CAUVERY COLLEGE FOR WOMEN (AUTONOMOUS)

NATIONALLY ACCREDITED WITH A^+ GRADE BY NAAC

TIRUCHIRAPPALLI

PG DEPARTMENT OF INFORMATION TECHNOLOGY

M. Sc. INFORMATION TECHNOLOGY

SYLLABUS

2025-2026

CAUVERY COLLEGE FOR WOMEN (AUTONOMOUS)

PG DEPARTMENT OF INFORMATION TECHNOLOGY

Vision

The Department of Information Technology envisions to create technically competent, skilled intellectual IT professionals, efficient problem solvers, innovators and entrepreneurs to meet the current challenges of the modern computing industry.

Mission

➢ To provide quality education and elevate the students towards higher educational programs

➢ To encourage and guide the students to improve their competency skills in information technology market

➢ To equip the students to cater the industrial demands through providing advance training

PROGRAMME EDUCATIONAL OBJECTIVES (PEOs)

PEOs	Statements
PEO1	**LEARNING ENVIRONMENT** To facilitate value-based holistic and comprehensive learning by integrating innovative learning practices to match the highest quality standards and train the students to be effective leaders in their chosen fields.
PEO2	**ACADEMIC EXCELLENCE** To provide a conducive environment to unleash their hidden talents and to nurture the spirit of critical thinking and encourage them to achieve their goal.
PEO3	**EMPLOYABILITY** To equip students with the required skills in order to adapt to the changing global scenario and gain access to versatile career opportunities in multidisciplinary domains.
PEO4	**PROFESSIONAL ETHICS AND SOCIAL RESPONSIBILITY** To develop a sense of social responsibility by formulating ethics and equity to transform students into committed professionals with a strong attitude towards the development of the nation.
PEO5	**GREEN SUSTAINABILITY** To understand the impact of professional solutions in societal and environmental contexts and demonstrate the knowledge for an overall sustainable development.

PROGRAMME OUTCOMES FOR M.Sc. INFORMATION TECHNOLOGY PROGRAMME

PO NO.	Programme Outcome On completion of M.Sc. Information Technology The students will be able to
PO1	**DOMAIN KNOWLEDGE** Acquire the in-depth computing knowledge both conceptual and applied pertaining to the core discipline
PO2	**PROBLEM SOLVING** Procure knowledge-based skills to satisfy the needs of society and the industry by providing hands on experience of various technologies in Computer Science
PO3	**INNOVATION AND CRITICAL THINKING** Critically evaluate global issues, recognize the need and identify sustainable solutions through research capabilities towards Nation building initiatives
PO4	**LIFE LONG LEARNING** Capable of upgrading and advancing knowledge through innovation and technology as evidenced by current developments
PO5	**LEADERSHIP AND TEAMWORK** Work in collaborative environment through applications of scientific reasoning and communicate effectively to the stakeholders

Annexure – I

PROGRAMME SPECIFIC OUTCOMES FOR M.Sc. INFORMATION TECHNOLOGY PROGRAMME

PSO No.	Programme Specific Outcomes Students of M.Sc. Information Technology will be able to	POs Addressed
PSO1	Acquiring in-depth knowledge of computing fundamentals and advanced topics in various information technology domains.	PO1
PSO2	Pursuing lifelong learning and research as computing professionals to meet industry standards and corporate demands by providing innovative solutions.	PO1 PO2
PSO3	Apply theoretical foundations and practical expertise in Information Technology to develop effective, efficient, and innovative real-time solutions that contribute to societal impact.	PO2 PO3
PSO4	Ability to adapt to rapidly evolving IT technologies and engage in continuous learning to stay current with industry trends and challenges.	PO3 PO4
PSO5	Demonstrate ability to collaborate, apply scientific principles, and communicate complex solutions to stakeholders	PO4 PO5

Cauvery College for Women (Autonomous), Trichy

PG Department of Information Technology

M.Sc. Information Technology

LEARNING OUTCOME BASED CURRICULUM FRAMEWORK (CBCS- LOCF)

(For the Candidates admitted from the Academic year 2025-2026 and onwards)

Semester	Course	Course Title	Course Code	Inst. Hrs. / Week	Credits	Exam			Total
						Hrs.	Marks		
							Int.	Ext.	
I	Core Course– I (CC)	J2EE Technologies	25PIT1CC1	6	5	3	25	75	100
	Core Course – II (CC)	Object Oriented Analysis and Design	25PIT1CC2	6	5	3	25	75	100
	Core Course –III (CC)	Mathematical Foundation for Information Technology	25PIT1CC3	6	5	3	25	75	100
	Core Practical - I (CP)	J2EE(P)	25PIT1CC1P	6	5	3	40	60	100
	Discipline Specific Elective Course-I (DSE)	A. Software Testing	25PIT1DSE1A	6	3	3	25	75	100
		B. Software Project Management	25PIT1DSE1B						
		C. Agile software Process	25PIT1DSE1C						
	Total			**30**	**23**	-	-	-	**500**
15 Days INTERNSHIP during Semester Holidays									
II	Core Course– IV(CC)	Advanced Database Management Systems	25PIT2CC4	6	5	3	25	75	100
	Core Course– V(CC)	Wireless Sensor Network	25PIT2CC5	6	5	3	25	75	100
	Core Choice Course–I (CCC)	A. Internet of Things	25PIT2CCC1A	6	4	3	25	75	100
		B. Open-Source Technologies	25PIT2CCC1B						
		C. Cloud Computing	25PIT2CCC1C						
	Core Practical-II(CP)	Advanced Database Management Systems (P)	25PIT2CC2P	6	5	3	40	60	100
	Discipline Specific Elective Course-II (DSE)	A. Internet of Things (P)	25PIT2DSE2AP	6	3	3	40	60	100
		B. Open-Source Technologies(P)	25PIT2DSE2BP						
		C. Cloud Computing (P)	25PIT2DSE2CP						
	Internship	Internship	25PIT2INT	-	2	-	25	75	100
	Extra Credit Course	SWAYAM	As per UGC Recommendation						
	Total			**30**	**24**	-	-	-	**600**

Semeser	Course	Course Title	Course Code	Inst. Hrs. / week	Credits	Exam			Total
						Hrs	Marks		
							Int.	Ext.	
III	Core Course– VI (CC)	Natural Language Processing	25PIT3CC6	6	5	3	25	75	100
	Core Course – VII (CC)	Data Mining Techniques and Tools	25PIT3CC7	6	5	3	25	75	100
	Core Choice Course– II (CCC)	A. Cyber Security	22PGCS3CCC2A	5	4	3	25	75	100
		B. Mobile Application Development using Android	25PIT3CCC2B						
		C. Parallel Computing	25PIT3CCC2C						
	Core Practical - III (CP)	Data Mining Techniques and Tools Using R (P)	25PIT3CC3P	5	4	3	40	60	100
	Discipline Specific Elective Course-III (DSE)	A. Computer Science for Competitive Examinations	22PCS3DSE3A	5	3	2	-	100	100
		B. Android Programming(P)	25PIT3DSE3BP			3	40	60	
		C. Parallel Computing (P)	25PIT3DSE3CP						
	Generic Elective Course -I (GEC)	Office Management Tools(P)	25PIT3GEC1P	3	2	3	40	60	100
	Extra Credit Course	SWAYAM	As per UGC Recommendation						
	Total			**30**	**23**	-	-	-	**600**
IV	Core Course–VIII (CC)	AI and Machine Learning	25PIT4CC8	6	5	3	25	75	100
	Core Choice Course– III (CCC)	A. Data Analysis using Python	25PIT4CCC3A	6	4	3	25	75	100
		B. Big Data Analytics	25PIT4CCC3B						
		C Neural Networks	25PIT4CCC3C						
	Core Practical - IV (CP)	AI and Machine Learning using Python(P)	25PIT4CC4P	6	5	3	40	60	100
	Generic Elective Course-II (GEC)	Content Development Tools (P)	25PIT4GEC2P	3	2	3	40	60	100
	Project	Project Work	25PIT4PW	9	4	-	-	100	100
	Total			**30**	**20**				**500**
	Grand Total			**120**	**90**				**2200**

The Internal and External marks for theory and practical courses are as follows:

Course	Internal Marks	External Marks
Theory	25	75
Practical	40	60
Project	-	100
Internship	25	75

For Theory courses:

a) The passing minimum for CIA shall be 40% out of 25 marks (i.e. 10 marks)

b) The passing minimum for End Semester Examinations shall be 40% out of 75 marks (i.e. 30 marks)

c) The passing minimum not less than 50% in the aggregate

For Practical courses:

a) The passing minimum for CIA shall be 40% out of 40 marks (i.e. 16 marks)

b) The passing minimum for End Semester Examinations shall be 40% out of 60 marks (i.e. 24 marks)

c) The passing minimum not less than 50% in the aggregate

For Project Work:

a) The passing minimum not less than 50% out of 100 marks

d) For Internship:

a) The passing minimum not less than 50% in the aggregate

Semester I	Internal Mark: 25		External Mark: 75	
COURSE CODE	**COURSE TITLE**	**CATEGORY**	**Hrs/Week**	**CREDITS**
25PIT1CC1	**J2EE Technologies**	**Core Course - I**	**6**	**5**

Course Objectives

- To explore key features and core concepts of J2EE
- To provide a comprehensive understanding of JavaServer Pages, Servlets and Entity Java Beans
- To impart knowledge on database connectivity and operations

Course Outcomes and Cognitive Level Mapping

On the successful completion of the course, students will be able to

CO Number	Course Outcome	Cognitive Level
CO1	Identify and understand the key concepts of J2EE, its multi-tier architecture, and foundational technologies like HTML, XML, and XHTML, along with generating and parsing XML documents	K1, K2
CO2	Describe and apply JSP concepts, including elements, implicit objects, error handling, Expression Language, and JSTL	K2
CO3	Apply servlet concepts by designing HTTP server programs, managing sessions, handling exceptions, and implementing the MVC architecture	K3
CO4	Develop different types of EJBs and analyze their use with JSP, Servlets, and design patterns in enterprise applications	K3, K4
CO5	Analyze database operations and evaluate transaction handling and concurrency control	K4, K5

Mapping of CO with PO and PSO

	PSO1	PSO2	PSO3	PSO4	PSO5	PO1	PO2	PO3	PO4	PO5
CO1	2	3	3	1	2	2	2	3	3	3
CO2	3	2	3	1	2	3	3	2	3	2
CO3	3	3	3	2	2	3	3	2	2	3
CO4	3	2	3	2	3	2	3	3	3	2
CO5	3	3	2	3	2	3	3	2	2	3

"1" – Slight (Low) Correlation **"2" – Moderate (Medium) Correlation**
"3" – Substantial (High) Correlation **"-" indicates there is no correlation**

SYLLABUS

UNIT	CONTENTS	HOURS	COs	COGNITIVE LEVEL
I	**J2EE ESSENTIALS:** Introduction to J2EE – Need for J2EE – J2EE Multi-Tier Architecture – Features and Concepts in J2EE – J2EE Foundation: HTML, XML and XHTML – Generating an XML document – Parsing XML	18	CO1, CO2, CO3, CO4, CO5	K1, K2, K3, K4, K5
II	**JAVASERVER PAGES:** Introduction to JSP – JSP Elements – Implicit Objects – Scope – Errors and Exceptions – Including and Forwarding from JSP Pages – Expression Language – Custom Actions and Tag Libraries – JavaServer Pages Standard Tag Library (JSTL) – Other Tag Libraries	18	CO1, CO2, CO3, CO4, CO5	K1, K2, K3, K4, K5
III	**SERVLETS:** HTTP and Server Programs – Basic Servlet Design – Request and Response Objects – Deployment Descriptors – Servlet Life Cycle – Handling Exceptions – Session Management – Filters – MVC Architecture	18	CO1, CO2, CO3, CO4, CO5	K1, K2, K3, K4, K5
IV	**EJB FUNDAMENTALS:** Introduction – Three kinds of EJBs – Anatomy of a Session Bean – Stateful Vs. Stateless Session Beans – Anatomy of an Entity Bean – Features of Entity Beans – Developing CMP and BMP Entity Beans – Developing Message-Driven Beans – Using JSP, Servlets and Design Patterns in EJB Applications	18	CO1, CO2, CO3, CO4, CO5	K1, K2, K3, K4, K5
V	**WORKING WITH DATABASES:** Connecting to Databases – Statements – Resultsets – Prepared Statements – Callable Statements – Data Sources and Connection Pools – Transactions – Locking and Isolation	18	CO1, CO2, CO3, CO4, CO5	K1, K2, K3, K4, K5
VI	**Self Study for Enrichment:** **(Not included for End Semester Examination)** Web Services – JavaMail – Remote Method Invocation – Java Message Services – J2EE Management and Deployment Specifications	-	CO1, CO2, CO3, CO4, CO5	K1, K2, K3, K4, K5

TEXT BOOKS

1. James L. Weaver, Kevin Mukhar and James Crume, "**Beginning J2EE 1.4: From Novice to Professional",** 1st Edition, APress Media LLC, 2013
2. Jim Keogh, "**The Complete Reference: J2EE**", Tata McGraw Hill Edition, 2017

REFERENCE BOOKS

1. Eric Armstrong, "The J2EE 1.4 Tutorial ", 2005
2. Martin Bond, "Teach Yourself J2EE", 2002

WEB REFERENCES

1. **https://www.java4coding.com/contents/servlet/jee-tutorial**
2. **https://mindmajix.com/j2ee-tutorial**
3. **https://www.digitalocean.com/community/tutorials/java-tutorial-java-ee-tutorials**

PEDAGOGY

Chalk and Talk, PPT, Discussion, Assignment, Demo, Quiz and Seminar.

COURSE DESIGNER

Dr. M. Thangam, Associate Professor, Dept. of Information Technology

Semester I	Internal Mark: 25		External Mark: 75	
COURSE CODE	**COURSE TITLE**	**CATEGORY**	**Hrs/Week**	**CREDITS**
25PIT1CC2	Object Oriented Analysis and Design	Core Course – II (CC)	**6**	**5**

Objectives:

- To understand the fundamentals of object analysis and design.
- To understand and differentiate Unified Process from other approaches.
- To design with static UML diagrams and identify the concept.
- To design with the UML dynamic and implementation diagrams.
- To design the software with appropriate design patterns.
- To test the software against its requirements specification techniques of conceptual UML modeling.

Course Outcomes and Cognitive Level Mapping

On the successful completion of the course, students will be able to

CO Number	Course Outcome	Cognitive Level
CO1	To understand the interface between the classes and objects.	K1,K2
CO2	Build a model for the user interface (UI) for a software application	K2,K3
CO3	Measure the Level of User satisfaction and software quality assurance.	K3,K4
CO4	Gain exposure from design to Implementation	K5
CO5	Create class diagrams that model both the domain and design model of a software system.	K5

Mapping of CO with PO and PSO

	PSO1	PSO2	PSO3	PSO4	PSO5	PO1	PO2	PO3	PO4	PO5
CO1	3	3	3	2	2	2	2	3	3	3
CO2	3	2	3	2	2	3	3	2	3	2
CO3	3	3	3	2	2	3	3	2	3	3
CO4	3	2	3	2	3	2	3	3	3	3
CO5	3	3	3	3	3	3	3	2	2	3

"1" – Slight (Low) Correlation "2" – Moderate (Medium) Correlation
"3" – Substantial (High) Correlation "-" indicates there is no correlation.

Syllabus:

UNIT	CONTENT	HOURS	COs	COGNITIVE LEVEL
I	An overview - Object basics - Object state and properties - Behavior - Methods - Messages – Information hiding - Class hierarchy - Relationships - Associations - Aggregations- Identity - Dynamic binding - Persistence - Metaclasses - Object oriented system development life cycle.	18	CO1, CO2, CO3, CO4, CO5	K1, K2, K3, K4, K5
II	Introduction - Rumbaugh, Booch, Jacobson methods - Patterns - Frameworks - Unified approach - Unified modeling language - Static and Dynamic models - UML diagrams - Class diagram - Usecase diagrams - Dynamic modeling - Model organization – Extensibility	18	CO1, CO2, CO3, CO4, CO5	K1, K2, K3, K4, K5
III	Identifying Usecase - Business object analysis - Usecase driven object oriented analysis - Usecase model - Documentation - Classification - Identifying object, relationships, attributes, methods - Super-sub class - A part of relationships Identifying attributes and methods - Object responsibility	18	CO1, CO2, CO3, CO4, CO5	K1, K2, K3, K4, K5
IV	Design process - Axioms - Designing classes - Class visibility - Refining attributes - Methods and protocols- Object storage and object interoperability - Databases - Object relational systems - Designing interface objects - The purpose of a view layer interface	18	CO1, CO2, CO3, CO4, CO5	K1, K2, K3, K4, K5
V	Quality assurance - Testing strategies - Object orientation testing - Test cases - Test Plan – Debugging principles - Usability - Satisfaction - Usability testing - Satisfaction testing	18	CO1, CO2, CO3, CO4, CO5	K1, K2, K3, K4, K5
VI	**Self Study for Enrichment:** **(Not included for End Semester Examination)** UML - Examples on: Behavioural models – Structural models – Architectural models from real world problems. **Case Study**-Student Admission System-Banking Systems	-	CO1, CO2, CO3, CO4, CO5	K1, K2, K3, K4, K5

TEXT BOOKS

1. Bahrami Ali, Object oriented systems development, Irwin McGrawHill, 2005 (First 4 units covered here).
2. Booch Grady, Rumbaugh James, Jacobson Ivar, The Unified modeling language – User Guide, Pearson education, 2006 (ISBN 81-7758-372-7)

REFERENCES BOOKS:

1. Grady Booch, Michael W. Engel, Kelli A. Houston, Robert A. Maksimchuk, Bobbi J. Young, Jim Conallen, "Object-Oriented Analysis and Design with Applications", 3rd Edition, Pearson Education, 2009
2. Michael Blaha and James Rumbaugh, "Object-Oriented Modeling and Design with UML", 2nd Edition, Pearson Education,2005
3. Erich Gamma, Richard Helm, Ralph Johnson & John Vlissides, "Design Patterns:
4. Elements of Reusable Object-oriented Software", Pearson Education India, 2004.

WEB REFERENCES

https://www.tutorialspoint.com/object_oriented_analysis_design/ooad_tutorial.pdf
https://www.scribd.com/doc/74801730/Case-Study-on-Traffic-Management-System

Course Designer

Dr.S.Suguna Devi Dept. of Information Technology

Semester I	Internal Mark: 40		External Mark: 60	
COURSE CODE	**COURSE TITLE**	**CATEGORY**	**Hrs/Week**	**CREDITS**
25PIT1CC1P	**J2EE (P)**	**Core Course Practical - I**	**6**	**3**

COURSE OBJECTIVES

- To deploy and manage J2EE applications
- To provide a comprehensive understanding of JavaServer Pages, Servlets and Entity Java Beans
- To impart knowledge on database connectivity and operations

Course Outcomes and Cognitive Level Mapping

On the successful completion of the course, students will be able to

CO Number	Course Outcome	Cognitive Level
CO1	Understand the architecture and components of J2EE applications	K1
CO2	Develop web-based applications using Servlets and JSP	K2
CO3	Demonstrate session management techniques	K3
CO4	Design and deploy enterprise-level applications using Enterprise JavaBeans	K4
CO5	Implement database connectivity using JDBC	K5

Mapping of CO with PO and PSO

	PSO1	PSO2	PSO3	PSO4	PSO5	PO1	PO2	PO3	PO4	PO5
CO1	3	2	2	2	2	3	1	2	2	1
CO2	3	3	2	3	3	2	3	2	3	2
CO3	3	2	3	3	3	3	3	3	2	3
CO4	3	3	3	2	3	2	3	3	3	3
CO5	3	3	3	3	3	3	3	2	3	3

"1" – Slight (Low) Correlation **"2" – Moderate (Medium) Correlation**
"3" – Substantial (High) Correlation **"-" indicates there is no correlation**

LAB EXERCISES

1. Create a basic HTML webpage with headings, paragraphs, and a link.
2. Write a servlet to accept username input from a form and display a greeting message.
3. Demonstrate the servlet life cycle by printing messages from `init()`, `service()`, and `destroy()` methods.
4. Implement basic session tracking using HttpSession
5. Develop a simple JSP page displaying static and dynamic content.
6. Demonstrate usage of JSP elements: Declarations, Scriplets, Expressions.
7. Create a JSP page using Expression Language (EL) to display data.
8. Create a simple Stateless Session Bean that returns a welcome message.
9. Create a simple Stateful Session Bean that stores and retrieves a student's name.
10. Write a simple JDBC program to connect to a MySQL database.
    - Insert a few student records into a database using a Java program and a PreparedStatement.
    - Retrieve and display all student names from a database table.
    - Demonstrate how to perform commit and rollback in JDBC transactions.

COURSE DESIGNER

Dr. M. Thangam, Associate Professor, Dept. of Information Technology

Semester I	Internal Mark: 25		External Mark: 75	
COURSE CODE	**COURSE TITLE**	**CATEGORY**	**Hrs/Week**	**CREDITS**
25PIT1DSE1A	**Software Testing**	**Discipline Specific Elective Course – I (DSE)**	**6**	**3**

Objectives:

To study the fundamental concepts of software testing which includes objectives, process, criteria, strategies, and methods.

To discuss various software testing types and levels of testing like black and white box testing along with levels unit test, integration, regression, and system testing.

It also helps to learn the types of bugs, testing levels with which the student can very well identify a bug and correct as when it happens.

To learn the domain testing, path testing and logic based testing to explore the testing process easier.

Course Outcomes and Cognitive Level Mapping

On the successful completion of the course, students will be able to

CO Number	Course Outcome	Cognitive Level
CO1	Know the basic concepts of software testing and its essentials.	K1
CO2	Able to identify the various bugs and correcting them after knowing the consequences of the bug.	K2
CO3	Use of program's control flow as a structural model is the corner stone of testing.	K3
CO4	Performing functional testing using control flow and transaction flow graphs.	K4
CO5	provides knowledge on transaction flow testing and data flow testing techniques so that the flow of the program is tested as well.	K5

Mapping of CO with PO and PSO

	PSO1	PSO2	PSO3	PSO4	PSO5	PO1	PO2	PO3	PO4	PO5
CO1	3	2	3	2	1	2	2	3	3	3
CO2	3	2	3	2	2	3	3	2	3	2
CO3	3	3	3	2	2	3	3	2	3	3
CO4	3	2	3	2	3	2	3	3	3	3
CO5	3	3	3	3	3	3	3	2	2	3

"1" – Slight (Low) Correlation "2" – Moderate (Medium) Correlation
"3" – Substantial (High) Correlation "-" indicates there is no correlation.

Syllabus

UNIT	CONTENT	HOURS	COs	COGNITIVE LEVEL
I	Introduction:-Purpose of testing, Dichotomies, model for testing, consequences of bugs, taxonomy of bugs, Flow graphs and Path testing:- Basics concepts of path testing, predicates, path predicates and achievable paths, path sensitizing, path instrumentation, application of path testing.	18	CO1, CO2, CO3, CO4, CO5	K1, K2, K3, K4, K5
II	Transaction Flow Testing:-transaction flows, transaction flow testing techniques. Dataflow testing:- Basics of dataflow testing, strategies in dataflow testing,application of dataflow testing.	18	CO1, CO2, CO3, CO4, CO5	K1, K2, K3, K4, K5
III	Domain Testing:-domains and paths, Nice & ugly domains, domain testing, domains and interfaces testing, domain and interface testing, domains and testability	18	CO1, CO2, CO3, CO4, CO5	K1, K2, K3, K4, K5
IV	Paths,Path products and Regular expressions:- path products &path expression, reduction procedure, applications, regular expressions & flow anomaly detection. Logic Based Testing:-overview,decision tables,path expressions,kev charts, specifications	18	CO1, CO2, CO3, CO4, CO5	K1, K2, K3, K4, K5
V	State, State Graphs and Transition testing:- state graphs, good & bad state graphs, state testing, Testability tips. Graph Matrices and Application:- Motivational overview, matrix of graph, relations, power of a matrix, node reduction algorithm, building tools	18	CO1, CO2, CO3, CO4, CO5	K1, K2, K3, K4, K5
VI	**Self Study for Enrichment:** **(Not included for End Semester Examination)** Implementation:Synopsis- Overview-Strategies for Programmers-Strategies for Independent Teste-r-Tests as software Products-Tools:-Test Execution Automation Test Design Automation-Tools Usage and Training	-	CO1, CO2, CO3, CO4, CO5	K1, K2, K3, K4, K5

TEXT BOOKS

1. Software Testing techniques – Boris Beizer, Dreamtech, second edition.
2. Software Testing Tools – Dr.K.V.K.K.Prasad, Dreamtech.

REFERENCES BOOKS:

1. The craft of software testing – Brian Marick, Pearson Education.
2. Software Testing Techniques – SPD(Oreille)
3. Software Testing in the Real World – Edward Kit, Pearson.
4. Effective methods of Software Testing, Perry, John Wiley.
5. Art of Software Testing – Meyers, John Wiley.

Pedagogy

Chalk and Talk, PPT, Discussion, Assignment, Demo, Quiz and Seminar.

Course Designer

Dr.J. Sangeetha, Dept. of Information Technology

Semester I	Internal Mark: 25		External Mark: 75	
COURSE CODE	**COURSE TITLE**	**CATEGORY**	**Hrs/Week**	**CREDITS**
25PIT1DSE1B	**Software Project Management**	**Discipline Specific Elective Course – I (DSE)**	**6**	**3**

Objectives:

- To introduce the primary important concepts of project management related to managing software development projects.
- To get familiar with the different activities involved in Software Project Management.
- To know how to successfully plan and implement a software project management activity, and to complete a specific project in time with the available budget

Course Outcomes and Cognitive Level Mapping

On the successful completion of the course, students will be able to

CO Number	Course Outcome	Cognitive Level
CO1	Understand the fundamental principles of software project management.	K1, K2
CO2	Apply the principles of modern software management and life cycle phases.	K3
CO3	Examine the work flows of the software process.	K4
CO4	Explain the responsibilities of project manager and project organizations	K5
CO5	Develop small scale and large-scale software products.	K5

Mapping of CO with PO and PSO

	PSO1	PSO2	PSO3	PSO4	PSO5	PO1	PO2	PO3	PO4	PO5
CO1	3	3	3	2	2	2	2	3	3	3
CO2	3	2	3	2	2	3	3	2	3	3
CO3	3	3	3	2	2	3	3	2	2	3
CO4	3	2	3	2	3	2	3	3	3	2
CO5	3	3	2	3	2	3	3	2	2	2

"1" – Slight (Low) Correlation **"2" – Moderate (Medium) Correlation**
"3" – Substantial (High) Correlation **"-" indicates there is no correlation.**

Syllabus

UNIT	CONTENT	HOURS	COs	COGNITIVE LEVEL
I	SOFTWARE MANAGEMENT & ECONOMICS: The Waterfall Model, Conventional Software Management Performance; Evolution of Software Economics - Software economics, Pragmatic software cost estimation. Improving Software Economics: Reducing software product size, improving software processes, Improving Team Effectiveness, improving automation.	18	CO1, CO2, CO3, CO4, CO5	K1, K2, K3, K4, K5
II	THE OLD AND THE NEW WAY OF PROJECT MANAGEMENT: The principles of conventional software Engineering, principles of modern software management, transitioning to an iterative process. Life cycle phases: Engineering and production stages, inception, Elaboration, construction, transition phases. Artifacts of the process: The artifact sets, Management artifacts, Engineering artifacts, programmatic artifacts. Model based software architectures : A Management perspective and technical perspective.	18	CO1, CO2, CO3, CO4, CO5	K1, K2, K3, K4, K5
III	WORK FLOWS OF THE PROCESS: Software process workflows, Iteration workflows. Checkpoints of the process: Major mile stones, Minor Milestones, Periodic status assessments. Iterative Process Planning: Work breakdown structures, planning guidelines, cost and schedule estimating, Iteration planning process, Pragmatic planning.	18	CO1, CO2, CO3, CO4, CO5	K1, K2, K3, K4, K5
IV	PROJECT ORGANIZATIONS AND RESPONSIBILITIES: Line-of-Business Organizations, Project Organizations, evolution of Organizations. Process Automation: Automation Building blocks, The Project Environment. Project Control and Process instrumentation: The seven core Metrics, Management indicators, quality indicators, life cycle expectations, pragmatic Software Metrics, Metrics automation.	18	CO1, CO2, CO3, CO4, CO5	K1, K2, K3, K4, K5
V	Tailoring the Process: Process discriminants, Small-scale project Vs Large scale project. Looking Forward: Modern Project Profiles, Next generation Software economics.	18	CO1, CO2, CO3, CO4, CO5	K1, K2, K3, K4, K5
VI	**Self Study for Enrichment:** **(Not included for End Semester Examination)** The COCOMO cost estimation model, Change metrics, CCPDS-R – Core Metrics, People factors.	-	CO1, CO2, CO3, CO4, CO5	K1, K2, K3, K4, K5

TEXT BOOKS

1. Walker Royce, “Software Project Management”, 1st Edition, Pearson Education, 2006.

REFERENCES BOOKS:

1. Bob Hughes and Mike Cotterell, “Software Project Management”, 3rd Edition, Tata McGraw Hill Edition, 2005.

2. Joel Henry, “Software Project Management”, 1st Edition, Pearson Education, 2006.

3. PankajJalote, “Software Project Management in practice”, 1st Edition, Pearson Education, 2005.

WEB REFERENCE:

https://www.tutorialspoint.com/software_engineering/software_project_management.html

https://onlinecourses.nptel.ac.in/noc19_cs70/preview

Pedagogy

Chalk and Talk, PPT, Discussion, Assignment, Demo, Quiz and Seminar.

Course Designer

Dr.A. Bhuvaneswari, Dept. of Information Technology

Semester I	Internal Mark: 25		External Mark: 75	
COURSE CODE	**COURSE TITLE**	**CATEGORY**	**Hrs/Week**	**CREDITS**
25PIT1DSE1C	**Agile Software Process**	**Discipline Specific Elective Course – I (DSE)**	**6**	**3**

Course Objective:

1. To familiarize the students with the features of agile development.
2. To familiarize the students with basic Agile Estimation & Planning.
3. To facilitate students in understanding the Tracking Agile Projects.
4. To facilitate students in differentiating tools

Course Outcomes and Cognitive Level Mapping

On the successful completion of the course, students will be able to

CO Number	Course Outcome	Cognitive Level
CO1	Describe the fundamental principles and practices associated with each of the agile development methods	K1
CO2	Compare agile software development model with traditional development models and identify the benefits and pitfalls	K2
CO3	Use techniques and skills to establish and mentor Agile Teams for effective software development	K3
CO4	Apply core values and principles of Agile Methods in software development	K4
CO5	Develop Extreme Programming	K5

Mapping of CO with PO and PSO

	PSO1	PSO2	PSO3	PSO4	PSO5	PO1	PO2	PO3	PO4	PO5
CO1	2	2	3	2	1	2	2	2	3	2
CO2	3	2	3	2	2	3	3	2	3	2
CO3	3	3	3	2	2	3	3	2	3	3
CO4	3	2	3	2	3	2	2	2	3	3
CO5	3	3	3	2	3	3	3	2	2	3

"1" – Slight (Low) Correlation **"2" – Moderate (Medium) Correlation**

"3" – Substantial (High) Correlation **"-" indicates there is no correlation.**

Syllabus

UNIT	CONTENT	HOURS	COs	COGNITIVE LEVEL
I	Fundamentals of Agile Process: Introduction and background, Agile Manifesto and Principles, Stakeholders and Challenges, Overview of Agile Development Models: Scrum, Extreme Programming, Feature Driven Development, Crystal, Kanban, and Lean Software Development.	18	CO1, CO2, CO3, CO4, CO5	K1, K2, K3, K4, K5
II	Agile Projects: Planning for Agile Teams: Scrum Teams, XP Teams, General Agile Teams, Team Distribution; Agile Project Lifecycles: Typical Agile Project Lifecycles, Phase Activities, Product Vision, Release Planning: Creating the Product Backlog, User Stories, Prioritizing and Estimating, Creating the Release Plan; Monitoring and Adapting: Managing Risks and Issues, Retrospectives	18	CO1, CO2, CO3, CO4, CO5	K1, K2, K3, K4, K5
III	Introduction to Scrum: Agile Scrum Framework, Scrum Artifacts, Meetings, Activities and Roles, Scrum Team Simulation, Scrum Planning Principles, Product and Release Planning, Sprinting: Planning, Execution, Review and Retrospective; User story definition and Characteristics, Acceptance tests and Verifying stories, Burn down chart, Daily scrum	18	CO1, CO2, CO3, CO4, CO5	K1, K2, K3, K4, K5
IV	Introduction to Extreme Programming (XP): XP Lifecycle, The XP Team, XP Concepts: Refactoring, Technical Debt, Time boxing, Stories, Velocity; Adopting XP: Pre-requisites, Challenges; Applying XP: Thinking- Pair Programming, Collaborating, Release, Planning, Development	18	CO1, CO2, CO3, CO4, CO5	K1, K2, K3, K4, K5
V	Agile Software Design and Development: Agile design practices, Role of design Principles, Need and significance of Refactoring, Refactoring Techniques, Continuous Integration, Automated build tools, Version control; Agility and Quality Assurance: Agile Interaction Design, Agile approach to Quality Assurance, Test Driven Development, Pair programming: Issues and Challenges	18	CO1, CO2, CO3, CO4, CO5	K1, K2, K3, K4, K5
VI	**Self Study for Enrichment:** **(Not included for End Semester Examination)** Scrum Case Study- XP Case Study-Payroll Case Study-Weather Station Study- Proxy and Stairway to Heaven	-	CO1, CO2, CO3, CO4, CO5	K1, K2, K3, K4, K5

Recommended Books:

1. Robert C. Martin, Agile Software Development- Principles, Patterns and Practices, Prentice Hall, 2013.
2. Kenneth S. Rubin, Essential Scrum: A Practical Guide to the Most Popular Agile Process, Addison Wesley, 2012.
3. James Shore and Shane Warden, The Art of Agile Development, O'Reilly Media, 2007.
4. Craig Larman, ―Agile and Iterative Development: A manager's Guide, Addison-Wesley, 2004.
5. Ken Schawber, Mike Beedle, Agile Software Development with Scrum, Pearson, 2001.
6. Cohn, Mike, Agile Estimating and Planning, Pearson Education, 2006.
7. Cohn, Mike, User Stories Applied: For Agile Software Development Addison Wisley, 2004.

Online Resources:

1. IEEE Transactions on Software Engineering
2. IEEE Transactions on Dependable and Secure Computing
3. IET Software
4. ACM Transactions on Software Engineering and Methodology (TOSEM)
5. ACM SIGSOFT Software Engineering Notes

Pedagogy

Chalk and Talk, PPT, Discussion, Assignment, Demo, Quiz and Seminar.

Course Designer

Dr.J. Sangeetha, Dept. of Information Technology.