case study

Recommended Books:

1. Software Architecture in Practice, Len Bass, Paul Clements, Rick Kaman , Pearson Addison (Text Book)
2. Software Architecture in Practice, Latest Edition
3. Evaluating Software Architectures: Methods and Case Studies.
4. Teach Yourself UML in 24 Hours, Joseph Schmuller, Latest Edition
5. Software Engineering, A Practitioner's Approach, Pressman R. S.& Maxim B. R., latest Edition, McGraw-Hill.
6. Object-Oriented Analysis, Design and Implementation, Brahma Dathan, Sarnath Ramnath, latest Edition, Springer, University Press.
7. Essential Software Architecture, Gorton I., latest Edition, Springer – Verlag

Approval:

Board of Studies:

Resolution No. 02

Dated: 29-08-2019

Board of Faculty:

Resolution No. 01

Dated: 07-10-2019

Academic Council:

Resolution No. 96.10

Dated: 07-10-2019



Mehran University of Engineering and Technology, Jamshoro
Department of Software Engineering



Title of Subject	:	<u>Information Security (SW-316)</u>	
Discipline	:	Software Engineering (5 th Semester)	
Effective	:	17 Batch & onwards	
Pre-requisite	:	Computer Networks	
Assessment	:	Theory: 20% Sessional, 80% Written Semester Examination (20% Mid, 60% Final)	
Credit Hours	:	03 + 00	Marks: 100
Minimum Contact Hours:		45	

COURSE LEARNING OUTCOMES:

Upon successful completion of the course, the student will be able to:

CLOs	Description	Taxonomy level	PLO
1	Explain the concept and knowledge of IS and implementation of cryptography and related techniques using information security concepts.	C2	1
2	Analyze enterprise level security by using different hardware and software Security solutions.	C4	2
3	Use Information security solution for an organization in order to cope with their security issues.	C3	1

PROGRAM LEARNING OUTCOMES (PLOs):

The course is designed so that students will achieve the following PLOs:

1	Engineering Knowledge:	<input checked="" type="checkbox"/>	7	Environment and Sustainability:	<input type="checkbox"/>
2	Problem Analysis:	<input checked="" type="checkbox"/>	8	Ethics:	<input type="checkbox"/>
3	Design/Development of Solutions:	<input type="checkbox"/>	9	Individual and Team Work:	<input type="checkbox"/>
4	Investigation:	<input type="checkbox"/>	10	Communication:	<input type="checkbox"/>
5	Modern Tool Usage:	<input type="checkbox"/>	11	Project Management:	<input type="checkbox"/>
6	The Engineer and Society:	<input type="checkbox"/>	12	Lifelong Learning:	<input type="checkbox"/>

Course outline:

- Introduction to Information Security
- Need for security
- Legal, ethical and Professional issues in information security
- Planning for security
- Risk Management
- Security technology: Access control, Firewalls and VPNs
- Intrusion detection and prevention systems
- Cryptography
- Unintentional programming oversights, Malicious code, countermeasures
- Browser attacks, Web attacks, E-mail attacks
- Security in operating systems
- Network security attacks and security counter measures
- Security in databases
- Cloud security tools and techniques

- Physical security

Recommended Books:

1. Corporate Computer and Network Security by Raymond Panko, Prentice Hall, Latest Edition.
2. CISSP, Shon Harris, McGraw-Hill Osborne Media, Latest Edition.
3. Cryptography and Network Security, William Stallings, Prentice Hall, Latest Edition.
4. Principles of Information Security, Michael E. Whitman and Herbert J. Mattord, latest Edition, Cengage Learning.
5. Security in Computing, Pfleeger, C.P., Pfleeger, S.L. & Margulies, J., latest Edition, Prentice Hall.

Approval:

Board of Studies:
Board of Faculty:
Academic Council:

Resolution No. 02
Resolution No. 01
Resolution No. 96.10

Dated: 29-08-2019
Dated: 07-10-2019
Dated: 07-10-2019



Mehran University of Engineering and Technology, Jamshoro
Department of Software Engineering



Title of Subject	:	<u>Software Construction & Development (SW-315)</u>
Discipline	:	Software Engineering (5 th Semester)
Effective	:	17 Batch & onwards
Pre-requisite	:	Software Design and Architecture
Assessment	:	Theory: 20% Sessional, 80% Written Semester Examination (20% Mid, 60% Final) Practical: 40% Sessional, 60% Final Examination
Credit Hours	:	02 + 01
Minimum Contact Hours:		30 + 45
		Marks: 50 + 50

COURSE LEARNING OUTCOMES:

Upon successful completion of the course, the student will be able to:

CLOs	Description	Taxonomy level	PLO
1	Recognize the software engineering process from basic to advanced level.	C4	2
2	Organize the intricacies of software deployment and configuration.	C5	3
3	Use knowledge to develop, deploy and configure variety of softwares.	P4	5

PROGRAM LEARNING OUTCOMES (PLOs):

The course is designed so that students will achieve the following PLOs:

1	Engineering Knowledge:	<input type="checkbox"/>	7	Environment and Sustainability:	<input type="checkbox"/>
2	Problem Analysis:	<input checked="" type="checkbox"/>	8	Ethics:	<input type="checkbox"/>
3	Design/Development of Solutions:	<input checked="" type="checkbox"/>	9	Individual and Team Work:	<input type="checkbox"/>
4	Investigation:	<input type="checkbox"/>	10	Communication:	<input type="checkbox"/>
5	Modern Tool Usage:	<input checked="" type="checkbox"/>	11	Project Management:	<input type="checkbox"/>
6	The Engineer and Society:	<input type="checkbox"/>	12	Lifelong Learning:	<input type="checkbox"/>

Course outline:

- Moving from design to implementation code
- Regular expressions and grammars, Parser generators
- Software Construction fundamentals
- Designing specifications
- Mutability and immutability
- Recursion and Abstract data types
- Abstraction functions and rep invariants
- Interfaces, Graphical user interfaces
- Recursive data types
- Concurrency, thread safety, Sockets and networking
- Queues and message passing, Locks and synchronization
- Code refactoring
- Exception handling
- Static checking and testing
- Code review

Practical Work to be carried out:

- 1 Software Development Methods
- 2 Class Diagram
- 3 Sequence Diagram
- 4 Object Diagram
- 5 Activity Diagram
- 6 Requirements gathering strategies
- 7 Development of SRS document
- 8 OOP paradigm
- 9 Agile Frameworks
- 10 OOAD optimization
- 11 Unit Testing
- 12 Integration Testing
- 13 Software Deployment
- 14 Configuration Management
- 15 Case study/Project

Recommended Books:

1. Introduction to Software engineering by Ian Sommerville. (Latest Edition)
2. Software Configuration management by Professor Hossein Saydian
3. Working with effectively with Legacy code by Michael C. Feathers. (Latest Edition)
4. Client-Centered Software Development, Allen B. Tucker, CRC Press, latest edition.
5. Software Essentials: Design and Construction, Adiar Dingle, CRC Press, latest edition.
6. Clean Code: A Handbook of Agile Software Craftsmanship, Robert C. Martin, Prentice Hall, latest edition.

Approval:**Board of Studies:****Board of Faculty:****Academic Council:****Resolution No. 02****Resolution No. 01****Resolution No. 96.10****Dated: 29-08-2019****Dated: 07-10-2019****Dated: 07-10-2019**



Mehran University of Engineering and Technology, Jamshoro
Department of Software Engineering



Title of Subject	:	<u>Mobile Application Development (SW-327)</u>
Discipline	:	Software Engineering (6 th Semester)
Effective	:	17 Batch & onwards
Pre-requisite	:	Operation Research
Assessment	:	Theory: 20% Sessional, 80% Written Semester Examination (20% Mid, 60% Final) Practical: 40% Sessional, 60% Final Examination
Credit Hours	:	03 + 01
Minimum Contact Hours:		45 + 45
		Marks: 100 + 50

COURSE LEARNING OUTCOMES:

Upon successful completion of the course, the student will be able to:

CLOs	Description	Taxonomy level	PLO
1	Explain the basic concepts used in various Mobile Application Development Frameworks.	C3	1
2	Use mobile application components and compare the different performance trade-offs	C4	2
3	Create mobile application solutions to real world problems	C5	3
4	Develop mobile applications using current software development environments	P3	5

PROGRAM LEARNING OUTCOMES (PLOs):

The course is designed so that students will achieve the following PLOs:

1	Engineering Knowledge:	<input checked="" type="checkbox"/>	7	Environment and Sustainability:	<input type="checkbox"/>
2	Problem Analysis:	<input checked="" type="checkbox"/>	8	Ethics:	<input type="checkbox"/>
3	Design/Development of Solutions:	<input checked="" type="checkbox"/>	9	Individual and Team Work:	<input type="checkbox"/>
4	Investigation:	<input type="checkbox"/>	10	Communication:	<input type="checkbox"/>
5	Modern Tool Usage:	<input checked="" type="checkbox"/>	11	Project Management:	<input type="checkbox"/>
6	The Engineer and Society:	<input type="checkbox"/>	12	Lifelong Learning:	<input type="checkbox"/>

Course outline:

- **Mobiles Application Development Platform**
 - HTML5 for Mobiles;
- **Android OS Architecture**
 - Framework and Application Development;
- **IOS: Architecture**
 - Framework and Application Development;
- **Windows Mobile Architecture**
 - Framework and Application Development;
- **Calling Built-in Applications using Intents;**
- **Displaying Notifications; Components of a Screen;**
 - Adapting to Display Orientation;
 - Managing Changes to Screen Orientation;
 - Utilizing the Action Bar;
- **Creating the User Interface;**
 - Listening for UI Notifications; Views; User Preferences;

- **Persisting Data; Sharing Data;**
- **Sending SMS Messages; Getting Feedback; Sending Email;**
- **Consuming Web Services Using HTTP;**
 - Web Services: Accessing and Creating;
 - Displaying Maps;
 - Location-aware Applications;
- **Publishing Android Applications; Deployment on App Stores;**
- **Mobile Platform Constraints; Emerging Technologies**

Practical Work to be carried out:

- 1 Setting up the Development Environment for building Mobile Applications
- 2 Creating User Interfaces for Mobile Apps
- 3 Writing programs for Mobile Application
- 4 Working with Intents
- 5 Writing programs for reading sensors
- 6 Using Native google Maps and Geocoding
- 7 Using Concurrent programming in Mobile Applications
- 8 Making program to use Bluetooth
- 9 Working with Broadcast Receivers
- 10 Using Fragments, fragment lifecycle and fragment transactions
- 11 Programming Services
- 12 Writing program for SQLite Database to store and retrieve data
- 13 Integrating Facebook API with Mobile Apps
- 14 App packaging and publishing it to an app store
- 15 Case study/Project

Recommended Books:

1. Reto Meier.; “Professional Android Application Development”, Latest Edition, Wrox publishers.
2. TommiMikonen; “Programming Mobile Devices an Introduction for Practitioners”, Latest Edition, Wiley publishers, Latest Edition.
3. Professional Android application development, Reto Meier, Wrox
4. Programmer to Programmer, latest edition.
5. iOS Programming: The Big Nerd Ranch Guide, Conway, J., Hillegass, A., & Keur, C., latest Edition.
6. Android Programming: The Big Nerd Ranch Guides, Phillips, B. & Hardy, B., latest Edition

Approval:

Board of Studies:

Resolution No. 02

Dated: 29-08-2019

Board of Faculty:

Resolution No. 01

Dated: 07-10-2019

Academic Council:

Resolution No. 96.10

Dated: 07-10-2019



Mehran University of Engineering and Technology, Jamshoro
Department of Software Engineering



Title of Subject	:	<u>Software Project Management (SW - 322)</u>	
Discipline	:	Software Engineering (6 th Semester)	
Effective	:	17 Batch & onwards	
Pre-requisite	:	Introduction To Software Engineering	
Assessment	:	Theory: 20% Sessional, 80% Written Semester Examination (20% Mid, 60% Final)	
Credit Hours	:	03 + 0	Marks: 100
Minimum Contact Hours:		45	

COURSE LEARNING OUTCOMES:

Upon successful completion of the course, the student will be able to:

CLOs	Description	Taxonomy level	PLO
1	Explain principles of project lifecycle and how to identify opportunities on relevant and appropriate project scenarios.	C3	1
2	Critically evaluate and discuss the issues around project management and its application in the real world.	C6	2, 4
3	Incorporate project management techniques to initiate, plan, execute and work in teams to create a project plan for a project scenario that includes key tasks, critical paths, dependencies and realistic timeline.	C5	6,9,11

PROGRAM LEARNING OUTCOMES (PLOs):

The course is designed so that students will achieve the following PLOs:

1	Engineering Knowledge:	<input checked="" type="checkbox"/>	7	Environment and Sustainability:	<input type="checkbox"/>
2	Problem Analysis:	<input checked="" type="checkbox"/>	8	Ethics:	<input type="checkbox"/>
3	Design/Development of Solutions:	<input type="checkbox"/>	9	Individual and Team Work:	<input checked="" type="checkbox"/>
4	Investigation:	<input checked="" type="checkbox"/>	10	Communication:	<input type="checkbox"/>
5	Modern Tool Usage:	<input type="checkbox"/>	11	Project Management:	<input checked="" type="checkbox"/>
6	The Engineer and Society:	<input checked="" type="checkbox"/>	12	Lifelong Learning:	<input type="checkbox"/>

Course outline:

- Introduction to Software Project Management
- Project Management concepts
- Project Management Tools
- PMI's Knowledge areas, PMI Framework
- PMI Process Groups.
- Understanding Organizations. Project Planning, Project Evaluation
- Selection of an Appropriate Approach in Project
- Software Effort Estimation, Activity Planning
- Risk Management, Evaluating the Risks to the Schedule, Risk Control
- Configuration Management and Maintenance
- Environment for Configuration Control
- Resource Allocation
- Monitoring & Control
- Review and Evaluation
- Challenges of Outsourcing in Project Management

Books Recommended:

1. Software Engineering- A practitioner's Approach by Rogers Pressman Latest Edition.
2. Software Project management, Bob Hughes and Mike Cottere Latest Edition
3. Information Technology Project Management, Schwalbe, K., latest Edition, Cengage Learning.
4. A guide to the project management body of knowledge (PMBOK guide), Project Management Institute Inc, latest Edition.
5. Software Project Management, Bob Hughes and Mike Cotterell, McGraw-Hill Education; latest Edition.

Approval:**Board of Studies:****Resolution No. 02****Dated: 29-08-2019****Board of Faculty:****Resolution No. 01****Dated: 07-10-2019****Academic Council:****Resolution No. 96.10****Dated: 07-10-2019**



Mehran University of Engineering and Technology, Jamshoro
Department of Software Engineering



Title of Subject	:	<u>Software Re-Engineering (SW-415)</u>
Discipline	:	Software Engineering (7 th Semester)
Effective	:	F16 Batch & onwards
Pre-requisite	:	Introduction to Software Engineering, Software requirement Engineering, Software Economics and Management, Software Design And Architecture
Assessment	:	Theory: 20% Sessional, 80% Written Semester Examination (20% Mid, 60% Final)
Credit Hours	:	03 + 0
Minimum Contact Hours:		45

Marks: 100

COURSE LEARNING OUTCOMES:

Upon successful completion of the course, the student will be able to:

CLOs	Description	Taxonomy level	PLO
1	Explain the concepts and technique of software re- engineering.	C3	1
3	Analyze problems associated with object oriented software systems.	C4	2
2	Use reengineering techniques to maintain and modify software systems	C5	3
4	Perform complex design reengineering and reverse engineering problems.	C6	4

PROGRAM LEARNING OUTCOMES (PLOs):

The course is designed so that students will achieve the following PLOs:

1	Engineering Knowledge:	<input checked="" type="checkbox"/>	7	Environment and Sustainability:	<input type="checkbox"/>
2	Problem Analysis:	<input checked="" type="checkbox"/>	8	Ethics:	<input type="checkbox"/>
3	Design/Development of Solutions:	<input checked="" type="checkbox"/>	9	Individual and Team Work:	<input type="checkbox"/>
4	Investigation:	<input checked="" type="checkbox"/>	10	Communication:	<input type="checkbox"/>
5	Modern Tool Usage:	<input type="checkbox"/>	11	Project Management:	<input type="checkbox"/>
6	The Engineer and Society:	<input type="checkbox"/>	12	Lifelong Learning:	<input type="checkbox"/>

Course outline:

- **SOFTWARE RE-ENGINEERING:**

The Reuse Process, Describing Reusable Components, Management Issues

- **FUNDAMENTAL RE-ENGINEERING TECHNIQUES:**

to modernize legacy systems including source code analysis, architecture recovery, and code restructuring, migration to Object Oriented platforms, migration to network-centric environments, and software integration,

- **SOFTWARE REFACTORING STRATEGIES:**

Forward and Reverse Engineering, program comprehension, source code transformation and refactoring strategies, software maintenance and re-engineering economics.

- **RE-ENGINEERING COST FACTORS:**

quality of the software, tool support, required data conversion, availability of expert staff for re-engineering

Recommended Books:

1. Re-engineering legacy software, David Lorge Parnas, Chris Birchall, Safari Books, Shelter Island, NY, Latest Edition
2. Reengineering, Priyadarshi Tripathy and Kshirasagar Naik, John Wiley & Sons, Inc. Latest Edition
3. Software Maintenance and Evolution: a Roadmap, K.H.Bennett and V.T Rajlich, The Future of Software Engineering, ACM Press, Latest Edition

Approval:

Board of Studies:

Resolution No. 02

Dated: 29-08-2019

Board of Faculty:

Resolution No. 01

Dated: 07-10-2019

Academic Council:

Resolution No. 96.10

Dated: 07-10-2019



Mehran University of Engineering and Technology, Jamshoro
Department of Software Engineering



Title of Subject	:	<u>Web Engineering (SW-417)</u>
Discipline	:	Software Engineering (7 th Semester)
Effective	:	F16 Batch & onwards
Pre-requisite	:	Programming Fundamentals
Assessment	:	Theory: 20% Sessional, 80% Written Semester Examination (20% Mid, 60% Final) Practical: 40% Sessional, 60% Final Examination
Credit Hours	:	03 + 01
Minimum Contact Hours:		45 + 45
		Marks: 100 +50

COURSE LEARNING OUTCOMES:

Upon successful completion of the course, the student will be able to:

CLOs	Description	Taxonomy level	PLO
1	Outline fundamentals of Web technologies and explain HTML, CSS and Javascript for client-side web interfacing.	C3	1
2	Work with server-side languages to create dynamic web applications.	C5	3
3	Host web applications and create databases to store and process data proficiently	P5	5,9

PROGRAM LEARNING OUTCOMES (PLOs):

The course is designed so that students will achieve the following PLOs:

1	Engineering Knowledge:	<input checked="" type="checkbox"/>	7	Environment and Sustainability:	<input type="checkbox"/>
2	Problem Analysis:	<input type="checkbox"/>	8	Ethics:	<input type="checkbox"/>
3	Design/Development of Solutions:	<input checked="" type="checkbox"/>	9	Individual and Team Work:	<input checked="" type="checkbox"/>
4	Investigation:	<input type="checkbox"/>	10	Communication:	<input type="checkbox"/>
5	Modern Tool Usage:	<input checked="" type="checkbox"/>	11	Project Management:	<input type="checkbox"/>
6	The Engineer and Society:	<input type="checkbox"/>	12	Lifelong Learning:	<input type="checkbox"/>

Course outline:

• **Client Side Technologies & Responsive Web:**

➤ **HTML:**

Text: headings, paragraphs, bold & italic, superscript & subscript, white space, line breaks & horizontal rules, visual editors & their code views, semantic markup, strong & emphasis, quotations, abbreviations & acronyms, citations & definitions, author details,

Lists & tables: ordered lists, unordered lists, definition lists, nested lists, basic table structure, table headings, spanning columns, spanning rows, long tables

Links: writing links, linking to other sites, linking to other pages on the same site, directory structure, relative urls, email links, opening links in a new window, linking to a specific part of the same page, linking to a specific part of another page

Images: choosing images for website, storing images on site, adding images, height & width of images, where to place images in your code, aligning images, image dimensions, cropping images, image resolution, vector images, animated gifs, transparency

Forms: form controls, form structure, text input, password input, text area, radio button, drop down list box, multiple select box, file input box, submit button, image button, html5: form validation

HTML5: Semantic elements, multimedia elements, graphic elements, form elements and APIs

- **CSS:** External CSS, Internal CSS, Inline CSS, CSS selectors, Color, Foreground color, Background color, text, Typeface terminology, Size of type, Type scales, Units of type size, Bold, Italic, lowercase, Underline & strike, Letter & word spacing, Alignment, Vertical alignment, Indenting text, Styling links, boxes, Border, margin & padding, White space & vertical margin, Border width, Border style, Border Color, lists, tables and forms, Bullet point styles, Images for bullets, List shorthand, Table properties, images, Controlling sizes of images, Aligning images, Background images, repeating images, CSS selectors, Bootstrap framework.
- **JavaScript:** Variables, Operators, Functions, Document Object Model, Expressions and Control Flow in JavaScript, Looping, JavaScript Events, JavaScript Objects, JavaScript Arrays, Validating User Input with JavaScript, Regular Expressions
- **Server Side Technologies:**
 - Introduction to PHP, Expressions and Control Flow in PHP, Operators, Conditionals, Looping, Implicit and Explicit Casting, PHP Functions and Objects, Including and Requiring Files, PHP Objects, PHP Arrays, Multidimensional Arrays, Array Functions, Date and Time Functions, File Handling, Form handling, Cookies, Sessions and Authentication
 - **Server Side Databases:** MySQL Basics, Accessing MySQL, Indexes, MySQL Functions, Querying a MySQL Database with PHP
 - **Server Side MVC Framework:** MVC Framework, PHP MVC Frameworks, Codeigniter Installation and Configuration, Creating Applications With Custom Codeigniter Controllers, Views and Models.

Practical Work to be carried out:

- 1 To understand HTML basics for developing web applications
- 2 To become familiar with HTML5 semantic and multimedia elements
- 3 To become familiar with Javascript basics for user interaction
- 4 To become familiar with CSS for styling web applications
- 5 To become familiar with Bootstrap styling framework
- 6 To become familiar with PHP basics and installation of web server
- 7 To become familiar with control structures and arrays in PHP
- 8 To become familiar with functions and form handling and file handling in PHP
- 9 To become familiar with AJAX for data fetching
- 10 To become familiar with sessions and cookie management in PHP
- 11 To become familiar with database connectivity using PHP
- 12 To become familiar with web services using PHP
- 13 To become familiar with Codeigniter PHP MVC Framework
- 14 To become familiar with Kubernetes environment and understand its basics
- 15 To understand basics of dockers for developing web applications

Recommended Books:

1. Jon Duckett, “HTML and CSS: Design and build web sites” (Wiley, Latest Edition)
2. Robin Nixon, “Learning PHP, MySQL & Java Script with jQuery, CSS & HTML5” (O’Rielly, Latest Edition).

Approval:

Board of Studies:

Resolution No. 02

Dated: 29-08-2019

Board of Faculty:

Resolution No. 01

Dated: 07-10-2019

Academic Council:

Resolution No. 96.10

Dated: 07-10-2019



Mehran University of Engineering and Technology, Jamshoro
Department of Software Engineering



Title of Subject	:	<u>Software Quality Engineering (SW-426)</u>
Discipline	:	Software Engineering (8 th Semester)
Effective	:	F16 Batch & onwards
Pre-requisite	:	Formal Methods in Software Engineering, Software Re-engineering
Assessment	:	Theory: 20% Sessional, 80% Written Semester Examination (20% Mid, 60% Final) Practical: 40% Sessional, 60% Final Examination
Credit Hours	:	03 + 01
Minimum Contact Hours:		45 + 45
		Marks : 100 + 50

COURSE LEARNING OUTCOMES:

Upon successful completion of the course, the student will be able to:

CLOs	Description	Taxonomy level	PLO
1	Explain basics of software quality assurance and testing fundamentals	C3	1
2	Analyze different scenarios to grasp working mechanism of various testing techniques	C4	2
3	Construct Test cases and perform testing on various applications using Modern Tools	P4	5

PROGRAM LEARNING OUTCOMES (PLOs):

The course is designed so that students will achieve the following PLOs:

1	Engineering Knowledge:	<input checked="" type="checkbox"/>	7	Environment and Sustainability:	<input type="checkbox"/>
2	Problem Analysis:	<input checked="" type="checkbox"/>	8	Ethics:	<input type="checkbox"/>
3	Design/Development of Solutions:	<input type="checkbox"/>	9	Individual and Team Work:	<input type="checkbox"/>
4	Investigation:	<input type="checkbox"/>	10	Communication:	<input type="checkbox"/>
5	Modern Tool Usage:	<input checked="" type="checkbox"/>	11	Project Management:	<input type="checkbox"/>
6	The Engineer and Society:	<input type="checkbox"/>	12	Lifelong Learning:	<input type="checkbox"/>

Course outline:

- Introduction to Software Quality and Quality Engineering
- Software quality factors and attributes, software quality models
- SQA Process Implementation Activities
- SQA plan and project plan
- Software inspections, Software reviews, Inspection checks and metrics
- Software testing concepts, issues and techniques, Software testing lifecycle
- Software quality metrics, product metrics, process metrics
- Test Planning Process, Testing documentation
- Software testing techniques, Testing philosophies
- Testing strategies, Model based testing,
- Software testing techniques, Testing using models,
- Domain and combinatorial testing,
- Unit and integration testing, Acceptance testing,
- Test automation
- Open issues on software testing

Practical Work to be carried out:

- 1 Validation and verification
- 2 Developing test cases
- 3 Application of Black box testing
- 4 Applying OO testing strategies
- 5 Unit testing using JUnit
- 6 Creating test suites using JUnit
- 7 Creating Mocks using in Mockito
- 8 Integration testing
- 9 Coding and testing cross reference
- 10 Web Functional Testing using Selenium
- 11 Application of formal methods
- 12 Producing log Using Log4J
- 13 Working with performance testing
- 14 Working with load testing
- 15 Case study

Recommended Books:

1. Software Engineering, Practitioners Approach, Roger S. Pressman, Mc.Graw Hill Inc, Latest Edition.
2. Foundation of Software Testing, Dorothy Graham, Cengage Learning EMEA Publishers, Latest Edition.
3. Software Quality: Concepts and Practice, Daniel Galin, John Wiley & Sons, latest edition
4. Introduction to Software Testing, Paul Ammann and Jeff Offutt, latest Edition, Cambridge University Press.
5. Introduction to Software Quality, Gerard O'Regan, Springer International, latest edition.

Approval:

Board of Studies:

Resolution No. 02

Dated: 29-08-2019

Board of Faculty:

Resolution No. 01

Dated: 07-10-2019

Academic Council:

Resolution No. 96.10

Dated: 07-10-2019