SAGI RAMA KRISHNAM RAJU ENGINEERING COLLEGE
(AUTONOMOUS)

(Approved by AICTE, New Delhi, Affiliated to JNTUK, Kakinada)

Accredited by NAAC with 'A+' Grade.

Recognised as Scientific and Industrial Research Organisation

SRKR MARG, CHINA AMIRAM, BHIMAVARAM – 534204 W.G.Dt., A.P., INDIA

Regulation: R25	**I - M.Tech. I - Semester**

INFORMATION TECHNOLOGY

COURSE STRUCTURE

(With effect from 2025-26 admitted Batch onwards)

Course Code	Course Name	Category	L	T	P	Cr	C.I.E.	S.E.E.	Total Marks
D2514001	Advanced Algorithms Analysis	PC	3	1	0	4	40	60	100
D2514002	Advanced Object Oriented Software Engineering	PC	3	1	0	4	40	60	100
D2514003	Artificial Intelligence	PC	3	1	0	4	40	60	100
#PE-I	Program Elective-I	PE	3	0	0	3	40	60	100
#PE-II	Program Elective-II	PE	3	0	0	3	40	60	100
D2514004	Advanced Algorithms Analysis Lab	PC	0	1	2	2	40	60	100
D2514005	Advanced Object Oriented Software Engineering lab	PC	0	1	2	2	40	60	100
D2514006	Seminar-I	PR	0	0	2	1	100	--	100
		TOTAL	**15**	**5**	**6**	**23**	**380**	**420**	**800**

	Course Code	Course Name
#PE-I	D25140A0	Secure coding
	D25140A1	Cryptography and Network Security
	D25140A2	High performance computing
	D25140A3	Computer Vision
#PE-II	D25140B0	Software Project Management
	D25140B1	Knowledge representation and reasoning
	D25140B2	Software Reliability And Quality Management
	D25140B3	Natural Language Processing

Course Code	Category	L	T	P	C	C.I.E.	S.E.E.	Exam
D2514001	PC	3	1	--	4	40	60	3 Hrs.

ADVANCED ALGORITHMS ANALYSIS

(For Information Technology)

Course Objectives:

1.	Introduce students to the advanced methods of designing and analyzing algorithms.
2.	The student should be able to choose appropriate algorithms and use it for a specific problem
3.	To familiarize students with basic paradigms and data structures used to solve advanced algorithmic problems.
4.	Students should be able to understand different classes of problems concerning their computation difficulties.
5.	To introduce the students to recent developments in the area of algorithmic design

Course Outcomes:

S.No	Outcome	Knowledge Level
1.	Analyze the complexity/performance of different algorithms.	K3
2.	Determine the appropriate data structure for solving a particular set of problems.	K3
3.	Categorize the different problems in various classes according to their complexity	K3
4.	Discuss Dynamic Programming and Fast Fourier Transform algorithm	K3
5.	Explain Linear Programming	K2

SYLLABUS

UNIT-I (10 Hrs)	**Sorting:** Review of various sorting algorithms, topological sorting Graph: Definitions and Elementary Algorithms: Shortest path by BFS, shortest path in edge-weighted case (Dijkstra's), depth-first search and computation of strongly connected components, emphasis on correctness proof of the algorithm and time/space analysis, example of amortized analysis.
UNIT-II (12 Hrs)	**Matroids:** Introduction to greedy paradigm, algorithm to compute a maximum weight maximal independent set. Application to MST.Graph **Matching:** Algorithm to compute maximum matching. Characterization of maximum matching by augmenting paths, Edmond's Blossom algorithm to compute augmenting path.
UNIT-III (12 Hrs)	**Flow-Networks:** Max flow-mincut theorem, Ford-Fulkerson Method to compute maximum flow, Edmond-Karp maximum-flow algorithm.

	Matrix Computations: Strassen's algorithm and introduction to divide and conquer paradigm, inverse of a triangular matrix, relation between the time complexities of basic matrix operations, LUP decomposition.
UNIT-IV (12 Hrs)	**Shortest Path in Graphs:** Floyd-Warshall algorithm and introduction to dynamic programming paradigm. More examples of dynamic programming. **Modulo Representation of integers/polynomials:** Chinese Remainder Theorem, Conversion between base-representation and modulo representation. Extension to polynomials. Application: Interpolation problem. **Discrete Fourier Transform (DFT):** In complex field, DFT in modulo ring. Fast Fourier Transform algorithm. Schonhage-Strassen Integer Multiplication algorithm
UNIT-V (12 Hrs)	**Linear Programming:** Geometry of the feasibility region and Simplex algorithm NP-completeness: Examples, proof of NP-hardness and NP-completeness. Recent Trends in problem solving paradigms using recent searching and sorting techniques by applying recently proposed data structures.

Text Books:	
1.	Cormen, Leiserson, Rivest, Stein, "Introduction to Algorithms".
2.	Aho, Hopcroft, Ullman "The Design and Analysis of Computer Algorithms".
3.	Kleinberg and Tardos."Algorithm Design".
4.	Data Structures: A Pseudocode Approach with C, 2nd Edition, Richard F.Gilberg, Behrouz A. Forouzon, Cengage Learning, 2004
5.	Data Structures, Algorithms and Applications in java, 2 nd Edition, SartajSahni, University Press/Orient BlackSwan, 2005
Reference Books:	
1.	Data Structures And Algorithm Analysis, 2 nd Edition, Mark Allen Weiss, Pearson, 2002
2.	Data Structures And Algorithms in C++, 3 rd Edition, Adam Drozdek, Cengage Learning, 2005
3.	C and Data Structures: A Snap Shot Oriented Treatise Using Live Engineering Examples, 1 st Edition, N.B.Venkateswarulu, E.V. Prasad, S Chand & Co, 2009
4.	Classic Data Structures, 2 nd Edition, Debasis Samantha, PHI Learning, 2009
e-Resources	
1.	https://nptel.ac.in/courses/106102064
2.	https://www.coursera.org/learn/advanced-algorithms-and-complexity

Course Code	Category	L	T	P	C	C.I.E.	S.E.E.	Exam
D2514002	PC	3	1	--	4	40	60	3 Hrs.

ADVANCED OBJECT ORIENTED SOFTWARE ENGINEERING

(For Information Technology)

Course Objectives:

1.	To introduce the principles of object-oriented programming (OOP) and its role in modern software development, including key concepts such as classes, objects, inheritance, and polymorphism.
2.	To equip students with skills in object-oriented analysis and design (OOAD) through the use of UML modeling, use cases, design patterns, and various diagrammatic techniques for effective software design.
3.	To develop an understanding of software construction, testing, and maintenance**,** including object-oriented languages (Java, C++, Python), testing strategies, version control, and refactoring methods.
4.	To explore advanced paradigms in software engineering**,** such as Model-Driven Engineering (MDE), Aspect-Oriented Programming (AOP), Component-Based and Service-Oriented Architectures (CBSE and SOA), and agile methodologies including Scrum.

Course Outcomes

S.No	Outcome	Knowledge Level
1.	Understand and apply the fundamental principles of Object-Oriented Programming (OOP) concepts and Unified Modeling Language (UML) basics, in the development of software solutions.	K3
2.	Analyze and specify software requirements, develop use cases and scenarios, apply object- oriented analysis and design (OOAD) principles	K4
3.	Experiment with the concept of test-driven development (TDD) and its practical implementation	K3
4.	Analyze and Evaluate Software Maintenance and Evolution Strategies	K4
5.	Apply Advanced Object-Oriented Software Engineering Concepts	K3

SYLLABUS

UNIT-I (10 Hrs)	**Introduction to Object-Oriented Programming:** Overview of software engineering, Introduction to Object-Oriented Programming (OOP) concepts (classes, objects, inheritance, polymorphism), Unified Modelling Language (UML) basics, Introduction to software development process and software development life cycle (SDLC).
UNIT-II	**Requirements Analysis and Design:** Requirements analysis and specification, Use cases and scenarios, Object-oriented analysis and design (OOAD), Design patterns, UML

(12 Hrs)	modelling techniques (class diagrams, sequence diagrams, state machine diagrams, activity diagrams)
UNIT-III (12 Hrs)	**Software Construction and Testing:** Software construction basics, Object oriented design principles, Object-oriented programming languages (Java, C++, Python), Software testing basics (unit testing, integration testing, system testing), Test-driven development (TDD)
UNIT-IV (12 Hrs)	**Software Maintenance and Evolution:** Software maintenance basics, refactoring techniques Software version control, Code review and inspection, Software evolution and reengineering
UNIT-V (12 Hrs)	**Advanced Topics in Object-Oriented Software Engineering:** Model-driven engineering (MDE), Aspect-oriented programming (AOP), Component based software engineering (CBSE), Service- oriented architecture (SOA), Agile software development and Scrum methodologies.

Text Books:	
1.	An Introduction to Object-Oriented Analysis and Design and the Unified Process, 3rd Edition, Craig Larman, Prentice-Hall.
2.	Programming in Java by Sachin Malhotra, Oxford University Press
Reference Books	
1.	Requirements engineering: processes and techniques, G.Kotonya and, I.Sommerville, 1998, Wiley
2.	Design Patterns, E.Gamma, R. Helm, R. Johnson, and J. Vlissides
3.	The Unified Modeling Language Reference Manual, J. Rumbaugh, I.Jacobson and G. Booch, Addison Wesley
e-Resources	
1.	https://nptel.ac.in/courses/106105153
2.	https://www.coursera.org/learn/object-oriented-design#modules

Course Code	Category	L	T	P	C	C.I.E.	S.E.E.	Exam
D2514003	PC	3	1	--	4	40	60	3 Hrs.

ARTIFICIAL INTELLIGENCE

(For Information Technology)

Course Objectives:

1.	Gain a historical perspective of AI and its foundations.
2.	Become familiar with basic principles of AI toward problem solving, inference, perception, knowledge representation, and learning.
3.	Investigate applications of AI techniques in intelligent agents, expert systems, artificial neural networks and other machine learning models.
4.	Experience AI development tools such as an „AI language", expert system shell, and/or data mining tool. Experiment with a machine learning model for simulation and analysis.
5.	Explore the current scope, potential, limitations, and implications of intelligent systems.

Course Outcomes

S.No	Outcome	Knowledge Level
1.	Apply foundational AI concepts, state-space search methods, and heuristic problems to design and analyze intelligent system solutions for defined problem scenarios.	K3
2.	Demonstrate problem reduction techniques, game search strategies to analyze and solve structured AI problems.	K3
3.	Apply suitable knowledge representation techniques to model and represent real-world problem domains in intelligent systems.	K3
4.	Implement probabilistic and reasoning frameworks to model and manage uncertainty in intelligent decision-making systems.	K3
5.	Utilize fuzzy set theory, membership functions, multi-valued logic, and fuzzy inference mechanisms to design fuzzy systems for handling imprecision and linguistic information in intelligent applications	K3

SYLLABUS

UNIT-I (10 Hrs)	**Introduction to artificial intelligence:** Introduction, history, intelligent systems, foundations of AI, applications, tic-tac-tie game playing, development of AI languages, current trends in AI, **Problem solving: state-space search and control strategies:** Introduction, general problem solving, characteristics of problem, exhaustive searches, heuristic search techniques, iterative-deepening a*, constraint satisfaction
UNIT-II	**Problem reduction and game playing:** Introduction, problem reduction, game playing,

(12 Hrs)	alpha-beta pruning, two-player perfect information games, **Logic concepts:** Introduction, propositional calculus, proportional logic, natural deduction system, axiomatic system, semantic tableau system in proportional logic, resolution refutation in proportional logic, predicate logic
UNIT-III (12 Hrs)	**Knowledge representation:** Introduction, approaches to knowledge representation, knowledge representation using semantic network, extended semantic networks for KR, knowledge representation using frames, **advanced knowledge representation techniques:** Introduction, conceptual dependency theory, script structure, cyc theory, case grammars, semantic web.
UNIT-IV (12 Hrs)	**Uncertainty measure: probability theory:** Introduction, probability theory, Bayesian belief networks, certainty factor theory, dempster-shafer theory, non-monotonic reasoning, TMS.
UNIT-V (12 Hrs)	**Fuzzy sets and fuzzy logic:** Introduction, fuzzy sets, fuzzy set operations, types of membership functions, multi valued logic, fuzzy logic, linguistic variables and hedges, fuzzy propositions, inference rules for fuzzy propositions, fuzzy systems.

Text Books:	
1.	Artificial intelligence, A modern Approach, 2nded, Stuart Russel, Peter Norvig, Prentice Hall,
2.	Artificial Intelligence, Saroj Kaushik, 1st Edition, CENGAGE Learning, 2011
Reference Books:	
1.	Artificial intelligence, structures and Strategies for Complex problem solving, 5th Edition, George F Lugar, PEA
2.	Introduction to Artificial Intelligence, Ertel, Wolf Gang, Springer, 2017
3.	Artificial Intelligence, A new Synthesis, 1st Edition, Nils J Nilsson, Elsevier, 1998
4.	Artificial Intelligence- 3rd Edition, Rich, Kevin Knight, Shiv Shankar B Nair, TMH
5.	Introduction To Artificial Intelligence And Expert Systems, 1st Edition, Patterson, Pearson India, 2015
e-Resources	
1.	https://nptel.ac.in/courses/106102220

Course Code	Category	L	T	P	C	C.I.E.	S.E.E.	Exam
D25140A0	PE	3	--	--	3	40	60	3 Hrs.

SECURE CODING

(For Information Technology)

Course Objectives:

1.	Understanding of the various security attacks and knowledge to recognize and remove common coding errors that lead to vulnerabilities.
2.	Knowledge of outline of the techniques for developing a secure application.
3.	Recognize opportunities to apply secure coding principles

Course Outcomes

S.No	Outcome	Knowledge Level
1.	Outline the secure systems and various security attacks	K2
2.	Demonstrate the development of process of software leads to secure Coding practices	K2
3.	Apply Secure programs and various risk in the software's	K3
4.	Classify various errors that lead to vulnerabilities	K3
5.	Design Real time software and vulnerabilities	K4

SYLLABUS

UNIT-I (10 Hrs)	**Introduction-**Need for secure systems, Proactive security development process, Security principles to live by and threat modelling.
UNIT-II (12 Hrs)	**Secure Coding in C-** Character strings- String manipulation errors, String Vulnerabilities and exploits Mitigation strategies for strings, Pointers, Mitigation strategies in pointer based vulnerabilities Buffer Overflow based vulnerabilities
UNIT-III (12 Hrs)	**Secure Coding in C++ and Java-** Dynamic memory management, Common errors in dynamic memory management, Memory managers, Double –free vulnerabilities, Integer security, Mitigation strategies
UNIT-IV (12 Hrs)	**Database and Web Specific Input Issues-** Quoting the Input, Use of stored procedures, Building SQL statements securely, XSS related attacks and remedies
UNIT-V (12 Hrs)	**Software Security Engineering-** Requirements engineering for secure software: Misuse and abuse cases, SQUARE process model Software security practices and knowledge

	for architecture and design
Text Books:	
1.	Writing Secure Code, 2nd Edition, Michael Howard, David LeBlanc, Microsoft Press, 2003
Reference Books:	
1.	Secure Coding in C and C++, Robert C. Sea cord, 2nd edition, Pearson Education, 2013
2.	Software Security Engineering: A guide for Project Managers, 1st ed, Julia H. Allen, Sean J.Barnum, Robert J. Ellison, Gary McGraw, Nancy R. Mead, Addison-Wesley Professional,2008
e-Resources	
1.	https://www.udemy.com/course/secure-coding-

Course Code	Category	L	T	P	C	C.I.E.	S.E.E.	Exam
D25140A1	PE	3	--	--	3	40	60	3 Hrs.

CRYPTOGRAPHY & NETWORK SECURITY

(For Information Technology)

Course Objectives:

1.	Explain the objectives of information security
2.	Explain the importance and application of each of confidentiality, integrity, authentication and availability
3.	Understand the basic categories of threats to computers and networks
4.	Discusses the Mathematics of Cryptography
5.	Discuss the fundamental ideas of Symmetric and Asymmetric cryptographic Algorithms
6.	Discusses the Network layer, Transport Layer and Application layer Protocols Enhanced security mechanisms

Course Outcomes

S.No	Outcome	Knowledge Level
1.	Student will be able to understand security issues related to computer networks and learn different symmetric key techniques	K2
2.	Students will be able learn mathematic of cryptography for symmetric and Asymmetric algorithms and apply this knowledge to understand the Cryptographic algorithms	K3
3.	Students will be able learn different types of symmetric and Asymmetric algorithms.	K3
4.	Students will be able learn different algorithms of Hash functions, message authentication and digital signature and their importance to the security.	K3
5.	Students will be able learn different Enhanced security protocols of Application Layer, Transport Layer and Network layer	K3

SYLLABUS

UNIT-I (10 Hrs)	**Security Concepts:** Introduction, The need for security, Security approaches, Principles of security, Types of Security attacks, Security services, Security Mechanisms, A model for Network Security Cryptography. Classical Encryption Techniques-symmetric cipher model, Substitution techniques, Transposition techniques, Rotor Machines, Stegnography.
UNIT-II (12 Hrs)	Introduction to Symmetric Cryptography: Algebraic Structures-Groups, Rings, Fields, GF(2^n) fields, Polynomials. Mathematics of Asymmetric cryptography: Primes, Checking

	For Primness, Eulers phi-functions, Fermat's Little Theorem, Euler's Theorem, Generating Primes, Primality Testing, Factorization, Chinese Remainder Theorem, Quadratic Congruence, Exponentiation And Logarithm.
UNIT-III (12 Hrs)	Symmetric key Ciphers: Block Cipher principles, DES, AES, Blowfish, IDEA, Block cipher operation, Stream ciphers: RC4, RC5 Asymmetric key Ciphers: Principles of public key cryptosystems, RSA algorithm, Diffie-Hellman Key Exchange, Elgamal Cryptographic system, Elliptic Curve Arithmetic, Elliptic Curve Cryptography**.**
UNIT-IV (12 Hrs)	**Cryptographic Hash Functions:** Applications of Cryptographic Hash Functions, Two Simple Hash Functions, Requirements and Security, Hash Functions Based on Cipher Block Chaining, Secure Hash Algorithms (SHA) **Message Authentication Codes:** Message Authentication Requirements, Message Authentication Functions, Requirements for Message Authentication Codes, Security of MAC'S, MAC'S Based On Hash Functions: HMAC, MAC'S Based On Block Ciphers: DAA and CMAC **Digital Signatures:** Digital Signatures, Elgamal Digital Signature Scheme, Elliptic Curve Digital Signature Algorithm, RSA-PSS Digital Signature Algorithm.
UNIT-V (12 Hrs)	**Network and Internet Security:** Transport-Level Security: Web Security Considerations, Transport Level Security, HTTPS, SSH. **IP Security:** IP Security Overview, IP Security Policy, Encapsulating Security Payload, Authentication Header Protocol. **Electronic-Mail Security:** Internet-mail Security, Email Format, Email Threats and Comprehensive Email Security, S/MIME, PGP.
Text Books:	
1.	Cryptography and Network Security - Principles and Practice: William Stallings, Pearson Education, 7th Edition, 2017
2.	Cryptography and Network Security: Behrouz A. ForouzanDebdeep, Mc Graw Hill, 3rd Edition, 2015
Reference Books:	
1.	Cryptography and Network Security: AtulKahate, Mc Graw Hill, 3rd Edition
2.	Introduction to Cryptography with Coding Theory: Wade Trappe, Lawrence C. Washington, Pearson.
3.	Modern Cryptography: Theory and Practice ByWenbo Mao. Pearson
e-Resources	
1.	https://nptel.ac.in/courses/106105162

Course Code	Category	L	T	P	C	C.I.E.	S.E.E.	Exam
D25140A2	PE	3	--	--	3	40	60	3 Hrs.

HIGH PERFORMANCE COMPUTING

(For Information Technology)

Course Objectives:

1.	The main objectives of the course is to study parallel computing hardware and programming models, performance analysis and modeling of parallel programs
2.	To develop the ability to design and analyze parallel algorithms, using decomposition techniques, task mapping strategies, and load balancing models, with an understanding of communication overheads and performance metrics.
3.	To explore parallel programming models and platforms, including shared-memory programming using threads and OpenMP, and message-passing strategies for efficient data communication in parallel environments.

Course Outcomes

S.No	Outcome	Knowledge Level
1.	Describe different parallel architectures, inter-connect networks, programming Models	K3
2.	Develop an efficient parallel algorithm to solve given problem	K4
3.	Analyze and measure performance of modern parallel computing systems	K4
4.	Build the logic to parallelize the programming task.	K2
5.	Use CUDA/OpenMP frameworks to implement and optimize parallel programs.	K4

SYLLABUS

UNIT-I (10 Hrs)	**Introduction:** Motivating Parallelism, Scope of Parallel Computing, Parallel Programming Platforms: Implicit Parallelism, Trends in Microprocessor and Architectures, Limitations of Memory, System Performance, Dichotomy of Parallel Computing Platforms, Physical Organization of Parallel Platforms, Communication Costs in Parallel Machines, Scalable design principles, Architectures: N-wide superscalar architectures, Multi-core architecture.
UNIT-II (12 Hrs)	**Parallel Programming:** Principles of Parallel Algorithm Design: Preliminaries, Decomposition Techniques, Characteristics of Tasks and Interactions, Mapping Techniques for Load Balancing, Methods for Containing Interaction Overheads, Parallel Algorithm Models, The Age of Parallel Processing, the Rise of GPU Computing, A Brief History of GPUs, Early GPU.

UNIT-III (12 Hrs)	**Basic Communication:** Operations- One-to-All Broadcast and All-to- One Reduction, All-to-All Broadcast and Reduction, All-Reduce and Prefix-Sum Operations, Scatter and Gather, All-to-All Personalized Communication, Circular Shift, Improving the Speed of Some Communication Operations. Programming shared address space platforms: threads-basics, synchronization, OpenMP programming
UNIT-IV (12 Hrs)	Analytical Models: Sources of overhead in Parallel Programs, Performance Metrics for Parallel Systems, and The effect of Granularity on Performance, Scalability of Parallel Systems, Minimum execution time and minimum cost, optimal execution time. Dense Matrix Algorithms: MatrixVector Multiplication, Matrix- Matrix Multiplication
UNIT-V (12 Hrs)	Parallel Algorithms- Sorting and Graph: Issues in Sorting on Parallel Computers, Bubble Sort and its Variants, Parallelizing Quick sort, All-Pairs Shortest Paths, Algorithm for sparse graph, Parallel Depth-First Search, Parallel BestFirst Search. CUDA Architecture : CUDA Architecture, Using the CUDA Architecture, Applications of CUDA Introduction to CUDA C-Write and launch CUDA C kernels, Manage GPU memory, Manage communication and synchronization, Parallel programming in CUDA- C.
Text Books:	
1.	AnanthGrama, Anshul Gupta, George Karypis, and Vipin Kumar, "Introduction to Parallel Computing", 2nd edition, Addison-Wesley, 2003, ISBN: 0-201-64865-2
2.	Jason sanders, Edward Kandrot, “CUDA by Example”, Addison-Wesley, ISBN-13: 978-0-13-138768-3
Reference Books:	
1.	Kai Hwang, ”Scalable Parallel Computing”, McGraw Hill 1998, ISBN:0070317984
2.	Shane Cook, “CUDA Programming: A Developer's Guide to Parallel Computing with GPUs”, Morgan Kaufmann Publishers Inc. San Francisco, CA, USA 2013 ISBN: 9780124159884
3.	David Culler Jaswinder Pal Singh, ”Parallel Computer Architecture: A Hardware/ Software Approach”, Morgan Kaufmann,1999, ISBN 978-1-55860-343-1
4.	Rod Stephens, “Essential Algorithms”, Wiley, ISBN: ISBN: 978-1-118-61210-1
e-Resources:	
1.	https://nptel.ac.in/courses/106108055

Course Code	Category	L	T	P	C	C.I.E.	S.E.E.	Exam
D25140A3	PE	3	--	--	3	40	60	3 Hrs.

COMPUTER VISION

(For Information Technology)

Course Objectives:

1.	To understand the Fundamental Concepts related to sources, shadows and shading
2.	To understand the Geometry of Multiple Views

Course Outcomes

S.No	Outcome	Knowledge Level
1.	**Demonstrate** different types of cameras their applications and inter reflections	K2
2.	**Apply** linear filtering ,Edge detections technics and its applications	K3
3.	**Develop** shot boundary detection and background subtraction techniques for video processing.	K3
4.	**Analyze** Segment regions, probabilistic, and dynamic modeling techniques.	K4
5.	**Analyze** geometric camera models , methods and evaluate model-based vision techniques	K4

SYLLABUS

UNIT-I (10 Hrs)	**Cameras :** Pinhole Cameras Radiometry–Measuring Light: Light in Space, Light Surfaces ,Important Special Cases Sources, Shadows, And Shading: Qualitative Radiometry, Sources and Their Effects, Local Shading Models, **Application:** Photometric Stereo, **Interreflections**: Global Shading Models Color: The Physics of Color, Human Color Perception, Representing Color, A Model for Image Color, Surface Color from Image Color.
UNIT-II (12 Hrs)	**Linear Filters:** Linear Filters and Convolution, Shift Invariant Linear Systems, Spatial Frequency and Fourier Transforms, Sampling and Aliasing, Filters as Templates, Edge Detection: Noise, Estimating Derivatives, Detecting Edges Texture0: Representing Texture, Analysis (and Synthesis) Using Oriented Pyramids, Application: Synthesis by Sampling Local Models, Shape from Texture.
UNIT-III (12 Hrs)	**The Geometry of Multiple Views:** Two Views Stereopsis: Reconstruction, Human Stereopsis, Binocular Fusion, Using More Cameras Segmentation by Clustering: What Is Segmentation? Human Vision: Grouping and Getstalt, Applications: Shot Boundary Detection and Background Subtraction, Image Segmentation by Clustering Pixels, Segmentation by Graph-Theoretic Clustering,

UNIT-IV (12 Hrs)	**Segmentation by Fitting a Model:** The Hough Transform, Fitting Lines, Fitting Curves, Fitting as a Probabilistic Inference Problem, Robustness Segmentation and Fitting Using Probabilistic Methods: Missing Data Problems, Fitting, and Segmentation, The EM Algorithm in Practice, Tracking With Linear Dynamic Models: Tracking as an Abstract Inference Problem, Linear Dynamic Models , Kalman Filtering, Data Association, Applications and Examples
UNIT-V (12 Hrs)	**Geometric Camera Models:** Elements of Analytical Euclidean Geometry, Camera Parameters and the Perspective Projection, Affine Cameras and Affine Projection Equations Geometric Camera Calibration: Least-Squares Parameter Estimation, A Linear Approach to Camera Calibration, Taking Radial Distortion into Account, Analytical Photogrammetry. **Case study:** Mobile Robot Localization Model- Based Vision: Initial Assumptions, Obtaining Hypotheses by Pose Consistency, Obtaining Hypotheses by pose Clustering, Obtaining Hypotheses Using Invariants, Verification, Case study: Registration In Medical Imaging Systems, Curved Surfaces and Alignment
Text Books:	
1.	David A. Forsyth and Jean Ponce: Computer Vision – A Modern Approach, PHI Learning (Indian Edition), 2009.
Reference Books:	
1	E. R. Davies: Computer and Machine Vision – Theory, Algorithms and Practicalities, Elsevier (Academic Press), 4th edition, 2013.
2	R. C. Gonzalez and R. E. Woods "Digital Image Processing" Addison Wesley 2008. 3. Richard Szeliski "Computer Vision: Algorithms and Applications" Springer-Verlag London Limited 2011.
e-Resources	
1.	https://nptel.ac.in/courses/106105216

Course Code	Category	L	T	P	C	C.I.E.	S.E.E.	Exam
D25140B0	PE	3	--	--	3	40	60	3 Hrs.

SOFTWARE PROJECT MANAGEMENT

(For Information Technology)

Course Objectives:

1.	To describe and determine the purpose and importance of project management from the perspectives of planning, tracking and completion of project
2.	To compare and differentiate organization structures and project structures
3.	To implement a project to manage project schedule, expenses and resources with the application of suitable project management tools

Course Outcomes

S.No	Outcome	Knowledge Level
1.	Recall how old software methods worked, why they had problems, and how new ways like teamwork and automation make software better and faster.	K1
2.	Demonstrate the different phases of software development and identify the key documents and outputs created in each phase, like plans, designs, and reports.	K2
3.	Organize software architecture from both management and technical views. Use iterative planning to organize tasks, set milestones, and estimate time and cost.	K3
4.	Determine how software project teams are organized and how automation and metrics help manage and improve the project and its quality.	K3
5.	Choose the basics of Agile and DevOps, how to use Scrum, and understand how DevOps tools, people, and processes help in faster and better software delivery.	K3

SYLLABUS

UNIT-I (10 Hrs)	**Conventional Software Management:** The waterfall model, conventional software Management performance. **Evolution of Software Economics:** Software Economics, pragmatics of software cost estimation. **Improving Software Economics:** Reducing Software product size, improving software processes, improving team effectiveness, improving automation, Achieving required quality, peer inspections. **The old way and the new:** The principles of conventional software Engineering, principles of modern software management, transitioning to aniterative process.
UNIT-II (12 Hrs)	**Life cycle phases:** Engineering and production stages, inception, Elaboration, construction, transition phases.

	Artifacts of the process: The artifact sets, Management artifacts, Engineering artifacts, program aticartifacts.
UNIT-III (12 Hrs)	**Model based software architectures:** A Management perspective and technical perspective. **Work Flows of the process:** Software processes work flows, Iteration workflows. **Check points of the process:** Major milestones, Minor Milestones, Periodic status assessments. **Iterative Process Planning:** Work breakdown structures, planning guidelines, cost and schedule estimating, Iteration planning process, Pragmatic planning.
UNIT-IV (12 Hrs)	**Project Organizations and Responsibilities:** Line-of Business Organizations, Project Organizations, evolution of Organizations. Process Automation: Automation Building blocks, The Project Environment. **Project Control and Process instrumentation:** The seven core Metrics, Management indicators, quality indicators, life cycle expectations, pragmatic Software Metrics, Metrics automation.
UNIT-V (12 Hrs)	Agile Methodology, ADAPTing to Scrum, Patterns for Adopting Scrum, Iterating towards Agility. **Fundamentals of DevOps:** Architecture, Deployments, Orchestration, Need, Instance of applications, DevOps delivery pipeline, DevOps ecosystem. DevOps adoption in projects: Technology aspects, Agiling capabilities, Tool stack implementation, People aspect, processes
Text Books:	
1.	Software Project Management, Walker Royce, PEA, 2005.
2.	Succeeding with Agile: Software Development Using Scrum, Mike Cohn, Addison Wesley.
3.	The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations, Gene Kim, John Willis, Patrick Debois, Jez Humb,1st Edition, O'Reilly publications, 2016.
Reference Books:	
1.	Software Project Management, BobHughes,3/e, Mike Cotterell, TMH
2.	Software Project Management, Joel Henry,PEA
3.	Software Project Management in practice, PankajJalote, PEA, 2005,
4.	Effective Software Project Management, Robert K.Wysocki,Wiley, 2006.
e-Resources	
1.	https://nptel.ac.in/courses/106105218

Course Code	Category	L	T	P	C	C.I.E.	S.E.E.	Exam
D25140B1	PE	3	--	--	3	40	60	3 Hrs.

KNOWLEDGE REPRESENTING AND REASONING

(For Information Technology)

Course Objectives:

1.	To investigate the key concepts of Knowledge Representation (KR) techniques and different notations
2.	To integrate the KR view as a knowledge engineering approach to model organizational knowledge.
3.	To introduce the study of ontologies as a KR paradigm and applications of ontologies.
4.	To understand various KR techniques and process, knowledge acquisition and sharing of ontology.

Course Outcomes

S.No	Outcome	Knowledge Level
1.	Analyze and design knowledge-based systems intended for computer implementation.	K4
2.	Demonstrate the oretical knowledge about principles for logic-based representation and reasoning	K2
3.	Make use of knowledge-engineering process	K3
4.	Experiment with production systems, frames, inheritance systems and approaches to handle uncertain or incomplete knowledge.	K3
5.	Apply reasoning techniques like fuzzy logic and nonmonotonic logic to manage vagueness and uncertainty in knowledge systems.	K4

SYLLABUS

UNIT-I (10 Hrs)	**The Key Concepts:** Knowledge, Representation, Reasoning, Why knowledge representation and reasoning, Role of logic: Historical background, Representing knowledge in logic, Varieties of logic, Name, Type, Measures, Unity Amidst diversity
UNIT-II (12 Hrs)	**Ontology:** Ontological categories, Philosophical background, Toplevel categories, Describing physical entities, Defining abstractions, Sets, Collections, Types and Categories, Space and Time
UNIT-III (12 Hrs)	**Knowledge Representations:** Knowledge Engineering, Representing structure in frames, Rules and data, Object-oriented systems, Natural language Semantics, Levels of representation

UNIT-IV (12 Hrs)	**Processes:** Times, Events and Situations, Classification of processes, Procedures, Processes and Histories, Concurrent processes, Computation, Constraint satisfaction, Change Contexts: Syntax of contexts, Semantics of contexts, First-order reasoning in contexts, Modal reasoning in contexts, Encapsulating objects in contexts.
UNIT-V (12 Hrs)	**Knowledge Soup:** Vagueness, Uncertainty, Randomness and Ignorance, Limitations of logic, Fuzzy logic, Nonmonotonic Logic, Theories, Models and the world, Semiotics Knowledge Acquisition and Sharing: Sharing Ontologies, Conceptual schema, Accommodating multiple paradigms, Relating different knowledge representations, Language patterns, Tools for knowledge acquisition
Text Books:	
1.	Knowledge Representation logical, Philosophical, and Computational Foundations by John F. Sowa, Thomson Learning.
2.	Knowledge Representation and Reasoning by Ronald J. Brachman, Hector J. Levesque, Elsevier.
Reference Books:	
1.	Logical Foundations of Artificial Intelligence by Michael Genesereth and Nils J. Nilsson, Morgan Kaufmann
2.	Handbook of Knowledge Representation by Frank van Harmelen, Vladimir Lifschitz, and Bruce Porter, Elsevier.
e-Resources	
1.	https://nptel.ac.in/courses/106106140

Course Code	Category	L	T	P	C	C.I.E.	S.E.E.	Exam
D25140B2	PE	3	--	--	3	40	60	3 Hrs.

SOFTWARE RELIABILITY AND QUALITY MANAGEMENT

(For Information Technology)

Course Objectives:

1.	To introduce the fundamentals of software reliability and its relationship with hardware reliability, availability, and modeling, emphasizing the need for reliable software in critical applications.
2.	To familiarize students with various software reliability models and metrics, including Halstead's software metric, McCabe's cyclomatic complexity, Markov models, and statistical failure prediction techniques.
3.	To develop skills in parameter estimation and model evaluation, applying methods such as Maximum Likelihood Estimation, Least Squares, and Bayesian Inference for assessing and comparing software reliability models.
4.	o provide knowledge of software quality management principles, including quality attributes, measurement, verification and validation techniques, and the role of software quality assurance (SQA) in the software development lifecycle.

Course Outcomes

S.No	Outcome	Knowledge Level
1.	Demonstrate Software Reliability during different phases of Software Development Life Cycle	K2
2	Estimate Software Reliability parameters using Maximum Likelihood and Analyze using Markovian Modeling.	K3
3	Evaluate performance of Binomial-Type, Poison-Type and Markovian Models.	K3
4	Predict Software Reliability using Intelligent Techniques	K3
5	Design Quality Attributes for Software Quality Assurance (SQA).	K3

SYLLABUS

UNIT-I (10 Hrs)	**Introduction to Software Reliability:** The need for Software Reliability, Some Basic Concepts, Software Reliability and Hardware Reliability, Availability, Modelling and General Model Characteristics.
UNIT-II (12 Hrs)	**Software Reliability Modeling:** Halstead's Software Metric, McCabe's Cyclomatic Complexity Metric, Error Seeding Models, Failure Rate Models, Curve Fitting Models, Reliability Growth Models, Markov Structure Models, Time Series Models, Nonhomogeneous Poison Process Models.

UNIT-III (12 Hrs)	**Markovian Models:** General Concepts, General Poison-Type Models, Binomial -Type Models, Poison- Type Models, Comparison of Binomial-Type and Poison-Type Models, Fault Reduction Factor for Poison- Type Models.
UNIT-IV (12 Hrs)	**Descriptions of Specific Models:** Finite Failure Category Models, Infinite Failure Category Models. Parameter Estimation: Maximum Likelihood Estimation, Least Squares Estimation, Bayesian Inference. Comparison of Software Reliability Models: Comparison Criteria, Comparison of Predictive Validity of Model Groups, Evaluation of other Criteria. Software Reliability Prediction: Problems associated with different Software Reliability Models, Software Reliability prediction parameters, Intelligent Techniques for Software Reliability Prediction.
UNIT-V (12 Hrs)	**Software Quality Management:** Software Quality Attributes, Quality Measurement & Metrics, Verification & Validation Techniques, Verification &Validation in the Life Cycle, Software Quality Assurance functions, Tool support for SQA.
Text Books:	
1.	M.Xie, Software Reliability Modelling, World Scientific;1991.
2.	John D.Musa, Anthony Iannino, Kazuhira Okumoto, Software Reliability Measurement, Prediction, Application. McGraw-Hill Book Company; 1987
3.	Hoang Pham, System Software Reliability, Springer;2005
4.	David C. Kung, Object-Oriented Software Engineering: An Agile Unified Methodology, McGraw Hill Education (India) Edition 2015.
Reference Books	
1.	M. Xie, T.N. Goh, Software Reliability: Measurement, Prediction, Application, World Scientific
2.	Daniel Galin, Software Quality Assurance: From Theory to Implementation, Pearson
e-Resources:	
1.	https://nptel.ac.in/courses/106105218

Course Code	Category	L	T	P	C	C.I.E.	S.E.E.	Exam
D25140B3	PE	3	--	--	3	40	60	3 Hrs.

NATURAL LANGUAGE PROCESSING

(For Information Technology)

Course Objectives:

1.	Students will gain an in-depth understanding of the computational properties of natural languages and the commonly used algorithms for processing linguistic information.
2.	The course examines NLP models and algorithms using both the traditional symbolic and the more recent statistical approaches.
3.	Enable students to be capable to describe the application based on natural language processing and to show the points of syntactic, semantic and pragmatic processing.

Course Outcomes

S.No	Outcome	Knowledge Level
1.	Demonstrate a given text with basic Language features	K2
2.	Design an innovative application using NLP components	K4
3.	Explain a rule based system to tackle morphology/syntax of a language	K1
4.	Design a tag set to be used for statistical processing for real-time applications	K4
5.	Compare and contrast the use of different statistical approaches for different types of NLP applications	K4

SYLLABUS

UNIT-I (10 Hrs)	**INTRODUCTION:** Origins and challenges of NLP – Language Modeling: Grammar-based LM, Statistical LM – Regular Expressions, Finite-State Automata – English Morphology, Transducers for lexicon and rules, Tokenization, Detecting and Correcting Spelling Errors, Minimum Edit Distance
UNIT-II (12 Hrs)	**WORD LEVEL ANALYSIS**: Unsmoothed N-grams, Evaluating Ngrams, Smoothing, Interpolation and Backoff – Word Classes, Partof- Speech Tagging, Rule-based, Stochastic and Transformation-based tagging, Issues in PoS tagging – Hidden Markov and Maximum Entropy models
UNIT-III (12 Hrs)	**SYNTACTIC ANALYSIS**: Context-Free Grammars, Grammar rules for English, Treebanks, Normal Forms for grammar – Dependency Grammar – Syntactic Parsing, Ambiguity, Dynamic Programming parsing – Shallow parsing Probabilistic CFG, Probabilistic CYK, Probabilistic Lexicalized CFGs – Feature structures, Unification of

	feature structures
UNIT-IV (12 Hrs)	**SEMANTICS AND PRAGMATICS:** Requirements for representation, First-Order Logic, Description Logics – Syntax-Driven Semantic analysis, Semantic attachments – Word Senses, Relations between Senses, Thematic Roles, selectional restrictions – Word Sense Disambiguation, WSD using Supervised, Dictionary & Thesaurus, Bootstrapping methods – Word Similarity using Thesaurus and Distributional methods.
UNIT-V (12 Hrs)	**DISCOURSE ANALYSIS AND LEXICAL RESOURCES:** Discourse segmentation, Coherence – Reference Phenomena, Anaphora Resolution using Hobbs and Centering Algorithm – Coreference Resolution – Resources: Porter Stemmer, Lemmatizer, Penn Treebank, Brill's Tagger, WordNet, PropBank, FrameNet, Brown Corpus, British National Corpus (BNC)

Text Books:	
1.	Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech, 2ndEdition, Daniel Jurafsky, James H. Martin -Pearson Publication, 2014.
2.	Natural Language Processing with Python, First Edition, Steven Bird, Ewan Klein and Edward Loper, OReilly Media,2009.
Reference Books:	
1.	Language Processing with Java and Ling Pipe Cookbook, 1stEdition, Breck Baldwin, Atlantic Publisher, 2015.
2.	Natural Language Processing with Java, 2ndEdition, Richard M Reese, OReilly Media, 2015.
3.	Handbook of Natural Language Processing, Second, Nitin Indurkhya and Fred J. Damerau, Chapman and Hall/CRC Press, 2010.Edition
4.	Natural Language Processing and Information Retrieval, 3rdEdition, Tanveer Siddiqui, U.S. Tiwary, Oxford University Press, 2008.
e-Resources	
1.	https://nptel.ac.in/courses/106105158

Course Code	Category	L	T	P	C	C.I.E.	S.E.E.	Exam
D2514004	PC	--	1	2	2	40	60	3 Hrs.

ADVANCED ALGORITHMS ANALYSIS LAB

(For Information Technology)

Course Objectives:

1.	The course is designed to develop skills to design and analyze different Algorithms
2.	It enables them to gain knowledge in practical applications of data structures .

Course Outcomes

S.No	Outcome	Knowledge Level
1.	Identify different classes of problems concerning their computation difficulties	K3
2.	Analyze the performance of the Advanced Algorithm	K4
3.	Apply advanced algorithm techniques to solve real world problems.	K4

LIST OF EXPERIMENTS

1.	Implement assignment problem using Brute Force method
2.	Perform multiplication of long integers using divide and conquer method.
3.	Implement a solution for the knapsack problem using the Greedy method.
4.	Implement Gaussian elimination method.
5.	Implement LU decomposition
6.	Implement Warshall algorithm
7.	Implement the Rabin Karp algorithm.
8.	Implement the KMP algorithm.
9.	Implement Harspool algorithm
10.	Implement max-flow problem.

Text Book:

1.	Design and Analysis of Algorithms, S.Sridhar, OXFORD University Press

Reference Books:

1.	Introduction to Algorithms, second edition, T.H. Cormen, C.E. Leiserson, R.L. Rivest and C. Stein, PHI Pvt. Ltd./ Pearson Education
2.	Fundamentals of Computer Algorithms, Ellis Horowitz, SatrajSahni and Rajasekharam, Universities Press
3.	Design and Analysis of algorithms, Aho, Ullman and Hopcroft, Pearson education

Course Code	Category	L	T	P	C	C.I.E.	S.E.E.	Exam
D2514005	PC	--	1	2	2	40	60	3 Hrs.

OBJECT ORIENTED SOFTWARE ENGINEERING LAB

(For Information Technology)

Course Objectives:

1.	To introduce Python programming language through its core language basics and program design techniques suitable for modern applications.
2.	To understand the wide range of programming facilities available in Python covering graphics, GUI, data visualization and Databases.
3.	To utilize high-performance programming constructs available in Python to develop solutions in real life scenarios using AI.

Course Outcomes

S.No	Outcome	Knowledge Level
1.	Design real life situational problems and think creatively about solutions of them.	K6
2.	Formulate a solution clearly and accurately in a program using Python.	K6
3.	Inspect the best features of Python to program real life problems using AI.	K4

Artificial Intelligence Specialization :

1.	Develop an IEEE standard SRS document. Also develop risk management and project plan (Gantt chart).
2.	Understanding of System modeling: Data model i.e. ER – Diagram and draw the ER Diagram with generalization, specialization and aggregation of specified problem statement
3.	Understanding of System modeling: Functional modeling: DFD level 0 i.e. Context Diagram and draw it
4.	Understanding of System modeling: Functional modeling: DFD level 1 and DFD level 2 and draw it.
5.	Identify use cases and develop the use case model.
6.	Identify the business activities and develop an UML Activity diagram.
7	Identity the conceptual classes and develop a domain model with UML Class diagram.
8	Using the identified scenarios find the interaction between objects and represent them using UML Interaction diagrams.
9	Draw the state chart diagram.
10	Identify the user interface, domain objects, and technical services. Draw the partial layered, logical architecture diagram with UML package diagram notation.
11	Implement the technical services layer.
12	Implement the domain objects layer.

13	Implement the user interface layer
14	Draw component and deployment diagrams.
Text Books:	
1.	Tremblay J.P. and Manohar R., Discrete Mathematical Structures, MGH, 1987
Reference Books:	
1.	Ellis Horowitz, SartajSahni and SanguthevarRajasekaran, Fundamentals of Computer
2.	Algorithms, Second Edition, Universities Press, 2011

Course Code	Category	L	T	P	C	C.I.E.	S.E.E.	Exam
D2514006	**PR**	**--**	**--**	**2**	**1**	**100**	**--**	**3 Hrs.**
SEMINAR-I								
A student under the supervision of a faculty member, shall collect the literature on a topic and critically review the literature and submit it to the department in a report form and shall make an oral presentation before the Project Review Committee consisting of Head of the Department, supervisor/mentor and two other senior faculty members of the department. For Seminar, there will be only internal evaluation of 100 marks. A candidate has to secure a minimum of 50% of marks to be declared successful.								

Regulation: R25	**I - M.Tech. II - Semester**

INFORMATION TECHNOLOGY

COURSE STRUCTURE
(With effect from 2025-26 admitted Batch onwards)

Course Code	Course Name	Category	L	T	P	Cr	C.I.E.	S.E.E.	Total Marks
D2524001	Full Stack Technologies	PC	3	1	0	4	40	60	100
D2524002	Machine Learning	PC	3	1	0	4	40	60	100
D2524003	Cloud Computing	PC	3	1	0	4	40	60	100
#PE-III	Program Elective-III	PE	3	0	0	3	40	60	100
#PE-IV	Program Elective-IV	PE	3	0	0	3	40	60	100
D2524004	Full Stack Technologies Lab	PC	0	1	2	2	40	60	100
D2524005	Machine Learning Lab	PC	0	1	2	2	40	60	100
D2524006	Seminar–II	PR	0	0	2	1	100	--	100
		TOTAL	**15**	**5**	**6**	**23**	**380**	**420**	**800**

	Course Code	Course Name
#PE-III	D25240A0	Software Requirements and Estimation
	D25240A1	Software Quality Engineering
	D25240A2	Software Design Methodologies
	D25240A3	Software Architecture & Design Patterns
#PE-IV	D25240B0	Agile Methodologies
	D25240B1	Social Media Analytics
	D25240B2	Design Patterns
	D25240B3	Block Chain Technologies

Course Code	Category	L	T	P	C	C.I.E.	S.E.E.	Exam
D2524001	PC	3	1	--	4	40	60	3 Hrs.

FULL STACK TECHNOLOGIES

(For Information Technology)

Course Objectives:

1.	Translate user requirements into the overall architecture and implementation of new systems and Manage Project and coordinate with the Client.
2.	Write backend code in Python/Java, PHP languages and Writing optimized front end code HTML and JavaScript.
3.	Understand, create and debug database related queries and Create test code to validate the applications against client requirement.
4.	Monitor the performance of web applications & infrastructure and troubleshooting web application with a fast and accurate a resolution

Course Outcomes

S.No	Outcome	Knowledge Level
1.	Identify the Basic Concepts of Web & Markup Languages	K3
2.	Develop web Applications using Scripting Languages & Frameworks	K3
3.	Creating & Running Back-end scripts & Connecting to Databases	K4
4.	Demonstrate Database Queries & DBMS and Working with JQuery Framework	K2
5.	Adapt to Deployment Techniques & Working with cloud	K4

SYLLABUS

UNIT-I (10Hrs)	**Introduction to web-** Internet and world wide web, Domain name service, Protocols: HTTP, FTP, SMTP, Html5 concepts, CSS3, Anatomy of a web page, XML- Document type Definition, XML schemas, Document object model, XSLT, DOM and SAX Approaches.
UNIT-II (12Hrs)	**Javascript- The Basics of Javascript:** Objects, Primitives Operations and Expressions, Control Statements, Arrays, Functions, Constructors, Pattern Matching using Regular Expressions, Angular **Java Script- AngularJS Expressions:** ARRAY, Objects, $eval, Strings, Angular JS Form Validation & Form Submission.
UNIT- III (12Hrs)	**PHP Programming: Back – end-** Scripts PHP, Node js **Working with PHP-** Using variables, Using constants, Data types, Operators. Conditional & Control statements, Arrays, functions. Working with forms and Databases such as MySQL, Node.js- Introduction, Advantages, **Node.js** Process Model, Node JS Modules

UNIT- IV (12Hrs)	**JQuery:** Introduction to JQuery, Syntax, Selectors & Events, **MySQL:** Practice MySQL Queries, Aggregate Functions, Regular Expressions, Joins & Unions, Sub-Queries, Database Connectivity with MySql.
UNIT-V (12Hrs)	**Mongo DB**- Introduction, Architecture, Features, Examples, Database Creation & Collection in Mongo DB, Deploying **Applications**- Web hosting & Domains, Deployment Using Cloud Platforms, **Web Services**- SOAP, WSDL and RESTful Architecture

Text Books:

1.	Programming the World Wide Web, 7th Edition, Robet W Sebesta, Pearson, 2013
2.	Web Technologies,1st Edition 7th impression, Uttam K Roy, Oxford, 2012.
3.	Pro Mean Stack Development, 1st Edition, ELadElrom, Apress O'Reilly, 2016
4.	JavaScript & jQuery the missing manual,2nd Edition, David sawyer mcfarland, O'Reilly, 2011
5.	Web Hosting for Dummies, 1st Edition, Peter Pollock, John Wiley & Sons,2013
6.	RESTful web services, 1st Edition, Leonard Richardson, Ruby, O'Reilly, 2007

Reference Books:

1.	Ruby on Rails Up and Running, Lightning fast Web development, 1st Edition, Bruce Tate, Curt Hibbs, Oreilly, 2006
2.	Programming Perl, 4th edition, Tom Christiansen, Jonathan Orwant, O'Reilly ,2012
3.	Web Technologies, HTML< JavaScript, PHP, Java, JSP, XML and AJAX, Black book, 1st Edition, Dream Tech, 2009
4.	An Introduction to Web Design, Programming, 1st Edition, Paul S Wang, Sanda S Katila, Cengage Learning, 2003

e-Resources

1.	http://www.upriss.org.uk/perl/PerlCourse.html
2.	https://nptel.ac.in/courses/106106156

Course Code	Category	L	T	P	C	C.I.E.	S.E.E.	Exam
D2524002	PC	3	1	--	4	40	60	3 Hrs.

MACHINE LEARNING

(For Information Technology)

Course Objectives:

1.	Define machine learning and its different types (supervised and unsupervised) and understand their applications.
2.	Apply supervised learning algorithms including decision trees and k-nearest neighbours (k-NN).
3.	Implement unsupervised learning techniques, such as K-means clustering.

Course Outcomes:

S.No	Outcome	Knowledge Level
1.	Enumerate the Fundamentals of Machine Learning	K2
2.	Build Nearest neighbour based models	K2
3.	Apply Models based on decision trees and Bayes rule	K4
4.	Choose appropriate clustering technique	K2
5.	Determine algorithms to generate code for a target machine	K4

SYLLABUS

UNIT-I (10Hrs)	**Introduction to Machine Learning:** Evolution of Machine Learning, Paradigms for ML, Learning by Rote, Learning by Induction, Reinforcement Learning, Types of Data, Matching, Stages in Machine Learning, Data Acquisition, Feature Engineering, Data Representation, Model Selection, Model Learning, Model Evaluation, Model Prediction, Search and Learning, Data Sets
UNIT- II (12Hrs)	**Nearest Neighbor-Based Models:** Introduction to Proximity Measures, Distance Measures, Non-Metric Similarity Functions, Proximity Between Binary Patterns, Different Classification Algorithms Based on the Distance Measures ,K-Nearest Neighbor Classifier, Radius Distance Nearest Neighbor Algorithm, KNN Regression, Performance of Classifiers, Performance of Regression Algorithms
UNIT- III (12Hrs)	**Models Based on Decision Trees:** Decision Trees for Classification, Impurity Measures, Properties, Regression Based on Decision Trees, Bias–Variance Trade-off, Random Forests for Classification and Regression. **The Bayes Classifier:** Introduction to the Bayes Classifier, Bayes' Rule and Inference, The Bayes Classifier and its Optimality, Multi- Class Classification | Class Conditional Independence and Naïve Bayes Classifier

	(NBC)
UNIT- IV (12Hrs)	**Linear Discriminants for Machine Learning:** Introduction to Linear Discriminants, Linear Discriminants for Classification, Perceptron Classifier, Perceptron Learning Algorithm, Support Vector Machines, Linearly Non-Separable Case, Non-linear SVM, Kernel Trick, Logistic Regression, Linear Regression, Multi-Layer Perceptron's (MLPs), Backpropagation for Training an MLP.
UNIT-V (12Hrs)	**Clustering :** Introduction to Clustering, Partitioning of Data, Matrix Factorization | Clustering of Patterns, Divisive Clustering, Agglomerative Clustering, Partitional Clustering, K-Means Clustering, Soft Partitioning, Soft Clustering, Fuzzy C-Means Clustering, Rough Clustering, Rough K-Means Clustering Algorithm, Expectation Maximization-Based Clustering, Spectral Clustering
Text Books:	
1.	"Machine Learning Theory and Practice", M N Murthy, V S Ananthanarayana, Universities Press (India), 2024
2.	Applied Machine Learning, M. Gopal, McGraw Hill Education
Reference Books:	
1.	Machine Learning", Tom M. Mitchell, McGraw-Hill Publication, 2017
2.	"Machine Learning in Action",Peter Harrington, DreamTech
3.	"Introduction to Data Mining", Pang-Ning Tan, Michel Stenbach, Vipin Kumar, 7th Edition, 2019
e-Resources	
1.	https://nptel.ac.in/courses/106106139

Course Code	Category	L	T	P	C	C.I.E.	S.E.E.	Exam
D2524003	PC	3	1	--	4	40	60	3 Hrs.

CLOUD COMPUTING

(For Information Technology)

Course Objectives:

1.	To explain the evolving utility computing model called cloud computing.
2.	To introduce the various levels of services offered by cloud.
3.	To discuss the fundamentals of cloud enabling technologies such as distributed computing, service-oriented architecture and virtualization
4.	To emphasize the security and other challenges in cloud computing
5.	To introduce the advanced concepts such as containers, server less computing and cloud-centric Internet of Things.

Course Outcomes

S.No	Outcome	Knowledge Level
1.	Illustrate the basic concepts, service models, deployment models, and benefits of cloud computing	K2
2.	Describe the enabling technologies such as parallel/distributed computing, RPC, SOA, and web services used in cloud environments.	K2
3.	Explain virtualization and container technologies and their role in building scalable cloud infrastructures.	K2
4.	Analyze major challenges in cloud computing including scalability, interoperability, security, and economics.	K2
5.	Explain advanced cloud concepts such as serverless computing, IoT, edge/fog computing, DevOps, and quantum cloud computing.	K2

SYLLABUS

UNIT-I (10 Hrs)	**Introduction to Cloud Computing Fundamentals:** Cloud computing at a glance, defining a cloud, cloud computing reference model, types of services(IaaS, PaaS, SaaS), cloud deployment models (public, private, hybrid), utility computing, cloud computing characteristics and benefits, cloud service providers (Amazon Web Services, Microsoft Azure, Google AppEngine).
UNIT-II (12 Hrs)	**Cloud Enabling Technologies:** Ubiquitous Internet, parallel and distributed computing, elements of parallel computing, hardware architectures for parallel computing (SISD, SIMD, MISD, MIMD), elements of distributed computing, Inter-process communication, technologies for distributed computing, remote procedure calls (RPC), service-oriented architecture (SOA), Web services, virtualization.

UNIT- III (12 Hrs)	**Virtualization and Containers:** Characteristics of virtualized environments, taxonomy of virtualization techniques, virtualization and cloud Computing, pros and cons of virtualization, technology examples (XEN, VMware), building blocks of containers, container platforms (LXC, Docker), container orchestration, Docker Swarm and Kubernetes, public cloud VM(e.g. Amazon EC2) and container (e.g. Amazon Elastic Container Service)offerings
UNIT-IV (12 Hrs)	**Cloud computing challenges:** Economics of the cloud, cloud interoperability and standards, scalability and fault tolerance, energy efficiency in clouds, federated clouds, cloud computing security, fundamentals of computer security, cloud security architecture, cloud shared responsibility model, security in cloud deployment models.
UNIT-V (12 Hrs)	**Advanced concepts in cloud computing:** Server less computing, Function-as-a-Service, serverless computing architecture, public cloud (e.g. AWS Lambda) and opensource (e.g. OpenFaaS) serverless platforms, Internet of Things (IoT), applications, cloud-centric IoT and layers, edge and fog computing, DevOps, infrastructure-as-code, quantum cloud computing.
Text Books:	
1	Andrew Stellman, Jill Alison Hart, Learning Agile, O'Reilly, 2015.
2	Cloud Computing, Theory and Practical approach, 1st Edition, Dan C Marinescu, MK Elsevier Publisher, 2013
3	Cloud Computing, A Practical approach, 1st Edition, Anthony T Velte, Toby J Velte, Robert Elsenpeter,TMH, 2017
Reference Books:	
1	Andrew stellman, Jennifer Green, Head first Agile, O'Reilly, 2017.
2	Rubin K , Essential Scrum : A practical guide to the most popular Agile process, Addison-Wesley, 2013
e-Resources	
1.	https://nptel.ac.in/courses/106105167

Course Code	Category	L	T	P	C	C.I.E.	S.E.E.	Exam
D25240A0	PE	3	--	--	3	40	60	3 Hrs.

SOFTWARE REQUIREMENTS AND ESTIMATION

(For Information Technology)

Course Objectives:

1.	Students will author a software requirements document.
2.	Students will demonstrate an understanding of the proper contents of a software requirements document and proficiency in software development cost estimation

Course Outcomes By the end of this course student will be able to

S.No	Outcome	Knowledge Level
1.	Identify the importance of software requirements and their validation	K2
2.	Analyse the software requirement management and its principles.	K4
3.	Apply and estimate the factors and approaches for software cost.	K4
4.	Understand the Schedule and Cost Estimation in software requirements.	K2
5.	Utilize the Tools for Requirements and Estimation Management.	K3

SYLLABUS

UNIT-I (10 Hrs)	**Software Requirements and Engineering:** Software Requirements - Essential Software requirement, Good practices for requirements engineering, improving requirements processes, Software requirements and risk management. 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-II (12 Hrs)	**Software Requirements Management and Modeling:** Software Requirements Management - Requirements Management Principles and practices, Requirements attributes, Change Management Process, Requirements Traceability Matrix, and Links in requirements chain. Software Requirements Modeling Use Case Modeling, Analysis Models, Dataflow diagram, state transition diagram, class diagrams, Object analysis, Problem Frames
UNIT- III (12 Hrs)	**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, and Conversion between size measures.

UNIT- IV (12 Hrs)	**Effort, Schedule and Cost Estimation:** Productivity, Estimation Factors, Approaches to Effort and Schedule Estimation, COCOMO II, Putnam Estimation Model, Algorithmic models, Cost Estimation.
UNIT-V (12 Hrs)	**Tools for Requirements Management and Estimation 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, SLIM (Software Life Cycle Management) Tools.
Text Books:	
1.	Karl Wiegers & Joy Beatty, Software Requirements (3rd Edition), Microsoft Press, 2013. ISBN: 978-0735679665
2.	Software Requirements and Estimation by Rajesh Naik and Swapna Kishore, Tata Mc Graw Hill.
Reference Books:	
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, TMH, 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
e-Resources	
1.	https://nptel.ac.in/courses/106101061

Course Code	Category	L	T	P	C	C.I.E.	S.E.E.	Exam
D25240A1	PE	3	--	--	3	40	60	3 Hrs.

SOFTWARE QUALITY ENGINEERING

(For Information Technology)

Course Objectives:

1.	To introduce the fundamentals of software quality by exploring perspectives, expectations, frameworks (such as ISO-9126), and measurement techniques for correctness, defects, and quality attributes.
2.	To equip students with quality assurance (QA) techniques including defect prevention, defect reduction, defect containment, and risk management strategies for developing reliable software systems.
3.	To develop skills in quality engineering through planning, assessment, process integration, and continuous improvement within the software development lifecycle.
4.	To provide practical knowledge of testing strategies and automation, including test planning, execution, coverage analysis, usage-based statistical testing, and operational profile construction for real-world applications.

Course Outcomes

S.No	Outcome	Knowledge Level
1	Understand software quality and its perspectives	K2
2	Analyze defect prevention and defect reduction in software quality assurance.	K4
3	Illustrate software quality engineering activities and its process	K3
4	Understand and apply various test planning, execution, automation, and management techniques for software testing. Analyze coverage and usage-based testing methods using checklists, partitions, and operational profiles with real-world case studies.	K2
5	Analyze coverage and usage-based testing methods using checklists, partitions, and operational profiles with real-world case studies.	K4

SYLLABUS

UNIT-I (10 Hrs)	**Software Quality:** perspectives and expectations, Quality frameworks and ISO-9126, correctness and defects: Definitions, properties and Measurements, Ahistorical perspective of quality, software quality.
UNIT-II (12 Hrs)	**Quality Assurance:** Classification: QA as dealing with defects, Defect prevention- Education and training, Formal method, Other defect prevention techniques, Defect Reduction- Inspection: Direct fault detection and removal, Testing: Failure observation and fault removal, other techniques and risk identification, Defect Containment- software

	fault tolerance, safety assurance and failure containment
UNIT-III (12 Hrs)	**Quality Engineering:** Activities and process, Quality planning: Goal setting and Strategy formation, Quality assessment and Improvement, Quality engineering in software process.
UNIT-IV (12 Hrs)	**Test Activities, Management and Automation:** Test planning and preparation, Test execution, Result checking and measurement, Analysis and follow- up, Activities People and Management, Test Automation.
UNIT-V (12 Hrs)	**Coverage and usage testing based on checklist and partitions:** Checklist based testing and its limitations, Testing for partition Coverage, Usage based Statistical testing with Musa's operational profiles, Constructing operational profiles Case Study: OP for the cartridge Support Software
Text Books:	
1.	JeffTia`n, Software Quality Engineering, Testing, Quality Assurance, and Quantifiable improvement
2.	Richard N.Taylor, Software Architecture: Foundations, Theory, and Practice
Reference Books:	
1	Pressman, R. S. & Maxim, B. R., Software Engineering: A Practitioner's Approach (8th Edition) McGraw-Hill Education, 2014.
2	Pfleeger, S. L. & Atlee, J. M., Software Engineering: Theory and Practice (4th Edition) Pearson Education, 2010.
e-Resources	
1.	https://nptel.ac.in/courses/106101061

Course Code	Category	L	T	P	C	C.I.E.	S.E.E.	Exam
D25240A2	PE	3	--	--	3	40	60	3 Hrs.

SOFTWARE DESIGN METHODOLOGIES

(For Information Technology)

Course Objectives:

1	To develop the knowledge, understanding, skills and values to solve problems through the creation of software solutions
2	To design and experiment with software prototypes
3	To elicit, analyze and specify software requirements through a productive working relationship with project stakeholders
4	To build solutions using different technologies, architectures and life-cycle approaches.

S.No	Outcome	Knowledge Level
1	Understand the need of software design and challenges.	K2
2	Demonstrate architectures and methods for software design.	K2
3	Analyze software design process with objects and components.	K4
4	Identify suitable metrics for project and process management	K3
5	Analyze project scheduling and risk management strategies	K4

SYLLABUS

UNIT-I (10 Hrs)	**Software Requirements:** Functional and non-functional requirements, User requirements, System requirements, Interface specification, the software requirements document. Requirements engineering process: Feasibility studies, Requirements elicitation and analysis, Requirements validation, Requirements management.
UNIT-II (12 Hrs)	**Software Design** The nature of the design process, transferring design knowledge, constraints upon the design process and product, recording design decisions, designing with others, context for design, economic factors, assessing design qualities, quality attributes of the design product, assessing the design process. Representing abstract ideas, design viewpoints, the architecture concept, design methods, design patterns, design representations, rationale for design methods. Design Processes and Strategies: The role of strategy in design methods, describing the design process – The D – Matrix, design by top-down decomposition, design by composition, organizational influences upon design.
UNIT-III (12 Hrs)	**Designing with objects and components:** Designing with objects- Design practices for object-oriented paradigm, Object-oriented

	paradigm, Object-oriented frameworks, Hierarchical object oriented design process and heuristics, the fusion method, the unified process. Component - based design- The component concept, designing with components, designing components, COTS. User Interface design-The Golden rules, Interface analysis and design models, user and task analysis, analysis of display content and work environment, applying interface design issues, design evaluation.
UNIT-IV (12 Hrs)	**Project Management Concepts:** Project Management- The management spectrum, people, product, process and project, W5HH principle, Critical practices. Metrics for Process and Projects- Process metrics, project metrics, size- oriented metrics, function-oriented metrics, Object-oriented and usecase metrics, metrics for software quality, integrating metrics within software process.
UNIT-V (12 Hrs)	**Project Scheduling and Management** Project Scheduling- Basic concepts, project scheduling, defining at askset andtask network, timeline charts, tracking the schedule, tracking the progress for an OO project, Earned value analysis. Risk Management- Reactive vs. Proactive risk strategies, software risks, risk identification, risk projection, risk refinement, risk mitigation and monitoring, the RMMM plan.

Text Books:	
1.	Software design, David Budgen, second edition, Pearson education,2003.
2.	Software Engineering: A practitioner's Approach, Roger S Pressman, sixth edition. McGraw-Hill International Edition, 2005.
Reference Books:	
1.	Applying domain-driven design and patterns, jimmy Nilsson, Pearson education,2006
2.	Software Engineering Foundations, Ian Somerville, seventh edition, Pearson education, 2004.
3.	Software Project Management, Bob Hughes & Mike Cotterell, Fourth edition, Tata Mc Graw Engineering: A Primer, Waman S Jawadekar, Tata McGraw-Hill, 2008
4.	The Art of Project Management, Scott Berkun, O'Reilly,2005.
5.	Software Engineering, Project Management, Richard H. Thayer & Edward Yourdon, second edition, Wiley India, 2004.
6.	Software Engineering foundations, Yingxu Wang Auerbach publications,2008.
7.	Applied Software Project Management, Andrew Stellman& Jennifer Greene,O'Reilly,200
e-Resources	
1.	https://nptel.ac.in/courses/106101061

Course Code	Category	L	T	P	C	C.I.E.	S.E.E.	Exam
D25240A3	PE	3	--	--	3	40	60	3 Hrs.

SOFTWARE ARCHITECTURE & DESIGN PATTERNS

(For Information Technology)

Course Objectives:

1	To introduce the concept of design patterns and their role in solving recurring software design problems, including selection, usage, and organization of pattern catalogs.
2	To develop skills in object-oriented analysis and design (OOAD) by identifying requirements, defining conceptual classes, and applying domain knowledge to create effective system designs.
3	To explore and implement key design patterns, particularly structural patterns (Adapter, Bridge, Composite, Decorator, Facade, Flyweight, Proxy) and architectural patterns such as MVC, for building interactive and maintainable systems.
4	To understand and apply distributed object-oriented design concepts, including client-server systems, Java RMI, web services (SOAP, RESTful), and enterprise integration using service-oriented architectures.

S.No	Outcome	Knowledge Level
1	Understand the basic concepts to identify state behavior of real world objects	K2
2	Apply Object Oriented Analysis and Design concepts to solve complex problems	K3
3	Construct various UML models using the appropriate notation for specific problem context	K3
4	Design models to Show the importance of systems analysis and design in solving complex problems using case studies	K3
5	Study of Pattern Oriented approach for real world problems	K2

SYLLABUS

UNIT-I (10 Hrs)	**Introduction:** What is a design pattern? Describing design patterns, the catalog of design pattern, organizing the catalog, how design patterns solve design problems, how to select a design pattern, how to use a design pattern What is object oriented development? key concepts of object oriented design other related concepts, benefits and drawbacks of the paradigm
UNIT-II (12 Hrs)	**Analysis a System:** Overview of the analysis phase, stage 1 gathering the requirements functional requirements specification, defining conceptual classes and relationships, using the knowledge of the domain Design and Implementation, discussions and further reading

UNIT-III (12 Hrs)	**Design Pattern Catalog:** Structural patterns, Adapter, bridge, composite, decorator, facade, flyweight, proxy.
UNIT-IV (12 Hrs)	**Interactive systems and the MVC architecture:** Introduction The MVC architectural pattern, analyzing a simple drawing program designing the system, designing of the subsystems, getting into implementation, implementing undo operation drawing incomplete items, adding a new feature pattern based solutions
UNIT-V (12 Hrs)	**Designing with Distributed Objects:** Client server system, java remote method invocation, implementing an object oriented system on the web, Web services (SOAP, Restful), Enterprise Service Bus

Text Books:	
1.	Object oriented analysis, design and implementation, brahma dathan, sarnathrammath , universities press,2013
2.	Design patterns, Erich Gamma, Richard helan , Ralph johman , john vlissides, PEARSON Publication,2013
Reference Books:	
1.	Frank Bachmann, ReginеMeunier , Hans Rohnert “Pattern Oriented Software Architecture” Volume 1, 1996.
2.	William J Brown et al., "Anti Patterns: Refactoring Software, Architectures and Projects in Crisis", John Wiley, 1998
e-Resources	
1.	https://www.youtube.com/watch?v=gcs8_l3fkVo&list=PLhwVAYxlh5dusp7Y8-KV0azc_KsCohEg

Course Code	Category	L	T	P	C	C.I.E.	S.E.E.	Exam
D25240B0	PE	3	--	--	3	40	60	3 Hrs.

AGILE METHODOLOGIES

(For Information Technology)

Course Objectives:

1.	The objective of this course are to introduce the important concepts of agile software development Process
2.	Emphasize the role of stand-up meetings in software collaboration
3.	Impart the knowledge on values and principles in understanding agility

Course Outcomes:

S. No	Outcome	Knowledge Level
1.	How to learn Agile and get into brain	K2
2.	Explain about the Agile Principles and Agile Project	K3
3.	Understand the foundational principles of Scrum and its rules.	K2
4.	Describe the concepts of XP, Simplicity, and Incremental Design	K2
5.	Develop a new Lean, Eliminating Waste, and Seeing	K4

SYLLABUS

UNIT-I (10 Hrs)	**Learning Agile:** Getting Agile into your brain, Understanding Agile values, No Silver Bullet, Agile to the Rescue, adding Agile makes a difference. A fractured perspective, How a fractured perspective causes project problems. The Agile Manifesto, Purpose behind Each Practice. Individuals and Interactions Over Processes and Tools, Working Software over Comprehensive Documentation, Customer Collaboration over Contract Negotiation, Responding to Change over Following a Plan, Principles over Practices. Understanding the Elephant, Methodologies Help You Get It All in Place at Once, Where to Start with a New Methodology
UNIT-II (12 Hrs)	**The Agile Principles:** The 12 Principles of Agile Software, The Customer Is Always Right, "Do As I Say, Not As I Said". Delivering the Project, Better Project Delivery for the Ebook Reader Project. Communicating and Working Together, Better Communication for the Ebook Reader Project. Project Execution—Moving the Project Along, A Better Working Environment for the Ebook Reader Project Team. Constantly Improving the Project and the Team. The Agile Project: Bringing All the Principles Together
UNIT-III	**SCRUM and Self-Organizing Teams:** The Rules of Scrum, Act I: I Can Haz Scrum?,

(12 Hrs)	Everyone on a Scrum Team owns the Project, The Scrum Master Guides the Team's Decisions, The Product Owner Helps the Team Understand the Value of the Software, Everyone Owns the Project, Scrum Has Its Own Set of Values ,Status Updates Are for Social Networks!, The Whole Team Uses the Daily Scrum, Feedback and the Visibility-Inspection-Adaptation Cycle, The Last Responsible Moment, How to Hold an Effective Daily Scrum. Sprinting into a Wall, Sprints, Planning, and Retrospectives, Iterative or Incremental?, The Product Owner Makes or Breaks the Sprint, Visibility and Value, How to Plan and Run an Effective Scrum Sprint **Scrum Planning And Collective Commitment:** Not Quite Expecting the Unexpected, User Stories, Velocity, and Generally Accepted Scrum Practices, Make Your Software Useful, User Stories Help Build Features Your Users Will Use, Conditions of Satisfaction, Story Points and Velocity, Burndown Charts, Planning and Running a Sprint Using Stories, Points, Tasks, and a Task Board. Victory Lap, Scrum Values Revisited, Practices Do Work Without the Values (Just Don't Call It Scrum), Is Your Company's Culture Compatible with Scrum Values.
UNIT-IV (12 Hrs)	**XP And Embracing Change:** Going into Overtime, The Primary Practices of XP, Programming Practices, Integration Practices, Planning Practices, Team Practices, Why Teams Resist Changes, and How the Practices Help. The Game Plan Changed, but We're Still Losing, The XP Values Help the Team Change Their Mindset, XP Helps Developers Learn to Work with Users, Practices Only "Stick" When the Team Truly Believes in Them, An Effective Mindset Starts with the XP Values, The XP Values, Paved with Good Intentions. The Momentum Shifts, Understanding the XP Principles Helps You Embrace Change, The Principles of XP, XP Principles Help You Understand Planning, XP Principles Help You Understand Practices—and Vice Versa, Feedback Loops. **XP, Simplicity, and Incremental Design:** Code and Design, Code Smells and Antipatterns (or, How to Tell If You're Being Too Clever), XP Teams Look for Code Smells and Fix Them, Hooks, Edge Cases, and Code That Does Too Much. Make Code and Design Decisions at the Last Responsible Moment, Fix Technical Debt by Refactoring Mercilessly, Use Continuous Integration to Find Design Problems, Avoid Monolithic Design, Incremental Design and the Holistic XP Practices. Teams Work Best When They Feel Like They Have Time to Think, Team Members Trust Each Other and Make Decisions Together. The XP Design, Planning, Team, and Holistic Practices Form an Ecosystem Incremental Design Versus Designing for Reuse, When Units Interact in a Simple Way, the System Can Grow Incrementally, Great Design Emerges from Simple Interactions, Final Score.
UNIT-V (12 Hrs)	**Lean, Eliminating Waste, and Seeing the whole:** Lean Thinking, Commitment, Options Thinking, and Set-Based Development, Creating Heroes and Magical Thinking. Eliminate Waste, Use a Value Stream Map to Help See Waste Clearly, Gain a Deeper Understanding of the Product, See the Whole, Find the Root Cause of Problems That You Discover. Deliver As Fast As Possible, Use an Area Chart to Visualize Work in Progress,

	Control Bottlenecks by Limiting Work in Progress. **Kanban, Flow, and Constantly Improving:** The Principles of Kanban, Find a Starting Point and Evolve Experimentally from There. Stories Go into the System; Code Comes Out, Improving Your Process with Kanban, Visualize the Workflow, Limit Work in Progress. Measure and Manage Flow, Managing Flow with WIP Limits Naturally Creates Slack. Make Process Policies Explicit So Everyone Is on the Same Page. Emergent Behavior with Kanban. **The Agile Coach:** Coaches Understand Why People Don't Always Want to Change. The Principles of Coaching.
Text Books:	
1.	Andrew Stellman, Jill Alison Hart, Learning Agile, O'Reilly, 2015.
Reference Books:	
1.	Andrew stellman, Jennifer Green, Head first Agile, O'Reilly, 2017.
2.	Rubin K , Essential Scrum : A practical guide to the most popular Agile process, Addison-Wesley, 2013
e-Resources	
1.	https://nptel.ac.in/courses/110104073

Course Code	Category	L	T	P	C	C.I.E.	S.E.E.	Exam
D25240B1	PE	3	--	--	3	40	60	3 Hrs.

SOCIAL MEDIA ANALYTICS

(For Information Technology)

Course Objectives:

1.	To introduce the evolution of the web and social media by exploring Web 1.0, Web 2.0, Web 3.0, types of social media platforms, and their applications for personal, professional, and business purposes
2.	To develop an understanding of social media analytics including its purpose, types, analytical frameworks, tools, and challenges in comparison with traditional business analytics.
3.	To equip students with practical skills in text, actions, and hyperlink analytics for extracting insights from social media data using appropriate analytical tools and techniques.
4.	To apply social media analytics through case studies for real-world problem-solving in marketing, customer engagement, and business decision-making.

Course Outcomes

S.No	Outcome	Knowledge Level
1	Understanding characteristics and types of social media.	K2
2	Knowledge on layers of social media analytics	K2
3	Apply text analysis tools on social media data	K3
4	Understand the significance of action analytics	K2
5	Detect viral topics on social media(YouTube)	K2

SYLLABUS

UNIT-I (10 Hrs)	**Introduction To Social Media:** World Wide Web, Web 1.0, Web 2.0, Web 3.0, Social Media, Core Characteristics Of Social Media ,Types Of Social Media, Social Networking Sites, Using Face book For Business Purposes, Content Communities
UNIT-II (12 Hrs)	**Social Media Analytics Overview:** Purpose of Social Media Analytics, Social Media Vs. Traditional Business Analytics, Seven Layers Of Social Media Analytics, Types Of Social Media Analytics, Social Media Analytics Cycle, Challenges To Social Media Analytics, Social Media Analytics Tools. **Case Study:** The Underground Campaign That Scored Big
UNIT-III (12 Hrs)	**Social Media Text Analytics:** Types Of Social Media Text, Purpose Of Text Analytics, Steps In Text Analytics, Social Media Text Analysis Tools. **Case Study:** Tapping into Online Customer Opinions

UNIT-IV (12 Hrs)	**Social Media Actions Analytics:** Introduction To Actions Analytics, Common Social Media Actions, Actions Analytics Tools. **Case Study:** Cover-More Group
UNIT-V (12 Hrs)	**Social Media Hyperlink Analytics:** Types Of Hyperlinks, Hyperlink Analytics, Types Of Hyperlink Analytics, Hyperlink Analytics Tools. **Case Study:** Hyperlinks and Viral YouTube Videos
Text Books:	
1.	Seven Layers of Social Media Analytics Mining Business Insights From Social Media Text, Actions, Networks, Hyperlinks, Apps, Search Engine, And Location Data By Gohar F.Khan Isbn: 1507823207, Isbn-13: 9781507823200
Reference Books:	
1	Social Media Analytics: Techniques and Insights for Extracting Business Value out of Social Media By Matthew Ganis, Avinash Kohirkar, Pearson Education.
2	Social Media Analytics: Effective Tools for Building, Interpreting, and Using Metrics, Marshall Sponder, MGH.
3	Big Data and Analytics, Seema Acharya, Subhasinin Chellappan, WileyPublications.
4	BigData, Black Booktm,Dreamtech Press,2015Edition
e-Resources	
1.	https://nptel.ac.in/courses/106106239

Course Code	Category	L	T	P	C	C.I.E.	S.E.E.	Exam
D25240B2	PE	3	--	--	3	40	60	3 Hrs.

DESIGN PATTERNS

(For Information Technology)

Course Objectives:

1	To understand various segments of storage technology and architectures
2	To explore the inherent power of information
3	To describe the different backup, recovery and replication strategies

Course Outcomes

S.No	Outcome	Knowledge Level
1	Understand the evolution of storage technology and Intelligent Storage Systems	**K2**
2	Explore the key concepts of various Storage Networking Technologies DAS, SANs, NAS and CAS	**K3**
3	Understand the basics of Storage Virtualization	**K2**
4	Understand the concepts of Storage security and Storage Infrastructure Mana gement	**K2**
5	Analyze the purpose of backup, recovery and replication Strategies	**K4**

SYLLABUS

UNIT-I (10 Hrs)	**Introduction to Information Storage and Management:** Information Storage, Evolution of Storage Technology and Architecture, Data Center Infrastructure, Key Challenges in Managing Information, Information Lifecycle. Storage System Environment - Data Protection: RAID - Intelligent Storage System.
UNIT-II (12 Hrs)	**Direct-Attached Storage and Introduction to SCSI:** Types of DAS, DAS Benefits and Limitations, Disk Drive Interfaces, Introduction to Parallel SCSI, SCSI Command Model, **Storage Area Networks:** Fibre Channel: Overview, The SAN and Its Evolution, Components of SAN, FC Connectivity, Fibre Channel Ports, Fibre Channel Architecture, Zoning, Fibre Channel Login Types, FC Topologies, Concepts in Practice: EMC Connectrix Network-Attached Storage: General-Purpose Servers vs. NAS Devices, Benefits of NAS, NAS File I/O, Components of NAS, NAS Implementations, NAS File-Sharing Protocols, NAS I/O Operations, Factors Affecting NAS Performance and Availability, Concepts in Practice: EMC Celerra
UNIT-III (12 Hrs)	**Content-Addressed Storage:** Fixed Content and Archives, Types of Archives, Features and Benefits of CAS, CAS Architecture, Object Storage and Retrieval in CAS, CAS

	Examples, **Concepts in Practice:** EMC Centera Storage Virtualization: Forms of Virtualization, SNIA Storage Virtualization Taxonomy, Storage Virtualization Configurations, Storage Virtualization Challenges, Types of Storage Virtualization, Concepts in Practice
UNIT-IV (12 Hrs)	**Backup and Recovery** Backup Purpose, Backup Considerations, Backup Granularity, Recovery Considerations, Backup Methods, Backup Process, Backup and Restore Operations, Backup Topologies, Backup in NAS Environments, Backup Technologies, Concepts in Practice: EMC Net Worker Local Replication: Local Replication, Source and Target, Uses of Local Replicas, Data Consistency, Local Replication Technologies, Restore and Restart Considerations, Creating Multiple Replicas, Management Interface, Concepts in Practice: EMC Time Finder and EMC Snap View Remote Replication: Modes of Remote Replication, Remote Replication Technologies, Network Infrastructure, Concepts in Practice: EMC SRDF, EMC SAN Copy, and EMC Mirror View
UNIT-V (12 Hrs)	**Securing the Storage Infrastructure** Storage Security Framework, Risk Triad, Storage Security Domains, Security Implementations in Storage Networking **Managing the Storage Infrastructure** Monitoring the Storage Infrastructure, Storage Management Activities, Storage Infrastructure Management Challenges, Developing an Ideal Solution, Concepts in Practice: EMC Control Center
Text Books:	
1.	Marc Farley Osborne, "Building Storage Networks'', Tata McGraw-Hill, 2001.
2.	Robert Spalding and Robert Spalding, "Storage Networks: The Complete Reference", Tata McGraw Hill, 2003.
3.	Meeta Gupta, "Storage Area Network Fundamentals", Pearson Education Ltd., 2002.
Reference Books:	
1.	GeraldJ Kowalski and Mark TMaybury," Information Storage Retrieval Systems theory &Implementation", BS Publications, 2000.
2.	The jendra BS, "Disaster Recovery& Business continuity", Shroff Publishers& Distributors, 2006.
e-Resources	
1.	https://nptel.ac.in/courses/106105224

Course Code	Category	L	T	P	C	C.I.E.	S.E.E.	Exam
D25240B3	PE	3	--	--	3	40	60	3 Hrs.

BLOCK-CHAINTECHNOLOGIES

(For Information Technology)

Course Objectives:

1.	To learn the fundamentals of Block Chain and various types of block chain and consensus mechanism.
2.	To understand public block chain system, Private block chain system and consortium block chain.
3.	Able to know the security issues of block chain technology

Course Outcomes

S.No	Outcome	Knowledge Level
1	Understand crypto currencies like Bitcoin, altcoins, and how they are used.	K2
2	Explain the concept, characteristics, types, and real-world applications of smart contracts and oracles.	K2
3	Analyze smart contracts, permissioned algorithms, and applications of blockchain in private and consortium environments.	K4
4	Understand the security, privacy, and regulatory aspects of blockchain systems including Bitcoin and Hyperledger Fabric.	K2
5	Build real-world blockchain applications across sectors like retail, banking, healthcare, and energy through case studies.	K3

SYLLABUS

UNIT-I (10Hrs)	**Fundamentals of Blockchain:** Introduction, Origin of Blockchain, Blockchain Solution, Components of Blockchain, Block in a Blockchain, 77 The Technology and the Future. Blockchain Types and Consensus Mechanism: Introduction, Decentralization and Distribution, Types of Blockchain, Consensus Protocol. Cryptocurrency: Bitcoin, Altcoin and Token: Introduction, Bitcoin and the Cryptocurrency, Cryptocurrency Basics, Typesof Cryptocurrencies, Cryptocurrency Usage.
UNIT-II (10Hrs)	**Public Blockchain System:** Introduction, Public Blockchain, Popular Public Blockchains, The Bitcoin Blockchain, Ethereum Blockchain. Smart Contracts: Introduction, Smart Contract, Characteristics of a Smart Contract, Types of Smart Contracts, Types of Oracles, Smart Contracts in Ethereum, Smart Contracts in Industry.
UNIT-III (10Hrs)	**Private Blockchain System:** Introduction, Key Characteristics of Private Blockchain, Private Blockchain, Private Blockchain Examples, Private Blockchain and Open Source, E- commerce Site Example, Various Commands (Instructions) in E-commerce

	Blockchain, Smart Contract in Private Environment, State Machine, Different Algorithms of Permissioned Blockchain, Byzantine Fault, Multichain. Consortium Blockchain: Introduction, Key Characteristics of Consortium Blockchain, Need of Consortium Blockchain, Hyperledger Platform, Overview of Ripple, Overview of Corda.
UNIT-IV (10Hrs)	**Security in Blockchain:** Introduction, Security Aspects in Bitcoin, Security and Privacy Challenges of Blockchain in General, Performance and Scalability, Identity Management and Authentication, Regulatory Compliance and Assurance, Safeguarding Blockchain Smart Contract (DApp), Security Aspects in Hyperledger Fabric.
UNIT-V (10Hrs)	**Blockchain Case Studies:** Case Study 1 – Retail, Case Study 2 – Banking and Financial Services, Case Study 3 – Healthcare, Case Study 4 – Energy and Utilities. Blockchain Platform using Python: Introduction, Learn How to Use Python Online Editor, Basic Programming Using Python, Python Packages for Blockchain. Blockchain platform using Hyperledger Fabric: Introduction, Components of Hyperledger Fabric Network, Chain codes from Developer.ibm.com, Blockchain Application Using Fabric Java SDK.
Text Books:	
1.	"Block chain Technology", Chandramouli Subramanian, Asha A.George, Abhilasj K A and MeenaKarthikeyan , Universities Press.
2	Mastering Blockchain: Unlocking the Power of Cryptocurrencies, Smart Contracts, and Decentralized Applications (4th Edition), Imran Bashir, Packt Publishing, 2023. ISBN: 978-1803241067
Reference Books:	
1	Block chain Blue print for Economy, Melanie Swan, SPD Oreilly.
2	Block chain for Business, Jai Singh Arun, Jerry Cuomo, Nitin Gauar, Pearson Addition Wesley
e-Resources	
1.	https://nptel.ac.in/courses/106105235

Course Code	Category	L	T	P	C	C.I.E.	S.E.E.	Exam
D2524004	PC	--	1	2	2	40	60	3 Hrs.

FULL STACK TECHNOLOGIES LAB

(For Information Technology)

Course Objectives:

1.	Learn the core concepts of both the frontend and backend programming course
2.	Get familiar with the latest web development technologies.
3.	Learn all about SQL and Mongo databases.

Course Outcomes

S.No	Outcome	Knowledge Level
1	Apply HTML, CSS, and JavaScript to design static and dynamic web pages with form validation, event handling, and responsive embedded multimedia content.	**K4**
2	Develop server-side and client-side programs using Node.js, jQuery, PHP, and Perl to implement interactive features, database connectivity, and user authentication mechanisms.	**K5**
3	Design and manipulate XML data using DTDs for validation, and integrate file system operations and modular programming in web-based applications.	**K5**
4	Implement web application functionalities with database integration by performing CRUD operations, executing SQL queries, and managing user data for registration and login systems.	**K5**

LIST OF EXPERIMENTS

1.	1. Design a web page consisting of a) Home page b) Login page c) Catalogue page.
2.	Design a webpage to Embed elements like google maps and youtube into the webpage and make them responsive
3.	Design a dynamic web page with validation using JavaScript.
4.	Design a HTML having a text box and four buttons viz Factorial, Fibonacci, Prime, and Palindrome. When a button is pressed an appropriate javascript function should be called to display a. Factorial of that number b. Fibonacci series up to that number c. Prime numbers up to that number d. Is it palindrome or not
5.	Write JavaScript programs on Event Handling

	a. Validation of registration form b. Open a Window from the current window c. Change color of background at each click of button or refresh of a page d. Display calendar for the month and year selected from combo box e. On Mouse over event
6.	Write an XML file which will display the Book information which includes the following: 1) Title of the book 2) 2) Author Name 3) 3) ISBN number 4) 4) Publisher name 5) 5) Edition 6) 6) Price 7) Write a Document Type Definition (DTD) to validate the above XML 8) file.
7.	Write a program to create Calculator Node.js Module with functions adds, subtract & multiply and use the Calculator module in another Node.js file.
8.	Write a Node.js for File System to perform the following operations i) Create a File ii) Read a File iii) Write to a File iv) Delete a File
9.	Write a program to implement jQuery Selectors and Operations
10.	Write a program to implement jQuery Event Handling
11.	Write an example perl program to connect to a MySQl database table and execute simple commands
12.	Write a PHP program for registering users of a website and login
13.	User Authentication: Assume four users user1,user2,user3 and user4 having the passwords pwd1, pwd2, pwd3 and pwd4 respectively. Write a PHP for doing the following. a. Write a program to Create a Cookie and add these four user id's and passwords to this Cookie. b. Read the user id and passwords entered in the Login form (week1) and authenticate with the values (user id and passwords) available in the cookies. If he is a valid user(i.e., user-name and password match) you should welcome him by name(user-name) else you should display " You are not an authenticated user ''. Use init-parameters to do this.
14.	Install a database (Mysql or Oracle): Create a table which should contain at least the following fields: name, password, email-id, phone number(these should hold the data from the registration form). a) Write a PHP program to connect to that database and extract data from the tables and display

	them. b) Experiment with various SQL queries. c) Insert the details of the users who register with the web site, Whenever a new user clicks the submit button in the registration page.
15.	Write a PHP program which does the following job: Insert the details of the 3 or 4 users who register with the web site by using registration form. Authenticate the user when he submits the login form using the user name and password from the database (Similar to week8 instead of cookies).
Reference Books:	
1.	Jeffrey C. Jackson, "Web Technologies--A Computer Science Perspective", Pearson Education, 2006
2.	Robert. W. Sebesta, "Programming the World Wide Web", Fourth Edition, Pearson Education, 2007
3.	AngularJS: Up and Running Enhanced Productivity with Structured Web Apps By Brad Green, Shyam Seshadri Publisher: O'Reilly Media
4.	Learning React Functional Web Development with React and Redux By Alex Banks, Eve Porcello Publisher: O'Reilly Media
5.	Head First Java, 2nd Edition by Bert Bates, Kathy Sierra Publisher: O'Reilly Media, Inc

Course Code	Category	L	T	P	C	C.I.E.	S.E.E.	Exam
D2524005	PC	--	1	2	2	40	60	3 Hrs.

MACHINE LEARNING LAB

(For Information Technology)

Course Objectives:

1.	To learn about computing central tendency measures and Data pre- processing techniques
2.	To learn about classification and regression algorithms
3.	To apply different clustering algorithms for a problem.

Course Outcomes

S.No	Outcome	Knowledge Level
1	Apply statistical measures and preprocessing techniques such as central tendency, dispersion, attribute selection, handling missing values, discretization, and outlier elimination to prepare datasets for analysis.	**K4**
2	Implement and evaluate supervised learning algorithms including KNN, Decision Trees, Random Forest, Naïve Bayes, Support Vector Machines, Logistic Regression, and Multi-layer Perceptron for classification and regression tasks.	**K5**
3	Apply and assess unsupervised learning algorithms such as K-Means, Fuzzy C-Means, and Expectation Maximization for clustering, and analyze their performance based on parameter tuning and distance measures.	**K4**
4	Demonstrate the ability to select and tune models by optimizing algorithm parameters, comparing performance metrics, and applying suitable models to solve real-world machine learning problems.	**K4**

LIST OF EXPERIMENTS

1.	Compute Central Tendency Measures: Mean, Median, Mode Measure of Dispersion: Variance, Standard Deviation.
2.	Apply the following Pre-processing techniques for a given dataset. a. Attribute selection b. Handling Missing Values c. Discretization d. Elimination of Outliers
3.	Apply KNN algorithm for classification and regression
4.	Demonstrate decision tree algorithm for a classification problem and Perform parameter tuning for better results
5.	Demonstrate decision tree algorithm for a regression problem
6.	Apply Random Forest algorithm for classification and regression

7.	Demonstrate Naïve Bayes Classification algorithm
8.	Apply Support Vector algorithm for classification
9.	Demonstrate simple linear regression algorithm for a regression problem
10.	Apply Logistic regression algorithm for a classification problem
11.	Demonstrate Multi-layer Perceptron algorithm for a classification problem
12.	Implement the K-means algorithm and apply it to the data you selected. Evaluate performance by measuring the sum of the Euclidean distance of each example from its class center. Test the performance of the algorithm as a function of the parameters K.
13.	Demonstrate the use of Fuzzy C-Means Clustering
14.	Demonstrate the use of Expectation Maximization based clustering algorithm
Reference Books:	
1.	Han, J., Kamber, M., & Pei, J. Data Mining: Concepts and Techniques (3rd Edition) Morgan Kaufmann, 2011.
2.	Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani, An Introduction to Statistical Learning: with Applications in R (2nd Edition) Springer, 2021.
3.	Aurélien Géron Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow (3rd Edition) O'Reilly Media, 2022.
4.	Mehmed Kantardzic Data Mining: Concepts, Models, Methods, and Algorithms (3rd Edition) Wiley-IEEE Press, 2020.

Course Code	Category	L	T	P	C	C.I.E.	S.E.E.	Exam
D2524006	**PR**	**--**	**--**	**2**	**1**	**100**	**--**	**3 Hrs.**
SEMINAR-II								
A student under the supervision of a faculty member, shall collect the literature on a topic and critically review the literature and submit it to the department in a report form and shall make an oral presentation before the Project Review Committee consisting of Head of the Department, supervisor/mentor and two other senior faculty members of the department. For Seminar, there will be only internal evaluation of 100 marks. A candidate has to secure a minimum of 50% of marks to be declared successful								

SAGI RAMA KRISHNAM RAJU ENGINEERING COLLEGE
(AUTONOMOUS)
(Approved by AICTE, New Delhi, Affiliated to JNTUK, Kakinada)
Accredited by NAAC with 'A+' Grade.
Recognised as Scientific and Industrial Research Organisation
SRKR MARG, CHINA AMIRAM, BHIMAVARAM – 534204 W.G.Dt., A.P., INDIA

Regulation: R25	**II - M.Tech. I - Semester**

INFROMATION TECHNOLOGY

COURSE STRUCTURE
(With effect from 2025-26 admitted Batch onwards)

Course Code	Course Name	Category	L	T	P	Cr	C.I.E.	S.E.E.	Total Marks
D2534001	Research Methodology and IPR / Swayam 12 week MOOC course – RM&IPR		3	0	0	3	40	60	100
D2534002	Summer Internship/ Industrial Training (8-10 weeks) *	PR	--	--	--	3	100	--	100
D2534003	Comprehensive Viva[#]	PR	--	--	--	2	100	--	100
D2534004	Dissertation Part – A[$]	PR	--	--	20	10	100	--	100
		TOTAL	**3**	**-**	**20**	**18**	**340**	**60**	**400**

Student attended during summer / year break and assessment will be done in 3rd Sem.

Comprehensive viva can be conducted courses completed upto second sem.

$ Dissertation – Part A, internal assessment

Course Code	Category	L	T	P	C	CIE	SEE	Exam
D2534001	PC	3	--	--	3	40	60	3 Hrs.

RESEARCH METHODOLOGY AND IPR

(For Information Technology)

Course Objectives:

1.	To bring awareness on Research Methodology and research ethics.
2.	Familiarize the concepts of IPR.

Course Outcomes:

S.No	Course Outcome	Knowledge Level
1.	Identify the research problem through effective literature review and data analysis	**K3**
2.	Develop a technical paper with essential sections	**K3**
3.	Choose the patents, trade, and copyrights for protecting intellectual creations	**K3**
4.	Identify patents rights and transfer of technology	**K3**
5.	Identify appropriate IPR mechanism for protecting various types of intellectual creations.	**K3**

SYLLABUS

UNIT-I (10Hrs)	Meaning of research problem, Sources of research problem, Criteria Characteristics of a good research problem, Errors in selecting a research problem, Scope and objectives of research problem. Approaches of investigation of solutions for research problem, data collection, analysis, interpretation, Necessary instrumentations
UNIT-II (12Hrs)	Effective literature studies approaches, analysis Plagiarism, Research ethics, 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-III (12Hrs)	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-IV (12Hrs)	Patent Rights: Scope of Patent Rights. Licensing and transfer of technology. Patent information and databases. Geographical Indications.
UNIT-V	New Developments in IPR: Administration of Patent System. New developments in IPR;

(12Hrs)	IPR of Biological Systems, Computer Software etc. Traditional knowledge Case Studies, IPR and IITs.
Textbooks:	
1.	Stuart Melville and Wayne Goddard, ―Research methodology: an introduction for science & engineering students'
2.	Wayne Goddard and Stuart Melville, ―Research Methodology: An Introduction
3.	Ranjit Kumar, 2nd Edition, ―Research Methodology: A Step by Step Guide for beginners
Reference Books:	
1.	Halbert, ―Resisting Intellectual Property‖, Taylor & Francis Ltd, 2007.
2.	Mayall, ―Industrial Design‖, McGraw Hill, 1992.
3.	Niebel, ―Product Design‖, McGraw Hill, 1974.
4.	Asimov, ―Introduction to Design‖, Prentice Hall, 1962
5.	Robert P. Merges, Peter S. Menell, Mark A. Lemley, ― Intellectual Property in New Technological Age, 2016.
6.	T. Ramappa, ―Intellectual Property Rights Under WTO‖, S. Chand, 2008

Course Code	Category	L	T	P	C	C.I.E.	S.E.E.	Exam
D2534002	**PR**	**--**	**--**	**--**	**3**	**100**	**--**	**3 Hrs.**
Summer Internship/ Industrial Training (8-10 weeks)*								
Students shall undergo mandatory summer internship / industrial training for a minimum of eight weeks duration at the end of second semester of the Programme/Summer Break. A student will be required to submit a summer internship/industrial training report to the concerned department and appear for an oral presentation before the committee. The Committee comprises of a Professor of the department and two faculty. The report and the oral presentation shall carry 40% and 60% weightages respectively. For summer internship / industrial training, there will be only internal evaluation of 100 marks. A candidate has to secure a minimum of 50% of marks to be declared successful.								

Student attended during summer / year break and assessment will be done in 3rd Sem

Course Code	Category	L	T	P	C	C.I.E.	S.E.E.	Exam
D2534003	PR	--	--	--	2	100	--	3 Hrs.
Comprehensive Viva								
The objective of comprehensive viva-voce is to assess the overall knowledge of the student in the relevant field of Engineering/Specialization in the PG program. Viva will be conducted in 3rd semester. The duration of the viva will be around 30 min. The examination committee will be constituted by the HoD and consist of Professor of the department and two faculty. For comprehensive viva-voce, there will be only internal evaluation of 100 marks. A candidate has to secure a minimum of 50% of marks to be declared successful.								

Comprehensive viva can be conducted courses completed up to second sem.

Course Code	Category	L	T	P	C	C.I.E.	S.E.E.	Exam
D2534004	PR	--	--	20	10	100	--	3 Hrs.

Dissertation Part – A

The Student has to register for Dissertation-I / Industrial project in III semester. Student has to submit, in consultation with his project supervisor, the title, objective and plan of action of his project work for approval. The student can initiate the Project work, only after obtaining the approval from the Project Review Committee (PRC).

Continuous assessment of Dissertation-I during the III-Semester will be monitored by the PRC.

Dissertation-Part A will be only internal evaluation by PRC for 100 marks. A candidate has to secure a minimum of 50% of marks to be declared successful.

The candidate shall submit a status report to the PRC in two stages, each accompanied by an oral presentation, with a minimum interval of three months between the two

SAGI RAMA KRISHNAM RAJU ENGINEERING COLLEGE
(AUTONOMOUS)
(Approved by AICTE, New Delhi, Affiliated to JNTUK, Kakinada)
Accredited by NAAC with 'A+' Grade.
Recognised as Scientific and Industrial Research Organisation
SRKR MARG, CHINA AMIRAM, BHIMAVARAM – 534204 W.G.Dt., A.P., INDIA

Regulation: R25	**II - M.Tech. II - Semester**

INFORMATION TECHNOLOGY

COURSE STRUCTURE
(With effect from 2025-26 admitted Batch onwards)

Course Code	Course Name	Category	L	T	P	Cr	C.I.E.	S.E.E.	Total Marks
D2544001	Dissertation Part – B[%]	PR	--	--	32	16	--	100	100
		TOTAL	**--**	**--**	**32**	**16**	**--**	**100**	**100**

% External Assessment

Course Code	Category	L	T	P	C	C.I.E.	S.E.E.	Exam
D2544001	PR	--	--	32	16	--	100	3 Hrs.

Dissertation Part – B

The student has to continue his/her work from Dissertation Part-A to complete Dissertation Part-B in IV semester.

Continuous assessment of Dissertation Part-B during IV-Semester will be monitored by the PRC.

Dissertation Part-B is evaluated for 100 external marks based on Review and Viva Voce.

Review and Viva-Voce examination shall be conducted by a board consisting of the Supervisor, Head of the Department and the examiner who adjudicated the Thesis. The Board shall jointly report the candidate's work for 100 marks.

If the report of the Viva-Voce is unsatisfactory (ie, < 50 marks), the candidate shall retake the Viva-Voce examination only after three months. If he fails to get a satisfactory report at the second Viva-Voce examination, the candidate has to reregister for the project and complete the project within the stipulated time after taking the approval from the College.