R19

M. Tech – SE

SIDDARTHA INSTITUTE OF SCIENCE AND TECHNOLOGY:PUTTUR
(AUTONOMOUS)
Master of Technology
Department of Computer Science & Engineering
Specialization: Software Engineering

I M. Tech. – I Semester (SE)

S. No.	Course Code	Subject	L	T	P	C
1	19HS0841	Discrete Mathematics and Applications	3	0	0	3
2	19CS5101	Object Oriented Software Engineering	3	0	0	3
3	19CS5102 19CS5103 19CS5104	**Program Elective I** Software Requirements & Estimation Professional Aspects in Software Engineering Protocol Software Engineering	3	0	0	3
4	19CS5105 19CS5106 19CS5107	**Program Elective II** Formal Methods of Software Engineering Software Metrics & Reuse Service Oriented Architecture	3	0	0	3
5	19HS0823	Research Methodology and IPR	2	0	0	2
6	19CS5108	Object Oriented Software Engineering Lab	0	0	4	2
7	19CS5109	Software Requirements & Estimation Lab	0	0	4	2
8	19HS0818 19CE1029 19HS0825 19HS0826	**Audit I** English for Research PaperWriting Disaster Management Sanskrit for Technical Knowledge Value Education	2	0	0	0
		Contact Periods / Week	16	0	8	**18**
			Total/Week 24			

I M. Tech. – II Semester (SE)

S. No.	Course Code	Subject	L	T	P	C
1	19CS5110	Advances in Software Testing	3	0	0	3
2	19CS5111	Software Patterns	3	0	0	3
3	19CS5112 19CS5113 19CS5114	**Program Elective III** Software Quality Assurance Software Reliability Model Driven Software Engineering	3	0	0	3
4	19CS5115 19CS5116 19CS5117	**Program Elective IV** Software Reengineering Big Data Analytics Software Configuration Management	3	0	0	3
5	19CS5118	Advances in Software Testing Lab	0	0	4	2
6	19CS5119	Software Patterns Lab	0	0	4	2
7	19CS5120	Mini Project with Seminar	2	0	0	2
8	19HS0829 19HS0827 19HS0828 19HS0819	**Audit II** Constitution of India Pedagogy Studies Stress Management by Yoga Personality Development through Life Enlightenment Skills.	2	0	0	0
Contact Periods / Week			16	0	8	**18**
			Total/Week 22			

II M. Tech. – I Semester (SE)

S. No.	Course Code	Subject	L	T	P	C
1	19CS5121 19CS5122 19CS5123	**Program Elective V** Secure Software Engineering Agile Methodologies Fundamentals of Data Science	3	0	0	3
2	19HS0824 19ME3121 19ME3021 19CE1028 19ME3022 19EE2128	**Open Elective** 1. Business Analytics 2. Industrial Safety 3. Advances in Operations Research 4. Cost Management of Engineering Projects 5. Composite Materials 6. Waste to Energy	3	0	0	3
3	19CS5124	Dissertation Phase – I/ Industrial Project	0	0	20	10
		Contact Periods / Week	6	0	20	**16**
			Total/Week 26			

II M. Tech. – I Semester (SE)

S. No.	Course Code	Subject	L	T	P	C
1	19CS5125	Dissertation Phase – II	0	0	32	16
		Contact Periods / Week	0	0	32	**16**
			Total/Week 32			

Note: L – Lecture hour, T – Tutorial, P- Practical, C-Credit

Total Credits: 18 + 18 + 16 + 1 6 = 78

R19 **M. Tech – SE**

SIDDARTHA INSTITUTE OF SCIENCE AND TECHNOLOGY:PUTTUR
(AUTONOMOUS)

I M. Tech. – I Sem. (SE)	**L**	**T**	**C**
	3	**0**	**3**

(19HS0841) DISCRETE MATHEMATICS AND APPLICATIONS

Course Objectives:

- To understand the mathematical fundamentals that is prerequisites for a variety of courses like Data mining, Network protocols, analysis of Web traffic, Computer security, Software engineering, Computer architecture, operating systems, distributed systems, Bioinformatics, Machine learning
- To develop the understanding of the mathematical and logical basis to many modern Techniques in information technology like machine learning, programming language design, and concurrency.

Course Outcomes:

After completion of the course, students would be able to:

- To understand the basic notions of discrete and continuous probability.
- To understand the methods of statistical inference, and the role that sampling distributions play in those methods.
- To be able to perform correct and meaningful statistical analyses, of simple to moderate complexity.

UNIT I

Probability mass, density, and cumulative distribution functions, Parametric families of distributions, Expected value, variance, conditional expectation, Applications of the univariate and multivariate Central Limit Theorem, Probabilistic inequalities, Markov chains.

UNIT II

Random samples, sampling distributions of estimators, Methods of Moments and Maximum Likelihood.

UNIT III

Graph Theory: Isomorphism, Planar graphs, graph colouring, hamilton circuits and euler cycles. Permutations and Combinations with and without repetition. Specialized techniques to solve combinatorial enumeration problems.

UNIT IV

Computer science and engineering applications

Data mining, Network protocols, analysis of Web traffic, Computer security, Software engineering, Computer architecture, operating systems, distributed systems, Bioinformatics, Machine learning.

UNIT V

Recent Trends in various distribution functions in mathematical field of computer science for varying fields like bio informatic, soft computing, and computer vision.

Text /References:

1. John Vince, Foundation Mathematics for Computer Science, Springer.
2. K. Trivedi.Probability and Statistics with Reliability, Queuing, and Computer Science Applications, Wiley.
3. M. Mitzenmacher and E. Upfal.Probability and Computing: Randomized Algorithms and Probabilistic Analysis.
4. Alan Tucker, Applied Combinatorics, Wiley

R19 **M. Tech – SE**

SIDDARTHA INSTITUTE OF SCIENCE AND TECHNOLOGY:PUTTUR
(AUTONOMOUS)

I M. Tech. – I – Semester(SE)

L	T	P	C
3	0	0	3

(19CS5101) OBJECT ORIENTED SOFTWARE ENGINEERING

Course Objectives:

- To investigate principles of object-oriented software engineering, from analysis through testing
- To learn techniques at each stage of development, including use cases and UML
- To practice these principles and techniques by developing a "real world" software system prototype
- To learn software development life cycle for Object-Oriented solutions for Real-World Problems.
- Identify and represent domain constraints on the objects and (or) on their relationships.
- Use Case Studies and tools for OOA design.

Course Outcomes:

- After completing, one should be able to
- Understand the fundamental principles underlying Object-Oriented software design.
- Employ formal methods to produce effective software designs as solutions to specific tasks.
- Develop the design structure using Object-Oriented Principles to achieve overall programming goals.
- Take responsibility for your own learning through reading and the preparation of assignments, and reflect upon your learning experience.

UNIT I

Introduction to Software Engineering: Introduction-Software Engineering Failures, What is Software Engineering? Software Engineering Concepts, Software Engineering Development Activities, Managing Software Development. Modeling with UML: Introduction, Overview of UML, Modeling Concepts, ARENA Case Study

UNIT II

Analysis: Introduction –An Optical Illusion, Overview of Analysis, Analysis Concepts, Analysis Activities, Managing Analysis, ARENA Case Study

UNIT III

System Design: Decomposing the System-Introduction: A Floor Plan Example, Overview of System Design, System Design Concepts, System Design Activities: From Objects to Subsystems

System Design: Addressing Design Goals: Introduction- A Redundancy Example, Overview of System Design Activities, Concepts: UML Deployment Diagrams, Addressing Design Goals, Managing System Design, ARENA Case Study

UNIT IV

Object Design: Specifying Interfaces: Introduction - A Railroad Example, Overview of Interface Specification, Interface Specification Concepts, Interface Specification Activities, Managing Object Design, ARENA Case Study

Mapping Models to Code: Introduction – A Book Example, Overview of Mapping, Mapping Concepts, Mapping Activities, Managing Implementation, ARENA Case Study

UNIT V

Testing: Testing overview, concepts, activities and managing testing

Project Management: Project management overview, concepts, activities and managing project management models and activities

TEXT BOOKS

1. Bernd Bruegge & Allen H. Dutoit, "Object-Oriented Software Engineering", 2009.
2. Ivar Jacobson, "Object-Oriented Software Engineering", Pearson Education, 2009.

REFERENCES

1. Stephen R. Schach, "Object-Oriented Classical Software Engineering", Mc Graw Hill, 2010.
2. Yogesh Singh, "Object-Oriented Software Engineering", 2012

R19 | **M. Tech – SE**

SIDDARTHA INSTITUTE OF SCIENCE AND TECHNOLOGY:PUTTUR
(AUTONOMOUS)

I M. Tech. – I – Semester(SE)

L	T	P	C
3	0	0	3

Program Elective I
(19CS5102) SOFTWARE REQUIREMENTS & ESTIMATION

Course Objectives:

The course should enable the student

- To demonstrate knowledge of the distinction between critical and non- critical systems.
- To demonstrate the ability to manage a project including planning, scheduling and risk Assessment / management.
- To author a software requirements document.
- To demonstrate an understanding of the proper contents of a software requirements document.
- To author a formal specification for a software system.

Course Outcomes:

- Student can able to demonstrate an understanding of distributed system architectures and application architectures.
- Student can able to demonstrate an understanding of the differences between real-time and non-real time systems.
- Student can able to demonstrate proficiency in rapid software development techniques.
- Student can able to demonstrate proficiency in software development cost estimation

UNIT-I

Software Requirements: What and Why Essential Software requirement, Good practices for requirements engineering, Improving requirements processes, Software requirements and risk management.

UNIT II

Software Requirements Engineering : Requirements elicitation, requirements analysis documentation, review, elicitation techniques, analysis models, Software quality attributes, risk reduction through prototyping, setting requirements priorities, verifying requirements quality.

UNIT-III

Software Requirements Modeling: Use Case Modeling, Analysis Models, Dataflow diagram, state transition diagram, and class diagrams, Object analysis, Problem Frames.

Software Requirements Management: Requirements management Principles and practices, Requirements attributes, Change Management Process, Requirements Traceability Matrix, Links in requirements chain.

UNIT - IV

Software Estimation: Components of Software Estimations, Estimation methods, Problems associated with estimation, Key project factors that influence estimation.

Size Estimation: Two views of sizing, Function Point Analysis, Mark II FPA, Full Function Points, LOC Estimation, Conversion between size measures.

Effort, Schedule and Cost Estimation: What is Productivity? Estimation Factors, Approaches to Effort and Schedule Estimation, COCOMO II, Putnam Estimation Model, Algorithmic models, Cost Estimation.

UNIT-V

Requirements Management Tools: Benefits of using a requirements management tool, commercial requirements management tool, Rational Requisite pro, Caliber – RM, implementing requirements management automation.

Software Estimation Tools: Desirable features in software estimation tools, IFPUG, USC's COCOMO II, and SLIM (Software Life Cycle Management) Tools.

TEXT BOOKS:

1. Software Requirements by Karl E. Weigers,Microsoft Press.
2. Software Requirements and Estimation by *Rajesh Naik and Swapna Kishore*, Tata Mc Graw Hill.

REFERENCES:

1.Managing Software Requirements, Dean Leffingwell & Don Widrig, Pearson Education,2003.
2. Mastering the requirements process, second edition, Suzanne Robertson & James Robertson, Pearson Education, 2006.
3. Estimating Software Costs, Second edition, Capers Jones, Tata McGraw-Hill, 2007.
4. Practical Software Estimation, M.A. Parthasarathy, Pearson Education, 2007.
5. Measuring the software process, William A. Florac & Anita D. Carleton, Pearson Education,1999.

R19 | **M. Tech – SE**

SIDDARTHA INSTITUTE OF SCIENCE AND TECHNOLOGY:PUTTUR
(AUTONOMOUS)

I M. Tech. – I – Semester(SE)

L	T	P	C
3	0	0	3

Program Elective I
(19CS5103) PROFESSIONAL ASPECTS IN SOFTWARE ENGINEERING

Course Objectives:

- Develops a broad understanding of the discipline of the software engineering.
- Seeks to complement this with the detailed knowledge of the techniques for the analysis and design of complex software systems.
- To introduce ethical and professional issues and how are those relevant to software Engineers
- Provides a brief account of associated professional and legal issues.

Course Outcomes:

- Understood the issues affecting the organization, planning and control of software based systems development.
- Should able to complete the analysis and design of the intensive software systems.
- Understood the Professional and technical literature on Software Engineering

UNIT-I:
Intellectual Property rights Confidential Information, Copyright, Infringement of Copyright, Acts permitted in Relation to Copyright Works, Licensing and Assignment of Copyright, Moral Rights, Designs, Trademarks, The tort of passing off, Domain Names, Patents.

UNIT-II:
Software Licenses, Copyright, Contract, Patent, Free Software and Open Source Software, MIT License, BSD, License, GNU General Public License, GNU Lesser General Public License, Q Public License, Proprietary License, Sun Community License.

UNIT-III:
Software Contracts: Basics of Software Contracts, Extent of liability, Contract for the supply of custom-built software at a fixed price, other types of software service Contract, Liability for defective software.

UNIT-IV:

Software Crime Prevention Computing and criminal Activity, Reforms of Criminal Law, Categories of Misuse, Computer Fraud, Obtaining Unauthorized Access to Computer, Unauthorized Alteration or Destruction of Information, Denying Access to an Authorized user, Unauthorized Removal of Information Stored in a Computer

UNIT-V:

Data Protection Regulations, Data Protection and Privacy, The impact of the Internet, Factors Influencing the Regulation of Data Processing, Convergence of Data Protection Practice, Defamation and the protection of Reputation.

REFERENCES:

1. Andrew M. St. Laurent, "Open Source and Free Software Licensing", O'Reilly, Publications.
2. Frank Bott, et. al, "Professional Issues in Software Engineering", Taylor

SIDDARTHA INSTITUTE OF SCIENCE AND TECHNOLOGY:PUTTUR
(AUTONOMOUS)

I M. Tech. – I – Semester(SE)

L	T	P	C
3	0	0	3

Program Elective I
(19CS5104) PROTOCOL SOFTWARE ENGINEERING

Course Objectives:

- Able to acquire the generic software development model.
- Understand the various stages of software development cycle
- Maintain the software quality through software development with various protocol based environment.

Course Outcomes:

- Able to generate the test cases for various protocol based environment.
- Understood the different protocol specification languages.
- Able to perform protocol testing.

UNIT I

Network Reference Model: Layered Architecture, Network Services and Interfaces, Protocol Functions, OSI Model, TCP/IP Protocol Suite, Application Protocols.

Formal Specification: Formal Specification in the Software Process, Sub-system Interface Specification, Behavioral Specification. Protocol Specification: Components of Protocol to be Specified, Communication Service Specification, Protocol Entity Specification, Interface Specifications, Interactions, Multimedia Protocol Specifications, Internet Protocol Specifications.

UNIT II

Architectural Design: Architectural Design Decisions, System Organization, Modular Decomposition Styles, Control Styles, Reference Architectures. Distributed Systems Architectures: Multiprocessor Architectures, Client-server Architectures, Distributed Object Architectures, Inter-organizational Distributed Computing.

UNIT III

Formal Description Testing for Protocol Specification, Extended State Transition Language, Language for temporal Ordering Specification, Format and Protocol Languages.

SDL: A Protocol Specification Language: SDL, Examples of SDL Based Protocol Specifications, Other Protocol Specification Languages.
Protocol Verification/Validation: Protocol Verification, Verification of a Protocol Using Finite State Machines, Protocol Validation, Protocol Design Errors, Protocol Validation Approaches, SDL Based Protocol Verification, SDL Based Protocol Validation.

UNIT IV
Protocol Conformance Testing: Conformance Testing, Conformance Testing Methodology and Framework, Conformance Test Architectures, Test Sequence Generation Methods, Distributed Architecture by Local Methods, Conformance Testing with TTCN, Conformance Testing in Systems with Semi controllable Interfaces, Conformance Testing of RIP, Multimedia Applications Testing, SDL Based Tools for Conformance Testing, SDL Based Conformance Testing of MPLS.

UNIT V
Protocol Performance Testing: Performance Testing, SDL Based Performance Testing of TCP, SDL Based Performance Testing of OSPF, Interoperability Testing, SDL Based Interoperability Testing of CSMA/CD and CSMA/CA Protocol Using Bridge, Scalability Testing.
Protocol Synthesis: Protocol Synthesis, Interactive Synthesis Algorithm, Automatic Synthesis Algorithm, Automatic Synthesis of SDL from MSC, Protocol Resynthesize. Testing Models, PICS and PIX IT, Abstract Test Methods, Simulation Based Evaluation of Conformance Testing Methodologies. Examples include actual implementation like OSINET, based on ESTELLE tools and TTCU, PICS, PIX IT for OSINET.

TEXT BOOKS:

1. Communication Protocol Engineering, Pallapa Venkataram, Sunilkumar S. Manvi, PHI.
2. Protocol Specification for OSI*1 , Gregor V. Bochmann, University of Motreal, Montreal, Quebec, Canada.
3. ASN.1: Communication Between Heterogeneous Systems, Olivier Dubuisson, Morgan Kaufmann.

REFERENCES:

1.Tools for Protocols Driven by Formal Specifications, Harry Rudin. Network Protocols and Tools to help produce them*, Harry Rudin, IBM Research Division, Zurich Research Laboratory, 8803 Ruschlikon, Switzerland.

R19 **M. Tech – SE**

SIDDARTHA INSTITUTE OF SCIENCE AND TECHNOLOGY:PUTTUR
(AUTONOMOUS)

I M. Tech. – I – Semester(SE)

L	T	P	C
3	0	0	3

Program Elective II
(19CS5105) FORMAL METHODS OF SOFTWARE ENGINEERING

Course Objectives:

- Introduce students to advanced techniques and methods in software engineering that reflect the current state of the art.
- Students will be equipped with the knowledge of both formal methods and informal but precise and rigorous methods of software development.
- The course places more focus on the early phases in the software development cycle.

Course Outcomes:

- Describe the characteristics and tradeoffs between different formal and informal methods of software development;
- Specify software using a formal specification language;
- Develop high quality software requirement specifications using informal or semi-formal notation.
- Appreciate the use of formal and rigorous techniques in program refinement and verification.

UNIT I

Introduction: Formal methods, The CICS Experience, The Z notation, The importance of Proof, Abstraction.
Propositional Logic: Proportional logic, Conjunction, Disjunction, Implication, Equivalence, Negation, Tautologies and Contradictions.
Predicate Logic: Predicate calculus, Quantifiers and declarations, Substitution, Universal Introduction and elimination, Existential introduction and elimination, Satisfaction and validity.
Equality and Definite Description: Equality, The one-point rule, Uniqueness and quantity, definite description.

UNIT II

Sets: Membership and extension, Set comprehension, Power sets, Cartesian products, Union, intersection, and difference, Types.

Definitions: Declarations, Abbreviations, Generic abbreviations, Axiomatic definitions, Generic definitions, Sets and predicates.

Relations: Binary relations, Domain and range, Relational inverse, Relational composition, Closures.

Functions: Partial functions, Lambda notation, Functions on relations, Overriding, Properties of functions, Finite sets.

UNIT III

Sequences: Sequence notation, A model for sequences, Functions on sequences, Structural induction, Bags.

Free Types: The natural numbers, Free type definitions, Proof by induction, Primitive recursion, Consistency.

Schemas: The schema, Schemas as types, Schemas as declarations, Schemas as predicates, Renaming, Generic schemas.

Schema Operators: Conjunction, Decoration, Disjunction, Negation, Quantification and hiding, Composition.

UNIT IV

Promotion: Factoring operations, Promotion, Free and constrained promotion.

Preconditions: The initialization theorem, Precondition investigation, Calculation and simplification, Structure and preconditions.

A File System: A Programming interface, Operations upon files, a more complete description, a file system, Formal analysis.

Data Refinement: Refinement, Relations and non-determinism, Data types and data refinement, Simulations, Relaxing and unwinding.

UNIT V

Data Refinement and Schemas: Relations and schema operations, Forwards simulation, backwards simulation.

Functional Refinement: Retrieve functions, Functional refinement, Calculating data refinements, Refining promotion.

Refinement Calculus: The specification statement, Assignment, Logical constants, Sequence composition, Conditional statements, Iteration.

Text Book:

1. Jim **Woodcock and Jim Davies, "Using Z:** Specification, Refinement, and Proof", Prentice Hall (ISBN 0-13-948472-8), 1996.

Reference Books:

1. Diller, Z An Introduction to Formal Methods (2nd ed.), Wiley, 1994.
2. J. M. Spivey, "The Z Notation: A Reference Manual", Second Edition, Prentice Hall, 1992.

R19 **M. Tech – SE**

SIDDARTHA INSTITUTE OF SCIENCE AND TECHNOLOGY:PUTTUR
(AUTONOMOUS)

I M. Tech. – I – Semester(SE)

L	T	P	C
3	0	0	3

Program Elective II
(19CS5106) SOFTWARE METRICS & REUSE

Course Objectives:

- To understand software metrics and measurement.
- To emphasize the use of product and quality metrics.
- To explain quality assurance and various tools used in quality management.
- To learn in detail about various quality assurance models.
- To understand the audit and assessment procedures to achieve quality.
- To understand the knowledge of reuse and reuse problems.
- To discuss COTS reuse

Course Outcomes:

After completing, one should able to know

- Able to lead and implement measurement plans for process and product assessments.
- Able to analyze data for project estimation, planning and quality control in software projects.
- Understood the reusable concepts that can be represented as patterns.

UNIT - I

Basics of measurement: Measurement in everyday life, measurement in software engineering, scope of software metrics, representational theory of measurement, measurement and models, measurement scales, meaningfulness in measurement, goal-based framework for software measurement, classifying software measures, determining what to measure, software measurement validation.

UNIT - II

Empirical investigation: types of investigation, planning and conducting investigations.
Software-metrics data collection and analysis: What is good data, how to define the data, how to collect the data, how to store and extract data, analyzing software-measurement data, frequency distributions, various statistical techniques.

Measuring internal product attributes: Measuring size, aspects of software size, length, functionality and complexity, measuring structure, types of structural measures, control-flow structure, and modularity and information flow attributes, data structures.

UNIT - III

Measuring external product attributes: Modeling software quality, measuring aspects of software quality.

Metrics for object-oriented systems: The intent of object-oriented metrics, distinguishing characteristics of object-oriented metrics, various object-oriented metric suites – LK suite, CK suite and MOOD metrics.

Metrics for component-based systems: The intent of component-based metrics, distinguishing characteristics of component-based metrics, various component-based metrics.

UNIT - IV

Introduction: Software Reuse and Software Engineering, Concepts and Terms, Software Reuse products, Software Reuse processes, Software Reuse paradigms. State of the Art and the Practice: Software Reuse Management, Software Reuse Techniques, Aspects of Software Reuse, Organizational Aspects, Technical Aspects and Economic Aspects.

Programming Paradigm and Reusability: Usability Attributes, Representation and Modeling Paradigms, Abstraction and Composition in development paradigm.

UNIT - V

Object-Oriented Domain Engineering: Abstraction and Parameterization Techniques, Composition Techniques in Object Orientation.

Application Engineering: Component Storage and Retrieval, Reusable Asset Integration. **Software Reuse Technologies:** Component Based Software Engineering, COTS based development, Software Reuse Metrics, Tools for Reusability.

Text books:

1. Norman E. Fenton and Shari Lawrence Pfleeger; Software Metrics – A Rigorous and Practical Approach, Thomson Asia Pte., Ltd, Singapore.
2. Stephen H. Kan; Metrics and Models in Software Quality Engineering, Addison Wesley, New York.
3. Reuse Based Software Engineering Techniques, Organization and Measurement by Hafedh Mili, Ali Mili, Sherif Yacoub and Edward Addy, John Wiley & Sons Inc

4. The Three Rs of Software Automation: Re-engineering, Repository, Reusability by Carma McClure, Prentice Hall New Jersey

References:

1. K. H. Möller and D. J. Paulish; Software Metrics - A Practitioner's Guide to Improved Product Development, Chapman and Hall, London.
2. Mark Lorenz and Jeff Kidd; Object-Oriented Software Metrics, Prentice Hall, New York.
3. McClure, Carma L. Software reuse techniques : adding reuse to the system development process / : Prentice Hall
4. Poulin, Jeffrey S. Measuring software reuse : principles, practices, and economic models / Jeffrey S. Poulin. Reading, Mass. : Addison-Wesley

R19 **M. Tech – SE**

SIDDARTHA INSTITUTE OF SCIENCE AND TECHNOLOGY:PUTTUR
(AUTONOMOUS)

I M. Tech. – I – Semester(SE)

L	T	P	C
3	0	0	3

Program Elective II
(19CS5107) SERVICE ORIENTED ARCHITECTURE

Course Objectives:

The course should enable the student

- Understand SOA and evolution of SOA.
- Understand web services and primitive, contemporary SOA.
- Understand various service layers.
- Understand service-oriented analysis and design based on guidelines.

Couse Outcomes:

By completing, one should be able to:

- Understand the concepts of Service Oriented Architecture along with the evolution of SOA.
- Understand primary concepts of SOA .
- Know the integration of SOA technological points with Web Services.
- Implementation of SOA in development cycle of Web Services.
- Integrate SOA technologies with Web Services paradigms.
- Can learn the reference model of Service Oriented base line backend design for any typical environment.

UNIT I

Introducing SOA: Fundamental SOA, Common Characteristics of Contemporary SOA, Common Tangible Benefits of SOA, and Common Pitfalls of Adopting SOA

The Evolution of SOA: An SOA Timeline, the Continuing Evolution of SOA, The Roots of SOA.

UNIT II

Web Services and Primitive SOA: The Web Services Frame Work, Services, Service Descriptions, Messaging.

Web Services and Contemporary SOA (Part I-Activity management and Composition): Message Exchange Patterns, Service Activity, Coordination, Atomic Transactions, Orchestration, and Choreography.
Web Services and Contemporary SOA (Part-II-Advanced Messaging, Metadata and Security): Addressing, Reliable Messaging, Correlation, Policies, Metadata exchange, Security.

UNIT III
Principles of Service-Orientation: Service–Orientation and the Enterprise, Anatomy of SOA, Common Principles of Service–Orientation, Interrelation between Principles of Service-Orientation, Service Orientation and Object Orientation, Native Web Services Support for Principles of Service-Orientation.

Service Layers: Service-Orientation and Contemporary SOA, Service Layer abstraction, Application Service Layer, Business Service Layer, Orchestration Service Layer, Agnostic Services, Service Layer Configuration Scenarios

UNIT IV
SOA Delivery Strategies: SOA Delivery Lifecycle Phases, The Top-Down Strategy, The Bottom-up Strategy, The Agile Strategy.

Service Oriented Analysis (Part I-Introduction): Introduction to Service Oriented Analysis, Benefits of a Business Centric SOA, Deriving Business Services.

Service Oriented Analysis (Part-II-Service Modeling): Service Modeling, Service Modeling Guidelines, Classifying Service Model Logic, Contrasting Service Modeling Approaches.

Service Oriented Design (Part I-Introduction): Introduction to Service-Oriented Design, WSDL Related XML Schema Language Basics, WSDL Language Basics, Service Interface Design Tools.

Service Oriented Design (Part II-SOA Composition Guidelines): SOA Composing Steps, Considerations for Choosing Service Layers, Considerations for Positioning Core SOA Standards, Considerations for Choosing SOA Extensions.

UNIT V
Service Oriented Design (Part III- Service Design): Service Design Overview, Entity-Centric Business Service Design, Application Service Design, Task-Centric Business Service Design, Service Design Guidelines.

Service Oriented Design (Part IV-Business Process Design): WS-BPEL Language Basics, WS-Coordination Overview, Service Oriented Business Process Design.

TEXT BOOKS:

1. Service-Oriented Architecture-Concepts, Technology, and Design, Thomas Erl, Pearson Education.
2. Understanding SOA with Web Services, Eric Newcomer, Greg Lomow, Pearson Education.

REFERENCE BOOKS:

1. The Definitive guide to SOA, Jeff Davies & others, Apress, Dreamtech.
2. Java SOA Cook book, E.Hewitt, SPD.
3. SOA in Practice, N.M.Josuttis, SPD.
4. Applied SOA, M.Rosen and others, Wiley India pvt. Ltd.
5. Java Web Services Architecture, J.Mc Govern, and others, Morgan Kaufmann Publishers, Elsevier.
6. SOA for Enterprise Applications, Shankar.K, Wiley India Edition.
7. SOA-Based Enterprise Integration, W.Roshen, TMH.
8. SOA Security, K.Rama Rao, C.Prasad, dreamtech press

R19 **M. Tech – SE**

SIDDARTHA INSTITUTE OF SCIENCE AND TECHNOLOGY:PUTTUR
(AUTONOMOUS)

I M. Tech. – I – Semester(SE)

L	T	P	C
2	0	0	2

(19HS0823)RESEARCH METHODOLOGY AND IPR

COURSE OBJECTIVES:

1. Understand some basic concepts of research and its methodologies.
2. Identify and discuss appropriate research topics, select appropriate research design, and implement a research project.
3. Understand the method of research writing and presenting research report and proposal
4. Provide an understanding on the importance of intellectual property rights
5. Understand the intricacies of grant of patent, patentability, licensing and revocation at national and international level.

COURSE OUTCOMES:

After the completion of the course, student would be able to:

1. Explain the key concepts and issues in research and basic framework of research process.
2. Formulate appropriate research problem and implement suitable research design for the research problem.
3. Identify various sources of information for literature review and data collection.
4. Develop an understanding of ethics in conducting applied research and make use of components of scholarly writing in report preparation.
5. Identify different types of Intellectual Properties (IPs), the right of ownership, scope of protection as well as the ways to create and to extract value from IP.
6. Recognize the crucial role of IP in organizations of different industrial sectors for the purposes of product and technology development.

UNIT I:

Research Methodology: Meaning, Objective and importance of research - Types of research - steps involved in research - Motivation in Research, Types of Research - Significance of Research - Research Methods versus Methodology - Importance of Knowing How Research is done - Research Process - Criteria of Good Research defining research problem - Errors in selecting a research problem

UNIT II:

Research Design and Data Collection: Research design - Different Research Designs - Effective literature studies - Classification of Data - Methods of Data Collection – Sampling - Sampling techniques, procedure and methods - Ethical considerations in research - Responsibility of ethics in research

UNIT III:

Research Report Writing: Effective technical writing, how to write report, Paper Developing a Research Proposal, Format of research proposal, a presentation and assessment by a review committee

UNIT IV:

Nature of Intellectual Property: Patents, Designs, Trade and Copyright - Process of Patenting and Development: technological research, innovation, patenting, development - International Scenario: International cooperation on Intellectual Property- Procedure for grants of patents - Patenting under PCT

UNIT V:

Patent Rights: Scope of Patent Rights - Licensing and transfer of technology - Patent information and databases - Geographical Indications - New Developments in IPR: Administration of Patent System - New developments in IPR: IPR of Biological Systems, Computer Software etc - Traditional knowledge - Case Studies - IPR and IITs

TEXT BOOKS:

1. Stuart Melville and Wayne Goddard, "Research methodology: an introduction for science & engineering students".
2. Wayne Goddard and Stuart Melville, "Research Methodology: An Introduction"

REFERENCES:

1. Ranjit Kumar, 2nd Edition, "Research Methodology: A Step by Step Guide for beginners" Halbert, "Resisting Intellectual Property", Taylor & Francis Ltd , 2007.
2. Mayall, "Industrial Design", McGraw Hill, 1992. Niebel , "Product Design", McGraw Hill, 1974.
3. Asimov, "Introduction to Design", Prentice Hall, 1962.
4. Robert P. Merges, Peter S. Menell, Mark A. Lemley, "Intellectual Property in New Technological Age", 2016.
5. T. Ramappa, "Intellectual Property Rights Under WTO", S. Chand, 2008

SIDDARTHA INSTITUTE OF SCIENCE AND TECHNOLOGY: PUTTUR
(AUTONOMOUS)

I M. Tech. – I – Semester(SE)

L	T	P	C
0	0	4	2

(19CS5108) OBJECT ORIENTED SOFTWARE ENGINEERING LAB

1. Identifying Requirements from Problem Statements

   Requirements | Characteristics of Requirements | Categorization of Requirements | Functional Requirements | Identifying Functional Requirements | Preparing Software Requirements Specifications

2. Estimation of Project Metrics

   Project Estimation Techniques | COCOMO | Basic COCOMO Model | Intermediate COCOMO Model | Complete COCOMO Model | Advantages of COCOMO | Drawbacks of COCOMO | Halstead's Complexity Metrics

3. Modeling UML Use Case Diagrams and Capturing Use Case Scenarios

   Use case diagrams | Actor | Use Case | Subject | Graphical Representation | Association between Actors and Use Cases | Use Case Relationships | Include Relationship | Extend Relationship | Generalization Relationship | Identifying Actors | Identifying Use cases | Guidelines for drawing Use Case diagrams

4. E-R Modeling from the Problem Statements

   Entity Relationship Model | Entity Set and Relationship Set | Attributes of Entity | Keys | Weak Entity | Entity Generalization and Specialization | Mapping Cardinalities | ER Diagram | Graphical Notations for ER Diagram | Importance of ER modeling

5. Identifying Domain Classes from the Problem Statements

   Domain Class | Traditional Techniques for Identification of Classes | Grammatical Approach Using Nouns | Advantages | Disadvantages | Using Generalization | Using Subclasses | Steps to Identify Domain Classes from Problem Statement | Advanced Concepts.

6. Statechart and Activity Modeling

Statechart Diagrams | Building Blocks of a Statechart Diagram | State | Transition | Action | Guidelines for drawing Statechart Diagrams | Activity Diagrams | Components of an Activity Diagram | Activity | Flow | Decision | Merge | Fork | Join | Note | Partition | A Simple Example | Guidelines for drawing an Activity Diagram

7. Modeling UML Class Diagrams and Sequence Diagrams

   Structural and Behavioral Aspects | Class diagram | Class | Relationships | Sequence diagram | Elements in sequence diagram | Object | Life-line bar | Messages

8. Modeling Data Flow Diagrams

   Data Flow Diagram | Graphical notations for Data Flow Diagram | Symbols used in DFD | Context diagram and leveling DFD.

9. Estimation of Test Coverage Metrics and Structural Complexity

   Control Flow Graph | Terminologies | McCabe's Cyclomatic Complexity | Computing Cyclomatic Complexity | Optimum Value of Cyclomatic Complexity | Merits | Demerits

10. Designing Test Suites

    Software Testing | Standards for Software Test Documentation | Testing Frameworks | Need for Software Testing | Test Cases and Test Suite | Types of Software Testing | Unit Testing | Integration Testing | System Testing | Example | Some Remarks.

REFERENCES

1. Bernd Bruegge & Allen H. Dutoit, “Object-Oriented Software Engineering”, 2009.
2. Ivar Jacobson, “Object-Oriented Software Engineering”, Pearson Education, 2009.
3 Stephen R. Schach, “Object-Oriented Classical Software Engineering”, Mc Graw Hill, 2010.
4 Yogesh Singh, “Object-Oriented Software Engineering”, 2012

R19

M. Tech – SE

SIDDARTHA INSTITUTE OF SCIENCE AND TECHNOLOGY:PUTTUR
(AUTONOMOUS)

I M. Tech. – I – Semester(SE)

L	T	P	C
0	0	4	2

(19CS5109) SOFTWARE REQUIREMENTS & ESTIMATION LAB

1 Draw the Work Breakdown Structure for the system to be automated

2 Schedule all the activities and sub-activities Using the PERT/CPM charts

3 Define use cases and represent them in use-case document for all the stakeholders of the system to be automated

4 Identify and analyze all the possible risks and its risk mitigation plan for the system to be automated

5 Define Complete Project plan for the system to be automated using Microsoft Project Tool

6 Define the Features, Vision, Business objectives, Business rules and stakeholders in the vision document

7 Define the functional and non-functional requirements of the system to be automated by using use cases and document them in SRS document

8 Define the following traceability matrices :

   1. Usecase vs Features
   2. Functional requirements Vs Usecases

9 Estimate the effort using the following the methods for the system to be automated:

   1. Function point metric
   2. Usecase point metric

10 Develop a tool which can be used for quantification of all the non-functional requirements

- For developing software project plan Microsoft Project or its equivalents may be used
- For developing SRS document Rational Requisite Pro Tool or its equivalents may be used

R19

M. Tech – SE

SIDDARTHA INSTITUTE OF SCIENCE AND TECHNOLOGY:PUTTUR
(AUTONOMOUS)

I M. Tech. – I – Semester(SE)

L	T	P	C
2	0	0	0

AUDIT-I
(19HS0818) ENGLISH FOR RESEARCH PAPER WRITING

UNIT I

Planning and Preparation, Word Order, Breaking up long sentences, Structuring Paragraphs and Sentences, Being Concise and Removing Redundancy, Avoiding Ambiguity and Vagueness

UNIT II

Clarifying Who Did What, Highlighting Your Findings, Hedging and Criticizing, Paraphrasing and Plagiarism, Sections of a Paper, Abstracts. Introduction

UNIT III

Review of the Literature, Methods, Results, Discussion, Conclusions, the Final Check

UNIT IV

key skills are needed when writing a Title, key skills are needed when writing an Abstract, key skills are needed when writing an Introduction, skills needed when writing a Review of the Literature,

UNIT V

skills are needed when writing the Methods, skills needed when writing the Results, skills are needed when writing the Discussion, skills are needed when writing the Conclusions useful phrases, how to ensure paper is as good as it could possibly be the first- time submission

REFERENCES

1. Goldbort R (2006) Writing for Science, Yale University Press (available on Google Books)
2. Day R (2006) How to Write and Publish a Scientific Paper, Cambridge University Press
3. Highman N (1998), Handbook of Writing for the Mathematical Sciences, SIAM. Highman's book.
4. Adrian Wall work , English for Writing Research Papers, Springer New York Dordrecht Heidelberg London, 2011

SIDDARTHA INSTITUTE OF SCIENCE AND TECHNOLOGY:PUTTUR
(AUTONOMOUS)

I M.Tech. – I Sem.(SE) **L T C**
3 0 0

AUDIT-I
(19CE1029) DISASTER MANAGEMENT

Course *Objective:*

The objectives of this subject are to give the basic knowledge of Environmental Hazards and disasters. The syllabus includes the basics of Endogenous and Exogenous hazards and gives a suitable picture on the different types of hazard and disaster mitigation methods.

Course Outcomes:

On completion of the course the students will have knowledge on
1. Types of disasters and their effects on environment
2. Causes of disasters
3. Disaster management through engineering applications

UNIT-I

Environmental Hazards & Disasters: Meaning of Environmental hazards, Environmental Disasters and Environmental stress. Concept of Environmental Hazards, Environmental stress & Environmental Disasters. Different approaches & relation with human Ecology- Landscape Approach - Ecosystem Approach - Perception approach - Human ecology& its application in geographical researches.

UNIT –II

Types of Environmental hazards & Disasters: Natural hazards and Disasters – Man induced hazards & Disasters - Natural Hazards- Planetary Hazards/ Disasters – Extra Planetary Hazards/ disasters - Planetary Hazards- Endogenous Hazards – Exogenous Hazards

UNIT–III

Endogenous Hazards - Volcanic Eruption – Earthquakes – Landslides – Volcanic Hazards/ Disasters - Causes and distribution of Volcanoes - Hazardous effects of volcanic eruptions - Environmental impacts of volcanic eruptions – Earthquake Hazards/ disasters - Causes of Earthquakes - Distribution of earthquakes – Hazardous effects of - earthquakes - Earthquake Hazards in India - - Human adjustment, perception & mitigation of earthquake.

UNIT –IV

Exogenous hazards/ disasters - Infrequent events- Cumulative atmospheric hazards/disasters infrequent events: Cyclones – Lightning – Hailstorms Cyclones: Tropicalcyclones & Local storms - Destruction by tropical cyclones & local storms (causes, distribution human adjustment, perception & mitigation) Cumulative atmospherichazards/ disasters: - Floods- Droughts- Cold waves- Heat waves. Floods:- Causes of floods- Flood hazards India- Flood

control measures (Human adjustment, perception &mitigation).Droughts:- Impacts of droughts- Drought hazards in India- Drought control measures- Extra Planetary Hazards/ Disasters- Man induced Hazards /Disasters-Physical hazards/ Disasters-Soil Erosion Soil Erosion:-- Mechanics & forms of Soil Erosion- Factors & causes of Soil Erosion- Conservation measures of Soil Erosion. Chemical hazards/ disasters: Release of toxic chemicals, nuclear explosion- Sedimentation processes. Sedimentation processes: -Global Sedimentation problems- Regional Sedimentation problems- Sedimentation &Environmental problems- Corrective measures of Eros ion & Sedimentation. Biologicalhazards/ disasters: - Population Explosion.

UNIT –V

Emerging approaches in Disaster Management- Three Stages

1. Pre- disaster stage (preparedness)
2. Emergency Stage
3. Post Disaster stage-Rehabilitation

Text books:

1. Disaster Management by Rajib Shah, Universities Press, India, 2003
2. Disaster Science and Management by Tushar Bhattacharya, TMHPublications.
3. Disaster Mitigation: Experiences and Reflections by PardeepSahni
4. Natural Hazards & Disasters by Donald Hyndman & David Hyndman –Cengage Learning

References:

1. The Environment as Hazards by Kates, B.I & White, G.F, Oxford Publishers, NewYork, 1978
2. Disaster Management by R.B. Singh (Ed), Rawat Publication, New Delhi, 2000
3. Disaster Management by H.K. Gupta (Ed), Universiters Press, India, 2003
4. Space Technology for Disaster Mitigation in India (INCED) by R.B. Singh, University of Tokyo, 1994.

SIDDARTHA INSTITUTE OF SCIENCE AND TECHNOLOGY:PUTTUR
(AUTONOMOUS)

I M.Tech- I Sem (SE)

L	P	C
2	0	0

AUDIT-I
(19HS0825) SANSKRIT FOR TECHNICAL KNOWLEDGE

Course Objectives:

- To get a working knowledge in illustrious Sanskrit, the scientific language in the world.
- Learning of Sanskrit to improve brain functioning.
- Learning of Sanskrit to develop the logic in mathematics, science & other subjects Enhancing the memory power.
- The engineering scholars equipped with Sanskrit will be able to explore the huge knowledge from ancient literature.

Course Outcomes:

Students will be able to

- Understanding basic Sanskrit language
- Ancient Sanskrit literature about science & technology can be understood
- Being a logical language will help to develop logic in students

UNIT-I

Alphabets in Sanskrit, Past/Present/Future Tenses, Simple Sentences.

UNIT-II

Order, Introduction of roots, Technical information about Sanskrit Literature.

UNIT-III

Technical concepts of Engineering-Electrical, Mechanical, Architecture, Mathematics

Text Books:

1. “Abhyaspustakam” – Dr.Vishwas, Samskrita-Bharti Publication, New Delhi
2. “Teach Yourself Sanskrit” PrathamaDeeksha-VempatiKutumbshastri, Rashtriya Sanskrit Sansthanam, New Delhi Publication
3. “India’s Glorious Scientific Tradition” Suresh Soni, Ocean books (P) Ltd., New Delhi.

SIDDARTHA INSTITUTE OF SCIENCE AND TECHNOLOGY::PUTTUR
(AUTONOMOUS)

I M.Tech- I Sem (SE)

L	P	C
2	0	0

AUDIT-I
(19HS0826) VALUE EDUCATION

Course Objectives:
Students will be able to

- Understand value of education and self- development
- Imbibe good values in students
- Let the should know about the importance of character

Course outcomes:

- Students will be able to:
- Knowledge of self-development.
- Learn the importance of Human values.

UNIT-I

Values and self-development – Social values and individual attitudes, Work ethics and Indian vision of humanism, Moral and non-moral valuation, Standards and principles, Value judgements.

UNIT-II

Importance of cultivation of values; Sense of duty, Devotion, Self-reliance; Confidence, Concentration. Truthfulness, Cleanliness, Honesty, Humanity, Power of faith, National Unity. Patriotism. Love for nature and Discipline.

UNIT-III

Personality and Behavior Development - Soul and Scientific attitude, Positive Thinking, Integrity and discipline, Punctuality, Love and Kindness, Avoid fault Thinking, Free from anger, Dignity of labour, Universal brotherhood and religious tolerance, True friendship, Happiness Vs suffering, love for truth, Aware of self-destructive habits. Association and Cooperation, Doing best for saving nature.

UNIT-IV

Character and Competence –Holy books vs Blind faith, Self-management and Good health, Science of reincarnation, Equality, Nonviolence, Humility, Role of Women, All religions and same message, Mind your Mind, Self-control, Honesty, Studying effectively.

Text Books:

1. *Values and Ethics for organizations Theory and practice*, Chakroborty, S.K. Oxford UniversityPress, New Delhi, 2010.
2. *Value Education*, N. Venkataiah, APH Publishing Corporation, 1998.

Reference Books:

1. *Value Education and Quality Teaching*: The double Helix effect, 2010
2.*Values Education and lifelong learning*: Principles, Policies,and Programs, N Aspin, D Chapman, Springer Publication, 2012.

SIDDARTHA INSTITUTE OF SCIENCE AND TECHNOLOGY:PUTTUR
(AUTONOMOUS)

I M. Tech. – II– Semester(SE)

L	T	P	C
3	0	0	3

(19CS5110) ADVANCES IN SOFTWARE TESTING

Course objectives:

- Study the significance of testing
- Study the testing to be done at various levels
- Understand the procedure for designing test cases

Course Outcomes:

- Ability to systematically test the applications
- Ability to write test cases
- Ability to use testing tools effectively

UNIT I

Introduction – Review, Software testing strategies.

Control flow graph – basic blocks, flow graphs, paths, basic paths, path conditions and domains, Dominators and post-dominators; Program dependence graph – data dependence, control dependence, call graph,

Tests generation - Test selection Problem, equivalence partitioning, Equivalence class partitioning, boundary value analysis and category partitioning method

UNIT II

Finite state machines (FSM) - properties of FSM, Conformance testing, test generation, test optimization, Fault detection. **Combinatorial designs** – combinatorial test design process. **Pairwise design**: Binary factors and multi-valued factors. **Orthogonal arrays** and multi-level orthogonal arrays

UNIT III

Test Adequacy: Basics, measurement of test adequacy, infeasibility and test adequacy. Adequacy criteria based control – statement, block, conditions and decisions coverage techniques. Basics of Junit tool for Java.

Software Metrics: Fundamentals – **Product and Quality Metrics -** Measurement of internet product attributes- size and structure-external product attributes-measurement of quality-Software quality metrics-product quality-process quality

UNIT IV

Regression Testing

What is Regression Testing? Regression test process. Regression test selection techniques: Test all, Random selection, modification traversing tests, using execution trace. Regression Testing Tools

UNIT V

Non-functional testing

Load testing, performance testing, GUI testing, Security testing techniques and tools.

Automation: Case studies functional test automation using Selenium.

TEXT BOOKS:

1. Aditya P Mathur, Foundations of software testing, 2nd edition, Pearson , 2013.
2. Boris Beizer, "Software Testing Techniques", 2nd Edition, Dream tech press, 2003.

REFERENCES:

1. M G Limaye, "Software Testing – Principles, Techniques and Tools", Tata McGraw Hill, 2009.
2. Edward Kit, "Software Testing in the Real World - Improving the Process", Pearson Education, 2004.
3. William E. Perry, "Effective methods for software testing", 2nd Edition, John Wiley, 2000.

R19 **M. Tech – SE**

SIDDARTHA INSTITUTE OF SCIENCE AND TECHNOLOGY:PUTTUR
(AUTONOMOUS)

I M. Tech. – II – Semester(SE)

L	T	P	C
3	0	0	3

(19CS5111) SOFTWARE PATTERNS

Course Objectives:

- Introduction to the fundamentals of software architecture.
- To understand various architectural patterns of software systems.
- To understand design patterns and their underlying object oriented concepts.
- Software architecture and quality requirements of a software system
- Identifying the appropriate patterns for design problems.
- To understand design patterns and their underlying object oriented concepts.
- To understand implementation of design patterns and providing solutions to real world software design problems.
- To understand patterns with each other and understanding the consequences of combining patterns on the overall quality of a system.

Course Outcomes:

The student will be able to:

- Design and motivate software architecture for large scale software systems
- Recognize major software architectural patterns, design patterns, and frameworks
- Know the underlying object oriented principles of design patterns.
- Understand the context in which the pattern can be applied.
- Understand how the application of a pattern affects the system quality and its tradeoffs.

UNIT I

Envisioning Architecture - What is Software Architecture, Architectural patterns, reference models, reference architectures, architectural structures and views and the Architecture Business Cycle.

Creating an Architecture - Quality Attributes, Achieving qualities, designing the Architecture, Documenting software architectures, Reconstructing Software Architecture.

UNIT II

Introduction to Patterns - What is a Pattern? What makes a Pattern? Pattern Categories, Relationships between Patterns, Pattern Description, Patterns and Software Architecture.

Architectural Patterns: Layers, Pipes and Filters, Blackboard, Broker, Microkernel, MVC, PAC, Reflection.

UNIT III

What is Design Pattern, Organizing catalogs, Role in solving design problems, Selection and usage.

Creational Patterns - Abstract factory, builder, factory method, prototype, and singleton.

UNIT IV

Structural Patterns - Adapter, bridge, composite, decorator, façade, flyweight, Proxy, Decorator, façade, flyweight, Proxy.

UNIT V

Behavioral Patterns - Chain of responsibility, command, Interpreter, iterator, mediator, memento, observer, state, strategy, template method, and visitor.

Case Studies – Designing a Document Editor **-** Design issues of Lexi Editor in Deign Patterns, The World Wide Web - a case study in interoperability

TEXT BOOKS:

1. Software Architecture in Practice, second edition, Len Bass, Paul Clements & Rick Kazman, Pearson Education,2003.
2. Pattern-Oriented Software Architecture", A System of Patterns, Frank Buschmann Regine Meunier, Hans Rohnert, Peter Sommerlad and Michael Stal, WlLEY.
3. Design Patterns: Elements of Reusable Object-Oriented Software, Erich Gamma, Pearson Education.

REFERENCES

1. AntiPatterns: Refactoring Software, Architectures, and Projects in Crisis, by William J. Brown, Raphael C. Malveau, Hays W. "Skip" McCormick , Thomas J. Mowbray (Author) 1st Edition,
2. Java testing patterns, John Thomas etc, wiley.
3. Software architecture, David M. Dikel, David Kane and James R. Wilson, Prentice Hall PTR,2001
4. Head First Design patterns, Eric Freeman & Elisabeth Freeman, O'REILLY, 2007.
5. Design Patterns in Java, Steven John Metsker & William C. Wake, Pearson education, 2006

SIDDARTHA INSTITUTE OF SCIENCE AND TECHNOLOGY:PUTTUR
(AUTONOMOUS)

I M. Tech. – II – Semester(SE)

L	T	P	C
3	0	0	3

Program Elective III
(19CS5112) SOFTWARE QUALITY ASSURANCE

Course Objectives:.

The student should be able to:

- Understand quality assurance as a fundamental component of software
- Define the scope of quality assurance projects
- Estimate cost of a quality assurance project and manage budgets
- Prepare schedules for a quality assurance project
- Develop testing & quality assurance project staffing requirements
- Effectively manage a testing & quality assurance project

Course Outcomes:

- Critically evaluate different software development environments and contexts with respect to the application of appropriate standards and models,
- Critically evaluate leading edge approaches in software development and attendant quality assurance methodologies, presenting the research using Harvard referencing.
- Understand and apply key quality assurance techniques tailored for specific software development environments.

UNIT I

FUNDAMENTALS OF SOFTWARE QUALITY ASSURANCE: The Role of SQA – SQA Plan – SQA considerations – SQA people – Quality Management – Software Configuration Management

UNIT II

MANAGING SOFTWARE QUALITY: Managing Software Organizations – Managing Software Quality – Defect Prevention – Software Quality Assurance Management

UNIT III

SOFTWARE QUALITY ASSURANCE METRICS: Software Quality – Total Quality Management (TQM) – Quality Metrics – Software Quality Metrics Analysis

UNIT IV

SOFTWARE QUALITY PROGRAM: Software Quality Program Concepts – Establishment of a Software Quality Program – Software Quality Assurance Planning – An Overview – Purpose & Scope.

UNIT V

SOFTWARE QUALITY MODELS AND ASSURANCE STANDARDIZATION: Models, SPICE, Malcolm Baldrige Model-P-CMM - Software Standards–ISO 9000 Quality System Standards - Capability Maturity Model and the Role of SQA in Software envelopments Maturity – SEI CMM Level 5 – Comparison of ISO 9000 Model with SEI's CMM

TEXT BOOKS:

1. Mordechai Ben-Menachem / Garry S Marliss, "Software Quality", Vikas Publishing House, Pvt, Ltd., New Delhi.(UNIT III to V)
2. Watts S Humphrey, " Managing the Software Process", Pearson Education Inc.(UNIT I and II)

REFERENCES:

1. Gordon G Schulmeyer, "Handbook of Software Quality Assurance", Third Edition, Artech House Publishers 2007
2. Nina S Godbole, "Software Quality Assurance: Principles and Practice", Alpha Science International, Ltd, 2004.

R19 | **M. Tech – SE**

SIDDARTHA INSTITUTE OF SCIENCE AND TECHNOLOGY:PUTTUR
(AUTONOMOUS)

I M. Tech. – II – Semester(SE)

L	T	P	C
3	0	0	3

Program Elective III
(19CS5113) SOFTWARE RELIABILITY

Course Objectives:

- To discuss the problems of reliability specification and measurement
- To introduce reliability metrics and to discuss their use in reliability specification
- To show how reliability predications may be made from statistical test results.

Course Outcome;

- Master attributes and assessment of quality, reliability and security of software
- To describe the statistical testing process

UNIT I

Introduction: The Need for Reliable Software, Software Reliability Engineering Concepts, Basic definitions, Software practitioner's biggest problem, software reliability engineering approach, software reliability engineering process, defining the product.

The Operational Profile: Reliability concepts, software reliability and hardware reliability, developing operational profiles, applying operational profiles, learning operations and run concepts.

UNIT II

Software Reliability Concepts: Defining failure for the product, common measure for all associated systems, setting system failure intensity objectives, determining develop software failure intensity objectives, software reliability strategies, failures, faults and errors, availability, system and component reliabilities and failure intensities, predicting basic failure intensity.

UNIT III:

Software Reliability Modeling Survey: Introduction, Historical Perspective and Implementation, Exponential Failure Time Class of Models, Weibull and Gamma Failure Time Class of Models, Infinite Failure Category Models, Bayesian Models, Model Relationship, Software Reliability Prediction in Early Phases of the Life Cycle.

UNIT IV:

Software Metrics for Reliability Assessment: Introduction, Static Program Complexity, Dynamic Program Complexity, Software Complexity and Software Quality, Software

Reliability Modeling.

Software Testing and Reliability: Introduction, Overview of Software Testing, Operational profiles, Time/Structure Based Software Reliability Estimation.

UNIT V

Best Practice of SRE: Benefits and approaches of SRE, SRE during requirements phase, and SRE during implementation phase, SRE during Maintenance phase.

Neural Networks for Software Reliability: Introduction, Neural Networks, Neural Networks for software reliability, software reliability growth modeling.

Text Books

1. Handbook of Software Reliability Engineering Edited by Michael R. Lyu, published by IEEE Computer Society Press and McGraw-Hill Book Company.
2. Software Reliability Engineering John D. Musa, second edition Tata McGraw-Hill.

Reference Books

1. Practical Reliability Engineering, Patric D. T. O connor 4th Edition, John Wesley & Sons, 2003.
2. Fault tolerance principles and Practice, Anderson and PA Lee, PHI, 1981.
3. Fault tolerant computing-Theory and Techniques, Pradhan D K (Ed.): Vol 1 and Vol 2, Prentice hall, 1986.
4. Reliability Engineering E. Balagurusamy, Tata McGrawHill, 1994.

SIDDARTHA INSTITUTE OF SCIENCE AND TECHNOLOGY:PUTTUR
(AUTONOMOUS)

I M. Tech. – II – Semester(SE)

L	T	P	C
3	0	0	3

Program Elective III
(19CS5114) MODEL DRIVEN SOFTWARE ENGINEERING

Couse Objectives

- Develop enabling technologies for supporting model driven engineering approaches to software development
- Develop improved techniques and tool support for using executable specifications and model-based testing to better capture, manage and test software against its requirements
- Better integrate social networking tools and techniques into the software development process to improve the efficiency of collaborative and community development of software
- Better support "early phase" decision making by providing tools and techniques to assess nonfunctional requirement adherence at early stages in the software development process.

Course Outcomes

- Explain the role and importance of modelling in software development
- Make and defend decisions regarding the appropriate use of modelling throughout the software development life-cycle
- Demonstrate the practical application of several general purpose modeling languages
- Design and demonstrate the practical application of domain specific modeling languages
- Integrate a set of models to form effective software specifications
- Describe concepts involved in the verification and translation of specifications
- Demonstrate the translation of specifications to form executable software

UNIT I

MDSD Basic Terminology: Goals of MDSD, MDSD Approach, Overview of MDA concepts, Architecture-Centric MDSD, Common MDSD concepts and terminology, Model-Driven Architecture, Generative Programming, Software factories, Model-Integrated computing, Language-Oriented Programming, Domain specific modeling.

UNIT II

Metamodeling: What is Metamodeling?, Metalevels vs. Level of Abstraction, MOF and

UML, UML Meta models - Extending UML, Prototyping Requirement Management Meta-Model, UML profiles, Metamodeling and OCL, Examples ,Tool-supported Model validation, Metamodeling and Behavior, Pitfalls in Metamodeling, MDSD classification.

UNIT III

Model Transformation with QVT

History, M2M language requirements, Overall Architecture, An Example Transformation, The OMG standardization Process and Tool Availability, Assessment.

MDSD Tools: Roles, Architecture, Selection Criteria, and Pointers: Role of Tools in the Development Process, Tool Architecture and selection criteria, pointers.

The MDA Standard: Goals, Core concepts

UNIT IV

MDSD Process Building Blocks and Best Practices: Introduction, Separation between Application and domain Architecture Development, Two track Iterative Development, Target Architecture Development Process, Product-line Engineering.

Testing: Test Types, Tests in Model-driven Application Development, Testing the Domain Architecture

Versioning: What is Versioned? Projects and Dependencies, The structure of Application Projects, Version management and Build Process for mixed files, modeling in a team and versioning of partial models

UNIT V

Quality: Quality in Model Driven Engineering

Case study: Embedded Component Infrastructures - Overview, Product-Line Engineering, Modeling, Implementation of Components, Generator Adaptation, Code Generation

TEXT BOOKS

1. Model-Drievn Software Development-Technology, Engineering, Management by Thomos Stahl, Markus Volter, jul 2006, John Wiley & Sons.
2. Model-Driven Software Development: Integrating Quality Assurance by Jorg Rech, Christian Bunse,2008, Information Science Publishing.

REFERENCES

1. Model-Driven Software Development by Sami Beydeda Matthias Book , Volker Gruhn, Springer.
2. Model Driven Systems Development with Rational Products By Brian Nolan, Barclay Brown, Dr. Laurent Balmelli, Et Al Tim Bohn, 2008,IBM.
3. Model Driven Development with Executable UML by Dragan Milicev, 2009,Wilei India pvt Ltd.
4. Model Driven Software Development by Kevin Lano, Apr 2009, Ci Business Press.

R19 | **M. Tech – SE**

SIDDARTHA INSTITUTE OF SCIENCE AND TECHNOLOGY:PUTTUR
(AUTONOMOUS)

I M. Tech. – II – Semester(SE)

L	T	P	C
3	0	0	3

Program Elective IV
(19CS5115) SOFTWARE REENGINEERING

Course Objectives

- To explain why software re-engineering is a cost-effective option for system evolution
- To describe the activities involved in the software re-engineering process
- To distinguish between software and data re-engineering and to explain the problems of data re-engineering.

Course Outcomes

- Assess which parts should be reengineered first;
- Identify the risks and opportunities for a given re-engineering project;
- Extract coarse-grained and fine-grained design models;
- select the most appropriate migration strategy;
- Solve the typical problems of an object-oriented re-engineering project.

UNIT I

Software, Software Evolution and Maintenance: Software, Legacy software, well designed software, Software evolution challenges, Lehman's laws, Software deterioration curve.

Software maintenance: Software change, Types of change encountered during the support phase, Maintenance costs, why is software maintenance expensive? Factors affecting maintenance, Maintenance process, Change and maintenance prediction.

Software Quality Factors, Quality and Maintainability Metrics: Internal and external attributes, McCall's quality factors, ISO 9126 quality factors, Need and importance of quality and maintainability metrics, Metric for software correctness (Defects/KLOC), Metric for software integrity, Software reliability (MTBF), Metrics for maintainability (Mean-time-to-change (MTTC), Spoilage metric, Software maturity index, McCabe and Halstead metrics).

UNIT II

Design maintainability: Cohesion, Coupling, Understandability and Adaptability.

Legacy software structure, Software reengineering process model: Software change strategies include: Software maintenance, Architectural transformation, Software reengineering. Legacy software structure and distribution: Ideal structure, Real structure, Layered distribution model, Legacy software distribution, Architectural problems.

Business process reengineering: Business processes, A BPR Model, Software reengineering and its importance, Goals of reengineering, A software reengineering process model, Software reengineering activities.

UNIT III

Design Extraction: Reverse Engineering: Goals of reverse engineering, Why design extraction is needed?, Reverse engineering process, Reverse engineering to understand processing, Code duplication detection, Reverse engineering to understand data, Reverse engineering user interfaces, Design extraction with UML, Heuristics to extract the design, Tools for reverse engineering.

Restructuring (In Traditional context): Code restructuring: Characteristics of unstructured code, Characteristics of structured code, Spaghetti logic, Structured control logic, Restructuring problems, Flow graph restructuring, Warnier's logical simplification techniques, Some basic code restructuring methods: Interchange, Transposition, Combination, Resolution, Substitution.

UNIT IV

Data restructuring (Data reengineering): Data reengineering process, Data problems, Approaches: Data cleanup, Data extension, Data migration. Tools for restructuring.

Refactoring (Restructuring in object oriented context): What is refactoring?, Principles in refactoring: Why should you refactor?, When should you refactor?, Problems with refactoring, Refactoring and design, Refactoring and performance. Refactoring opportunities, Top ten of code bad smells, Different refactorings and their use, Refactoring tools.

UNIT V

Forward Engineering: What is forward engineering ? Goals of forward engineering, Forward engineering for client/server applications, Forward engineering for object oriented architectures, Forward engineering user interfaces, Tools for forward engineering.

Reengineering Metrics, Repositories, and Economics: Metrics in Reengineering: Why metrics in Reengineering?, Metrics as a reengineering tool, Which metrics to collect ?(Goal Question Metric (GQM) paradigm), Reengineering repositories: Why repositories?, Taxonomy (Functionality + Integration options), Issues, Reengineering economics.

TEXT BOOKS

1. Software Reengineering, Ed. Robert S. Arnold, IEEE Computer Society, 1993.
2. Software Evolution, Tom Mens, Serge Demeyer, Springer publication company, 2008.

REFERENCES

1. Software Engineering, Ian Sommerville, Addison-Wesley, 6th Edition.
2. Software Engineering, A Practitioner's Approach, Roger S. Pressman, 6th Edition.
3. Refactoring: Improving the Design of Existing Code, Martin Fowler, K.Beck, J.Brant.

R19 **M. Tech – SE**

SIDDARTHA INSTITUTE OF SCIENCE AND TECHNOLOGY:PUTTUR
(AUTONOMOUS)

I M. Tech. – II – Semester(SE)

L	T	P	C
3	0	0	3

Program Elective IV
(19CS5116) BIG DATA ANALYTICS

Course Objectives

- To learn to analyze the big data using intelligent techniques.
- To understand the various search methods and visualization techniques.
- To learn to various techniques for mining data stream.
- To understand the applications using Map Reduce Concepts.

Course Outcomes

On completion of this course the student will able to

- Analyze the big data analytics techniques for useful business application.
- Design efficient algorithms for mining the data from large volumes.
- Analyze the HADOOP and Map Reduce technologies associated with big data analytics.
- Explore on big data applications using Pig and Hive.

UNIT- I

Introduction to Big Data: Big Data and its Importance – Four V's of Big Data – Introduction to Big Data Analytics – Big Data Analytics applications - Challenges of Conventional System – Intelligent data analysis – Nature of Data – Analytic Processes and Tool – Analysis vs Reporting – Modern Data Analytic Tool – Statistical Concepts: Sampling Distributions – Re-Sampling – Statistical Inference – Prediction Error.

UNIT- II

Mining Data Streams: Introduction To Stream Concepts – Stream Data Model and Architecture - Stream Computing – Sampling Data in a Stream – Filtering Stream – Counting Distinct Elements in a Stream – Estimating Moments – Counting Oneness in a Window – Decaying Window – Real time Analytics Platform(RTAP) Applications – Case Studies – Real Time Sentiment Analysis, Stock Market Predictions.

UNIT – III

Hadoop: History of Hadoop- The Hadoop Distributed File System – Components of Hadoop – Analyzing the Data with Hadoop – Scaling Out – Hadoop Streaming – Design of HDFS-Java interfaces to HDFS Basics- Developing a Map Reduce Application – How Map Reduce

Works – Anatomy of a Map Reduce Job run – Failures – Job Scheduling – Shuffle and Sort – Task Execution – Map Reduce Types and Formats – Map Reduce Features.

UNIT – IV

Hadoop Environment: Setting up a Hadoop Cluster – Cluster specification – Cluster Setup and Installation –Hadoop Configuration – Security in Hadoop – Administering Hadoop – HDFS – Monitoring – Maintence – Hadoop Benchmarks – Hadoop in the Cloud

UNIT –V

Frameworks Applications on Big Data Using Pig and Hive – Data Processing operators in Pig – Hive Services – HiveQL – Querying Data in Hive – fundamentals of HBase and Zookeeper – IBM Info Sphere Big Insights and Streams. Visualization - Visual data analysis techniques, interaction techniques; Systems and applications.

TEXT BOOKS

1. Michael Berthold, David J.Hand, Intelligent Data Analysis, Spingers, 2007.
2. Tom White, Hadoop: The Definitive Guide Third Edition, O'reilly Media, 2012.
3. Chris Eaton, Dirk DeRoos, Tom Deutsch, George Lapis, Paul Zikopoulos, Uderstanding Big Data : Analytics for Enterprise Class Hadoop and Streaming Data, McGrawHill Publishing, 2012.
4. AnandRajaraman and Jeffrey David UIIman, Mining of Massive Datasets Cambridge University Press, 2012.

REFERENCES

1. Bill Franks, Taming the big Data tidal Wave: Finding Opportunities in Huge Data Streams with Advanced Analytics, John Wiley & sons, 2012.
2. Glenn J. Myatt, Making Sense of Data , John Wiley & Sons, 2007 Pete Warden, Big Data Glossary, O'Reilly, 2011.
3. Jiawei Han, MichelineKamber, Data Mining Concepts and Techniques, Second Edition.
4. Elsevier, Reprinted 2008. Da Ruan, Guoquing Chen, Etienne E.Kerre, Geert Wets, Intelligent Data Mining, Springer, 2007.
5. Paul Zikopoulos, Dirk deRoos, Krishnan Parasuraman, Thomas Deutsch, James Giles, David Corrigan, Harness the Power of Big Data the IBM Big Data Platform, Tata McGraw Hill Publications, 2012.
6. Michael Minelli (Author), Michele Chambers (Author), AmbigaDhirraj (Author), Big Data, BigSnalytics.

SIDDARTHA INSTITUTE OF SCIENCE AND TECHNOLOGY:PUTTUR
(AUTONOMOUS)

I M. Tech. – II – Semester(SE)

L	T	P	C
3	0	0	3

Program Elective IV
(19CS5117) SOFTWARE CONFIGURATION MANAGEMENT

Course Objectives:

- To learn the changing nature of software and need for change management.
- To study the different phases involved in software configuration management.
- To learn about the SCM plans, audits and reviews
- To study the various SCM tools and implementation techniques
- To study the SCM different scenarios and future directions

Course Outcomes:

- Identifying items that need to be controlled for changes.
- Systematically controlling changes to them.
- Establishing & maintaining integrity of these items and providing accurate status of items to relevant stakeholders (like developers, end users, and customers) throughout the Software Development Lifecycle.

UNIT I
OVERVIEW THE SOFTWARE CONFIGURATION MANAGEMENT

SCM: Concepts and definitions – SCM Plan – Software development life cycle models – SDLC Phases – Need and importance of Software configuration management –SCM: Basic concepts – Baselines – Check-in and Check-out- Versions and Variants –System Building – Releases.

UNIT II
DIFFERENT PHASES OF SOFTWARE CONFIGURATION MANAGEMENT

Different Phases Of Scm – SCM System design - SCM Plan preparation – SCM Team organization – SCM Infrastructure organization – SCM Team training – Project team training – Configuration identification – Configuration Control –Configuration status accounting – Configuration audits.

UNIT III
CONFIGURATION AUDITS AND MANAGEMENT PLANS

When, what and who of auditing - Functional Configuration audit – Physical Configuration audit – Auditing the SCM System – Role of SCM Team in configuration audits – SCM plan and the incremental approach – SCM Plan and SCM Tools – SCM Organization.

UNIT IV

SOFTWARE CONFIGURATION MANAGEMENT TOOLS AND IMPLEMENTATION

Advantages of SCM tools – Reasons for the increasing popularity of SCM tools – SCM Tools and SCM Functions – SCM tool selection – Role of Technology –Selection criteria – Tool implementation – SCM implementation plan

UNIT V

TRENDS IN SCM: FUTURE DIRECTIONS

SCM in different scenarios – SCM and project size – SCM in integrated development environments – SCM In distributed environments – SCM and CASE Tools - Trends in SCM - Hardware and Software Management – Better integration with IDE'S and CASE environments – Customization – Better decision making capabilities – Reduction in SCM Team size – Market snapshot.

REFERENCES

1. Jessica Keyes, Software Configuration Management, Auerbach Publications, 2008.
2. Alexis Leon, Software Configuration Management Handbook, Artech Print on Demand; 2 edition, 2009.
3. Robert Aiello and Leslie Sachs Configuration Management Best Practices: Practical Methods that work in Real World, , Addison-Wesley Professional; 1 edition, 2010.
4. Stephen P. Berczuk, Brad Appleton and Kyle Brown , "Software Configuration Management Patterns: Effective Teamwork and Practical Integration", Addison-Wesley , 2003.

R19

M. Tech – SE

SIDDARTHA INSTITUTE OF SCIENCE AND TECHNOLOGY:PUTTUR
(AUTONOMOUS)

I M. Tech. – II – Semester(SE)

L	T	P	C
0	0	4	2

(19CS5118) ADVANCES IN SOFTWARE TESTING LAB

Course Objectives:

- To learn to use the following (or Similar) automated testing tools to automate testing:
  - Win Runner/QTP for functional testing.
  - Load Runner for Load/Stress testing.
  - Test Director for test management.
  - JUnit, HTMLUnit, CPPUnit.
- To study state-of-art tools for software testing and Middleware technologies

Course Outcomes:

- Test the software applications using standard tools available in the market

Experiment/Program:

1. Write programs in C Language to demonstrate the working of the following constructs: i) do...while ii) while….do iii) if…else iv) switch v) for
2. A program written in C language for Matrix Multiplication fails. Introspect the causes for its failure and write down the possible reasons for its failure.
3. Consider ATM System and Study its system specifications and report the various bugs.
4. Write the test cases for banking application.
5. Create test plan document for Library Management System.
6. Create test cases for Railway Reservation.
7. Create test plan document for Online Shopping.

Working with Tool's:

Understand the Automation Testing Approach, Benefits, Workflow, Commands and Perform

Testing on one application using the following Tool's.

1. Win runner Tool for Testing.
2. Load runner Tool for Performance Testing.
3. Selenium Tool for Web Testing.
4. Bugzilla Tool for Bug Tracking.
5. Test Director Tool for Test Management.
6. Test Link Tool for Open Source Testing.

REFERENCES

1. M G Limaye, "Software Testing – Principles, Techniques and Tools", Tata McGraw Hill, 2009.
2. Edward Kit, "Software Testing in the Real World - Improving the Process", Pearson Education, 2004.
3. William E. Perry, "Effective methods for software testing", 2nd Edition, John Wiley, 2000.

SIDDARTHA INSTITUTE OF SCIENCE AND TECHNOLOGY:PUTTUR
(AUTONOMOUS)

I M. Tech. – II – Semester(SE)

L	T	P	C
0	0	4	2

(19CS5119) SOFTWARE PATTERNS LAB

Course Objectives:

- Construct UML diagrams for static view and dynamic view of the system.
- Generate creational patterns by applicable patterns for given context.
- Create refined model for given Scenario using structural patterns.
- Construct behavioral patterns for given applications.
- Construct architectural patterns for given applications.

Course Outcomes:

- Understand the Case studies and design the Model..
- Understand how design patterns solve design problems.
- Develop design solutions using creational patterns.
- Construct design solutions by using structural, behavioral and architectural patterns

Experiment/Program:

1. Identify the application where you can use single pattern and implement it.
2. Using UML design one of the architectural patterns.
3. Using UML design one of the creational patterns.
4. Using UML design one of the structural patterns.
5. Using UML design one of the behavioral patterns.
6. User gives a print command from a word document. Design to represent this chain of responsibility design pattern.
7. User gives a print command from a word document. Design to represent this Singleton design pattern.
8. Identify the application where you can use multiple creational patterns and implement it.
9. Identify the application where you can use multiple structural patterns and implement it.
10. Identify the application where you can use multiple behavioral patterns and implement it.
11. Identify the application where you can use architectural patterns and implement it.

REFERENCES

1. AntiPatterns: Refactoring Software, Architectures, and Projects in Crisis, by William J. Brown, Raphael C. Malveau, Hays W. "Skip" McCormick , Thomas J. Mowbray (Author) 1st Edition,
2. Java testing patterns, John Thomas etc, wiley.
3. Software architecture, David M. Dikel, David Kane and James R. Wilson, Prentice Hall PTR,2001
4. Head First Design patterns, Eric Freeman & Elisabeth Freeman, O'REILLY, 2007.
5. Design Patterns in Java, Steven John Metsker & William C. Wake, Pearson education, 2006

SIDDARTHA INSTITUTE OF SCIENCE AND TECHNOLOGY:PUTTUR
(AUTONOMOUS)

I M. Tech. – II Sem. (SE)

L	T	C
2	0	2

AUDIT-II
(19HS0829) CONSTITUTION OF INDIA

Course Objectives:

Students will be able to:

- Understand the premises informing the twin themes of liberty and freedom from a civil rights perspective.
- To address the growth of Indian opinion regarding modern Indian intellectuals' constitutional role and entitlement to civil and economic rights as well as the emergence of nationhood in the early years of Indian nationalism.
- To address the role of socialism in India after the commencement of the Bolshevik Revolution in 1917 and its impact on the initial drafting of the Indian Constitution.

Course Outcomes:

Students will be able to:

- Discuss the growth of the demand for civil rights in India for the bulk of Indians before the arrival of Gandhi in Indian politics.
- Discuss the intellectual origins of the framework of argument that informed the conceptualization of social reforms leading to revolution in India.
- Discuss the circumstances surrounding the foundation of the Congress Socialist Party [CSP] under the leadership of Jawaharlal Nehru and the eventual failure of the proposal of direct elections through adult suffrage in the Indian Constitution.
- Discuss the passage of the Hindu Code Bill of 1956.

UNIT –I
History of Making of the Indian Constitution:
History, Drafting Committee, (Composition & Working)
UNIT-II
Philosophy of the Indian Constitution:
Preamble, Salient Features
UNIT-III
Contours of Constitutional Rights & Duties: Fundamental Rights Right to Equality Right to Freedom.

Right against Exploitation, Right to Freedom of Religion, Cultural and Educational Rights Right to Constitutional Remedies Directive Principles of State Policy Fundamental Duties.

UNIT- IV

Organs of Governance: Parliament, Composition, Qualifications and Disqualifications Powers and Functions.

Executive, President, Governor, Council of Ministers, Judiciary, Appointment and Transfer of Judges, Qualifications, Powers and Functions.

UNIT-V

Local Administration:

District's Administration head: Role and Importance, Municipalities: Introduction, Mayor and role of Elected Representative, CEO of Municipal Corporation.

Pachayati raj: Introduction, PRI: Zila Pachayat, Elected officials and their roles, CEO Zila Pachayat: Position and role, Block level: Organizational Hierarchy (Different departments), Village level: Role of Elected and Appointed officials, Importance of grass root democracy

Election Commission: Role and Functioning, Chief Election Commissioner and Election Commissioners. State Election Commission: Role and Functioning, Institute and Bodies for the welfare of SC/ST/OBC and women

REFERENCES:

1. The Constitution of India, 1950 (Bare Act), Government Publication.
2. Dr. S. N. Busi, Dr. B. R. Ambedkar framing of Indian Constitution, 1st Edition, 2015.
3. M. P. Jain, Indian Constitution Law, 7th Edn., Lexis Nexis, 2014.
4. D.D. Basu, Introduction to the Constitution of India, Lexis Nexis, 2015.

SIDDARTHA INSTITUTE OF SCIENCE AND TECHNOLOGY:PUTTUR
(AUTONOMOUS)

I M. Tech- II Sem (SE)

L	T	C
2	0	0

AUDIT-II
(19HS0827) PEDAGOGY STUDIES

Course Objectives:

Students will be able to:

- Review existing evidence on the review topic to inform programme design and policy making undertaken by the DfID, other agencies and researchers.
- Identify critical evidence gaps to guide the development.

Course Outcomes:

Students will be able to understand:

- What pedagogical practices are being used by teachers in formal and informal classrooms in developing countries?
- What is the evidence on the effectiveness of these pedagogical practices, in what conditions, and with what population of learners?
- How can teacher education (curriculum and practicum) and the school curriculum and guidance materials best support effective pedagogy?

UNIT-I

Introduction and Methodology: Aims and rationale, Policy background, Conceptual framework and Terminology.Theories of learning, Curriculum, Teacher education.Conceptual framework, Research questions**.**Overview of methodology and Searching.

UNIT-II

Thematic overview: Pedagogical practices are being used by teachers in formal and informal classrooms in developing countries. Curriculum, Teacher education.

UNIT-III

Evidence on the effectiveness of pedagogical practices: Methodology for the in depth stage: quality assessment of included studies. How can teacher education (curriculum and practicum) and the school curriculum and guidance materials best support effective pedagogy? Theory of change. Strength and nature of the body of evidence for effective pedagogical practices. Pedagogic theory and pedagogical approaches. Teachers' attitudes and beliefs and Pedagogic strategies.

UNIT-IV

Professional development: alignment with classroom practices and follow-up support. Peer support: Support from the head teacher and the community. Curriculum and assessment Barriers to learning: limited resources and large class sizes

UNIT-V

Research gaps and future directions: Research design – Contexts- Pedagogy- Teacher education- Curriculum and assessment- Dissemination and research impact.

Text Books:

1. Ackers J, Hardman F (2001) Classroom interaction in Kenyan primary schools, Compare, 31 (2): 245-261.
2. Agrawal M (2004) Curricular reform in schools: The importance of evaluation, Journal of Curriculum Studies, 36 (3): 361-379.
3. Akyeampong K (2003) Teacher training in Ghana - does it count? Multi-site teacher education research project (MUSTER) country report 1. London: DFID.
4. Akyeampong K, Lussier K, Pryor J, Westbrook J (2013) Improving teaching and learning of basic maths and reading in Africa: Does teacher preparation count? International Journal Educational Development, 33 (3): 272–282.
5. Alexander RJ (2001) Culture and pedagogy: International comparisons in primary education. Oxford and Boston: Blackwell.
6. Chavan M (2003) Read India: A mass scale, rapid, 'learning to read' campaign.
7. www.pratham.org/images/resource%20working%20paper%202.pdf.

SIDDARTHA INSTITUTE OF SCIENCE AND TECHNOLOGY: PUTTUR
(AUTONOMOUS)

I M.Tech- II Sem (SE)

L	T	C
2	0	0

AUDIT-II
(19HS0828) STRESS MANAGEMENT BY YOGA

Course Objectives:

- To achieve overall health of body and mind
- To overcome stress

Course Outcomes:

Students will be able to:

- Develop healthy mind in a healthy body thus improving social health also
- Improve efficiency.

UNIT-I

Definitions of Eight parts of yoga(Ashtanga)

UNIT-II

Yam and Niyam. Do`s and Don'ts in life:
i) Ahinsa, satya, astheya, bramhacharya and aparigraha.
ii) Shaucha, santosh, tapa, swadhyay, ishwarpranidhan.

UNIT-III

Asan and Pranayam:
i) Various yog poses and their benefits for mind & body.
ii)Regularization of breathing techniques and its effects-Type of pranayam.

Text Books:

1. 'Yogic Asanas for Group Tarining-Part-I" :Janardan Swami Yogabhyasi Mandal, Nagpur Model Curriculum of Engineering & Technology PG Courses [Volume-I] [47].
2. "Rajayoga or conquering the Internal Nature" by Swami Vivekananda, AdvaitaAshrama (Publication Department) Kolkata.

SIDDARTHA INSTITUTE OF SCIENCE AND TECHNOLOGY:PUTTUR
(AUTONOMOUS)

I M. Tech. – II – Semester(SE)

L	T	P	C
2	0	0	0

AUDIT-II
(19HS0819) PERSONALITY DEVELOPMENT THROUGH LIFE ENLIGHTENMENT SKILLS

Course Objectives:

- To learn to achieve the highest goal happily.
- To become a person with stable mind, pleasing personality and determination.
- To awaken wisdom in students.

Course Outcomes:

Students will be able to:

- Study of Shrimad-Bhagwad-Geeta will help the student in developing his personality and achieve the highest goal in life.
- The person who has studied Geeta will lead the nation and mankind to peace and prosperity.
- Study of Neetishatakam will help in developing versatile personality of students

UNIT-I

Neetisatakam-Holistic development of personality

- Verses- 19,20,21,22 (wisdom)
- Verses- 29,31,32 (pride & heroism)
- Verses- 26,28,63,65 (virtue)
- Verses- 52,53,59 (dont's)
- Verses- 71,73,75,78 (do's)

UNIT-II

- Approach to day to day work and duties.
- Shrimad BhagwadGeeta : Chapter 2-Verses 41, 47,48,
- Chapter 3-Verses 13, 21, 27, 35, Chapter 6-Verses 5,13,17, 23, 35,
- Chapter 18-Verses 45, 46, 48.

UNIT-III

- Statements of basic knowledge.
- Shrimad BhagwadGeeta: Chapter2-Verses 56, 62, 68
- Chapter 12 -Verses 13, 14, 15, 16,17, 18
- Personality of Role model. Shrimad BhagwadGeeta: Chapter2-Verses 17, Chapter 3-Verses 36,37,42,
- Chapter 4-Verses 18, 38,39

- Chapter18 – Verses 37,38,63

Suggested Reading:

1. "Srimad Bhagavad Gita" by Swami SwarupanandaAdvaita Ashram (Publication Department), Kolkata.
2. Bhartrihari's Three Satakam (Niti-sringar-vairagya) by P.Gopinath, 4. Rashtriya Sanskrit Sansthanam, New Delhi.

SIDDARTHA INSTITUTE OF SCIENCE AND TECHNOLOGY:PUTTUR
(AUTONOMOUS)

II M. Tech. – I - Semester

L	T	P	C
3	0	0	3

Program Elective V
(19CS5121) SECURE SOFTWARE ENGINEERING

Course Objectives:

1. *To identify specific components of a software design that can be targeted for reuse.*
2. *To learn software testing plan and metrics for secure software engineering.*

Course Outcomes:

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

1. *Demonstrate knowledge of the distinction between critical and non-critical systems.*
2. *Manage a project including planning, scheduling and risk assessment/management.*
3. *Understand the proper contents of a software requirements document for secure software engineering.*
4. *Author a formal specification for secure software systems.*
5. *Describe the understanding of distributed system architectures and application architectures.*
6. *Differentiate the real-time and non-real time systems.*

UNIT I

Why Is Security a Software Issue?

Introduction, The problem, Software assurance and software security, Threats to software security, Sources of software insecurity, the benefits of detecting software security defects early, managing secure software development.

What Makes Software Secure?

Defining properties of secure software, How to influence the security properties of software, How to assert and specify desired security properties.

UNIT II

Requirements Engineering for Secure Software

Introduction, Misuse and Abuse Cases, The SQUARE process model: SQUARE sample outputs, Requirements elicitation, Requirements Prioritization.

Secure Software Architecture and Design

Introduction, Software security practices for architecture and design: Architectural risk analysis. Software security knowledge for architecture and design: Security principles, Security guidelines, and Attack patterns.

UNIT III

Considerations for Secure Coding and Testing

Introduction, Code analysis, Coding practices, Software security testing, Security testing considerations throughout the SDLC.

Security and Complexity: System Assembly Challenges

Introduction, Security failures, Functional and attacker perspectives for security analysis, System complexity drivers and security, Deep technical problem complexity.

UNIT IV

Governance, and Managing for More Secure Software

Introduction, Governance and security, Adopting an enterprise software security framework, How much security is enough?, Security and project management, maturity of practice.

UNIT V

Security Metrics

Defining security metrics, Diagnosing problems and measuring technical security, Analysis Techniques, Organize, aggregate, and analyze data to bring out key insights.

TEXT BOOKS

1. Julia H. Allen, Sean Barnum, Robert J. Ellison, Gary McGraw, Nancy R. Mead,Software Security Engineering: A Guide for Project Managers, Addison-Wesley, 1st edition, 2008.
2. Andrew Jaquith, Addison-Wesley, Security Metrics: Replacing Fear, Uncertainty, and Doubt, 1st edition , 2007.

REFERENCES

1. Haralambos Mouratidis, Paolo Giorgini, Integrating Security and Software Engineering: Advances and Future Vision, IGI Global, 2006.
2. Gary Software Security: Building Security In, McGraw , Addison-Wesley, 2006
3. Mark Dowd, John McDonald, Justin Schuh, The Art of Software Security Assessment: Identifying and Preventing Software Vulnerabilities, Addison-Wesley, 1st edition, 2006.
4. John Viega, Gary Building Secure Software: How to Avoid Security Problems the Right Way, McGraw, Addison-Wesley, 2001
5. M. Howard, D. LeBlanc, Writing Secure Code, Microsoft Press, 2nd Edition, 2003.
6. G. Hoglund, G. McGraw, Exploiting Software: How to break code, Addison Wesley, 2004.

R19 **M. Tech – SE**

SIDDARTHA INSTITUTE OF SCIENCE AND TECHNOLOGY:PUTTUR
(AUTONOMOUS)

II M. Tech. – I - Semester

L	T	P	C
3	0	0	3

Program Elective V
(19CS5122) AGILE METHODOLOGIES

Course Objective:

1. *To provide students with a theoretical as well as practical understanding of agile software development practices and how small teams can apply them to create high-quality software.*
2. *To provide a good understanding of software design and a set of software technologies and APIs.*
3. *To do a detailed examination and demonstration of Agile development and testing techniques.*
4. *To understand the benefits and pitfalls of working in an Agile team.*
5. *To understand Agile development and testing.*

Course Outcomes:

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

1. *Realize the importance of interacting with business stakeholders in determining the requirements for a software system*
2. *Perform iterative software development processes: how to plan them, how to execute them.*
3. *Point out the impact of social aspects on software development success.*
4. *Develop techniques and tools for improving team collaboration and software quality.*
5. *Perform Software process improvement as an ongoing task for development teams.*
6. *Show how agile approaches can be scaled up to the enterprise level.*

UNIT I

Agile Methodology

Theories for Agile Management – Agile Software Development – Traditional Model vs. Agile Model - Classification of Agile Methods – Agile Manifesto and Principles – Agile Project Management – Agile Team Interactions – Ethics in Agile Teams - Agility in Design, Testing – Agile Documentations – Agile Drivers, Capabilities and Values

UNIT II

Agile Processes

Lean Production - SCRUM, Crystal, Feature Driven Development- Adaptive Software Development - Extreme Programming: Method Overview – Lifecycle – Work Products, Roles and Practices.

UNIT III

Agility And Knowledge Management

Agile Information Systems – Agile Decision Making - Earl_S Schools of KM – Institutional Knowledge Evolution Cycle – Development, Acquisition, Refinement, Distribution, Deployment , Leveraging – KM in Software Engineering – Managing Software Knowledge – Challenges of Migrating to Agile Methodologies – Agile Knowledge Sharing – Role of Story-Cards – Story-Card Maturity Model (SMM).

UNIT IV

Agility And Requirements Engineering

Impact of Agile Processes in RE–Current Agile Practices – Variance – Overview of RE Using Agile – Managing Unstable Requirements – Requirements Elicitation – Agile Requirements Abstraction Model – Requirements Management in Agile Environment, Agile Requirements Prioritization – Agile Requirements Modeling and Generation – Concurrency in Agile Requirements Generation.

UNIT V

Agility And Quality Assurance

Agile Product Development – Agile Metrics – Feature Driven Development (FDD) – Financial and Production Metrics in FDD – Agile Approach to Quality Assurance - Test Driven Development – Agile Approach in Global Software Development.

TEXT BOOKS

1. David J. Anderson and Eli Schragenheim, ―Agile Management for Software Engineering: Applying the Theory of Constraints for Business Results‖, Prentice Hall, 2003.
2. Hazza and Dubinsky, ―Agile Software Engineering, Series: Undergraduate Topics in Computer Science‖, Springer, 2009.

REFERENCES

1. Craig Larman, ―Agile and Iterative Development: A Manager_s Guide‖, Addison-Wesley, 2004.
2. Kevin C. Desouza, ―Agile Information Systems: Conceptualization, Construction, and Management‖, Butterworth-Heinemann, 2007.

R19 **M. Tech – SE**

SIDDARTHA INSTITUTE OF SCIENCE AND TECHNOLOGY:PUTTUR
(AUTONOMOUS)

II M. Tech. – I - Semester

L	T	P	C
3	0	0	3

Program Elective V
(19CS5123) FUNDAMENTALS OF DATA SCIENCE

Course Objective

1. *To understand the machine learning*
2. *To get familier with regression and its techniques*
3. *To Perform effective data analysis using "R"*
4. *To solve computing issues which includes "Reading R Data", "accessing R Packages" etc.*
5. *To work practically in data warehouse modeling i.e OLAP, data cube etc.*

Course Outcomes

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

1. *Describe the different types of statistical learning.*
2. *Explore various types of regression models.*
3. *Have a good understanding of the fundamental issues and challenges of machine learning: data, model selection, model complexity, etc.*
4. *Apply data wrangling techniques.*
5. *Understood the strengths and weaknesses of many popular machine learning algorithms.*
6. *Solve the complexity issues using "R" Packages*

UNIT - I

Introduction, What Is Statistical Learning?, Why Estimate?, How Do We Estimate?, The Trade-Off Between Prediction Accuracy and Model Interpretability, Supervised Versus Unsupervised Learning, Regression Versus Classification Problems, Assessing Model Accuracy, Measuring the Quality of Fit, The Bias-Variance Trade-of, The Classification Setting, Introduction to R, Basic Commands, Graphics, Indexing Data, Loading Data, Additional Graphical and Numerical Summaries.

UNIT – II

Linear Regression, Simple Linear Regression, Multiple Linear Regression, Other Considerations in the Regression Model, Comparison of Linear Regression with K-Nearest Neighbours, Linear Regression.

UNIT-III

Classification, Logistic Regression, Linear Discriminant Analysis, A Comparison of Classification Methods, Logistic Regression, LDA, QDA, and KNN.

UNIT- IV

Programming for basic computational methods such as Eigen values and Eigen vectors

Data Wrangling: Data Acquisition, Data Formats, Imputation, The split-apply-combine paradigm. Data wrangling with r.

UNIT-V

Data Objects and Attribute Types, Basic Statistical Descriptions of Data, Data Visualization, Measuring Data Similarity and Dissimilarity.

Data Warehouse: Basic Concepts, Data Warehouse Modeling: Data Cube and OLAP, Data Warehouse Design and Usage, Data Warehouse Implementation.

TEXT BOOKS

1. Gareth James Daniela Witten Trevor Hastie, Robert Tibshirani, An Introduction to Statistical Learning with Applications in R, February 11, 2013, web link: www.statlearning.com.
2. Mark Gardener, Beginning R The statistical Programming Language, Wiley, 2015.
3. Han , Kamber, and J Pei, Data Mining Concepts and Techniques, 3rd edition, Morgan Kaufman, 2012.

REFERENCES

1. Sinan Ozdemir, Principles of Data Science, Packt Publishing Ltd Dec 2016.
2. Joel Grus, Data Science from Scratch, Oreilly media, 2015.

R19 | **M. Tech – SE**

SIDDARTHA INSTITUTE OF SCIENCE AND TECHNOLOGY:PUTTUR
(AUTONOMOUS)

II M. Tech. – I - Semester

L	T	P	C
3	0	0	3

Open Elective
(19HS0824) BUSINESS ANALYTICS

COURSE OBJECTIVES:

1. Understand the concepts and methods of business analytics.
2. To gain an understanding of how managers use business analytics to formulate and solve business problems and to support managerial decision making.
3. Identify the management related issues and processes to resolve
4. Understand the significance of forecasting models helpful in decision making
5. To become familiar with processes needed to develop, report, and analyze business data.

COURSE OUTCOMES:

After the completion of the course, student would be able to:

1. Analyze data using statistical and data mining techniques and understand relationships between the underlying business processes of an organization.
2. Design alternatives to solve business problems utilizing quantitative analysis, critical thinking and sound ethical decision making.
3. Summarize, process and transform data for obtaining meaningful conclusions
4. Interpret data using latest data analytics tools to address organisational problems
5. Organize and critically apply the concepts and methods of business analytics
6. Assess decision problems and build models for creating solutions using business analytical tools.

UNIT I

Business analytics: Overview of Business analytics - Scope of Business analytics - Business Analytics Process - Relationship of Business Analytics Process and organisation - competitive advantages of Business Analytics - Statistical Tools: Statistical Notation, Descriptive Statistical methods, Review of probability distribution and data modelling, sampling and estimation methods overview.

UNIT II

Trendiness and Regression Analysis: Modelling Relationships and Trends in Data - simple Linear Regression - Important Resources - Business Analytics Personnel - Data and models for Business analytics - problem solving - Visualizing and Exploring Data, Business Analytics Technology

UNIT III

Organization Structures of Business analytics: Team management - Management Issues - Designing Information Policy – Outsourcing - Ensuring Data Quality - Measuring contribution of Business analytics - Managing Changes - Descriptive Analytics - predictive analytics - predicative Modelling - Predictive analytics analysis - Data Mining - Data Mining Methodologies - Prescriptive analytics and its step in the business analytics Process - Prescriptive Modelling - nonlinear Optimization.

UNIT IV

Forecasting Techniques: Qualitative and Judgmental Forecasting - Statistical Forecasting Models - Forecasting Models for Stationary Time Series - Forecasting Models for Time Series with a Linear Trend - Forecasting Time Series with Seasonality - Regression Forecasting with Casual Variables - Selecting Appropriate Forecasting Models - Monte Carlo Simulation and Risk Analysis: Monte Carle Simulation Using Analytic Solver Platform - New-Product Development Model - Newsvendor Model - Overbooking Model - Cash Budget Model.

UNIT V

Decision Analysis: Formulating Decision Problems - Decision Strategies with the Outcome Probabilities - Decision Trees - The Value of Information - Utility and Decision Making - Recent Trends in Embedded and collaborative business intelligence - Visual data recovery - Data Storytelling and Data journalism.

TEXT BOOKS:

1. S. Christian Albright, *Business Analytics: Data analysis & Decision making,* Cengage Learning
2. Jeffery Camm, & others, *Essentials of Business Analytics,* Cengage Learning

REFERENCES:

1. Marc J. Schniederjans, Dara G. Schniederjans, Christopher M. Starkey,*Business analytics Principles, Concepts, and Applications,* Pearson FT Press
2. James Evans, *Business Analytics,* Pearsons Education.
3. Galit Shmueli, Peter C. Bruce, Inbal Yahav, Nitin R. Patel, Kenneth C. Lichtendahl Jr., *Data mining for business analytics: Concepts, Techniques and Applications,* WILEY

R19

M. Tech – SE

SIDDARTHA INSTITUTE OF SCIENCE AND TECHNOLOGY:PUTTUR
(AUTONOMOUS)

II M. Tech. – I - Semester

L	T	P	C
3	0	0	3

Open Elective
(19ME3121) INDUSTRIAL SAFETY

COURSE OBJECTIVES:

1. To learn about mechanical and electrical hazards.
2. To learn about Fundamentals of Maintenance Engineering.
3. To learn about Wear and Corrosion and their prevention.
4. To know about Fault Tracking
5. To learn about Periodic and preventive maintenance.

COURSE OUTCOMES:

Students undergoing this course are able *to*

1. Understand the points of factories act 1948 for health and safety.
2. Understand the cost & its relation with replacement economy.
3. Understand the concepts of Wear and Corrosion Prevention
4. Understand the concepts of sequence of fault finding activities
5. Understand the Program and schedule of preventive maintenance of mechanical and electrical equipment.
6. Understand the Periodic Maintenance of Equipment's

UNIT-I:

Industrial Safety: Accident, causes, types, results and control, mechanical and electrical hazards, types, causes and preventive steps/procedure, describe salient points of factories act 1948 for health and safety, wash rooms, drinking water layouts, light, cleanliness, fire, guarding, pressure vessels, etc, Safety color codes. Fire prevention and firefighting, equipment and methods.

UNIT-II:

Fundamentals of Maintenance Engineering: Definition and aim of maintenance engineering, Primary and secondary functions and responsibility of maintenance department, Types of maintenance, Types and applications of tools used for maintenance, Maintenance cost & its relation with replacement economy, Service life of equipment.

UNIT-III:

Wear and Corrosion and their Prevention: Wear- types, causes, effects, wear reduction methods, lubricants-types and applications, Lubrication methods, general sketch, working and applications, i. Screw down grease cup, ii. Pressure grease gun, iii. Splash lubrication, iv. Gravity lubrication, v. Wick feed lubrication vi. Side feed lubrication, vii. Ring lubrication, Definition, principle and factors affecting the corrosion. Types of corrosion, corrosion prevention methods.

UNIT-IV:

Fault Tracing: Fault tracing-concept and importance, decision tree concept, need and applications, sequence of fault finding activities, show as decision tree, draw decision tree for problems in machine tools, hydraulic, pneumatic, automotive, thermal and electrical equipment's like, I. Any one machine tool, ii. Pump iii. Air compressor, iv. Internal combustion engine, v. Boiler, vi. Electrical motors, Types of faults in machine tools and their general causes.

UNIT-V:

Periodic and Preventive Maintenance: Periodic inspection-concept and need, degreasing, cleaning and repairing schemes, overhauling of mechanical components, overhauling of electrical motor, common troubles and remedies of electric motor, repair complexities and its use, definition, need, steps and advantages of preventive maintenance. Steps/procedure for periodic and preventive maintenance of: I. Machine tools, ii. Pumps, iii. Air compressors, iv. Diesel generating (DG) sets, Program and schedule of preventive maintenance of mechanical and electrical equipment, advantages of preventive maintenance. Repair cycle concept and importance

TEXT BOOKS:

1. Higgins & Morrow, Maintenance Engineering Handbook, Da Information Services.
2. H. P. Garg, Maintenance Engineering, S. Chand and Company.

REFERENCE BOOKS:

1. Audels, Pump-hydraulic Compressors, Mcgrew Hill Publication.
2. Winterkorn, Foundation Engineering Handbook, Chapman & Hall London.

R19 **M. Tech – SE**

SIDDARTHA INSTITUTE OF SCIENCE AND TECHNOLOGY:PUTTUR
(AUTONOMOUS)

II M. Tech. – I - Semester

L	T	P	C
3	0	0	3

Open Elective

(19ME3021) ADVANCES IN OPERATIONS RESARCH

COURSE OBJECTIVES:

On successful Completion of this course the student will be able to

1. Enumerate the fundamentals of Linear Programming
2. Learn classical optimization techniques
3. Develop the best strategy of Game and identifying the Queuing theory.
4. Understand about sequence and optimum Duration of the Project
5. Develop the importance of Replacement models and Inventory control

COURSE OUTCOMES:

On successful Completion of this course the student will be able to

1. Create mathematical models of the real time situations.
2. Implement Transportation and Assignment problems to solve in real time industry
3. choose the best strategy of Game and capable of identifying the suitable queuing theory
4. Enumerate fundamental techniques and apply it to solve various optimization areas
5. Investigate, study, Apply knowledge in Replacement models and
6. Understand the Inventory control Models

UNIT-I

Introduction to OR and Linear Programming-OR definition–Types of Operations Research models; Linear Programming- Problem Formulation, Graphical Method, Simplex Method, Big-M Method, Degeneracy - Problems

UNIT-II

Transportation Problem – Formulation; Initial Basic Feasible Solution-North-West Corner Rule, Least Cost Method, Vogel‘s Approximation Method, Modified Distribution (MODI) Method, Unbalanced Transportation - Problems

Assignment Problem – Formulation, Optimal Solution -Traveling Salesman problem.

UNIT-III

Game Theory - Introduction – Minimax (Maxi mini) Criterion and Optimal Strategy, Saddle Point, Solution of Games with Pure Strategy and Mixed Strategies – 2 X 2 Games – Dominance Principle.

Queuing Theory- Introduction to queuing system–Service Channel, Arrival Pattern, Size of Population, Service Pattern, Queue Discipline, Customer Behavior, Probability Distribution-Birth & Death Process, Simple Problems on Single Service channel only.

UNIT-IV

Sequencing –Terminology - Johnson’s Algorithm for n-jobs x 2 Machines and n-jobs x 3 machines models - Problems

PERT & CPM: Introduction, Difference between PERT and CPM, Terminology-Activities, Events, Predecessor, Early Start, Early Finish, Late Start & Late Finish Times, Earliest Occurrence and Latest Occurrence of the Event, Total Float, Free Float, Independent

Float; CPM- Deterministic Model; PERT- Probabilistic Model, Critical Path, Optimal Project Duration, Least Possible Project Duration- Problems.

UNIT-V

Replacement – Failure Mechanism of Items**,** Types of Replacements- Individual Replacement policy, Group Replacement policy, Replacement of items fail suddenly – problems

Inventory - Necessity for maintaining inventory, inventory costs, classification of fixed order quantity inventory models, selective inventory management techniques.

TEXT BOOKS:

1. S D. SHARMA Operations Research KNRN Publications. 17th edition 2015
2. Hamdy A Taha , Operations Research Pearson Publications, 9 th edition 2015

REFERENCES BOOKS:

1. Manohar Mahajan Operations Research, Dhanpat Rai &Co 2016
2. Er. Prem kumar Guptha & Dr.D.S.Hira Operations Research, Schand publications 2012.
3. R Panneerselvam Operations Research PHI, 2nd edition, 2012.

R19

M. Tech – SE

SIDDARTHA INSTITUTE OF SCIENCE AND TECHNOLOGY: PUTTUR
(AUTONOMOUS)

II M. Tech. – I - Semester

L	T	P	C
3	0	0	3

Open Elective
(19CE1028) COST MANAGEMENT OF ENGINEERING PROJECTS

COURSE OBJECTIVES:
To Understand
1. Establish systems to help streamline the transactions between corporate support departments and the operating units
2. Devise transfer pricing systems to coordinate the buyer-supplier interactions between decentralized organizational operating units
3. Use pseudo profit centers to create profit maximizing behavior in what were formerly cost centers

COURSE OUTCOMES (COs)
At the end of the course, the student should be able to
1. Summarize the concept of strategic cost management, strategic cost analysis – target costing, life cycle costing and Kaizen costing and the cost drive concept
2. Describe the decision-making; relevant cost, differential cost, incremental cost and opportunity cost, objectives of a costing system
3. Summarize the meaning and different types of project management and project execution, detailed engineering activities
4. Understand the project contracts
5. Describe the cost behavior and profit planning types and contents, Bar charts and Network diagram
6. Analyze by using quantitative techniques for cost management like PERT/CPM

UNIT – I
Introduction and Overview of the Strategic Cost Management Process

UNIT-II
Cost Concepts: Cost concepts in decision-making - Relevant cost - Differential cost - Incremental cost and Opportunity cost - Objectives of a Costing System - Inventory valuation - Creation of a Database for operational control - Provision of data for Decision Making

Unit – III
Project Management: Project meaning - Different types - why to manage - cost overruns centers - various stages of project execution: conception to commissioning - Project execution as conglomeration of technical and nontechnical activities - Detailed Engineering activities - Pre project execution main clearances and documents - Project team: Role of each member -Importance Project site: Data required with significance - Project contracts - Types

and contents - Project execution Project cost control - Bar charts and Network diagram – Project commissioning: mechanical and process

UNIT – IV

Cost Behavior and Profit Planning: Cost Behavior and Profit Planning Marginal Costing - Distinction between Marginal Costing and Absorption Costing - Break-even Analysis - Cost-Volume-Profit Analysis - Various decision-making problems - Standard Costing and Variance Analysis - Pricing strategies: Pareto Analysis - Target costing - Life Cycle Costing - Costing of service sector - Just-in-time approach - Material Requirement – Planning - Enterprise Resource Planning -Total Quality Management and Theory of constraints - Activity-Based Cost Management - Bench Marking - Balanced Score Card and Value-Chain Analysis - Budgetary Control - Flexible Budgets - Performance budgets - Zero-based budgets- Measurement of Divisional profitability pricing decisions including transfer pricing

UNIT-V

Quantitative Techniques: Quantitative techniques for cost management - Linear Programming, PERT/CPM - Transportation Problems - Assignment problems – Simulation - Learning Curve Theory

TEXT BOOKS:

1. Robert S Kaplan Anthony A. Alkinson, Management & Cost Accounting
2. N.D. Vohra, Quantitative Techniques in Management, Tata McGraw Hill Book Co. Ltd 2006

REFERENCES:

1. Cost Accounting A Managerial Emphasis, Prentice Hall of India, New Delhi
2. Charles T. Horngren and George Foster, Advanced Management Accounting
3. Ashish K. Bhattacharya, Principles & Practices of Cost Accounting, A. H. Wheeler publisher
4. https://nptel.ac.in/courses/110/101/110101132/
5. https://nptel.ac.in/courses/105104161/

SIDDARTHA INSTITUTE OF SCIENCE AND TECHNOLOGY:PUTTUR
(AUTONOMOUS)

II M.Tech – I Sem (SE)

L T C
3 0 3

(19ME3022) COMPOSITE MATERIALS

COURSE OBJECTIVES:

1. To understand the mechanical behavior of composite materials
2. To get an overview of the methods of manufacturing composite materials.
3. To know the fundamentals of composite materials.
4. To understand the fabrication and process of composites.
5. To recognize the applications of composite materials.

COURSE OUTCOMES:

Upon completion of this course, the students will have an overview of

1. Fundamental concept of composite materials.
2. Different types of composite materials.
3. Fabrication and processing of composite materials.
4. MMC & CMC
5. Mechanical behavior of composite materials.
6. Application of composite materials.

UNIT-I

Introduction To Composites: Fundamentals of composites – need– enhancement of properties – classifications —Introduction to Reinforcement composites–types. Applications. Fiber production techniques for glass, carbon and ceramic fibers –Resin materials-Types.

UNIT-II

Polymer Matrix Composites: Fabrication of PMC's ,Fabrication of Fibers, Plastic Fiber Forms, Pre-pregs, Molding Compounds-Processes, Lay-Ups, Filament Winding, Pultrusion, and Recycling. Matrix – Reinforcement Interface, Wettability.

UNIT-III

MMC&CMC : Fabrication of MMC'S, Liquid Infiltration- Casting, Solid State Processes-Diffusion Bonding &In Situ Technique. Fabrication of CMC's, Hot-Pressing, Infiltration, In Situ Chemical reaction Techniques. CVD& CVI, Sol-gel.

UNIT-IV

Mechanics of Composites: Basic assumptions of laminated anisotropic plates, symmetric laminates, angle ply laminates, cross ply laminates, laminate structural moduli, evaluation of lamina properties, determination of lamina stresses, maximum stress and strain criteria, Von - Mises Yield criterion for isotropic materials, generalized Hill's criterion for anisotropic materials, Tsai-Hill's criterion for composites, prediction of laminate failure, thermal analysis of composite laminates

UNIT-V

Applications Of Composites: Applications of advanced composite materials. Environmental effects in Composites, Green composites, Synthesis and Properties of Nano composites. Surface Composites & Surface metal matrix composites: Need, Synthesis, Properties and applications.

TEXT BOOKS:

1. Mathews F. L. and Rawlings R. D., "Composite Materials: Engineering and Science", 1st Edition, Chapman and Hall, London, England, 1994.
2. Chawla K. K., "Composite materials", Second Edition, Springer – Verlag, 1998.

REFERENCES:

1. Clyne, T. W. and Withers, P. J., "Introduction to Metal Matrix Composites", Cambridge University Press, 1993.
2. Strong, A.B., "Fundamentals of Composite Manufacturing", SME, 1989.
3. Sharma, S.C., "Composite materials", Narosa Publications, 2000.

SIDDARTHA INSTITUTE OF SCIENCE AND TECHNOLOGY:PUTTUR
(AUTONOMOUS)

II M.Tech – I Sem (SE)

L	T	C
3	0	3

(19EE2128) WASTE TO ENERGY

COURSE OBJECTIVES:

The Objectives of this course

1. To understand the importance of gaining energy from the waste
2. To Understand and analyze the pattern of renewable energy resources Suggest methodologies / technologies for its utilization Economics of the utilization and environmental aspects.
3. To undusted the need and production of for biogas.

COURSE OUTCOMES:

On successful completion of the course students will be able to

1. Identify the new methodologies / technologies for effective utilization of renewable energy sources.
2. Analyse over different types of waste for energy conception.
3. Understand different types of Bio mass utilizations.

UNIT-I

INTRODUCTION TO ENERGY FROM WASTE:

Classification of waste as fuel – Agro based, Forest residue, Industrial waste - MSW – Conversion devices – Incinerators, gasifiers, digestors.

UNIT-II

BIOMASS PYROLYSIS

Pyrolysis – Types, slow fast – Manufacture of charcoal – Methods -Yields and application – Manufacture of pyrolytic oils and gases, yields and applications.

UNIT-III

BIOMASS GASIFICATION

Gasifiers – Fixed bed system – Downdraft and updraft gasifiers –Fluidized bed gasifiers – Design, construction and operation – Gasifier burner arrangement for thermal heating – Gasifier engine arrangement and electrical power – Equilibrium and kinetic consideration in gasifier operation.

UNIT-IV

BIOMASS COMBUTION

Biomass stoves – Improved chullahs, types, some exotic designs, fixed bed combustors, Types, inclined grate combustors, Fluidized bed combustors, Design, construction and operation - Operation of all the above biomass combustors.

UNIT-V

PROPERTIES OF BIOGAS (CALORIFIC VALUE AND COMPOSITION)

Biogas plant technology and status - Bio energy system - Design and constructional features -

Biomass resources and their classification - Biomass conversion processes - Thermo chemical conversion - Direct combustion - biomass gasification - pyrolysis and liquefaction - biochemical conversion - anaerobic digestion - Types of biogas Plants – Applications- Alcohol production from biomass - Bio diesel production - Urban waste to energy conversion - Biomass energy programme in India.

TEXT BOOKS:

1. Desai, Ashok V., Non-Conventional Energy, Wiley Eastern Ltd.,1990.
2. Khandelwal, K. C. and Mahdi, S. S.,Biogas Technology - A Practical HandBook - Vol. I & II, Tata McGraw Hill Publishing Co. Ltd.,1983.
3. Challal, D. S., Food, Feed and Fuel from Biomass, IBH Publishing Co. Pvt. Ltd., 1991.

R19

M. Tech – SE

SIDDARTHA INSTITUTE OF SCIENCE AND TECHNOLOGY:PUTTUR
(AUTONOMOUS)

II M. Tech. – I Sem. (SE)	**P**	**C**
	20	**10**

(19CS5124) Dissertation-I /Industrial Project

****

SIDDARTHA INSTITUTE OF SCIENCE AND TECHNOLOGY:PUTTUR
(AUTONOMOUS)

II M. Tech. – II Sem. (SE)	**P**	**C**
	32	**16**

(19CS5125) Phase –II Dissertation II

****