INSTITUTE OF AERONAUTICAL ENGINEERING

(Autonomous)
Dundigal, Hyderabad -500 043

COMPUTER SCIENCE AND ENGINEERING

COURSE DESCRIPTOR

Course Title	**SOFTWARE ENGINEERING**				
Course Code	ACS008				
Programme	B.Tech				
Semester	IV	IT			
	V	CSE			
Course Type	Core				
Regulation	IARE - R16				
Course Structure	**Theory**			**Practical**	
	Lectures	**Tutorials**	**Credits**	**Laboratory**	**Credits**
	3	1	4	3	2
Chief Coordinator	Mr. C Raghavendra, Assistant Professor				
Course Faculty	Dr.Y Mohana Roopa, Professor Ms.CH Srividya, Assistant Professor Ms. J Hareesha, AssistantProfessor				

I. COURSEOVERVIEW:

The Present course concentrates on developing basic understanding about various activities that are involved in a software development. This course enables the student to develop necessary skills for developing a product or applications. The course focuses on all activities involved in software development (communication, planning, modeling, construction, deployment).In this course; students will gain a broad understanding of the discipline of software engineering and its application to the development and management of software systems.

II. COURSEPRE-REQUISITES:

Level	Course Code	Semester	Prerequisites	Credits
-	-	-	Fundamentals of computer hardware and software	-

III. MARKSDISTRIBUTION:

Subject	SEE Examination	CIAExamination	Total Marks
Software Engineering	70 Marks	30 Marks	100

IV. DELIVERY /INSTRUCTIONALMETHODOLOGIES:

X	Chalk & Talk	√	Quiz	√	Assignments	X	MOOCs
√	LCD/ PPT	√	Seminars	X	Mini Project	√	Videos
X	Open Ended Experiments						

V. EVALUATIONMETHODOLOGY:

The course will be evaluated for a total of 100 marks, with 30 marks for Continuous Internal Assessment (CIA) and 70 marks for Semester End Examination (SEE). Out of 30 marks Allotted for CIA during the semester, marks are awarded by taking average of two CIA examinations or the marks scored in the make-up examination**.**

Semester End Examination (SEE): The SEE is conducted for 70 marks of 3 hours duration. The syllabus for the theory courses is divided into five units and each unit carries equal weightage In terms of marks distribution. The question paper pattern is as follows: Two full questions with “either‟ or ‟ choice” will be drawn from each unit. Each question carries 14 marks. There could be a maximum of two sub divisions in a question**.**

The emphasis on the questions is broadly based on the following criteria:

50 %	To test the objectiveness of the concept.
50 %	To test the analytical skill of the concept OR to test the application skill of the concept.

Continuous Internal Assessment (CIA):

CIA is conducted for a total of 30 marks (Table 1), with 25 marks for Continuous Internal Examination (CIE), 05 marks for Quiz / Alternative Assessment Tool (AAT).

1

Table 1: Assessment pattern for CIA

Component	**Theory**		**Total Marks**
Type of Assessment	CIE Exam	Quiz / AAT	
CIA Marks	25	05	30

Continuous Internal Examination(CIE):

Two CIE exams shall be conducted at the end of the 8th and 16th week of the semester respectively. The CIE exam is conducted for 25 marks of 2 hours duration consisting of two parts. Part–A shall have five compulsory questions of one mark each. In part–B, four out of five questions have to be answered where, each question carries 5 marks. Marks are awarded by taking average of marks scored in two CIEexams.

Quiz / Alternative Assessment Tool (AAT):

Two Quiz exams shall be online examination consisting of 20 multiple choice questions and are be answered by choosing the correct answer from a given set of choices (commonly four). Marks shall be awarded considering the average of two quizzes for every course. The AAT may include seminars, assignments, term paper, open ended experiments, five minutes video andMOOCs.

VI. HOW PROGRAM OUTCOMES AREASSESSED:

Program Outcomes (POs)		**Strength**	**Proficiency assessed by**
PO 1	**Engineering knowledge**: Apply the knowledge of mathematics, science, engineering fundamentals, and anengineeringspecialization on the Solution of complex engineeringproblems.	2	Assignments
PO 2	**Problem analysis**: Identify, formulate, review research literature, andanalyzecomplex engineering problemsreaching substantiated Conclusions using first principles of mathematics, natural sciences, and engineering sciences.	2	Assignments
PO 3	**Design/development of solutions**: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and thecultural,societal, and environmental considerations.	2	Assignments

PO 10	**Communication**: Communicate effectively 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.	2	Seminars

3= High; 2 = Medium; 1 = Low

VII. HOW PROGRAM SPECIFIC OUTCOMES AREASSESSED:

Program Specific Outcomes (PSOs)		Strength	Proficiency assessed by
PSO 1	**Professional Skills:** The ability to understand, analyze and develop computer programs in the areas related to algorithms, system software, multimedia, web design, big data analytics andnetworking for efficient design of computer-based systems of varying complexity.	1	Seminar
PSO 2	**Problem-Solving Skills:** The ability to apply standard practices and strategies in software project development using open-ended programming environments to deliver a quality productforbusiness success.	1	Mini Projects
PSO 3	**Successful Career and Entrepreneurship:** The ability to employ modern computer languages, environments and platforms inCreating innovative career paths to be an entrepreneur and a zest for higher studies.	1	5 minutes video

3= High; 2 = Medium; 1 =Low

VIII. COURSE OBJECTIVES:

The course should enable the students to:	
I	Learn how to elicitate requirements and develop software life cycles.
II	Understand the design considerations for enterprise integration and deployment.
III	Analyze quality assurance techniques and testing methodologies. .
IV	Understand implementation issues such as modularity and coding standards
V	Prepare a project plan for a software project that includes estimates of size and effort, a schedule, resource allocation, configuration control, and project risk

IX. COURSE OUTCOMES (COs):

COs	COURSE OUTCOMES	CLOs	COURSE LEARNING OUTCOMES
CO1	Identify the approach to risks management through risk identification, risk measurementandrisk mitigation.	CLO 1	Understand the key concerns that are common to all software development processes.
		CLO 2	Identify the appropriate process models approaches and techniques to manage a given software development process.
		CLO 3	Identify the approach to risks management through risk identification, risk measurementandrisk mitigation.
		CLO 4	Use the concept of Earned Value Analysis (EVA) to measure the projects progress at any given point in time, forecasting its completion date and final cost, and analyzing variances in the schedule and budget as the projectproceeds.
		CLO 5	Memorize project planning activities that accurately help in selection and initiationof Individual projects and of portfolios of projects in the enterprise.
		CLO 6	Identify dependability and security issues that affect a given software product.

CO2	Use the concept of classical analysis to determine the acceptance criteria part of specification	CLO 7	Use the concept of classical analysis to determine the acceptance criteria part of specification.
		CLO 8	Memorize the importance of eliciting the requirements for a software product and translate these into a documented design.
		CLO 9	Understand the concept of data dictionary in order to manage the details in large-scale systems, tolocateerrors and omissions in the system.
		CLO 10	Understand the concept of petri nets thatexhibitConcurrency, synchronization and used as a visual communication aid to model the systembehavior.
CO3	Understand the principles of graphical user interface design.	CLO 11	Memorize the design of object oriented softwareusing with the aid of a formal system modeling notation.
		CLO 12	Learn to model the structure and behavior of a software system.
		CLO 13	Memorize different architectural styles, patternsand architectural mapping using data
		CLO 14	Understand the principles of graphical user interface design.
		CLO 15	Understand the concept of component-level design used to define interface characteristics and communication mechanisms for each software component identified in the architectural design.
CO4	Identify the major differences between white box testing and black box testing.	CLO 16	Understand the importance of testing with the performance of root cause analysis.
		CLO 17	Memorize the conceptsof softwaretestingapproaches such as unit testing and integration
		CLO 18	Understand the approaches to verification and validation including static analysis andreviews.
		CLO 19	Identify the major differences between white box testing and black box testing.
		CLO 20	Understand the importance of refactoring which improves the performance of non-functional Attributes of the software.
CO5	Identify the importance of earned value analysis related to project scheduling and also understandthe Various process and project metric used to improve the quality of software.	CLO 21	Learn to manage time, processes and resources effectively by prioritizing competing demands to Achieve personal and team goals.
		CLO 22	Use a proactive, structured risk assessment and analysis activity to identify and analyze rootcauses.
		CLO 23	Understand the concept of risk management through risk identification, risk measurement and Mitigation.
		CLO 24	Memorize the relationship between people and effort.
		CLO 25	Identify the importance of earned value analysis related to project scheduling and also understandthe Various process and project metric used to improve the quality of software.
		CLO 26	Possess the knowledge and skills for employability and to succeed in national and international level competitive exams.

X. COURSE LEARNINGOUTCOMES(CLOs):

CLO Code	CLO's	At the end of the course, the student will have the ability to:	PO's Mapped	Strength of Mapping
ACS008.01	CLO 1	Understand the key concerns that are common to all software development processes.	PO1, PO2	2
ACS008.02	CLO 2	Identify the appropriate process models, approaches and techniques to manage a given software development process.	PO 1, PO 2	2

ACS008.03	CLO 3	Identify the approach to risks management through risk identification, risk measurementandRisk mitigation.	PO 1, PO 2	2
ACS008.04	CLO 4	Use the concept of Earned Value Analysis (EVA) to measure the projects progress at any given point in time, forecasting its completion date and final cost, and analyzing variances in the schedule and budget as the projectproceeds.	PO 1, PO 2, PO 3	2
ACS008.05	CLO 5	Memorize project planning activities that accurately help in selection and initiationofIndividual projects and of portfolios of projects in the enterprise.	PO 1, PO 2	1
ACS008.06	CLO 6	Identify dependability and security issues that affect a given software product.	PO 10	3
ACS008.07	CLO 7	Use the concept of classical analysis to determine the acceptance criteria as part of specification.	PO 3	2
ACS008.08	CLO 8	Memorize the importance of eliciting the requirements for a software product andTranslate these into a documented design.	PO 3	2
ACS008.09	CLO 9	Understand the concept of data dictionary in order to manage the details in large-scale systems, tolocateerrors and omissions in the system.	PO 3	2
ACS008.10	CLO 10	Understand the concept of petri nets thatexhibit Concurrency, synchronization and used as a visual communication aid to model the systembehavior.	PO 2	2
ACS008.11	CLO 11	Memorize the design of object oriented softwareusing with the aid of a formal system modeling notation.	PO 10	3
ACS008.12	CLO 12	Learn to model the structure and behavior of a software system.	PO 1, PO 2	2
ACS008.13	CLO 13	Memorize different architectural styles, patternsand architectural mapping using data	PO 3	3
ACS008.14	CLO 14	Understand the principles of graphical user interface design.	PO 3	3
ACS008.15	CLO 15	Understand the concept of component-level design used to define interface characteristics and communication mechanisms for each software component identified in the architectural design.	PO 3, PO 10	3
ACS008.16	CLO 16	Understand the importance of testing with the performance of root cause analysis.	PO 2	2
ACS008.17	CLO 17	Memorize the concepts ofsoftwaretestingapproaches such as unit testing and integration	PO 1, PO 2	2
ACS008.18	CLO 18	Understand the approaches to verification and validation including static analysis andreviews.	PO 10	3
ACS008.19	CLO 19	Identify the major differences between white box testing and black box testing.	PO 3	2
ACS008.20	CLO 20	Understand the importance of refactoring which improves the performance of non-functional attributes of the software.	PO 1, PO 2	2
ACS008.21	CLO 21	Learn to manage time, processes and resources effectively by prioritizing competing demands toachieve personal and team goals.	PO 3	3
ACS008.22	CLO 22	Use a proactive, structured risk assessment and analysis activity to identify and analyze rootcauses.	PO 3	2
ACS008.23	CLO 23	Understand the concept of risk management through risk identification, risk measurement andmitigation.	PO 10	2
ACS008.24	CLO 24	Memorize the relationship between people and effort.	PO 1, PO10	1
ACS008.25	CLO 25	Identify the importance of earned value analysis related to project scheduling and also	PO 10	2

		understandthevarious process and project metric used to improve the quality of software.		
ACS008.26	CLO 26	Possess the knowledge and skills for employability and to succeed in national and international level competitive exams.	PO 1, PO 10	2

3 = High; 2 = Medium; 1 = Low

XI. MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES:

COURSE OUTCOMES(Cos)	PROGRAM OUTCOMES(POs)			
	PO1	PO2	PO3	PO10
CO1	1	1	3	2
CO2		3	1	
CO3	2	2	2	3
CO4	1	1	1	2
CO5	2	2		1

3 = High; 2 = Medium; 1 = Low

XI. MAPPING COURSE LEARNING OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

CLOs	Program Outcomes (POs)												Program Specific Outcomes (PSOs)		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CLO 1	3	2											2		
CLO 2	2	3											2	1	
CLO 3	2	3													
CLO 4	3	2	3										1		
CLO 5	2	1													1
CLO 6										3					
CLO 7			2												
CLO 8			2												
CLO 9			2												
CLO 10		2													
CLO 11										3			1		
CLO 12	3	1													
CLO 13			3										1		
CLO 14			3										1		
CLO 15			3							3					
CLO 16		2												1	
CLO 17	2	2												2	
CLO 18										3				1	
CLO 19			2											1	
CLO 20	2	2											1		

CLO 21			3											1	
CLO 22			2												
CLO 23										2					1
CLO 24	1									2					1
CLO 25										2					
CLO 26	2									3					

3= High; 2 = Medium; 1 = Low

XII. ASSESSMENT METHODOLOGIES –DIRECT:

CIE Exams	PO1, PO2, PO 3, PSO1, PSO 2, PSO 3	SEE Exams	PO 1, PO 2, PO 3, PSO1, PSO 2, PSO 3	Assignments	PO1, PO2, PO3	Seminars	PO 10, PSO 1
Laboratory Practices	-	Student Viva	-	Mini Project	-	Certification	-
Term Paper	PO 4, PSO2, PSO 3						

XIV. ASSESSMENT METHODOLOGIES –INDIRECT:

✔	Early Semester Feedback	✔	End Semester OBE Feedback
✘	Assessment of Mini Projects by Experts		

XV. SYLLABUS:

UNIT- I	**SOFTWARE PROCESS AND PROJECT MANAGEMENT**
Introduction to software engineering, software process, perspective and specialized process models; Software project management: Estimation: LOC and FP based estimation, COCOMO model; Project scheduling: Scheduling, earned value analysis, risk management	
UNIT-II	**REQUIREMENT ANALYSIS AND SPECIFICATION**
Software requirements: Functional and nonfunctional, user requirements, system requirements, software requirements document; Requirement engineering process: Feasibility studies, requirements elicitation and analysis, requirements validation, requirements management; Classical analysis: Structured system analysis, Petri nets, data dictionary.	
UNIT-III	**SOFTWARE DESIGN**
Design process: Design concepts, design model, design heuristic, architectural design, architectural styles, accessing alternative architectural designs, and architectural mapping using data flow. User interface design: Interface analysis, interface design; Component level design: Designing class based components, traditional components.	
UNIT-IV	**TESTING AND IMPLEMENTATION**
Software testing fundamentals: Internal and external views of testing, white box testing, basis path testing, control structure testing, black box testing, regression testing, unit testing, integration testing, validation testing, system testing and debugging; Software implementation techniques: Coding Practices, refactoring.	
UNIT -V	**PROJECT MANAGEMENT**
Estimation: FP based, LOC based, make/buy decision; COCOMO II: Planning, project plan, planning process, RFP risk management, identification, projection; RMMM: Scheduling and tracking, relationship between people and effort, task set and network, scheduling; EVA: Process and project metrics.	
TEXT BOOKS:	
1. Roger S. Pressman,” Software Engineering – A Practitioner’s Approach”, McGraw-Hill International Edition, 7thEdition,2010. 2. Ian Somerville, “Software Engineering”, Pearson Education Asia, 9thEdition,2011.	

REFERENCES:
1.Rajib Mall, “Fundamentals of Software Engineering”, PHI Learning Private Limited, 3rdEdition, 2009 2.Pankaj Jalote, “Software Engineering, A Precise Approach”, Wiley India, 1stEdition, 2010

XVI. COURSEPLAN:

The course plan is meant as a guideline. Probably there may be changes.

Lecture No	Topics to be covered	Course Learning Outcomes (CLOs)	Reference
1-2	Describe professional software development, Software engineering ethics, Casestudies	CLO 1	T2: 1.1-1.3
2-5	Understand process assessment andimprovement, prescriptive provesmodels,	CLO 2	T1: 2.2-2.3
6-9	Explain generic process model, Specialized process models, the unified process, personal and team process models.	CLO 2	T1:2.1, 2-3-2.6
10-11	Estimate about estimation	CLO 3	R2:2.4
11-12	Describe Introduction about LOC and FP based estimation, COCOMO model	CLO 4	R2:2.5
12-13	Interpret project scheduling and EVA	CLO 2	T1: 27.1, 27.2, 27.6
14	Estimate the Introduction to risk management	CLO 3	T1: 28.1
15-17	Define requirement, Functional and nonfunctional requirements, user requirements, system requirements, Software requirements document.	CLO 8	T2: 4.1-4.3
18-19	Understand Requirements Elicitation, Analysis, validation, Documentation, reviews, Feasibility study and Requirement management.	CLO 8	T1: 4.4-4.7
20-21	Explain structured system analysis, petri nets, data dictionary.	CLO 9	R1:4.5
22-24	Distinguish between design concepts, design model, design heuristic.	CLO 11	T1 8.1-8.4
25-28	State and apply Software Architecture, Architectural design, Architectural styles, Architectural design, and Architectural mapping using dataflow.	CLO 13	T1:9.1, 9.3 9.4,9.6
29-33	Recognize Interface analysis, interface design	CLO 14	T1:11.1,11.3-11.4
34-37	Distinguish between Designing class based components and Traditionalcomponents	CLO 15	T1:10.2, 10.5
38-44	Identify Internal and external view of testing, white box testing, basis path testing, control structure testing, black box testing, regression testing, unit testing, integration testing, validation testing, system testing and debugging.	CLO 17	T1:17.3,17.6-17.8 T1:18.1-18.6
45-47	State and apply Coding practices and refactoring.	CLO 16	T1:20.2
48-51	Contrast Estimation, FP based, LOC based, make/buy decision.	CLO 20	T1: 26.2, 26.6.4, 26.6.6, 26.10
52-55	Understand Planning, project plan, planning process, RFP risk management, identification, projection, RMMM.	CLO 22	T1:26.1-26.3 28.1- 28.7

XVII.GAPS IN THE SYLLABUS - TO MEET INDUSTRY /PROFESSIONREQUIREMENTS:

S No	Description	Proposed actions	Relevance with POs	Relevance with PSOs
1	How to collect usefulrequirements to build right product	Seminars	PO 1, PO 2, PO 3	PSO 1, PSO 2
2	Real time Risk management System	Seminars/ NPTEL	PO 2, PO 3	PSO 1
3	Generation of test cases for usage of ATM machine and Banking Applications	Assignments / NPTEL	PO 1, PO 3	PSO 2

Prepared by:
Mr. C Raghavendra, Assistant Professor

HOD, CSE