M.E. (Software Engineering)

2013 Regulations, Curriculum & Syllabi

BANNARI AMMAN INSTITUTE OF TECHNOLOGY
(An Autonomous Institution Affiliated to Anna University, Chennai
Approved by AICTE - Accredited by NBA New Delhi, NAAC with 'A' Grade and ISO 9001:2008 Certified)
SATHYAMANGALAM – 638 401 Erode District Tamil Nadu
Phone : 04295 226000 Fax : 04295 226666
Web:www.bitsathy.ac.in E-mail : bitsathy@bannari.com

Contents

Rules and Regulations

M. E. / M. Tech. Programmes

(For the batch of students admitted in 2013-2014 and onwards)

NOTE: *The regulations hereunder are subject to amendments as may be decided by the Academic Council of the Institute from time to time. Any or all such amendments will be effective from such date and to such batches of students including those already in the middle of the programme) as may be decided by the Academic Council.*

1. Conditions for Admission

(i) Candidates for admission to the I Semester of M. E. / M. Tech. degree programme will be required to satisfy the conditions of admission thereto prescribed by the Anna University, Chennai and Government of Tamil Nadu.

(ii) Part–time candidates should satisfy conditions regarding experience, sponsorship, place of work, etc., that may be prescribed by Anna University, Chennai from time to time, in addition to satisfying requirements as in Clause 1(i).

2. Duration of the Programme

(i) **Minimum Duration:** The programme will lead to the Degree of Master of Engineering (M.E.) / Master of Technology (M. Tech.) of the Anna University, Chennai extend over a period of two years. The two academic years (Part-time three academic years) will be divided into four semesters (Part-time six Semesters) with two semesters per year.

(ii) **Maximum Duration**: The candidate shall complete all the passing requirements of the M. E. / M. Tech. degree programmes within a maximum period of 4 years / 8 semesters in case of full-time programme and 6 years / 12 semesters in case of part-time programme, these periods being reckoned from the commencement of the semester to which the candidate was first admitted.

3. Branches of Study

The following are the branches of study of M.E. / M.Tech. Programmes

M.E.

Branch I	Applied Electronics
Branch II	CAD/CAM
Branch III	Communication Systems
Branch IV	Computer Science and Engineering
Branch V	Embedded Systems
Branch VI	Engineering Design
Branch VII	Power Electronics and Drives
Branch VIII	Software Engineering
Branch IX	Structural Engineering
Branch X	VLSI Design

M. Tech.

Branch I	Biotechnology

4. Structure of Programmes

(i) **Curriculum:** The curriculum for each programme includes Courses of study and detailed syllabi. The Courses of study include theory Courses (including electives), seminar, practicals, Industrial training / Mini-project, Project Work (Phase I) and Project Work (Phase II) as prescribed by the respective Boards of Studies from time to time.

Full-time Programme: Every full-time candidate shall undergo the Courses of his/her programme given in clause 12 in various semesters as shown below:

Semester 1:	6 Theory Courses and two Practicals
Semester 2:	6 Theory Courses, one Practical and a Technical Seminar
Semester 3:	3 Theory Courses and Project Work (Phase I)
Semester 4:	Project work (Phase II).

Part-time Programme: Every part-time candidate shall undergo the Courses of his/her programme in various semesters as shown below:

Semester 1:	3 Theory Courses and one Practical
Semester 2:	3 Theory Courses and one Practical
Semester 3:	3 Theory Courses, Technical Seminar and one Practical
Semester 4:	3 Theory Courses
Semester 5:	3 Theory Courses and Project Work (Phase I)
Semester 6:	Project Work (Phase II)

(ii) **Theory Courses:** Every candidate shall undergo core theory, elective, and practical Courses including project work of his/her degree programme as given in clause 12 and six elective theory Courses. The candidate shall opt electives from the list of electives relating to his/her degree programme as given in clause 12 in consultation with the Head of the Department. However, a candidate may be permitted to take a maximum of two electives from the list of Courses of other M.E. / M.Tech. Degree programmes with specific permission from the respective Heads of the Departments.

(iii) **Project Work:** Every candidate individually shall undertake the Project Work (Phase I) during the third semester (fifth semester for part-time programme) and the Project Work (Phase II) during the fourth semester (Sixth semester for part-time programme). The Project Work (Phase II) shall be a continuation work of the Project Work (Phase I). The Project Work can be undertaken in an industrial / research organisation or in the Institute in consultation with the faculty guide and the Head of the Department. In case of Project Work at industrial / research organization, the same shall be jointly supervised by a faculty guide and an expert from the organization.

(iv) **Industrial Training / Mini Project:** Every full-time candidate shall opt to take-up either industrial training or Mini Project under the supervision of a faculty guide.

(v) **Value added / Certificate Courses:** Students can opt for any one of the Value added Courses in II and III semester. A separate certificate will be issued on successful completion of the Course.

(vi) **Special Self-Study Elective Courses:** Students can opt for any one of the special elective Courses as Self-Study in addition to the electives specified in the curriculum in II and III semesters, under the guidance of the faculty. The grades of only passed candidates will be indicated in the mark sheet, but will not be taken into account for assessing CGPA.

(vii) **Application oriented and Design Experiments**: The students are to carryout Application oriented and Design Experiments in each laboratory in consultation with the respective faculty and Head of the department.

(viii) **Mini project**: A Mini Project shall be undertaken individually or in a group of not more than 3 in consultation with the respective faculty and the Heads of the Department, in any one of the laboratories from I to III semesters.

M.E. SOFTWARE ENGINEERING

PROGRAMME EDUCATIONAL OBJECTIVES (PEOs)

I. Graduates will analyze, design and implement software systems to meet the client requirements in industry

II. Graduates will create, disseminate and transfer skills and knowledge through publications, and research activities

III. Graduates will communicate technical concepts in a concise and correct format that is suitable for the audience through their academic career

PROGRAMME OUTCOMES (POs)

a. Apply the principles of software engineering and mathematics for the design and implementation of software systems
b. Ability to analyze existing software systems and computing algorithms available in the literature
c. Ability to design processes and quality system components to meet the specific needs.
d. Ability to interpret and synthesize information to provide valid conclusions of the existing literature
e. Ability to select and apply appropriate modern tools for the design and development of software solutions
f. Ability to disseminate knowledge in professional forums for professional society needs
g. Ability to understand the impact of software systems in societal contexts
h. Apply ethical principles and commit to the norms of professional practice
i. Ability to function effectively as an individual
j. Ability to make effective reports, documentation and effective presentation on software project development activities
k. Ability to apply project management principles in practice for managing projects
l. Ability to engage in independent and life-long learning

MAPPING OF PEOs & POs

	Programme Educational Objectives	Programme Outcomes
PEO: I	Graduates will analyze, design and implement software systems to meet the client requirements in industry	(a), (c), (d), (f), (g)
PEO: II	Graduates will create, disseminate and transfer skills and knowledge through publications, and research activities	(b), (e), (k)
PEO: III	Graduates will communicate technical concepts in a concise and correct format that is suitable for the audience through their academic career	(h), (i), (j), (l)

M.E. Software Engineering

First Semester							
Code No.	**Course**	**Objectives & Outcomes**		**L**	**T**	**P**	**C**
		PEOs	**POs**				
13SE11	Applied Probability and Operations Research	I	(a)	3	1	0	4
13SE12	Software Engineering Methodologies	I,II	(a),(c),(j)	3	1	0	4
13SE13	Data Structures and Algorithms	I	(b),(e)	3	0	0	3
13SE14	Database Systems	I	(b),(e)	3	0	0	3
13SE15	Software Project Management	I,II,III	(d),(e),(h),(k)	3	1	0	4
13SE16	Network Engineering	I	(e)	3	1	0	4
13SE17	Data Structures and Algorithms Laboratory	I	(b),(e)	0	0	3	2
13SE18	Database Systems Laboratory	I	(b),(e)	0	0	3	2
			Total	**18**	**4**	**6**	**26**
Second Semester							
Code No.	**Course**	**Objectives & Outcomes**		**L**	**T**	**P**	**C**
		PEOs	**POs**				
13SE21	Software Component Architecture	I,II	(a),(c),(j)	3	0	0	4
13SE22	Software Testing and Quality Assurance	I,II	(a),(b),(d)	3	0	0	3
13SE23	Operating Systems	I	(e),(g)	3	1	0	4
	Elective			3	0	0	3
	Elective			3	0	0	3
	Elective			3	0	0	3
13SE24	Software Component and Testing Laboratory	I,II	(a),(c),(j)	0	0	3	2
13SE25	Technical Seminar	III	(l)	0	0	2	1
			Total	**18**	**1**	**5**	**23**
Third Semester							
Code No.	**Course**	**Objectives & Outcomes**		**L**	**T**	**P**	**C**
		PEOs	**POs**				
	Elective			3	0	0	3
	Elective			3	0	0	3
	Elective			3	0	0	3
13SE31	Project Work (Phase – I) & Viva-Voce	I,II,III	(a),(b),(d),(e),(j),(k),(i)				6
			Total	9	-	-	**15**
Fourth Semester							
Code No.	**Course**	**Objectives & Outcomes**		**L**	**T**	**P**	**C**
		PEOs	**POs**				
13SE41	Project Work (Phase – II) & Viva-Voce	I,II,III	(a),(b),(d),(e),(j),(k),(i)	-			**12**

List of Electives							
Code No.	**Course**	**Objectives & Outcomes**		**L**	**T**	**P**	**C**
		PEOs	**POs**				
13SE51	Cloud Computing	I	(d),(e)	3	0	0	3
13SE52	Data Warehousing and Mining	I	(a),(g)	3	0	0	3
13SE53	XML and Web Services	I	(d),(e)	3	0	0	3
13SE54	Soft Computing	I	(a),(g)	3	0	0	3
13SE55	Digital Image Processing	I	(a),(e)	3	0	0	3
13SE56	Software Reliability & Reuse	I	(a),(g)	3	0	0	3
13SE57	Artificial Intelligence	I	(a),(e)	3	0	0	3
13SE58	Real Time Systems	I	(d),(e)	3	0	0	3
13SE59	Network Routing	I	(d),(e)	3	0	0	3
13SE60	Enterprise Resource Planning	I	(a),(e)	3	0	0	3
13SE61	Multimedia Systems	I	(d),(e)	3	0	0	3
13SE62	Mobile and Pervasive Computing	I	(a),(e)	3	0	0	3
13SE63	Software Architecture	I,II	(a),(e),(g)	3	0	0	3
13SE64	Research Methodologies	III	(l)	3	0	0	3
13SE65	Software Development Skill*	I	(a),(e),(g)	3	0	0	3
13SE66	Open Source Software*	I,II	(a),(d),(e)	3	0	0	3
13SE67	App Engine*	I	(d),(e)	3	0	0	3
Self Study Elective							
13SE01	BigData Analytics	II,III	(h),(i),(l)	3	0	0	3

* Open elective

13SE11 APPLIED PROBABILITY AND OPERATIONS RESEARCH

3 1 0 4

Course Objectives (COs):

- To understand the basics of standard distributions and their significances
- To understand the applications of stochastic processes and queue models with simulation techniques
- To understand and master the optimization theory through different problem solving methods

.

Course Learning Outcomes (CLOs):

- Understand the concepts of basics of standard distributions
- Provide solution to the problems related to stochastic processes and queue models with simulation techniques
- Relate the concepts to engineering applications

Program Outcome (PO):

(a) The ability to apply software engineering theory, principles, tools and processes, as well as the theory and principles of computer science and mathematics, to the development and maintenance of complex software systems

Unit I

Probability and Random Variables

Probability concepts – Random Variables – Moment generating function – Standard distributions - Binomial - Poisson - Rectangular or Uniform – Normal - Exponential distributions - Functions of random variables - Two dimensional random variables.

12 Hours

Unit II

Stochastic Processes

Classification – Stationary and Random process – Markov process – Markov chains – Transition probability – Classification of Markov chain – Limiting distribution – First passage time – Poisson process – Birth and death process.

12 Hours

Unit III

Queue Models

Characteristics of Queueing Models- Kendal's Notation-Single and Multi-Server Markovian queuing models M/M/1, M/M/C (finite and infinite capacity) and (M/G/1) :(∞/GD) - Queuing applications.

12 Hours

Unit IV

Simulation and Applications

Introduction – Types of simulation – Limitations of simulation techniques – Phases of simulation study – Generation of random numbers – Monte Carlo simulation – Applications to queuing problems.

12 Hours

Unit V

Classical Optimization Theory

Unconstrained external problem – Newton Raphson method – Equality constraints – Lagrangian method – Kuhn Tucker conditions.

12 Hours

Total: 60 Hours

References

1. Richard Johnson, *Miller and Freund's Probability and Statistics for Engineers*, Prentice Hall, New Delhi, 7th Edition, 2007.
2. S. D. Sharma, *Operations Research*, Kedar Nath, Ramnath & Co, Meerut, 2004.
3. S. C. Gupta and V. K. Kapoor, *Fundamentals of Mathematical Statistics*, Sultan Chand & Sons, 2001.
4. H. A. Taha, *Operations Research – An Introduction*, 8th Edition, Prentice Hall of India Ltd, New Delhi, 2008.

13SE12 SOFTWARE ENGINEERING METHODOLOGIES

3 1 0 4

Course Objective (CO):

- Ability to design, synthesize, and analyze, software systems of increasing size and complexity at various abstraction levels, from the individual component to the entire system architecture

Course Learning Outcomes (CLOs):

- Ability to define, assess, and tailor software quality practices, and software processes and methodologies for appropriate application on software development projects in a variety of domain areas
- Ability to be an effective member of a multi-disciplinary software-intensive product development with an awareness of individual professional and ethical responsibilities

Programme Outcomes (POs):

a. Apply the principles of software engineering and mathematics for the design and implementation of software systems
c. Ability to design processes and quality system components to meet the specific needs.
j. Ability to make effective reports, documentation and effective presentation on software project development activities

Unit I
The Process
Software Engineering the nature of Software -Software Process Models: Waterfall Model-Incremental process models- Evolutionary process models: Prototyping-Spiral model – Concurrent model- Comparison study of Software Process Models -Introduction to Agile process – Computer Based Systems – System Engineering Hierarchy. System Modeling

9 Hours

Unit II
Requirements Analysis
Requirements Engineering- tasks – Initialization the Requirement Engineering process - Establishing the groundwork-Eliciting requirements-Building the requirements model-Validating Requirements - Requirements analysis-Model Approaches – Data Modeling Concepts- Class Based Modeling - Behavioral Model. Analysis Modeling

9 Hours

Unit III
Design Concepts and Principles
The Design concepts-The Design model-Architectural design-Designing Class Based Components -User interface design: user analysis and design, Interface analysis, Interface design steps-Requirements modeling: Flow oriented modeling- Software risk management, Case Studies. Real Time Systems

9 Hours

Unit IV
Managing the Software Projects
The Management spectrum – W5HH principle – Metrics for Process and Projects – Software Scope and Feasibility - Software Measurement – Software Project Estimation – Decomposition Techniques – Project Scheduling –Risk Management – Identification – Projection – RMMM Plan. Management Spectrum

9 Hours

Unit V

Software Quality Assurance

A Strategic Approach to Software Testing- Unit Testing- Integration Testing- Validation Testing - System Testing- The Art of Debugging - Elements of Software Quality Assurance- SQA Tasks, Goals and Metrics- Software Review Techniques: Informal reviews-Formal Technical Reviews-Software Reliability- Software Configuration Management-The SCM Process- Version Control- Change Control- Configuration Audit. Software Maintenance

9 Hours

Case Studies **15 Hours**

Total: 60 Hours

References

1. Roger S. Pressman, Software Engineering: A Practitioner Approach Seventh Edition , McGraw-Hill, 2010
2. Fairley, Software Engineering Concepts, McGraw-Hill, 2009
3. I. Summerville, Software Engineering, Addison Wesley, Eighth Edition,2006
4. David Gustafson, Software Engineering, Schaums outlines, Tata McGraw-Hill, 2003

13SE13 DATA STRUCTURES AND ALGORITHMS

3 0 0 3

Course Objectives (COs):

- To introduce the linear and nonlinear data structures to the students
- To understand the various algorithm design techniques
- To analyze the various algorithms and their complexity including NP completeness

Course Learning Outcomes (CLOs):

- Students will be able to design and apply appropriate data structures for the given problem
- Students will improve their problem solving skills and algorithm design skills
- Students will have proficiency in algorithm and complexity analysis

Programme Outcomes (POs):

b. Ability to analyze existing software systems and computing algorithms available in the literature
c. Ability to select and apply appropriate modern tools for the design and development of software solutions

Unit I

Algorithm Analysis, Linear and Non-Linear Data Structures

Mathematical background, Running time calculation, Linear data structures, Array and linked implementation, Applications. Trees: AVL trees, Red black trees, B-trees, Tries, Introduction to Binomial and Fibonacci heaps

9 Hours

Unit II

Algorithm Design Techniques

Design Techniques: Greedy technique, Divide and conquer, Dynamic Programming, Backtracking, Branch and bound. Problems: Knapsack problem, Quick Sort, Optimal binary search trees, Long Integer Multiplication, Longest Common Subsequence, n-queens problem, subset sum problem, Travelling salesman problem

9 Hours

Unit III:

Graph Algorithms

Minimum Spanning Trees - Single-Source Shortest Paths - Bellman-Ford algorithm - All-Pairs Shortest Paths - Shortest paths and matrix multiplication - Floyd-Warshall algorithm - Johnson's algorithm for sparse graphs, Biconnectivity, Strong Components

9 Hours

Unit IV:

Number Theoritic and String Matching Algorithms

Elementary number-theoretic notions - Greatest common divisor - Modular arithmetic - Solving modular linear equations - The Chinese remainder theorem - Powers of an element - The RSA public-key cryptosystem - Primality testing - Integer factorization. String Matching: Naive string-matching algorithm - BM algorithm - String matching with finite automata - KMP algorithm

9 Hours

Unit V:

Computational geometry and NP problem

Computational Geometry: Line-segment properties - Determining whether any pair of segments intersects - Finding the convex hull - Finding the closest pair of points. NP-Completeness: Polynomial time - Polynomial-time verification - NP-completeness and reducibility

9 Hours

Total: 45 Hours

References

1 Mark allen weiss, “Data Structures and Algorithm analysis in C”, Second Edition, Pearson education, 2008.
2 T H Cormen, Charles Leiserson, Ronald R Rivest, Clifford Stein, “Introduction to Algorithms, Third Edition, PHI, 2010.
3 Anany Levitin, “Design and Analysis of Algorithms”, Pearson Education, 2010.

13SE14 DATABASE SYSTEMS

3 0 0 3

Course Objectives (COs):

- To understand the Relational Database Design concepts
- To study the Database Systems Architectures
- To introduce the optimization techniques in Query processing
- To build and maintain applications and administer Database Systems

Course Learning Outcomes (CLOs):

- Student will be able to understand the concepts of Relational Database Concepts
- Student will be able to build and optimize the Database Queries
- Student will be able to develop and maintain databases for real time application

Programme Outcomes (POs):

d. Ability to analyze existing software systems and computing algorithms available in the literature
e. Ability to select and apply appropriate modern tools for the design and development of software solutions

Unit I

Relational Database

Features of Good Relational Design – Atomic Domains and First Normal Form – Decomposition Using Functional Dependencies – Functional Dependency Theory - Algorithms for Decomposition- Decomposition Using Multi valued Dependencies - More Normal Forms - Database-Design Process - Modeling Temporal Data.

9 Hours

Unit II

Query Processing and Optimization

Measures of Query Cost-Selection Operation – Sorting - Join Operation-Other Operations - Evaluation of Expressions; Query Optimization – Transformation of Relational Expressions - Estimating Statistics of Expression - Choice of Evaluation Plans- Materialized Views-Advanced Topics in Query Optimization.

9 Hours

Unit III

Database-System Architectures

Centralized and Client–Server Architectures - Server System Architectures - Parallel Systems - Distributed Systems -Network Types. Parallel Databases: Introduction - I/O Parallelism – Inter query Parallelism - Intra query Parallelism – Intra operation Parallelism – Inter operation Parallelism.

9 Hours

Unit IV

Query Optimization and Distributed Database

Query Optimization- Design of Parallel Systems - Parallelism on Multicore Processors. Distributed Databases: Homogeneous and Heterogeneous Databases - Distributed Data Storage- Distributed Transactions – Commit Protocols - Concurrency Control in Distributed Databases- Availability - Distributed Query Processing - Heterogeneous Distributed Databases Cloud-Based Databases- Directory Systems.

9 Hours

Unit V

Mobility And Transaction Processing

Spatial, Temporal Data and Mobility: Time in Databases- Spatial and Geographic Data - Multimedia Databases- Mobility and Personal Databases - Transaction Processing: Transaction-Processing Monitors – Transactional Workflows - E-Commerce - Main-Memory Databases - Real-Time Transaction Systems - Long-Duration Transactions.

9 Hours

Total Hours: 45 Hours

References

1 Silberschatz,Henry F. Korth,S. Sudarshan, “Database System Concepts”, McGraw-Hill ,Sixth Edition, 2010.

2 Ramez Elmasri,Shamkant B. Navathe, "Fundamentals of Database Systems", Pearson, Sixth Edition, 2010.

13SE15 SOFTWARE PROJECT MANAGEMENT

3 1 0 4

Course Objectives (COs):

- At the end of the course, the student should get familiarize with the characteristics of a project, project management principles, risk in environment and the management challenges for effective project management
- The student will be able to understand and use the project management principles across all phases of a project.
- They will be able to demonstrate competency in the management of a project plan, especially in monitor and controlling a project schedule and budget, tracking project progress

Course Learning Outcomes (CLOs):

- They will understand how to work as team member and as individual without affecting the quality of project
- The student should be able to coordinate the close out of a project contract
- They will share best practices and tools related to project management

Programme Outcomes (POs):

d. Ability to interpret and synthesize information to provide valid conclusions of the existing literature
e. Ability to select and apply appropriate modern tools for the design and development of software solutions
h. Apply ethical principles and commit to the norms of professional practice
k. Ability to function effectively as an individual

Unit I

Introduction to Software Project Management

Introduction, the state of IT project management, context of project management, need of project management, project goals, project life cycle and IT development, extreme project management, PMBOK. Project Definition – Contract Management – Activities Covered By Software Project Management – Overview of Project Planning – Stepwise Project Planning.

9 Hours

Unit II

Developing the Project Charter and Project Plan

Introduction, project management process, project integration management, the project charter, project planning framework, the contents of a project plan, the planning process, The Work Breakdown Structure (WBS), the linear responsibility chart, multidisciplinary teams.

9 Hours

Unit III

The Project's Schedule, Budget and Risk Management

Introduction, developing the project schedule, project management software tools, methods of budgeting, developing the project budget, improving cost estimates, finalizing the project schedule and budget. IT project risk management planning process, identifying IT project risks, risk analysis and assessment, risk strategies, risk monitoring, and control, risk responses and evaluation.

9 Hours

Unit IV

The Project Communication Plan

Introduction, monitoring and controlling the project, the project communications plan, project metric, project control, designing the control system, the plan-monitor-control cycle, data collection and reporting, reporting performance and progress, information distribution.

9 Hours

Unit V

Managing People and Organizing Teams

Introduction – Understanding Behavior – Organizational Behaviour: A Background – Selecting The Right Person For The Job – Instruction In The Best Methods – Motivation – The Oldman – Hackman Job Characteristics Model – Working In Groups – Becoming A Team –Decision Making – Leadership – Organizational Structures – Stress – Health and Safety – Case Studies.

9 Hours

Case Studies:

For every application Create the following:

1. Work Packages

Specify the work packages for the activities and tasks that must be completed in order to satisfy the project agreement. Each work package is uniquely identified. A diagram depicting the breakdown of project activities and tasks (a work breakdown structure) may be used to depict hierarchical relationships among work packages.

2. Dependencies

Specify the ordering relations among work packages to account for interdependencies among them and dependencies on external events. Techniques such as dependency lists, activity networks, and the critical path method may be used to depict dependencies among work packages.

3. Resource Requirements

Provide, as a function of time, estimates of the total resources required to complete the project. Numbers and types of personnel, computer time, support software, computer hardware, office and laboratory facilities, travel, and maintenance requirements for the project resources are typical resources that should be specified.

4. Budget and Resource Allocation

Specify the allocation of budget and resources to the various project functions, activities, and tasks.

5. Schedule

Provide the schedule for the various project functions, activities, and tasks, taking into account the precedence relations and the required milestone dates. Schedules may be expressed in absolute calendar time or in increments relative to a key project milestone.

15 Hours

Total: 60 Hours

References

1. Bob Hughes, Mikecotterell, "Software Project Management", Third Edition, Tata McGraw Hill, 2004
2. Jalote, "Software Project Manangement in Practive", Pearson Education, 2002
3. Jack T. Marchewka,"Information Technology Project Management", 3rd edition, Wiley India, 2009
4. E-Book – Project Management Body of Knowledge
5. S. J. Mantel, J. R. Meredith and etl., "Project Management"1st edition, Wiley India, 2009

13SE16 NETWORK ENGINEERING

3 1 0 4

Course Objectives (COs):

- Able to Understand the concepts of Networking and its management
- To study the principles and mechanisms of wireless and multimedia networks
- To study the concepts and working principles of LAN topology

Course Learning Outcomes (CLOs):

- Able to formulate and solve problems creatively in network design, routing, management, security and performance
- Design and develop the LAN topology
- Apply the skills to use different wireless networks in real time communication systems

Program Outcomes (POs):

e. Ability to select and apply appropriate modern tools for the design and development of software solutions

Unit I
Networking: Functional Elements
Networking as resource sharing- The Functional Elements: Multiplexing, Switching, Routing, Network Management, Traffic controls and time scales- Network Infrastructure, Networking Architecture, X.25 and frame relay networks, The Internet, ATM networks.

9 Hours

Unit II
Wireless and Mobile Networks
Wireless links and Network characteristics- Wi-Fi:802.11 Wireless LANs: The 802.11 architecture, MAC Protocol, Frame Structure, Mobility in IP Subnet, 802.15 and Bluetooth- Cellular Internet Access- Mobility management principles-Mobile IP-Managing Mobility in cellular Networks- Wireless and Mobility: Impact on higher layer Protocols.

9 Hours

Unit III
Multimedia Networking
Multimedia Networking Applications-Streaming Stored Audio and Video- Protocols for real time Interactive Applications, RTP,RTCP, SIP- Distributing Multimedia- Scheduling and Policing Mechanisms- RSVP

9 Hours

Unit IV
Network Management
The infrastructure for network Management- The Internet- Standard Management Framework, Structure of Management Information: SMI, Management Information Base, SNMP protocol operations and Transport mappings, Security and Administrations- ASN.1

9 Hours

Unit V
Designing the LAN-I and LAN II
Campus Network Design Goals- Designing a LAN Topology-Campus Hierarchical design- Spanning Tree Protocol – IP Multicasting

9 Hours

Lab Component:

1. Wireless LAN protocols

To create scenario and study the performance of network with CSMA / CA protocol and compare with CSMA/CD protocols.

2. Routing algorithms
   Implementation of distance vector routing algorithm.
   Implementation of Link state routing algorithm.
3.Transfer of files from PC to PC using Windows / UNIX socket processing.
4. Simulation and performance analysis of LAN with ns2.

15 Hours
Total: 60 Hours

References

1. James F. Kurose, Keith W. Ross, Computer Networking- A Top-Down Approach Featuring the Internet, Third Edition, Pearson Publishers, 2009.
2. Anurag Kumar, D. Manjunath, Joy Kuri, Communication Networking- An Analytical Approach, Morgan Kaufmann Publishers, An Imprint of Elsevier, 2006.
3. Corman Long, IP Network Design, Tata Mc Graw-hill Publishing Company Limited, New Delhi, 2004.

13SE17 DATA STRUCTURES AND ALGORITHMS LABORATORY

0 0 3 2

Course Objectives (COs):

- To introduce the linear and nonlinear data structures to the students
- To understand the various algorithm design techniques
- To analyze the various algorithms and their complexity including NP completeness

Course Learning Outcomes (CLOs):

- Students will be able to design and apply appropriate data structures for the given problem
- Students will improve their problem solving skills and algorithm design skills
- Students will have proficiency in algorithm and complexity analysis

Programme Outcomes (POs):

b. Ability to analyze existing software systems and computing algorithms available in the literature
e. Ability to select and apply appropriate modern tools for the design and development of software solutions

List of Experiments

1 Linear data structures
2 B-Tree / Red Black Tree
3 Tries
4 Divide and Conquer and Dynamic Programming Algorithms
5 Backtracking and Branch and Bound Algorithms
6 Shortest path algorithms
7 Strong Components / Bi-connectivity Implementation
8 Number Theoretic Algorithms
9 String Matching Algorithms
10 Computational Geometry

13SE18 DATABASE SYSTEMS LABORATORY

0 0 3 2

Course Objectives (COs):

- To apply the Relational Database Design concepts
- To analyze and optimize the SQL Query
- To design a database for real time Application

Course Learning Outcomes (CLOs):

- Student will be able to build and optimize the Database Queries
- Student will be able to design and implement databases for real time Applications
- Student will be able to create Web based Applications

Programme Outcomes (POs):

b. Ability to analyze existing software systems and computing algorithms available in the literature
e. Ability to select and apply appropriate modern tools for the design and development of software solutions

List of Experiments

1 SQL Queries and No-SQL
2 Design of Databases (Normalization)
3 Database Query Optimization (TOAD for MYSQL)
4 Database Application Development - I*
5 Database Application Development - II*
6 Application development with Cloud Databases
7 Mini Project

* HTML, Client side scripting and Server side scripting should be duly incorporated

13SE21 SOFTWARE COMPONENT ARCHITECTURE

3 1 0 4

Course Objectives (COs):

- To study about the software components and their interaction methodologies
- A brief knowledge about Component Based Software development
- To gain practical understanding on Component Technologies

Course Learning Outcomes (CLOs):

- Induces the basic knowledge on software components with their interoperability among various other domains
- Enhances the use of software components in the distributed environment with practical knowledge on various component technologies evolving in the market
- Enable the students to develop a component software that can be used in any real time applications with greater functionality
- Eases the design a software component that can be integrated in any of the business environment
- Greater understanding of cloud based component development and the traditional approach

Programme Outcomes (POs):

a. Apply the principles of software engineering and mathematics for the design and implementation of software systems
c. Ability to design processes and quality system components to meet the specific needs.
j. Ability to make effective reports, documentation and effective presentation on software project development activities

Unit I

Introduction

Software Components - Component models and Component Services-myths in Component Based Technology - Risk Factors - Success Factors, Component Based Software Development- Case Study: Testing tools for Component based Development.

9 Hours

Unit II

Components, Architecture and Process

Component Architecture, Component Frameworks, Component Development, Component distribution and acquisition, Component assembly, markets and components-Case Study: Earth System Modeling Framework (ESMF), Middleware Vs Cloud in distributed computing.

9 Hours

Unit III

Design of Software Components

Software Components and the UML Component Infrastructures - Business Components - Components and Connectors - Designing Models of Modularity & Integration-Case Study: Buildap.

9 Hours

Unit IV

Distributed Systems

Introduction to Distributed systems-examples of distributed systems, challenges-Architectural models- Fundamental models - Introduction to interprocess communications-External data representation and marshalling- Client server communication-Group communication – Case study: IPC in UNIX .

9 Hours

Unit V

Component Technologies

Overview of the Following Component Models: CORBA, COM+, Enterprise Java Beans, Software Agents. – Case study: CORBA RMI.

9 Hours

Total: 45 Hours

References

1. GeorgeT.Heinemen, William T. Councill, "Component Based Software Engineering", Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA, 2001 -digitized 2011.
2. George Coulouris, Jean Dollimore, Tim Kindberg "Distributed Systems Concepts and Design" Fifth Edition – 2012- Pearson Education Asia.
3. Clemens Szyperski, "Component Software: Beyond Object-Oriented Programming", Pearson Education publishers, 2009 Reprint.
4. Ed Roman, "Mastering Enterprise Java Beans", John Wiley & Sons Inc., 2005.
5. Thomas J..Mowbray, William A.Ruh, "Inside CORBA Distributed Object Standards and Applications", Addison – Wesley, 2001.
6. Dale Rojerson, "Inside COM", Microsoft Press, 2001.

13SE22 SOFTWARE TESTING AND QUALITY ASSURANCE

3 0 0 3

Course Objectives (COs):

This course equips the students with a solid understanding of:

- Practices that support the production of quality software
- Software testing techniques
- Life-cycle models for requirements, defects, test cases, and test results
- Process models for units, integration, system, and acceptance testing
- Quality Models

Course Learning Outcomes (CLOs):

- Know the input and deliverables of the testing process.
- Understand appropriate technological tools to analyze, test and automate the elements of software quality assurance.

Programme Outcomes (POs):

(a) The ability to apply software engineering theory, principles, tools and processes, as well as the theory and principles of computer science and mathematics, to the development and maintenance of complex software systems

(b) The ability to participate productively on software project teams involving students from both software engineering and other majors

UNIT I

FUNDAMENTALS OF SOFTWARE QUALITY ASSURANCE

Quality concepts – quality, quality control, quality assurance, cost of quality Software quality assurance – SQA activities-Software reviews, inspections, audits-Software reliability-Quality Attributes: correctness, reliability, usability, integrity, portability, maintainability, interoperability-Ishikawa's Seven Basic Tools.

8 Hours

UNIT II

QUALITY STANDARDS

Ethical Basis for Software Quality – Total Quality Management Principles – Software Processes and methodologies-Quality Standards, Practices and Conventions – Software Configuration Management –Enterprise Resource Planning Software- ISO 9000 & 9001, CMM, six sigma.

8 Hours

UNIT III

QUALITY METRIC SYSTEM

Measurement Theory – Software Quality Metrics – Designing Software Measurement Programs – complexity Metrics and Models (CMM) – Following KPAs : requirements management (RM), software project tracking and oversight (SPTO)– Organizational Learning – Improving Quality with Methodologies – Structured/Information Engineering.

8 Hours

UNIT IV

SOFTWARE TESTING - INTRODUCTION

Testing as an Engineering Activity - Role of Process in Software Quality – Testing as a Process – Basic Definitions, Software Testing Principles – The Tester's Role in a Software Development Organization – Origins of Defects – Defect Metrics-Defect Classes – The Defect Repository and Test Design – Defect Examples – Developer/Tester Support for Developing a Defect Repository.

10 Hours

UNIT V

TESTING ISSUES

Introduction to Testing Design Strategies – The Smarter Tester –Test Case Design Strategies – Using Black Box Approach to Test Case Design – Random Testing – Equivalence Class Partitioning – Boundary Value Analysis – Other Black-box Test Design Approaches – Black-box testing and COTS – Using White-Box Approach to Test design – Test Adequacy Criteria – Coverage and Control Flow Graphs – Covering Code Logic – Paths – White-box Based Test Design – Additional White Box Test Design Approaches – Evaluating Test Adequacy Criteria.

11 Hours

Case Studies **15 Hours**

Total: 60 Hours

References

1. Schulmeyer, G. Gordon, James McManus, "Handbook of Software Quality Assurance", Fourth Edition, Van Nostrand Reinhold, 2008.
2. Edward Kit, "Software Testing in the Real World – Improving the Process", Pearson Education, 2004.
3. William E.Perry , "Effective methods for Software Testing", Second Edition, Wiley, 2000.

13SE23 OPERATING SYSTEMS

4 1 0 4

Course Objectives (COs):

- To know the components of an operating system.
- To have a thorough knowledge of process and storage management
- To know the concepts of I/O and file systems.

Course Learning Outcomes (CLOs):

- Ability to design and implement efficient algorithms for the problem under consideration
- Ability to choose appropriate solutions for the given problem.
- A thorough knowledge on coding skills.

Programme Outcomes (POs):

e. Ability to select and apply appropriate modern tools for the design and development of software solutions
g. Ability to understand the impact of software systems in societal contexts

UNIT I
PROCESSES AND THREADS

Introduction to operating systems – review of computer organization – operating system structures – system calls – system programs – system structure – virtual machines. Processes: Process concept – Process scheduling – Operations on processes – Cooperating processes – Inter-process communication – Communication in client-server systems. Case study: IPC in Linux. Threads: Multi- threading models – Threading issues. Case Study: Pthreads library.

9 Hours

UNIT II
PROCESS SCHEDULING AND SYNCHRONIZATION

CPU Scheduling: Scheduling criteria – Scheduling algorithms – Multiple-processor scheduling – Real time scheduling – Algorithm Evaluation. Case study: Process scheduling in Linux. Process Synchronization: The critical-section problem – Synchronization hardware – Semaphores – Classic problems of synchronization – critical regions – Monitors. Deadlock: System model – Deadlock characterization – Methods for handling deadlocks – Deadlock prevention – Deadlock avoidance – Deadlock detection – Recovery from deadlock.

9 Hours

UNIT III
STORAGE MANAGEMENT

Memory Management: Background – Swapping – Contiguous memory allocation – Paging – Segmentation – Segmentation with paging. Virtual Memory: Background – Demand paging – Process creation – Page replacement – Allocation of frames – Thrashing. Case Study: Memory management in Linux

9 Hours

UNIT IV

FILE SYSTEMS

File-System Interface: File concept – Access methods – Directory structure –File-system mounting – Protection. File-System Implementation : Directory implementation – Allocation methods – Free- space management – efficiency and performance – recovery – log-structured file systems. Case studies: File system in Linux – file system in Windows XP

9 Hours

UNIT V

I/O SYSTEMS

I/O Systems – I/O Hardware – Application I/O interface – kernel I/O subsystem – streams – performance. Mass-Storage Structure: Disk scheduling – Disk management – Swap-space management – RAID – disk attachment – stable storage – tertiary storage. Case study: I/O in Linux

9 Hours

Lab Components:

1 Process creation and execution
2 Implementation of process scheduling algorithm.
3 Design of thread using Linux/UNIX thread
4 Simulation of Inter process.

15 Hours

Total : 60 Hours

References

1 Silberschatz, Galvin, and Gagne, “Operating System Concepts”, Ninth Edition, Wiley India Pvt Ltd, 2012.
2 Andrew S. Tanenbaum, “Modern Operating Systems”, Third Edition, Pearson Education/PHI, 2007.

13SE24 SOFTWARE COMPONENT & TESTING LABORATORY

0 0 3 2

Course Objectives (COs):

- To comprehend vital distinctiveness of tool used for test automation.
- Software tool rationale substantiates that software meet the precise requirements.

Course Learning Outcomes (CLOs):

- Know the input and deliverables of the testing process.
- Understand appropriate technological tools to analyze, test and automate the elements of software quality assurance.

Programme Outcomes (POs):

a. Apply the principles of software engineering and mathematics for the design and implementation of software systems
c. Ability to design processes and quality system components to meet the specific needs.
j. Ability to make effective reports, documentation and effective presentation on software project development activities

List of Experiments

1 Develop a White Box Testing for various work models
   a ATM System
   b Expert System
   c Stock Maintenance etc.,

2. Do the following testing for a chosen software implementation using any commercial or freeware tools:
   a Path-testing
   b Transaction-flow testing.
   c Data-flow testing.

3. Do the following measurements on a chosen software implementation:
   a JUnit Test
   b McCabe's Cyclomatic Complexity
   c Simulate a test driver

4. Develop a simple software testing tool for implementing in web based systems.
   a Simulate a tool that reflects
   b Black Box Testing
   c System Testing

5. Implement a Load testing process using JSP program.
6. Perform Mobile Testing

13SE51 CLOUD COMPUTING

3 0 0 3

Course Objectives (COs):

- To introduce the fundamentals of cloud computing
- To introduce the data storage and collaboration in the cloud
- To enable the students to design and code cloud based applications.

Course Learning Outcomes (CLOs):

- The students will be able to design and code cloud based applications.
- The students will be able to deploy and manage cloud solutions.

Programme Outcomes (POs):

d. Ability to interpret and synthesize information to provide valid conclusions of the existing literature
e. Ability to select and apply appropriate modern tools for the design and development of software solutions

Unit I
Introducing Cloud Computing and services
Introducing Cloud Computing, Software as a Service, Platform as a service, Infrastructure as a service, Identity as a service, Introduction to SalesForce

9 Hours

Unit II
Storage and Collaboration
Understanding Cloud based data storage, Cloud based backup systems, database solutions and Block storage, Understanding file systems. Collaboration in the cloud.

9 Hours

Unit III
Virtualization
Understanding virtualization, Leveraging blade servers, server virtualization, desktop virtualization, Desktop solutions on demand, Virtual networks, data storage virtualization, Securing the cloud.

9 Hours

Unit IV Cloud Application Development
Design and Coding
Designing cloud based applications, Coding cloud based Applications, Creating a mashup using yahoo pipes, Google App Engine, Creating windows Azure application. Case Study: Amazon Web Services

9 Hours

Unit V Scripting
Google Scripts
Understanding Google Script, First Steps in Google Script, Setting up Development environment, Building interface, Adding actions.

9 Hours
Total: 45 Hours

References

1 Kris Jamsa, "Cloud Computing", Jones & Bartlett Learning, 2013
2 James Ferreria,“Google Script, Enterprise Application Essentials”, O'reilly, 2012
3 Michael Miller, "Cloud Computing: Web-Based Applications That Change the Way You Work and Collaborate Online", Que Publishing, 2009

13SE52 DATA WAREHOUSING AND MINING

3 0 0 3

Course Objectives (COs):

- To introduce the need for Business Intelligence
- To Introduce the design of data warehouse and its implementation
- To understand the various classical data mining algorithms including prediction, classification and clustering
- To introduce the concepts including text mining and web mining.

Course Learning Outcomes (CLOs):

- Ability design and implement data warehouse for specific business domains
- Ability to analyze various data mining algorithms and ability to choose apply appropriate algorithms

Programme Outcomes (POs):

a. Apply the principles of software engineering and mathematics for the design and implementation of software systems
g. Ability to understand the impact of software systems in societal contexts

Unit I

Data Warehouse and OLAP Technology

Introduction to data warehouse, Multidimensional Data Model, Data warehouse architecture, data warehouse implementation, Introduction to data cube computation – Data-cube computation algorithms

9 Hours

Unit II

Classification and prediction

Classification and Prediction, Issues, Decision Tree Induction, Bayesian classification, Back propagation, Support vector machines, prediction, accuracy and error measures.

9 Hours

Unit III

Data Clustering

Types of data, Partitioning methods, Hierarchical methods, density based methods, model based methods, outlier analysis, *Cluster tendency Analysis - vat algorithm.*

9 Hours

Unit IV

Association Mining

Frequent itemset mining methods, Apriori Algorithm, FP-growth Algorithm, Mining Closed Itemsets - Algorithms, Graph mining, Social network analysis - *DCI-Close and High Dimensional Algorithms*

9 Hours

Unit V

Advanced Topics

Text Mining, Mining the world wide web, Current research trends, Introduction to Bigdata Analytics

9 Hours

Total: 45 Hours

References

1 Jiawei Han, Kamber, "Data Mining - Concepts and Techniques", Third Edition, Elsivier, 2012

2 Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's Businesses, Wiley, 2013

13SE53 XML AND WEB SERVICES

3 0 0 3

Course Objectives (COs):

- To study the essentials of Data interoperability using XML and well-formed XML documents.
- To understand the concepts of SOA and Web Service Protocols.
- To learn about the key building blocks of web services.

Course Learning Outcomes (CLOs):

- Students will be able to assess and evaluate the role of XML for the management and delivery of electronic information.
- Students will have an understanding of the principles and role of structured generic markup.
- Students will be able to Deploy and Publish Web Services
- Able to understand Web Services and its Infrastructure.

Programme Outcomes (POs):

d. Ability to interpret and synthesize information to provide valid conclusions of the existing literature
e. Ability to select and apply appropriate modern tools for the design and development of software solutions

Unit I

Introduction of XML

XML – benefits – Structuring with schemas - DTD – XML Schemas – XML processing – DOM – SAX – presentation technologies – XSL – XFORMS – XHTML – Transformation – XSLT – XLINK –XPOINTERS - XPATH – XQuery.

9 Hours

Unit II

Applied XML & Web Services

Standards - Stack Layers - Stack Aspects –presentation - security – query – semantics - Service-oriented Architecture (SOA) – Architecting web services – Implementation view – Web services technology stack – Logical view – Composition of web services – Deployment view.

9 Hours

Unit III

Web Services Protocols

Transport protocols for web services – Messaging with web services - Protocols - SOAP - Describing web services – WSDL – Anatomy of WSDL – Manipulating WSDL – Web service policy – Discovering web services – UDDI – Anatomy of UDDI – Web service inspection – Ad-Hoc Discovery - Securing web services.

9 Hours

Unit IV

Implementing XML in E-Business

B2B – B2C Applications – Different types of B2B interaction – Components of e-business XML systems – EbXML – RosettaNet - Applied XML in vertical industry – Web services for mobile devices.

9 Hours

Unit V

XML Content Management and Security

Semantic Web – Role of Meta data in web content - Resource Description Framework – RDF schema – Architecture of semantic web – Content management workflow – XLANG – WSFL – Securing web services

9 Hours

Total: 45 Hours

References

1 Ron Schmelzer "XML and Web Services – Unleashed", Pearson Education, 2009.
2 Ethan Ceram, "Web Services Essentials", O'reilly 2012.
3 B V Kumar , S V Subrahmanya "Web Services an Introduction" Tata McGraw-Hill 2008.
4 Frank P Coyle "Xml, Web Services and the Data Revolution", Addison-Wesley, 2002.

13SE54 SOFT COMPUTING

3 0 0 3

Course Objectives (COs):

- To introduce the basics of Fuzzy Set Theory, Optimization and Neural networks
- To learn Feature extraction , Classification and Recognition techniques
- To implement the methods for handwritten characters in Matlab

Course Learning Outcomes (CLOs):

- The students will acquire the theoretical knowledge on Fuzzy Set theory, Optimization and Neural
- The students will learn about supervised and unsupervised Learning methodologies of Neural Networks and its applications

Programme Outcomes (POs):

a. Apply the principles of software engineering and mathematics for the design and implementation of software systems
g. Ability to understand the impact of software systems in societal contexts

Unit I

Fuzzy set theory

Introduction to Neuro Fuzzy and Soft Computing – Fuzzy Sets – Basic Definition and Terminology – Set-theoretic operations – Member Function Formulation and parameterization – Fuzzy Rules and Fuzzy Reasoning - Extension principle and Fuzzy Relations – Fuzzy If-Then Rules – Fuzzy Reasoning – Fuzzy Inference Systems – Mamdani Fuzzy Models - Sugeno Fuzzy Models – Tsukamoto Fuzzy Models – Input Space Partitioning and Fuzzy Modeling

9 Hours

Unit II

Optimization

Derivative-based Optimization – Descent Methods – The Method of steepest Descent – Classical Newton's Method – Step Size Determination – Derivative-free Optimization – Genetic Algorithms – Simulated Annealing – Random Search – Downhill Simplex Search

9 Hours

Unit III

Neural networks

Supervised Learning Neural Networks – Perceptrons - Adaline – Backpropagation Multilayer perceptrons – Radial Basis Function Networks – Unsupervised Learning and Other Neural Networks – Competitive Learning Networks – Kohonen Self – Organizing Networks – Learning Vector Quantization – Hebbian Learning

9 Hours

Unit IV

Neuro fuzzy modeling

Adaptive Neuro-Fuzzy Inference Systems – Architecture – Hybrid Learning Algorithm – learning Methods that Cross-fertilize ANFIS and RBFN – Coactive Neuro-Fuzzy Modeling – Framework – Neuron Functions for Adaptive Networks – Neuro Fuzzy Spectrum

9 Hours

Unit V

Application of computational intelligence

Printed Character Recognition – Inverse Kinematics Problems – Automobile Fuel Efficiency prediction – Soft Computing for Color Recipe Prediction

Case Studies

1 Discuss the logic behind in the football playing robots
2 Discuss the logic of OCR software
3 Implement the back propagation algorithm for the hand written analyzer system

9 Hours

Total: 45 Hours

References

1 S. R. Jang, C. T. Sun, E. Mizutani, *Neuro-Fuzzy and Soft Computing*, Pearson Education, 2007.
2 Davis E. Goldberg, *Genetic Algorithms: Search, Optimization and Machine Learning* Addison Wesley, New York, 2005.
3 S. N. Sivanandam, S. N. Deepa, *Principles Of Soft Computing*, Willey, 2007.

13SE55 DIGITAL IMAGE PROCESSING

3 0 0 3

Course Objectives (COs):

- The primary objective of this course is to introduce students to basic principles of digital images, image data structures, and image processing algorithms.
- Gain experience in applying image processing algorithms to real problems.
- Useful skill base that would allow them to carry out further study should they be interested and to work in the field.

Course Learning Outcomes (CLOs):

- At the end of the course the student should have a clear impression of the breadth and practical scope of digital image processing and have arrived at a level of understanding that is the foundation for most of the work currently underway in this field.
- Explore advanced topics of Digital Image Processing.
- Make a positive professional contribution in the field of Digital Image Processing.

Programme Outcomes (POs):

a. Apply the principles of software engineering and mathematics for the design and implementation of software systems
e. Ability to select and apply appropriate modern tools for the design and development of software solutions

Unit I

Fundamentals of a Digital Image and Transforms

Elements of visual perception – Image sampling and quantization - Basic relationship between pixels – Basic geometric transformations - Two dimensional Orthogonal and Unitary Transforms - Properties of Unitary Transforms - 2 D Discrete Fourier Transform (DFT) – Properties of 2D Fourier Transform – Separable Image Transforms -Walsh – Hadamard – Discrete Cosine Transform, Haar, Slant – Karhunen - Loeve (Hotelling) transforms - Properties of the above listed transforms

9 Hours

Unit II

Image Enhancement Techniques

Spatial Domain methods: Basic grey level transformation – Histogram equalization – Image subtraction – Image averaging –Spatial filtering: Smoothing, sharpening filters – Laplacian filters – Frequency domain filters: Smoothing – Sharpening filters – Homomorphic filtering

9 Hours

Unit III

Image Restoration and Feature Extraction

Model of Image Degradation/restoration process – Noise models – Inverse filtering -Least mean square filtering – Constrained least mean square filtering – Blind image restoration – Pseudo inverse – Singular value decomposition. Image feature description - Interpretation of Line drawings, Image pattern recognition algorithms

9 Hours

Unit IV

Image Compression

Encoder-Decoder model - Types of redundancies - Lossy and Lossless compression - Entropy of an information source, Shannon's 1st Theorem - Huffman Coding - Arithmetic Coding - LZW coding - Run length coding, FAX compression (CCITT Group-3 and Group-4) – LZW coding – BIT plane coding- predictive coding - DPCM. Lossy Compression: Transform coding – Wavelet coding – Image compression standards: JPEG 2000, MPEG, Basics of Vector quantization.

9 Hours

Unit V

Image Segmentation and Representation

Edge detection – Thresholding - Region Based segmentation – Boundary representation: chain codes- Polygonal approximation – Boundary segments – boundary descriptors: Simple descriptors - Fourier descriptors – Regional descriptors –Simple descriptors –Texture

9 Hours

Total: 45 Hours

References

1 Rafael C Gonzalez, Richard E Woods 2nd Edition, “Digital Image processing”, Pearson Education, 2013
2 Anil K Jain, “Fundamentals of Digital Image processing”, Prentice Hall India, 2012.
3 S. Jayaraman, S. Esakkirajan, T. Veerakumar, “Digital Image Processing”, Tata McGraw Hill education private limited, 2009.
4 William K Pratt, “Digital Image Processing”, John Willey,2007.
5 Millman Sonka, Vaclav Hlavac, Roger Boyle, Broos/colic, “Learning Image Processing analysis and Machine Vision”, Thompson, 2006
6 http://nptel.iitm.ac.in/courses/Webcourse-contents/ Digi_Img_Pro

13SE56 SOFTWARE RELIABILITY AND REUSE

3 0 0 3

Course Objectives (COs):

To impart Knowledge on

- Software Reliability
- Reliability approaches
- Reliability models
- Metrics used in software reusable components
- Development of reusable components and Reuse in business

Course Learning Outcomes (CLOs):

- Demonstrate proficiency in rapid software development techniques.
- Able to identify specific components of a software design that can be targeted for reuse.
- Improves proficiency in software development cost estimation.
- Develop a proper software testing and reuse plan.

Programme Outcomes (POs):

a. Apply the principles of software engineering and mathematics for the design and implementation of software systems
g. Ability to understand the impact of software systems in societal contexts

Unit I

INTRODUCTION TO SOFTWARE RELIABILITY

Basic Concepts – Failure and Faults – Environment – Availability –Modeling – uses- Software Reliability Modeling: 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-Software Reliability: A Case Study in a Zero Defect initiative.

9 Hours

Unit II

COMPARISON OF SOFTWARE RELIABILITY MODELS

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 – Reliability Growth Model – Model Evaluation- Successful Application of Software Reliability: A Case Study.

9 Hours

Unit III

MEASUREMENTS THEORY AND RELIABILITY ASSESSMENT

Fundamentals of Measurement – Measurements in Software Engineering – Scope of Software metrics – Measurements theory – Goal based Framework – Software Measurement Validation -- Measurement of Quality – Quality Management Models- Ability to Test Entire System -- Software Reliability Improvement Techniques Measurement of Internet Product Attributes- Orthogonal Classification - A Systematic Review of Software Reliability Studies: A Case Study.

9 Hours

Unit IV

INTRODUCTION TO REUSE MANAGEMENT

Organizing Reuse – Introduction – Motivation for Reuse – Framework for Reuse- Evolution of Reuse - Reuse in industry – Managing a reuse project – Software Reuse Products- Software Reuse Processes and paradigms – Reuse tools-Managing a repository – The REBOOT component model – Classification – Configuration management of the repository – Managerial aspects of software Reuse– Software Reuse Metrics – Software Reuse Cost estimation – Forming a reuse Strategy– Assessing reuse maturity - Software Reuse Case Study – TRILLIUM.

9 Hours

Unit V

REUSE PHASES

Development with reuse – with reuse specific activities – Common reuse processes –Phases of development with reuse – Impact of reuse on development cycle- Reuse Technologies- cleanroom Software Engineering: Overview of cleanroom software engineering – Phases in cleanroom method – Box structures algorithms – Adapting the box structures-case studies on NPP Science Data Segment reuse,S4PM Reuse ,SHAirED Reuse , SOSE.

9 Hours

Total : 45 Hours

References

1. Guadalupe Ortiz, Javier Cubo, "Adaptive Web Services for Modular and Reusable Software Development: Tactics and Solutions", (Premier Reference Source), 2012.
2. Doron A.Peled, "Software Reliability Methods", 2010.
3. John D. Musa, "Software Reliability Engineering:More Reliable Software Faster and Cheaper",Tata McGraw Hill, 2nd edition, 2004.
4. Wayne C.Lim, "Managing Software Reuse", Prentice Hall, 2004.
5. Hafedh Mili, Ali Mili, Sherif Yacoub, "Reuse based Software Engineering: Techniques, Organizations and Controls", John Wiley and Sons, 2002.
6. Michael E.Whitman, Herbert J.Mattord, "Principles of Information Security", Thompson (Vikas Publishing House), 2003.

13SE57 ARTIFICIAL INTELLIGENCE

3 0 0 3

Course Objectives (COs):

- To provide a strong foundation of fundamental concepts in Artificial Intelligence
- To provide a basic exposition to the goals and methods of Artificial Intelligence for developing computer applications
- To enable the student to apply these techniques in applications which involve perception, reasoning and learning

Course Learning Outcomes (CLOs):

- The students will learn about basics of Artificial Intelligence and Symbolic and Uncertainty theories.
- The students will able to design and implement complex AI systems.

Programme Outcomes (POs):

a. Apply the principles of software engineering and mathematics for the design and implementation of software systems
e. Ability to select and apply appropriate modern tools for the design and development of software solutions

Unit I

Introduction

Artificial Intelligence Definition – Importance of Artificial Intelligence – Knowledge based Systems – Knowledge Representation – State space search – Production systems – Artificial Intelligence Programming Language – PROLOG – Heuristic search - Depth First Breadth first –Hill climbing – 4th algorithms – Game Playing.

9 Hours

Unit II

Knowledge Representation

Prepositional Logic – Clause form – Predicate logic – Resolution – Inference Rules –Unification – Semantic networks – frames – conceptual dependency – Scripts – Representing - Knowledge using rules.

9 Hours

Unit III

Symbolic Reasoning And Uncertainty

Non monotanic Reasoning – Truth maintenance systems – closed world assumption – modal and temporal Logics – Bayes Theorem - certainty factors – Bayesian networks – Dempster – Shafer Theory – Fuzzy logic.

9 Hours

Unit IV

Natural Language Processing And Distributed Artificial Intelligence

Overview of Linguistics – grammars and Languages – Basic parsing techniques – semantic Analysis and representation structures – Natural language generation – natural language systems – Distributed Reasoning systems – Intelligent agents.

9 Hours

Unit V

Expert Systems

Architecture – Non production systems Architectures – Knowledge acquisition and validation – Knowledge system building tools – Types of Learning – General Learning model – Learning by induction – Generalization and specialization – Inductive bias – Explanation based Learning.

9 Hours

Total: 45 Hours

References

1. Elaine Rich and Kevin Knight, "Artificial Intelligence" Tata McGraw Hill, 3rd edition, 2009
2. George F Luger, "Artificial Intelligence, structures and strategies for complex problem solving", Pearson Education Delhi, 2009.
3. Dan W. Patterson, "Introduction to Artificial Intelligence and Expert Systems", Prentice Hall of India, Delhi, 2001

13SE58 REAL TIME SYSTEMS

3 0 0 3

Course Objectives (COs):

- To know about the specification and design techniques of a Real Time System.
- To study the real time scheduling approaches.
- To understand about real time task communication and synchronization.

Course Learning Outcomes (CLOs):

- Upon completion of this course, the student will be able to:
- Formulate real-time constraints.
- Employ standard real-time programming constructs.
- Develop a concept to handle the requests level of reliability

Programme Outcomes (POs):

d. Ability to interpret and synthesize information to provide valid conclusions of the existing literature
e. Ability to select and apply appropriate modern tools for the design and development of software solutions

Unit I
Introduction
Basic Real-Time Concepts: Terminology – Systems Concepts – Real-Time Definition – Hard and soft RTS- Processors and resources- temporal parameters of real time workload-Periodic task model-Precedence constraints and data dependency- other types of dependencies-Real Time System Design Issues

9 Hours

Unit II
Real time Scheduling Approach
Resource parameters of jobs and parameters of resources, Scheduling Hierarchy- clock driven approach- weighted round-robin approach-priority driven approach-dynamic versus static systems-effective release times and deadlines-optimality of the EDF and LST algorithm-Non optimality of the EDF and LST algorithms-offline versus online scheduling

9 Hours

Unit III
Hardware Considerations
Basic Architecture – Hardware Interfacing – Latching – Edge versus Level Triggered – Tristate Logic – Wait States – System Interfaces and Buses- CPU-Memory – Memory Access – Memory Technology – Memory Hierarchy – Memory Organization - Programmed I/O - DMA – Memory Mapped I/O – Interrupts - Enhancing performance – Locality of Reference – Cache – Pipelining – Coprocessors

9 Hours

Unit IV
RTOS
Real-Time Kernels – Pseudo kernels – Interrupt - Driven Systems –Preemptive - priority Systems - Hybrid Systems - Process Scheduling – Round-Robin Scheduling - Fixed Priority Scheduling – Dynamic-Priority Scheduling - Inter task Communication and Synchronization - Memory Management

9 Hours

Unit V

Software Systems Design

Requirement Engineering Process – Types of requirement – Requirements specification for Real-Time Systems- Properties of Software- Procedural Orientation Design – Object Orientation Design

9 Hours

Total: 45 Hours

References

1. Phillip A. Laplante, Real-Time Systems Design and Analysis, Third Edition, Wiley-India, 2011.
2. Jane. W. S. Liu, Real Time Systems, Pearson Education, 2008.

13SE59 NETWORK ROUTING

3 0 0 3

Course Objectives (COs):

- To create in-depth awareness of packet routing in computer communication networks
- To provide comprehensive details of routing algorithms, protocols and architectures of routers

Course Learning Outcomes (CLOs):

Students will be able to:

- Understand network routing and algorithms
- Compare routing techniques and protocols
- Explain Router Architectures and IP Address Lookup Algorithms
- Explain Routing in Ad hoc Network

Programme outcomes (POs):

d. Ability to interpret and synthesize information to provide valid conclusions of the existing literature
e. Ability to select and apply appropriate modern tools for the design and development of software solutions

Unit I

Network Routing and Algorithms

Networking and Network Routing: An Introduction: Addressing and Internet Service: An Overview, Network Routing: An Overview, IP Addressing, On Architectures, Service Architecture, Protocol Stack Architecture, Router Architecture, Network Topology Architecture, Routing Algorithms: Shortest Path and Widest Path:Bellman–Ford Algorithm and the Distance Vector Approach, Dijkstra's Algorithm, Comparison of the Bellman–Ford Algorithm and Dijkstra's Algorithm, Shortest Path Computation with Candidate Path Caching, Widest Path Computation with Candidate Path Caching, Widest Path Algorithm, k-Shortest Paths Algorithm

9 Hours

Unit II

Routing Protocols

Routing Protocols: Framework and Principles:Routing Protocol, Routing Algorithm and Routing Information, Representation and Protocol Messages, Distance Vector Routing Protocol, Link State Routing Protocol, Path Vector Routing Protocol, Link Cost, IP Routing and Distance Vector Protocol Family: Routers, Networks, and Routing Information: Some Basics, Static Routes, Routing Information Protocol Version 1 (RIPv1), Routing Information Protocol Version 2 (RIPv2), Interior Gateway Routing Protocol (IGRP), Enhanced Interior Gateway Routing Protocol (EIGRP), Route Redistribution

9 Hours

Unit III

OSPF, Integrated IS-IS and BGP

OSPF: Protocol Features, OSPF Packet Format, Examples of Router LSAs and Network LSAs, Integrated IS-IS: Key Features, Similarities and Differences Between IS-IS and OSPF, BGP:A Brief Overview, Basic Terminology, BGP Operations: Message Operations, BGP Timers, BGP Configuration Initialization, Two Faces of BGP: External BGP and Internal BGP, Path Attributes, BGP Decision Process, Internal BGP Scalability, Significance of Route Flap Dampening, BGP Additional Features, Finite State Machine of a BGP Connection

9 Hours

Unit IV

Router Architectures and IP Address Lookup Algorithms

Router Architectures: Functions of a Router, Types of Routers, Elements of a Router, Packet Flow, Packet Processing: Fast Path versus Slow Path, Router Architectures, IP Address Lookup Algorithms: Impact of Addressing on Lookup, Longest Prefix Matching, Naïve Algorithms, Binary Tries, Multi bit Tries, Search by Length Algorithms, Search by Value Approaches

9 Hours

Unit V

Towards Next Generation Routing and Routing in Ad hoc Network

Towards Next Generation Routing: Background of QoS and QoS Routing, QoS Attribues, Traffic Engineering Extension to Routing Protocols, Multiprotocol Label Switching (MPLS), Generalized MPLS, MPLS Virtual Private Networks, Routing in Ad hoc Network: Introduction to Ad hoc Networks – Features/ Characteristics, Types and Applications, Limitations, Advantages and Disadvantages, Classification of Routing Protocols in Ad hoc Networks – Proactive Routing Protocols (DSDV, OLSR), Reactive Routing Protocols (DSR, AODV), Hybrid Routing Protocols (ZRP)

9 Hours

Total: 45 Hours

References

1. Deepankar Medhi, Karthikeyan Ramasamy, “Network Routing – Algorithms, Protocols, Architecture”, Morgan Kauffman Series Publication, 2007.
2. Subir Kumar Sarkar, T G Basavaraju and C Puttamadappa, “Ad Hoc Mobile Wireless Networks – Principles, Protocols and Applications”, Auerbach publications, 2008.
3. Dharma Prakash Agrawal and Carlos De Morais Cordeiro, “Adhoc and Sensor Networks – Theory and Applications”, World Scientific publication, 2011.

10SE60 ENTERPRISE RESOURCE PLANNING

3 0 0 3

Course Objectives (COs):

- To apply the basic concepts, tools and techniques in developing Enterprise Resource Planning applications.
- understanding the business model and implementing ERP

Course Learning Outcomes (CLOs):

- Student will be able to design and implement ERP for real time Applications
- Student will be able to create Web based ERP Applications

Programme Outcomes (POs):

a. Apply the principles of software engineering and mathematics for the design and implementation of software systems
e. Ability to select and apply appropriate modern tools for the design and development of software solutions

Unit I

Introduction

ERP Concepts -Enterprise System - Tangible and Intangible Benefits -Emerging Trends in ERP adoption - ERP Implementation Stages- ERP as Integrated Management Information System- Evolution of ERP-ERP vs Traditional Information Systems

9 Hours

Unit II

Implementation stage

ERP Implementation Lifecycle-Implementation Methodology- Hidden Costs- Organizing the Implementation - ERP Implementation- issues- Role of Consultants- Vendors- Users- Need for training - customization- ERP implementation methodology and post
implementation issues and options

9 Hours

Unit III

The business modules

Business modules in an ERP Package, Finance, Manufacturing, Human Resources, Plant Maintenance, Materials Management, Quality Management, Sales and Distribution

9 Hours

Unit IV

Project management

Project Management – Project Team – Steering Committee – Project Manager – Functional Team – IS Team – Security Specialists. Project Deliverables – Change Management – System integration – Systems Integration standards – Middleware Development – Forward and Reverse Engineering – ERP Infrastructure Planning – System Architecture

9 Hours

Unit V

ERP – present and future

Turbo Charge the ERP System, EIA, ERP and e-Commerce, ERP and Internet, Future Directions

9 Hours

Total: 45 Hours

References

1. Alexis Leon, “Enterprise Resource Planning Demystified”, Tata McGraw-Hill Publishing Company Ltd., New Delhi, 2008
2. Mahadeo Jaiswal and Ganesh Vanapalli, “Enterprise Resource Planning”, Macmillan India Ltd., 2005.
3. S Sadagopan, “Enterprise Resource Planning”, PHI, New Delhi, 2004.

13SE61 MULTIMEDIA SYSTEMS

3 0 0 3

Course Objectives (COs):

- To learn the devices and tools for generating and representing multimedia
- To study the text and images in multimedia
- Learning how to organize the Multimedia Project and building intelligent systems

Course Learning Outcomes (CLOs):

- Enables the students to develop a Real time Processing system with better adaptability for multimedia files with greater speed and effective memory consumption
- Provides the better architectural understanding between traditional file systems and the modern file systems supporting Multimedia content types
- Ensures the development of the product with all the QoS requirements being satisfied
- Globalizes the multimedia content by providing thorough knowledge on various Networking Architectures and devices used for transferring data packets universally
- Offers the practical knowledge on usage of various lab components and editing software

Programme Outcomes (POs):

d. Ability to interpret and synthesize information to provide valid conclusions of the existing literature
e. Ability to select and apply appropriate modern tools for the design and development of software solutions

Unit I

Introduction and QoS

Introduction - QoS Requirements and Constraints – Concepts -Resources - Establishment Phase - Run-Time Phase - Management Architectures

9 Hours

Unit II

Operating systems

Real Time Processing – Scheduling - Interprocess Communication - Memory and Management - Server Architecture - Disk Management

9 Hours

Unit III

File systems and networks

Traditional and Multimedia File Systems - Caching Policy – Batching - Piggy backing –Ethernet - Gigabit Ethernet - Token Ring - 100VG Any LAN - Fiber Distributed Data Interface (FDDI) - ATM Networks - MAN – WAN

9 Hours

Unit IV

Communication

Transport Subsystem - Protocol Support for QoS - Transport of Multimedia - Computer Supported Cooperative Work – Architecture - Session Management - MBone Applications

9 Hours

Unit V

Synchronization

Synchronization in Multimedia Systems - Presentation-Synchronization Types -Multimedia Synchronization Methods - Case Studies – MHEG – MODE - ACME.

9 Hours

Total: 45 Hours

References:

1 Ralf Steinmetz and Klara Nahrstedt, “Multimedia Systems”, Springer, Edition 2004.
2 Ralf Steinmetz and Klara Nahrstedt, “Media Coding and Content Processing”, Prentice Hall of India, 2002.
3 Ze-Nian Li and Mark S. Drew, “Fundamentals of Multimedia”, Pearson Education, 2004.

13SE62 MOBILE AND PERVASIVE COMPUTING

3 0 0 3

Course Objectives (COs):

- To learn the basics of Wireless voice and data communications technologies.
- To provide the student with knowledge and skills about a new trend in computing

Course Learning Outcomes (CLOs):

Students will be able to:

- analyse the strengths and limitations of the tools and devices for development of pervasive computing systems
- Apply current software development methodologies, working effectively as an individual or within a team, in the production of a substantial piece of pervasive computing software in consultation with a client

Programme outcomes (POs):

a. Apply the principles of software engineering and mathematics for the design and implementation of software systems
e. Ability to select and apply appropriate modern tools for the design and development of software solutions

Unit I

Introduction

Wireless networks- emerging technologies- Bluetooth, WiFi, WiMAX,3G, WATM.-Mobile IP protocols -WAP push architecture-Wml scripts and applications

9 Hours

Unit II

Mobile computing management

Mobile computing environment – functions - architecture - design considerations, content architecture - CC/PP exchange protocol, context manager - Data management in WAE - Coda file system - caching schemes-Mobility QOS - security in mobile computing

9 Hours

Unit III

Mobile networks scheme

Handoff in wireless mobile networks - reference model - handoff schemes - Location management in cellular networks - Mobility models - location and tracking management schemes - time, movement, profile and distance based update strategies - ALI technologies

9 Hours

Unit IV

Pervasive computing

Principles, Characteristics - interaction transparency, context aware, automated experience capture -Architecture for pervasive computing - Pervasive devices - embedded controls - smart sensors and actuators - Context Communication and access services

9 Hours

Unit V

Context awareness

Open protocols - Service discovery technologies - SDP, Jini, SLP, UPNP protocols – data synchronization-SyncML framework - Context aware mobile services - Context aware sensor networks, addressing and communications - Context aware security

9 Hours

Total: 45 Hours

References

1. Asoke K Taukder, Roopa R Yavagal, “Mobile Computing”, Tata McGraw Hill Pub Co., New Delhi, 2005.
2. Uwe Hansmann et al.,“Pervasive Computing”,Springer, New York, 2013.
3. Seng Loke, “Context-Aware Computing Pervasive Systems”, Auer Bach Pub., New York, 2007.
4. Ivan Stojmenovic, “Handbook of Wireless Networks and Mobile Computing”, John Wiley & sons Inc, Canada, 2002.

10SE63 SOFTWARE ARCHITECTURE

3 0 0 3

Course Objectives (COs):

- Introduce the relationship between software requirements and architecture
- To understand the relationship between various architecture styles
- To study the elements of software architecture

Course Learning Outcomes (CLOs):

- Apply basic dependency analysis and visualization techniques and tools to understand the architectural design of an existing software system
- Understand existing architectural descriptions and recognized common architectural styles and patterns layered, peer-to-peer, service-oriented, etc.) and evaluate architectural documentation for quality and completeness
- Describe and explain common techniques for analyzing and managing architectural knowledge
- Create a complete architectural description of an existing system

Programme Outcomes (POs):

b. Apply the principles of software engineering and mathematics for the design and implementation of software systems
f. Ability to select and apply appropriate modern tools for the design and development of software solutions
g. Ability to understand the impact of software systems in societal contexts

Unit I

Introduction

Introduction to software Architecture- Architectural Structures and Views - Architectural Patterns- Inhibiting or enabling a system's Quality attributes – Reasoning about and managing changes- Predicting system's Qualities- Enhancing communication among stakeholders - The main contexts of software architecture-Architecture in a project life cycle context - Architecture in a business context - Architecture in a professional context.

9 Hours

Unit II

Quality Attributes

Understanding Quality attributes – Architecture and Requirements, Functionality, Quality attributes consideration, Specifying Quality Attribute Requirements, Guiding Quality Design decisions- Availability, Interoperability, and Modifiability.

9 Hours

Unit III

Architecture Design

Architecture in the Life Cycle – Architecture in Agile projects-Architecture and Requirements-Designing an architecture-Design strategy, The Attribute-Driven Design Methods- The steps of ADD- Documenting Software Architecture- Architecture, Implementation and Testing.

9 Hours

Unit IV

Architecture Evaluation

Evaluation Factor- The Architecture Tradeoff Analysis Method – Lightweight Architecture Evaluation – Economic Analysis of Architecture- Decision-Making Context, The basis for the Economic Analysis, Putting theory into practices The CBAM- Case Study: The NASA ECS project.

9 Hours

Unit V

Software Architecture in the Future

Architecture in the cloud- Basic cloud definition, Service Models and Deployment Options, Sample Technologies, Architecting in a Cloud Environment, Architectures for the Edge-The Ecosystem of Edge-Dominant Systems, Changes to the Software Development Lifecycle.-The DirSA case study.

9 Hours

Total: 45 Hours

References

1. Len Bass, Paul Clements, Rick Kazman “Software Architecture in Practice”, Addison-Wesley Professional, Third Edition, 2013.
2. Mary Shaw, David Garlan, "Software Architecture Perspectives on an emerging discipline ", Prentice Hall of India, 2012.
3. Wolfgang Pree, “Design patterns for object Oriented Software Development ", Addison Wesley, 2010.

13SE64 RESEARCH METHODOLOGIES

3 0 0 3

Course Objectives (COs):

- To understand the appropriateness and problem being investigated and level of data analysis
- To develop different steps following in reporting and thesis writing.
- Analysis the quantitative and qualitative research methods moderately competing with ethical impacts.

Course Learning Outcomes (CLOs):

- Students will be able to identify and develop data collection instruments and measures for planning and conducting sociological research.
- Students will be able to distinguish and classify different steps to organize their report and thesis writing.

Program Outcomes (POs)

l. Ability to engage in independent and life-long learning

Unit I

Objectives and Types of Research

Motivation and objectives - Research methods *vs* Methodology. Types of research - Descriptive *vs*. Analytical, Applied *vs*. Fundamental, Quantitative *vs*. Qualitative, Conceptual *vs*. Empirical. Defining and formulating the research problem - Selecting the problem - Necessity of defining the problem - Importance of literature review in defining a problem - Literature review - Primary and secondary sources - reviews, treatise, monographs - patents - web as a source - searching the web - Critical literature review - Identifying gap areas from literature review - Development of working hypothesis.

10 Hours

Unit II

Research Design and Methods

Research design - Basic Principles - Need of research design - Features of good design - Important concepts relating to research design - Observation and Facts, Laws and Theories, Prediction and explanation, Induction, Deduction, Development of Models. Developing a research plan - Exploration, Description, Diagnosis, Experimentation. Determining experimental and sample designs.

9 Hours

Unit III

Data Collection and Analysis

Execution of the research - Observation and Collection of data - Methods of data collection - Sampling Methods- Data Processing and Analysis strategies - Data Analysis with Statistical Packages - Hypothesis - testing - Generalization and Interpretation.

8 Hours

Unit IV

Reporting and Thesis Writing

Structure and components of scientific reports - Types of report - Technical reports and thesis - Significance - Different steps in the preparation - Layout, structure and Language of typical reports - Illustrations and tables - Bibliography, referencing and footnotes - Oral presentation - Planning - Preparation - Practice - Making presentation - Use of visual aids - Importance of effective communication.

9 Hours

Unit V

Application of Results and Ethics

Environmental impacts - Ethical issues - ethical committees - Commercialisation - Copy right - royalty - Intellectual property rights and patent law - Trade Related aspects of Intellectual Property Rights - Reproduction of published material - Plagiarism - Citation and acknowledgement - Reproducibility and accountability.

9 Hours

Total: 45 Hours

References

1 Garg, B.L., Karadia, R., Agarwal, F. and Agarwal, U.K., "An introduction to Research Methodology", RBSA Publishers, 2002.
2 Kothari, C.R., "Research Methodology: Methods and Techniques", New Age International, 2009.
3 Sinha, S.C. and Dhiman, A.K., "Research Methodology", Ess Ess Publications, 2002.
4 Trochim, W.M.K., "Research Methods: the concise knowledge base", Atomic Dog Publishing, 2005.
5 Wadehra, B.L., "Law relating to patents, trademarks, copyright designs and geographical indications", Universal Law Publishing", 2000.

13SE65 SOFTWARE DEVELOPMENT SKILLS

3 0 0 3

Course Objectives (COs):

- The main objective of the course is to help the students get conceptual knowledge for various methods used under software development process
- Ability to design, synthesize, and analyze, software systems of increasing size and complexity at various abstraction levels, from the individual component to the entire system architecture

Course Learning Outcomes (CLOs):

- Ability to define, assess, and tailor software quality practices, and software processes and methodologies for appropriate application on software development projects in a variety of domain areas
- Ability to be an effective member of a multi-disciplinary software-intensive product development with an awareness of individual professional and ethical responsibilities

Programme Outcomes (POs):

a. Apply the principles of software engineering and mathematics for the design and implementation of software systems
e. Ability to select and apply appropriate modern tools for the design and development of software solutions
g. Ability to understand the impact of software systems in societal contexts

Unit I

Concepts and Issues in Design and Development of Software

Hardware – Software - relationship between hardware and software - Structured Programming approach – prototyping approach – Rapid Application Development (RAD) approach – End user approach.

9 Hours

Unit II

Introduction to Software Development

Defining the problem – Abstraction / Refinement – Data types – Structured algorithm – building the software solution – coding in an appropriate PL – error correction techniques – user interface development – documentation – test data – evaluation of design – modifying solutions.

9 Hours

Unit III

Developing Software Solutions

SDLC (Analysis- Design - Implementation - Coding) - Implementing Project – project management – project documentation – testing the solution – reporting –Maintenance.

9 Hours

Unit IV

Software Testing

Objectives, principles, testability, Test cases: White box & Black box testing, Testing strategies: verification and validation, unit test, integration testing, validation testing, system testing.

9 Hours

Unit V

Case Studies

Modern methods in software engineering - Software Engineering tools - GameForge – Eclipse - OpenOffice - Gaming requirements.

9 Hours

Total: 45 Hours

References

1. Ian Somerville, "Software Engineering", eighth edition Addison-Wesley Longman, Incorporated, 2007
2. Abran, Alain; Moore, James W.; Bourque, Pierre; Dupuis, Robert; Tripp, Leonard L. (2004). Guide to the Software Engineering Body of Knowledge. IEEE. ISBN 0-7695-2330-7.
3. Pressman, Roger S (2010). Software Engineering: A Practitioner's Approach (7th ed.). Boston, Mass: McGraw-Hill. ISBN 0-07-285318-2.

13SE66 OPEN SOURCE SOFTWARE

3 0 0 3

Course Objectives (COs):

- Define open source software
- Understand the motivation, theory, strengths and weaknesses of open source software.

Course Learning Outcomes (CLOs):

- Being able to approach and apply advanced frameworks and tools in a real project.
- Identify and discuss various software licensing models

Programme Outcomes (POs):

a. Apply the principles of software engineering and mathematics for the design and implementation of software systems
d. Ability to interpret and synthesize information to provide valid conclusions of the existing literature
e. Ability to select and apply appropriate modern tools for the design and development of software solutions

Unit I
History and Emergence of Open Source Software
The philosophy of OSS, Richard Stallman, The Cathedral and the Bazaar (CatB), commercial software vs OSS, free software vs freeware. Open source development models. Application Programming Interface (API). GNU Project, Free Software Foundation.

9 Hours

Unit II
Community Building
Importance of Communities in Open Source Movement. JBoss Community. Developing blog, group, forum, social network for social purpose.

9 Hours

Unit III
Open Standards
National Information Standards Organization (NISO), The Digital Library Federation (DLF). The Dublin Core Metadata Initiative. MARC standards, Resource Description and Access (RDA). Open Archives Initiative. OAI-PMH. Search / Retrieval via URL (SRU), SRW/CQL. Java Platform, Enterprise Edition (Java EE).

9 Hours

Unit IV
Open Source Licenses
GNU General Public License (GPL) version 2,3, GNU Lesser General Public License (LGPL) version 2.1,3, GNU Affero General Public License (AGPL) version 3, Apache License, Version 2.0, Artistic License 2.0, etc.

9 Hours

Unit V
Case Studies
Operating System - The Linux operating system and its use both for desktops and as server software. Webserver - Apache HTTP Server and its flavors. WAMP server (Windows, Apache, MySQL, PHP). Open Source MySQL. Apache, MySQL, PHP, JAVA as development platform. Open Source Software - Category of Open Source Software. OSS for podcasts, RDBMS, online social networks. Eclipse development Environment - open source bibliometric softwares like pajek, ucinet, etc.

9 Hours

Total: 45 Hours

References

1 B. Venkateshwarlu (Ed); Introduction to Linux: Installation and Programming, B S Publishers; 2005.
2 Matt Welsh, Matthias Kalle Dalheimer, Terry Dawson, and Lar Kaufman, Running Linux, Fourth Edition, O'Reilly Publishers, 2002.
3 Carla Schroder, Linux Cookbook, First Edition, O'Reilly Cookbooks Series, 2004
4 The Linux Cookbook: Tips and Techniques for Everyday Use, First Edition,Michael Stutz, 2001. URL: http://dsl.org/cookbook/cookbook_toc.html
5 http://directory.fsf.org/GNU/

13SE67 APP ENGINE

3 0 0 3

Course Objectives (COs):

- To introduce the fundamentals of cloud computing and App Engine
- To Introduce the Google Apps Scripts
- To introduce the Google Cloud Platform and its various features
- To Introduce the Google Cloud Sql for Application Development

Course Learning Outcomes (CLOs):

- Being able to apply App Engine for implementing real time projects in cloud
- Ability to use Google Apps Scripts

Programme Outcomes (POs):

d. Ability to interpret and synthesize information to provide valid conclusions of the existing literature
e. Ability to select and apply appropriate modern tools for the design and development of software solutions

Unit I

Introducing Cloud Computing and services:

Introducing Cloud Computing, Software as a Service, Platform as a service, Infrastructure as a service, Identity as a service.

9 Hours

Unit II

App Engine:

Introducing App Engine, The runtime environment, The static file servers, The datastore, services, Google Accounts, Administration console

9 Hours

Unit III

Google Apps Scripts:

Google Apps Scripting, Spreadsheet Limitations, Apps Script Examples, User Defined Functions, Spreadsheet and sheets, Range Object

9 Hours

Unit IV

Google Cloud Platform

Introduction to Google Cloud Platform, Infrastructure, Google Compute Engine, Google Cloud Storage, Introduction to Google App Engine, Data Store and Cloud SQL, Advanced Level – Google Bigquery, App Engine 101, Setting up the SDK, Hello World application. GAE admin console, Understanding the Java webapp environment in GAE – Servlets, JPA, Forms, Users Service, Datastore – Overview, API, Queries, Transactions

9 Hours

Unit V

Google Cloud SQL and Storage

Google API console, Using Google Cloud SQL with app engine, Managing the instances, Tools – Command line, Squirrel, Using with app engine. Google Cloud Storage, Overview, API – Authorization, using client libraries, Using with app engine, A static website with Google Cloud Storage

9 Hours
Total: 45 Hours

References

1. Kris Jamsa, "Cloud Computing", Jones & Bartlett Learning, 2013
2. Opabinia,"Google Spreadsheet Programming,Use JavaScript To Build Spreadsheet Applications In The Cloud", leanpub, 2013
3. Dan Sanderson, “Programming Google App Engine”, O’Reilly 2010

13SE01 BIG DATA ANALYTICS

3 0 0 3

Course Objectives (COs):

- To study about Big Data Analytics with the help of cloud resource fundamentals.
- To develop cloud data systems are secure and dynamic in data transfer.
- To analyze the graph modeling and analytics of operation in data mining methods.

Course Learning Outcomes (CLOs):

- Students will be competent to analysis the data patterns that helps make decisions about how to improve at the industry and institutional level.
- Students will be able to identify and develop data analysis that helps improve team practices in distributed cloud services environment.

Program Outcomes (POs)

h. Apply ethical principles and commit to the norms of professional practice
i. Ability to function effectively as an individual
l. Ability to engage in independent and life-long learning

Unit I

Introduction

Introduction to Big Data Analytics - Cloud Resource Fundamentals - Cloud & Big Data Overview - Cloud Storage - Cloud Data Transfer - Hadoop - DBs in the Cloud.

9 Hours

Unit II

Cloud Data Systems

Cloud DBs - Overview of SQL and intro to R DBs - Using R for Initial Analysis of the Data - Relational Storage Models - Data Sharing Incentives and Markets - Key Value Stores - In-Memory Data Systems - Data Transfer - Security: Data Protection, models - Data Consistency and Availability in the Cloud.

9 Hours

Unit III

Graphical Models

Pipelines and Graphs - Parallel programming environment - Graphical Models - Graph structure learning - Graph mining - Graph modeling - Time series analysis. Cluster Sizing for Data - intensive Analytics - Advanced Analytics and Statistical Modeling for Big Data: Theory and Methods - Statistical Modeling for Big Data: Technology and Tools.

9 Hours

Unit IV

Operationalizing Analytics

Massive Data Analytics: parallel algorithms - online learning algorithms - locality sensitive hashing - Big Data Analytics Lifecycle - Programming Abstractions: MR, Dryad, Pipelines, Graphs, Iteration - Cloud Storage - Classic Systems - intensive cloud applications - data confidentiality - In-Memory Data Storage - Cloud Data Privacy and Security.

9 Hours

Unit V

Cloud Case Studies

Team practices statistical learning - data mining techniques & process of data analytics - troubleshoot massive data analytics algorithms - Finding a needle in Haystack: face book's photo storage - Volley: Automated Data Placement for Geo-Distributed Cloud Services - Scaling the mobile millennium system in the cloud.

9 Hours

Total: 45 Hours

References

1 Arvind Sathi, "Big Data Analytics: Disruptive Technologies for Changing the Game", Kindle Edition, Mc Press, 2012.
2 C. Bishop, "Pattern Recognition and Machine Learning", Springer 2007.
3 Noreen Burlingame, "The Little Book of BIG DATA", Kindle Edition, New Street Communications, LLC, 2012.
4 Trevor Hastie, Robert Tibshirani, Jerome. H. Friedman, "The elements of statistical learning: data mining, inference and prediction", Springer, 2009.
5 http://www-stat-class.stanford.edu/~tibs/ElemStatLearn/
6 http://www.cs.ucsb.edu/~sudipto/papers/socc10-das.pdf
7 http://www.cs.ucsb.edu/research/tech_reports/reports/2010-04.pdf
8 http://www-users.cselabs.umn.edu/classes/Fall-2012/csci8980-2/
9 http://deesha-cloudviews.blogspot.in/