

ANNA UNIVERSITY, CHENNAI
UNIVERSITY DEPARTMENTS
M.E. SOFTWARE ENGINEERING
REGULATIONS – 2015
CHOICE BASED CREDIT SYSTEM

PROGRAMME EDUCATIONAL OBJECTIVES (PEOs) :

- I. To prepare students to acquire excellent knowledge in software engineering fundamentals
- II. To train students in software industry related practices and disciplines
- III. To prepare students to take up advance core and applied research related to software engineering research
- IV. To provide students with a solid foundation in mathematical fundamentals required to solve software engineering problems
- V. To train students with good domain engineering knowledge so as to comprehend, analyze, design, and create novel products and solutions for the real life problems.
- VI. To inculcate in students professional and ethical attitude, effective communication skills, teamwork skills, multidisciplinary approach, and ability to relate software engineering issues to broader social context.
- VII. To provide students with an academic environment which will lead to achieving excellence, leadership qualities, following ethical codes and guidelines, life-long learning for a successful professional career

PROGRAMME OUTCOMES (POs):

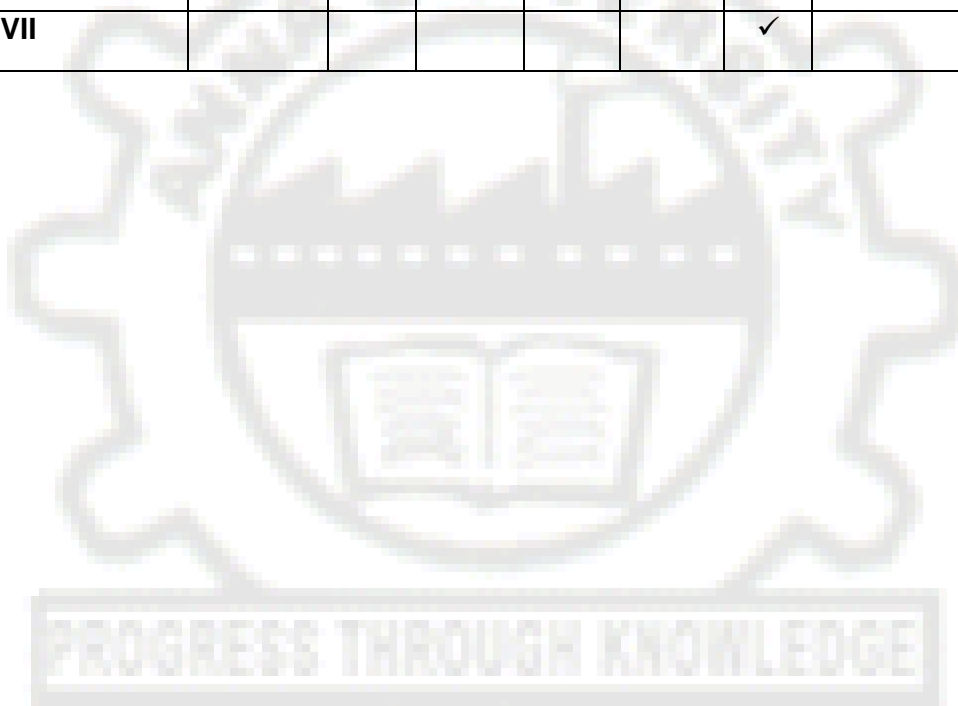
On successful completion of the programme,

1. Graduates will demonstrate an ability to identify, formulate and solve software engineering problems.
2. Graduates would be able to apply mathematics to solve problems in software engineering.
3. Graduates would be able to design and conduct experiments, analyze and interpret data.
4. Graduates would possess the ability to design a system, component or process as per needs and specifications.
5. Graduates would acquire skills to use automation softwares and tools to analyze problems.
6. Graduates would be able to communicate effectively in both verbal and written forms.
7. Graduates will know the impact of engineering solutions on the society and will be aware of contemporary issues.
8. Graduates would develop confidence for self-education and ability for life-long learning.

MAPPING OF PROGRAMME EDUCATIONAL OBJECTIVES WITH PROGRAMME OUTCOMES:

A broad relation between the programme objective and the outcomes is given in the following table

Programme Educational Objectives	Programme Outcomes							
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
I	✓		✓	✓				
II			✓	✓	✓		✓	
III	✓	✓	✓	✓	✓		✓	
IV		✓	✓					
V	✓		✓	✓			✓	
VI			✓	✓			✓	
VII						✓		✓



			PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	
YEAR 1	SEM 1	Advanced Mathematics for Computing		✓	✓						
		Advanced Data Structures and Algorithms	✓	✓	✓	✓			✓		
		Software Architecture	✓		✓	✓					
		Advances in Operating Systems			✓	✓					
		Software Requirements Engineering	✓		✓	✓					
		Advanced Data Structures and Algorithms Lab	✓			✓	✓				
		Professional Practices							✓		✓
	SEM 2	Software Testing and Quality Assurance						✓		✓	
		Machine Learning Techniques	✓	✓	✓	✓	✓			✓	
		Software Reliability Metrics and Models			✓	✓	✓			✓	
		Software Security	✓		✓	✓	✓				
		Advances Database Management Systems			✓		✓				
		Elective I									
Software Development Lab		✓		✓	✓	✓					
YEAR 2	SEM 1	Integrated Software Project Management				✓	✓		✓		
		Elective II									
		Elective III									
		Elective IV									
	Project Work Phase I	✓		✓	✓	✓			✓		
	SEM 2	Project Work Phase II	✓		✓	✓	✓			✓	

PROGRESS THROUGH KNOWLEDGE

Attested

Sobhan
DIRECTOR

ANNA UNIVERSITY, CHENNAI
UNIVERSITY DEPARTMENTS
M.E. SOFTWARE ENGINEERING
REGULATIONS – 2015
CHOICE BASED CREDIT SYSTEM
CURRICULA AND SYLLABI

SEMESTER - I

SL. No	COURSE CODE	COURSE TITLE	CATEGORY	CONTACT PERIODS	L	T	P	C
THEORY								
1.	MA7153	Advanced Mathematics for Computing	FC	4	4	0	0	4
2.	CP7151	Advanced Data Structures and Algorithms	PC	3	3	0	0	3
3.	SW7151	Software Architecture	PC	3	3	0	0	3
4.	CP7153	Advances in Operating Systems	PC	3	3	0	0	3
5.	SW7101	Software Requirements Engineering	PC	3	3	0	0	3
PRACTICALS								
6.	CP7161	Advanced Data Structures and Algorithms Lab	PC	4	0	0	4	2
7.	CP7162	Professional Practices	EEC	2	0	0	2	1
TOTAL				22	16	0	6	19

II SEMESTER

SL. No	COURSE CODE	COURSE TITLE	CATEGORY	CONTACT PERIODS	L	T	P	C
THEORY								
1.	SW7251	Software Testing and Quality Assurance	PC	5	3	0	2	4
2.	CP7253	Machine Learning Techniques	PC	5	3	0	2	4
3.	SW7201	Software Reliability Metrics and Models	PC	3	3	0	0	3
4.	SW7202	Software Security	PC	3	3	0	0	3
5.	SO7251	Advanced Database Management Systems	PC	3	3	0	0	3
6.		Elective I	PE	3	3	0	0	3
PRACTICALS								
7.	SW7211	Software Development Lab	PC	4	0	0	4	2
TOTAL				26	18	0	10	23

III SEMESTER

SL. No	COURSE CODE	COURSE TITLE	CATEGORY	CONTACT PERIODS	L	T	P	C
THEORY								
1.	SW7301	Integrated Software Project Management	PC	3	3	0	0	3
2.		Elective II	PE	3	3	0	0	3
3.		Elective III	PE	3	3	0	0	3
4.		Elective IV	PE	3	3	0	0	3
PRACTICALS								
5.	SW7311	Project Work Phase I	EEC	12	0	0	12	6
TOTAL				24	12	0	12	18

IV SEMESTER

SL.NO	COURSE CODE	COURSE TITLE	CATEGORY	CONTACT PERIODS	L	T	P	C
PRACTICALS								
1.	SW7411	Project Work Phase II	EEC	24	0	0	24	12
TOTAL				24	0	0	24	12

TOTAL NO. OF CREDITS: 71

FOUNDATION COURSES (FC)

SL. NO	COURSE CODE	COURSE TITLE	CATEGORY	CONTACT PERIODS	L	T	P	C
1.		Advanced Mathematics for Computing	FC	4	4	0	0	4

PROFESSIONAL CORE (PC)

SL.No	COURSE CODE	COURSE TITLE	CATEGORY	CONTACT PERIODS	L	T	P	C
1.		Advanced Data Structures and Algorithms	PC	3	3	0	0	3
2.		<u>Software Architecture</u>	PC	3	3	0	0	3
3.		Advances in Operating Systems	PC	3	3	0	0	3
4.		Software Requirements Engineering	PC	3	3	0	0	3
5.		Advanced Data Structures and Algorithms Lab	PC	4	0	0	4	2
6.		Software Testing and Quality Assurance	PC	5	3	0	2	4
7.		Machine Learning Techniques	PC	5	3	0	2	4
8.		Software Reliability Metrics and Models	PC	3	3	0	0	3
9.		Software security	PC	3	3	0	0	3
10.		Advanced Database Management Systems	PC	3	3	0	0	3
11.		Software Development Lab	PC	4	0	0	4	2
12.		Integrated Software Project Management	PC	3	3	0	0	3

PROFESSIONAL ELECTIVES (PE)

SL. NO	COURSE CODE	COURSE TITLE	CATEGORY	CONTACT PERIODS	L	T	P	C
1.	SW7001	Agile Software Engineering	PE	3	3	0	0	3
2.	SW7002	Business Intelligence	PE	3	3	0	0	3
3.	SW7003	Enterprise Application Integration	PE	3	3	0	0	3
4.	SW 7004	Enterprise Resource Planning Concepts	PE	3	3	0	0	3
5.	SW7005	Formal Methods in Software Engineering	PE	3	3	0	0	3
6.	SW7006	Knowledge Management	PE	3	3	0	0	3
7.	SW7007	Managing Human Resource	PE	3	3	0	0	3
8.	SW7009	Principles of Supply Chain Management	PE	3	3	0	0	3
9.	SW7010	Software Agents	PE	3	3	0	0	3
10.	SW7011	Software Test Automation	PE	3	3	0	0	3
11.	SW7014	User Interface Design and Evaluation	PE	3	3	0	0	3
12.	SW7015	Web Design and Management	PE	3	3	0	0	3
13.	CP7251	Cloud Computing Technologies	PE	3	3	0	0	3
14.	CP7089	Real Time Systems Design	PE	3	3	0	0	3
15.	SW7012	Software Verification and Validation	PE	3	3	0	0	3
16.	CP7092	Social Network Mining and Analysis	PE	3	3	0	0	3
17.	CP7254	Security Principles and Practices	PE	3	3	0	0	3
18.	CP7076	Data Mining Techniques	PE	3	3	0	0	3
19.	CP7095	Virtualization Techniques and Applications	PE	3	3	0	0	3
20.	CP7082	Information Retrieval Techniques	PE	3	3	0	0	3
21.	CP7083	Internet of Things In The Cloud	PE	3	3	0	0	3
22.	SW7013	Test Driven Development	PE	3	3	0	0	3
23.	SW7008	Personal Software Process	PE	3	3	0	0	3

EMPLOYABILITY ENHANCEMENT COURSES (EEC)

SL. No	COURSE CODE	COURSE TITLE	CATEGORY	CONTACT PERIODS	L	T	P	C
1.		Professional Practices	EEC	2	0	0	2	1
2.		Project Work Phase I	EEC	12	0	0	12	6
3.		Project Work Phase II	EEC	24	0	0	24	12



OBJECTIVES:

- To understand the basics of random variables and standard distributions
- To understand the arrival process and various queuing 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 QUEUING 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
- Design appropriate queuing model for a given problem/system situation
- 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 Queuing Theory", John Wiley and Sons, New York, 1998.
7. J.Medhi," Stochastic models of Queuing Theory", Academic Press, Elsevier, Amsterdam, 2003

OBJECTIVES:

- To extend the students' knowledge of algorithms and data structures
- To enhance their expertise in algorithmic analysis and algorithm design techniques.
- To learn a variety of useful algorithms and techniques
- To extrapolate from them in order to 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 – Deaps – 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 PERIODS**OUTCOMES:**

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

- Have a basic ability to analyze algorithms and to determine algorithm correctness and time efficiency
- Master a variety of advanced data structures and their implementations and different algorithm design techniques in computational geometry and in parallel algorithms
- Apply and implement the learnt algorithm design techniques and data structures to solve problems

REFERENCES:

1. Ellis Horowitz, Sartaj Sahni, Dinesh Mehta, "Fundamentals of Data Structures in C", Silicon Pr, 2007.
2. Gilles Brassard, Paul Bratley, "Algorithmics: Theory and Practice", Prentice Hall, 1988.
3. Mark de Berg, Otfried Cheong, Marc van Kreveld, Mark Overmars, "Computational Geometry Algorithms and Applications", Third Edition, 2008.
4. J.A. Storer, "An Introduction to Data Structures and Algorithms", Birkhäuser Boston, 2002.
5. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, "Introduction to Algorithms", MIT Press, 2009.

SW7151

SOFTWARE ARCHITECTURE

L T P C
3 0 0 3

OBJECTIVES:

The student should be able to

- Understand the fundamentals of software architecture.
- Study the various software development methodologies.
- Learn the various software architecture design components.
- Relate software architecture and software quality.

UNIT I INTRODUCTION 9

Basic Concepts of Software Architecture - Architecture business cycle - architectural patterns - reference models - architectural structures, views - Introduction to Styles - Simple Styles - Distributed and Networked Architectures-Architecture for network based applications - 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 ARCHITECTURE DESCRIPTION DOCUMENTATION AND EVALUATION 9

Early Architecture Description Languages –Domain and Style Specific ADLs –Extensible ADLs - Documenting Software architecture -Architecture Evaluation –ATAM.

UNIT IV ARCHITECTURE DESIGN 9

Typical Architectural Design - Data Flow - Independent Components - Call and Return - Using Styles in Design – choices of styles – Architectural design space – Theory of Design Spaces – Design space of Architectural Elements – Design space of Architectural styles.

UNIT V CREATING AN ARCHITECTURE 9

Understanding Quality Attributes - Functionality and Architecture –Architecture and Quality Attributes-System Quality Attributes –Quality attribute Scenarios in Practice - Introducing Tactics - Availability Tactics –Modifiability Tactics –Performance Tactics -Security Tactics –Testability Tactics –Usability Tactics –Relationship of Tactics to Architectural Patterns –Architectural Patterns and Styles.

TOTAL : 45 PERIODS

OUTCOMES:

At the end the student will be able to

- Develop Software applications starting from software architecture and design.
- Learn and evaluate existing software architectures.
- 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.
2. David Budgen, "Software Design", Second Edition, Pearson Education, 2004.
3. Richard N.Taylor, NenadMedvidovic and Eric M.Dashofy, "Software Architecture, Foundations, Theory and Practice", Wiley 2010.
4. Hong Zhu, "Software Design Methodology from Principles to Architectural Styles", Elsevier, 2005.
5. Mary shaw and David Garlan, Software Architecture –Perspectives on an emerging discipline, Pearson education, 2008.

CP7153

ADVANCES IN OPERATING SYSTEMS

L T P C
3 0 0 3

OBJECTIVES

- To understand the concepts of distributed systems
- To get an insight into the various issues and solutions in distributed operating systems
- To learn about mobile and real-time operating systems
- To gain knowledge on the design concepts of mainframe operating systems

UNIT I BASICS OF OPERATING SYSTEMS 9

Overview – Synchronization Mechanisms – Processes and Threads – Process Deadlocks – Issues in Distributed Operating Systems – Communication Primitives – Limitations of a Distributed System

UNIT II DISTRIBUTED OPERATING SYSTEMS 9

Lamport's Logical Clocks – Vector Clocks – Causal Ordering of Messages – Distributed Mutual Exclusion Algorithms – Centralized, Distributed and Hierarchical Deadlock Detection Algorithms – Agreement Protocols.

UNIT III DISTRIBUTED RESOURCE MANAGEMENT 9

Distributed File Systems – Design Issues – Google File System – Hadoop Distributed File System – Distributed Shared Memory – Algorithms for Implementing Distributed Shared Memory – Load Distributed Algorithms – Issues in Task Migration – Synchronous and Asynchronous Check Pointing and Recovery – Fault Tolerance – Two-Phase Commit Protocol – Nonblocking Commit Protocol.

UNIT IV MOBILE AND REAL TIME OPERATING SYSTEMS 9

Basic Model of Real Time Systems – Characteristics – Applications of Real Time Systems – Real Time Task Scheduling – Handling Resource Sharing. Mobile Operating Systems – Architecture – Layers – Microkernel Design – Kernel Extensions – Processes and Threads – Memory Management – File system – Android – iOS.

UNIT V MAINFRAME AND LINUX OPERATING SYSTEMS 9

Mainframe – z/OS – Overview of z/OS Facilities – Virtual Storage and other Mainframe Concepts – Workload Management – I/O and Data Management – Supervising the Execution of Work in the System – Cross-memory Services – Characteristics of z/OS. Linux – Design Principles – Kernel Modules – Process Management – Scheduling – Memory Management – I/O Management – File System – Interprocess Communication

TOTAL : 45 PERIODS

OUTCOMES:

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

- Demonstrate the various protocols of distributed operating systems
- Identify the different features of mobile and real-time operating systems
- Discuss the various features of mainframe operating systems

REFERENCES:

1. Mukesh Singhal, Niranjana Shivaratri, "Advanced Concepts in Operating Systems – Distributed, Database and Multiprocessor Operating Systems", Tata McGraw-Hill, 2001.
2. Rajib Mall, "Real-Time Systems: Theory and Practice", Prentice Hall, 2006.
3. Neil Smyth, "iPhone iOS 4 Development Essentials – Xcode", Payload Media, Fourth Edition, 2011.
4. Nikolay Elenkov, "Android Security Internals: An In-Depth Guide to Android's Security Architecture", No Starch Press, 2014.
5. Jonathan Levin, "Mac OS X and iOS Internals: To the Apple's Core", John Wiley & Sons, 2012.
6. Andrew S. Tanenbaum and Herbert Bos, "Modern Operating Systems", Fourth Edition, Prentice Hall, 2014.
7. Mike Ebberts, John Kettner, Wayne O'Brien, Bill Ogden, "Introduction to the New Mainframe: z/OS Basics", Third Edition, International Business Machines Corporation, 2011.
8. Abraham Silberschatz, Peter B. Galvin, Greg Gagne, "Operating System Concepts", Wiley, Eighth edition, 2008.

SW7101

SOFTWARE REQUIREMENTS ENGINEERING

LT PC
3 0 0 3

OBJECTIVES

The student should be able to

- Understand the basics of requirements engineering
- Learn different techniques used for requirements elicitation
- Know the role played by requirements analysis in requirement integration
- Appreciate the use of various methodologies for requirements development
- Study the current trends in requirements prioritization and validation.

UNIT I REQUIREMENTS ENGINEERING OVERVIEW 9

Software Requirement Overview – Software Development Roles –Software Development Process
Kernels – Commercial Life Cycle Model – Vision Development – Stakeholders Needs & Analysis –
Stakeholder needs –Stakeholder activities.

UNIT II REQUIREMENTS ELICITATION 9

The Process of Requirements Elicitation – Requirements Elicitation Problems – Problems of Scope –
Problems of Understanding – Problems of Volatility – Current Elicitation Techniques – Information
Gathering – Requirements Expression and Analysis – Validation – An Elicitation Methodology
Framework – A Requirements Elicitation Process Model – Methodology over Method – Integration of
Techniques – Fact-Finding – Requirements Gathering – Evaluation and Rationalization – Prioritization
– Integration and Validation.

UNIT III REQUIREMENTS ANALYSIS 9

Identification of Functional and Non Functional Requirements – Identification of Performance
Requirements – Identification of safety Requirements – 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 – Case tools for requirements engineering.

TOTAL : 45 PERIODS

OUTCOMES

At the end the student will be able to:

- Prepare SRS including the details of requirements engineering
- Describe the stages of requirements elicitation
- Analyze software requirements gathering

REFERENCES

1. Ian Sommerville, Pete Sawyer, "Requirements Engineering: A Good Practice Guide", Sixth Edition, Pearson Education, 2004
2. Dean Leffingwe, Don Widrig, "Managing Software Requirements A Use Case Approach", Second Addition, Addison Wesley, 2003
3. Karl Eugene Wiegers, "Software Requirements", Word Power Publishers, 2000
4. Ian Graham, "Requirements Engineering and Rapid Development", Addison Wesley, 1998
5. Wiegers, Karl, Joy Beatty, "Software requirements", Pearson Education, 2013

CP7161

**ADVANCED DATA STRUCTURES AND ALGORITHMS
LAB**

L	T	P	C
0	0	4	2

OBJECTIVES:

- To understand heap and various tree structures like AVL, Red-black, B and Segment trees
 - To know the problems such as line segment intersection, convex shell and Voronoi diagram
1. Min/Max Heap
 2. Leftist Heap
 3. AVL Trees
 4. Red-Black Trees
 5. B-Trees
 6. Segment Trees
 7. Line segment intersection
 8. Convex Hull
 9. Voronoi Diagram

TOTAL : 60 PERIODS

OUTCOMES

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

- Implement heap and various tree structure like AVL, Red-black, B and Segment trees
- Solve the problems such as line segment intersection, convex shell and Voronoi diagram

OBJECTIVES:

- To facilitate analysis, design and problem solving skills
- To have a thorough domain knowledge
- To understand the best Industry practices by reading case studies
- To kindle innovative and professional thinking
- To explore possible alternative solutions
- To estimate feasibility, cost, risk and ROI

Identify an application (may be of social relevance) – Understand customer requirements – analyze and understand customers and stakeholders – value additions – innovations and research component – preparing plan / SRS document indicating feasibility, cost, risk, ROI and related design – suggest implementation methodology – perform risk assessment and management

TOTAL : 30 PERIODS**OUTCOMES:**

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

- Identify and formulate the problem
- Describe the background of the problem
- Assess the needs of stakeholders
- Make estimates like cost, risk, ROI etc., to justify the business opportunity.
- Describe the industry standards and procedures
- Predict the business opportunity
- Suggest system implications

SW7251

SOFTWARE TESTING AND QUALITY ASSURANCE

L	T	P	C
3	0	2	4

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 in making 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.

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 + 30 = 75 PERIODS

OUTCOMES

At the end the student will be able to

- Analyze the product Quality.
- Use various testing methods.
- 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, 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.

CP7253

MACHINE LEARNING TECHNIQUES

L	T	P	C
3	0	2	4

OBJECTIVES:

- To understand the concepts of machine learning
- To know supervised and unsupervised learning and their applications
- To learn the theoretical and practical aspects of Probabilistic Graphical Models
- To appreciate the concepts and algorithms of reinforcement learning
- To learn aspects of computational learning theory

UNIT I INTRODUCTION

8+6

Machine Learning - Machine Learning Foundations –Overview – Design of a Learning system - Types of machine learning –Applications Mathematical foundations of machine learning - random variables and probabilities - Probability Theory – Probability distributions -Decision Theory- Bayes Decision Theory - Information Theory

UNIT II SUPERVISED LEARNING**10+6**

Linear Models for Regression - Linear Models for Classification – Naïve Bayes - Discriminant Functions - Probabilistic Generative Models - Probabilistic Discriminative Models - Bayesian Logistic Regression. Decision Trees - Classification Trees- Regression Trees - Pruning. Neural Networks - Feed-forward Network Functions - Back-propagation. Support vector machines - Ensemble methods- Bagging- Boosting

UNIT III UNSUPERVISED LEARNING**8+6**

Clustering- K-means - EM Algorithm- Mixtures of Gaussians. The Curse of Dimensionality - Dimensionality Reduction - Factor analysis - Principal Component Analysis - Probabilistic PCA- Independent components analysis

UNIT IV PROBABILISTIC GRAPHICAL MODELS**10 + 6**

Graphical Models - Undirected graphical models - Markov Random Fields - Directed Graphical Models - Bayesian Networks - Conditional independence properties - Inference – Learning- Generalization - Hidden Markov Models - Conditional random fields(CRFs)

UNIT V ADVANCED LEARNING**9+6**

Sampling – Basic sampling methods – Monte Carlo. Reinforcement Learning- K-Armed Bandit- Elements - Model-Based Learning- Value Iteration- Policy Iteration. Temporal Difference Learning- Exploration Strategies- Deterministic and Non-deterministic Rewards and Actions Computational Learning Theory - Mistake bound analysis, sample complexity analysis, VC dimension. Occam learning, accuracy and confidence boosting

TOTAL : 45 + 30 = 75 PERIODS**OUTCOMES:****Upon completion of this course, the student should be able to**

- 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
- Implement typical clustering algorithms for different types of applications
- Design and implement an HMM for a sequence model type of application
- Identify applications suitable for different types of machine learning with suitable justification

REFERENCES:

1. Christopher Bishop, "Pattern Recognition and Machine Learning" Springer, 2007.
2. Kevin P. Murphy, "Machine Learning: A Probabilistic Perspective", MIT Press, 2012.
3. Ethem Alpaydin, "Introduction to Machine Learning", MIT Press, Third Edition, 2014.
4. Tom Mitchell, "Machine Learning", McGraw-Hill, 1997.
5. Trevor Hastie, Robert Tibshirani, Jerome Friedman, "The Elements of Statistical Learning", Springer, Second Edition, 2011.
6. Stephen Marsland, "Machine Learning - An Algorithmic Perspective", Chapman and Hall/CRC Press, Second Edition, 2014.

OBJECTIVES**The student should be able to**

- Learn different definitions of software quality
- Know different notions of defects and classify them
- Understand the basic techniques of data collection and how to apply them
- Learn software metrics that define relevant metrics in a rigorous way.
- 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 MEASURING SOFTWARE PRODUCT 9

Measurement of Internet Product Attributes – Size and Structure – External Product Attributes – Measurement of Quality – Software Reliability: Measurement and Prediction

TOTAL : 45 PERIODS**OUTCOMES:****At the end the student will be able to**

- Perform some simple statistical analysis relevant to software measurement data.
- Use from practical examples both the benefits and limitations of software metrics for quality control and assurance

REFERENCES:

1. Norman Fenton, James Bieman, “Software Metrics: A Rigorous and Practical Approach”, 3rd edition, CRC Press, 2015
2. John D. Musa, Anthony Iannino, Kazuhira Okumoto, “Software Reliability – Measurement, Prediction, Application, Series in Software Engineering and Technology”, McGraw Hill, 1987
3. John D. Musa, “Software Reliability Engineering”, Tata McGraw Hill, 1999

OBJECTIVES

The student should be able to

- Know the importance and need of software security
- Know about various attacks
- Learn about secure software design
- Understand risk management in secure software development
- Know the working of tools related to software security

UNIT I INTRODUCTION**9**

Need for software security – Memory based attacks – low level attacks against heap and stack - stack smashing – format string attacks – stale memory access attacks – ROP (Return oriented programming) – malicious computation without code injection. Defense against memory based attacks – stack canaries – non-executable data - address space layout randomization (ASLR), memory-safety enforcement, control-flow Integrity (CFI) – randomization

UNIT II SECURE DESIGN**9**

Isolating the effects of untrusted executable content - stack inspection – policy specification languages – vulnerability trends – buffer overflow – code injection - Generic network fault injection – local fault injection - SQL injection - Session hijacking. Secure design - threat modeling and security design principles - good and bad software design - Web security-browser security: cross-site scripting (XSS) , cross-site forgery (CSRF) – database security – file security

UNIT III SECURITY RISK MANAGEMENT**9**

Risk Management Life cycle – Risk Profiling – Risk exposure factors – Risk Evaluation and Mitigation - Risk Assessment Techniques – Threat and Vulnerability Management.

UNIT IV SECURITY TESTING**9**

Traditional software testing – comparison - secure software development life cycle - risk based security testing – prioritizing security testing with threat modeling – shades of analysis: white, grey and black box testing.

UNIT V ADVANCED SOFTWARE SECURITY**9**

Advanced penetration testing – planning and scoping – DNS groper – DIG (Domain Information Graph) – Enumeration – Remote Exploitation – Web Application Exploitation - Exploits and Client side Attacks – Post Exploitation – Bypassing Firewalls and Avoiding Detection - Tools for penetration testing

TOTAL : 45 PERIODS**OUTCOMES:**

At the end the student will be able to

- Use tools for securing software
- Apply security principles in software development
- Involve selection of testing techniques related to software security in testing phase of software development

REFERENCES:

1. Robert C. Seacord, "Secure Coding in C and C++ (SEI Series in Software Engineering)", Addison-Wesley Professional, 2005
2. Mike Shema, "Hacking Web Apps: Detecting and Preventing Web Application Security Problems", First edition, Syngress Publishing, 2012
3. Bryan Sullivan and Vincent Liu, "Web Application Security, A Beginner's Guide", Kindle Edition, McGraw Hill, 2012
4. Evan Wheeler, "Security Risk Management: Building an Information Security Risk Management Program from the Ground Up", First edition, Syngress Publishing, 2011
5. Chris Wysopal, Lucas Nelson, Dino Dai Zovi, and Elfriede Dustin, "The Art of Software Security Testing: Identifying Software Security Flaws (Symantec Press)", Addison-Wesley Professional, 2006
6. Lee Allen, "Advanced Penetration Testing for Highly-Secured Environments: The Ultimate Security Guide (Open Source: Community Experience Distilled)", Kindle Edition, Packt Publishing, 2012

SO7251

ADVANCED DATABASE MANAGEMENT SYSTEMS

L	T	P	C
3	0	0	3

OBJECTIVES:

- To understand the underlying principles of Relational Database Management System.
- To understand and implement the advanced features of DBMS.
- To develop database models using distributed databases.
- To implement and maintain an efficient database system using emerging trends.

UNIT I RELATIONAL MODEL

9

Data Model – Types of Data Models: – Entity Relationship Model – Relational Data Model – Mapping Entity Relationship Model to Relational Model – Structured Query Language – Database Normalization – Transaction Management.

UNIT II PARALLEL AND DISTRIBUTED DATABASES

9

Centralized and Client-Server Architectures – Parallel Systems – Distributed Systems – Parallel Databases – I/O Parallelism – Inter- and Intra-Query Parallelism – Inter- and Intra-operation Parallelism – Distributed Database Concepts: – Distributed Data Storage – Distributed Transactions – Commit Protocols – Concurrency Control – Distributed Query Processing.

UNIT III XML DATABASES

9

XML Databases: XML Data Model – DTD – XML Schema – XML Querying – Web Databases – Open Database Connectivity.

UNIT IV MULTIMEDIA DATABASES

9

Multidimensional Data Structures – Image Databases – Text / Document Databases – Video Databases – Audio Databases – Multimedia Database Design.

UNIT V CURRENT ISSUES

9

Active Databases – Deductive Databases – Data Warehousing – Data Mining – Database Tuning – Database Security

TOTAL : 45 PERIODS

OUTCOMES:

On successful completion of this course, the student will be able to:

- Design and implement relational databases, distributed databases, XML databases and multimedia databases.
- Implement the concept of database connectivity with the applications.

REFERENCES

1. R. Elmasri, S.B. Navathe, "Fundamentals of Database Systems", Addison-Wesley, 2011.
2. Thomas Cannolly and Carolyn Begg, "Database Systems, A Practical Approach to Design, Implementation and Management", Third Edition, Pearson Education, 2007.
3. Henry F Korth, Abraham Silberschatz, S. Sudharshan, "Database System Concepts", Fifth Edition, McGraw Hill, 2006.
4. C.J.Date, A.Kannan and S.Swamynathan, "An Introduction to Database Systems", Eighth Edition, Pearson Education, 2006.
5. V.S.Subramanian, "Principles of Multimedia Database Systems", Harcourt India Pvt. Ltd., 2001.

SW7211

SOFTWARE DEVELOPMENT LAB

L	T	P	C
0	0	4	2

OBJECTIVES:

- To learn the stages of software development
- To know about preparing software project documentation

LIST of EXERCISES:

60

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, Site
check and Site monitor. – TEST CASE DOCUMENTATION

TOTAL : 60 PERIODS

SUGGESTED LIST OF APPLICATIONS:

Student Marks Analyzing System.
Quiz System.
Online Ticket Reservation System
Payroll System
Course Registration System
Stock Maintenance.

OUTCOMES:

At the end the student will be able to

- Prepare project plan, SRS, design document, code document and test case documentations at appropriate stages of software development.

SW7301	INTEGRATED SOFTWARE PROJECT MANAGEMENT	L	T	P	C
		3	0	0	3

OBJECTIVES

The student should be able to

- 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) - ISO 9000: 2000 - 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.

UNIT IV 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**OUTCOMES****At the end 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. Murali Chemuturi, Thomas M. Cagley, “Mastering Software Project Management: Best Practices, Tools and Techniques”, J. Ross Publishing, 2010
2. Stark, John, “Decision Engineering: Product Lifecycle Management:21st Century Paradigm for Product Realisation”, 2nd Edition., Springer London, 2011
3. Antonio Borghesi, Barbara Gaudenzi, “Risk Management: How to Assess, Transfer and Communicate Critical Risks: Perspectives in Business Culture”, Illustrated Edition, Springer, 2012
4. Norman Fenton, James Bieman, “Software Metrics: A Rigorous and Practical Approach”, 3rd edition, CRC Press, 2015.

SW7001**AGILE SOFTWARE ENGINEERING**

L	T	P	C
3	0	0	3

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 - 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 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**OUTCOMES:****At the end the student will be able to**

- The know importance of interacting with business stakeholders in determining the requirements for a software system.
- Apply iterative software development process
- Apply the impact of social aspects on software development success.

REFERENCES:

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

SW7002**BUSINESS INTELLIGENCE**

L	T	P	C
3	0	0	3

OBJECTIVES:**The student should be able 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
- Understand and help develop the strategies of modern enterprise decision makers
- Acquire knowledge in many scientific and technological fields including data warehouses, data mining, content analytics, business process management, visual analytics
- Gain 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 - BI 2.0 – GoOLAP Fact Retrieval Framework.

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

Knowledge Discovery for BI – Markov Logic Networks – BI Search and Text Analytics – Advanced Visualisation – Semantic Web Technologies for building BI - Service oriented BI – Collaborative BI - Evaluating BI – Stakeholder model of BI.

TOTAL : 45 PERIODS

OUTCOMES:

At the end the student will be able to

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

REFERENCES:

1. CindiHowson,"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. David Loshin," Business Intelligence: The Savvy Manager's Guide", Morgan Kaufmann, 2nd Edition, Newnes Publishers, 2012
5. Rajiv Sabherwal, Irma Becerra-Fernandez, "Business Intelligence", illustrated Edition, John Wiley & Sons, 2010
6. Elizabeth Vitt, Michael Luckevich, Stacia Misner, "Business Intelligence", O'Reilly Media, Inc., 2010.
7. Marie - Aude Aufaure, Esteban Zimány, "Business Intelligence", First European Summer School eBISS, 2011.

OBJECTIVES

The student should be able to

- Create a Business Data Catalog Application Definition file
- Discuss the most important topics and technologies related to enterprise application integration
- Practice implementation strategies.
- Know 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 PATTERNS OF ENTERPRISE APPLICATION INTEGRATION 9

Domain Logic Patterns – Data source architectural patterns – Object Relational Patterns – Web Presentation Patterns – Distribution Patterns – Concurrency Patterns – Session state patterns - base patterns.

TOTAL : 45 PERIODS

OUTCOMES:

At the end the student will be able to

- Use object-oriented concepts for developing web applications
- Demonstrate the enterprise application integration
- Deploy the components of EIA with solutions

REFERENCES

1. David S.Linthicum, —Enterprise Application Integration, Addison-Wesley Professional, 2000
2. Martin Fowler, "Patterns of Enterprise Application Architecture", Pearson Education, 17th edition, 2011.
3. Gregor Hohpe, Bobby Woolf, "Enterprise Integration Patterns: Designing, Building, and Deploying Messaging Solutions", Pearson Education, 15th edition, 2011.

OBJECTIVES:

The student should be able to:

- Know the concepts and knowledge of ERP.
- Focus on illustrating procurement, production, and sales business processes using ERP software.
- Bridge the gap between the need of business process knowledge and its application to the business environment.
- Learn ERP Implementation Success & Failure for an application.
- Appreciate ERP in various public and private sector.

UNIT I INTRODUCTION TO ERP 9

Overview – Benefits of ERP – ERP and Related Technologies – ERP Risks – Benefits - Data Warehousing – Data Mining – On-line Analytical Processing – Data Migration – ERP, Internet and WWW.

UNIT II ERP IMPLEMENTATION 9

Implementation Life Cycle – cost model - Implementation Methodology – Hidden Costs – Organizing Implementation – Vendors, Consultants and Users – Contracts – ERP Project Management and Monitoring - Business case and ROI analysis - ERP and business process reengineering.

UNIT III BUSINESS MODULES 9

Finance Management – Manufacturing Management – Human capital Management – Procurement and Inventory Management – Supplier Relationship Management – Supply chain planning & Management - Logistics Management - Plant Maintenance – Materials Management – Quality Management – Sales and Distribution – Enterprise Asset Management – Product Lifecycle Management.

UNIT IV ERP MARKET 9

ERP & E-business – ERP & CRM - ERP Market Place – SAP-ERP financials – Auditing ERP – ERP Business Intelligence and Performance Management – ERP for manufacturing: Auto, Pharma, Consumer Products, Mining – ERP for service sector: Retail, Healthcare, Telecom, Banking, Insurance, Educational Institutions.

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 – Managing ERP security.

TOTAL : 45 PERIODS

OUTCOMES:

At the end the student will be able to

- Become a manager of computer service offerings, across business processes of an organization.
- Understand the structure of an ERP system
- Prepare them to become knowledgeable ERP user professionals suitable to Industry and Information Technology Companies.

REFERENCES:

1. Alexis Leon, "Enterprise Resource Planning", Tata McGraw Hill, 2nd Edition, 2008
2. Ray, "Enterprise Resource Planning", Tata McGraw Hill, 2011
3. Veena Bansal, "*Enterprise Resource Planning*", Pearson Education India. 2013
4. Marianne Bradford, "Modern ERP – Select, Implement and Use" – Today's Advanced Business Systems, North Carolina State University, Second Edition, 2010
5. V. Narayanan, "Implementing SAR-ERP Financials – A configuration Guide", Tata McGraw Hill, 2010
6. Joseph A. Brady, Ellen F. Monk, Bret J. Wangner, "Concepts in Enterprise Resource Planning", Thomson Learning, 2001.

SW7005

FORMAL METHODS IN SOFTWARE ENGINEERING

L	T	P	C
3	0	0	3

OBJECTIVES:

The student should be able to

- Translate the informal diagrammatic and textual notations into formal interpretation sufficient for particular forms of automated reasoning.
- Understand the formal semantics and tools.
- Understand VDM and Z notations.

UNIT I INTRODUCTION

9

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

9

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

UNIT III VDM

9

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

UNIT IV THE Z NOTATION

9

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

At the end the student will be able to

- Apply knowledge of logic, formal methods and software engineering methods.
- Use VDM and Z notation in the context of software engineering
- Assess the software against semantics and tools.

REFERENCES:

1. Jean-Francois Monin, "Understanding formal methods (FACIT)", Springer, 2002
2. Andrew Harry, "Formal Methods: Fact File VDM and Z", John Wiley and Sons, 1996.
3. Jim Woodcock, Jim Davies, "Using Z Specification, Refinement and Proof", Prentice Hall International, 1996.

SW7006	KNOWLEDGE MANAGEMENT	L	T	P	C
		3	0	0	3

OBJECTIVES

The student should be able to

- Learn knowledge engineering basics
- Know the knowledge models
- Know the techniques of knowledge management and implementation
- Learn the knowledge elicitation techniques
- Learn scope of knowledge management in project management.

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 9

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

UNIT IV KNOWLEDGE SYSTEM IMPLEMENTATION 9

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

UNIT V ADVANCED KNOWLEDGE MANAGEMENT 9

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

TOTAL : 45 PERIODS

OUTCOMES:

At the end the student will be able to

- Apply knowledge engineering basics.
- Design the knowledge models.
- Apply the techniques of knowledge management and implementation.

REFERENCES:

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
3. Debowski, Shelda, "Knowledge Management: A Strategic Management Perspective", John Wiley & Sons Ltd, 2005
4. Awad, Elias M., and Hassan M. Ghaziri. "Knowledge Management", Prentice Hall; United States ed edition, 2011
5. C.W. Holsapple, "Handbooks on Knowledge ManagementII", International Handbooks on Information Systems, Vol 1 and 2, 2003.

SW7007

MANAGING HUMAN RESOURCE

L	T	P	C
3	0	0	3

OBJECTIVES:

The student should be able to

- Know the importance of resources for a task.
- Compare all the resources with Human resources so the employee constraints are checked to meet the completion of the task.
- Study Training types
- Forecast Human Resource requirement.
- 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:

At the end the student will be able to

- Apply recruitment techniques in employee selection process.
- Apply appropriate training process
- Motivate employees by sustaining employee Interests.

REFERENCES:

1. Decenzo and Robbins, "Human Resource Management", Wilsey, 6th edition, 2001
2. BiswajeetPattanayak, "Human Resource Management", Prentice Hall of India, 2001
3. Human Resource Management, EugenceMckenna 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
8. Mejia, Luis Gomez, David Balkin, and Robert Cardy, "Managing human resources", 7th edition, New Delhi: Prentice-Hall of India, 2012.

SW7009

PRINCIPLES OF SUPPLY CHAIN MANAGEMENT

L	T	P	C
3	0	0	3

OBJECTIVES

The student should be able to

- Learn about the E-business environment driven by the Automation Software in quick movement of supply of products
- Study the fundamentals of supply chain management comprising of Inventory management and warehousing etc as co parts of entire business
- Learn the cost management for the supply of products
- Improve the overall organization performance and customer satisfaction by improving product or service delivery to consumer.

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:

At the end the student will be able to

- Learn about the e- business for supply of products managed with the appropriate methodologies and management techniques
- Know Supply Chain Management consisting of all parties (Including Manufacturer, Marketer, Suppliers, transporters, Warehouses, Retailers and even customers) directly or indirectly involved in fulfilment of a customer
- Ensure that the supply chains deliver varying degrees of six outcomes — the traditional cost-related benefit plus responsiveness, security, sustainability, resilience and innovation — depending on key customers' needs
- Know automated back office software systems
- Know 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, 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
5. Agarwal D. K., "Supply Chain Management: Strategy, Cases and Best Practices", Macmillan, 2010
6. Chandrasekaran, Nagarajan, "Supply Chain Management: Process, System, and Practice", Oxford University Press, 2010.

OBJECTIVES

The student should be able to

- Understand the basics of test automation
- Appreciate the different aspects of test tool evaluation and test automation approach selection
- Understand the role played by test planning and design in test execution
- Appreciate the use of various testing tools for testing varied applications
- 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:**

At the end the student will be able to

- Identify the different test tools
- 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 for automating test execution

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:****At the end the student will be able to**

- Design a more user friendly software.
- Utilize the existing functionalities provided and develop a better design.
- Predict the need of the end user and design the interface accordingly.

REFERENCES:

1. Wilbent. O. Galitz ,—The Essential Guide To User Interface Design, John Wiley& Sons, 2001
2. Alan Cooper, “The Essential of User Interface Design”, Wiley – Dream Tech Ltd., 2002
3. Sharp, Rogers, Preece, “Interaction Design”, Wiley India Edition, 2007
4. Carol M. Barnum, “Usability Testing Essentials: Ready, Set...Test”, Elsevier, 2010
5. Wilbert O. Galitz, The Essential Guide to User Interface Design – An Introduction to GUI Design Principles and techniques, 2007, wiley.

SW7015**WEB DESIGN AND MANAGEMENT**

L	T	P	C
3	0	0	3

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**OUTCOMES:****At the end the student will be able to**

- Apply various scripting languages for the development of web applications
- Follow Web design standards
- Develop websites for local community organizations.

REFERENCES:

1. Thomas A. Powell, "The Complete Reference – Web Design", Tata McGraw Hill, Third Edition, 2003
2. Ashley Friedlein, "Web Project ManagementII", 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 Duyn, Landay, 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
7. Wendy Willard, "Web Design: A Beginner's Guide", Second Edition, McGraw Hill Education (India) Private Limited, 2010

CP7251**CLOUD COMPUTING TECHNOLOGIES**

L	T	P	C
3	0	0	3

OBJECTIVES:

- To understand the concepts 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**

Evolution of Cloud Computing -System Models for Distributed and Cloud Computing - NIST Cloud Computing Reference Architecture -IaaS - On-demand Provisioning - Elasticity in Cloud - Examples of IaaS Providers - PaaS - Examples of PaaS Providers - SaaS - Examples of SaaS Providers - Public , Private and Hybrid Clouds – Google App Engine, Amazon AWS - Cloud Software Environments -Eucalyptus, Open Nebula, Open Stack, Nimbus

UNIT II VIRTUALIZATION**9**

Basics of Virtual Machines - Process Virtual Machines – System Virtual Machines – Emulation – Interpretation – Binary Translation - Taxonomy of Virtual Machines. Virtualization –Management Virtualization — Hardware Maximization – Architectures – Virtualization Management – Storage Virtualization – Network Virtualization

UNIT III VIRTUALIZATION INFRASTRUCTURE 9

Comprehensive Analysis – Resource Pool – Testing Environment –Server Virtualization – Virtual Workloads – Provision Virtual Machines –Desktop Virtualization – Application Virtualization – Work with AppV – Mobile OS for smart phones – Mobile Platform Virtualization – Collaborative Applications for Mobile platforms.

UNIT IV PROGRAMMING MODEL 9

Map Reduce Hadoop Distributed File Systems – Hadoop I/O – Developing Map Reduce Applications – Working of Map Reduce – Types and Formats – Setting up Hadoop Cluster.

UNIT V CLOUD INFRASTRUCTURE AND SECURITY 9

Architectural Design of Compute and Storage Clouds - Inter Cloud Resource Management - Resource Provisioning and Platform Deployment - Global Exchange of Cloud Resources - 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 this course, the student 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", McGraw-Hill Osborne Media, 2009.
4. Jim Smith, Ravi Nair, "Virtual Machines: Versatile Platforms for Systems and Processes", Elsevier/Morgan Kaufmann, 2005.
5. Danielle Ruest, Nelson Ruest, "Virtualization: A Beginner's Guide", McGraw-Hill Osborne Media, 2009.
6. Tom White, "Hadoop: The Definitive Guide", Yahoo Press, 2012.
7. Rajkumar Buyya, Christian Vecchiola, and Thamarai Selvi, "Mastering Cloud Computing", Tata McGraw Hill, 2013.

CP7089

REAL TIME SYSTEMS DESIGN

L	T	P	C
3	0	0	3

OBJECTIVES:

- To learn real time operating system concepts and the associated issues & techniques.
- To understand design and synchronization problems in Real Time System.
- To understand the evaluation techniques present in Real Time System.

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 SOFTWARE REQUIREMENTS ENGINEERING

9

Requirements engineering process – types of requirements – requirements specification for real time systems – Formal methods in software specification – structured Analysis and Design – object oriented analysis and design and unified modelling language – organizing the requirements document – organizing and writing documents – requirements validation and revision.

UNIT III INTERTASK COMMUNICATION AND MEMORY MANAGEMENT

9

Buffering data – Time relative Buffering- Ring Buffers – Mailboxes – Queues – Critical regions – Semaphores – other Synchronization mechanisms – deadlock – priority inversion – process stack management – run time ring buffer – maximum stack size – multiple stack arrangement – memory management in task control block - swapping – overlays – Block page management – replacement algorithms – memory locking – working sets – real time garbage collection – contiguous file systems.

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 PROGRAMMING LANGUAGES

9

Assembly language – procedural languages – OO languages – Brief survey of languages – Faults, Failures and bugs – Fault Tolerance – Software integration – refactoring Real time code.

TOTAL : 45 PERIODS

OUTCOMES:

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

- Apply principles of real time systems design.
- Make use of architectures and behavior of real time operating systems and database in real time applications.

OUTCOMES:

At the end 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:

1. MouradDebbabi, 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, 1995
4. Avner Engel, "Verification, Validation & Testing of Engineered Systems", Wiley series in systems Engineering and Management, 2010.

CP7092	SOCIAL NETWORK MINING AND	L	T	P	C
ANALYSIS		3	0	0	3

OBJECTIVES

- To gain knowledge about the current web development and emergence of social web
- To study about the modeling, aggregating and knowledge representation of semantic web
- To appreciate the use of machine learning approaches for web content mining
- To learn about the extraction and mining tools for social networks
- To gain knowledge on web personalization and web visualization of social networks

UNIT I INTRODUCTION TO SOCIAL NETWORK ANALYSIS AND KNOWLEDGE REPRESENTATION 9

Introduction to Web - Limitations of current Web – Development of Semantic Web – Emergence of the Social Web - Network analysis - Development of Social Network Analysis - Key concepts and measures in network analysis- Knowledge Representation on the Semantic Web – Ontology languages for the Semantic Web – RDF and OWL - Modeling and aggregating social network data.

UNIT II SOCIAL MEDIA MINING 9

Data Mining Essential –Data Mining Algorithm - Web Content Mining – Supervised Learning – Decision tree - Naïve Bayesian Text Classification - Support Vector Machines - Ensemble of Classifiers. Unsupervised Learning - K-means Clustering - Hierarchical Clustering –Partially Supervised Learning – Markov Models - Probability-Based Clustering - Classification and Clustering – Vector Space Model – Latent semantic Indexing – Automatic Topic Extraction - Opinion Mining and Sentiment Analysis – Document Sentiment Classification

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 of security practices

UNIT I CLASSICAL CIPHERS 9

Classical Cryptography- Shift Cipher - Substitution Cipher - Affine Cipher – Cryptanalysis - Cryptanalysis of the Affine Cipher - Cryptanalysis of the Substitution Cipher - Cryptanalysis of the Vigenere Cipher - Shannon's Theory

UNIT II SYMMETRIC CIPHERS AND HASH FUNCTIONS 9

Substitution-Permutation Networks - Linear Cryptanalysis - Differential Cryptanalysis - Data Encryption Standard - Advanced Encryption Standard - Modes of Operation - Cryptography Hash Function - Hash Function and Data Integrity - Security of Hash Function - Iterated Hash Functions - Message Authentication Codes

UNIT III PUBLIC-KEY ENCRYPTION TECHNIQUES 9

Introduction to Public-key Cryptography - Number theory - RSA Cryptosystem - Attacks on RSA – El-Gamal Cryptosystem - Shanks' Algorithm - Elliptic Curves over the Reals - Elliptical Curves Modulo a Prime - Signature Scheme – Digital Signature Algorithm

UNIT IV KEY MANAGEMENT 9

Identification Scheme and Entity Attenuation - Challenge and Response in the Secret-key Setting - Challenge and Response in the Public key Setting - Schnorr Identification Scheme - Key distribution - Diffie-Hellman Key - Pre-distribution - Unconditionally Secure key Pre-distribution - Key Agreement Scheme - Diffie-Hellman Key agreement - Public key infrastructure - PKI, Certificates, Trust Models

UNIT V SECURITY PRACTICES 9

Transport-Level Security – SSL – TLS - HTTPS – SSH - Electronic Mail Security - Pretty Good Privacy - IP Security - IP Security Architecture – Authentication Header – Encapsulating Security Payload – Key Management - Legal and Ethical Issues

TOTAL : 45 PERIODS**OUTCOMES:****Upon completion of this course, the student should be able to**

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

REFERENCES:

1. Douglas R. Stinson, "Cryptography Theory and Practice", Third Edition, Chapman & Hall/CRC, 2006.
2. William Stallings, "Cryptography and Network Security: Principles and Practices", Sixth Edition, Pearson Education, 2013.
3. Wenbo Mao, "Modern Cryptography – Theory and Practice", Pearson Education, 2003.
4. Charles B. Pfleeger, Shari Lawrence Pfleeger, "Security in Computing", Fourth Edition, Pearson Education, 2007.
5. Wade Trappe and Lawrence C. Washington, "Introduction to Cryptography with Coding Theory" Second Edition, Pearson Education, 2007.

OBJECTIVES

- To understand data mining principles and techniques and Introduce DM as a cutting edge business intelligence
- To expose the students to the concepts of data warehousing architecture and implementation
- To study the overview of developing areas – web mining, text mining and ethical aspects of data mining
- To identify business applications and trends of data mining

UNIT I INTRODUCTION TO DATA WAREHOUSING**9**

Evolution of Decision Support Systems- Data warehousing Components – Building a Data warehouse, Data Warehouse and DBMS, Data marts, Metadata, Multidimensional data model, OLAP vs OLTP, OLAP operations, Data cubes, Schemas for Multidimensional Database: Stars, Snowflakes and Fact constellations

UNIT II DATA WAREHOUSE PROCESS AND ARCHITECTURE**9**

Types of OLAP servers, 3-Tier data warehouse architecture, distributed and virtual data warehouses. Data warehouse implementation, tuning and testing of data warehouse. Data Staging (ETL) Design and Development, data warehouse visualization, Data Warehouse Deployment, Maintenance, Growth, Business Intelligence Overview- Data Warehousing and Business Intelligence Trends - Business Applications- tools-SAS

UNIT III INTRODUCTION TO DATA MINING**9**

Data mining-KDD versus data mining, Stages of the Data Mining Process-task primitives, Data Mining Techniques -Data mining knowledge representation – Data mining query languages, Integration of a Data Mining System with a Data Warehouse – Issues, Data preprocessing – Data cleaning, Data transformation, Feature selection, Dimensionality reduction, Discretization and generating concept hierarchies-Mining frequent patterns- association-correlation

UNIT IV CLASSIFICATION AND CLUSTERING**9**

Decision Tree Induction - Bayesian Classification – Rule Based Classification – Classification by Back propagation – Support Vector Machines – Associative Classification – Lazy Learners – Other Classification Methods – Clustering techniques – , Partitioning methods- k-means- Hierarchical Methods – distance based agglomerative and divisible clustering, Density-Based Methods – expectation maximization -Grid Based Methods – Model-Based Clustering Methods – Constraint – Based Cluster Analysis – Outlier Analysis.

UNIT V PREDICTIVE MODELING OF BIG DATA AND TRENDS IN DATAMINING**9**

Statistics and Data Analysis – EDA – Small and Big Data –Logistic Regression Model - Ordinary Regression Model-Mining complex data objects – Spatial databases – Temporal databases – Multimedia databases – Time series and sequence data – Text mining – Web mining – Applications in Data mining

TOTAL : 45 PERIODS**OUTCOMES:**

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

- Evolve multidimensional intelligent model from typical system
- Discover the knowledge imbibed in the high dimensional system
- Evaluate various mining techniques on complex data objects

REFERENCES:

1. Jiawei Han, Micheline Kamber, "Data Mining: Concepts and Techniques", Morgan Kaufmann, Third edition, 2011.
2. Alex Berson, Stephen J. Smith, "Data Warehousing, Data Mining & OLAP", Tata McGraw Hill, Tenth Reprint, 2007.
3. G. K. Gupta, "Introduction to Data Mining with Case Studies", Eastern Economy Edition, Prentice Hall of India, Third Edition, 2014.
4. Ian.H.Witten, Eibe Frank and Mark.A.Hall, "Data Mining: Practical Machine Learning Tools and Techniques", Morgan Kaufmann, Third edition, 2011.
5. Bruce Ratner, "Statistical and Machine - Learning Data Mining: Techniques for Better Predictive Modeling and Analysis of Big Data", CRC Press, Second Edition, 2012.
6. Mehmed kantardzic, "Data mining: Concepts, Models, Methods, and Algorithms", Wiley-Blackwell, Second Edition, 2011.
7. Ian Witten, Eibe Frank, "Data Mining: Practical Machine Learning Tools and Techniques", Third Edition, Morgan Kaufmann, 2011.
8. George M Marakas, "Modern Data Warehousing, Mining and Visualization: Core Concepts", Prentice Hall, 2002.

CP7095

VIRTUALIZATION TECHNIQUES AND APPLICATIONS

L T P C
3 0 0 3

OBJECTIVES

- To understand the concepts of virtualization and virtual machines
- To understand the implementation of process and system virtual machines
- To explore the aspects of high level language virtual machines
- To gain expertise in server, network and storage virtualization.
- To understand and deploy practical virtualization solutions and enterprise solutions

UNIT I OVERVIEW OF VIRTUALIZATION

9

System architectures - Virtual Machine basics - Process vs System Virtual Machines - Taxonomy. Emulation: Basic Interpretation - Threaded Interpretation - Precoded and Direct Threaded Interpretation - Binary Translation. System Virtual Machines - Key concepts - Resource utilization basics

UNIT II PROCESS VIRTUAL MACHINES

9

Implementation – Compatibility – Levels – Framework – State Mapping – Register – Memory Address Space – Memory Architecture Emulation – Memory Protection – Instruction Emulation – Performance Tradeoff - Staged Emulation – Exception Emulation – Exception Detection – Interrupt Handling – Operating Systems Emulation – Same OS Emulation – Different OS Emulation – System Environment

UNIT III HIGH LEVEL LANGUAGE VIRTUAL MACHINES AND SERVER VIRTUALIZATION

9

HLL virtual machines: Pascal P-Code – Object Oriented HLLVMs - Java VM architecture - Java Native Interface - Common Language Infrastructure. Server virtualization: Partitioning techniques - virtual hardware - uses of virtual servers - server virtualization platforms

UNIT IV NETWORK AND STORAGE VIRTUALIZATION

9

Design of Scalable Enterprise Networks – Layer2 Virtualization – VLAN - VFI - Layer 3 Virtualization – VRF - Virtual Firewall Contexts - Network Device Virtualization - Data- Path Virtualization - Routing Protocols. Hardware Devices – SAN backup and recovery techniques – RAID – Classical Storage Model – SNIA Shared Storage Model – Virtual Storage: File System Level and Block Level

UNIT V APPLYING VIRTUALIZATION**9**

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

TOTAL : 45 PERIODS**OUTCOMES:**

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

- Deploy legacy OS on virtual machines.
- Analyze the intricacies of server, storage and network virtualizations
- Design and develop applications on virtual machine platforms

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. Kenneth Hess , Amy Newman, "Practical Virtualization Solutions: Virtualization from the Trenches", Prentice Hall, 2010

CP7082**INFORMATION RETRIEVAL TECHNIQUES**

L	T	P	C
3	0	0	3

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

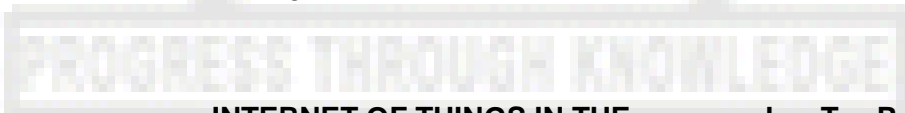
OUTCOMES:

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. Ricardo Baeza – Yates, Berthier Ribeiro – Neto, “Modern Information Retrieval: The concepts and Technology behind Search” (ACM Press Books), Second Edition, 2011.
2. Christopher D. Manning, Prabhakar Raghavan, Hinrich Schutze, “Introduction to Information Retrieval”, Cambridge University Press, First South Asian Edition, 2008.
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.



CP7083

INTERNET OF THINGS IN THE CLOUD

L	T	P	C
3	0	0	3

OBJECTIVES:

- To understand the basics of Internet of things and protocols
- To get an idea of some of the application areas where Internet of Things can be applied
- To understand the middleware for Internet of Things
- To understand the concepts of Web of Things
- To understand the concepts of Cloud of Things with emphasis on Mobile cloud computing

UNIT I INTRODUCTION 10

Definitions and Functional Requirements –Motivation – Architecture - Web 3.0 View of IoT– Ubiquitous IoT Applications – Four Pillars of IoT – DNA of IoT - The Toolkit Approach for End-user Participation in the Internet of Things. Middleware for IoT: Overview – Communication middleware for IoT –IoT Information Security

UNIT II IOT PROTOCOLS 8

Protocol Standardization for IoT – Efforts – M2M and WSN Protocols – SCADA and RFID Protocols – Issues with IoT Standardization – Unified Data Standards – Protocols – IEEE 802.15.4–BACNet Protocol– Modbus – KNX – Zigbee– Network layer – APS layer – Security

UNIT III WEB OF THINGS 10

Web of Things versus Internet of Things – Two Pillars of the Web – Architecture Standardization for WoT– Platform Middleware for WoT – Unified Multitier WoT Architecture – WoT Portals and Business Intelligence. Cloud of Things: Grid/SOA and Cloud Computing – Cloud Middleware – Cloud Standards – Cloud Providers and Systems – Mobile Cloud Computing – The Cloud of Things Architecture.

UNIT IV INTEGRATED 9

Integrated Billing Solutions in the Internet of Things Business Models for the Internet of Things - Network Dynamics: Population Models – Information Cascades - Network Effects – Network Dynamics: Structural Models - Cascading Behavior in Networks - The Small-World Phenomenon

UNIT V APPLICATIONS 8

The Role of the Internet of Things for Increased Autonomy and Agility in Collaborative Production Environments - Resource Management in the Internet of Things: Clustering, Synchronisation and Software Agents. Applications - Smart Grid – Electrical Vehicle Charging

TOTAL : 45 PERIODS

OUTCOMES:

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

- Identify and design the new models for market strategic interaction
- Design business intelligence and information security for WoB
- Analyze various protocols for IoT
- Design a middleware for IoT
- Analyze and design different models for network dynamics

REFERENCES:

SW7013

TEST DRIVEN DEVELOPMENT

L T P C
3 0 0 3

OBJECTIVES

The student should be able to

- Understand TDD and MDD
- Learn fundamentals of refactoring
- Perform unit testing in TDD
- Apply TDD
- Appreciate the emphasis of TDD in extreme programming.

UNIT I BACKGROUND 9

Software Development – Evolution – Agile environments - Test Driven Development – definition – importance - TDD and traditional testing – TDD vs. MDD, TDD vs. Acceptance TDD – Fakes , Mocks and Integration tests - TDD and documentation - Test-driven database development - Scaling TDD via Agile Model-Driven Development - Shortcomings – TDD Tools

UNIT II REFACTORING 9

Principles in Refactoring - Building Tests - Toward a Catalog of Refactorings - Composing Methods - Moving Features Between Objects - Organizing Data - Simplifying Conditional Expressions - Dealing with Generalization - Big Refactorings - Refactoring, Reuse, and Reality - Refactoring Tools.

UNIT III UNIT TESTING 9

Unit Testing – Motivators – Types of Tests – Improving Assertion – Improving Test Cases – Improving Test Suites – JUnit – Fundamentals – Developing an application with JUnit.

UNIT IV PATTERNS FOR TEST DRIVEN DEVELOPMENT 9

Test Driven Development Patterns – Red Bar Patterns – Testing Patterns – Green Bar Patterns – xUnit Patterns – Design Patterns – Mastering TDD.

UNIT V EXTREME PROGRAMMING 9

Extreme Principles – Creating a Vision of the System – Writing Acceptance Testing – Planning – Providing Estimates – Planning Releases – Planning Iterations – Tactical Planning – Pair Programming – Test First – Design – Code with intension – Relentless Integration – Delivering the System.

TOTAL : 45 PERIODS

OUTCOMES:

At the end the student will be able to

- Differentiate Test Driven Development and Model Driven Development
- Apply Refactoring for improving code
- Use Test Driven Development patterns in project development.

REFERENCES:

1. Dave Astels, "Test driven development: A practical guide", Prentice Hall Professional Technical Reference, 2003
2. Fowler, Martin, "Refactoring: improving the design of existing code", Pearson Education India, 2002
3. Kent Beck, "Test-driven development: by example", Addison-Wesley Professional, 2003
4. David Astels, Granville Miller, Miroslav Novak, "The Practical Guide to Extreme Programming", Prentice Hall PTR, 2002
5. Lasse Koskela. Test Driven. Manning, 2007
6. Lech Madeyski, "Test-driven development: An empirical evaluation of agile practice", Springer Science & Business Media, 2009
7. Roy Osherove. "The art of unit testing: with examples in." NET 2009

SW7008

PERSONAL SOFTWARE PROCESS

L	T	P	C
3	0	0	3

OBJECTIVES

The student should be able to

- Understand the nature of PSP
- Apply PSP principles in measuring software
- Appreciate the role of PSP in assessing software quality
- Relate PSP and TSP in software development.
- Learn to use PSP in Software engineering.

UNIT I INTRODUCTION

9

Personal Process Strategy – PSP Purpose – Logic for Software Engineering Discipline – Operational Processes – Defining and Using a Personal Process – Learning to Use a Personal Process – Baseline Personal Process – Contents – PSP Process Elements – PSP Structure and Levels – Incremental Development – PSP Tool Support.

UNIT II PSP SIZE ESTIMATION

9

Measuring Software Size – Size Measures – Establishing a Database Counting Standard – Establishing a LOC Counting Standard – Size Accounting – Using Size Data – Calculating Productivity – Size Counters – Other Size Measures – Software Estimating – Principles – Conceptual Design – Proxy Based Estimating – Producing Relative Size Table – Estimating Considerations – Probe Estimating Method.

UNIT III PSP QUALITY MANAGEMENT

9

PSP Quality Strategy – Software Quality – Economics of Software Quality – Defect Types – Personal Quality Practices – Quality Measures – Quality Management – Managing Product Quality – PSP Improvement Practices – Defect Prevention.

UNIT IV PSP DESIGN TEMPLATE

9

Design Process – Design Levels – Design Strategies – Design Quality – Design Representation – Design Templates (Operational, Functional, State and Logic) – State machine design example – Using PSP Design in Large Scale Design – Design Verification.

UNIT V TEAM SOFTWARE PROCESS

9

Development – Structure of TSP – Launching TSP Team – TSP Team working Process – TSP Quality Management – TSP Status and Future Trends.

TOTAL : 45 PERIODS

OUTCOMES

At the end the student will be able to

- Analyse software using PSP
- Use PSP tools to measure software quality
- Use PSP in software design

REFERENCES:

1. Watts S Humphrey, "PSP (SM): a self-improvement process for software engineers", Addison-Wesley Professional, 2005
2. Marsha Pomeroy-Huff, Robert Cannon, Timothy A. Chick, Julia Mullaney, and William Nichols, "The Personal Software Process SM (PSP SM) Body of Knowledge, Version 2.0", 2009
3. Watts S Humphrey, "Team Software Process (TSP)", John Wiley & Sons, Inc., 2000.

