INSTITUTE OF AERONAUTICAL ENGINEERING

(Autonomous)
Dundigal, Hyderabad -500 043

INFORMATION TECHNOLOGY

COURSE DESCRIPTOR

Course Title	**SOFTWARE ENGINEERING**				
Course Code	AHS008				
Programme	B.Tech				
Semester	IV	IT			
	III	CSE			
Course Type	Core				
Regulation	IARE - R16				
Course Structure	**Theory**			**Practical**	
	Lectures	**Tutorials**	**Credits**	**Laboratory**	**Credits**
	3	1	4	3	2
Chief Coordinator	Mr. A.Praveen, Assistant Professor,IT				
Course Faculty	Ms. B.Dhana Laxmi, Assistant Professor,IT				

I. COURSE OVERVIEW:

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. Student can implement and get knowledge about development of the software and gains knowledge of basic engineering methods and practices, and their appropriate application. A general understanding of software process models such as the waterfall and evolutionary models. An understanding of the role of project management including planning, scheduling, risk management, etc. An understanding of software requirements and the SRS document and different software architectural styles, implementation issues such as modularity and coding standards. An understanding of approaches to verification and validation including static analysis, and reviews.

II. COURSE PRE-REQUISITES:

Level	Course Code	Semester	Prerequisites
UG	-	-	Basic knowledge of computer hardware and software

III. MARKS DISTRIBUTION:

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

IV. DELIVERY / INSTRUCTIONAL METHODOLOGIES:

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

V. EVALUATION METHODOLOGY:

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).

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 CIE exams.

Quiz / Alternative Assessment Tool (AAT):
Two Quiz exams shall be online examination consisting of 25 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 and MOOCs

VI. HOW PROGRAM OUTCOMES ARE ASSESSED:

Program Outcomes (POs)		Level	Proficiency assessed by
PO 1	**Engineering knowledge**: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.	3	Assignments
PO 2	**Problem analysis**: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.	2	Guest Lectures
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 the cultural, societal, and environmental considerations.	2	Assignments
PO 4	**Conduct investigations of complex problems**: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.	2	Seminars
PO 5	**Modern tool usage**: 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.	2	Assignments

3 = High; 2 = Medium; 1 = Low

VII. HOW PROGRAM SPECIFIC OUTCOMES ARE ASSESSED:

Program Specific Outcomes		Level	Proficiency assessed by
PSO1	**Professional Skills:** The ability to research, understand and implement computer programs in the areas related to algorithms, system software, multimedia, web design, big data analytics, and networking for efficient analysis and design of computer-based systems of varying complexity.	3	Lectures, Assignments
PSO2	**Software Engineering Practices:** The ability to apply standard practices and strategies in software service management using open-ended programming environments with agility to deliver a quality service for business success.	3	Guest Lectures
PSO3	**Successful Career and Entrepreneurship:** The ability to employ modern computer languages, environments, and platforms in creating innovative career paths, to be an entrepreneur, and a zest for higher studies.	2	Seminars

3 = High; 2 = Medium; 1 = Low

VIII. COURSE OBJECTIVES (COs):

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 LEARNING OUTCOMES (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.	PO 1, PO3	3
ACS008.02	CLO 2	Identify the appropriate process models, approaches and techniques to manage a given software development process.	PO4	2
ACS008.03	CLO 3	Identify the approach to risks management through risk identification, risk measurement and risk mitigation.	PO1	3
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 project proceeds.	PO 2	2
ACS008.05	CLO 5	Memorize project planning activities that accurately help in selection and initiation of individual projects and of portfolios of projects in the enterprise.	PO 1	3
ACS008.06	CLO 6	Identify dependability and security issues that affect a given software product.	PO 2	2
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 and translate these into a documented design.	PO 2	2
ACS008.09	CLO 9	Understand the concept of data dictionary in order to manage the details in large-scale systems, to locate errors and omissions in the system.	PO 1	3
ACS008.10	CLO 10	Understand the concept of petri nets that exhibit concurrency, synchronization and used as a visual communication aid to model the system behavior.	PO 2	2
ACS008.11	CLO 11	Memorize the design of object oriented software using with the aid of a formal system modeling	PO 5	2

		notation.		
ACS008.12	CLO 12	Learn to model the structure and behavior of a software system.	PO 2	2
ACS008.13	CLO 13	Memorize different architectural styles, patterns and architectural mapping using data flow.	PO 1	3
ACS008.14	CLO 14	Understand the principles of graphical user interface design.	PO 1	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 2	2
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 of software testing approaches such as unit testing and integration testing.	PO 1	3
ACS008.18	CLO 18	Understand the approaches to verification and validation including static analysis and reviews.	PO 4	2
ACS008.19	CLO 19	Identify the major differences between white box testing and black box testing.	PO 1	3
ACS008.20	CLO 20	Understand the importance of refactoring which improves the performance of non functional attributes of the software.	PO 5	2
ACS008.21	CLO 21	Learn to manage time, processes and resources effectively by prioritizing competing demands to achieve personal and team goals.	PO 1	3
ACS008.22	CLO 22	Use a proactive, structured risk assessment and analysis activity to identify and analyze root. causes.	PO 1	3
ACS008.23	CLO 23	Understand the concept of risk management through risk identification, risk measurement and mitigation.	PO 3	2
ACS008.24	CLO 24	Memorize the relationship between people and effort.	PO 3	2
ACS008.25	CLO 25	Identify the importance of earned value analysis related to project scheduling and also understand the various process and project metric used to improve the quality of software.	PO 4	2

3 = High; 2 = Medium; 1 = Low

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

Course Learning Outcomes (CLOs)	Program Outcomes (POs)												Program Specific Outcomes (PSOs)		
	PO 1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CLO 1	3		2										1		
CLO 2				2										3	
CLO 3	3														
CLO 4		2													
CLO 5	3													2	
CLO 6		2													
CLO 7			3											2	
CLO 8			2												
CLO 9	3														
CLO 10		2													
CLO 11					3										2
CLO 12															
CLO 13	3													3	
CLO 14	3														
CLO 15		2													
CLO 16		2											1		
CLO 17	3														
CLO 18				3											
CLO 19	3														
CLO 20					1										
CLO 21	3													1	
CLO 22	3														
CLO 23													1		3
CLO 24			3												
CLO 25				2									2		

3 = High; 2 = Medium; 1 = Low

XI. ASSESSMENT METHODOLOGIES – DIRECT

CIE Exams	PO1,PO2, PO3,PO4, PO5	SEE Exams	PO1,PO2, PO3,PO4, PO5	Assignments	PO2	Seminars	PO3
Laboratory Practices	PO1	Student Viva	-	Mini Project	-	Certification	PO2
Term Paper	-						

XII. ASSESSMENT METHODOLOGIES - INDIRECT

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

XIII. SYLLABUS

Unit-I	**SOFTWARE PROCESS AND PROJECT MANAGEMENT**
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**
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-II	**SOFTWARE DESIGN**
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-V	**TESTING AND IMPLEMENTATION**
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**
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, 7th Edition, 2010. 2. Ian Somerville, "Software Engineering", Pearson Education Asia, 9th Edition, 2011.	

Reference Books:
1. Rajib Mall, "Fundamentals of Software Engineering", PHI Learning Private Limited, 3rd Edition, 2009. 2. Pankaj Jalote, "Software Engineering, A Precise Approach", Wiley India, 1st Edition, 2010.

XIV. COURSE PLAN:

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	Introduction to Software Engineering	CLO1	T2: 1.1-1.3
2-5	Software processes	CLO2	T1: 2.2-2.3
6-9	Process models	CLO4	T1: 2.1, 2-3-2.6
11-12	Software Project Management	CLO3	R2: 3.4-3.9
11-12	LOC and FP based estimation COCOMO model	CLO5	R2: 4.1-4.3
12-13	Project Scheduling, EVA	CLO8	T1: 27.1, T1:27.2, 27.6
14	Risk management	CLO7	T1: 28.1
15-17	Software Requirements	CLO9	T2: 4.1-4.3
18-19	Requirements Engineering process	CLO6,CLO10	T1: 4.4-4.7
20-21	Classical Analysis	CLO11	R1: 1.1-1.4
22-24	Design process	CLO12	T1 8.1-8.4
25-28	Architectural design	CLO13, CLO14	T1:9.1, 9.3,9.4,9.6
29-33	User interface design	CLO15	T1:11.1,11.3-11.4
34-37	Component level design	CLO16	T1:10.2, 10.5
38-44	Software Testing fundamentals	CLO18	T1:17.3,17.6-17.8 T1:18.1-18.6
45-47	Software implementation techniques	CLO17	T1:10.1-1.3
48-51	Project management	CLO19	T1: 26.2, 26.6.4, T2:26.6.6, 26.10
52-55	COCOMO II	CLO21,CLO22	T1:26.1-26.3 28.1- 28.7
56-60	Project Scheduling, Project Metrics	CLO23,CLO24	T1:25.1-25.6 T1:27.1-27.6

XV. GAPS IN THE SYLLABUS - TO MEET INDUSTRY / PROFESSION REQUIREMENTS:

S No	Description	Proposed Actions	Relevance With POs	Relevance With PSOs
1	How to collect useful requirements to build right product	Seminars/Guest Lectures / NPTEL	PO 1, PO 2, PO 3	PSO 1, PSO 2
2	Real time Risk management System	Seminars/Guest Lectures / NPTEL	PO 2, PO 3	PSO 1
3	Generation of test cases for usage of ATM machine and Banking Applications	Assignments/ Laboratory Practices	PO 1, PO 3, PO 4	PSO 2

Prepared by:
Ms. B.Dhana Laxmi, Assistant Professor,IT
Mr.A.Praveen, Assistant Professor,IT

HOD, IT