

JERUSALEM COLLEGE OF ENGINEERING

(An Autonomous Institution)

(Approved by AICTE, Affiliated to Anna University

Accredited by NBA and NAAC with 'A' Grade)

Velachery Main Road, Narayanapuram, Pallikaranai, Chennai – 600 100



REGULATION 2021

CHOICE BASED CREDIT SYSTEM

M.E

SOFTWARE ENGINEERING SYLLABUS AND CURRICULUM



JERUSALEM COLLEGE OF ENGINEERING
(An Autonomous Institution Affiliated to Anna University, Chennai)

VISION OF THE INSTITUTION

Jerusalem College of Engineering is committed in emerging as an international institution of excellence in imparting finest quality engineering, technology and management education rooted in ethical and societal values through various academic programmes, multi-disciplinary research, consultancy and entrepreneurship activities and hence to contribute towards social transformation and nation building.

MISSION OF THE INSTITUTION

- Generating abundant resources and making conducive policies, the management led by the Chief Executive Officer strives towards promoting globally competitive academic programmes augmented with value added courses, in-plant training activities, co-curricular activities and ambience that support intellectual growth and skill acquisition
- Promoting collaborative trans-border research programmes continuing education in synergy with academia, industries and research organizations leading to real time solutions and life-long learning
- Transforming young men and women into competent professionals and entrepreneurs motivated by a passion for professional excellence, driven by human values and proactively engage in the betterment of the society through innovative practices and academic excellence
- Facilitating effective interaction among faculty members and students and fostering network of alumni, industries, institutions and other stake-holders for successful career gain and placement

VISION OF THE DEPARTMENT

Department of Information Technology strives to provide quality education, academic excellence based on ethical and societal values, exposing students to all concepts, so as to promote global competitiveness in higher education, multi-disciplinary research and entrepreneurship.

MISSION OF THE DEPARTMENT

- To attain academic excellence through innovative practices in teaching and research methodologies.
- To produce globally competent information technologists and entrepreneurs.
- To motivate students to pursue higher education interlaced with communication skills leading to lifelong learning and societal transformations.
- To provide excellence in multi-disciplinary research and development activities rooted in ethical and moral values.

PROGRAMME EDUCATIONAL OBJECTIVES (PEOs)

PEO1: Use all the phases of the Software Development Life Cycle to build robust software systems.

PEO2: Analyze and evaluate problems critically using the theoretical and technical knowledge to develop sustainable solutions and systems.

PEO3: Identify the requirements and implement reliable, innovative and appropriate software solutions for the industrial need.

PEO4: Enhance skills through lifelong learning as software professionals to progress in managerial and leadership roles.

PEO5: Work efficiently in multidisciplinary teams with effective communication and follow ethical principles.

PROGRAMME OUTCOMES

1. An ability to independently carry out research/investigation and development work to solve practical problems.
2. An ability to write and present a substantial technical report/document Students should be able to demonstrate a degree of mastery over the area as per the specialization of the program.
3. The mastery should be at a level higher than the requirements in the appropriate bachelor program.
4. Collect requirements from the stakeholders and design software engineering applications with deep understanding of best software principles and practices.
5. Apply software testing techniques to produce error free and reliable software and ensure quality.
6. Manage software project with state of the art approaches to ensure balance in all project areas like time, cost, quality, risk and human resource.

JERUSALEM COLLEGE OF ENGINEERING
(AN AUTONOMOUS INSTITUTION AFFILIATED ANNA UNIVERSITY, CHENNAI)
M.E. SOFTWARE ENGINEERING
REGULATION 2021
CHOICE BASED CREDIT SYSTEM I
TO IV SEMESTERS CURRICULUM
COURSE SUMMARY SHEET

Sl.No.	Category	Credits as per Semester				Total Credits
		1	2	3	4	
1	FC	3	-	-	-	3
2	PC	16	14			30
3	PE	-	6	6	-	12
4	OE	-	-	3	-	3
5	EEC	-	2	8	12	22
TOTAL (M.E)		19	22	17	12	70

JERUSALEM COLLEGE OF ENGINEERING
(AN AUTONOMOUS INSTITUTION AFFILIATED ANNA UNIVERSITY, CHENNAI)
M.E. SOFTWARE ENGINEERING
REGULATION 2021
CHOICE BASED CREDIT SYSTEM
I TO IV SEMESTERS CURRICULUM AND SYLLABI

SEMESTER 1

Sl. No	Course Code	Course Title	Category	Contact Periods	L	T	P	C
THEORY								
1	JMA5103	Applied Probability and Algebra	FC	3	3	0	0	3
2	JCP5101	Advanced Data Structures and Algorithms	PC	3	3	0	0	3
3	JSE5101	Software Architecture	PC	3	3	0	0	3
4	JSE5102	Software Requirements Engineering	PC	3	3	0	0	3
5	JSE5103	Software Testing and Quality Assurance	PC	3	3	0	0	3
6		Non credit Mandatory Course 1	NCM	2	2	0	0	0
PRACTICALS								
7	JCP5111	Data Structures and Algorithms Laboratory	PC	4	0	0	4	2
8.	JSE5111	Software Development Process Laboratory	PC	4	0	0	4	2
TOTAL				25	17	0	8	19

SEMESTER 2

Sl. No	Course Code	Course Title	Category	Contact Periods	L	T	P	C
THEORY								
1	JSE5201	Advanced Databases	PC	3	3	0	0	3
2	JCP5201	Advanced Operating Systems	PC	4	3	1	0	4
3	JSE5202	Advanced Software Engineering	PC	3	3	0	0	3
4		Professional Elective 1	PE	3	3	0	0	3
5		Professional Elective 2	PE	3	3	0	0	3
6		Non credit Mandatory Course 2	NCM	2	2	0	0	0
PRACTICALS								
7	JSE5211	Advanced Databases Laboratory	PC	4	0	0	4	2
8	JSE5212	Advanced Software Engineering Laboratory	PC	4	0	0	4	2
9	JSE5241	Mini Project	EEC	2	0	0	2	2
TOTAL				27	17	0	10	21

SEMESTER 3

Sl.No	Course Code	Course Title	Category	Contact Periods	L	T	P	C
THEORY								
1		Professional Elective 3	PE	3	3	0	0	3
2		Professional Elective 4	PE	3	3	0	0	3
3		Open Elective	OE	3	3	0	0	3
4	JRM5301	Research Methodology and IPR	EEC	2	2	0	0	2
PRACTICALS								
5	JSE5361	Project Work - Phase 1	EEC	12	0	0	12	6
TOTAL				23	11	0	12	17

SEMESTER 4

Sl. No	Course Code	Course Title	Category	Contact Periods	L	T	P	C
PRACTICALS								
1	JSE5461	Project Work - Phase 2	EEC	24	0	0	24	12
TOTAL				24	0	0	24	12

TOTAL NO. OF CREDITS: 70**FOUNDATION COURSE (FC)****SEMESTER 1**

Sl. No	Course Code	Course Title	Category	Contact Periods	L	T	P	C
PRACTICALS								
1	JMA5103	Applied Probability and Algebra	FC	3	3	0	0	3

PROFESSIONAL CORE (PC)

Sl. No	Course Code	Course Title	Category	Contact Periods	L	T	P	C
1.	JCP5101	Advanced Data Structures and Algorithms	PC	3	3	0	0	3
2.	JSE5101	Software Architecture	PC	3	3	0	0	3
3.	JSE5102	Software Requirements Engineering	PC	3	3	0	0	3
4.	JSE5103	Software Testing and Quality Assurance	PC	3	3	0	0	3
5.	JCP5111	Data Structures and Algorithms Laboratory	PC	4	0	0	4	2
6.	JSE5111	Software Development Process Laboratory	PC	4	0	0	4	2
7.	JSE5201	Advanced Databases	PC	3	3	0	0	3
8.	JCP5201	Advanced Operating Systems	PC	4	3	1	0	4
9.	JSE5202	Advanced Software Engineering	PC	3	3	0	0	3
10.	JSE5211	Advanced Databases Laboratory	PC	4	0	0	4	2
11.	JSE5212	Advanced Software Engineering Laboratory	PC	4	0	0	4	2

PROFESSIONAL ELECTIVES (PE)

SEMESTER 2

ELECTIVE – 1

Sl. No	Course Code	Course Title	Category	Contact Periods	L	T	P	C
1.	JSE5001	Fundamentals of Machine Learning	PE	3	3	0	0	3
2.	JSE5002	Cloud Computing	PE	3	3	0	0	3
3.	JSE5003	Information Retrieval Techniques	PE	3	3	0	0	3
4.	JSE5004	User Interface Design	PE	3	3	0	0	3
5.	JSE5005	Cryptocurrency and Blockchain Technologies	PE	3	3	0	0	3
6.	JSE5006	Agile Software Engineering	PE	3	3	0	0	3

**SEMESTER 2
ELECTIVE - 2**

Sl. No	Course Code	Course Title	Category	Contact Periods	L	T	P	C
1.	JSE5007	Deep Learning	PE	3	3	0	0	3
2.	JSE5008	Semantic Web	PE	3	3	0	0	3
3.	JSE5009	Cyber Security	PE	3	3	0	0	3
4.	JSE5010	Integrated Software Project Management	PE	3	3	0	0	3
5.	JSE5011	Web Content Design and Management	PE	3	3	0	0	3
6.	JSE5012	Ethical Hacking	PE	3	3	0	0	3

**SEMESTER 3
ELECTIVE - 3**

Sl. No	Course Code	Course Title	Category	Contact Periods	L	T	P	C
1.	JSE5013	GPU Computing	PE	3	3	0	0	3
2.	JSE5014	Information Storage Management	PE	3	3	0	0	3
3.	JSE5015	Software Security	PE	3	3	0	0	3
4.	JSE5016	Enterprise Application Integration	PE	3	3	0	0	3
5.	JSE5017	Software Reliability Metrics and Models	PE	3	3	0	0	3
6.	JSE5018	Software Test Automation	PE	3	3	0	0	3

**SEMESTER 3
ELECTIVE - 4**

Sl. No	Course Code	Course Title	Category	Contact Periods	L	T	P	C
1.	JSE5019	Multimedia Systems and Applications	PE	3	3	0	0	3
2.	JSE5020	Software Verification and Validation	PE	3	3	0	0	3
3.	JSE5021	Software Agents	PE	3	3	0	0	3
4.	JSE5022	Operating Systems Security	PE	3	3	0	0	3
5.	JSE5023	Sentiment Analysis	PE	3	3	0	0	3
6.	JSE5024	Principles Of Supply Chain Management	PE	3	3	0	0	3

NON CREDIT MANDATORY COURSE (NCM)

Sl. No	Course Code	Course Title	Category	Contact Periods	L	T	P	C
1.	JNC5001	English for Research Paper Writing	NCM	2	2	0	0	0
2.	JNC5002	Disaster Management	NCM	2	2	0	0	0
3.	JNC5003	Sanskrit for Technical Knowledge	NCM	2	2	0	0	0
4.	JNC5004	Value Education	NCM	2	2	0	0	0
5.	JNC5005	Constitution of India	NCM	2	2	0	0	0
6.	JNC5006	Pedagogy Studies	NCM	2	2	0	0	0
7.	JNC5007	Stress Management by Yoga	NCM	2	2	0	0	0
8.	JNC5008	Personality Development Through Life Enlightenment Skills	NCM	2	2	0	0	0
	JNC5009	Unnat Bharat Abhiyan	NCM	2	2	0	0	0

EMPLOYABILITY ENHANCEMENT COURSES (EEC)

Sl. No	Course Code	Course Title	Category	Contact Periods	L	T	P	C
1.	JSE5241	Mini Project	EEC	2	0	0	2	2
2.	JRM5301	Research Methodology and IPR	EEC	2	2	0	0	2
3.	JSE5361	Project Work - Phase 1	EEC	12	0	0	12	6
4.	JSE5461	Project Work - Phase 2	EEC	24	0	0	24	12

OPEN ELECTIVES (OE)

***(Offered to other Departments)**

Sl. NO	Course Code	Course Title	Category	Contact Periods	L	T	P	C
SEMESTER - 3								
1	JSE9001	Software Testing and Reuse	OE	3	3	0	0	3
2	JSE9002	Software Development and Management	OE	3	3	0	0	3
3	JSE9003	Software Engineering Practices	OE	3	3	0	0	3
4	JSE9004	Design Patterns	OE	3	3	0	0	3
5	JSE9005	Web Design	OE	3	3	0	0	3
6	JSE9006	Software Testing using Automated Tools	OE	3	3	0	0	3

COURSE OBJECTIVES:

- To extend the student's knowledge of algorithms and data structures
- To enhance their expertise in algorithmic analysis and algorithm design techniques
- To understand various types of search and heap structures
- To study various types of geometric, randomized and approximation algorithms
- To extrapolate from them in order to apply those algorithms and techniques to solve problems

UNIT I FUNDAMENTALS OF ALGORITHM ANALYSIS 9

Properties of Big-oh Notation – Conditional Asymptotic Notation – Floors and Ceilings– Algorithm Analysis – Amortized Analysis – Probabilistic analysis – Introduction to NP- Completeness/NP-Hard – Recurrence Equations – Solving Recurrence Equations – Time-Space Trade off.

UNIT II SEARCH STRUCTURES 9

Binary Search Trees – AVL Trees – Red-Black trees –WAVL trees – Multi-way Search Trees – B-Trees – Splay Trees – Tries

UNIT III HEAP STRUCTURES 9

Min/Max heaps – Deaps – Leftist Heaps – Binomial Heaps – Fibonacci Heaps – Skew Heaps – Lazy Binomial Heaps – TREAPS

UNIT IV GEOMETRIC ALGORITHMS 9

Segment Trees – 1-Dimensional Range Searching – k-d Trees – Line Segment Intersection – Computing the Overlay of Two Subdivisions – Range Trees – Voronoi Diagram –Point Location

UNIT V CLASSES OF ALGORITHMS 9

Approximation Algorithms: Vertex Cover & Euclidean Travelling Salesperson Problem – Randomized Algorithms: Closest Pair Problem & Minimum Spanning Trees – Online Algorithm : Euclidean Spanning Tree & K-server Problems

TOTAL: 45 PERIODS

COURSE OUTCOMES:

On completion of the course, the students should be able to:

- Analyse algorithms for its efficiency
- Determine algorithm correctness

- Choose appropriate data structures for the problems to be solved
- Design algorithms for problems from different domains
- Identify various research strategies on algorithmic design

TEXT BOOKS:

1. R.C.T Lee, S.S Tseng, R.C Chang and Y.T Tsai, “Introduction to the Design and Analysis of Algorithms”, Tata McGraw-Hill Edition, 2012.

REFERENCES:

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, "Introduction to Algorithms", Third Edition, Prentice-Hall, 2011.
2. Ellis Horowitz, Sartaj Sahni, Susan Anderson-Freed, "Fundamentals of Data Structures in C", Second Edition, University Press, 2008.
3. Mark de Berg, Otfried Cheong, Marc van Kreveld, Mark Overmars, "Computational Geometry Algorithms and Applications", Third Edition, Springer, 2008.
4. Anany Levitin , "Introduction to the Design and Analysis of Algorithms", Pearson Education; Third edition February 2017.
5. G. A. V. Pai, "Data Structures and Algorithms: Concepts - Techniques and Applications", McGraw Hill Education; 1st edition 2017.

Mapping of CO with PO

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO-1	3				2	
CO-2	3		3		3	
CO-3	3	3	3		3	
CO-4		3	3		3	
CO-5	3	2	3		2	

JSE5101

SOFTWARE ARCHITECTURE

L T P C

3 0 0 3

COURSE OBJECTIVES:

- Understand the fundamentals of software architecture.
- Study the various software development methodologies.
- Learn the importance of architectural documentation and evaluation.

- Learn the various software architecture design components.
- Relate software architecture and software quality.

UNIT I INTRODUCTION 9

Basic concepts of software architecture – Context of Software Architecture – ABC cycle – What software architecture is and what it isn't – Architectural patterns – Good Architecture- Reference models – Architectural structures and views-Introduction to styles – Decentralized Architectures.

UNIT II DESIGN METHODOLOGIES 9

Structured design- Design practices -Stepwise refinement – Incremental design- Structured system analysis and design –Jackson structured programming – Jackson system Development.

UNIT III ARCHITECTURAL DESCRIPTION DOCUMENTATION AND EVALUATION 9

Early architecture description languages-Domain and style specific ADL's- Extensible ADL's – Documenting software architecture – Uses and Audiences for Architecture Documentation – Views – Choosing Views – Combining Views –Architecture evaluation – Evaluation Factors – Architecture Tradeoff Analysis Method – Lightweight Architecture Evaluation – ATAM.

UNIT IV ARCHITECTURE DESIGN 9

Typical architectural design-Dataflow-Independent components-Call and return – Using styles in design – Architectural design space-Design space of architectural elements – Design space of architectural styles.

UNIT V IMPLEMENTATION AND CONFORMANCE TO ARCHITECTURE 9

Understanding quality attributes- Implementation of Quality attributes in Architecture – Architecture and requirements conformance –Functionality– Quality attribute considerations – System quality attributes-Introduction to tactics – Achieving Quality Attributes through Tactics –Tactics types – Architectural patterns and styles – Architecture and Quality Attributes – Quality attribute scenarios in practice.

TOTAL: 45 PERIODS

COURSE OUTCOMES:

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

- Develop Software applications starting from software architecture and design.
- Learn and evaluate existing software architectures.
- Realize importance of architectural documentation and document them.
- Employ various software architecture design components.
- Design methods for improving software quality from the perspective of software architecture.

REFERENCES:

1. Len Bass, Paul Clements, Rick Kazman, “Software Architecture in Practice”, Third Edition, Addison, Wesley, 2012.

Method – Integration of Techniques – Fact-Finding – Requirements Gathering – Evaluation and Rationalization – Prioritization– Integration and Validation – Risk Assessment and Risk Treatment (RART).

UNIT III REQUIREMENTS ANALYSIS 9

Identification of Functional and Non Functional Requirements – Identification of Quality Requirements – Six Quality Attributes– Analysis – Feasibility and Internal Compatibility of System Requirements – Definition of Human Requirements Baseline.

UNIT IV REQUIREMENTS DEVELOPMENT 9

Requirements analysis – Requirements Documentation – Requirements Development Workflow – Fundamentals of Requirements Development – Requirements Attributes Guidelines Document – Supplementary Specification Document – Use Case Specification Document – Methods for Software Prototyping – Evolutionary prototyping –Throwaway prototyping.

UNIT V REQUIREMENTS VALIDATION 9

Validation objectives – Analysis of requirements validation – Activities – Properties – Requirement reviews – Requirements testing – ISO 31000 – Case tools for requirements engineering.

TOTAL : 45 PERIODS

COURSE OUTCOMES:

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

- Prepare SRS including the details of requirements engineering.
- Describe the stages of requirement elicitation.
- Analyze software requirements gathering.
- Integrate the requirements well during requirements analysis.
- Use various methodologies for requirements development.

REFERENCES:

1. Ian Sommerville, Pete Sawyer,” Requirements Engineering: A Good Practice Guide”, Sixth Edition, Pearson Education, 2004.
2. Dean Leffingwe, DonWidrig, “Managing Software Requirements A UseCaseApproach”, Second Edition, Addison Wesley,2003.
3. Karl Eugene Wiegers, “Software Requirements”, Word Power Publishers, 2000.
4. Ian Graham, “Requirements Engineering and Rapid Development”, AddisonWesley,1998.
5. Wiegers, Karl,JoyBeatty, “Software Requirements”, Pearson Education,2013.

Mapping of CO with PO

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO-1	3					
CO-2	3		3	2		
CO-3	3		3	3	2	
CO-4			3	3		
CO-5	3		3	3		

JSE5103 SOFTWARE TESTING AND QUALITY ASSURANCE L T P C
3 0 0 3

COURSE OBJECTIVES:

The student should be able to

- Know what is software and the usage of different types of softwares.
- Know the Quality Metrics of various Softwares.
- Know the methodologies used in developing Software.
- Test the product finally to check the product Quality.

UNIT I INTRODUCTION

9

Introduction to Software Quality – Challenges – Objectives – Quality Factors – Components of SQA – Contract Review – Development and Quality Plans – SQA Components in Project Life Cycle – SQA Defect Removal Policies – Reviews.

UNIT II TESTING METHODOLOGIES

9

Basics of Software Testing – Test Generation from Requirements – Finite State Models – Combinatorial Designs – Test Selection, Minimization and Prioritization for Regression Testing – Test Adequacy, Assessment and Enhancement.

Suggested Activities:

Install Selenium to test a code.

UNIT III TEST STRATEGIES

9

Testing Strategies – White Box and Black Box Approach – Integration Testing – System and Acceptance Testing – Performance Testing – Regression Testing – Internationalization Testing Ad-hoc Testing – Website Testing – Usability Testing – Accessibility Testing.

UNIT IV TEST AUTOMATION AND MANAGEMENT

9

Test plan – Management – Execution and Reporting – Software Test Automation – Automated

Testing tools – Hierarchical Models of Software Quality – Configuration Management – Documentation Control.

UNIT V SQA IN PROJECT MANAGEMENT

9

Project progress control – costs – quality management standards – project process standards – management and its role in SQA – SQA unit.

TOTAL:45 PERIODS

COURSE OUTCOMES

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

- Develop Quality plans and use SQA components in project life cycle.
- Analyze the product Quality.
- Judge the use of infrastructure components and use configuration items for Quality control.
- Use various testing methods and verify.
- Assess Quality standards of various software products.

REFERENCES

1. Daniel Galin, “Software Quality Assurance – from Theory to Implementation” Pearson Education, 2009
2. Yogesh Singh, “Software Testing”, Cambridge University Press, 2012
3. AdityaMathur, “Foundations of Software Testing”, Pearson Education, 2008
4. Ron Patton, “Software Testing” , Second Edition, Pearson Education, 2007
5. Srinivasan Desikan, Gopaldaswamy Ramesh, “Software Testing – Principles and Practices”, Pearson Education, 2006
6. Alan C Gillies, “Software Quality Theory and Management”, Cengage Learning, Second Edition, 2003.
7. Robert Furtell, Donald Shafer, and Linda Shafer, “Quality Software Project Management”, Pearson Education Asia, 2002.

Mapping of CO with PO

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO-1	3					
CO-2	3		3	2	3	
CO-3	3		3	3	3	
CO-4			3	3	2	
CO-5	3		3	3	3	

COURSE OBJECTIVES:

- To familiarize various data structure implementations.
- To implement heap and various tree structures like AVL, Red-black, B- Tree and segment trees.
- To understand efficient implementation of line segment intersection.
- To understand various search structures.
- To get understanding of problem to program mapping.

LAB EXERCISES: It is recommended that all implementations are carried out in Java. If C or C++ has to be used, then the threads library will be required for concurrency. Exercises should be designed to cover the following topics:

LIST OF EXPERIMENTS:

1. Binary Search Trees
2. Min/Max Heap
3. Leftist Heap
4. Fibonacci Heap
5. AVL Trees
6. Red-Black Trees
7. B-Trees
8. Segment Trees
9. Line segment intersection
10. Spanning Tree Implementation
11. Closest Pair Problem

TOTAL : 60 PERIODS**COURSE OUTCOMES:**

On completion of the course, the students should be able to:

- Design and implement Hierarchical structures
- Design and implement heap data structures
- Design algorithms using graph structures
- Implement problems using Dynamic programming
- Use Greedy algorithms to solve problems

COURSE OBJECTIVES:

- To learn the stages of software development process.
- To carry out Planning.
- To understand the importance of documentation.
- To equip in preparing software project documentation.
- To use models developed, code and test.

SUGGESTED LIST OF EXERCISES:

Choose any one application for performing the following phases.

1. Program Analysis and Project Planning.
Thorough study of the problem – Identify project scope, Objectives, Infrastructure –PROJECT PLAN DOCUMENTATION
2. Software Requirement Analysis
Describe the individual Phases / Modules of the project, Identify deliverables – SRS DOCUMENTATION
3. Data Modeling
Use work products – Data dictionary, Use case diagrams and activity diagrams, build and test class diagrams, Sequence diagrams, add interface to class diagrams. – DESIGN DOCUMENTATION
4. Software Development and Debugging
Use technology of your choice to develop and debug the application– CODE DOCUMENTATION
5. Software Testing
Perform validation testing, Coverage analysis, memory leaks, develop test case hierarchy and Site monitor. – TEST CASE DOCUMENTATION

TOTAL: 60 PERIODS

SUGGESTED LIST OF APPLICATIONS:

1. Inventory System.
2. Book Lending Application.
3. Car Booking Application.
4. Online Help Portal.
5. Online Payment Portal.
6. Student Marks Analyzing System.

7. Quiz System.
8. Online Ticket Reservation System.
9. Payroll System.
10. Course Registration System.

COURSE OUTCOMES:

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

- Prepare project plan.
- Prepare SRS as per standards.
- Prepare and use design document.
- Prepare and use code document.
- Prepare and use test case documentation at appropriate stages of software development process.

JSE5201

ADVANCED DATABASES

L T P C

3 0 0 3

COURSE OBJECTIVES:

- To comprehend the underlying principles of Relational Database Management System
- To develop database models using parallel and distributed databases
- To understand the concepts of XML and Web databases
- To apprehend the design and implementation of active temporal and deductive databases
- To develop applications based on NoSQL database

UNIT I RELATIONAL MODEL

9

Entity Relationship Model – Relational Data Model – Mapping Entity Relationship Model to Relational Model – Relational Algebra – Structured Query Language – Database Normalization – Transaction Management-Recovery

UNIT II PARALLEL AND DISTRIBUTED DATABASES

9

Parallel Databases – I/O Parallelism – Inter-Query and Intra-Query Parallelism – Inter-Operation and Intra-operation Parallelism – Distributed Database Architecture – Distributed Data Storage – Distributed Transactions – Distributed Query Processing – Distributed Transaction Management

UNIT III XML AND WEB DATABASES

9

XML Data Model – DTD – XML Schema – XML Querying – Web Databases – Open Database Connectivity – Java Database Connectivity – Accessing Relational database using PHP

UNIT IV ACTIVE TEMPORAL AND DEDUCTIVE DATABASES

9

Event Condition Action Model – Design and Implementation Issues for Active Databases – Temporal Databases – Interpreting Time in Relational Databases – Deductive Databases – Data log Queries

UNIT V NoSQL DATABASES

9

NoSQL database vs traditional RDBMS database – Migrating from RDBMS to NoSQL –MongoDB
– Database creation and Querying – Web Application development using MongoDB

TOTAL : 45 PERIODS

COURSE OUTCOMES:

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

- Design and implement relational databases.
- Design and implement parallel and distributed databases.
- Design and implement XML databases, Active, Temporal and Deductive databases.
- Implement the concept of database connectivity with the applications.
- Apply various data mining techniques.

REFERENCES:

1. R. Elmasri, S.B. Navathe, “Fundamentals of Database Systems”, Addison-Wesley, 2011.
2. Han, Jiawei, Jian Pei, and Micheline Kamber. Data mining: concepts and Techniques. 2011.
3. Thomas Cannolly and Carolyn Begg, “Database Systems, A Practical Approach to Design, Implementation and Management”, Third Edition, Pearson Education, 2007.
4. Henry F Korth, Abraham Silberschatz, S. Sudharshan, “Database System Concepts”, Fifth Edition, McGraw Hill, 2006.
5. C.J.Date, A.Kannan and S.Swamynathan, “An Introduction to Database Systems”, Eighth Edition, Pearson Education, 2006.
6. V.S.Subramanian, “Principles of Multimedia Database Systems”, Harcourt India Pvt. Ltd.,2001.
7. ShashankTiwari, ”Professional NoSQL”, Wiley, 2011.

Mapping of CO with PO

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO-1	3					
CO-2	3		3	2		
CO-3	3	3	3	3	2	
CO-4		3	3	3		
CO-5	3	2	3	3		

COURSE OBJECTIVES:

- To learn the fundamentals of Operating Systems
- To gain knowledge on Distributed operating system concepts
- To gain insight on to the distributed resource management components
- To know the components and management aspects of Real time and Mobile operating systems
- To learn the implications of OS in an Android System

UNIT I FUNDAMENTALS OF OPERATING SYSTEMS**12**

Overview – Synchronization Mechanisms – Processes and Threads - Process Scheduling – Deadlocks: Detection, Prevention and Recovery – Models of Resources – Memory Management Techniques - Introduction to multiprocessor OS

UNIT II DISTRIBUTED OPERATING SYSTEMS**12**

Issues in Distributed Operating System – Architecture – Communication Primitives – Lamport's Logical clocks – Causal Ordering of Messages – Distributed Mutual Exclusion Algorithms – Centralized and Distributed Deadlock Detection Algorithms – Agreement Protocols

UNIT III DISTRIBUTED RESOURCE MANAGEMENT**12**

Distributed File Systems – Design Issues - Distributed Shared Memory – Algorithms for Implementing Distributed Shared memory–Issues in Load Distributing – Scheduling Algorithms – Synchronous and Asynchronous Check Pointing and Recovery – Fault Tolerance – Two-Phase Commit Protocol – Nonblocking Commit Protocol – Security and Protection

UNIT IV REAL TIME AND MOBILE OPERATING SYSTEMS**12**

Basic Model of Real Time Systems - Characteristics- Applications of Real Time Systems – Real Time Task Scheduling - Handling Resource Sharing - Mobile OS: Architecture, Android OS, iOS, Virtual OS, Cloud OS and their design issues

UNIT V CASESTUDIES**12**

Linux System: Design Principles - Kernel Modules - Process Management Scheduling - Memory Management - Input-Output Management - File System - Interprocess Communication. iOS and Android: Architecture and SDK Framework - Media Layer - Services Layer - Core OS Layer - File System

TOTAL : 60 PERIODS

COURSE OUTCOMES:

Upon completion of the course, the students will be able to

- Discuss the various synchronization, scheduling and memory management issues
- Demonstrate the Mutual exclusion, Deadlock detection and agreement protocols of Distributed operating system
- Discuss the various resource management techniques for distributed systems
- Identify the different features of real time and mobile operating systems
- Install and use available open source kernel Modify existing open source kernels in terms of functionality or features used

REFERENCES:

1. Mukesh Singhal and Niranjan G. Shivaratri, “Advanced Concepts in Operating Systems – Distributed, Database, and Multiprocessor Operating Systems”, Tata McGraw-Hill, 2001.
2. Abraham Silberschatz; Peter Baer Galvin; Greg Gagne, “Operating System Concepts”, Seventh Edition, John Wiley & Sons, 2004.
3. Daniel P Bovet and Marco Cesati, “Understanding the Linux kernel”, 3rd edition, O’Reilly, 2005.
4. Rajib Mall, “Real-Time Systems: Theory and Practice”, Pearson Education India, 2006.
5. Neil Smyth, “iPhone iOS 4 Development Essentials – Xcode”, Fourth Edition, Payload media, 2011.

Mapping of CO with PO

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO-1	3					
CO-2	3		3	2		
CO-3	3	3	3	3	2	
CO-4	3	3	3	3		
CO-5	3	2	3	3		

JSE5202

ADVANCED SOFTWARE ENGINEERING

L T P C

3 0 0 3

COURSE OBJECTIVES:

- Comprehend the different stages of Software Development Lifecycle.
- Comprehend the Process of developing Analysis models and map the Analysis models to Design Models.
- Comprehend the Design Issues related to Web applications and Mobile Apps.

- Comprehend the Quality Factors associated with Software Development.
- Comprehend the use of different Testing Strategies in Software Development.

UNIT I PROCESS MODELS

9

Prescriptive process models–Specialized process models–The unified process–personal and team process models–Product and Process–Agile development–Extreme Programming–Other Agile process models–Human aspects of Software Engineering

UNIT II REQUIREMENTS MODELING AND DESIGN CONCEPTS

9

Understanding Requirements–Scenario based methods–Class Based Methods–Behavior, Patterns and Web/Mobile Apps–The design process–Design concepts–The Design model

UNIT III SOFTWARE DESIGN

9

Architectural design–Component level Design–User Interface Design–Pattern based design–WebApp design– Mobile App design

UNIT IV SOFTWARE QUALITY

9

Garvin’s Quality dimensions–McCall’s Quality factors–ISO9126 Quality factors–The software Quality Dilemma–Achieving Software Quality–Review Techniques–Elements of Software Quality Assurance–SQA Processes and Product Characteristics–SQA Tasks, Goals, and Metrics–Statistical Software Quality Assurance–Software Reliability–The ISO 9000 Quality Standards– The SQA Plan

UNIT V SOFTWARE TESTING AND SOFTWARE CONFIGURATION MANAGEMENT

9

Software Testing Strategies–Testing Conventional Applications–Testing Object Oriented Applications–Testing Web applications–Testing Mobile Apps–Software Configuration management–The SCM process–Configuration Management for Web and Mobile App.

TOTAL : 45 PERIODS

COURSE OUTCOMES:

Upon Completion of the Course, the Student will be able to

- Select Appropriate Process Model for Software Development.
- Develop Analysis Models and Map the Analysis Models to Design Models.
- Address the Design Issues related To Web Applications and Mobile Apps.
- Incorporate Appropriate Quality Factors and Standards during Software Development.
- Select Appropriate Testing Strategies for Software Testing.

REFERENCES:

1. Roger S. Pressman, “Software Engineering – A Practioner’s Approach”, MC Graw Hill, 8th edition.
2. Ian Sommerville, “Software Engineering”, Addison-Wesley, 9th Edition, 2010.
3. Bernd Bruegge, Allen H. Dutoit, “Object-Oriented Software Engineering”, Prentice Hall, Third Edition, 2009.
4. Robert E. Filman, TzillaElrad, Siobhán Clarke, Mehmet Aksit, “Aspect-Oriented Software Development”, Addison-Wesley Professional, 2004.

Mapping of CO with PO

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO-1	3					
CO-2	3			2	3	
CO-3	3	3		3	3	
CO-4		3		3	3	
CO-5	3	2		3	3	

JSE5211

DATABASES LABORATORY

L T P C

0 0 4 2

COURSE OBJECTIVES:

- To learn the DDL and DML operations.
- To understand the use of various Joins.
- To acquire knowledge on creation of views and nested queries.
- To write and use functions, triggers, and stored procedures.
- To use distributed databases, heterogeneous databases and XML databases.

LIST OF EXPERIMENTS:

1. Data Definition Language–Create–Alter–Drop– Enforcing Primary Key and Foreign Key Constraints–Data Manipulation Language–Insert–Delete–Update–Transaction Control Language– Commit– Rollback– Save Points
2. Cartesian Product – Equijoin – Left Outer Join – Right Outer Join – Full Outer Join

3. Set Operations – Creating Views – Creating Sequence – Indexing–Aggregate Functions – Analytic Functions – Nested Queries
4. Creating Triggers and Stored Procedures
5. Implementation of Distributed Databases
6. Connecting Heterogeneous Databases
7. XML Databases
8. Accessing and Updating a Relational Database using PHP
9. Accessing and Updating a Relational Database using JDBC
10. Accessing and Updating MongoDB using PHP

TOTAL: 60 PERIODS

COURSE OUTCOMES:

Upon completion of this course, the student should be able to

- Create and carry out all Data Manipulation operations.
- Create queries using various Joins appropriately.
- Create and use views and nested queries.
- Write and use functions, triggers, and stored procedures.
- Use distributed databases, heterogeneous databases and XML databases.

JSE5212 ADVANCED SOFTWARE ENGINEERING LABORATORY L T P C

0 0 4 2

COURSE OBJECTIVES:

- To learn to prepare SRS for real time applications.
- To identify the entity relationship and data flow in the application.
- To learn modelling the requirements using various models.
- To use developed models for generating code from the model.
- To get equipped on various levels of testing the developed application.

LIST OF EXPERIMENTS:

1. Prepare SRS for a real time Application
2. Model Entity Relationship Diagram and Data Flow Diagram for the selected case study
3. Model the following UML Diagrams for the selected Case study
 - a) Class Diagram
 - b) Use Case Diagram
 - c) Sequence Diagram
 - d) Component diagram
 - e) State Transition Diagram
 - f) Activity Diagram
 - g) Deployment Diagram

4. Code Generation from UML Diagram
5. Testing
 - a) Unit Testing
 - b) Integration Testing

TOTAL: 60 PERIODS

COURSE OUTCOMES:

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

- Prepare SRS for real time applications.
- Identify the entity relationship and data flow in the application.
- Model the requirements for analysis as well as for design.
- Generate code from the model developed.
- Perform various levels of testing on the developed application.

JSE5001

FUNDAMENTALS OF MACHINE LEARNING

L T P C

3 0 0 3

COURSE OBJECTIVES:

- To understand the concepts of Machine Learning.
- To appreciate supervised learning and their applications.
- To appreciate the concepts and algorithms of unsupervised learning.
- To understand the theoretical and practical aspects of Probabilistic Graphical Models.
- To appreciate the concepts and algorithms of advanced learning.

UNIT I INTRODUCTION

9

Machine Learning– Machine Learning Process- Preliminaries for Machine Learning algorithms – Turning data into Probabilities and Statistics for Machine Learning- Probability theory – Probability Distributions – Decision Theory.

UNIT II SUPERVISED LEARNING

9

Linear Models for Regression – Linear Models for Classification- Discriminant Functions, Probabilistic Generative Models, Probabilistic Discriminative Models – Decision Tree Learning Bayesian Learning, Naïve Bayes – Ensemble Methods, Bagging, Boosting, Neural Networks ,Multi-layer Perceptron – Deriving Back Propagation - Support Vector Machines.

UNIT III UNSUPERVISED LEARNING

9

Clustering- K-means – EM Algorithm- Mixtures of Gaussians – Dimensionality Reduction – Linear Discriminant Analysis - Principal Components Analysis – Locally Linear Embedding – Isomap - Model Evaluation

UNIT IV PROBABILISTIC GRAPHICAL MODELS**9**

Graphical Models – Undirected Graphical Models – Markov Random Fields – Directed Graphical Models – Bayesian Networks – Conditional Independence properties – Markov Random Fields – Hidden Markov Models – Conditional Random Fields (CRFs).

UNIT V ADVANCED LEARNING**9**

Sampling-Basic Sampling methods, Monte Carlo, Gibbs Sampling – Computational Learning Theory – Mistake Bound Analysis – Reinforcement learning – Markov Decision processes, Deterministic and Non-deterministic Rewards and Actions, Temporal Difference Learning Exploration.

TOTAL :45 PERIODS**COURSE OUTCOMES:**

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

- Design a learning model appropriate to the application.
- Design a Neural Network for an application of your choice.
- Implement Probabilistic Discriminative and Generative algorithms for an application of your choice and analyze the results.
- Use a tool to implement typical Clustering algorithms for different types of applications.
- Design and implement an HMM for a Sequence Model type of application.

REFERENCES:

1. Christopher Bishop, “Pattern Recognition and Machine Learning” Springer, 2007.
2. Stephen Marsland, “Machine Learning – An Algorithmic Perspective”, Chapman and Hall, CRC Press, Second Edition, 2014.
3. Kevin P. Murphy, “Machine Learning: A Probabilistic Perspective”, MIT Press, 2012.
4. Ethem Alpaydin, “Introduction to Machine Learning”, MIT Press, Third Edition, 2014.
5. Tom Mitchell, “Machine Learning”, McGraw-Hill, 1997.

Mapping of CO with PO

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO-1	3					
CO-2	3		3	2		
CO-3	3	3	3	3	2	
CO-4		3	3	3		
CO-5	3	2	3	3		

COURSE OBJECTIVES:

- To understand the concept of cloud and utility computing.
- To understand the various issues in cloud computing.
- To familiarize themselves with the lead players in cloud.
- To appreciate the emergence of cloud as the next generation computing paradigm.
- To be able to set up a private cloud.

UNIT I INTRODUCTION**9**

Introduction- Historical Development – Cloud Computing Architecture – The Cloud Reference Model – Cloud Characteristics –Cloud Deployment Models: Public, Private, Community, Hybrid Clouds- Cloud Delivery Models: IaaS, PaaS, SaaS – Open Source Private Cloud Software: Eucalyptus, Open Nebula, Open Stack.

UNIT II VIRTUALIZATION**9**

Data Center Technology – Virtualization – Characteristics of Virtualized Environments - Taxonomy of Virtualization Techniques – Virtualization and Cloud Computing –Pros and Cons of Virtualization – Implementation Levels of Virtualization – Tools and Mechanisms: Xen, VMWare, Microsoft Hyper-V, KVM, Virtual Box

UNIT III CLOUD COMPUTING MECHANISM**9**

Cloud Infrastructure Mechanism: Cloud Storage, Cloud Usage Monitor, Resource Replication – Specialized Cloud Mechanism: Load Balancer, SLA Monitor, Pay-per-use Monitor, Audit monitor, Failover System, Hypervisor, Resource Cluster, Multi Device Broker, State Management database – Cloud Management Mechanism: Remote Administration System, Resource Management System, SLA Management System, Billing Management System

UNIT IV HADOOP AND MAP REDUCE**9**

Apache Hadoop – Hadoop MapReduce –Hadoop Distributed File System- Hadoop I/O Developing a MapReduce Application – MapReduce Types and Formats – MapReduce Features– Hadoop Cluster Setup –Administering Hadoop.

UNIT V SECURITY IN THE CLOUD**9**

Basic Terms and Concepts – Threat Agents – Cloud Security Threats –Cloud Security Mechanism: Encryption, Hashing, Digital Signature, Public Key Infrastructure, Identity and Access Management, Single Sign-on, Cloud Based Security Groups, Hardened Virtual Server Images.

TOTAL: 45 PERIODS

COURSE OUTCOMES:

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

- Articulate the main concepts, key technologies, strengths and limitations of cloud computing.
- Identify the architecture, infrastructure and delivery models of cloud computing.
- Explain the core issues of cloud computing such as security, privacy and interoperability.
- Choose the appropriate technologies, algorithms and approaches for the related issues.
- Facilitate Service Level Agreements (SLA).

REFERENCES:

1. Thomas Erl, ZaighamMahood, Ricardo Puttini, “Cloud Computing, Concept, Technology & Architecture”, Prentice Hall, 2013.
2. RajkumarBuyya, Christian Vecchiola, S. ThamaraiSelvi, “Mastering Cloud Computing”,Tata McGraw-Hill,2013.
3. Toby Velte, Anthony Velte, Robert C. Elsenpeter, “Cloud Computing, A Practical Approach”,Tata McGraw-Hill Edition, 2010.
4. ArshdeepBahga, Vijay Madiseti, “Cloud Computing: A Hands-On Approach”, Universities Press(India) Private Limited, 2014.
5. Tom White, “Hadoop: The Definitive Guide”, O’Reilly Media, 4th Edition, 2015.
6. James E Smith and Ravi Nair, “Virtual Machines”, Elsevier, 2005.
7. John Ritting house& James Ransome, Cloud Computing, Implementation, Management and Strategy, CRC Press, 2010.

Mapping of CO with PO

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO-1	3					3
CO-2	3		3	2		3
CO-3	3	3	3	3	2	2
CO-4		3	3	3		
CO-5	3	2	3	3		

COURSE OBJECTIVES:

- To understand the basics of information retrieval with pertinence to modeling, query operations and indexing
- To get an understanding of machine learning techniques for text classification and clustering.
- To understand the various applications of information retrieval giving emphasis to multimedia IR, web search
- To understand the concepts of digital libraries

UNIT I INTRODUCTION: MOTIVATION**9**

Basic Concepts – Practical Issues - Retrieval Process – Architecture - Boolean Retrieval – Retrieval Evaluation – Open Source IR Systems–History of Web Search – Web Characteristics–The impact of the web on IR —IR Versus Web Search–Components of a Search engine

UNIT II MODELING**9**

Taxonomy and Characterization of IR Models – Boolean Model – Vector Model - Term Weighting – Scoring and Ranking – Language Models – Set Theoretic Models - Probabilistic Models – Algebraic Models – Structured Text Retrieval Models – Models for Browsing

UNIT III INDEXING**9**

Static and Dynamic Inverted Indices – Index Construction and Index Compression. Searching - Sequential Searching and Pattern Matching. Query Operations - Query Languages – Query Processing - Relevance Feedback and Query Expansion - Automatic Local and Global Analysis – Measuring Effectiveness and Efficiency

UNIT IV CLASSIFICATION AND CLUSTERING**9**

Text Classification and Naïve Bayes – Vector Space Classification – Support vector machines and Machine learning on documents. Flat Clustering – Hierarchical Clustering – Matrix decompositions and latent semantic indexing – Fusion and Meta learning

UNIT V SEARCHING THE WEB**9**

Searching the Web – Structure of the Web – IR and web search – Static and Dynamic Ranking – Web Crawling and Indexing – Link Analysis - XML Retrieval Multimedia IR: Models and Languages – Indexing and Searching Parallel and Distributed IR – Digital Libraries

TOTAL : 45 PERIODS

COURSE OUTCOMES:

Upon completion of this course, the student should be able to

- Build an Information Retrieval system using the available tools
- Identify and design the various components of an Information Retrieval system
- Apply machine learning techniques to text classification and clustering which is used for efficient Information Retrieval
- Design an efficient search engine and analyze the Web content structure

REFERENCES:

1. Christopher D. Manning, Prabhakar Raghavan, Hinrich Schutze, — Introduction to Information Retrieval, Cambridge University Press, First South Asian Edition, 2008.
2. Ricardo Baeza – Yates, Berthier Ribeiro – Neto, —Modern Information Retrieval: The concepts and Technology behind Search (ACM Press Books), Second Edition, 2011.
3. Stefan Buttcher, Charles L. A. Clarke, Gordon V. Cormack, —Information Retrieval Implementing and Evaluating Search Engines, The MIT Press, Cambridge, Massachusetts London, England, 2010.

Mapping of CO with PO

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO-1	3					
CO-2	3			2		
CO-3	3	3		3	2	
CO-4		3		3		
CO-5	3	2		3		

JSE5004

USER INTERFACE DESIGN

L T P C

3 0 0 3

COURSE OBJECTIVES:

- To understand the basics of User Interface Design
- To design the user interface, design, menu creation and windows creation
- To understand the concept of menus, windows, interfaces, business functions, various

problems in windows design with colour, text, Non-anthropomorphic Design

- To study the design process and evaluations
- To learn the structure of user Interface and design process

UNIT I INTERACTIVE SOFTWARE AND INTERACTION DEVICE 9

Human–Computer Interface – Characteristics Of Graphics Interface –Direct Manipulation Graphical System – Web User Interface –Popularity –Characteristic & Principles

UNIT II HUMAN COMPUTER INTERACTION 9

User Interface Design Process – Obstacles –Usability –Human Characteristics In Design – Human Interaction Speed –Business Functions –Requirement Analysis – Direct – Indirect Methods – Basic Business Functions – Design Standards – General Design Principles – Conceptual Model Design – Conceptual Model Mock-Ups

UNIT III WINDOWS 9

Characteristics– Components– Presentation Styles– Types– Managements– Organizations– Operations– Web Systems– System Timings - Device– Based Controls Characteristics– Screen – Based Controls — Human Consideration In Screen Design – Structures Of Menus – Functions Of Menus– Contents Of Menu– Formatting – Phrasing The Menu – Selecting Menu Choice–Navigating Menus– Graphical Menus. Operate Control – Text Boxes– Selection Control– Combination Control– Custom Control– Presentation Control

UNIT IV MULTIMEDIA 9

Text For Web Pages – Effective Feedback– Guidance & Assistance– Internationalization– Accessibility– Icons– Image– Multimedia – Coloring- Case Study: Addressing usability in E-Commerce sites

UNIT V DESIGN PROCESS AND EVALUATION 9

User Interface Design Process - Usability Testing - Usability Requirements and Specification procedures and techniques- User Interface Design Evaluation

TOTAL : 45 PERIODS

COURSE OUTCOMES:

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

- Knowledge on development methodologies, evaluation techniques and user interface building tools
- Explore a representative range of design guidelines
- Gain experience in applying design guidelines to user interface design tasks.
- Understand the concepts of multimedia
- Ability to design their own Human Computer

TEXT BOOKS:

1. Wilbent. O. Galitz ,“The Essential Guide To User Interface Design”, John Wiley& Sons, 2001.

REFERENCES:

1. Deborah Mayhew, The Usability Engineering Lifecycle, Morgan Kaufmann, 1999Ben Shneiderman, “Design The User Interface”, Pearson Education, 1998.
2. Alan Cooper, “The Essential Of User Interface Design”, Wiley – Dream Tech Ltd., 2002. Sharp, Rogers, Preece, ‘Interaction Design’, Wiley India Edition, 2007
3. Alan Dix et al, " Human - Computer Interaction ", Prentice Hall, 1993.
4. Ben Schneiderman, " Designing the User Interface ", Addison Wesley, 2000.
5. Universal Principles of Design, Revised and Updated: 125 Ways to Enhance Usability, Influence Perception, Increase Appeal, Make Better Design Decisions, and Teach through Design Flexibound – 1 January 2010.

Mapping of CO with PO

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO-1	3					
CO-2	3		3	2		
CO-3	3		3	3	2	
CO-4			3	3		
CO-5	3		3	3		

JSE5005 CRYPTO CURRENCY AND BLOCKCHAIN TECHNOLOGIES L T P C
3 0 0 3

COURSE OBJECTIVES:

- To study the basic concepts of crypto currencies and blockchains.
- To explain the details of Bitcoin and its different components.
- To study the basics Hyper ledger and Web3.
- To analyze the position of Web 3 and Hyper ledger with different aspects of blockchain technologies.
- To differentiate between alternate blockchains and their advantages in application areas.
- To understand the Ethereum development environment and the application development process.

UNIT I INTRODUCTION**9**

Cryptographic hash functions - hash pointers - digital signatures - public keys as identities - an example cryptocurrency. Bitcoin, history of blockchain and Bitcoin - Types of Blockchain - Consensus - Decentralization.

UNIT II BITCOIN**9**

Bitcoin – Digital Keys and Addresses – Transactions, life cycle, data structure, types – Structure of the blockchain - Mining – Bitcoin Networks and Payments - Wallets - Alternative coins - Smart Contracts - Definition - Recardian contracts.

UNIT III WEB3 AND HYPERLEDGER**9**

Web 3 Contract development - POST requests - Frontend - Development framework – Hyperledger Projects - Protocol - Reference architecture - Hyperledger Fabric - Corda.

UNIT IV ALTERNATIVE BLOCKCHAINS AND APPLICATIONS**9**

Alternative blockchains - Applications, Internet of Things, Government, Health, Finance - Scalability - Privacy.

UNIT V ETHEREUM**9**

Setting up Ethereum development tools - Solidity language. - Ethereum accounts, key pairs, working with Externally Owned Accounts (EOA), contract accounts - Smart contracts, structure, setting up and interaction, examples - Decentralised applications, implementation, case studies Whisper protocol - Swarm architecture and concepts.

TOTAL : 45 PERIODS**COURSE OUTCOMES:**

On Completion of the course, the students should be able to:

- Explain cryptocurrencies and their relationship with the blockchain technology.
- Explain the different steps in the use of Bitcoins.
- Relate Web 3 and Hyperledger to concepts in blockchain technologies.
- Apply blockchains to different real-life problems
- Implement a simple application using Ethereum.

REFERENCES:

1. Imran Bashir, “Mastering Blockchain: Distributed Ledger Technology, Decentralization, and Smart Contracts Explained”, Second Edition, Packt Publishing, 2018.
2. A. Narayanan, J. Bonneau, E. Felten, A. Miller, S. Goldfeder, “Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction” Princeton University Press, 2016.
3. Arshdeep Bahga and Vijay Madisetti, "Blockchain Applications: A Hands-On Approach", 2017
4. Andreas Antonopoulos, Satoshi Nakamoto, “Mastering Bitcoin”, O’Reilly Publishing, 2014.
5. Roger Wattenhofer, “The Science of the Blockchain” Create Space Independent Publishing

Platform, 2016.

6. Alex Leverington, "Ethereum Programming" Packt Publishing Limited, 2017.

Mapping of CO with PO

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO-1	3					
CO-2	3		3	2		
CO-3	3	3	3	3	2	
CO-4		3	3	3		
CO-5	3	2	3	3		

JSE5006

AGILE SOFTWARE ENGINEERING

L T P C

3 0 0 3

COURSE OBJECTIVES:

The student should be able to

- Understand agile software development practices
- Demonstrate Agile development and testing techniques
- Know the benefits and pitfalls of working in an Agile team
- Understand agile development and testing.

UNIT I AGILE METHODOLOGY

9

Theories for Agile management – agile software development – traditional model vs. agile model -classification of agile methods – agile manifesto and principles – agile project management – agile team interactions – ethics in agile teams - agility in design, testing – agile documentations – agile drivers, capabilities and values.

UNIT II AGILE PROCESSES

9

Lean production - SCRUM, Crystal, Feature Driven Development, Adaptive Software Development, and Extreme Programming: Method overview – lifecycle – work products, roles and practices.

UNIT III AGILITY AND KNOWLEDGE MANAGEMENT

9

Agile information systems – agile decision making - Earls schools of KM – institutional knowledge evolution cycle – development, acquisition, refinement, distribution, deployment , leveraging – KM

in software engineering – managing software knowledge – challenges of migrating to agile methodologies – agile knowledge sharing – role of story-cards – Story-card Maturity Model (SMM).

UNIT IV AGILITY AND REQUIREMENTS ENGINEERING

9

Impact of agile processes in RE – current agile practices – variance – overview of RE using agile – managing unstable requirements – requirements elicitation – agile requirements abstraction model – requirements management in agile environment, agile requirements prioritization – agile requirements modeling and generation – concurrency in agile requirements generation.

UNIT V AGILITY AND QUALITY ASSURANCE

9

Agile Interaction Design - Agile product development – Agile Metrics – Feature Driven Development (FDD) – Financial and Production Metrics in FDD – Agile approach to Quality Assurance - Test Driven Development – Pair programming: Issues and Challenges - Agile approach to Global Software Development.

TOTAL : 45 PERIODS

COURSE OUTCOMES:

At the end of this course, the students should be able to:

- The know importance of interacting with business stakeholders in determining the requirements for a software system.
- Be an effective member of a scrum team
- Analyze existing problems with the team, development process and wider organization
- Apply iterative software development process
- Apply the impact of social aspects on software development success.

REFERENCES:

1. Craig Larman, —Agile and Iterative Development: A manager_s Guidel, Addison-Wesley, 2004
2. David J. Anderson; Eli Schragenheim, —Agile Management for Software Engineering: Applying the Theory of Constraints for Business Resultsl, Prentice Hall, 2003
3. Dingsoyr, Torgeir, Dyba, Tore, Moe, Nils Brede (Eds.), —Agile Software Development, Current Research and Future Directionsl, Springer-Verlag Berlin Heidelberg, 2010
4. Hazza& Dubinsky, “Agile Software Engineering, Series: Undergraduate Topics in Computer Science”, Springer, VIII edition, 2009
5. Kevin C. Desouza, “Agile information systems: conceptualization, construction, and Management”, Butterworth-Heinemann, 2007.

Mapping of CO with PO

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO-1	3					
CO-2	3		3	2		3
CO-3	3	3	3	3	2	3
CO-4		3	3	3		2
CO-5	3	2	3	3		3

JSE5007**DEEP LEARNING****L T P C
3 0 0 3****COURSE OBJECTIVES:**

- To understand the basic ideas and principles of Neural Networks
- To understand the basic concepts of Big Data and Statistical Data Analysis
- To familiarize the student with The Image Processing facilities like Tensorflow and Keras
- To appreciate the use of Deep Learning Applications
- To understand and implement Deep Learning Architectures

UNIT I BASICS OF NEURAL NETWORKS**9**

Basic concept of Neurons – Perceptron Algorithm – Feed Forward and Back Propagation Networks.

UNIT II INTRODUCTION TO DEEP LEARNING**9**

Feed Forward Neural Networks – Gradient Descent – Back Propagation Algorithm – Vanishing Gradient problem – Mitigation – ReLU Heuristics for Avoiding Bad Local Minima – Heuristics for Faster Training – Nestors Accelerated Gradient Descent – Regularization – Dropout.

UNIT III CONVOLUTIONAL NEURAL NETWORKS**9**

CNN Architectures – Convolution – Pooling Layers – Transfer Learning – Image Classification using Transfer Learning

UNIT IV MORE DEEP LEARNING ARCHITECTURES**9**

LSTM, GRU, Encoder/Decoder Architectures – Autoencoders – Standard- Sparse – Denoising – Contractive- Variational Autoencoders – Adversarial Generative Networks – Autoencoder and DBM

UNIT V APPLICATIONS OF DEEP LEARNING

9

Image Segmentation – Object Detection – Automatic Image Captioning – Image generation with Generative Adversarial Networks – Video to Text with LSTM Models – Attention Models for Computer Vision – Case Study: Named Entity Recognition – Opinion Mining using Recurrent Neural Networks – Parsing and Sentiment Analysis using Recursive Neural Networks – sentence Classification using Convolutional Neural Networks – Dialogue Generation with LSTMs.

Suggested Activities:

Model with Keras and TensorFlow.

TOTAL: 45 PERIODS

COURSE OUTCOMES:

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

- Understand the role of Deep learning in Machine Learning Applications.
- To design and implement Deep Learning Applications.
- Critically Analyse Different Deep Learning Models in Image Related Projects.
- To design and implement Convolutional Neural Networks.
- To know about applications of Deep Learning in NLP and Image Processing.

REFERENCES:

1. Ian Good Fellow, Yoshua Bengio, Aaron Courville, “Deep Learning”, MIT Press, 2017.
2. Francois Chollet, “Deep Learning with Python”, Manning Publications, 2018.
3. Phil Kim, “Matlab Deep Learning: With Machine Learning, Neural Networks and Artificial Intelligence”, Apress , 2017.
4. Ragav Venkatesan, Baoxin Li, “Convolutional Neural Networks in Visual Computing”, CRC Press, 2018.
5. Navin Kumar Manaswi, “Deep Learning with Applications Using Python”, Apress, 2018.
6. Joshua F. Wiley, “R Deep Learning Essentials”, Packt Publications, 2016.

Mapping of CO with PO

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO-1	3					3
CO-2	3		3	2		3
CO-3	3	3	3	3	2	3
CO-4		3	3	3		
CO-5	3	2	3	3		3

COURSE OBJECTIVES:

- To learn the fundamentals of semantic web and to conceptualize and depict Ontology for semantic web.
- To understand the languages for semantic web.
- To learn about the ontology learning algorithms and to utilize in the development of an application.
- To know the fundamental concepts of ontology management.
- To learn the applications related to semantic web.

UNIT I THE QUEST FOR SEMANTICS**9**

Building Models – Calculating with Knowledge – Exchanging Information – Semantic Web Technologies – Layers – Architecture – Components – Types – Ontological Commitments – Ontological Categories – Philosophical Background – Sample Knowledge Representation Ontologies – Top Level Ontologies – Linguistic Ontologies – Domain Ontologies – Semantic Web – Need – Foundation.

UNIT II LANGUAGES FOR SEMANTIC WEB AND ONTOLOGIES**9**

Web Documents in XML – RDF – Schema – Web Resource Description using RDF – RDF Properties – Topic Maps and RDF – Overview – Syntax Structure – Semantics – Pragmatics – Traditional Ontology Languages – LOOM – OKBC – OCML – Flogic Ontology Markup Languages – SHOE – OIL – DAML+OIL – OWL.

UNIT III ONTOLOGY LEARNING FOR SEMANTIC WEB**9**

Taxonomy for Ontology Learning – Layered Approach – Phases of Ontology Learning – Importing and Processing Ontologies and Documents – Ontology Learning Algorithms – Methods for evaluating Ontologies.

UNIT IV ONTOLOGY MANAGEMENT AND TOOLS**9**

Overview – Need for management – Development process – Target Ontology – Ontology mapping – Skills management system – Ontological class – Constraints – Issues – Evolution – Development of Tools and Tool Suites – Ontology Merge Tools – Ontology based Annotation Tools.

UNIT V APPLICATIONS**9**

Web Services – Semantic Web Services – Case Study for specific domain – Security issues – Web Data Exchange and Syndication – Semantic Wikis – Semantic Portals – Semantic Metadata in Data Formats – Semantic Web in Life Sciences – Ontologies for Standardizations – Rule Interchange Format.

Suggested Activities:

Use Protege ontology editor and framework for building systems.

TOTAL: 45 PERIODS

COURSE OUTCOMES:

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

- Create ontology for a given domain.
- Understand the concepts of semantic Web.
- Use ontology related tools and technologies for application creation.
- Design and develop applications using semantic web.
- Understand the standards related to semantic web.

REFERENCES:

1. Pascal Hitzler, Markus Krötzsch, Sebastian Rudolph, “Foundations of Semantic Web Technologies”, Chapman & Hall/CRC, 2009.
2. Asuncion Gomez-Perez, Oscar Corcho, Mariano Fernandez-Lopez, “Ontological Engineering: with Examples from the Areas of Knowledge Management, e-Commerce and the Semantic Web”, Springer, 2004.
3. Grigoris Antoniou, Frank van Harmelen, “A Semantic Web Primer (Cooperative Information Systems)”, MIT Press, 2004.
4. Alexander Maedche, “Ontology Learning for the Semantic Web”, First Edition, Springer. 2002.
5. John Davies, Dieter Fensel, Frank Van Harmelen, “Towards the Semantic Web: Ontology Driven Knowledge Management”, John Wiley, 2003.
6. John Davies, Rudi Studer, Paul Warren, (Editor), “Semantic Web Technologies: Trends and Research in Ontology-Based Systems”, Wiley, 2006.

Mapping of CO with PO

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO-1	3					
CO-2	3		3	2		
CO-3	3	3	3	3	2	
CO-4		3	3	3		
CO-5	3	2	3	3		

COURSE OBJECTIVES:

- To understand the nature of threats and cyber security management goals and technology
- To understand the landscape of hacking and perimeter defense mechanisms
- To develop strategies for cyber security and protecting critical infrastructure
- To understand policies to mitigate cyber risks
- To understand the IT Act, scheme, amendments and emerging cyber law and desired cyber ecosystem capabilities

UNIT I OVERVIEW OF CYBER SECURITY**9**

Introduction – Cyberspace – Cyber Crime – Nature of Threat – Cyber security – Policy, Mission and Vision of Cyber security Program. Cyber security management system – goals, technology categories – perimeter defense and encryption. Cyber security management framework.

UNIT II ATTACKS AND COUNTERMEASURES**9**

Malicious Attacks, Threats, and Vulnerabilities – Scope of cyber-attacks – Tools used to attack computer systems – security breach – Risks, vulnerabilities and threats. Malware – malicious software attack – social engineering attack – wireless network attack – web application attack – Countermeasures – Types of Network Security Devices – Firewalls, Intrusion Detection Systems, Content Filtering, Virtual Private Networks – Encryption

UNIT III STRATEGIES FOR CYBER SECURITY**9**

Creating a Secure Cyber, Types of Attacks, Comparison of Attacks, Creating an Assurance Framework, Encouraging Open Standards, Strengthening the Regulatory framework, Creating Mechanisms for IT Security, Securing E-Governance Services, and Protecting Critical Information Infrastructure. Areas for Intervention – Legal Responses – Harmonization of Legislation – Criminalization of Cyber Offences – National Security and issues related to Privacy and Freedom of Expression – Investigation Procedures – International Cooperation – Electronic Evidence – Liability of ISPs – Recommendations

UNIT IV POLICIES TO MITIGATE CYBER RISK**8**

Promotion of R&D in Cyber security – Reducing Supply Chain Risks – Mitigate Risks through Human Resource Development – Creating Cyber security Awareness – Information sharing – Implementing a Cyber security Framework. Signatures – Digital Signature, Electronic Signature

UNIT V CRITICAL INFORMATION INFRASTRUCTURE PROTECTION 10

National Security – Information Sharing and Coordination – Innovation in Regulatory Approach – Innovation in Security Programs – Proactive Threat and Vulnerability Management – Promoting Best Practices in Critical Infrastructure Sectors – Assessing and Monitoring Security Preparedness of Sectors (Security Index) – Security in Information Technology Supply Chain – Taking Leadership and Participating in International Efforts – Capacity Building in Security Skills and training and Awareness. The Indian Cyberspace– Cyber Threats – Need for a Comprehensive Cyber Security Policy – Need for a Nodal Authority – Need for an International Convention on Cyberspace – Cyber War – Fifth Domain of Warfare – Meeting the Cyber Warfare Challenges.

TOTAL: 45 PERIODS**COURSE OUTCOMES:**

- Gain knowledge on the nature of threats and cyber security management goals and framework
- Knowledge on the landscape of hacking and perimeter defense mechanisms
- Ability to differentiate and integrate strategies for cyber security and protecting critical infrastructure
- Able to understand policies to mitigate cyber risks
- Knowledge on IT Act, and amendments, copy rights, IPR and cyber law to deal with offenses.

REFERENCES:

1. David Kim and Michael G. Solomon, Fundamentals of Information Systems Security, Third Edition Transition Guide, Jones & Bartlett Learning, 2018.
2. Peter Trim and Yang - Im Lee, —Cyber Security Management- A Governance, Risk and Compliance Framework, Gower Publishing, England 2014.
3. Institute for Defence Studies and Analysis Report, India’s Cyber Security Challenge, 2012 https://idsa.in/system/files/book/book_indiacybersecurity.pdf
4. John G. Voeller, Cyber Security, John Wiley & Sons, England, 2014.
5. Carol C. Woody, Nancy R. Mead, Cyber Security Engineering: A Practical Approach for Systems and Software Assurance, Addison-Wesley, 2016.
6. Edward Griffor, Handbook of System Safety and Security, Syngress an Elsevier Publications, 1st Edition, 2017.
7. Thomas A. Johnson Cyber Security- Protecting Critical Infrastructures from Cyber Attack and Cyber Warfare, CRC Press, 2015.
8. NIST Cyber security Framework, Version 1.0, 2014.
9. CGI, —Cyber security in Modern Critical Infrastructure Environments, 2014.
10. Stuart Broderick J, Cyber Security Program, Cisco Security Solutions, June 2016.

Mapping of CO with PO

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO-1	3					3
CO-2	3		3	2		3

CO-3	3	3	3	3	2	3
CO-4		3	3	3		3
CO-5	3	2	3	3		3

JSE5010 INTEGRATED SOFTWARE PROJECT MANAGEMENT L T P C
3 0 0 3

COURSE OBJECTIVES:

- Understand the basic concept of project management.
- Learn the various costing and life cycle management.
- Understand the role played by risk in software project.
- Appreciate the use of metrics for software project management.
- Know the challenges in people management.

UNIT I PROJECT MANAGEMENT & COSTING 9

Software Project Management approaches – Project Acquisition – Initiation – Planning – PERT Execution and Control – CPM – Change Management – Project Closure – Agile SPM Problems in Software Estimation – Algorithmic Cost Estimation Process, Function Points, COCOMO II (Constructive Cost Model) – Estimating Web Application Development – Concepts of Finance, Activity Based Costing and Economic Value Added (EVA) – Balanced Score Card.

UNIT II PROCESS MODELS & LIFECYCLE MANAGEMENT 9

Software Engineering Process Models - Adaptive Software Development (ASD) - DSDM – SCRUM– Crystal -Feature Driven Development (FDD)–SPICE – SIX SIGMA – CMMI, SLIM (Software Life cycle Management) – PLM (Product Lifecycle Management) – PDM (Product Data Management) - PLM, PDM Applications – Pre-PLM Environment – Change Management.

UNIT III RISK MANAGEMENT 9

Perspectives of Risk Management - Risk Definition – Risk Categories – Risk Assessment: Approaches, techniques and good practices – Risk Identification / Analysis / Prioritization – Risk Control (Planning / Resolution / Monitoring) – Risk Retention – Risk Transfer - Failure Mode and Effects Analysis (FMEA) – Operational Risks – Supply Chain Risk Management.

UNITIV METRICS 9

Need for Software Metrics – scope – basics – framework for software measurement -Classification of Software Metrics: Product Metrics (Size Metrics, Complexity Metrics, Halstead’s Product Metrics, Quality Metrics), and Process metrics (Empirical Models, Statistical Models, Theory-based Models, Composite Models, and Reliability Models)– measuring internal and external product attributes.

UNIT V PEOPLE MANAGEMENT 9

Leadership styles – Developing Leadership skills – Leadership assessment – Motivating People – Organizational strategy – Management – Team building – Delegation – Art of Interviewing People

- Team Management – Rewarding - Client Relationship Management.

TOTAL: 45 PERIODS

COURSE OUTCOMES:

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

- Identify the various elements of software management process framework.
- Use available open source estimation tools for cost estimation.
- Identify existing risk and perform risk assessment.
- Design a software metric for software project management.
- Modify the art of interviewing people for a given scenario.

REFERENCES:

1. Bob Hughes, Mike Cotterell, “Software Project Management”, Fifth edition, The McGraw – Hill Companies, 2009.
2. Pankaj Jalote, “Software Project Management in Practice”, Eight Impression, Pearson Publication, 2009.
3. Robert K. Wysocki, “Effective Software Project Management”, John Wiley and sons Publications, 2010.
4. Murali Chmuturi, Thomas M. Cagley, “Mastering Software Project Management: Best Practices, Tools and Techniques”, J. Ross Publishing, 2010.
5. Richard E. (DICK) Fairley, “Managing and Leading Software Projects”, John Wiley and Sons Publications, 2009.

Mapping of CO with PO

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO-1	3					3
CO-2	3		3	2		3
CO-3	3	3	3	3	2	3
CO-4		3	3	3		3
CO-5	3	2	3	3		3

JSE5011

WEB CONTENT DESIGN AND MANAGEMENT

L T P C

3 0 0 3

COURSE OBJECTIVES:

Understand the design principles and interaction.

- To explore the detailed design practices, standards.
- To gain an insight into Content Management System for content design.
- To use any Content Management System tool for better content management.
- To get familiarized with Web Analytics for better management.

UNIT I PRINCIPLES OF WEB DESIGN**9**

User Centered Design, Web Medium, Information Architectures, Site Types and Architectures, Page Structure, Site Maps, Navigation, Search, Web Design Process, Designing for Multiple Screen Resolutions, Creating a Unified Site Design, Evaluating Web Sites.

UNIT II ELEMENTS OF PAGE DESIGN**9**

Elements of Page Design, Adding Styles With CSS, Pages and Layout, Typography, Color, Images, GUI Widgets And Forms, Responsive Web Designs, User Input Forms, Working With Data Tables, Web Standards and Styles.

UNIT III WEB CONTENT DESIGN**9**

Features – Automated Templates – Template Processor – Front Controller Pattern – Content Modeling – Content Aggregation – Plug-Ins – Search Engine Optimization – Recommended Usage of Tools – WORDPRESS

UNIT IV WEB CONTENT MANAGEMENT**9**

Work Flow Management – Document Management – Collaboration – Versioning – Recommended Usage of Tools – WORDPRESS

UNIT V WEB ANALYTICS**9**

Web Analytics Process – Data Collection – Qualitative Analysis – Log File Analysis – Page Tagging – Hybrid Methods – Click Analytics – Onsite and Offsite Analytics – Web Analytics Methods

TOTAL : 45 PERIODS**COURSE OUTCOMES:**

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

- Design web pages that follow standards and are usable.
- Design web sites that are appealing.
- To be able to use Content management System for designing web Content.
- To take advantage of Content Management System tools for managing content for large web sites.
- To be able to use analytics tools for better management.

REFERENCES:

1. Patrich J. Lynch, Sarah Horton, “Web Style Guide-Foundations of User ExperienceDesign”, Yale University Press, 4th Edition, 2016.
2. Thomas A. Powell, “The Complete Reference– Web Design”, Tata McGraw Hill,Second Edition, 2003.
3. Joel Sklar, “Principles of Web Design, Cengage Learning”, Web Warrior Series, 6th Edition,2015.
4. Deane Barker, “Web Content management-Systems, Features and Best Practices”, O’reilly Media, 1st Edition, 2016.
5. Brian Clifton, “Advanced web Metrics with Google Analytics”, Third Edition, Sybex Publishers, 2012.

6. AvinashKaushik, “Web Analytics 2.0: The Art of Online Accountability and Science of Customer Centricity”, 1st edition, Sybex publishers, 2009.

Mapping of CO with PO

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO-1	3					3
CO-2	3		3	2		3
CO-3	3	3	3	3	2	3
CO-4		3	3	3		3
CO-5	3	2	3	3		3

JSE5012

ETHICAL HACKING

**L T P
C**

3 0 0 3

COURSE OBJECTIVES:

- To learn about the importance of information security.
- To learn different scanning and enumeration methodologies and tools.
- To understand various hacking techniques and attacks.
- To be exposed to programming languages for security professionals.
- To understand the different phases in penetration testing.

UNIT I INTRODUCTION TO HACKING

9

Introduction to Hacking – Importance of Security – Elements of Security – Phases of an Attack – Types of Hacker Attacks – Hacktivism – Vulnerability Research – Introduction to Footprinting – Information Gathering Methodology – Footprinting Tools – WHOIS Tools – DNS Information Tools – Locating the Network Range – Meta Search Engines

UNIT II SCANNING AND ENUMERATION

9

Introduction to Scanning – Objectives – Scanning Methodology – Tools – Introduction to Enumeration – Enumeration Techniques – Enumeration Procedure – Tools

UNIT III SYSTEM HACKING

9

Introduction – Cracking Passwords – Password Cracking Websites – Password Guessing – Password Cracking Tools – Password Cracking Countermeasures – Escalating Privileges – Executing Applications – Keyloggers and Spyware

UNIT IV PROGRAMMING FOR SECURITY PROFESSIONALS

9

Programming Fundamentals – C language – HTML – Perl – Windows OS Vulnerabilities – Tools

for Identifying Vulnerabilities – Countermeasures – Linux OS Vulnerabilities – Tools for Identifying Vulnerabilities – Countermeasures

UNIT V PENETRATION TESTING

9

Introduction – Security Assessments – Types of Penetration Testing- Phases of Penetration Testing– Tools – Choosing Different Types of Pen-Test Tools – Penetration Testing Tools

TOTAL: 45 PERIODS

COURSE OUTCOMES:

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

- Identify threats to computers.
- Defend hacking attacks.
- Protect data assets.
- Defend a computer against a variety of security attacks using various tools.
- Practice and use safe techniques on the World Wide Web.

REFERENCES:

1. EC-Council, “Ethical Hacking and Countermeasures: Attack Phases”, Cengage Learning, 2010.
2. Jon Erickson, “Hacking, 2nd Edition: The Art of Exploitation”, No Starch Press Inc., 2008.
3. Michael T. Simpson, Kent Backman, James E. Corley, “Hands-On Ethical Hacking and Network Defense”, Cengage Learning, 2013.
4. Patrick Engebretson, “The Basics of Hacking and Penetration Testing – Ethical Hacking and Penetration Testing Made Easy”, Second Edition, Elsevier, 2013.
5. RafayBoloach, “Ethical Hacking and Penetration Testing Guide”, CRC Press, 2014.

Mapping of CO with PO

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO-1	3					3
CO-2	3		3	2		3
CO-3	3	3	3	3	2	3
CO-4		3	3	3		3
CO-5	3	2	3	3		3

JSE5013

GPU COMPUTING

L T P C

3 0 0 3

COURSE OBJECTIVES

- To study architecture and capabilities of modern GPUs
- To learn programming techniques for the GPU such as CUDA programming model.
- To introduce CUDA for parallel computing on the Graphics Processing Unit(GPU)
- To introduce OpenCL
- To introduce knowledge about parallel programming

UNIT 1 INTRODUCTION

9

History, GPU Architecture, Clock speeds, CPU / GPU comparisons, Heterogeneity, Accelerators, Parallel Programming, CUDA OpenCL / OpenACC, Kernels Launch parameters, Thread hierarchy, Warps/Wavefronts, Threadblocks/Workgroups, Streaming multiprocessors, 1D/2D/3D thread mapping, Device properties, Simple Programs

UNIT II MEMORY

9

Memory hierarchy, DRAM / global, local / shared, private / local, textures, Constant Memory, Pointers, Parameter Passing, Arrays and dynamic Memory, Multi-dimensional Arrays, Memory Allocation, Memory copying across devices, Programs with matrices, Performance evaluation with different memories

UNIT III SYNCHRONIZATION

9

Memory Consistency, Barriers (local versus global), Atomics, Memory fence. Prefix sum, Reduction. Programs for concurrent Data Structures such as Worklists, Linked-lists. Synchronization across CPU and GPU Functions: Device functions, Host functions, Kernels functions, Using libraries (such as Thrust), and developing libraries.

UNIT IV SUPPORT

9

Debugging GPU Programs. Profiling, Profile tools, Performance aspects Streams: Asynchronous processing, tasks, Task-dependence, Overlapped data transfers, Default Stream, Synchronization with streams. Events, Event-based Synchronization - Overlapping data transfer and kernel execution, pitfalls.

Image Processing, Graph algorithms, Simulations, Deep Learning ,Advanced topics: Dynamic parallelism, Unified Virtual Memory, Multi-GPU processing, Peer access, Heterogeneous processing

TEXT BOOKS	
<p>T1 Programming Massively Parallel Processors: A Hands-on Approach; David Kirk, Wen-mei Hwu; Morgan Kaufman; 2010 (ISBN: 978-0123814722)</p> <p>T2. Sanders, J. and Kandrot, E., CUDA by Example: An Introduction to General-Purpose GPU Programming, Addison-Wesley Professional (2012) 4th Edition.</p> <p>T3. Kirk, D. and Hwu, M., W., Programming Massively Parallel Processors: A Hands-on Approach. Morgan Kaufmann (2016) 3rd Edition.</p>	
REFERENCES	
<p>1. CUDA Programming: A Developer's Guide to Parallel Computing with GPUs; Shane Cook; Morgan Kaufman; 2012 (ISBN: 978-0124159334)</p> <p>2. Hwu, M., W., A GPU Computing Gems Emerald Edition (Applications of GPU Computing Series), Morgan Kaufmann (2011) 1st Edition.</p>	
WEBSITE REFERENCES	
<p>1. https://onlinecourses.nptel.ac.in/noc20_cs41/preview</p> <p>2. http://www.cse.iitm.ac.in/course_details.php?arg=MTIw</p> <p>3. http://www.cs.fsu.edu/~asriniva/courses/GPU15/</p>	

Mapping of CO with PO

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO-1	3					
CO-2	3		3	2		
CO-3	3	3	3	3	2	
CO-4		3	3	3		
CO-5	3	2	3	3		

JSE5014	INFORMATION STORAGE MANAGEMENT	L	T	P	C
		3	0	0	3

COURSE OBJECTIVES

- To understand the basic components of Storage System Environment.
- To understand the Storage Area Network Characteristics and Components.
- To examine emerging technologies including IP-SAN.
- To describe the different backup and recovery topologies and their role in providing disaster recovery and business continuity capabilities.
- To understand the local and remote replication technologies.

UNIT-I STORAGE SYSTEMS 9

Introduction to Information Storage and Management: Information Storage, Evolution of Storage Technology and Architecture, Data Center Infrastructure, Key Challenges in Managing Information, Information Lifecycle. Storage System Environment: Components of the Host. RAID: Implementation of RAID, RAID Array Components, RAID Levels, RAID Comparison, RAID Impact on Disk Performance, Hot Spares. Intelligent Storage System: Components, Intelligent Storage Array.

UNIT-II STORAGE NETWORKING TECHNOLOGIES 9

Direct-Attached Storage and Introduction to SCSI: Types of DAS, DAS Benefits and Limitations, Disk Drive Interfaces, Introduction to Parallel SCSI, SCSI Command Model. Storage Area Networks: Fiber Channel, SAN Evolution, SAN Components, Fiber Channel Connectivity, Fiber Channel Ports, Fiber Channel Architecture, Zoning, Fiber Channel Login Types, Fiber Channel Topologies. Network Attached Storage: Benefits of NAS, NAS File I/Components of NAS, NAS Implementations, NAS-Implementations, NAS File Sharing Protocols, NAS I/O Operations.

UNIT-III ADVANCED STORAGE NETWORKING AND VIRTUALIZATION 9

IP SAN: iSCSI, FCIP. Content-Addressed Storage: Fixed Content and Archives, Types of Archives, Features and Benefits of CAS, CAS Architecture, Object Storage and Retrieval in CAS, CAS Examples. Storage Virtualization: Forms of Virtualization, NIA Storage Virtualization Taxonomy, Storage Virtualization Configurations, Storage Virtualization Challenges, Types of Storage Virtualization.

UNIT-IV BUSINESS CONTINUITY

9

Introduction to Business Continuity: Information Availability, BC Terminology, BC Planning Lifecycle, Failure Analysis, Business Impact Analysis, BC Technology Solutions. Backup and Recovery: Backup Purpose, Considerations, Granularity, Recovery Considerations, Backup Methods and Process, Backup and Restore Operations, Backup Topologies, Backup in NAS Environments, Backup Technologies.

UNIT-V REPLICATION

9

Local Replication: Source and Target, Uses of Local Replicas, Data Consistency, Local Replication Technologies, Restore and Restart Considerations, Creating Multiple Replicas, Management Interface. Remote Replication: Modes of Remote Replication and its Technologies, Network Infrastructure.

TOTAL: 45 PERIODS

COURSE OUTCOMES

Upon completion of the course, the students will be able to:

- Understand the logical and physical components of a Storage infrastructure.
- Evaluate storage architectures, including storage subsystems, DAS, SAN, NAS, and CAS.
- Understand the various forms and types of Storage Virtualization.
- Describe the different role in providing disaster recovery and business continuity capabilities.
- Distinguish different remote replication technologies.

TEXT BOOKS	
1. EMC Corporation, Information Storage and Management, Wiley, India.	
REFERENCES	
1. Robert Spalding, "Storage Networks: The Complete Reference", Tata McGraw Hill, Osborne, 2003. 2. Marc Farley, "Building Storage Networks", Tata McGraw Hill, Osborne, 2001. 3. Meeta Gupta, Storage Area Networks Fundamentals, Pearson Education Limited, 2002.	
WEBSITE REFERENCES	

1. <https://www.youtube.com/watch?v=E3BY2yYQAdOT2>
2. <https://csenotescorner.blogspot.com/search/label/ISM%20Notes>
3. <https://www.youtube.com/watch?v=mZvmtZA2YH4>
4. <https://www.slideshare.net/AkashBadone/information-storage-and-management>
5. <https://www.slideserve.com/jolie-williams/introduction-to-information-storage-and-management>

Mapping of CO with PO

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO-1	3					
CO-2	3		3	2		3
CO-3	3	3	3	3	2	3
CO-4		3	3	3		3
CO-5	3	2	3	3		

		L	T	P	C
JSE5015	SOFTWARE SECURITY	3	0	0	3

COURSE OBJECTIVES

- To understand the concept of Security Principles.
- To become familiar on memory and type safety
- To learn about attack vectors
- To gain knowledge on Risk Management
- To study the concept of Software Penetration Testing

UNIT-I Introduction

9

Software and System Security Principles:Confidentiality, Integrity, and Availability-Isolation-Least Privilege-Compartmentalization-Threat Model-Bug versus Vulnerability- Secure Software Life Cycle:Software Design-Software Implementation-Software Testing-Continuous Updates and Patches-Modern Software Engineering

UNIT-II Memory and Type Safety

9

Introduction -Pointer Capabilities-Memory Safety types:Spatial Memory Safety,Temporal Memory Safety,A Definition of Memory Safety, Practical Memory Safety-Type Safety

1. Mathias Payer, "Software Security Principles, Policies, and Protection" July 2021,
2. McGraw, G.; , "Software security," Security & Privacy, IEEE , vol.2, no.2, pp. 80- 83, Mar-Apr 2004doi: 10.1109/MSECP.2004.1281254 URL: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=1281254&isnumber=28622>
3. Haley, C.B.; Laney, R.; Moffett, J.D.; Nuseibeh, B.; , "Security Requirements Engineering: A Framework for Representation and Analysis," Software Engineering, IEEE Transactions on , vol.34, no.1, pp.133-153, Jan.-Feb. 2008.

WEBSITE REFERENCES

1. <https://www.tutorialspoint.com/principles-of-information-system-security>
2. <http://www.securingjava.com/chapter-two/chapter-two-10.html>
3. <https://www.techtarget.com/searchsecurity/definition/denial-of-service>
4. <https://www.techtarget.com/searchcio/definition/Risk-Management-Framework-RMF#:~:text=The%20Risk%20Management%20Framework%20is,of%20the%20United%20States%20government.>
5. <https://www.getastra.com/blog/security-audit/software-penetration-testing/>

Mapping of CO with PO

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO-1	3					
CO-2	3		3	2		3
CO-3	3	3	3	3	2	3
CO-4		3	3	3		
CO-5	3	2	3	3		

JSE5016	ENTERPRISE APPLICATION	L	T	P	C
		3	0	0	3

COURSE OBJECTIVES

- Learn the Business Enterprise modeling
- Learn requirements validation, planning and estimation
- Learn the Architecture of business application

- Learn about building a business software.
- Learn different testing involved with enterprise application and the process of rolling out an enterprise application.

UNIT-I INTRODUCTION 9

Introduction to enterprise applications and their types, software engineering methodologies, life cycle of raising an enterprise application, introduction to skills required to build an enterprise application, key determinants of successful enterprise applications, and measuring the success of enterprise applications.

UNIT-II MODELLING 9

Inception of enterprise applications, enterprise analysis, business modeling, requirements elicitation, use case modelling, prototyping, non-functional requirements, requirements validation, planning and estimation.

UNIT-III ARCHITECTURE 9

Concept of architecture, views and viewpoints, enterprise architecture, logical architecture, technical architecture - design, different technical layers, best practices, data architecture and design - relational, XML, and other structured data representations, Infrastructure architecture and design elements - Networking, Internetworking, and Communication Protocols, IT Hardware and Software, Middleware, Policies for Infrastructure Management, Deployment Strategy, Documentation of application architecture and design.

UNIT-IV ENTERPRISE APPLICATION I 9

Construction readiness of enterprise applications - defining a construction plan, defining a package structure, setting up a configuration management plan, setting up a development environment, introduction to the concept of Software Construction Maps, construction of technical solutions layers, methodologies of code review, static code analysis, build and testing, dynamic code analysis - code profiling and code coverage.

UNIT-V ENTERPRISE APPLICATION II 9

Types and methods of testing an enterprise application, testing levels and approaches, testing environments, integration testing, performance testing, penetration testing,

usability testing, globalization testing and interface testing, user acceptance testing, rolling out an enterprise application.

TOTAL: 45 PERIODS

COURSE OUTCOMES

Upon completion of the course, the students will be able to:

- Familiarize with concept of Enterprise Analysis and Business Modeling.
- Write requirements validation, planning and estimation.
- Design and document the application architecture.
- Construct and develop different solution layers.
- Implement different testing involved with enterprise application and the process of rolling out an enterprise application.

TEXT BOOKS	
<ol style="list-style-type: none"> 1. Raising Enterprise Applications - Published by John Wiley, authored by Anubhav Pradhan, Satheesha B. Nanjappa, Senthil K. Nallasamy, Veerakumar Esakimuthu. 2. Building Java Enterprise Applications - Published by O'Reilly Media, authored by Brett McLaughlin. 	
REFERENCES	
<ol style="list-style-type: none"> 1. Software Requirements: Styles & Techniques - published by Addison-Wesley Professional. 2. Software Systems Requirements Engineering: In Practice - published by McGraw-Hill/Osborne Media. 3. Designing Enterprise Applications with the J2EE Platform (PDF available at- http://java.sun.com/blueprints/guidelines/designing_enterprise_applications_2e/) 4. Software Testing, Principles and Practices - published by Oxford University Press. 	
WEBSITE REFERENCES	
<ol style="list-style-type: none"> 1. https://anyconnector.com/en/enterprise-application.html 2. https://bdtechtalks.com/2019/07/03/what-is-enterprise-application/ 3. https://www.bmc.com/blogs/enterprise-application-software-defined-how-is-it-different-from-other-software/ 4. https://www.youtube.com/watch?v=d9R5ni48uOU 5. https://www.tcs.com/enterprise-applications-services 	

Mapping of CO with PO

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO-1	3					
CO-2	3		3	2		2
CO-3	3	3	3	3	2	3
CO-4		3	3	3		
CO-5	3	2	3	3		

	SOFTWARE	L	T	P	C
	RELIABILITY METRICS				
JSE5017	AND MODELS	3	0	0	3

COURSE OBJECTIVES

- To gain basic knowledge about metrics and related terminologies
- To be familiar to measure the quality level of internal and external attributes of the software product
- To introduce the basics of software reliability and to illustrate how to perform planning, executing and testing for software reliability
- To explore various metrics and models of software reliability
- To compare various models of software reliability based on its application

UNIT-I SOFTWARE OVERVIEW AND BASICS 9

Overview-What is software Quality?-Quality Assurance-Quality Assurance in Context-Quality Engineering.

UNIT-II SOFTWARE TESTING 9

Testing: Concepts, Issues, and Techniques- Test Activities, Management, and Automation- Coverage and Usage Testing Based on Checklists and Partitions- Input Domain Partitioning and Boundary Testing- Input Domain Partitioning and Boundary Testing.

UNIT-III QUALITY ASSURANCE BEYOND TESTING 9

3 Defect Prevention and Process improvement- Software Inspection- Formal Verification- Fault Tolerance and Failure Containment- Quality Models and Measurements- Defect Classification and Analysis -Risk Identification for Quantifiable Quality Improvement

UNIT-IV MEASUREMENT AND EXPERIMENTATION 9

Definition of Measurement –The Basics of measurement-Framework of software measurement-Empirical investigation-Software metrics data collection

UNIT-V SOFTWARE ENGINEERING MEASUREMENT 9

Measuring internal product attributes: Size and Structure-Measuring external product attributes-Software reliability: Measurement and prediction-Resource Measurement-Making Process prediction

TOTAL: 45 PERIODS

COURSE OUTCOMES

Upon completion of the course, the students will be able to:

- Identify and apply various software metrics, which determines the quality level of software
- Identify and evaluate the quality level of internal and external attributes of the software product
- Compare and Pick out the right reliability model for evaluating the software
- Evaluate the reliability of any given software product
- Design new metrics and reliability models for evaluating the quality level of the softwarebased on the requirement.

TEXT BOOKS	
<ol style="list-style-type: none"> 1. Norman E-Fentor and Share Lawrence Pflieger.” Software Metrics”. International Thomson Computer Press, 1997. 2. Stephen H Khan: Metrics and Models in Software Quality Engineering, Pearson 2nd edition 2013. 	
REFERENCES	
<ol style="list-style-type: none"> 1. Mary Beth Chrissis, Mike Konrad and Sandy Shrum, “CMMI”, Pearson Education (Singapore) Pvt. Ltd., 2003 2. Philip B Crosby, ” Quality is Free: The Art of Making Quality Certain “, Mass Market, 1992. 3. S.A. Kelkar, “Software quality and Testing, PHI Learning, Pvt., Ltd.,New Delhi 2012. 4. Watts S Humphrey, “Managing the Software Process”, Pearson Education Inc, 2008. 	
WEBSITE REFERENCES	
<ol style="list-style-type: none"> 1. https://www.researchgate.net/publication/269287977_Metrics_and_Mode_Mo_for_Software_Reliability_A_Systematic_Review 2. https://link.springer.com/chapter/10.1007/11424857_81 	

Mapping of CO with PO

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO-1	3					2
CO-2	3		3	2		3
CO-3	3	3	3	3	2	
CO-4		3	3	3		2
CO-5	3	2	3	3		3

JSE5018	SOFTWARE TEST AUTOMATION	L	T	P	C
		3	0	0	3

COURSE OBJECTIVES

- To understand the basics of test automation
- To appreciate the different aspects of test tool evaluation and test automation approachselection
- To understand the role played by test planning and design in test execution
- To appreciate the use of various testing tools for testing varied applications
- To understand test automation using case studies

UNIT-I INTRODUCTION 9

Fundamentals of test automation – Management issues – technical issues - Background on software testing – Automated test life cycle methodology (ATLM) – Test Maturity Model – Test Automation Development – Overcoming false expectations of automated testing – benefits – test tool proposal.

UNIT-II TEST FRAMEWORK AND AUTOMATION 9

Test Tool Evaluation and selection – organisations ‘system engineering environment – tools that support the testing life cycle – test process analysis – test tool consideration Test framework – Test Library Management –selecting the test automation approach - test team management.

UNIT-III SYSTEM DEVELOPMENT 9

Test planning – Test program scope – Test requirements management – Test Events, Activities and Documentation – Test Environment – Evolving a Test plan Test analysis and design – Test requirements analysis – Test program design – Test procedure design – Test development architecture – guidelines – automation infrastructure – test execution and review – test metrics.

UNIT-IV TESTING THE APPLICATIONS 9

Testing Web Applications – Functional Web testing with Twill – Selenium – Testing a simple Web Application – Testing Mobile Smartphone Applications – Running automated test scripts – Test tools for Browser based applications – Test Automation with Emulators.

UNIT-V CASE STUDIES 9

Test automation and agile project management – database automation – test automation in cloud – Mainframe and Framework automation – Model based test case generation – Model based testing of Android applications – exploratory test automation

TOTAL: 45 PERIODS

COURSE OUTCOMES

Upon completion of the course, the students will be able to:

- Identify the different test tool proposal ways

- Use available testing tools to test some software applications
- Modify existing test metrics based on functionality or features used
- Design test cases and execute them
- Implement test scripts that can be used for automating test execution of an application of your choice

TEXT BOOKS	
<ol style="list-style-type: none"> 1. Elfriede Dustin, Jeff Rashka, Automated software testing: Introduction, Management and Performance, Pearson Education, 2008. 2. C. Titus Brown, Gheorghe Gheorghiu, Jason Huggins, An Introduction to Testing Web Applications with twill and Selenium, O'Reilly Media, Inc., 2007 	
REFERENCES	
<ol style="list-style-type: none"> 1. Dorothy Graham, Mark Fewster, Experiences of Test Automation: Case Studies of Software Test Automation, illustrated Edition, Addison-Wesley Professional, 2012. 2. Kanglin Li, Mengqi Wu, Effective Software Test Automation: Developing an Automated Software Testing Tool, John Wiley & Sons, 2006 3. Linda Hayes, The Automated Testing Handbook, Software testing Inst., 1995. 4. Julian Harty, A Practical Guide to Testing Mobile Smartphone Applications, Vol. 6 of Synthesis Lectures on Mobile and Pervasive Computing Series, Morgan & Claypool Publishers, 2009. 5. Mark Fewster & Dorothy Graham, Software Test Automation, Addison Wesley, 1999. 	
WEBSITE REFERENCES	
<ol style="list-style-type: none"> 1. https://nptel.ac.in/courses/101/104/106101163/ 2. https://www.informit.com/articles/article.aspx?p=21468 3. https://www.coursera.org/specializations/software-testing-automation 4. https://www.tutorialspoint.com/stlc/stlc_requirement_analysis.htm 5. https://www.edx.org/course/automated-software-testing-model-and-state-based-2 	

Mapping of CO with PO

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO-1	3				2	
CO-2	3		3	2	3	
CO-3	3	3	3	3	2	2

Introducton - media preparations - media composition-media integration-media communication- media consumption- media entertainment - case study

TOTAL: 45 PERIODS

COURSE OUTCOMES

Upon completion of the course, the students will be able to:

- 1 To study the fundamentals of multimedia systems
- 2 To understand the evolution, latest trends, and state-of-the-art in multimediatechnology
- 3To familarise the Features about Creative Industries

- 4 To understand the basic concepts related to multimedia communication
- 5 To understand the technologies behind multimedia applications
-

TEXT BOOKS	
<ol style="list-style-type: none"> 1. T1Multimedia Computing, Communication and application” By Steinmetz and Klara Nahrstedt. 2. T2. S.Gokul, “Multimedia Magic”, BPB Publications, 2nd Edition. 3. T3 Tay Vaughen , “Multimedia Making it Work”, TMH, 6th Edition. 	
REFERENCES	
<ol style="list-style-type: none"> 1. T1”“Multimedia Systems, Standards, and Networks” By Atul Puri, Tsuhan Chen 2. T2“Multimedia Storage and Retrieval: An Algorithmic Approach” By Jan Korst, Verus Pronk 3. T3 Kiran Thakrar, Prabhat k.andleigh, “Multimedia System Design”, Prentice Hall India. 	
WEBSITE REFERENCES	
<ol style="list-style-type: none"> 1. T1NPTEL & MOOC courses titled Multi media 2. T2W3schools.com/html/html-media.asp 3. T3https://nptel.ac.in/courses/106105163/ 	

Mapping of CO with PO

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO-1	3					2
CO-2	3		3	2		3
CO-3	3	3	3	3	2	2
CO-4		3	3	3		3
CO-5	3	2	3	3		

			L	T	P	C
JSE5020	Software Verification and Validation		3	0	0	3

COURSE OBJECTIVES

The student should be able to

- Understand the principles of verification and validation
- Appreciate the different verification and validation techniques
- Understand the various stages of testing
- Appreciate the use of tools for verification and validation
- Appreciate the benefits of using metrics for verification and validation

UNIT I INTRODUCTION 9

Principles of verification and validation – software architecture frameworks – model driven architecture – UML – systems modeling language – verification, validation and accreditation.

UNIT II METHODS OF SOFTWARE VERIFICATION 9

Verification and validation life cycle – traceability analysis – interface analysis – design and code verification – test analysis - Reviews – inspections - walkthroughs – audits – tracing – formal proofs – Model based verification and validation - Program verification techniques – formal methods of software verification – clean room methods.

UNIT III TESTING 9

Stages of Testing: Test Planning – Test design – Test case definition – Test procedure – Test reporting – Unit testing: white box, black box and performance testing – system testing: Function, performance, interface, operations, resource, security, portability, reliability, maintainability, safety, regression and stress testing – integration testing – acceptance testing: capability, constraint

testing- structured testing – structured integration testing

UNITIV TOOLS FOR SOFTWAREVERIFICATION

9

Tools for verification and validation: static analyser – configuration management tools – reverse engineering tools – tracing tools – tools for formal analysis – tools for testing – test case generators – test harnesses – debuggers – coverage analysers – performance analysers – test management tools

UNITV ADVANCEDAPPROACHES

9

Automatic approach for verification and validation – validating UML behavioral diagrams – probabilistic model checking of activity diagrams in SysML – metrics for verification and validation.

TOTAL: 45PERIODS

COURSE OUTCOMES

Upon completion of the course, the students will be able to:

- Identify the different techniques for verification and validation
- Use available traceability analysis tools on sample requirements
- Modify existing coverage analysers in terms of functionality or features used
- Design system test cases
- Use test case generators and test management tools

REFERENCES
<ol style="list-style-type: none"> 1. AvnerEngel, Verification, Validation & Testing of Engineered Systems, Wileyseriesin systems Engineering and Management,2010. 2. ESA Board for Software Standardisation and Control(BSSC),—Guide to software verification and Validationl, European Space Agency ESA PSS-05-10 Issue 1 Revision 1,1995 3. MarcusS.Fisher,—SoftwareVerificationandValidation:AnEngineeringandScientific Approach, Springer, 2007 4. MouradDebbabi,HassaineF,JarryaY.,SoeanuA.,AlawnehL.,—VerificationandValid ationin Systems Engineering, Springer, 2010

Mapping of CO with PO

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO-1	3					2
CO-2	3		3	2		3
CO-3	3	3	3	3	2	2
CO-4		3	3	3		
CO-5	3	2	3	3		

		L	T	P	C
JSE5021	SOFTWARE AGENTS	3	0	0	3

COURSE OBJECTIVES

- Have an overview of the agent systems and software agents.
- Understand the basic concepts of intelligent software agents.
- Design and build a multiagent system.
- Have a basic understanding about software agent technology and to be familiar with some of the communicating languages, standardization and applications.
- Learn the use of software agents to represent and share information to coordinate activities of the agents for the purpose of group problem solving.

UNIT I AGENTS – OVERVIEW 9

Agent Definition – Agent Programming Paradigms – Agent Vs Object – Aglet – Mobile Agents – Agent Frameworks – Agent Reasoning.

UNIT II JAVA AGENTS 9

Processes – Threads – Daemons – Components – Java Beans – ActiveX – Sockets – RPCs – Distributed Computing – Aglets Programming – Jini Architecture – Actors and Agents – Typed and proactive messages.

UNIT III MULTIAGENT SYSTEMS 9

Interaction between agents – Reactive Agents – Cognitive Agents – Interaction protocols – Agent coordination – Agent negotiation – Agent Cooperation – Agent Organization – Self-Interested agents in Electronic Commerce Applications.

UNIT IV INTELLIGENT SOFTWARE AGENTS 9

Interface Agents – Agent Communication Languages – Agent Knowledge Representation – Agent Adaptability – Belief Desire Intension – Mobile Agent Applications.

UNIT V AGENTS AND SECURITY 9

Agent Security Issues – Mobile Agents Security – Protecting Agents against Malicious Hosts – Untrusted Agent – Black Box Security – Authentication for agents – Security issues for Aglets.

TOTAL: 45 PERIODS

COURSE OUTCOMES

Upon completion of the course, the students will be able to:

- Create / develop an agent based system for a particular task.
- Design an application that uses different security issues for intelligent agents.
- Effectively apply agent-based technologies in the development
- Be familiar with application of distributed information systems that use software agents.
- Design an application that uses different Mobile Agent Applications.

REFERENCES	
<ol style="list-style-type: none"> 1. Bigus&Bigus, " Constructing Intelligent agents with Java ", Wiley, 1997 2. Bradshaw, " Software Agents ", MIT Press, 2010 3. Gerhard Weiss, —Multi Agent Systems – A Modern Approach to Distributed Artificial Intelligence, MIT Press, 2000. 4. Richard Murch, Tony Johnson, "Intelligent Software Agents", Prentice Hall, 2000 5. Russel, Norvig, "Artificial Intelligence: A Modern Approach", Second Edition, Pearson Education, 2003 	
WEBSITE REFERENCES	
<ol style="list-style-type: none"> 1. https://www.researchgate.net/publication/269287977_Metrics_and_ModeMo_for_Software_Reliability_A_Systematic_Review 2. https://link.springer.com/chapter/10.1007/11424857_81 	

Mapping of CO with PO

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO-1	3					2
CO-2	3		3	2		3
CO-3	3	3	3	3	2	2
CO-4		3	3	3		3
CO-5	3	2	3	3		2

JSE5022

OPERATING SYSTEMS
SECURITY

L	T	P	C
3	0	0	3

COURSE OBJECTIVES:

- Study the basic concepts and functions of operating systems.
- Understand the structure and functions of OS.
- Learn about Processes and memory management schemes.
- Study I/O management and File systems.
- To gain insight on to the Protection, Security issues

UNIT I FUNDAMENTALS OF OPERATING SYSTEMS 9

Overview – Operating system concepts – Functions – Structure of Operating system – Types of operating system – Dead lock Prevention, Recovery, Detection and Avoidance

UNIT II PROCESS MANAGEMENT 9

Introduction to processes – Process Scheduling - Threads-C PU Scheduling objectives, criteria –Types of scheduling algorithms – Performance comparison – Inter process communications Synchronization – Semaphores.

UNIT III MEMORY MANAGEMENT 9

Single contiguous allocation – Partitioned allocation – Paging – Virtual memory concepts – Swapping – Demand paging – Page replacement algorithms – Segmentation.

UNIT IV DEVICE AND FILE MANAGEMENT 9

Principles of I/O hardware – I/O software – Disks – Disk Scheduling Algorithms--File Systems – Files and Directories- File System Implementation - Allocation Methods.

UNIT V SECURITY ISSUES 9

Protection in General Purpose Operating Systems: protected objects and methods of protection –memory and address protection – control of access to general objects – file protection Mechanisms –user authentication - Designing Trusted Operating Systems.

TOTAL : 45 PERIODS

COURSE OUTCOMES:

- Compare and contrast various memory management schemes.
- Design and Implement a prototype file systems.
- Discuss the various synchronization, and memory management issues.
- Demonstrate the Mutual exclusion, Deadlock detection and agreement protocols of Distributed operating system.
- Discuss the various Security issues.

TEXT BOOKS	
-------------------	--

<p>1. Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, —Operating System Concepts, 9th Edition, John Wiley and Sons Inc., 2012.</p> <p>2. Andrew S. Tanenbaum, —Modern Operating Systems, Second Edition, Addison Wesley, 2001.</p> <p>3. Charles Crowley, —Operating Systems: A Design-Oriented Approach, Tata McGraw Hill Education, 1996.</p> <p>4. Charles P. Pleegeer, "Security in Computing", Prentice Hall, New Delhi, 2009</p>	
REFERENCES	
<p>1. D M Dhamdhare, —Operating Systems: A Concept-Based Approach, Second Edition, Tata McGraw-Hill Education, 2007.</p> <p>2. Michael Palmer, Guide to Operating Systems Security, Course Technology – Cengage Learning, New Delhi, 2008.</p> <p>3. William Stallings, —Operating Systems – Internals and Design Principles, 7th Edition, Prentice Hall, 2011.</p> <p>1.</p>	
WEBSITE REFERENCES	
<p>1. https://courses.grainger.illinois.edu/CS598OSS/sp2021/</p> <p>2. https://ep.jhu.edu/courses/695612-operating-systems-security/</p>	

Mapping of CO with PO

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO-1	3					2
CO-2	3		3	2		3
CO-3	3	3	3	3	2	2
CO-4		3	3	3		3
CO-5	3	2	3	3		2

JSE5023	SENTIMENT ANALYSIS	L	T	P	C
		3	0	0	3

COURSE OBJECTIVES

- To understand the need for sentiment analysis.
- To explore the various methodologies involved in text sentiment classification.
- To learn the fusion of Natural Language processing with sentiment analysis.
- To explore available sentiment summarization methods.
- To learn the various tools used for sentiment analysis.

UNIT I INTRODUCTION

9

Need for Sentiment Analysis – Problem of Sentiment Analysis - Subjectivity – Stance – Words to Discourse – Pragmatics – Natural Language Processing issues – Opinion Definition – Sentiment analysis Tasks – Opinion Summarization – Types of opinion – Subjectivity and emotion – Author and Reader Standpoint

UNIT II DOCUMENT SENTIMENT CLASSIFICATION

9

Sentiment Classification Using Supervised Learning – Unsupervised Learning – Rating Prediction – Cross-Domain Sentiment Classification – Cross-Language Sentiment Classification – Sentence Subjectivity and Classification – Subjectivity Classification – Sentence Sentiment Classification – Conditional Sentences - Sarcastic Sentences – Cross- Language Subjectivity and Sentiment Classification – Discourse Information for Sentiment Classification

UNIT III ASPECT BASED SENTIMENT ANALYSIS

9

Aspect Sentiment Classification – Rules Of Opinions and Compositional Semantics – Aspect Extraction – Identifying Resource Usage Aspect – Simultaneous Opinion Lexicon Expansion and Aspect Extraction – Grouping Aspects Into Categories – Entity, Opinion Hold And Timing Extraction – Coreference Resolution and Word Sense Disambiguation – Aspect and Entity Extraction - Sentiment Lexicon Generation – Corpus Based Approach – Dictionary Based Approach – Desirable and Undesirable Facts

UNIT IV OPINION SUMMARIZATION

9

Aspect Based Opinion Summarization – Improvements to Aspect-Based Opinion Summarization – Contrastive View Summarization – Traditional Summarization – Analysis of Comparative Opinions – Identifying Comparative Sentences – Identifying Preferred Entities – Opinion Search and Retrieval – Opinion Spam Detection – Types of Spam Detection - Supervised and Un-Supervised Approach – Group Spam Detection

UNIT V TOOLS FOR SENTIMENT ANALYSIS

9

Detecting Fake or Deceptive Opinions - Quality of Review – Quality as Regression Model – Other Methods – Case Study – Sentiment Analysis Applications – Tools for Sentiment Analysis – Semantria – Meltwater – Google Analytics – Face Book Insights – Tweetstats.

TOTAL: 45 PERIODS

COURSE OUTCOMES

Upon completion of the course, the students will be able to:

- Apply the various algorithms to perform opinion mining and classification.
- Learn various supervised and unsupervised machine learning methods for sentiment analysis.
- Generate sentiment lexicons by applying NLP techniques.
- Solve problems on opinion summarization.
- Learn to use tools for sentiment analysis.

TEXT BOOKS	
<ol style="list-style-type: none"> 1. Bing Liu, “Sentiment Analysis and Opinion Mining”, Morgan and Claypool publishers, 2012. 2. Bing Liu, “Sentiment Analysis – Mining opinion, Sentiments and Emotions”, Cambridge University Press, 2015. 3. Bo Pang and Lillian Lee, “Opinion Mining and Sentiment Analysis”, Now Publishers Inc,2008. 4. Roy De Groot, “Data mining for Tweet Sentiment Classification – Twitter Sentiment Analysis”, LAP Lambert Academic Publishing, 2012. 	
REFERENCES	
<ol style="list-style-type: none"> 1. Bing Liu, “Sentiment Analysis and Opinion Mining”, Morgan and Claypool publishers, 2012. 2. Bing Liu, “Sentiment Analysis – Mining opinion, Sentiments and Emotions”, Cambridge University Press, 2015. 3. Bo Pang and Lillian Lee, “Opinion Mining and Sentiment Analysis”, Now Publishers Inc,2008. 4. Roy De Groot, “Data mining for Tweet Sentiment Classification – Twitter Sentiment Analysis”, LAP Lambert Academic Publishing, 2012. 	
WEBSITE REFERENCES	
<ol style="list-style-type: none"> 1. https://www.udemy.com/course/r-social-media-mining-scraping-with-twitter 2. https://courses.analyticsvidhya.com/courses/twitter-sentiment-analysis 3. https://www.datacamp.com/courses/sentiment-analysis-in-python 	

Mapping of CO with PO

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO-1	3					2
CO-2	3		3	2		3
CO-3	3	3	3	3	2	2
CO-4		3	3	3		3
CO-5	3	2	3	3		2

JSE9001

Software Testing and Reuse

3

0

0

3

COURSE OBJECTIVES

- To introduce the basics and necessity of software testing.
- To provide various testing techniques along with concepts of software bugs and its impact.
- To develop and validate a test plan.
- To build a testing team required.
- To understand different ways of implementing software reuse.

UNIT-I TESTING PRINCIPLES AND AXIOMS

9

Introduction – Making rational choices: basics of Games – strategy – preferences – payoffs – Mathematical basics – Game theory – Rational Choice – Basic solution concepts – non-cooperative versus cooperative games – Basic computational issues – finding equilibria and learning in games Typical application areas for game theory (e.g. Google's sponsored search, eBay auctions, electricity trading markets).

UNIT-II TEST STRATEGIES

9

Testing Strategies – White Box and Black Box Approach – Integration Testing – System and Acceptance Testing – Performance Testing – Regression Testing - Internationalization Testing – Adhoc Testing – Website Testing – Usability Testing – Accessibility Testing.

UNIT-III TESTING METHODOLOGIES

9

Basics of Software Testing – Test Generation from Requirements – Finite State Models – Combinatorial Designs - Test Selection, Minimization and Prioritization for Regression Testing – Test Adequacy, Assessment and Enhancement.

UNIT-IV SQA IN PROJECT MANAGEMENT

9

Project progress control – costs – quality management standards – project process standards – management and its role in SQA – SQA unit. Introduction to Software Quality - Challenges – Objectives – Quality Factors - Components of SQA – Contract Review – Development and Quality Plans - SQA Components in Project Life Cycle – SQA Defect Removal Policies – Reviews

UNIT-V SOFTWARE REUSE

9

Reuse architecture - Application Reuse - Component Reuse - Object and function Reuse – Layers of Reuse, Adopting organization for Reuse, Object oriented techniques for Reuse, Changes required in development environment and people to adopt reuse

TOTAL: 45 PERIODS

COURSE OUTCOMES

Upon completion of the course, the students will be able to:

- Obtain an insight to software testing.
- Apply both black box testing and white box testing.
- Understand and apply multiple levels of testing.
- Understand the role of a tester as an individual and as a teammember.
- Apply software testing for large projects using automated testingtools.

TEXT BOOKS	
<ol style="list-style-type: none">1. Paul C. Jorgensen, “Software Testing: A Craftsman’s Approach”, Fourth Edition, CRC Press, 20132. Dorothy Graham, Mark Fewster, “Experiences of Test Automation: Case Studies of Software Test Automation”, Pearson Education, 2012.3. Erich Gamma, —Design Patterns: Elements of Reusable Object-Oriented Software, Pearson Education, 2015.	
REFERENCES	
<ol style="list-style-type: none">1. Glenford J. Myers, Tom Badgett, Corey Sandler, “The Art of Software Testing”, Third Edition, John Wiley & Sons, 2012..2. Srinivasan Desikan, Gopalaswamy Ramesh, “Software Testing – Principles and Practices”, Pearson Education, 20093. Boris Beizer, “Software Testing Techniques”, Dream Tech Press, 2009.4. Mauro Pezze, Michal Young, “Software Testing and Analysis Process Principles and Techniques”, Wiley India, 2008.5. Aditya P. Mathur, “Foundations of Software Testing _ Fundamental Algorithms and Techniques”, Dorling Kindersley (India) Pvt. Ltd., Pearson Education, 2008.	

Mapping of CO with PO/PSO

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO-1	3					2
CO-2	3		3	2		3
CO-3	3	3	3	3	2	2
CO-4		3	3	3		3
CO-5	3	2	3	3		2

			L	T	P	C
JSE9002	SOFTWARE DEVELOPMENT AND MANAGEMENT		3	0	0	3

COURSE OBJECTIVES:

- To understand the Software Project Planning and Evaluation techniques.
- To plan and manage projects at each stage of the software development life cycle (SDLC).
- To learn about the activity planning and risk management principles.
- To manage software projects and control software deliverables.
- To develop skills to manage the various phases involved in project management and peoplemanagement.

UNIT I PROJECT EVALUATION AND PROJECT PLANNING 9

Importance of Software Project Management – Activities - Methodologies – Categorization of Software Projects – Setting objectives – Management Principles – Management Control – Project portfolio Management – Cost-benefit evaluation technology – Risk evaluation – Strategic program Management – Stepwise Project Planning.

UNIT II PROJECT LIFE CYCLE AND EFFORT ESTIMATION 9

Software process and Process Models – Choice of Process models - Rapid Application development – Agile methods – Dynamic System Development Method – Extreme Programming– Managing interactive processes – Basics of Software estimation – Effort and Cost estimation techniques – COSMIC Full function points - COCOMO II - a Parametric Productivity Model.

UNIT III ACTIVITY PLANNING AND RISK MANAGEMENT 9

Objectives of Activity planning – Project schedules – Activities – Sequencing and scheduling – Network Planning models – Formulating Network Model – Forward Pass & Backward Pass techniques – Critical path (CRM) method – Risk identification – Assessment – Risk Planning – Risk Management – – PERT technique – Monte Carlo simulation – Resource Allocation – Creation of critical paths – Cost schedules.

UNIT IV PROJECT MANAGEMENT AND CONTROL**9**

Framework for Management and control – Collection of data – Visualizing progress – Cost monitoring – Earned Value Analysis – Prioritizing Monitoring – Project tracking – Change control – Software Configuration Management – Managing contracts – Contract Management.

UNIT V STAFFING IN SOFTWARE PROJECTS**9**

Managing people – Organizational behavior – Best methods of staff selection – Motivation – The Oldham – Hackman job characteristic model – Stress – Health and Safety – Ethical and Professional concerns – Working in teams – Decision making – Organizational structures – Dispersed and Virtual teams – Communications genres – Communication plans – Leadership.

TOTAL: 45 PERIODS**COURSE OUTCOMES:**

At the end of the course, the students should be able to:

- Understand Project Management principles while developing software.
- Gain extensive knowledge about the basic project management concepts, framework and the process models.
- Obtain adequate knowledge about software process models and software effort estimation techniques.
- Estimate the risks involved in various project activities.
- Define the checkpoints, project reporting structure, project progress and tracking mechanisms using project management principles.

TEXT BOOK:

1. Bob Hughes, Mike Cotterell and Rajib Mall: Software Project Management – Fifth Edition, TataMcGraw Hill, New Delhi, 2012.

REFERENCES:

1. Robert K. Wysocki —Effective Software Project Management— Wiley Publication, 2011.
2. Walker Royce: —Software Project Management— Addison-Wesley, 1998.
3. Gopalswamy Ramesh, —Managing Global Software Projects— McGraw Hill Education (India), Fourteenth Reprint 2013.

Mapping of CO with PO/PSO

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO-1	3	2				2
CO-2	3	3	3	2		3
CO-3	3	3	3	3	2	2
CO-4		3	3	3		
CO-5	3	2	3	3		

JSE9003

**Software Engineering
Practices**

L	T	P	C
3	0	0	3

COURSE OBJECTIVES

- To understand Software Engineering Lifecycle Models
- To do project management and cost estimation
- To gain knowledge of the System Analysis and Design concepts.
- To understand software testing approaches
- To be familiar with DevOps practices

UNIT I INTRODUCTION 9

Software engineering concepts – Development activities – Software lifecycle models - Classical waterfall - Iterative waterfall – Prototyping – Evolutionary - Spiral – Software project management – Project planning – Estimation – Scheduling – Risk management – Software configuration management.

UNIT II SOFTWARE REQUIREMENT SPECIFICATION 9

Requirement analysis and specification – Requirements gathering and analysis – Software Requirement Specification – Formal system specification – Finite State Machines – Petrinets – Object modelling using UML – Use case Model – Class diagrams – Interaction diagrams – Activity diagrams – State chart diagrams – Functional modelling – Data Flow Diagram.

UNIT III ARCHITECTURE AND DESIGN 9

Software design – Design process – Design concepts – Coupling – Cohesion – Functional independence – Design patterns – Model-view-controller – Publish-subscribe – Adapter – Command – Strategy – Observer – Proxy – Facade – Architectural styles – Layered - Client-server - Tiered - Pipe and filter.- User interface design

UNIT IV TESTING 9

Testing- Unit testing – Black box testing– White box testing – Integration and System testing– Performance Testing – Regression Testing - Internationalization Testing – Adhoc Testing – Website Testing – Usability Testing – Accessibility Testing.- Debugging - Program analysis – Symbolic execution – Model Checking.

DevOps:Motivation-Cloud as a platform-Operations- Deployment Pipeline:Overall Architecture Building and Testing-Deployment- Case study:Migrating to Microservices.

TOTAL: 45 PERIODS

COURSE OUTCOMES:

At the end of this course, the students will be able to:

- Understand the advantages of various Software Development Lifecycle Models
- Gain knowledge on project management approaches as well as cost and schedule estimation strategies
- Use UML diagrams for analysis and design
- Architect and design using architectural styles and design patterns
- Understand the advantages of DevOps practices

TEXT BOOKS	
<ol style="list-style-type: none"> 1. Roger S. Pressman, —Software Engineering – A Practitioner’s Approach, Seventh Edition, Mc Graw-Hill International Edition, 2010 2. Ian Sommerville, —Software Engineering, 9th Edition, Pearson Education Asia, 2011. 	
REFERENCES	
<ol style="list-style-type: none"> 1. Bernd Bruegge, Alan H Dutoit, Object-Oriented Software Engineering, 2nd edition, Pearson Education, 2004. 2. Carlo Ghezzi, Mehdi Jazayeri, Dino Mandrioli, Fundamentals of Software Engineering, 2nd edition, PHI Learning Pvt. Ltd., 2010 3. Craig Larman, Applying UML and Patterns, 3rd ed, Pearson Education, 2005. 4. Len Bass, Ingo Weber and Liming Zhu, —DevOps: A Software Architect’s Perspective, Pearson Education, 2016 	

5. Rajib Mall, Fundamentals of Software Engineering, 3rd edition, PHI Learning Pvt. Ltd., 2009. 6. Stephen Schach, Software Engineering 7th ed, McGraw-Hill, 2007.

WEBSITE REFERENCES

1. <http://nptel.ac.in/>.

Mapping of CO with PO/PSO

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO-1	3		3			
CO-2	2		3	2		
CO-3	3	3	3	3	2	
CO-4		3	2	3		
CO-5	3	2	3	3		

JSE9004

DESIGN PATTERNS

L	T	P	C
3	0	0	3

COURSE OBJECTIVES

- To acquaint students with the basic of patterns, categories, and their usage.
- To make the student understand the relation between OOPS paradigm and design patterns
- To make the students understand how design patterns simplify the object creation process.
- To make the students understand how design patterns simplify the structural rearrangement
- To understand the usage of design patterns for industry scenarios

UNIT-I	INTRODUCTION	9
Introduction to patterns – Pattern categories – Relationship – Pattern description – Description of architectural patterns.		
UNIT-II	DESIGNPATTERN	9
Introduction – MVC, Describing Design Patterns -Problem solving by Design Pattern – Guidelines for selecting & using Design pattern		
UNIT-III	CREATIONAL PATTERN	9
Abstract factory – Builder – Factory methods – Prototype – Singleton – Real world examples		
UNIT-IV	STRUCTURAL PATTERN	9
Adapter – Bridge – Composite – Decorator – Real world example, Façade – Flyweight – Proxy – Real world examples.		
UNIT-V	BEHAVIORAL PATTERN	9
Chain of responsibility – Command – Interpreter – Iterator – Mediator – Real world examples, Memento - Observer - State - Strategy – Template method – Visitor –Real world examples..		

TOTAL: 45 PERIODS

COURSE OUTCOMES

Upon completion of the course, the students will be able to:

- Ability to understand the need for pattern, remember their types and significance.
- Ability to understand the relation between OOPS paradigm and designpatterns.
- Ability to apply the suitable creational pattern for the object creation problem and evaluate their effectiveness.
- Ability to apply the suitable structural pattern to make structural rearrangements.
- Ability to apply the suitable behavioral pattern to provide special purpose for objects and analyze their interaction

TEXT BOOKS	
-------------------	--

1. Erich Gamma, Ralph Johnson, Richard Helm and John Vlissides, —Design Patterns: Elements of Reusable Object-Oriented Software, Pearson Education, 2015.

REFERENCES

1. Frank Buschmann, Regine Meunier, Hans Rohnert, Peter Sommerlad, Michael Stal, —PatternOriented Software Architecture: A System of Patterns, Wiley India Pvt. Ltd., 2011.
2. Cay Horstmann, —Object-Oriented Design and Patterns, Wiley India Pvt. Ltd, 2012

Mapping of CO with PO/PSO

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO-1	3					2
CO-2	3		3	2		3
CO-3	3	3	3	3	2	3
CO-4		3	3	3		2
CO-5	3	2	3	3		3

JSE9005

WEB DESIGN

L	T	P	C
3	0	0	3

COURSE OBJECTIVES

The student should be able to

- Know the importance of web technologies for the real world applications
- Learn appropriate scripting languages
- Know the testing techniques to test the product
- Gain the skills and project-based experience needed for entry into web design and development careers

- To use a variety of strategies and tools to create websites.

UNIT-I SITE ORGANIZATION AND NAVIGATION 9

User centered design – Web medium – Web design process – Evaluating process – Site types and architectures – Navigation theory – Basic navigation practices – Search – Site maps

UNIT-II ELEMENTS OF PAGE DESIGN 9

Browser compatible design issues - Pages and Layout – Templates – Text – Color – Images – Graphics and Multimedia - GUI Widgets and Forms – Web Design patterns

UNIT-III SCRIPTING LANGUAGES 9

Client side scripting: XHTML – DHTML– JavaScript– XML Server side scripting: Perl – PHP – ASP/JSP Designing a Simple web application

UNIT-IV PRE-PRODUCTION MANAGEMENT 9

Principles of Project Management – Web Project Method – Project Road Map – Project Clarification – Solution Definition – Project Specification – Content – Writing and Managing content.

UNIT-V PRODUCTION, MAINTENANCE AND EVALUATION 9

Design and Construction – Testing, Launch and Handover – Maintenance – Review and Evaluation – Case Study.

TOTAL: 45 PERIODS

COURSE OUTCOMES

Upon completion of the course, the students will be able to:

- Obtain an insight to software testing.
- Apply various scripting languages for the development of web applications
- Follow Web design standards
- Develop websites for local community organizations.

TEXT BOOKS	
<ol style="list-style-type: none"> 1. Jennifer Niederst Robbins, “Learning Web Design”, OREILLY 4th Edition 2. Ricardo Zea, “Mastering Responsive Web Design”, PACKT Publishing, 2015 3. Justin Emond, Chris Steins, “Pro Web Project Management”, Apress, 2011 	
REFERENCES	

1. Jon Duckett, “HTML and CSS: Design and Build Websites”, John Wiley and Sons, edition 2014
2. Jon Duckett, Jack Moore, “JavaScript and JQuery: Interactive Front-End Web Development”, John Wiley and Sons, edition 2014
3. Uttam K. Roy “Web Technologies” Oxford University Press, 13th impression, 2017
4. Joel Sklar, —Principles of Web Design, Thomson Learning, 2001
5. Lynch, Horton and Rosenfeld, —Web Style Guide: Basic Design Principles for Creating Web Sites, 2nd Edition, Yale University Press, 2002
6. Thomas A. Powell, —The Complete Reference – Web Design, Tata McGraw Hill, Third Edition, 2003
7. Van Duyne, Landay, Hong, —The Design of Sites: Patterns for creating winning web sites, 2nd Edition, Prentice Hall, 2006
7. Wendy Willard, —Web Design: A Beginner's Guide, Second Edition, McGraw Hill Education (India) Private Limited, 2010

WEBSITE REFERENCES	
http://www.wpbeginner.com/category/wp-tutorials/	

Mapping of CO with PO/PSO

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO-1	3					3
CO-2	3		3	2		3
CO-3	3	3	3	3	2	3
CO-4		3	3	3		
CO-5	3	2	3	3		3

COURSE OBJECTIVES

- To introduce the basics and necessity of software testing.
- To provide various testing techniques along with concepts
- To be familiar with DevOps practices.
- To build a testing team required.
- To understand different ways of implementing software reuse

UNIT-I INTRODUCTION 9

Introduction –Evolution of Software testing- Myths and Facts-Goals-Psychology – Software Testing Models- Different Schools of software testing-Software testing Life cycle – Testing methodology. – **Software lifecycle models** :-Classical waterfall - Iterative waterfall – Prototyping – Evolutionary - Spira

UNIT-II TESTSTRATEGIES 9

Black box testing strategies: Black-Box Testing Techniques- Equivalent partitioning- Boundary Value Analysis (BVA)- State Transition Testing-Decision table based Testing - Cause-Effect Graphing Based Testing - Error Guessing- **White box testing strategies:** White-Box Testing Techniques- Logic Coverage criteria-Basic path testing- Graph matrices-Loop testing-Data flow testing-Mutation testing

UNIT-III AUTOMATION TESTING TOOLS 9

Selenium-UFT/QTP- Appium- JMeter- LoadRunner- SoapUI- Katalon Studio- Micro Focus ALM/QC-postman –test complete- **Telerik Test Studio**

UNIT-IV AUTOMATION TESTING TOOLS PROCESS 9

Automated Functional Testing using Selenium - Automated Functional Testing using UFT/QTP- Automated Mobile Testing Using Appium- Jira tool for Test Management- Automated Performance Testing using JMeter -Automated API/WebServices Testing using Postman- Cucumber for Acceptance Testing.

UNIT V SOFTWARE QUALITY ASSURANCE

9

Need for SQA – SQA activities – Building blocks of SQA – SQA planning & standards
–Reliability measures.

TOTAL: 45 PERIODS

COURSE OUTCOMES

Upon completion of the course, the students will be able to:

- Understand the basics and necessity of software testing.
- Apply both black box testing and white box testing.
- Understand the advantages of DevOps practices.
- Understand the concept of various automated testing tools.
- Apply software testing for large projects using automated testing tools.

TEXT BOOKS	
1. Paul C. Jorgensen, “Software Testing: A Craftsman’s Approach”, Fourth Edition, CRC Press, 2013	2. Dorothy Graham, Mark Fewster, “Experiences of Test Automation: Case Studies of Software Test Automation”, Pearson Education, 2012.
3. Arnon Axelrod ,” Complete Guide to Test Automation: Techniques, Practices, and Patterns for Building and Maintaining Effective Software Projects”, Kindle Edition,2018	4. Roger S.Pressman, ”Software Engineering, A practitioner’s approach”, Tata

McGraw Hill, 5th Edition, 2001.

5. Gerardus Blokdyk”, Automated Software Testing A Complete Guide”, -Reilly Edition 2020

REFERENCES

1. Glenford J. Myers, Tom Badgett, Corey Sandler, “The Art of Software Testing”, Third Edition, John Wiley & Sons, 2012..
2. Srinivasan Desikan, Gopalaswamy Ramesh, “Software Testing – Principles and Practices”, Pearson Education, 2009
3. Boris Beizer, “Software Testing Techniques”, Dream Tech Press, 2009.
4. Mauro Pezze, Michal Young, “Software Testing and Analysis Process Principles and Techniques”, Wiley India, 2008.
5. Elfriede Dustin, Thom Garrett, Bernie Gauf, “Implementing Automated Software Testing”, Addison-Wesley, 2009

Mapping of CO with PO/PSO

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO-1	3					3
CO-2	3		3	2		3
CO-3	3	3	3	3	2	2
CO-4		3	3	3		3
CO-5	3	2	3	3		3