INSTITUTE OF AERONAUTICAL ENGINEERING
(Autonomous)
Dundigal - 500 043, Hyderabad, Telangana

COURSE CONTENT

SOFTWARE ENGINEERING AND DEVELOPMENT PRACTICES								
VII Semester: CSE | CSE (DS) | CSE(CS) | IT								
Course Code	**Category**	**Hours / Week**			**Credits**	**Maximum Marks**		
ACSE35	**Open Elective**	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		3	0	0	3	40	60	100
Contact Classes: 48	**Tutorial Classes: Nil**	**Practical Classes: Nil**				**Total Classes: 48**		
Prerequisites: Basic understanding of operating systems, computer networks, and Network Security and Cryptography".								

I. COURSE OVERVIEW:

This course provides a comprehensive understanding of software engineering principles, methodologies, and best development practices used in building high-quality software systems. It focuses on the complete software development life cycle (SDLC), including requirements analysis, system design, implementation, testing, deployment, and maintenance. The course emphasizes both traditional and agile development models, version control, coding standards, software quality assurance, documentation, and collaborative development practices. Through practical case studies and hands-on activities, learners gain the ability to design, develop, test, and maintain scalable, reliable, and maintainable software solutions aligned with industry standards and professional ethics.

II. COURSE OBJECTIVES:

The students will try to learn:

I. Learn how to felicitate requirements and develop software life cycles.
II. Understand the design considerations for enterprise integration and deployment.
III. Analyze testing methodologies.
IV. Prepare a project plan for a software project that includes estimates of size and effort, a schedule, resource allocation, configuration control, and project risk.

III. COURSE OUTCOMES:

CO1 Demonstrate knowledge of tools, techniques, and resources available to support ethical hacking activities.
CO2 Analyze and interpret the results of controlled attacks to assess system vulnerabilities effectively
CO3 Understand the influence of organizational politics, inherent and imposed limitations, and metrics in planning and executing penetration tests.
CO4 Apply structured frameworks for planning and executing penetration tests, including reconnaissance, enumeration, vulnerability analysis, exploitation, and reporting.
CO5 Evaluate the role of organizational policies, limitations, and metrics in planning and conducting ethical hacking exercises.
CO6 Recognize and mitigate the risks, ethical concerns, and legal implications associated with penetration testing.

IV. COURSE CONTENT:

MODULE-1:SOFTWARE PROCESS AND PROJECT MANAGEMENT(08)

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

MODULE-2:REQUIREMENTS ANALYSIS AND SPECIFICATION(09)

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, data dictionary.

MODULE-3:SOFTWARE DESIGN(09)

Design process: Design concepts, design mode, design heuristic, architectural design architectural styles, architectural design, and architectural mapping using data flow.

User interface design: Interface analysis, interface design; Component level design: Designing class based components, traditional components.

MODULE 4:IMPLEMENTATION AND TESTING(10)

Software testing fundamentals: Internal and external views of testing, white box testing, basis path testing, control structure testing, black box testing, regression testing, MODULE testing, integration testing, validation testing, system testing and debugging; Software implementation techniques: Coding practices, refactoring.

MODULE 5:PROJECT MANAGEMENT(09)

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.

V. TEXTBOOKS:

1. Roger S. Pressman, ―Software Engineering – A Practitioner‘s Approach‖, Tata McGraw-Hill International Edition, 7th Edition, 2010.
2. Ian Somerville, ―Software Engineering‖, Pearson Education Asia, 9th Edition, 2011. Defense", Cengage Learning.

VI. 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.

VII. WEB REFERENCES

1. http://www.softwareengineerinsider.com/articles/what-is-software-engineering.html
2. https://www.udacity.com/courses/software-engineering
3. http://www.tutorialspoint.com/software_engineering
4. http://computingcareers.acm.org/?page_id=12
5. http://en.wikibooks.org/wiki/Introduction_to_Software_Engineering).

VIII. MATERIALS ONLINE

1. Course template
2. Tutorial question bank
3. Tech talk topics
4. Open-ended experiments
5. Definitions and terminology
6. Assignments
7. Model question paper – I
8. Model question paper – II
9. Lecture notes
10. PowerPoint presentation
11. E-Learning Readiness Videos (ELRV)