

UNIVERSITY DEPARTMENTS
ANNA UNIVERSITY, CHENNAI 600 025
REGULATIONS - 2013
M.E. SOFTWARE ENGINEERING (FT)
I – IV SEMESTER CURRICULUM AND SYLLABUS

PROGRAM EDUCATIONAL OBJECTIVES:

The Software Engineering program educational objectives are to produce post graduates who demonstrate:

- an ability to design, synthesize, and analyze, software systems of increasing size and complexity at various abstraction levels, from the individual component to the entire system architecture
- an ability to define, assess, and tailor software quality practices, and software processes and methodologies for appropriate application on software development projects in a variety of domain areas
- an ability to be an effective member of a multi-disciplinary software-intensive product development team with an awareness of individual professional and ethical responsibilities
- an ability to communicate, to varied stakeholder audiences, technical concepts in a complete, concise, and correct manner in a format appropriate for the audience
- an ability to critically analyze issues in industry and research trends and analyze and use various state-of-the-art practices and tools

PROGRAM OUTCOMES

Students will be able to:

- a. Use of formal methods, use of finite state machines, software quality measurement and analysis, requirements gathering, software system design.
- b. Identifying project constraints and solutions, problem decomposition, requirements elicitation, design trade-off analysis
- c. Design to constraints and requirements; scope management for time constraints; addressing aspects such as maintainability, testability, or evolvability; software security.
- d. Design and analysis of usability tests, system debugging, performance testing and analysis, project metric tracking.
- e. Function effectively on teams to accomplish a common goal
- f. Communicate effectively to varied stakeholder audiences, technical concepts in a complete, concise, and correct manner in a format appropriate for the audience
- g. Analyze the local and global impact of computing on individuals, organizations, and society
- h. Recognize the need for and possess an ability to engage in lifelong learning and continuing professional development
- i. Use effective, proper and state-of-the-art software engineering tools and technologies.
- j. Critically analyze existing literature in an area of specialization and develop innovative and research oriented methodologies to tackle gaps identified

ANNA UNIVERSITY, CHENNAI 600 025

UNIVERSITY DEPARTMENTS

REGULATIONS - 2013

M.E. SOFTWARE ENGINEERING (FT)

I – IV SEMESTER CURRICULUM AND SYLLABUS

SEMESTER I

COURSE CODE	COURSE TITLE	L	T	P	C
THEORY					
SW8101	Software Architecture and Design	3	0	0	3
CP8151	Advanced Data Structures and Algorithms	3	0	2	4
CP8152	Object Oriented Systems Engineering	3	0	2	4
CP8153	Open Source Systems and Networking	3	0	0	3
MA8154	Advanced Mathematics for Computing	3	1	0	4
PRACTICAL					
SW8111	Professional Practice	0	0	2	1
TOTAL		15	1	6	19

SEMESTER II

COURSE CODE	COURSE TITLE	L	T	P	C
THEORY					
SW8201	Software Quality Assurance and Testing	3	0	2	4
SW8202	Integrated Software Project Management	3	0	0	3
CP8251	Virtualization Techniques	3	0	2	4
	Elective I	3	0	0	3
	Elective II	3	0	0	3
PRACTICAL					
SW8211	Case Study	0	0	2	1
SW8212	Technical Seminar	0	0	2	1
TOTAL		15	0	8	19

SEMESTER III

COURSE CODE	COURSE TITLE	L	T	P	C
THEORY					
CP8351	Security Principles and Practices	3	0	0	3
	Elective III	3	0	0	3
	Elective IV	3	0	0	3
	Elective V	3	0	0	3
PRACTICAL					
SW8311	Project Work Phase I	0	0	12	6
TOTAL		12	0	12	18

SEMESTER IV

COURSE CODE	COURSE TITLE	L	T	P	C
PRACTICAL					
SW8411	Project Work Phase II	0	0	24	12
TOTAL		0	0	24	12

TOTAL NO OF CREDITS: 68

LIST OF ELECTIVES

COURSE CODE	COURSE TITLE	L	T	P	C
SW8001	Agile Software Engineering	3	0	0	3
SW8002	Business Intelligence	3	0	0	3
SW8003	Enterprise Application Integration	3	0	0	3
SW8004	Enterprise Resource Planning Concepts	3	0	0	3
SW8005	Formal Methods in Software Engineering	3	0	0	3
SW8006	IT Systems Management	3	0	0	3
SW8007	Knowledge Management	3	0	0	3
SW8008	Managing Human Resource	3	0	0	3
SW8009	Principles of Supply Chain Management	3	0	0	3
SW8010	Software Agents	3	0	0	3
SW8011	Software Engineering Process Models	3	0	0	3
SW8012	Software Reliability and Metrics	3	0	0	3
SW8013	Software Requirements Management	3	0	0	3
SW8014	Software Test Automation	3	0	0	3
SW8015	User Interface Design	3	0	0	3
SW8016	Web Design and Management	3	0	0	3
IF8252	Cloud Computing Technologies	3	0	0	3
CP8074	Real Time Systems Design	3	0	0	3
SW8071	Software Verification and Validation	3	0	0	3

OBJECTIVES:

- To understand the architecture and quality attributes of the software.
- To understand the design fundamentals and methodologies of the software.
- To learn the tools of architectural design for the current trends.

UNIT I INTRODUCTION TO ARCHITECTURE 9
Notion of Architecture – Notion of Software Architecture - Architectural Styles – Description of Software Architecture –Visual Notation – Examples.

UNIT II DESIGN FUNDAMENTALS 9
The Basic Concepts of Design – Characteristics of Design Activities – Essential Elements of Design – The Factors that Affect the Design – Design Principles Basic Rules of Software Design – Design Processes

UNIT III DESIGN METHODOLOGIES 9
Structured Design – Design Practices – Stepwise Refinement – Incremental Design – Structured System Analysis and Design – Jackson Structured Programming – Jackson System Development – Designing with Objects – User Interface Design.

UNIT IV ARCHITECTURAL DESIGN 9
Typical Architectural Design – Data Flow – Independent Components – Call and Return – Using Styles in Design – Choices of Style – Combination of Styles – Architectural Design Space – Theory of Design Spaces – Design Space of Architectural Elements – Design Space of Architectural Styles.

UNIT V CASE STUDIES 9
Tools for Architectural design – Case Studies.

TOTAL: 45 PERIODS

OUTCOMES:

Upon Completion of the course, □ the students will be able to

- To reconstruct the software architecture that can be used for an application of your choice.
- Able to design the software using design fundamentals and methodologies.
- To create a good software by using the styles, architectural design space.

REFERENCES:

1. Hong Zhu, "Software Design Methodology From Principles to Architectural Styles", Elsevier, 2005.
2. David Budgen, "Software Design", Second Edition, Pearson Education, 2004.
3. Mary Shaw David Garlan, "Software Architectural Perspectives on an emerging discipline", EEE, PHI 1996.
4. John Robinson, "Software Design for Engineers and Scientists", Newnes, 2004.
5. A. G. Sutcliffe, "Human Computer Interface Design", Second Edition Macmillan, 1995.
6. Hassan Gomaa, Software Modeling Design, Cambridge University Press, 2011

OBJECTIVES:

- To extend the students' knowledge of algorithms and data structures, and to enhance their expertise in algorithmic analysis and algorithm design techniques.
- Expected to learn a variety of useful algorithms and techniques and extrapolate from them in order to then apply those algorithms and techniques to solve problems

UNIT I FUNDAMENTALS**9**

Mathematical Proof Techniques: Induction, proof by contradiction, direct proofs - Asymptotic Notations – Properties of Big-oh Notation –Conditional Asymptotic Notation – Algorithm Analysis – Amortized Analysis – Introduction to NP-Completeness/NP-Hard – Recurrence Equations – Solving Recurrence Equations – Time-Space Tradeoff.

UNIT II HEAP STRUCTURES**9**

Min/Max heaps – Dears – Leftist Heaps – Binomial Heaps – Fibonacci Heaps – Skew Heaps – Lazy-Binomial Heaps.

UNIT III SEARCH STRUCTURES**9**

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

UNIT IV GEOMETRIC ALGORITHMS**9**

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

UNIT V PARALLEL ALGORITHMS**9**

Flynn's Classifications – List Ranking – Prefix computation – Array Max – Sorting on EREW PRAM – Sorting on Mesh and Butterfly – Prefix sum on Mesh and Butterfly – Sum on mesh and butterfly – Matrix Multiplication – Data Distribution on EREW, Mesh and Butterfly.

TOTAL: 45 + 30 : 75 PERIODS**OUTCOMES:**

- Basic ability to analyze algorithms and to determine algorithm correctness and time efficiency class.
- Master a variety of advanced data structures and their implementations.
- Master different algorithm design techniques in computational geometry and in parallel algorithms.
- Ability to apply and implement learned algorithm design techniques and data structures to solve problems.

REFERENCES:

1. E. Horowitz, S. Sahni and Dinesh Mehta, Fundamentals of Data structures in C++, University Press, 2007.
2. G. Brassard and P. Bratley, Algorithmics: Theory and Practice, Printice –Hall,1988.
3. Mark de Berg, Otfried Cheong, Marc van Kreveld, Mark Overmars, Computational Geometry Algorithms and Applications, Third Edition, 2008
4. James A. Storer, An Introduction to Data Structures and Algorithms, Springer, New York, 2002.
5. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein Introduction to Algorithms,Third Edition,2009.

CP8152

OBJECT ORIENTED SYSTEMS ENGINEERING

L T P C
3 0 2 4

OBJECTIVES:

- To understand the importance of object oriented software engineering.
- To study the various lifecycle models for developing software's.
- To analyze and design software using tools.
- To develop efficient software, deploy and maintain after production.

UNIT I CLASSICAL PARADIGM

9+6

System Concepts – Project Organization – Communication – Project Management

UNIT II PROCESS MODELS

9+6

Life cycle models – Unified Process – Iterative and Incremental – Workflow – Agile Processes

UNIT III ANALYSIS:

9+6

Requirements Elicitation – Use Cases – Unified Modeling Language, Tools – Analysis Object Model (Domain Model) – Analysis Dynamic Models – Non-functional requirements – Analysis Patterns

UNIT IV DESIGN

9+6

System Design, Architecture – Design Principles - Design Patterns – Dynamic Object Modeling – Static Object Modeling – Interface Specification – Object Constraint Language

UNIT V IMPLEMENTATION, DEPLOYMENT AND MAINTENANCE

9+6

Mapping Design (Models) to Code – Testing - Usability – Deployment – Configuration Management – Maintenance

TOTAL:45 + 30 =75 PERIODS

OUTCOMES:

- To prepare object oriented design for small/ medium scale problem.
- To evaluate the appropriate life cycle model for the system under consideration.
- To apply the various tools and patterns while developing software
- Testing the software against usability, deployment, maintenance.

REFERENCES:

1. Bernd Bruegge, Alan H Dutoit, Object-Oriented Software Engineering, 2nd ed, Pearson Education, 2004.
2. Craig Larman, Applying UML and Patterns 3rd ed, Pearson Education, 2005.
3. Stephen Schach, Software Engineering 7th ed, McGraw-Hill, 2007.
4. Ivar Jacobson, Grady Booch, James Rumbaugh, The Unified Software Development Process, Pearson Education, 1999.
5. Alistair Cockburn, Agile Software Development 2nd ed, Pearson Education, 2007.

CP8153

OPEN SOURCE SYSTEMS AND NETWORKING

L T P C
3 0 0 3

OBJECTIVES:

- To understand the basic issues in open source kernels
- To appreciate the different aspects of processes
- To understand the role played by files and devices
- To understand the basic issues in open source networking
- To appreciate the different aspects of internetworking

UNIT I	FOUNDATION	8
Introduction – Memory addressing – processes – interrupts and exceptions – kernel synchronization – timing measurements.		
UNIT II	PROCESSES	9
Process scheduling – memory management – process address space – system calls - signals.		
UNIT III	FILES AND DEVICES	9
Virtual File System – I/O architecture and device drivers – block device drivers – page cache – accessing files		
UNIT IV	NETWORKING	10
Introduction, critical data structures, user space to kernel interface – System initialization: notification chains, device initialization, PCI layer, network interface cards, component initialization, device registration and initialization - Transmission and reception: interrupts and network drivers, frame reception and transmission, protocol handlers – Bridging: concepts, spanning tree protocol, implementation.		
UNIT V	INTERNETWORKING	9
IPv4: concepts, foundation and features, forwarding and local delivery, transmission, handling fragmentation – Neighbouring subsystem: concepts, infrastructure, ARP – Routing: concepts, advanced features, implementation.		
		TOTAL: 45 PERIODS

OUTCOMES:

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

- Identify the different features of open source kernels
- Install and use available open source kernel
- Modify existing open source kernels in terms of functionality or features used
- Identify different features of open source networking
- Modify and use existing open source networking modules

REFERENCES:

1. Daniel P Bovet and Marco Cesati, “Understanding the Linux kernel”, 3rd edition, O’Reilly, 2005.
2. Christian Benvenuti, “Understanding Linux Network Internals”, O’Reilly, 2006.
3. Y-D Lin, R-H Hwang and Fred Baker, “Computer networks – an open source approach”, McGraw-Hill, 2012.
4. Alessandro Rubini and Jonathan Corbet, “Linux device drivers”, 2nd edition, O’Reilly, 2001.
5. Maurice J Bach, “The design of the Unix operating system”, Pearson, 1986.

MA8154	ADVANCED MATHEMATICS FOR COMPUTING	L T P C
		3 1 0 4

OBJECTIVES:

To understand the basics of random variables and standard distributions

- To understand the arrival process and various queueing and server models
- To appreciate the use of simulation techniques
- To apply testing of hypothesis to infer outcome of experiments
- To apply mathematical linear programming techniques to solve constrained problems.

UNIT I	RANDOM VARIABLES	12
Random variables – Bernoulli, Binomial, Geometric, Poisson, Uniform, Exponential, Erlang and Normal distributions – Function of a Random variable - Moments, Moment generating function.		
UNIT II	QUEUEING MODELS	12
Poisson Process – Markovian Queues – Single and Multi-server Models – Little’s formula – Machine Interference Model – Steady State analysis – Self Service Queue.		
UNIT III	SIMULATION	12
Discrete Event Simulation – Monte – Carlo Simulation – Stochastic Simulation – Applications to Queuing systems.		
UNIT IV	TESTING OF HYPOTHESIS	12
Sampling distributions – Estimation of parameters - Statistical hypothesis – Tests based on Normal, t, Chi-square and F distributions for mean, variance and proportion.		
UNIT V	LINEAR PROGRAMMING	12
Formulation – Graphical solution – Simplex method – Two phase method -Transportation and Assignment Problems.		

TOTAL: 60 PERIODS

OUTCOMES:

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

- Identify the type of random variable and distribution for a given operational conditions/scene
- Study and Design appropriate queuing model for a given problem/system situation
- To understand and simulate appropriate application/distribution problems
- Differentiate/infer the merit of sampling tests.
- Formulate and find optimal solution in the real life optimizing/allocation/assignment problems involving conditions and resource constraints.

REFERENCES:

1. Johnson, R.A. Miller and Freund’s,” Probability and Statistical for Engineers, Prentice Hall of India Pvt., Ltd., New Delhi, Seventh Edition, 2005.
2. Hamdy A. Taha, “Operations Research: An Introduction”, Prentice Hall of India Pvt., Ltd. New Delhi, Eighth Edition, 2007.
3. Jay L. Devore,” Probability and Statistics for Engineering and the Sciences”, Cengage Learning, Seventh Edition, 2009.
4. Ross. S.M., “Probability Models for Computer Science”, Academic Press, 2002.
5. Winston, W.L., “Operations Research”, Thomson – Brooks/Cole, Fourth Edition, 2003.
6. Gross D. and Harris C.M., “Fundamentals of Queueing Theory”, John Wiley and Sons, New York, 1998.
7. J.Medhi,” Stochastic models of Queueing Theory”, Academic Press, Elsevier, Amsterdam, 2003.

SW8111

PROFESSIONAL PRACTICE

L T P C
0 0 2 1

THE OBJECTIVES OF PROFESSIONAL PRACTICE:

- To Facilitate Research, Analysis, and Problem Solving.
- To Interview people who know the context of the Problem and the Solution
- To Explore various possible alternative solutions
- To Estimate Risk

THE OUTCOMES OF PROFESSIONAL PRACTICE:

Formulating a Problem
Describing the Background of the Problem
Assessing the needs of the People
Framing a Policy
Predicting Business Opportunity
Understanding System Implications.

TOTAL: 30 PERIODS

SW8201

SOFTWARE QUALITY ASSURANCE AND TESTING

L T P C
3 0 2 4

OBJECTIVES:

- To Know what is software and the usage of different types of software's.
- To know the Quality Metrics of various Software's.
- Plans, methods and process are executed to get a good Quality software.
- Knowing the methodologies in making Software.
- To test the product finally to check the product Quality.

UNIT I INTRODUCTION

9+6

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+6

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 III TEST STRATEGIES

9+6

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 Test plan – management – execution and reporting – software test automation – automated testing tools

UNIT IV SOFTWARE QUALITY

9+6

Hierarchical models of software quality – software quality metrics –function points -Software product quality – software maintenance quality – effect of case tools – software quality infrastructure – procedures – certifications – configuration management – documentation control.

UNIT V SQA IN PROJECT MANAGEMENT

9+6

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

TOTAL: 45+30 = 75 PERIODS

OUTCOMES:

- To analyze the product Quality by various testing methods.
- To use various testing methods for the appropriate applications.
- To assess Quality standards.

REFERENCES:

1. Daniel Galin, Software quality assurance – from theory to implementation , Pearson education, 2009.
2. Yogesh Singh, "Software Testing", Cambridge University Press, 2012.
3. Aditya Mathur, Foundations of software testing, Pearson Education, 2008
4. Ron Patton, Software testing , second edition, Pearson education, 2007
5. Srinivasan Desikan and Gopalaswamy 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.

SW8202

INTEGRATED SOFTWARE PROJECT MANAGEMENT

L T P C
3 0 0 3

OBJECTIVES:

- To understand the basic concept of project management
- To appreciate the various costing and life cycle management
- To understand the role played by risk in software project
- To appreciate the use of metrics for software project management
- To appreciate the challenges in people management

UNIT I PROJECT MANAGEMENT CONCEPTS 9

Evolution of Software Economics – Software Management Process Framework (Phases, Artifacts, Workflows, Checkpoints) – Software Management Disciplines (Planning / Project Organization and Responsibilities / Automation / Project Control) – Modern Project Profiles – Maturity Models

UNIT II COSTING & LIFECYCLE MANAGEMENT 9

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.SLIM (Software Life cycle Management) – PLM (Product Lifecycle Management) – PDM (Product Data Management) - PLM, PDM Applications – Pre-PLM Environment – Pre-PLM Lifecycle – 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

UNIT IV METRICS 9

Need for Software Metrics – 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).

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: 45PERIODS

OUTCOMES:

Upon Completion of the course the students 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. Stark, John, Decision Engineering: Product Lifecycle Management: 21st Century Paradigm for Product Realisation, 2nd Edition., 2011, XXII, 559 p., Springer London
2. Antonio Borghesi, Barbara Gaudenzi, Risk Management: How to Assess, Transfer and Communicate Critical Risks, Volume 5 of Perspectives in Business Culture, Illustrated Edition, Springer 2012
3. McConnell, S. "Software Project: Survival Guide", Microsoft Press, 1998.
4. Boehm, B. W. "Software Risk Management: Principles and Practices" in IEEE Software, January 1991, pp32-41.
5. Fenton, N.E., and Pfleeger, S.L.. "Software Metrics: A Rigorous and Practical Approach, Revised" Brooks Cole, 1998.
6. Royce, W. "Software Project management: A Unified Framework", Addison-Wesley, 1998.

CP8251

VIRTUALIZATION TECHNIQUES

L T P C
3 0 2 4

OBJECTIVES:

- To understand the need of virtualization
- To explore the types of virtualization
- To understand the concepts of virtualization and virtual machines
- To understand the practical virtualization solutions and enterprise solutions
- To understand the concepts of cloud computing
- To have an introduction to cloud programming giving emphasis to Hadoop MapReduce
- To understand the security issues in cloud computing

UNIT I OVERVIEW OF VIRTUALIZATION

9+6

Basics of Virtualization – Types of Virtualization Techniques – Merits and demerits of Virtualization – Full Vs Para-virtualization – Virtual Machine Monitor/Hypervisor - Virtual Machine Basics – Taxonomy of Virtual machines – Process Vs System Virtual Machines – Emulation: Interpretation and Binary Translation - HLL Virtual Machines

UNIT II SERVER AND NETWORK VIRTUALIZATION 9+6

Server Virtualization: Virtual Hardware Overview - Server Consolidation – Partitioning Techniques - Uses of Virtual server Consolidation – Server Virtualization Platforms, **Network Virtualization:** Design of Scalable Enterprise Networks – Layer2 Virtualization – VLAN - VFI - Layer 3 Virtualization – VRF - Virtual Firewall Contexts - Network Device Virtualization - Data- Path Virtualization - Routing Protocols

UNIT III STORAGE, DESKTOP AND APPLICATION VIRTUALIZATION 9+6

Storage Virtualization: Hardware Devices – SAN backup and recovery techniques – RAID – Classical Storage Model – SNIA Shared Storage Model – Virtual Storage: File System Level and Block Level, **Desktop Virtualization:** Concepts - Desktop Management Issues - Potential Desktop Virtualization Scenarios - Desktop Virtualization Infrastructures, **Application Virtualization:** Concepts - Application Management Issues - Redesign Application Management – Application Migration

UNIT IV APPLYING VIRTUALIZATION 9+6

Practical Virtualization Solutions: Comparison of Virtualization Technologies: Guest OS/ Host OS – Hypervisor – Emulation – Kernel Level – Shared Kernel, **Enterprise Solutions:** VMWare Server – VMWare ESXi – Citrix Xen Server – Microsoft Virtual PC – Microsoft Hyper-V – Virtual Box, **Server Virtualization:** Configuring Servers with Virtualization – Adjusting and Tuning Virtual servers – VM Backup – VM Migration, **Desktop Virtualization:** Terminal services – Hosted Desktop – Web-based Solutions – Localized Virtual Desktops, **Network and Storage Virtualization:** Virtual Private Networks – Virtual LAN – SAN and VSAN – NAS

UNIT V CLOUD COMPUTING 9+6

Cloud Computing Basics - Cloud Computing Definition – Evolution of Cloud Computing - General Cloud Environments – Cloud Services – Service Providers – Google – Amazon – Microsoft – IBM – EMC – NetApp - Salesforce – Tools for building private cloud - Open Issues in Cloud Computing – Cloud security challenges, **Cloud Programming:** Hadoop - MapReduce – HDFS – Hadoop I/O – Developing a MapReduce Application

TOTAL: 45 + 30 = 75 PERIODS

OUTCOMES:

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

- Deploy legacy OSs on virtual machines
- Understand the intricacies of server, storage, network, desktop and application virtualizations
- Design new models for virtualization
- Design and develop cloud applications on virtual machine platforms
- Design new models for Big data processing in cloud

REFERENCES:

1. James E. Smith, Ravi Nair, - Virtual Machines: Versatile Platforms for Systems and Processes, Elsevier/Morgan Kaufmann, 2005.
2. David Marshall, Wade A. Reynolds, - Advanced Server Virtualization: VMware and Microsoft Platform in the Virtual Data Center, Auerbach Publications, 2006.
3. Kumar Reddy, Victor Moreno, - Network virtualization, Cisco Press, July, 2006.
4. Chris Wolf, Erick M. Halter, - Virtualization: From the Desktop to the Enterprise, APress 2005.
5. Danielle Ruest, Nelson Ruest - Virtualization: A Beginner's Guide, TMH, 2009
6. Kenneth Hess , Amy Newman: Practical Virtualization Solutions: Virtualization from the Trenches Prentice Hall 2010
7. John Rittinghouse, James Ransome, Cloud Computing, Implementation, Management and Strategy, CRC Press, 2010

8. Anthony T. Velte, Toby J. Velte, Robert Elsenpeter - Cloud Computing: A Practical Approach, TMH, 2010
9. Lee Badger , Tim Grance , Robert Patt-Corner , Jeff Voas - Cloud Computing Synopsis and Recommendations NIST, May 2011
10. Tom White - Hadoop: The Definitive Guide Storage and Analysis at Internet Scale O'Reilly Media Press May 2012
11. Dave Shackelford - Virtualization security- Protecting Virtualized Environments, Sybex Publishers, First Edition, 2012

SW8211

CASE STUDY

L T P C
0 0 2 1

The case study approach is to engage students in critical thinking for realworld situations. As Students, they turn basic knowledge into principles that can be applied across cases. By placing them in real situations, and requiring them to make decisions, students learn to connect their knowledge with analytical skills.

A Case Study includes Research, Analysis, and Problem Solving. Interviewing people who know the place or the situation is a vital step. There is no single solution. A case study includes.

- Introduction
- Background
- People
- Policy
- Business Opportunity
- System Implications

TOTAL: 30 PERIODS

SW8212

TECHNICAL SEMINAR

L T P C
0 0 2 1

THE OBJECTIVES OF TECHNICAL SEMINAR ARE:

1. To elicit pro-active participation of the students through
2. To entrust assignment to present
3. To inculcate presentation and leadership skills among students
4. To involving students to learn actively
5. To offer opportunities of interaction with peer students and staff

THE OUTCOMES OF THE TECHNICAL SEMINAR ARE:

1. Good Communications Skills.
2. Knowing the Audience.
3. Choosing the Topic.
4. Setting the Goals for the Talk.
5. Talking to the Audience.
6. Knowing the Content of the Talk.
7. Preparation of the Slides.
8. Answering Questions.
9. Managing Time.

TOTAL: 30 PERIODS

OBJECTIVES:

- To understand the mathematical foundations of security principles
- To appreciate the different aspects of encryption techniques
- To understand the role played by authentication in security
- To appreciate the current trends security practices

UNIT I INTRODUCTION AND MATHEMATICAL FOUNDATION 9

An illustrative communication game – safeguard versus attack – Probability and Information Theory - Algebraic foundations – Number theory.

UNIT II ENCRYPTION – SYMMETRIC TECHNIQUES 9

Substitution Ciphers – Transposition Ciphers – Classical Ciphers – DES – AES – Confidentiality Modes of Operation – Key Channel Establishment for symmetric cryptosystems.

UNIT III ENCRYPTION – ASYMMETRIC TECHNIQUES AND DATA TECHNIQUES 9

Diffie-Hellman Key Exchange protocol – Discrete logarithm problem – RSA cryptosystems & cryptanalysis – ElGamal cryptosystem – Need for stronger Security Notions for Public key Cryptosystems – Combination of Asymmetric and Symmetric Cryptography – Key Channel Establishment for Public key Cryptosystems - Data Integrity techniques – Symmetric techniques - Asymmetric techniques

UNIT IV AUTHENTICATION 9

Authentication Protocols Principles – Authentication protocols for Internet Security – SSH Remote logic protocol – Kerberos Protocol – SSL & TLS – Authentication frame for public key Cryptography – Directory Based Authentication framework – Non - Directory Based Public-Key Authentication framework .

UNIT V SECURITY PRACTICES 9

Protecting Programs and Data – Information and the Law – Rights of Employees and Employers – Software Failures – Computer Crime – Privacy – Ethical Issues in Computer Security.

TOTAL: 45 PERIODS**OUTCOMES:**

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

- Use the mathematical foundations in security principles
- Identify the features of encryption and authentication
- Use available security practices

REFERENCES:

1. William Stallings, "Cryptography and Network security: Principles and Practices", Pearson/PHI, 5th Edition, 2010.
2. Behrouz A. Forouzan, "Cryptography and Network Security", 2nd Edition, Tata McGraw Hill Education, 2010.
3. Wade Trappe, Lawrence C Washington, "Introduction to Cryptography with coding theory", 2nd Edition, Pearson, 2007.
4. Douglas R. Stinson, "Cryptography Theory and Practice ", 3rd Edition, Chapman & Hall/CRC, 2006.
5. W. Mao, "Modern Cryptography – Theory and Practice", Pearson Education, 2nd Edition, 2007.
6. Charles P. Pfleeger, Shari Lawrence Pfleeger, "Security in computing", 3rd Edition, Prentice Hall of India, 2006.
7. Wenbo Mao, "Modern Cryptography – Theory and Practice", Pearson Education, 2006.
8. Charlie Kaufman, Radia Perlman and Mike Speciner, " Network Security Private Communication in a Public World", PHI, Second Edition, 2012

OBJECTIVES:

- To provide students with a theoretical as well as practical understanding of agile software development practices and how small teams can apply them to create high-quality software
- a good understanding of software design and a set of software technologies and APIs
- Detailed examination and demonstration of Agile development and testing techniques
- 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, Extreme Programming: Method overview – lifecycle – work products, roles and practices

UNIT III AGILITY AND KNOWLEDGE MANAGEMENT 9

Agile information systems – agile decision making - Earl's 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 product development – Agile Metrics – Feature Driven Development (FDD) – Financial and Production Metrics in FDD – Agile approach to Quality Assurance - Test Driven Development – Agile approach in Global Software Development

TOTAL: 45 PERIODS**OUTCOMES:**

- The importance of interacting with business stakeholders in determining the requirements for a software system.
- Iterative software development processes: how to plan them, how to execute them.
- The impact of social aspects on software development success.
- Techniques and tools for improving team collaboration and software quality.
- Software process improvement as an ongoing task for development teams.
- How agile approaches can be scaled up to the enterprise level.

REFERENCES:

1. David J. Anderson; Eli Schragenheim, Agile Management for Software Engineering: Applying the Theory of Constraints for Business Results, Prentice Hall, 2003
2. Hazza & Dubinsky, Agile Software Engineering, Series: Undergraduate Topics in Computer Science, Springer 2009, VIII, 296 p.
3. Craig Larman, Agile and Iterative Development: A manager's Guide, Addison-Wesley, 2004.
4. Kevin C. Desouza, Agile information systems: conceptualization, construction, and management, Butterworth-Heinemann, 2007.
5. Chetankumar Patel, Muthu Ramachandran, Story Card Maturity Model (SMM): A Process Improvement Framework for Agile Requirements Engineering Practices, Journal of Software, Academy Publishers, Vol 4, No 5 (2009), 422-435, Jul 2009.

SW8002

BUSINESS INTELLIGENCE

L T P C
3 0 0 3

OBJECTIVES:

- To identify the enormous opportunities that currently exists in providing business intelligence services
- Gain a practical understanding of the key data mining methods of classification, prediction, data reduction and exploration
- To understand and help develop the strategies of modern enterprise decision makers
- Knowledge in many scientific and technological fields including data warehouses, data mining, content analytics, business process management, visual analytics, and requires competences in information systems, web science, decision science, software engineering, and innovation and entrepreneurship

UNIT I INTRODUCTION

9

BI Basics – Meeting the BI challenge – BI user models – Basic reporting and querying – BI Markets - BI and Information Exploitation – Value of BI – BI cycle – Bridging the analysis gap – BI Technologies – BI Decision Support Initiatives – BI Project Team

UNIT II BI BIG PICTURE

9

Advanced Emerging BI Technologies – Human factors in BI implementations – BI design and development – OO Approach to BI - BI Environment – BI business process and information flow – Identifying BI opportunities – Evaluating Alternatives - BI solutions – BI Project Planning

UNIT III BI ARCHITECTURE

9

Components of BI Architecture – BI Design and prototyping – Importance of Data in Decision Making - Data requirements Analysis - Using OLAP for BI – Data warehouse and Technical BI Architecture – Business Rules – Data Quality – Data Integration – High performance BI

UNIT IV BI TECHNOLOGIES

9

Successful BI – LOFT Effect – Importance of BI Tools – BI standardization - Creating business value through location based intelligence – Technologies enabling BI – technologies for information integration - Building effective BI Systems – Strategic, Tactical, Operational and Financial Intelligence

UNIT V FUTURE OF BI

9

BI Case Studies - Evaluating BI – Stakeholder model of BI - Knowledge Discovery and Data Mining for Predictive Analytics – Knowledge Delivery – BI Trends – BI Search and Text Analytics – Advanced Visualisation – Semantic Web Technologies for building BI

TOTAL: 45 PERIODS

OUTCOMES:

- Be aware of the business intelligence potential of today's data rich environment
- Know how to decide when to use which technique
- Understand how to implement major techniques using Excel add-ins
- Gain the intellectual capital required to provide business analytics services

REFERENCES:

1. Cindi Howson, Successful Business Intelligence, Tata McGraw-Hill Education, 2007
2. Murugan Anandarajan, Asokan Anandarajan, Cadambi A. Srinivasan, Business Intelligence Techniques: A Perspective from Accounting and Finance, illustrated Springer, 2003
3. Larissa Terpeluk Moss, S. Atre, Business Intelligence Roadmap: The Complete Project Lifecycle for Decision-Support Applications, Addison-Wesley Information Technology Series, illustrated edition, Addison-Wesley Professional, 2003
4. Robert J. Thierauf, Effective Business Intelligence Systems - ABC-Clio e-book, illustrated Edition, Greenwood Publishing Group, 2001
5. Swain Scheps, Business Intelligence For Dummies, John Wiley & Sons, 2011
6. David Loshin, Business Intelligence: The Savvy Manager's Guide, Morgan Kaufmann, 2nd Edition, Newnes Publishers, 2012
7. Rajiv Sabherwal, Irma Becerra-Fernandez, Business Intelligence, illustrated Edition, John Wiley & Sons, 2010
8. Elizabeth Vitt, Michael Luckevich, Stacia Misner, Business Intelligence, O'Reilly Media, Inc., 2010

SW8003

ENTERPRISE APPLICATION INTEGRATION

L T P C
3 0 0 3

OBJECTIVES:

- Create a Business Data Catalog Application Definition file
- discuss the most important topics and technologies related to enterprise application integration
- To expose the students to practice implementation strategies.
- To familiarize the student with the Integration facilities for an application.
- To appreciate the current trends in Enterprise Application Integration.

UNIT I INTRODUCTION

9

Business Imperative for Enterprise Integration – Business agility – Business transactions – need of E-business - ROI of Enterprise Integration – Challenges – Business drivers – Defining Requirements – Enterprise Integration strategy.

UNIT II ENTERPRISE INTEGRATION ARCHITECTURE

9

Overview – Business case – Components of EIA – Organizational Structure – Architectural Governance - Understanding Integration Technology – Current Integration Architecture – Technical Integration Architecture specification.

UNIT III SERVICE AND INFORMATION INTEGRATION ARCHITECTURE

9

Service Oriented Architecture – Benefits – Defining Services – Event driven service design – specification – Understanding Metadata – Metadata Architecture – standards – Information Integration Patterns – Architecture Specification.

UNIT IV PROCESS AND APPLICATION INTEGRATION ARCHITECTURE 9
 Process to Business – Process Integration Technology – Process Standards – Architecture Specification - Choosing Technology - Application Integration Technology – Implementation Specification – Composite Application – Composite integration specification.

UNIT V CASE STUDY 9
TOTAL: 45 PERIODS

OUTCOMES:

- Learn how to use object-oriented concepts in developing web applications
- Understand the enterprise application integration
- Deploy the components of EIA with solutions
- Better collaboration needed in application integration.

REFERENCES:

1. David S.Linthicum, “Enterprise Application Integration”, Addison – Wesley Information Technology Services, 2006.
2. Martin Fowler Patterns of Enterprise Application Architecture (Addison-Wesley Signature Series) 2002

SW8004 ENTERPRISE RESOURCE PLANNING CONCEPTS L T P C
3 0 0 3

OBJECTIVES:

- To enrich students with concepts and knowledge of ERP.
- To focus on illustrating procurement, production, and sales business processes using ERP software.
- To bridge the gap between the need of business process knowledge and its application to the business environment.
- To learn ERP Implementation Success & Failure for an application.

UNIT I INTRODUCTION TO ERP 9
 Overview – Benefits of ERP – ERP and Related Technologies – Business Process Reengineering – Data Warehousing – Data Mining – On–line Analytical Processing – Supply Chain Management.

UNIT II ERP IMPLEMENTATION 9
 Implementation Life Cycle – cost model - Implementation Methodology – Hidden Costs – Organizing Implementation – Vendors, Consultants and Users – Contracts – Project Management and Monitoring.

UNIT III BUSINESS MODULES 9
 ERP and business process reengineering - Business Modules in an ERP Package – Finance – Manufacturing – Human Resource – Plant Maintenance – Materials Management – Quality Management – Sales and Distribution – ERP & E-business – ERP & CRM

UNIT IV ERP MARKET 9
 ERP Market Place – SAP AG – PeopleSoft – Baan Company – JD Edwards World Solutions Company – Oracle Corporation – QAD – System Software Associates – ERP financials – Auditing ERP – ERP Business Intelligence and Performance Management

UNIT V ERP – APPLICATIONS**9**

Lean manufacturing and ERP - Turbo Charge the ERP System – EIA Study of ERP selection process – Big Bang ERP implementation – Impact of ERP systems on organizational effectiveness – Knowledge management for enterprise systems

TOTAL: 45 PERIODS**OUTCOMES:**

- To make the student a manager of computer service offerings across business processes of an organization.
- understand the structure of an ERP system
- To prepare them to become knowledgeable ERP user professionals suitable to Industry and Information Technology Companies.

REFERENCES:

1. Marianne Bradford, Modern ERP – Select, Implement and Use – Today's Advanced Business Systems , North Carolina State University, Second Edition, 2010
2. Alexis Leon, "ERP Demystified", Tata McGraw Hill, 1999.
3. Joseph A. Brady, Ellen F. Monk, Bret J. Wangner, "Concepts in Enterprise Resource Planning" , Thomson Learning, 2001.
4. Vinod Kumar Garg and N.K .Venkata Krishnan, "Enterprise Resource Planning – concepts and Planning", Prentice Hall, 1998.
5. Jose Antonio Fernandez, " The SAP R /3 Hand book", Tata McGraw Hill, 1998.

SW8005**FORMAL METHODS IN SOFTWARE ENGINEERING****L T P C
3 0 0 3****OBJECTIVE:**

- Students will learn to translate the informal diagrammatic and textual notations into formal interpretation sufficient for particular forms of automated reasoning. To understand the formal semantics and tools

UNIT I INTRODUCTION**8**

Need for Formal methods – Problems in Natural Language Specifications, Formal Versus Informal Programming – Advantages of Formal Methods – Requirements of Formal System – Types – Propositional Logic – Predicate Logic – Relationships and Functions.

UNIT II FORMAL SPECIFICATION STYLE**8**

Model-Oriented – Specifications – Concurrency-Based Specifications –Example Specification Languages.

UNIT III VDM**10**

Introduction to VDM – Basic Types – Quote Types – Compound Types – Optional Types – Functions – Operations – Additional Constructs – Modules.

UNIT IV THE Z NOTATION**10**

The Interchange Language – User-Defined Identifiers – Data Types – Basic Types – Compound Types – Schemas – Additional Constructs.

UNIT V FORMAL SEMANTICS AND TOOLS

9

Operational Semantics – Denotational Semantics – Axiomatic Semantics Proof Editors – Proof Analyser – Symbolic Simulators – Translators – Test Generation Tools.

TOTAL: 45 PERIODS

OUTCOMES:

- Ability to apply knowledge of logic, formal methods and software engineering methods.
- To use VDM and Z notation in the context of software engineering
- To assess the software against semantics and tools.

REFERENCES:

1. Andrew Harry, “ Formal Methods: Fact File VDM and Z”, John Wiley and Sons, 1996.
2. Jim Woodcock, Jim Davies, “Using Z Specification, Refinement and Proof”, Prentice Hall International, 1996.

SW8006

IT SYSTEMS MANAGEMENT

L T P C
3 0 0 3

OBJECTIVES:

- To study the importance of change, configuration and operations in Systems Management.
- To manage the people, tools and process based on the change.
- It provides frameworks and management principles that current or aspiring managers can employ to cope with the challenges inherent in the implementation of rapidly advancing technology.
- Formulate a project proposal that includes a broad research issue anchored in the management of information systems literature

UNIT I INTRODUCTION

9

Introduction to IT Systems Management - Scope and definitions, Current and future Technology – hardware, software, storage, networks, Systems Management frameworks.

UNIT II CHANGE MANAGEMENT

9

Change Management Processes - Identifying the need for change, Making a business case and measuring return on investment, Managing change – people, tools, processes.

UNIT III CONFIGURATION MANAGEMENT

9

Configuration Management Processes - Current and Target Architecture Definitions, Vendor Selection, Solution Deployment.

UNIT IV OPERATIONS MANAGEMENT I

9

Operations Management Processes - Application Management, Performance Management, Capacity Planning.

UNIT V OPERATIONS MANAGEMENT II

9

Operations Management Processes - Business Continuity and Disaster Recovery, Problem Problem Management, Exception Management.

TOTAL: 45 PERIODS

OUTCOMES:

- Analyze the need of system management.
- Ability to predict the need for change, and manage the same.
- To evaluate the risk ahead and plan for the business continuity.
- Write up a report and a project report of high quality that draws upon and synthesizes a number of sources and fulfills the objectives in the project proposal
- Apply critical thinking to academic and practice sources

REFERENCES:

1. Schiesser, Rich, "IT Systems Management", Prentice Hall of India, New Delhi, 2002.
2. Frenzel, Carrol W, "Management of Information Technology", Boyd and Fraser, Boston, 1992.
3. Davenport, Thomas H, "Information Ecology - Mastering the information and knowledge environment", Oxford University Press, 1997.
4. Phillips, Joseph, "IT Project Management - On track from start to finish", Tata McGraw Hill, 2002.
5. Williams, Brian K., Stacey C. Sawyer, Sarah E. Hutchinson; "Using Information Technology - A practical introduction to computers and communications", 3rd Edition, Tata McGraw Hill, 1999.
6. Turban, Efraim, Rainer, R. Kelly, Potter, Richard E., "Introduction to Information Technology", John Wiley, 2003.

SW8007**KNOWLEDGE MANAGEMENT****L T P C
3 0 0 3****OBJECTIVES:**

- To learn knowledge engineering basics
- To Knowing the knowledge models
- To know the techniques of knowledge management and implementation

UNIT I INTRODUCTION**9**

The value of Knowledge – Knowledge Engineering Basics – Knowledge Economy – The Task and Organizational Content – Knowledge Management – Knowledge Management Ontology.

UNIT II KNOWLEDGE MODELS**9**

Knowledge Model Components – Template Knowledge Models –Reflective Knowledge Models– Knowledge Model Construction – Types of Knowledge Models.

UNIT III TECHNIQUES OF KNOWLEDGE MANAGEMENT**8**

Knowledge Elicitation Techniques – Modeling Communication Aspects – Knowledge Management and Organizational Learning.

UNIT IV KNOWLEDGE SYSTEM IMPLEMENTATION**11**

Case Studies – Designing Knowledge Systems – Knowledge Codification – Testing and Deployment – Knowledge Transfer and Knowledge Sharing – Knowledge System Implementation.

UNIT V ADVANCED KNOWLEDGE MANAGEMENT**8**

Advanced Knowledge Modeling – Value Networks – Business Models for Knowledge Economy – UML Notations – Project Management.

TOTAL: 45 PERIODS**OUTCOMES:**

- Learning knowledge engineering basics
- Knowing the knowledge models
- knowing the techniques of knowledge management and implementation

TEXT BOOKS:

1. Guus Schreiber, Hans Akkermans, Anjo Anjewierden, Robert de Hoog, Nigel Shadbolt, Walter Van de Velde and Bob Wielinga, "Knowledge Engineering and Management", Universities Press, 2001.
2. Elias M.Awad & Hassan M. Ghaziri, "Knowledge Management", Pearson Education, 2003.

REFERENCES:

- 1.C.W. Holsapple, "Handbooks on Knowledge Management", International Handbooks on Information Systems, Vol 1 and 2, 2003.
2. <http://www.epistemics.co.uk>
3. http://depts.washington.edu/pettt/papers/WIN_poster_text.pdf

SW8008**MANAGING HUMAN RESOURCE****L T P C****3 0 0 3****OBJECTIVES:**

To know the importance of resources for a task.

- To compare all the resources with Human resources so the employee constraints are checked to meet the completion of the task.
- Training types are studied for employees
- To forecasting Human Resource requirement.
- To know the selection procedures.

UNIT I PERSPECTIVES IN HUMAN RESOURCE MANAGEMENT**9**

Evolution of human resource management – the importance of the human factor – objectives of human resource management – role of human resource manager – human resource policies – computer applications in human resource management.

UNIT II THE CONCEPT OF BEST FIT EMPLOYEE**9**

Importance of human resource planning – forecasting human resource requirement – internal and external sources. Selection process-screening – tests - validation – interview - medical examination – recruitment introduction – importance – practices – socialization benefits.

UNIT III TRAINING AND EXECUTIVE DEVELOPMENT**9**

Types of training, methods, purpose, benefits and resistance. Executive development programmes – common practices - benefits – self development – knowledge management.

UNIT IV SUSTAINING EMPLOYEE INTEREST**9**

Compensation plan – reward – motivation – theories of motivation – career management – development, mentor – protégé relationships.

UNIT V PERFORMANCE EVALUATION AND CONTROL PROCESS**9**

Method of performance evaluation – feedback – industry practices. Promotion, demotion, transfer and separation – implication of job change. The control process – importance – methods – requirement of effective control systems grievances – causes – implications – redressal methods.

TOTAL: 45 PERIODS**OUTCOMES:**

To apply recruitment techniques in employee selection process.

- To apply appropriate training process
- To motivate employees by sustaining employee Interests.explain the development, management and exploitation of information systems, discuss their impact on organizations and evaluate

REFERENCES:

1. Decenzo and Robbins, Human Resource Management, Wilsey, 6th edition, 2001.
2. Biswajeet Pattanayak, Human Resource Management, Prentice Hall of India, 2001.
3. Human Resource Management, Eugence Mckenna and Nic Beach, Pearson Education Limited, 2002.
4. Dessler Human Resource Management, Pearson Education Limited, 2002.
5. Mamoria C.B. and Mamoria S. Personnel Management, Himalaya Publishing Company, 1997.
6. Wayne Cascio, Managing Human Resource, McGraw Hill, 1998.
7. Ivancevich, Human Resource Management, McGraw Hill 2002.

SW8009

PRINCIPLES OF SUPPLY CHAIN MANAGEMENT

L T P C
3 0 0 3

OBJECTIVES:

To learn about the E-business environment driven by the Automation Software in quick movement of supply of products

- To study the fundamentals of supply chain management comprised of Inventory management and warehousing etc as co parts of entire business
- Learning the cost management for the supply of products
- The main objectives of Supply chain management are to improve the overall organization performance and customer satisfaction by improving product or service delivery to consumer.
- The higher the supply chain profitability or surplus, the more successful is the supply chain

UNIT I FUNDAMENTALS OF SUPPLY CHAIN MANAGEMENT 9

Supply chain networks, Integrated supply chain planning, Decision phases in s supply chain, process view of a supply chain, supply chain flows, Overview of supply chain models and modeling systems, Supply chain planning: Strategic, operational and tactical, Understanding supply chain through process mapping and process flow chart.

UNIT II SCM STRATEGIES, PERFORMANCE 9

Supply chain strategies, achieving strategic fit, value chain, Supply chain drivers and obstacles, Strategic Alliances and Outsourcing, purchasing aspects of supply chain, Supply chain performance measurement: The balanced score card approach, Performance Metrics. Planning demand and supply: Demand forecasting in supply chain, Aggregate planning in supply chain, Predictable variability.

UNIT III PLANNING AND MANAGING INVENTORIES 9

Introduction to Supply Chain Inventory Management. Inventory theory models: Economic Order Quantity Models, Reorder Point Models and Multi echelon Inventory Systems, Relevant deterministic and stochastic inventory models and Vendor managed inventory models.

UNIT IV DISTRIBUTION MANAGEMENT 9

Role of transportation in a supply chain - direct shipment, warehousing, cross-docking; push vs. pull systems; transportation decisions (mode selection, fleet size), market channel structure, vehicle routing problem. Facilities decisions in a supply chain. Mathematical foundations of distribution management, Supply chain facility layout and capacity planning,

UNIT V STRATEGIC COST MANAGEMENT IN SUPPLY CHAIN

9

The financial impacts, Volume leveraging and cross docking, global logistics and material positioning, global supplier development, target pricing, cost management enablers, Measuring service levels in supply chains, Customer Satisfaction/Value/Profitability/Differential Advantage.

TOTAL: 45 PERIODS

OUTCOMES:

- Learning about the e- business for supply of products managed with the appropriate methodologies and management techniques
- Knowing Supply Chain Management consists of all parties (Including Manufacturer, Marketer, Suppliers, transporters, Warehouses, Retailers and even customers) directly or indirectly involved in fulfillment of a customer
- The supply chains of tomorrow must deliver varying degrees of six outcomes — the traditional cost-related benefit plus responsiveness, security, sustainability, resilience and innovation — depending on key customers' needs
- Knowing automated back office software systems
- Knowing basic business process

REFERENCES:

1. David Simchi-Levi, Philip Kaminsky, and Edith Simchi-Levi Designing and Managing the Supply Chain: Concepts, Strategies, and Case Studies, Second Edition, , McGraw- Hill/Irwin, New York, 2003.
2. Sunil Chopra and Peter Meindel. Supply Chain Management: Strategy, Planning, and Operation, Prentice Hall of India, 2002.
3. Sunil Chopra & Peter Meindl, Supply Chain Management , Prentice Hall Publisher, 2001
4. Robert Handfield & Ernest Nichols, Introduction to Supply Chain Management, Prentice hall Publishers, 1999

SW8010

SOFTWARE AGENTS

L T P C
3 0 0 3

OBJECTIVES:

- To overview of the agent systems and **software agents**.
- To understand the basic concepts of intelligent software agents.
- To enable the students to design and build a multiagent system.
- To have a basic understanding about software agent technology and to be familiar with some of the communicating languages, standardization and applications.
- To 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 – Aplets 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

OUTCOMES:

- 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 and application of distributed information systems that use software agents

REFERENCES:

1. Bigus & Bigus, " Constructing Intelligent agents with Java ", Wiley, 1997.
2. Bradshaw, " Software Agents ", MIT Press, 2000.
3. Russel, Norvig, "Artificial Intelligence: A Modern Approach", Second Edition, Pearson Education, 2003.
4. Richard Murch, Tony Johnson, "Intelligent Software Agents", Prentice Hall, 2000.
5. Gerhard Weiss, "Multi Agent Systems – A Modern Approach to Distributed Artificial Intelligence", MIT Press, 2000.

SW8011	SOFTWARE ENGINEERING PROCESS MODELS	L T P C
		3 0 0 3

OBJECTIVES:

- To learn the Software Engineering process models to make the project effectively
- To learn different types of process models
- To learn software development phases
- To know set of methods in project reviews and inspections
- To know work practices tools and techniques in developing software

UNIT I	PROCESS AND BASIC PROCESS MODELS	9
Process Definition – Process for Software Development and Maintenance – Process Models – Waterfall – Prototypes – Throwaway – Evolutionary – Incremental.		
UNIT II	ADVANCED PROCESS MODELS	8
Spiral – Rapid Application Development – Unified Process Models.		
UNIT III	ADVANCED PROCESS MODELS – II	12
Agile – Extreme Programming (XP) – Adaptive Software Development (ASD) – DSDM – Scrum – Crystal – Feature Driven Development (FDD) – Comparison of Different Models.		

UNIT IV PROCESS IMPROVEMENT MODELS – I **8**
Need for Process Improvement – ISO 9000: 2000 – SPICE.

UNIT V PROCESS IMPROVEMENT MODELS – II **8**
Six Sigma – CMMI.

TOTAL: 45 PERIODS

OUTCOMES:

Demonstrate the Software Process models for the Adaptive development of Software Knowing the process improvement for the product quality improvement.

- To categorizes and examines a number of methods for describing or modeling how software systems are developed.
- Knowing the software life cycle
- contemporary models of software development must account for software the interrelationships between software products and production processes, as well as for the roles played by tools, people and their workplaces

REFERENCES:

1. Pankaj Jalote , “An Integrated Approach to Software Engineering”, Second Edition, Springer Verlag, 1997.
2. Roger S. Pressman, “Software Engineering: A Practitioner’s Approach”, Fifth Edition, McGraw Hill, 2001.
3. Ian Sommerville, “Software Engineering”, Sixth Edition, Addison Wesley, 2000.
4. Jim Highsmith , "Agile Software Development Ecosystems", First Edition, Addison Wesley, 2002.
5. Alistair Cockburn , "Agile Software Development", First Edition, Pearson Education Asia, 2001.
6. Kent Beck, “eXtreme Programming eXplained : EMBRACE CHANGE”, First Edition, Pearson Education Asia, 1999.
7. Philippe Kruchten, “The Rational Unified Process, an introduction”, Second Edition, Addison Wesley, 2000.
8. Humphrey Watts S, “Managing the Software Process”, Addison Wesley, 1989.
9. Alan C. Gillies, “Software Quality - Theory and Management”, Second Edition, International Thomson Computer Press, 1999.
10. David Hoyle, “ISO 9000 Quality Systems Handbook”, Fourth Edition, Butterworth – Heinemann, 2001.
11. Peter S. Pande, Larry Holpp, Pete Pande, Lawrence Holpp, “ What Is Six Sigma?”, McGraw-Hill Trade, 2001.

SW8012 SOFTWARE RELIABILITY AND METRICS **L T P C**
3 0 0 3

OBJECTIVES:

- To understand different definitions of software quality and how you might measure it
- To understand different notions of ‘defects’ and be able to classify them
- To understand the basic techniques of data collection and how to apply them
- To learn software metrics that define relevant metrics in a rigorous way.
- To gain confidence in ultra-high reliability

UNIT I INTRODUCTION TO SOFTWARE RELIABILITY **9**
Basic Concepts – Failure and Faults – Environment – Availability –Modeling –uses – requirements reliability metrics – design & code reliability metrics – testing reliability metrics

UNIT II SOFTWARE RELIABILITY MODELING 9
Concepts – General Model Characteristic – Historical Development of models – Model Classification scheme – Markovian models – General concepts – General Poisson Type Models – Binomial Type Models – Poisson Type models – Fault reduction factor for Poisson Type models.

UNIT III COMPARISON OF SOFTWARE RELIABILITY MODELS 9
Comparison Criteria – Failure Data – Comparison of Predictive Validity of Model Groups – Recommended Models – Comparison of Time Domains – Calendar Time Modeling – Limiting Resource Concept – Resource Usage model – Resource Utilization – Calendar Time Estimation and confidence Intervals

UNIT IV FUNDAMENTALS OF MEASUREMENT 9
Measurements in Software Engineering – Scope of Software metrics – Measurements theory – Goal based Framework – Software Measurement Validation.

UNIT V PRODUCT METRICS 9
Measurement of Internet Product Attributes – Size and Structure – External Product Attributes – Measurement of Quality –Reliability Growth Model – Model Evaluation

TOTAL : 45 PERIODS

OUTCOMES:

- To be able to perform some simple statistical analysis relevant to software measurement data.
- To understand from practical examples both the benefits and limitations of software metrics for quality control and assurance.

REFERENCES:

1. John D. Musa, Anthony Iannino, Kazuhira Okumoto, “Software Reliability – Measurement, Prediction, Application, Series in Software Engineering and Technology”, McGraw Hill, 1987.
2. John D. Musa, “Software Reliability Engineering”, Tata McGraw Hill, 1999.
3. Norman E . Fenton, Shari Lawrence Pfleeger, "Software metrics", Second Edition, International Student Edition, 2003.

SW8013 SOFTWARE REQUIREMENTS MANAGEMENT L T P C
3 0 0 3

OBJECTIVES:

- To understand the basics of requirements engineering
- To appreciate the different techniques used for requirements elicitation
- To understand the role played by requirements analysis in requirement integration
- To appreciate the use of various methodologies for requirements development
- To appreciate the current trends in requirements prioritization and validation

UNIT I REQUIREMENTS ENGINEERING OVERVIEW 9
Software Requirement Essentials – Requirements from Customer Perspective - REM (Requirements Engineering & Management) in Agile Projects - Software Development Roles –Software Development Process Kernels – Commercial Life Cycle Model – Vision & Scope Development – Stakeholders Needs - Root Cause Analysis

OBJECTIVES:

- To understand the basics of test automation
- To appreciate the different aspects of test tool evaluation and test automation approach selection
- 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 TEST PLANNING AND DESIGN**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**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

REFERENCES:

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
3. Dorothy Graham, Mark Fewster, Experiences of Test Automation: Case Studies of Software Test Automation, illustrated Edition, Addison-Wesley Professional, 2012.

4. Kanglin Li, Mengqi Wu, Effective Software Test Automation: Developing an Automated Software Testing Tool, John Wiley & Sons, 2006
5. Linda Hayes, The Automated Testing Handbook, Software testing Inst., 1995.
6. 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.
7. Mark Fewster & Dorothy Graham, Software Test Automation, Addison Wesley, 1999.

SW8015

USER INTERFACE DESIGN

L T P C
3 0 0 3

OBJECTIVES:

- To understand how to study the tasks that the user needs to accomplish with the software system.
- To learn the constraints that affect the UI design.
- To study the importance of human- computer interaction.
- To identify the various facilities provided in WINDOWS including multimedia.

UNIT I INTRODUCTION

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.

UNIT V EVALUATION

9

Conceptual Model Evaluation – Design Standards Evaluation – Detailed User Interface Design Evaluation – User centered design processes – heuristic evaluation
Usability Testing – understanding users and their goals – planning for usability testing – analyzing and reporting usability test results

TOTAL: 45 PERIODS

OUTCOMES:

- To design a more user friendly software.
- To utilize the existing functionalities provided and develop a better design.
- To predict the need of the end user and design the interface accordingly.
- To evaluate the design with respect to the benchmarked standards.

REFERENCES:

1. Wilbent. O. Galitz, "The Essential Guide To User Interface Design", John Wiley & Sons, 2001.
2. Deborah Mayhew, The Usability Engineering Lifecycle, Morgan Kaufmann, 1999 Ben Shneiderman, "Design The User Interface", Pearson Education, 1998.
3. Alan Cooper, "The Essential of User Interface Design", Wiley – Dream Tech Ltd., 2002. Sharp, Rogers, Preece, 'Interaction Design', Wiley India Edition, 2007
4. Carol M. Barnum, Usability Testing Essentials: Ready, Set...Test, Elsevier, 2010.

SW8016

WEB DESIGN AND MANAGEMENT

L T P C
3 0 0 3

OBJECTIVES:

- To know the importance of web technologies for the real world applications
- To learn the web page design with the appropriate scripting languages in producing a quality web application
- To know the testing techniques to test the product
- Will gain the skills and project-based experience needed for entry into web design and development careers.
- Will be able to use a variety of strategies and tools to create websites.
- Will develop awareness and appreciation of the many ways that people access the web, and will be able to create standards-based websites that can be accessed by the full spectrum of web access technologies.

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

OUTCOMES:

- Various scripting languages are studied for the development of web applications
- Web design standards are studied
- provides students with an opportunity for "real world" experience designing and developing websites for local community organizations.
- emphasizes standards-based and accessible design.
- begins by building a foundation of design theory principles, and all web design techniques are taught with these principles in mind

REFERENCES:

1. Thomas A. Powell, "The Complete Reference – Web Design", Tata McGraw Hill, Third Edition, 2003.
2. Ashley Friedlein, "Web Project Management", Morgan Kaufmann Publishers, 2001.
3. H. M. Deitel, P. J. Deitel, A. B. Goldberg, "Internet and World Wide Web – How to Program", Third Edition, Pearson Education 2004.
4. Joel Sklar, "Principles of Web Design", Thomson Learning, 2001.
5. Van Duyne, Landay, and Hong "The Design of Sites: Patterns for creating winning web sites", 2nd Edition, Prentice Hall, 2006.
6. Lynch, Horton and Rosenfeld, "Web Style Guide: Basic Design Principles for Creating Web Sites", 2nd Edition, Yale University Press, 2002.

IF8252**CLOUD COMPUTING TECHNOLOGIES****L T P C**
3 0 0 3**OBJECTIVES:**

- To understand the concept of cloud and utility computing.
- To understand the various issues in cloud computing.
- To familiarize themselves with the state of the art 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****8**

Evolution of Cloud Computing – System Models for Distributed and Cloud Computing – NIST Cloud Computing Reference Architecture -IaaS – On-demand Provisioning – Elasticity in Cloud – E.g. of IaaS Providers - PaaS – E.g. of PaaS Providers - SaaS – E.g. of SaaS Providers – Public , Private and Hybrid Clouds.

UNIT II**VIRTUALIZATION****9**

Basics of Virtualization - Types of Virtualization - Implementation Levels of Virtualization - Virtualization Structures - Tools and Mechanisms - Virtualization of CPU, Memory, I/O Devices - Desktop Virtualization – Server Virtualization.

UNIT III**CLOUD INFRASTRUCTURE****9**

Architectural Design of Compute and Storage Clouds – Layered Cloud Architecture Development – Design Challenges - Inter Cloud Resource Management – Resource Provisioning and Platform Deployment – Global Exchange of Cloud Resources.

UNIT IV**PROGRAMMING MODEL****10**

Parallel and Distributed Programming Paradigms – Map Reduce , Twister and Iterative MapReduce – Hadoop Library from Apache – Mapping Applications - Programming Support - Google App Engine, Amazon AWS - Cloud Software Environments -Eucalyptus, Open Nebula, Open Stack.

UNIT V**SECURITY IN THE CLOUD****9**

Security Overview – Cloud Security Challenges – Software-as-a-Service Security – Security Governance – Risk Management – Security Monitoring – Security Architecture Design – Data Security – Application Security – Virtual Machine Security.

TOTAL: 45 PERIODS

OUTCOMES:

Upon Completion of the course, the students should 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

REFERENCES:

1. Kai Hwang, Geoffrey C Fox, Jack G Dongarra, "Distributed and Cloud Computing, From Parallel Processing to the Internet of Things", Morgan Kaufmann Publishers, 2012.
2. John W.Rittinghouse and James F.Ransome, "Cloud Computing: Implementation, Management, and Security", CRC Press, 2010.
3. Toby Velte, Anthony Velte, Robert Elsenpeter, "Cloud Computing, A Practical Approach", TMH, 2009.
4. George Reese, "Cloud Application Architectures: Building Applications and Infrastructure in the Cloud" O'Reilly, 2009.
5. James E. Smith, Ravi Nair, "Virtual Machines: Versatile Platforms for Systems and Processes", Elsevier/Morgan Kaufmann, 2005.
6. Katarina Stanoevska-Slabeva, Thomas Wozniak, Santi Ristol, "Grid and Cloud Computing – A Business Perspective on Technology and Applications", Springer, 2010.

CP8074

REAL TIME SYSTEM DESIGN

L T P C
3 0 0 3

OBJECTIVE :

- To learn real time operating system concepts and the associated issues & techniques

UNIT I REAL TIME SPECIFICATION AND DESIGN TECHNIQUES

9

Introduction– Structure of a Real Time System –Task classes – Performance Measures for Real Time Systems – Estimating Program Run Times – Issues in Real Time Computing – Task Assignment and Scheduling – Classical uniprocessor scheduling algorithms –Fault Tolerant Scheduling.

UNIT II REAL TIME SPECIFICATION AND DESIGN TECHNIQUES

9

Natural languages – mathematical specification – flow charts – structured charts – pseudocode and programming design languages – finite state automata – data flow diagrams – petri nets – Warnier Orr notation – state charts – polled loop systems – phase / state driven code – coroutines – interrupt – driven systems – foreground/background system – full featured real time operating systems.

UNIT III INTERTASK COMMUNICATION AND SYNCHRONIZATION

9

Buffering data – mailboxes – critical regions – semaphores – deadlock – process stack management – dynamic allocation – static schemes – response time calculation – interrupt latency – time loading and its measurement – scheduling is NP complete – reducing response times and time loading – analysis of memory requirements – reducing memory loading – I/O performance.

UNIT IV REAL TIME DATABASES

9

Real time Databases – Basic Definition, Real time Vs General Purpose Databases, Main Memory Databases, Transaction priorities, Transaction Aborts, Concurrency control issues, Disk Scheduling

Algorithms, Two – phase Approach to improve Predictability – Maintaining Serialization Consistency – Databases for Hard Real Time Systems.

UNIT V EVALUATION TECHNIQUES

9

Reliability Evaluation Techniques – Obtaining parameter values, Reliability models for Hardware Redundancy – Software error models. Clock Synchronization – Clock, A Nonfault – Tolerant Synchronization Algorithm – Impact of faults – Fault Tolerant Synchronization in Hardware – Fault Tolerant Synchronization in software.

TOTAL: 45 PERIODS

OUTCOME:

- Understanding principles of real time systems design; be aware of architectures and behaviors of real time operating systems, database and applications.

REFERENCES:

1. C.M. Krishna, Kang G. Shin, “Real – Time Systems”, McGraw – Hill International Editions, 1997.
2. Rajib Mall, ”Real-time systems: theory and practice”, Pearson Education, 2007
3. Stuart Bennett, “Real Time Computer Control – An Introduction”, Prentice Hall of India, 1998.
4. R.J.A Buhur, D.L Bailey, “An Introduction to Real – Time Systems”, Prentice – Hall International, 1999.
5. Philip.A.Laplante, “Real Time System Design and Analysis”, Prentice Hall of India, 3rd Edition, April 2004.
6. Allen Burns, Andy Wellings, “Real Time Systems and Programming Languages”, Pearson Education, 2003.

SW8071

SOFTWARE VERIFICATION AND VALIDATION

L T P C
3 0 0 3

OBJECTIVES:

- To understand the principles of verification and validation
- To appreciate the different verification and validation techniques
- To understand the various stages of testing
- To appreciate the use of tools for verification and validation
- To appreciate the benefit 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 – cleanroom 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 –

UNIT IV TOOLS FOR SOFTWARE VERIFICATION**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

UNIT V ADVANCED APPROACHES**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 : 45 PERIODS**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 some sample requirements
- Modify existing coverage analysers in terms of functionality or features used
- Design system test cases for application of your choice
- Use test case generators and test management tools for sample application

REFERENCES:

1. Mourad Debbabi, Hassaine F, Jarrya Y., Soeanu A., Alawneh L., Verification and Validation in Systems Engineering, Springer, 2010.
2. Marcus S. Fisher, Software Verification and Validation: An Engineering and Scientific Approach, Springer, 2007
3. ESA Board for Software Standardisation and Control (BSSC), Guide to software verification and Validation, European Space Agency ESA PSS-05-10 Issue 1 Revision 1, March 1995
4. Avner Engel, Verification, Validation & Testing of Engineered Systems, Wiley series in systems Engineering and Management, 2010.