Shirpur Education Society's

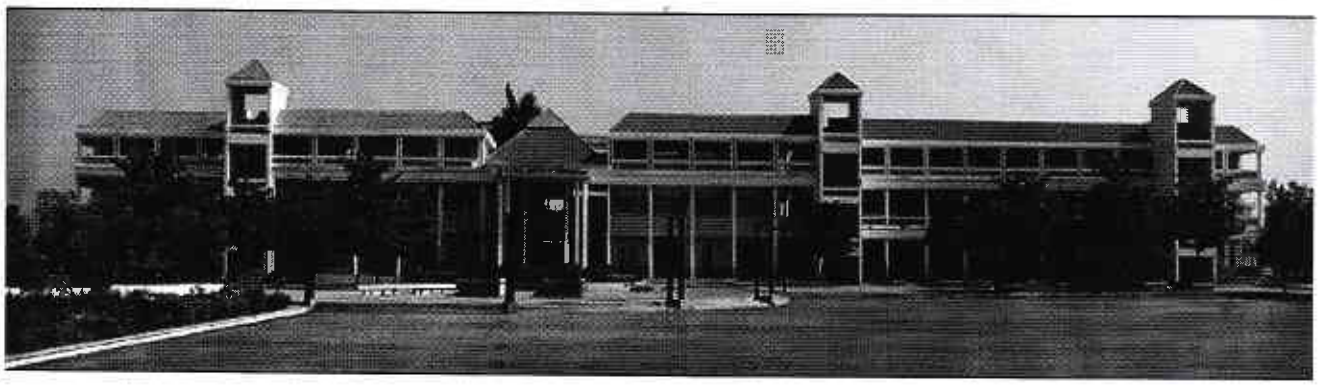
R. C. Patel Institute of Technology, Shirpur

(An Autonomous Institute)

Course Structure and Syllabus

Third Year B. Tech. (Computer Engineering)

With effect from Year 2026-27
(RCP23 NEP Scheme: Revision 1)



Shahada Road, Near Nimzari Naka, Shirpur, Maharashtra 425405
Ph: 02563 259 802, Web: www.rcpit.ac.in

Third Year B.Tech Computer Engineering Semester-V (w.e.f. 2026-27)

Sr. No.	Course Category	Course Code	Course Title	Teaching Scheme			Evaluation Scheme					Total	Credits	
							Continuous Assessment (CA)							
							TA	Term Test 1 (TT1)	Term Test 2 (TT2)	Average of(TT1 & TT2)	ESE			
				L	T	P	[A]			[B]	[C]	[A+B+C]		
1	PC	RCP23CPC301	Machine Learning	3			25	15	15	15	60	100	3	4
	PC	RCP23CPC301L	Machine Learning Laboratory			2	25				25	50	1	
2	PC	RCP23CPC302	Automata Theory and Compiler Design	3			25	15	15	15	60	100	3	4
	PC	RCP23CPC302L	Automata Theory and Compiler Design Laboratory			2	25				25	50	1	
3	PC	RCP23CPC303	Business Intelligence and Analytics	3			25	15	15	15	60	100	3	4
	PC	RCP23CPC303L	Business Intelligence and Analytics Laboratory			2	25				25	50	1	
4	PC	RCP23CPC304L	Web Programming Laboratory			2	25				25	50	1	1
5#	PE	RCP23CPE311	Advanced Algorithms	3			25	15	15	15	60	100	3	4
		RCP23CPE311L	Advanced Algorithms Laboratory			2	25				25	50	1	
		RCP23CPE312	Statistical Modeling	3			25	15	15	15	60	100	3	
		RCP23CPE312L	Statistical Modeling Laboratory			2	25				25	50	1	
		RCP23CPE313	Enterprise Data Systems	3			25	15	15	15	60	100	3	
		RCP23CPE313L	Enterprise Data Systems Laboratory			2	25				25	50	1	
		RCP23CPE314	Computer Graphics	3			25	15	15	15	60	100	3	
		RCP23CPE314L	Computer Graphics Laboratory			2	25				25	50	1	
6	MD	RCP23CMD301	Information and Cyber Security	3			25	15	15	15	60	100	3	4
	MD	RCP23CMD301L	Information and Cyber Security Laboratory			2	25				25	50	1	
7	SC	RCP23XSC301P	Semester Project - III			2	25				25	50	1	
8	HS	RCP23XHS311	Constitution of India	1									A	
9	SC	RCP23CSC301	Mastering Neural Networks	1			40					40	1	
			Total	17		14	340			75	475	890	23	

Any 1 Programme Elective Course and A Audit Course.

Prepared by: Dr. J. S. Sonawane

Checked by: Dr. S. M. Pardeshi

Prof. Dr. R. B. Wagh
BOS Chairman

Prof. S. P. Shukla
C.O.E.

Prof. Dr. P. J. Deore
Dean Academics/Dy. Director

Prof. Dr. J. B. Patil
Director

R. C. Patel Institute of Technology, Shirpur
SES
Autonomous

Third Year B.Tech Computer Engineering Semester-VI (w.e.f. 2026-27)

Sr. No.	Course Category	Course Code	Course Title	Teaching Scheme			Evaluation Scheme					Total	Credits	
							Continuous Assessment (CA)							
							TA	Term Test 1 (TT1)	Term Test 2 (TT2)	Average of(TT1 & TT2)	ESE			
				L	T	P	[A]			[B]	[C]	[A+B+C]		
1	PC	RCP23CPC351	Deep Learning	3			25	15	15	15	60	100	3	4
	PC	RCP23CPC351L	Deep Learning Laboratory			2	25				25	50	1	
2	PC	RCP23CPC352	Blockchain and Crypto Technology	3			25	15	15	15	60	100	3	4
	PC	RCP23CPC352L	Blockchain and Crypto Technology Laboratory			2	25				25	50	1	
3	PC	RCP23CPC353L	Software Engineering and DevOPs Laboratory			2	25				25	50	1	1
4#	PE	RCP23CPE361	Big Data Analytics	3			25	15	15	15	60	100	3	4
		RCP23CPE361L	Big Data Analytics Laboratory			2	25				25	50	1	
		RCP23CPE362	UI/UX Design	3			25	15	15	15	60	100	3	
		RCP23CPE362L	UI/UX Design Laboratory			2	25				25	50	1	
		RCP23CPE363	Virtual Reality	3			25	15	15	15	60	100	3	
		RCP23CPE363L	Virtual Reality Laboratory			2	25				25	50	1	
5#	PE	RCP23CPE371	Computer Vision	3			25	15	15	15	60	100	3	4
		RCP23CPE371L	Computer Vision Laboratory			2	25				25	50	1	
		RCP23CPE372	Development Frameworks	3			25	15	15	15	60	100	3	
		RCP23CPE372L	Development Frameworks Laboratory			2	25				25	50	1	
		RCP23CPE373	High Performance Computing	3			25	15	15	15	60	100	3	
		RCP23CPE373L	High Performance Computing Laboratory			2	25				25	50	1	
6	MD	RCP23CMD351	Ethical Hacking and Digital Forensics	2			25	15	15	15	60	100	2	3
	MD	RCP23CMD351L	Ethical Hacking and Digital Forensics Laboratory			2	25				25	50	1	
7	EL	RCP23XEL351P	Project Stage - I			4	25				25	50	2	
8	HS	RCP23XHS361T	Environmental Studies		1		25					25	1	
9	SC	RCP23CSC351	Generative Artificial Intelligence	1			40					40	1	
			Total	15	1	16	365			75	475	915	24	

Any 1 Programme Elective Course

Prepared by: JSonawane
Dr. J. S. Sonawane

Checked by:
Dr. S. M. Pardeshi

Prof. Dr. R. B. Wagh
BOS Chairman

Prof. S. P. Shukla
C.O.E.

Prof. Dr. P. J. Deore
Dean Academics/Dy. Director

Prof. Dr. J. B. Patil
Director

R. C. Patel Institute of Technology, Shirpur
SES
Autonomous

R. C. PATEL
INSTITUTE OF TECHNOLOGY
An Autonomous Institute

Department of Computer Engineering

Vision of the Institute

To become a leading Institute in Technical education fostering innovation, research, ethical values, and sustainable development for the betterment of society.

Mission of the Institute

To impart high quality Technical Education through:

M1. Innovative and Interactive learning process and high quality, globally recognized instructional programs.

M2. Fostering a collaborative scientific temper among students with ethical responsibility towards the society.

M3. Preparing students from diverse backgrounds to have aptitude for employment, entrepreneurship and research with a spirit of Professionalism.

M4. To contribute to nation's sustainable development.

Vision of the Department

To provide high quality Computer Engineering education with socio-moral values.

Mission of the Department

M1 To provide state-of-the-art ICT based teaching-learning process.

M2 To groom the students to become professionally sound computer engineers to meet emerging needs of industry and society.

M3 To make the students employable professionals by inculcating ethical values.

Program Educational Objectives (PEOs) of the Department

PEO1 Graduates will have technical proficiency, lifelong learning skills to become a professional, entrepreneur and leader.

PEO2 Graduates will function effectively in diverse cultural and professional environments, respecting societal perspectives and inclusive practices.

PEO3 To foster ethical and social values to be socially responsible human being.

R. C. PATEL
INSTITUTE OF TECHNOLOGY
An Autonomous Institute

Department of Computer Engineering

Program Specific Outcomes (PSOs)

PSO 1 Apply programming principles, algorithms, and data structures to design efficient software solutions and intelligent systems using structured, object-oriented, and emerging technologies**.**

PSO 2 Design, develop, and deploy responsive web and mobile applications integrated with databases and cloud platforms, leveraging modern frameworks and tools for digital transformation.

Semester - V

Program: Computer Engineering	T.Y. B.Tech.	Semester: V
Machine Learning(RCP23CPC301)		
Machine Learning Laboratory (RCP23CPC301L)		

Prerequisites: Data Structures, Basic Probability and Statistics, Data Mining.

Course Objectives:

1. To introduce students to the basic concepts and techniques of Machine Learning.
2. To become familiar with regression, classification, and clustering tasks.
3. To become familiar with dimensionality reduction techniques.

Course Outcomes:

On completion of the course, the learner will be able to:

COs	Course Outcomes	Blooms Level	Blooms Description
CO1	Summarize the knowledge about basic concepts of Machine Learning.	L2	Understand
CO2	Apply dimensionality reduction techniques.	L3	Apply
CO3	Apply various supervised and unsupervised machine learning algorithms.	L3	Apply
CO4	Design application using machine learning techniques.	L6	Create

Machine Learning(RCP23CPC301)
Course Contents

Unit-I Introduction to Machine Learning: 07 Hrs.

Types of Machine Learning, Steps involved in developing a Machine Learning Application, Evaluating a Learning Algorithm: Deciding what to try next, Evaluating Hypothesis, Model Selection and Train/Validation/Test Sets, Bias vs. variance: Regularization and Bias/Variance, Learning Curve, Error Analysis, Handling Skewed Data: Error Matrices for Skewed Classes, Tradeoff between Precision and recall, Issues in Machine Learning, Application of Machine Learning.

Unit-II Learning with Regression and Trees: 07 Hrs.

Learning with Regression: Simple Linear Regression, Multiple Linear Regression, Logistic Regression. Learning with Trees: Decision Trees, Constructing Decision Trees using Gini Index, Classification and Regression Trees (CART).

Unit-III Dimensionality Reduction: 07 Hrs.

Dimensionality Reduction Techniques: Principal components analysis (Eigen values, Eigen vectors, orthogonality), LDA, Singular value decomposition.

Unit-IV Classification: 08 Hrs.

Classification using Bayesian Belief networks, Hidden Markov Models Support Vector Machine: Maximum Margin Linear Separators, Quadratic Programming solution to finding maximum margin separators, Kernels for learning non-linear functions, KNN algorithm.

Unit-V Clustering: 07 Hrs.

Basics of clustering, Hard vs. Soft Clustering, Density-based Clustering: DBSCAN, Expectation maximization (EM) algorithm for soft clustering, Semi-supervised learning with EM using labelled and unlabeled data, Radial Basis functions.

Unit-VI Applications of Machine Learning: 06 Hrs.

Recommendation Systems, Machine Learning for Image Recognition, Sentiment Analysis, Machine Learning for Video Surveillance, Autonomous Vehicles, Healthcare Diagnostics, Agriculture.

Machine Learning Laboratory (RCP23CPC301L)

List of Laboratory Experiments

Suggested Experiments:(Any 08)

1. To implement Linear Regression.
2. To implement Logistic Regression.
3. To implement CART decision tree algorithm.
4. To implement Support Vector Machine.
5. To implement Bayesian Classification.
6. To implement PCA.
7. To implement K-Nearest Neighbour.
8. To implement Radial basis functions.
9. Mini project based on any machine learning application.

Any other experiment based on syllabus may be included which would help the learner to understand topic/concept.

Oral examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.

Text Books:

1. Ethem Alpaydın, "Introduction to Machine Learning", 4^{th} Edition, MIT Press, 2020.
2. Peter Harrington, "Machine Learning In Action", 1^{st} Edition, DreamTech Press, 2012.
3. Tom M. Mitchell, "Machine Learning", 1^{st} Edition, McGraw Hill, 2017.
4. Andreas C. Müller and Sarah Guido, "Introduction to Machine Learning with Python: A Guide for Data Scientists", O'reilly, 2016.
5. Kevin P. Murphy, "Probabilistic Machine Learning: An Introduction" MIT Press, 2022.
6. Wenmin Wang, "Principles of Machine Learning: The Three Perspectives" Springer, 2024.

Reference Books:

1. Aurélien Géron, "Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow", 2^{nd} Edition, 2019.

2. Witten Ian H., Eibe Frank, Mark A. Hall, and Christopher J. Pal. "Data Mining: Practical machine learning tools and techniques", 1st Edition, Morgan Kaufmann, 2016.

3. Han Kamber, "Data Mining Concepts and Techniques", 3rd Edition, Morgann Kaufmann Publishers, 2012.

4. Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar, "Foundations of Machine Learning", The MIT Press, 2012.

5. H. Dunham, "Data Mining: Introductory and Advanced Topics", 1st Edition, Pearson Education, 2006.

Online Resource:

1. NPTEL - Introduction to Machine Learning Link:https://nptel.ac.in/courses/106106139)

Program: Computer Engineering	T.Y. B.Tech.	Semester: V
Automata Theory and Compiler Design (RCP23CPC302)		
Automata Theory and Compiler Design Laboratory (RCP23CPC302L)		

Prerequisite: Algebraic Number Theory.

Course Objective(s):

1. To introduce the fundamental concepts of formal languages, grammar and automata theory.
2. Introduce the major concepts of language translation and compiler design.

Course Outcomes:

On completion of the course, the learner will be able to:

CO	Course Outcomes	Blooms Level	Blooms Description
CO1	Explain the basic properties of formal languages and differentiate between them.	L2	Understand
CO2	Design relevant automata for different classes of formal languages.	L6	Create
CO3	Analyze problems and understand the difference between decidable and undecidable problems.	L4	Analyze
CO4	Illustrate fundamental understanding of the structure of a Compiler.	L2	Understand

Automata Theory and Compiler Design (RCP23CPC302) Course Contents

Unit-I Fundamentals: 10 Hrs.

Formal Languages, Strings, Alphabets, Languages, Chomsky Hierarchy of languages.

Finite Automata: Introduction to Finite State Machine, Acceptance of strings and languages, Deterministic Finite Automata (DFA) and Non deterministic Finite Automata (NFA), Equivalence of NFA and DFA – Equivalence of NFAs with and without ϵ -moves, Minimization of Finite Automata, Equivalence between two DFA's, Finite Automata with output – Moore and Mealy machines, conversion of Moore to Mealy and Mealy to Moore.

Unit-II Regular Languages: 05 Hrs.

Regular expressions, Identity rules, Conversion of a given regular expression into a Finite Automata, Conversion of Finite Automata into a regular expression, Pumping lemma for regular sets, Closure properties of regular sets (proofs not required).

Unit-III Context Free Grammars: 07 Hrs.

Context free grammars and languages, Derivation trees, Leftmost and rightmost derivation of strings and Sentential forms, Ambiguity, left recursion and left factoring in context free grammars, Simplification of context free grammars, Normal forms for context free grammars, Chomsky normal form, Greibach normal form.

Unit-IV Push Down Automata: 06 Hrs.

Introduction to Pushdown automata, Acceptance of context free languages, Acceptance by final state and acceptance by empty state and its equivalence, Equivalence of context free grammars and pushdown automata.

Unit-V Turing Machine: 08 Hrs.

Introduction to Turing Machine, Design of Turing Machines, Types of Turing Machines. Undecidability: Recursive and recursively enumerable languages, halting problem, Post correspondence problem

Unit-VI Introduction to Compilers: 06 Hrs.

Phases of a Compiler. Lexical Analysis: The Role of Lexical Analyzer, Input Buffering, Specification of Tokens, Recognition of Tokens. Syntax Analysis: The role of the Parser, First and Follow, Predictive Parsing, LR Parsers- Intermediate Code Generation: Intermediate Languages- Graphical Representations, three address code, Implementations.

Automata Theory and Compiler Design Laboratory (RCP23CPC302L)

List of Laboratory Experiments

Suggested Experiments:(Any 08)

1. Design and implement a program to simulate a Deterministic Finite Automaton (DFA) that accepts all strings over the alphabet $\Sigma = \{a, b\}$.
2. Given a Non-deterministic Finite Automaton (NFA) with ϵ-transitions, write a program to convert it into an equivalent NFA without ϵ-transitions, then convert it into a DFA, and finally minimize the resulting DFA to its smallest possible number of states.
3. Program to generate Regular Expression for a given finite automata.
4. Program should take a CFG as input and output the equivalent, simplified grammar in Chomsky Normal Form, showing the steps of the conversion process.
5. Design a DFA that recognizes valid identifiers in the C.
6. Write a program that generates the parse tree (or derivation) for a given valid arithmetic expression string, demonstrating the structure defined by your grammar. The program should also be able to identify and reject invalid expressions.
7. To create and simulate automata using JFLAP.
8. Case Study: LLVM for machine learning inference with Google's TensorFlow.

Any other experiment based on syllabus may be included which would help the learner to understand topic/concept.

Oral examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.

Text Books:

1. John E Hopcroft, Rajeev Motwani, Jeffrey D. Ullman, "Introduction to Automata Theory, Languages and Computation", 3rd Edition, Pearson, 2008.
2. Alfred V.Aho, Monica S.Lam,Ravi Sethi, Jeffrey D. Ullman, " Compilers Principles, Techniques and Tools", 2nd Edition, Pearson, 2014.
3. M. Sipser, "Introduction to the Theory of Computation," Cengage Learning, 3rd Edition, 2012.

Reference Books:

1. Elain Rich, "Automata, Computability and complexity", 1st Edition, Pearson Education, 2

2. K.L.P Mishra, N Chandrashekaran, 'Theory of Computer Science", 3^{rd} Edition, PHI, 2012.

3. Peter Linz, "An introduction to Formal Languages and Automata", 6^{th} Edition, Narosa Publishers, 2017.

Online Resource:

1. NPTEL: Introduction to Automata, Languages and Computation
(Link: https://nptel.ac.in/courses/106105196)

Program: Computer Engineering	T.Y. B.Tech.	Semester: V
Business Intelligence and Analytics (RCP23CPC303)		
Business Intelligence and Analytics Laboratory (RCP23CPC303L)		

Prerequisites: DBMS, Data Structures, Algorithms, Statistics and Probability.

Course Objective(s):

1. To introduce the concepts, architecture, and components of Business Intelligence systems.
2. To develop understanding of data warehousing, OLAP, and data mining techniques.
3. To explore business analytics and data visualization tools for decision-making.
4. To examine advanced BI technologies and their applications in real-world domains.

Course Outcomes:

On completion of the course, the learner will be able to:

CO	Course Outcomes	Blooms Level	Blooms Description
CO1	Interpret the architecture, components, and applications of Business Intelligence systems.	L2	Understand
CO2	Design data warehouse using dimensional modelling and apply OLAP operations for any application.	L6	Create
CO3	Apply data mining and analytical techniques for business decision-making.	L3	Apply
CO4	Develop dashboards and visual analytics solutions using modern BI tools.	L6	Create

Business Intelligence and Analytics(RCP23CPC303) Course Contents

Unit-I Introduction to Business Intelligence: 05 Hrs.

Definition, need, and scope of Business Intelligence, BI vs Traditional Decision Support Systems, Framework for BI: BI Architecture and Components, Role of BI in strategic decision-making, Business analytics overview, Introduction to Big Data analytics.

Unit-II Data Warehousing and OLAP 06 Hrs.

Data Warehousing Concepts and Architecture,

Schema Design: Star, Snowflake, Fact Constellation,

ETL Process: Extraction, Transformation, and Loading,

OLAP: Different Types of OLAP,

OLAP Operations: Roll-up, Drill-down, Slice, Dice, Pivot.

Unit-III Data Mining for BI 10 Hrs.

Introduction to Data Mining and Knowledge Discovery Process,

Data Preprocessing: Cleaning, Transformation, Normalization

Association Rule Mining: Apriori, FP-Growth

Classification: Decision Tree, Naïve Bayes

Clustering: K-Means, Hierarchical

Model Evaluation and Validation: Confusion Matrix, Accuracy, Precision, Recall, F1-score, ROC Curve

Unit-IV Business Analytics 08 Hrs.

Concept and Types of Analytics: Descriptive, Predictive, Prescriptive Predictive Modeling and Forecasting Techniques: Regression, Time Series Analysis, Handling Time Series, Regression Based Forecasting: A model with Seasonality, A model with Trend and Seasonality, ARIMA models.

Prescriptive Analytics: What-if analysis, Decision Analysis with Decision Tables and Decision Trees.

Unit-V Visualization 07 Hrs.

Data Visualization: Importance, Principles, Different types of Charts and Graphs, The emergence of data visualization and Visual Analytics Performance Dashboards: Dashboard Design, Best practices, KPIs, Business Performance Management, Balanced Scorecards, Six Sigma as a Performance Measurement System: Balanced Scorecard Versus Six Sigma.

Unit-VI BI and emerging trends 06 Hrs.

Business Intelligence Applications in marketing, sales, finance, operations, Emerging Trends: Recommendation Engines, Cloud Computing and BI, Text Analytics, Web Analytics, Social Analytics.

Business Intelligence and Analytics Laboratory (RCP23CPC303L)

List of Laboratory Experiments

Suggested Experiments:(Any 08)

1. Study BI architecture and its components using case studies.
2. Design a Star Schema and a Snowflake Schema for a given business scenario.
3. Implement ETL using Pentaho or Talend on sample datasets.
4. Perform OLAP operations (roll-up, drill-down, slice, and dice) using SQL.
5. Apply association rule mining using the Apriori algorithm.
6. Implement classification algorithms (Decision Tree / Naïve Bayes) and evaluate the classifier.
7. Perform clustering using the K-Means algorithm and visualize the clusters.
8. Design an interactive dashboard using Power BI or Tableau.
9. Analyze web or social media data for trends or sentiment using Python.
10. Mini Project: Develop an end-to-end Business Intelligence solution using open datasets.

Any other experiment based on syllabus may be included which would help the learner to understand topic/concept.

Oral examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.

Text Books:

1. Ramesh Sharda, Dursun Delen, and Efraim Turban, "Business Intelligence, Analytics, and Data Science: A Managerial Perspective on Analytics", 5^{th} Edition, Pearson Education, 2023, ISBN-13: 9780137931361.
2. Paulraj Ponniah, "Data Warehousing: Fundamentals for IT Professionals", 2^{nd} Edition, Wiley India, 2013, ISBN-13: 9780470462072.

3. Jiawei Han, Micheline Kamber, and Jian Pei, "Data Mining: Concepts and Techniques", 3rd Edition, Morgan Kaufmann (Elsevier), 2011, ISBN-13: 9780123814791.

4. Galit Shmueli, Peter C. Bruce, Peter Gedeck, and Nitin R. Patel, "Data Mining for Business Analytics: Concepts, Techniques and Applications in Python", 1st Edition, Wiley India, 2019, ISBN-13: 9781119549840.

Reference Books:

1. Cindi Howson, "Successful Business Intelligence: Unlock the Value of BI and Big Data", 2nd Edition, McGraw-Hill, 2013.

2. Thomas H. Davenport and Jeanne Harris, "Competing on Analytics: Updated with New Examples", Harvard Business Review Press, 2017, ISBN-13: 9780071809184.

3. Foster Provost and Tom Fawcett, "Data Science for Business", 1st Edition, O'Reilly Media, 2013, ISBN-13: 9781449361327.

4. Anil Maheshwari, "Data Analytics Made Accessible", 2nd Edition, Amazon Digital Services, 2019, ISBN-13: 9789355324559.

5. R. N. Prasad and Seema Acharya, "Fundamentals of Business Analytics", 2nd Edition, Wiley, 2019.

6. U. Dinesh Kumar, "Business Analytics: The Science of Data-Driven Decision Making", 2nd Edition, Wiley, 2021.

7. Regi Mathew, "Business Analytics for Decision Making", 1st Edition, Pearson Education, 2020.

8. Ralph Kimball and Margy Ross, "The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling", 3rd Edition, Wiley, 2013, ISBN-13: 9781118530801.

Online Resources:

1. Regression Modeling Fundamentals https://www.coursera.org/learn/regression-modeling-sas

2. "Predictive Modeling with Logistic Regression using SAS" https://www.coursera.org/learn/sas-predictive-modeling-using-logistic-regression

Program: Computer Engineering	T.Y. B.Tech.	Semester: V
Web Programming Laboratory (RCP23CPC304L)		

Prerequisites: Object Oriented Programming using Java, HTML .

Course Objectives:

1. To get familiar with the basics of Web Programming.
2. To expose students to Basics and Advanced concepts in REACT.
3. To orient students to Fundamentals of node.js and express framework.
4. To understand REST API and MongoDB for Frontend and Backend Connectivity.

Course Outcomes:

On completion of the course, the learner will be able to:

CO	Course Outcomes	Blooms Level	Blooms Description
CO1	Design interactive web page(s) using CSS3 and JavaScript.	L6	Create
CO2	Design Single Page Application using React.js and Node.js Framework.	L6	Create
CO3	Develop web based Node.js applications using Express.	L6	Create
CO4	Apply MongoDB for frontend and backend connectivity using REST API.	L3	Apply

Web Programming Laboratory(RCP23CPC304L) Course Contents

Unit-I HTML5, CSS3, Tailwind: 02 Hrs.

HTML5: Overview, Study of AI tools assisting creation of HTML5 web pages.

CSS3: Introducing CSS3, CSS Preprocessors (SASS/SCSS), Selectors, Box Model, Border, Margin & Padding, Background Images & Colors and Other Decorative (Texts, Fonts, Links, Lists, Tables), Positioning, Combinators, Pseudo-class and Pseudo-element, 2D and 3D Transformations, Transitions and Animations, Flexbox, Media Queries.

Tailwind: Introduction, setting up Tailwind, Folder structure and build setup, utility-classes, Layout, Flex and Grid, Responsive design, customization, components and plugins.

Unit-II JavaScript: 04 Hrs.

Introduction to JavaScript, JavaScript DOM Model, var, let, const, Operators, primitive data types & strings, conditional, loop, operators, RegExp. Arrow functions, normal functions - Lexical this - Events, Handling events - Spread operator, Destructuring - named imports, default import, map, filter, reduce, Date and Objects. Call back system, Asynchronous, promises - Async, await, JSON Introduction, Syntax.

Unit-III React Fundamental: 06 Hrs.

Introduction to Vue.js vs. Angular vs. React, Installation, installing libraries, Folder and file structure, Components, Component lifecycle, Props, State, Events, React Conditional, map, keys, React Router and Single page applications, Forms, Form Handling. Refs, Use effects, useActionState, useFormStatus, and useOptimistic, Hooks, Flux.

Unit-IV Node.js 04 Hrs.

Node.js, Setup Development Environment: Installation of Node.js, Creating simple Node Server, Request and Response, Routing responses, MVC architecture in Node.js Event Loop and Emitters, File System Interaction, Modules, Native Node drivers.

Unit-V Express.js: 06 Hrs.

Introduction, Installation, Express router, REST API (CRUD Operations), Generator, Authentication, sessions, Integrating with React.

Unit-VI Database Connectivity: 06 Hrs.

MongoDB Installation, connecting to MongoDB, Frontend Integration with React, User Authentication (JWT), Role-based Access Control, connecting MongoDB to Node-RED, Hosting of web application.

List of Laboratory Experiments

Suggested Experiments:(Any 08)

1. Using HTML5 layout tags develop informative page with sections which include various images, links to other pages for navigation, make use of all possible formatting (for example font, color etc.).
2. Create form in HTML5 with all form elements. Apply form validations (e.g., Email, mobile, Pin code, Password) using JavaScript.
3. Apply CSS properties, Border, margins, Padding, Navigation, dropdown list to page created in First and Second Experiments
4. Build an interactive JavaScript application implementing CRUD logic. Use DOM manipulation, array methods (map, filter, reduce), and localStorage for data persistence.
5. Create an application to demonstrate JSX, Components, Props, State in React.
6. Create an application to demonstrate Forms, Events, Routers, Refs, Keys in React.
7. Create an application to demonstrate use of Conditional rendering in React JS.
8. Create an application to build a simple web server that serves static content. They can learn how to use the http module to create a server, and how to handle requests and responses.
9. Create an application to demonstrate the implementation of Call back system, Asynchronous, promises - Async, await in node js.
10. Create an application to demonstrate connection of Node-RED with MongoDB.
11. Build a RESTful API using MongoDB.

Mini Project: Develop website using MERN stack. Website must include home page, and at least 3 forms (with Validation), use at least HTML5, CSS/Bootstrap, JavaScript, React.js web technologies (Students can also use Node-RED to create an IOT based project). Database support is needed using RESTful API. Deploy website on live webserver and access through URL.

Practical examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.

Text Books:

1. John Dean, "Web Programming with HTML5, CSS3 and JavaScript", Jones & Bartlett Learning, 2nd Edition, 2019.
2. Glenn Johnson, "Programming in HTML5 with JavaScript and CSS3", Microsoft Press, 1st Edition, 2013.
3. Adam Bretz and Colin J. Ihrig, "Full Stack JavaScript Development with MEAN", SitePoint Pty. Ltd., 1st Edition, 2015.
4. Simon Holmes Clive Harber, "Getting MEAN with Mongo, Express, Angular, and Node", Manning Publications, 2nd Edition, 2019.
5. Venkat Subramaniam, "Rediscovering JavaScript, Master ES6, ES7, and ES8", The Pragmatic Bookshelf, 1st Edition, 2018.
6. Alex Banks and Eve Porcello, "Learning React Functional Web Development with React and Redux", O'Reilly, 2nd Edition, 2020.
7. Andrew Mead, "Learning Node.js Development", Packt Publishing, 3rd Edition, 2020.
8. Valentin Bojinov, "RESTful Web API Design with Node.js 10", Packt Publication, 3rd Edition, 2018.

Reference Books:

1. Ethan Brown, "Web Development with Node and Express", O'Reilly, 2nd Edition, 2019.
2. Shama Hoque "Full-Stack React Projects: Learn MERN stack development by building modern web apps using MongoDB, Express, React, and Node.js", Packt Publication, 2nd Edition, 2020.

Online Resources:

1. The Complete JavaScript Course 2025: From Zero to Expert! (Udemy Course).
2. React - The Complete Guide 2025 (incl. Next.js, Redux) (Udemy Course).
3. https://www.coursera.org/professional-certificates/ibm-full-stack-cloud-developer

Program: Computer Engineering	T.Y. B.Tech.	Semester: V
Advanced Algorithms(RCP23CPE311)		
Advanced Algorithms Laboratory (RCP23CPE311L)		

Prerequisites: Data structures, Analysis of Algorithms.

Course Objective: To provide conceptual and practical knowledge of Advanced Algorithms.

Course Outcomes:

On completion of the course, the learner will be able to:

COs	Course Outcomes	Blooms Level	Blooms Description
CO1	Analyze the chosen algorithm.	L4	Analyze
CO2	Choose appropriate data structure and algorithm for given problem statement.	L3	Apply
CO3	Design the algorithm.	L6	Create
CO4	Classify the algorithms based on P NP and NP hard parameters.	L2	Understand

Advanced Algorithms(RCP23CPE311)
Course Contents

Unit-I Analysis of Algorithm Based on Time 04 Hrs.

Asymptotic notations: Omega, Theta, Big-O, Small-o, small Omega and Tilde.

Amortized Analysis: Aggregate Method, Accounting Method, Potential Method, RAM model analysis of algorithm.

Unit-II Probabilistic and Randomized Algorithm 06 Hrs.

Probabilistic approach to algorithm and Randomized Analysis, Indicator Random Variable (IRV), Randomized Quick Sort, Analysis of Hiring Problem, Las Vegas and Monte Carlo algorithm,

Unit-III Advanced Data Structures 10 Hrs.

Balanced Search Trees: Red-Black Tree, Randomized BST

Heap and Operations: Binomial Tree, Binomial Heap, Treap.

Spatial Data Structure: KD Tree, R Tree.

Probabilistic Data Structure: LogLog and HyperLogLog.

Unit-IV Graph Based Algorithms 08 Hrs.

Flow Network Introduction: Residual Network, Augmenting Path, Ford-Fulkerson Method, Edmonds-Karp Method, Push-Relable Algorithm, Relable to Front algorithm.

Bipartite Matching: Maximum Bipartite Matching.

Unit-V Computational Geometry: 06 Hrs.

Line Segment Properties, Convex Hull Graham's scan algorithm.

Online Algorithms: Competitive Ratio, K-Server

Special topic: Reward-guided Tree search (e,g, reasoning with LLM)

Unit-VI Algorithm Classes: 08 Hrs.

P, NP, NP Hardness and NP Completeness Np Completeness Proofs: Satisfiability(3 sat), Reducibility, TSP. Approximation Algorithms: Vertex Cover Problem, Travelling Sales Person problem.

Network Approximation: Randomized Rounding, Primal Dual algorithms

Advanced Algorithms Laboratory (RCP23CPE311L)

List of Laboratory Experiments

Suggested Experiments:(Any 08)

1. To perform Amortized Analysis.
2. To implement Randomized Algorithms (Randomized Quick Sort).
3. To implement Randomized Algorithms (Hiring Problem).
4. To implement Advanced Data Structure (Red-black Tree Operations).
5. To implement Advanced Data Structure (KD Tree Operations).
6. To implement Graph Based Algorithms (Ford Fulkerson Method).
7. To implement Graph Based Algorithms (Push Relable Method).
8. To implement Computational Geometry (Graham Scan Algorithm).
9. To implement Online Algorithms (K-Server algorithm).
10. To implement Approximation Algorithm (Approximate TSP implementation).

Any other experiment based on syllabus may be included which would help the learner to understand topic/concept.

Oral examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.

Text Books:

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L Rivest, Clifford Stein, "Introduction to Algorithms", 4^{th} Edition,The MIT Press, 2022 .
2. S. Sridhar, "Design and analysis of algorithms", 2^{nd} Edition, Oxford University Press, 2023.
3. Horowitz, Sahani and Rajsekaran, "Fundamentals of Computer Algorithms", 2^{nd} Edition,Universities Press, 2008.
4. Harsh Bhasin, "Algorithms Design and Analysis", 1^{st} Edition, Oxford University Press, 2015.

Reference Books:

1. Rajeev Motwani, Prabhakar Raghavan, "Randomized Algorithm", 1^{st} Edition, Cambridge University,1995.
2. S. K. Basu, "Design Methods and Analysis of Algorithm", 2^{nd} Edition, PHI,2013.

3. Vijay V. Vajirani, "Approximation Algorithms", 1st Edition, Springer,2003.

4. Sanjeev Arora, Boaz Barak,"Computational Complexity", Princeton University, 2007.

Online Resources:

1. "Beyond Worst-Case Analysis",https://dl.acm.org/doi/10.1145/3232535

2. "Introduction to Approximation Algorithms", https://www.iitg.ac.in/gkd/aie/slide/approximation-algorithms.pdf

3. "Enhancing LLM Reasoning with Reward-guided Tree Search", https://arxiv.org/abs/2411.11694

Program: Computer Engineering	T.Y. B.Tech.	Semester: V
Statistical Modeling (RCP23CPE312)		
Statistical Modeling Laboratory(RCP23CPE312L)		

Prerequisites: Probability.

Course Objective(s):

1. To understand fundamental concepts of statistics, probability distributions, and hypothesis testing for analyzing engineering data.
2. To develop skills in building and validating regression models for prediction and inference in computational applications.
3. To master time series analysis techniques including ARIMA modeling, forecasting, and advanced methods for sequential data analysis.
4. To apply statistical modeling techniques using Python/R to solve real-world problems in computer engineering domains.

Course Outcomes:

On completion of the course, the learner will be able to:

CO	Course Outcomes	Blooms Level	Blooms Description
CO1	Apply descriptive and inferential statistical techniques to interpret engineering datasets.	L3	Apply
CO2	Evaluate linear and multiple regression models for prediction tasks in software and computational systems	L5	Evaluate
CO3	Design time series models (ARIMA, SARIMA) to forecast sequential data in diverse computing applications.	L6	Create
CO4	Create end-to-end statistical modeling solutions using Python/R for real-world data analysis, system monitoring, and predictive analytics.	L6	Create

Statistical Modeling (RCP23CPE312)
Course Contents

Unit-I Fundamentals of Statistics 07 Hrs.

Introduction to Statistical Analysis: Role of statistics in computer engineering and data science, Types of data: categorical, numerical, time ordered, Population vs. Sample

Descriptive Statistics: Measures of central tendency: mean, median, mode; Measures of dispersion: variance, standard deviation, IQR Statistical Inference: Confidence intervals, Hypothesis testing: t-tests (one-sample, two-sample), p-values and significance levels, Chi-square test for independence

Unit-II Correlation and Regression Models 07 Hrs.

Correlation Analysis: Pearson correlation coefficient, Spearman rank correlation Simple Linear Regression: Least squares estimation, Regression equation and interpretation, Coefficient of determination (R^2), Residual plots: normality, homoscedasticity, independence Multiple Linear Regression: Multiple regression model formulation, Matrix notation and parameter estimation, Adjusted R^2 and F-statistic Interpretation of regression coefficients, Multicollinearity: Variance Inflation Factor (VIF) and detection, Model selection criteria: Akaike Information Criterion (AIC), the Bayesian Information Criterion (BIC), and Mallows's Cp Advanced Regression Methods: Polynomial regression, Ridge (L2 norm) and LASSO (L1 norm) regression (regularization), Logistic regression for binary classification (overview).

Unit-III Introduction to Time Series and Decomposition 07 Hrs.

Time Series Fundamentals: Definition and characteristics of time series, Examples in computer engineering, Time series vs. cross-sectional data, Time series components: trend, seasonal, cyclical, irregular Classical Decomposition: Additive vs. multiplicative models, Moving average decomposition, Seasonal decomposition methods, STL decomposition.

Smoothing Techniques: Simple moving average (SMA), Weighted moving average (WMA)

Unit-IV ARIMA Modeling and Forecasting 09 Hrs.

Concept of stationarity: weak (covariance) stationarity, Importance in time series modeling, Testing for stationarity: Augmented Dickey-Fuller (ADF) test, Kwiatkowski-Phillips-Schmidt-Shin (KPSS) test.

Transformations: Logarithmic and Box-Cox transformation,

Differencing: first-order, second-order, Seasonal differencing

Autocorrelation Analysis: Autocorrelation Function (ACF), Partial Autocorrelation Function (PACF), Interpretation of ACF and PACF plots

ARMA and ARIMA Models: Autoregressive (AR) models: AR(p), properties and stationarity,

rameter estimation, Model order selection using PACF.

Moving Average (MA) models: MA(q), Properties and invertibility, Parameter estimation, Model order selection using ACF

ARMA(p,q) models: properties and identification, ARIMA(p,d,q) models for non-stationary series

Box-Jenkins Methodology: Model identification, Parameter estimation: Maximum Likelihood Estimation, Diagnostic checking: Ljung-Box test, residual analysis, Forecasting and prediction intervals

Model selection: AIC, BIC, AICc criteria

Unit-V Seasonal Time Series and Model Evaluation 06 Hrs.

Seasonal ARIMA (SARIMA): Seasonal patterns and periodicity detection, SARIMA(p,d,q)s notation, Seasonal differencing, Model identification for seasonal data, Parameter estimation Model Evaluation: Forecast accuracy measures: Mean Absolute Error (MAE), Root Mean Square Error (RMSE), Mean Absolute Percentage Error (MAPE), Mean Absolute Scaled Error (MASE), Training vs. test set evaluation, Cross-validation for time series.

Unit-VI Advanced Topics and Real-World Applications of Time Series Data 06 Hrs.

Intervention analysis and outlier detection, Multivariate time series: Vector Autoregression (VAR), Volatility modeling: ARCH/GARCH models Machine learning for time series: LSTM networks (overview), Facebook Prophet (overview), Automated ARIMA (auto.arima, pmdarima) Real-World Applications: Network traffic prediction and capacity planning, System performance monitoring and anomaly detection, Resource utilization forecasting (CPU, memory, bandwidth), Server load prediction and balancing, IoT sensor data analysis and predictive maintenance, Cloud computing: workload prediction and auto-scaling, Cybersecurity: intrusion detection patterns

Statistical Modeling Laboratory (RCP23CPE312L)

List of Laboratory Experiments

Suggested Experiments:(Any 08)

1. To compute statistical measures and create visualizations for data exploration.
    - Load dataset and compute mean, median, mode, variance, standard deviation
    - Create histograms, box plots, and scatter plots
    - Identify outliers using box plots and statistical methods
    - Generate correlation matrix and heatmap
2. To perform statistical tests and understand inference.

- Compute confidence intervals for mean
- Perform one-sample t-test
- Perform two-sample t-test (independent samples)
- Conduct chi-square test for independence
- Interpret p-values and draw conclusions

3. To analyze relationships between variables and build simple regression models.
   - Calculate Pearson and Spearman correlation coefficients
   - Fit simple linear regression model
   - Compute R^2 and interpret results
   - Create scatter plot with regression line
   - Analyze residuals
4. To build and validate multiple regression models.
   - Build multiple regression model with multiple predictors
   - Interpret regression coefficients
   - Compute adjusted R^2 and F-statistic
   - Perform residual analysis
   - Split data into training and testing sets
   - Evaluate model performance on test data
5. To check regression assumptions and detect multicollinearity.
   - Check normality of residuals (Q-Q plot, Shapiro-Wilk test)
   - Test for homoscedasticity (residual vs. fitted plot)
   - Calculate VIF to detect multicollinearity
   - Identify influential observations (Cook's distance)
   - Apply remedial measures if needed
6. To implement advanced regression techniques.
   - Fit polynomial regression for non-linear relationships
   - Implement Ridge regression
   - Implement LASSO regression
   - Compare models using cross-validation
   - Visualize results and compare performance

7. To decompose time series and apply smoothing techniques.
    - Load and visualize time series data
    - Perform seasonal decomposition (additive and multiplicative)
    - Plot trend, seasonal, and residual components
    - Implement Simple Exponential Smoothing (SES)
    - Implement Holt's double exponential smoothing
    - Implement Holt-Winters triple exponential smoothing
    - Generate short-term forecasts and evaluate
8. To test for stationarity and analyze autocorrelation structure.
    - Plot time series and check for trends/seasonality
    - Perform Augmented Dickey-Fuller (ADF) test
    - Perform KPSS test
    - Apply transformations (log, Box-Cox)
    - Apply differencing (first-order, seasonal)
    - Plot and interpret ACF and PACF
9. To build ARIMA models using Box-Jenkins methodology.
    - Identify ARIMA model order using ACF and PACF
    - Fit ARIMA model using statsmodels/pmdarima
    - Perform diagnostic checking:
        - Residual analysis
        - Ljung-Box test
        - ACF of residuals
    - Compare multiple ARIMA models using AIC/BIC
    - Generate forecasts with 95% confidence intervals
    - Calculate forecast accuracy metrics (MAE, RMSE, MAPE)
10. Complete time series analysis on real-world dataset.
    - Select dataset with seasonal patterns (e.g., network traffic, energy consumption, sales)
    - Perform complete exploratory data analysis (EDA)
    - Test for stationarity and apply necessary transformations
    - Build SARIMA model

- Compare SARIMA with other models (ARIMA, exponential smoothing)
- Generate multi-step forecasts
- Visualize actual vs. predicted values
- Prepare technical report with:
  (a) Introduction and problem statement
  (b) Data description and EDA
  (c) Methodology
  (d) Results and model comparison
  (e) Forecasts and visualizations
  (f) Conclusions and future work

Any other experiment based on syllabus may be included which would help the learner to understand topic/concept.

Oral examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.

Text Books:

1. James, G., Witten, D., Hastie, T., and Tibshirani, R., "An Introduction to Statistical Learning with Applications in R", 2^{nd} Edition, Springer, 2021.
2. Hyndman, R. J., and Athanasopoulos, G., "Forecasting: Principles and Practice", 3^{rd} Edition, OTexts, 2021.
3. VanderPlas, J., "Python Data Science Handbook", 2^{nd} Edition, O'Reilly Media, 2023.

Reference Books:

1. Shumway, R. H., and Stoffer, D. S., "Time Series Analysis and Its Applications: With R Examples", 4^{th} Edition, Springer, 2019.
2. Cryer, J. D., and Chan, K. S., "Time Series Analysis: With Applications in R", 3^{rd} Edition, Springer, 2023.
3. McKinney, W., "Python for Data Analysis", 3^{rd} Edition, O'Reilly Media, 2023.
4. Tsay, R. S., "An Introduction to Analysis of Financial Data with R", 2^{nd} Edition, Wiley, 2024.

Online Resources:

1. https://www.coursera.org/learn/practical-time-series-analysis
2. https://nptel.ac.in/courses/111105091
3. https://onlinecourses.swayam2.ac.in/ini24_cs02/preview

Program: Computer Engineering	T.Y. B.Tech.	Semester: V
Enterprise Data Systems (RCP23CPE313)		
Enterprise Data Systems Laboratory(RCP23CPE313L)		

Prerequisite: Basic knowledge of Database Management System.

Course Objective(s):

1. To provide an overview of enterprise data systems, platforms, and challenges.
2. To understand the usage of advanced data models and architectures for real life applications.
3. To impart knowledge of enterprise data security and optimization.

Course Outcomes:

On completion of the course, the learner will be able to:

COs	Course Outcomes	Blooms Level	Blooms Description
CO1	Describe enterprise data systems, platforms, and identify challenges.	L2	Understand
CO2	Explain enterprise data models, architectures and specialized databases.	L2	Understand
CO3	Design Optimize and secure enterprise data systems.	L5	Evaluate

Enterprise Data Systems (RCP23CPE313)
Course Contents

Unit-I Introduction to Enterprise Data System 06 Hrs.

Definition, components, features, factors for choosing enterprise data platforms, enterprise data platforms: NoSQL, Multimedia database, Native XML databases (NXD), Object Oriented Database, Federated Databases, Mobile databases.

Unit-II Enterprise Data Models and Architectures 08 Hrs.

Temporal data models: Aspects of valid time, transaction time, and bi temporal time with examples of each.

Spatial model: Types of spatial data models - Raster, Vector and Image

Graph Database: Introduction, Graph database architecture, Types of graph database, Graph vs Relational database, Data modeling with graph, Neo4j.

Distributed Database: Introduction, Fragmentation and its types.

Unit-III Document Oriented Database 06 Hrs.

Need of Document Oriented database, difference between Document Oriented Database and Traditional database, Types of encoding XML, JSON, BSON, Representation XML, JSON Objects. Case study on document oriented based such as MongoDB.

Unit-IV Optimization 08 Hrs.

Query Optimization: Overview, Measures of Query cost, Selection operation, Sorting and Join Operations, Evaluation of expression query, Translations of SQL Queries into relational algebra, Heuristic approach and cost-based optimization.

Unit-V Data Security 08 Hrs.

Introduction to Database Security Issues; Authentication and authorization, Database auditing, Discretionary Access Control Based on Granting and Revoking Privileges, Mandatory Access Control and Role Based Access Control for Multilevel Security Introduction to Statistical Database Security.

Unit-VI Challenges in Enterprise Data System 06 Hrs.

Data fragmentation, data quality issues, complex data integration, governance and compliance, scaling with data growth, change management and user adoption, Resource constraints.

Enterprise Data Systems Laboratory (RCP23CPE313L)

List of Laboratory Experiments

Suggested Experiments:(Any 08)

1. Perform a case study on enterprise database system: Summary and Comparison
   - Consider any existing enterprise database system.
   - Identify the strengths and weaknesses.
   - Recommend changes and/or enhancements.
2. Query execution on the XML database.
   Use any XML editor (eg XMLSPY) to show query execution in XML.
3. Implement an enterprise data system based on spatial and temporal models.
4. Data handling using JSON. Using JSON, exhibit how data can be stored and queried.
5. Implementation of simple graph database using Neo4j. Create nodes, properties, and relationships.
6. Perform Fragmentation in DDBS design. Design a distributed database for a case study of your choice. Apply fragmentation. (Range, List, Hash and Key)
7. Implementation of simple document-oriented database using MongoDB. Store data using MongoDB for any appropriate case study which uses a document-oriented database.
8. Implementation of Query monitor (QEP – Query Execution Plan, Query Statistics). In MySQL, use the Query monitor to understand the query execution plan and monitor the query statistics.
9. Simulate Query optimization by applying an SQL Query on any database. Apply the optimization techniques for query processing to select the query with the least processing time.
10. Optimization using B/B^+ Tree. Write a program in C++/Java/python to analyze the impact of B/B^+ tree in searching records.
11. Case study on challenges in enterprise data systems.
    - Study a research paper / patent / product.
    - Identify gaps.
    - Suggest Improvement/s.
    - Draft & apply for journal / conference / book / book-chapter / magazine / patent / copyright.

Any other experiment based on syllabus may be included which would help the learner to understand topic/concept.

Oral examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.

Text Books:

1. Ole Olesen-Bagneur, "Enterprise Data Catalog", 1st Edition, O'Reilly Publication, 2023.
2. Abraham Silberschatz, Henry F. Korth, and Sudarshan, "Database System Concepts", 7th Edition, McGraw Hill, 2021.
3. Shannon Bradshaw and Eoin Brazil, "MongoDB: The Definitive Guide – Powerful and Scalable Data Storage", 3rd Edition, O'Reilly Publication, 2020.
4. Christos Tjortjis, "Graph Databases Applications on Social Media Analytics and Smart Cities", 1st Edition, CRC Press, 2023.

Reference Books:

1. Vinicius M. Grippa and Sergey Kuzmichev, "Learning MySQL", 2nd Edition, O'Reilly Publication, 2021.
2. Tamer OEzsu and Patrick V, "Principles of Distributed Database System", Springer Publication, 2020.

Online Resources:

1. Neo4j documentation https://neo4j.com/docs/

Program: Computer Engineering	T.Y. B.Tech.	Semester: V
Computer Graphics (RCP23CPE314)		
Computer Graphics Laboratory (RCP23CPE314L)		

Prerequisites: C Programming.

Course Objective(s):

1. To equip students with the fundamental knowledge of computer graphics and provide an understanding of how to scan convert the basic geometrical primitives, how to transform the shapes to fit them as per the picture definition.
2. To provide an understanding of mapping from a world coordinate to device coordinates, clipping, solid modeling, rendering, and projections.

Course Outcomes:

On completion of the course, the learner will be able to:

COs	Course Outcomes	Blooms Level	Blooms Description
CO1	Interpret the fundamental knowledge of computer graphics applications of Computer Graphics.	L2	Understand
CO2	Apply and compare the algorithms for drawing 2D images and explain aliasing, anti-aliasing and half toning techniques.	L3	Apply
CO3	Analyze and apply clipping algorithms and transformation on 2D images.	L4	Analyze
CO4	Describe the basics of shading, shadows, curves and surfaces and also solve the problems of curves.	L2	Understand
CO5	Apply the fundamental OpenGL concepts—including its features, abstractions, and 3D viewing pipeline—to develop basic interactive graphics programs.	L3	Apply

Computer Graphics (RCP23CPE314)
Course Contents

Unit-I Introduction to Computer Graphics: 06 Hrs.

Overview of Computer Graphics, Computer Graphics Application and Software, Description of some graphics devices, Input Devices for Operator Interaction, Active and Passive Graphics Devices, Display Technologies, Storage Tube Graphics Displays, Calligraphic Refresh Graphics Displays, Raster Refresh (Raster-Scan) Graphics Displays, Cathode Ray Tube Basics, Color CRT Raster Scan Basics, Video Basics, The Video Controller, Random Scan Display Processor, LCD displays.

Unit-II Scan conversion: lines, circles and Ellipses and Filling polygons: 08 Hrs.

Scan Converting Lines, Mid-point criteria, Problems of Aliasing, Scan Converting Circles, Scan Converting Ellipses, Filling Polygons, edge data structure.

Unit-III Two-Dimensional and Three Dimensional Transformations: 07 Hrs.

Transformations and Matrices, Transformation Conventions, 2D Transformations, Homogeneous Coordinates and Matrix Representation of 2D Transformations, Translations and Homogeneous Coordinates, Rotation, Reflection, Scaling, Combined Transformation, Transformation of Points, Transformation of The Unit Square, Solid Body Transformations, Rotation About an Arbitrary Point, Reflection through an Arbitrary Line, A Geometric Interpretation of Homogeneous Coordinates.

Three-Dimensional Transformations: Scaling, Shearing, Rotation, Reflection, Translation, Multiple Transformation, Rotation about an Arbitrary Axis in Space, Reflection through an Arbitrary Plane, Matrix Representation of 3D Transformations, Composition of 3D Transformations, Affine and Perspective Geometry, Perspective Transformations, Techniques for Generating Perspective Views, Vanishing Points, the Perspective Geometry and camera models, Orthographic Projections, Axonometric Projections, Oblique Projections

Unit-IV Two-Dimensional Viewing: 08 Hrs.

Introduction, Viewing Pipeline View Coordinate reference frame, Window to viewport transformation point clipping, Text Clipping, Line Clipping: Cohen Sutherland Algorithm, Liang Barsky algorithms, Polygon clipping: Sutherland Hodgeman polygon clipping and Weiler Atherton. Fractal Geometry: Fractal Dimension, Koch Curve. Piano Curve, Hilbert Curve.

Unit-V Visible-Surface Determination: 08 Hrs.

Techniques for efficient Visible-Surface Algorithms, Categories of algorithms, Back face removal,

z-Buffer Algorithm, Scan-line method, Painter's algorithms (depth sorting), Area sub-division method, BSP trees, Visible-Surface Ray Tracing, comparison of the methods. Illumination and Shading Illumination and Shading Models for Polygons, Reflectance properties of surfaces, Ambient, Specular and Diffuse reflections, Atmospheric attenutation, Phong's model, Gouraud shading, some examples.

Unit-VI Graphics Programming using OPENGL: 05 Hrs.

Why OpenGL, Features in OpenGL, OpenGL operations, Abstractions in OpenGL – GL, GLU & GLUT, 3D viewing pipeline, viewing matrix specifications, a few examples and demos of OpenGL programs.

Computer Graphics Laboratory(RCP23CPE314L)

List of Laboratory Experiments

Suggested Experiments:(Any 08)

1. Implementation of Line Drawing algorithms: DDA, Bresenham's and using them generating line with different styles like dotted, dashed, centered and thick line.
2. Implementation of Circle generation algorithm: Midpoint and using it generating concentric circles.
3. Implementation of Area Filling Algorithm: Boundary Fill, Flood Fill and Scan line, Polygon Fill.
4. Curve Generation: Bezier for n control points, B Spline (Uniform), Fractal Generation (Koch Curve).
5. Program for performing Two Dimensional Transformations: Translation, Scaling, Rotation, Reflection, Shear by using a homogeneous Matrix representation, use of a function for matrix multiplication is desirable, to perform composite transformation.
6. Implementation of Line Clipping Algorithm: Cohen Sutherland, Liang Barsky.
7. Implementation of Polygon Clipping Algorithm: Sutherland Hodgman.
8. Program to represent a 3D object using polygon surfaces and then perform 3D transformation.
9. Program to perform projection of a 3D object on Projection Plane: Parallel and Perspective.
10. Implement Illumination and shading apply on sphere using two light sources in OpenGL.

Any other experiment based on syllabus may be included which would help the learner to understand topic/concept.

Oral examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.

Text Books:

1. A. P. Godse and Dr. D. A. Godse, "Computer Graphics and Multimedia, Concepts, Algorithms and Implementation using C", KDP Print US, 2020.

2. Donald Hearn, M. Pauline Baker, and Warren Carithers, "Computer Graphics with OpenGL", 4^{th} Edition, Pearson Education, 2024.

Reference Books:

1. Jun Mitani, Kenshi Takayama, Yoshinori Dobashi, Tomohiko Mukai, and Makoto Fujisawa, "Mathematics and Applications of Computer Graphics", 1^{st} Edition, Springer Singapore, 2025.

2. Computer Graphics and Multimedia: Concepts, Algorithms and Implementation using C, Technical Publications, 2020.

3. William M. Newman and Robert F. Sproull, "Principles of Interactive Computer Graphics", McGraw Hill, 2001.

4. B. M. Havaldar ,"C Graphics and Projects", 2006.

Online Resources:

1. https://onlinecourses.nptel.ac.in/noc20_cs90/preview

2. https://www.edx.org/learn/computer-graphics/the-university-of-california-san-diego-computer-graphics

Program: Computer Engineering	T.Y. B.Tech.	Semester: V
Information and Cyber Security (RCP23CMD301)		
Information and Cyber Security Laboratory (RCP23CMD301L)		

Prerequisites: Computer Networks, Database Management System, Algebraic Number Theory.

Course Objective(s):

1. Understand the fundamental concepts of cybersecurity, including security principles, classical cryptography, and modern symmetric and asymmetric encryption techniques.
2. Explain authentication, integrity, and secure communication mechanisms such as digital signatures, PKI, hashing, SSL/TLS, and network defense tools.
3. Identify and analyze system, network, and application-level vulnerabilities and attacks, along with corresponding defense techniques and secure software practices.
4. Gain awareness of modern cybersecurity domains, including malware analysis, system hardening, security operations (SOC), incident response, and AI-enabled security mechanisms.

Course Outcomes:

On completion of the course, the learner will be able to:

CO	Course Outcomes	Blooms Level	Blooms Description
CO1	Apply cryptographic techniques to achieve data confidentiality, integrity, and authentication.	L3	Apply
CO2	Evaluate systems and networks against security threats and attacks using appropriate defense mechanisms.	L5	Evaluate
CO3	Analyze software and web applications to identify vulnerabilities and recommend appropriate secure development practices.	L4	Analyze
CO4	Explain security governance, system hardening, incident response, SOC operations, and legal and regulatory compliance.	L2	Understand

Information and Cyber Security (RCP23CMD301) Course Contents

Unit-I Cyber Security Essentials and Classical Cryptography: 06 Hrs.

CIA Triad, security goals, threats, vulnerabilities, attacks, Security services & mechanisms, OSI security architecture, network security model, Classical encryption: substitution (Caesar, Vigenère) and transposition ciphers, Security models: Bell-LaPadula, Biba, Zero Trust Model, Risk management & threat intelligence, Malware types: Viruses, Worms, Trojans, Ransomware, Logic Bombs, Rootkits, Bots, Social Engineering Attacks: Phishing, Spear Phishing, Baiting, Tailgating.

Unit-II Symmetric Cryptography & Secure Communication: 08 Hrs.

Symmetric encryption basics, block vs stream ciphers, DES, Double DES, Triple DES, AES, Simplified AES, Block cipher modes: ECB, CBC, CFB, OFB, CTR, Applications of symmetric cryptography in SSL/TLS VPNs, disk encryption.

Unit-III Asymmetric Cryptography, Digital Signatures & PKI: 08 Hrs.

Symmetric vs asymmetric encryption, RSA algorithm (conceptual without number theory), Diffie–Hellman key exchange, Elliptic Curve Cryptography (ECC – concept only), ECC-based DH, Digital signatures: RSA, ElGamal, DSA, Public Key Infrastructure (PKI), X.509 digital certificates, HTTPS, SSL/TLS handshake overview, Introduction to quantum-safe cryptography.

Unit-IV Integrity, Authentication & Network Security Basics: 07 Hrs.

Cryptographic hash functions: MD5, SHA-1, SHA-256, SHA-3, Message Authentication Codes: MAC, HMAC, CMAC, User & entity authentication: passwords, biometrics, multi-factor authentication, Authentication protocols: Needham–Schroeder, Kerberos, Network attacks: ARP spoofing, IP spoofing, packet sniffing, DNS spoofing, TCP SYN flood, Firewalls, IDS, IPS, VPN basics, Wi-Fi security: WEP, WPA2, WPA3.

Unit-V Application, Web & Software Security: 06 Hrs.

Secure OS design, access controls: DAC, MAC, RBAC, Software & web vulnerabilities: Buffer overflow, SQL injection, XSS, CSRF, Format string attacks, OWASP Top 10 (latest list), Secure Soft Development Life Cycle (SSDLC), Introduction to DevSecOps and security in CI/CD pipelines, C

tainer security basics (Docker, Kubernetes).

Unit-VI Security Operations, Compliance, and Quantum Cryptography: 07 Hrs.

Basic System Hardening Techniques: Patch management, secure configuration, password policies, Security Policies and Compliance (GDPR, DPDPA 2023 - basic awareness), Security Operations Center (SOC) – Introduction, Incident Response: Preparation, Detection, Containment, Recovery (NIST model), Basics of SIEM (Security Information and Event Management), Introduction to AI in security (SOC automation, threat detection), Introduction to Quantum Cryptography: Basics of quantum computing threats, Quantum Key Distribution (QKD), post-quantum cryptography fundamentals.

Information and Cyber Security Laboratory (RCP23CMD301L)

List of Laboratory Experiments

Suggested Experiments:(Any 8)

1. Implement Caesar Cipher (encryption & decryption)
2. Implement Vigenère Cipher & Transposition Cipher
3. Implement DES and AES Encryption (using any programming language or tool like Cryptool/OpenSSL)
4. Implement RSA Algorithm (key generation, encryption, decryption – conceptual coding without number theory)
5. Implement Diffie–Hellman Key Exchange (simulate shared key generation)
6. Implement Digital Signature (using RSA or DSA) and verify integrity
7. Demonstrate Hashing Algorithms (MD5, SHA-1, SHA-256) and HMAC verification
8. Packet sniffing and protocol analysis using Wireshark (capture HTTP/HTTPS, ARP, DNS traffic)
9. Simulate ARP Spoofing or MITM attack using tools like Ettercap/Scapy and analyze using Wireshark
10. Demonstrate Firewall configuration / Simple IDS using Snort (detect port scan or ping sweep)
11. Demonstrate SQL Injection attack and its prevention
12. Demonstrate XSS (Cross-Site Scripting) or CSRF attack on a test web page

13. Buffer Overflow attack simulation in C/Python and mitigation using safe functions / DEP

14. Study of Malware & Incident Response: Analyze a malware sample (safe simulation), observe SOC tools/SIEM logs, prepare an incident response report (NIST phases)

Any other experiment based on syllabus may be included which would help the learner to understand topic/concept.

Oral examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.

Text Books:

1. William Stallings, "Network Security Essentials: Applications and Standards", 6^{th} ed., Boston, MA: Pearson, 2023.

2. B. A. Forouzan and D. Mukhopadhyay, "Cryptography and Network Security", 3^{rd} ed., McGraw Hill, 2021.

3. Michael E. Whitman and Herbert J. Mattord, "Principles of Information Security", 7^{th} ed., Boston, MA: Cengage, 2024.

Reference Books:

1. C. Kaufman, R. Perlman, and M. Speciner, "Network Security: Private Communication in a Public World", 3^{rd} Edition, Pearson, 2022.

2. Mike Chapple, J. M. Stewart and D. Gibson, "$(ISC)^2$ CISSP Certified Information Systems Security Professional Official Study Guide", 9^{th} Edition, Indianapolis, IN: Sybex, 2024.

3. Ross Anderson, "Security Engineering: A Guide to Building Dependable Distributed Systems", 3^{rd} Edition, Wiley, 2021.

4. M. Sikorski and A. Honig, "Practical Malware Analysis: The Hands-On Guide to Dissecting Malicious Software", No Starch Press, 2012.

Program: Computer Engineering	T.Y. B.Tech.	Semester: V
Semester Project - III (RCP23XSC301P)		

Course Objective:

Students are expected to design, simulate/implement a project based on the knowledge acquired from current semester subjects.

Course Outcomes:

On completion of the course, the learner will be able to:

COs	Course Outcomes	Blooms Level	Blooms Description
CO1	Analyze a survey of several available literatures in the preferred field of study.	L4	Analyze
CO2	Describe various/alternate approaches to complete a project.	L2	Understand
CO3	Apply a collaborative project environment by interacting and dividing project work among team members.	L3	Apply
CO4	Use technical communication skills to present project work in the form of a technical report/paper.	L3	Apply
CO5	Apply teamwork and project management skills to plan, execute, and manage the research study.	L3	Apply

Semester Project:

The purpose of semester project is to provide exposure to students with a variety of projects based on the knowledge acquired from the semester subjects. This activity is supposed to enrich their academic experience and bring enough maturity in student while selecting the project. Students should take this as an opportunity to develop skills in implementation, presentation and discussion of technical ideas/topics. Therefore, proper attention shall be paid to the content of semester project report which is being submitted in partial fulfillment of the requirements of the Second Year and it is imperative that a standard format be prescribed for the report.

Each student shall work on project approved by departmental committee approved by the Head of Department, a group of 03 to 05 students (max allowed: 5 students in extraordinary cases, subject to the approval of the department committee and the Head of the department) shall be allotted for each Semester Project. Each group shall submit at least 3 topics for the Semester Project. The departmental committee shall finalize one topic for every group. Semester Project Title or Theme should be based on knowledge acquired during semester. The project work shall involve sufficient work so that students get acquainted with different aspects of knowledge acquired from semester subjects.

Student is expected to:

- Select appropriate project title based on acquired knowledge from current semester subjects.
- Maintain Log Book of weekly work done(Log Book Format will be as per Table 1).
- Report weekly to the project guide along with log book.

Assessment Criteria:

- At the end of the semester, after confirmation by the project guide, each project group will submit project completion report in prescribed format for assessment to the departmental committee (including project guide).
- Assessment of the project (at the end of the semester) will be done by the departmental committee (including project guide).

Prescribed project report guidelines:

Size of report shall be of minimum 25 pages. Project Report should include appropriate content for:

- Introduction
- Literature Survey
- Related Theory
- Implementation details

- Project Outcomes
- Conclusion
- References

Assessment criteria for the departmental committee (including project guide) for Continuous Assessment:

Guide will monitor weekly progress and marks allocation will be as per Table 2.

Assessment criteria for the departmental committee (including project guide) for End Semester Exam:

Departmental committee (including project guide) will evaluate project as per Table 3.

Table 1: Log Book Format

Sr	Week (Start Date:End Date)	Work Done	Sign of Guide	Sign of Coordinator
1				
2				

Table 2: Continuous Assessment Table

Sr	Exam Seat No	Name of Student	Student Attendance	Log Book Maintenance	Literature Review	Depth of Understanding	Report	Total
			5	5	5	5	5	25

Table 3: Evaluation Table

Sr	Exam Seat No	Name of Student	Project Selection	Design/ Simulation/ Logic	Hardware/ Programming	Result Verification	Presentation	Total
			5	5	5	5	5	25

Program: Computer Engineering	T.Y. B.Tech.	Semester: V
Constitution of India (RCP23XHS311)		

Course Objective(s):

1. To provide basic information about Indian constitution.
2. To identify individual role and ethical responsibility towards society.
3. To understand human rights and its implications.

Course Outcomes:

On completion of the course, the learner will be able to:

CO	Course Outcomes	Blooms Level	Blooms Description
CO1	Use general knowledge and legal literacy and thereby to take up competitive examinations.	L3	Apply
CO2	Explain state and central policies, fundamental duties.	L2	Understand
CO3	Discuss Electoral Process, special provisions.	L2	Understand
CO4	Explain powers and functions of Municipalities, Panchayats and Co-operative Societies.	L2	Understand
CO5	Discuss Engineering ethics and responsibilities of Engineers.	L2	Understand
CO6	Explain Engineering Integrity & Reliability.	L2	Understand

Constitution of India (RCP23XHS311)
Course Contents

Unit-I Introduction to the Constitution of India 02 Hrs.

The Making of the Constitution and Salient features of the Constitution, Preamble to the Indian Constitution, Fundamental Rights & its limitations.

Unit-II Directive Principles of State Policy 02 Hrs.

Relevance of Directive Principles State Policy, Fundamental Duties.

Union Executives – President, Prime Minister, Parliament, Supreme Court of India.

Unit-III State Executives 03 Hrs.

Governor, Chief Minister, State Legislature, High Court of State.

Electoral Process in India, Amendment Procedures, 42^{nd}, 44^{th}, 74^{th}, 76^{th}, 86^{th} & 91^{st} Amendments.

Unit-IV Special Provisions 02 Hrs.

For SC & ST, Special Provision for Women, Children & Backward Classes, Emergency Provisions.

Unit-V Human Rights 03 Hrs.

Meaning and Definitions, Legislation Specific Themes in Human Rights- Working of National Human Rights Commission in India, Powers and functions of Municipalities, Panchayats and Co–Operative Societies.

Unit-VI Scope & Aims of Engineering Ethics 02 Hrs.

Responsibility of Engineers Impediments to Responsibility. Risks, Safety and liability of Engineers, Honesty, Integrity & Reliability in Engineering.

Text Books:

1. Durga Das Basu, "Introduction to the Constitution on India", (Student Edition) Prentice-Hall EEE, 25^{th} Edition, 2021.
2. Charles E. Haries, Michael S. Pritchard and Michael J. Robins, "Engineering Ethics", Thompson Asia, 2003.

Reference Books:

1. M. V. Pylee, "An Introduction to Constitution of India", Vikas Publishing, 3rd Edition, 2003.
2. M. Govindarajan, S. Natarajan, V. S. Senthilkumar, "Engineering Ethics", Prentice-Hall of India Pvt. Ltd. New Delhi, 2013.
3. Brij Kishore Sharma, pl"Introduction to the Constitution of India", 7th Edition, PHI Learning Pvt. Ltd., New Delhi, 2015.
4. Latest Publications of Indian Institute of Human Rights, New Delhi.

Online Resources:

1. www.nptel.ac.in
2. www.hnlu.ac.in
3. www.nspe.org
4. www.preservearticles.com

Program: Computer Engineering	T.Y. B.Tech.	Semester: V
Mastering Neural Networks (RCP23CSC301)		

Prerequisites: Python Programming, and Engineering Mathematics (Linear Algebra and Calculus).

Course Objective(s):

1. To build a deep understanding of neural network architecture, from the perceptron model to back-propagation and training from scratch.
2. To develop the ability to apply regularization techniques that improve model generalization and reduce overfitting.
3. To enable practical implementation of deep learning and convolutional neural networks for image and audio tasks using PyTorch.

Course Outcomes:

On completion of the course, the learner will be able to:

CO	Course Outcomes	Blooms Level	Blooms Description
CO1	Build multilayer neural networks from scratch using forward and back-propagation.	L6	Create
CO2	Apply L1, L2, dropout, and decision tree pruning techniques to reduce model overfitting.	L3	Apply
CO3	Implement deep learning models with PyTorch using tensors, computational graphs, and GPU acceleration.	L3	Apply
CO4	Develop convolutional neural networks for image and audio classification tasks.	L6	Create
CO5	Analyze neural network performance across different architectures and regularization strategies.	L4	Analyze

Mastering Neural Networks (RCP23CSC301)
Course Contents

Unit-I 04 Hrs.

Multilayer Artificial Neural Networks:

Introduction, perceptron model and model parameters, neural networks fundamentals, forward propagation algorithm, back-propagation, neural network design and implementation; building a neural network from scratch and classifying MNIST images, evaluated against sklearn.

Unit-II 04 Hrs.

Model Regularization:

The need for regularization to mitigate overfitting, L1 vs. L2 regularization, decision tree pruning, dropout regularization in neural networks, and how regularization shapes model decision boundaries; cyber intrusion detection practice lab.

Unit-III 06 Hrs.

Deep Learning with PyTorch:

Computational graphs in supervised learning, creating and manipulating tensors in PyTorch, activation and loss functions, neural network training steps and library functions, and GPU-accelerated model optimization; fraud detection lab with PyTorch and Scikit-Learn.

Unit-IV 06 Hrs.

Convolutional Neural Networks:

Implementing CNN filters, comparing CNN architectures, developing image-processing models in PyTorch, and modeling audio time signals using Spectrogram features for classification; image classification capstone using PyTorch.

Web Resource:

https://www.coursera.org/learn/mastering-neural-networks-and-model-regularization

Semester - VI

Program: Computer Engineering	T.Y. B.Tech.	Semester: VI
Deep Learning (RCP23CPC351)		
Deep Learning Laboratory (RCP23CPC351L)		

Prerequisites: Artificial Intelligence, Machine Learning.

Course Objective(s):

1. To understand Hyper parameter Tuning.
2. To explore Deep Learning Techniques with different learning strategies.
3. To design Deep Learning Models for real time applications.

Course Outcomes:

On completion of the course, the learner will be able to:

CO	Course Outcomes	Blooms Level	Blooms Description
CO1	Discuss the fundamentals and tools of Deep Learning.	L2	Understand
CO2	Apply Hyper-parameter Tuning techniques.	L3	Apply
CO3	Illustrate and analyze Convolutional and Sequential Models.	L4	Analyze
CO4	Select and evaluate Generative and Adversarial Networks.	L5	Evaluate
CO5	Use suitable Deep Learning models for real-world applications.	L3	Apply

Deep Learning (RCP23CPC351)
Course Contents

Unit I Introduction to Deep Learning: 04 Hrs.

Overview of Neural Network, Deep learning and human brain, Why is Deep Learning taking off?, Deep Learning applications.

Overview of Tools: Torch, TensorFlow, Keras.

Unit II Convolutional Neural Network: 09 Hrs.

Introduction to CNNs, Kernel filter, Principles behind CNNs, Multiple Filters, CNN applications. ConvNet Architectures:

Discussions on famous convnet architectures: AlexNet, VGG, GoogLeNet, ResNet.

Unit III Hyperparameter Tuning, Batch Normalization: 06 Hrs.

Tuning Process, Using an Appropriate Scale to pick Hyperparameters, Hyperparameters Tuning in Practice: Pandas vs. Caviar, Normalizing Activations in a Network, Fitting Batch Norm into a Neural Network, why does Batch Norm work, Batch Norm at Test Time.

Unit IV Sequential Models: 09 Hrs.

Introduction to Sequence Models and RNNs, Recurrent Neural Network Model, Backpropagation Through Time, Different Types of RNNs: Unfolded RNNs, Seq2Seq RNNs, Long Short-Term Memory (LSTM), Bidirectional RNN, Vanishing Gradients with RNNs, Gated Recurrent Unit (GRU), RNN applications, Sequence models and attention mechanism, Attention over Images, Hierarchical Attention.

Unit V Adversarial Networks: 09 Hrs.

Introduction to adversarial Networks, Auto encoders (standard, denoising, contractive, etc.), Variational Auto encoders, Generative Adversarial Networks, Diffusion Models: Basics of Diffusion models, Advantages of over GANs, Popular Diffusion tools: Stable Diffusion, DALL-E, Midjourney, Applications of Adversarial Networks.

Unit VI Deep Learning Case Studies: 05 Hrs.

Image Processing, Natural Language Processing, Speech Recognition, Video Analytics.

Deep Learning Laboratory (RCP23CPC351L)

List of Laboratory Experiments

Suggested Experiments:

1. To implement backpropagation algorithm.
2. Understanding ANN using Pytorch/Tensor Flow.
3. Visualizing Convolutional Neural Network using Pytorch/Tensor Flow.
4. Object detection using RNN using Pytorch/Tensor Flow.
5. Handwritten Digit Generation using GANs.
6. Students are supposed to complete any one mini project not limited to following list of projects.
   (a) Sequence Prediction
   (b) Object Detection
   (c) Traffic Sign Classification
   (d) Automatic Music Generation
   (e) Music Genre Classification
   (f) Text Summarizer
   (g) Gender and Age Detection Using Voice
   (h) Chatbot Using Deep Learning
   (i) Neural Style Transfer
   (j) Face Aging
   (k) Driver Drowsiness Detection
   (l) Language Translator
   (m) Image Reconstruction

Any other practical covering the syllabus topics and subtopics can be conducted.

Oral examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.

Text Books:

1. C. Aggarwal, "Neural Networks and Deep Learning: A Textbook", 2nd ed., Springer, 2023.
2. Goodfellow I., Bengio Y., and Courville A., "Deep Learning", MIT Press, 2016.
3. Umberto Michelucci, "Advanced Applied Deep Learning: Convolutional Neural Networks and Object Detection", 2019.
4. Michael Nielsen (Goodreads Author), "Neural Networks and Deep Learning", 2015.

Reference Books:

1. F. Chollet and M. Watson, "Deep Learning with Python", 3rd Edition, Manning Publications, 2025.
2. B. Yegnanarayana, "Artificial Neural Networks", PHI Learning Pvt. Ltd., 2009.
3. Satish Kumar, "Neural Networks: A Classroom Approach", Tata McGraw-Hill Education, 2004.
4. David Foster, "Generative Deep Learning: Teaching Machines to Paint, Write, Compose, and Play", 2nd Edition, O'Reilly, 2019.
5. Maxim Lapan, "Deep Reinforcement Learning Hands-On: Apply Modern RL Methods with Deep Q-Networks, Value Iteration, Policy Gradients, TRPO, AlphaGo Zero and More", 3rd Edition, O'Reilly, 2020.
6. Gulli and Kapoor, "TensorFlow 1.x Deep Learning Cookbook", Packt, 2017.

Online Resources:

1. NPTEL: *Deep Learning*, By Prof. Prabir Kumar Biswas, IIT Kharagpur.
   Available at: https://onlinecourses.nptel.ac.in/noc22_cs22/preview
2. Coursera: *Deep Learning Specialization*, By DeepLearning.AI.
   Available at: https://www.coursera.org/specializations/deep-learningcourses

Program: Computer Engineering	T.Y. B.Tech.	Semester: VI
Blockchain and Crypto Technology (RCP23CPC352)		
Blockchain and Crypto Technology Laboratory (RCP23CPC352L)		

Prerequisites: Computer Networks, Data Structures, Information and Cyber Security.

Course Objective(s):

1. To introduce Blockchain architecture, distributed ledgers, cryptographic foundations, and consensus mechanisms.
2. To expose learners to public Blockchain systems such as Bitcoin and Ethereum, including smart contracts and Web3 concepts.
3. To develop practical skills for designing and deploying smart contracts, tokens, and decentralized applications.
4. To provide knowledge of crypto systems including token standards, stablecoins, NFTs, DAOs, DeFi protocols, and enterprise Blockchain platforms.
5. To create awareness of security threats and best practices in Blockchain, smart contract development, and wallet security.

Course Outcomes:

On completion of the course, the learner will be able to:

COs	Course Outcomes	Blooms Level	Blooms Description
CO1	Explore Blockchain architecture, cryptographic foundations, and consensus mechanisms.	L2	Understand
CO2	Analyze public Blockchain systems (Bitcoin, Ethereum), smart contracts, and Web3 applications.	L4	Analyze
CO3	Develop basic smart contracts, crypto tokens, and decentralized applications.	L6	Create
CO4	Evaluate security vulnerabilities in Blockchain systems and compare public and permissioned Blockchain platforms.	L5	Evaluate

Blockchain and Crypto Technology (RCP23CPC352) Course Contents

Unit-I Blockchain & Crypto Fundamentals: 08 Hrs.

Evolution of Web1 → Web2 → Web3, Introduction to Distributed Ledger Technology (DLT), Blockchain architecture: blocks, hash pointers, Merkle trees, Peer-to-peer network and nodes, Cryptographic foundations: Hashing (SHA-256, Keccak-256), Public-key cryptography, ECDSA, Digital signatures, Consensus concepts (overview): PoW, PoS, Crypto basics: Coins vs Tokens, UTXO vs Account-based model, Tokenomics basics: supply models, inflation & deflation.

Unit-II Public Blockchain Systems & Smart Contracts: 12 Hrs.

Bitcoin Blockchain: Motivation for Bitcoin, UTXO model and transaction structure, Mempool, confirmations, and fee mechanism, Mining and difficulty adjustment, Block structure and propagation.

Ethereum Blockchain: Transition from PoW to PoS, Ethereum architecture: EVM, accounts, state, Transactions, gas mechanism, Merkle Patricia Tree – conceptual overview

Smart Contracts & Web3: Smart contracts: concepts & characteristics, Solidity basics: syntax, functions, events, Contract deployment using Remix, Oracles: types and role, Web3 workflow: Wallet → Contract → Blockchain, Basic DApp architecture and interaction

Layer-2 Scaling (Overview): Rollups: Optimistic & ZK, Sidechains vs Layer-2

Unit-III Crypto Systems & Web3 Applications 08 Hrs.

Crypto Ecosystem: Token standards: ERC-20, ERC-721, ERC-1155, Stablecoins (USDT, USDC, DAI), DAOs: governance models and examples, Decentralized identity (DID) – basics, Real-World Assets (RWAs)

Decentralized Finance (DeFi): Introduction and motivation, AMMs: Uniswap model (concept), Liquidity pools & impermanent loss (high level), Lending and borrowing (Aave model – overview), Real-World Assets in DeFi

Web3 Applications: NFTs: architecture, storage, use cases, Tokenized digital and physical assets, Web3 in gaming, metaverse, and creator economy, Wallets and DApp integration

Unit-IV Enterprise Blockchain & Interoperability 07 Hrs.

Private & Consortium Blockchains: Permissioned vs Permissionless, Enterprise use cases, Platforms: Hyperledger, Corda, Ripple (overview)

Hyperledger Fabric (Core Concepts): Components: MSP, peers, orderers, endorsement policies, Channels and privacy, Chaincode (overview), World State and ledger, Transaction flow (conceptual)

Interoperability: Cross-chain bridges, Polkadot overview (relay chain), Cosmos overview (IBC)

Unit-V Blockchain, Crypto Security & Regulation 07 Hrs.

Blockchain Security: Majority attack (51% attack), Sybil attack, Replay and double-spending attacks, MEV (concept only), Wallet security: hot/cold wallets, seed phrase management, Bridge vulnerabilities (high level)

Smart Contract Security: Vulnerabilities: Reentrancy, Integer overflow/underflow, Access control issues, Secure coding practices,

Overview of security tools: Slither, MythX

Zero Knowledge Proofs (Concept): Motivation, zk-SNARKs & zk STARKs (no mathematical depth)

Regulations & Compliance: Global crypto regulations (overview), Indian regulatory landscape (high-level), Ethical and legal considerations.

Blockchain and Crypto Technology Laboratory (RCP23CPC352L)

List of Laboratory Experiments

Suggested Experiments:(Any 8)

1. Implement SHA-256 hashing and develop a simple linked-block simulation.
2. Construct a Merkle tree and verify proof of inclusion.
3. Create Bitcoin/Ethereum testnet wallets; perform and trace transactions.
4. Simulate basic Proof-of-Work mining with difficulty and nonce calculation.
5. Write and deploy a simple smart contract on Remix IDE.
6. Develop and deploy an ERC-20 or ERC-721 token on a testnet.
7. Build a basic Web3 DApp integrated with MetaMask.
8. Analyze vulnerabilities in sample smart contracts using tools.
9. Interact with DeFi testnet protocols (AMM swap simulation).
10. Mini Project integrating blockchain + crypto concepts.

Any other experiment based on syllabus may be included which would help the learner to understand topic/concept.

Oral examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.

Text Books:

1. S. Chandramouli, Asha A. George, Abhilash K. A., and Meena Kartikeyan, "Blockchain Technology", University Press, 2022.
2. Lorne Lantz and Daniel Cawrey, "Mastering Blockchain", O'Reilly, 2020.
3. Andreas M. Antonopoulos and Gavin Wood, "Mastering Ethereum", O'Reilly, 2018.
4. Kumar Saurabh and Ashutosh Saxena, "Blockchain Technology: Concepts and Applications", Wiley, 2020.

Reference Books:

1. Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller, and Steven Goldfeder, "Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction", Princeton University Press, 2016.
2. Dr. Ramchandra Mangrulkar and Dr. Pallavi Mahajan, "Blockchain Essentials: Core Concepts and Implementations", Apress, 2024.
3. Sudeep Tanwar, "Blockchain Technology: from Theory to Practice", Springer, 2022.

Online Resources:

1. NPTEL: Blockchain and its Applications, Prof. Sandip Chakraborty, Prof. Shamik Sural, IIT Kharagpur
2. Coursera: Introduction to Blockchain, James Won-Ki HONG

Program: Computer Engineering	T.Y. B.Tech.	Semester: VI
Software Engineering and DevOPs Laboratory (RCP23CPC353L)		

Prerequisites:

1. Concepts of Object Oriented Programming & Methodology.
2. Knowledge of developing applications with front end & back end connectivity.
3. Knowledge of Linux Operating system, installation and configuration of services and command line basics.

Course Objective(s):

1. To explore the essential phases and critical aspects of an overall software development process in order to design a high-quality software solution in a cost-effective manner for a real world problem.
2. To understand the fundamentals of DevOps engineering.
3. To be proficient with DevOps terminologies, concepts, benefits, and deployment options to meet real world software development requirements.

Course Outcomes:

On completion of the course, the learner will be able to:

CO	Course Outcomes	Blooms Level	Blooms Description
CO1	Explain basic concepts of Software Engineering process and models.	L2	Understand
CO2	Identify requirements, analyze and design for real world software projects.	L4	Analyze
CO3	Apply various principles, phases and activities of DevOps to meet software development process requirements.	L3	Apply
CO4	Describe and implement DevOps principles for CI/CD.	L2	Understand
CO5	Apply deployment and management tools for agile software projects using DevOps.	L3	Apply

Software Engineering and DevOPs Laboratory (RCP23CPC353L) Course Contents

Unit-I 07 Hrs.

Introduction to Software Engineering and Process Models: Nature of Software, Software Engineering, Software Process, Prescriptive Process.

Models: The Waterfall Model, V Model; Evolutionary Process Models: The Spiral Model.

Agile Methodology: Agility Principles.

Agile Process Models: Extreme Programming (XP), Scrum, Agile Modeling (AM), Kanban Model, LEAN models.

Requirement Analysis and Models: Software Requirement Specification (SRS), Scenario Based Models, Class Based Models, Behavioral Models, and Flow Models.

AI based software development process tools: Claude Code, Gradio, OpenUI, etc.

Unit-II 04 Hrs.

Design Engineering and Analysis: Design Principles, Design Concepts, Effective Modular Design-Cohesion and Coupling.

Introduction to DevOps: Minimum Viable Product (MVP) & Cross-functional Teams, Lean, ITIL,DevOps as a prominent culture to achieve agility in the software development process, DevOps Stakeholders, Goals, DevOps and Agile, DevOps Tools.

Unit-III 04 Hrs.

Version Control: Introduction, Overview of Version Control Systems, Role of Version Control System, Types of Control Systems and their Supporting Tools, Importance of version control in CICD pipeline.

Unit-IV 04 Hrs.

Continuous Integration: Introduction to Jenkins (With Master –Slave Architecture), Choosing a launch method, Administering Jenkins slaves, Labels, groups and load balancing. Creating Views and Jobs in Jenkins: The Jenkins user interface, Jobs in Jenkins, Creating Views, Managing Views and Jobs in Jenkins: Managing Views in Jenkins, Navigating a job's project page, Job Execution, The Job Execution Configuration Panel, The Status Panel, Console Panel.

Unit-V 04 Hrs.

Continuous Deployment: Overview of Docker, Benefits of Docker Workflow, Process Simplification, Architecture, Docker Containers, Docker Workflow, Anatomy of Dockerfile, Building an Image, Running an Image, Custom base Images, Storing Images.

Unit-VI 05 Hrs.

Continuous Management: The Parts of an Infrastructure System, Infrastructure Platforms, Infrastructure Resources, Compute Resources, Storage Resources, Network Resources. Puppet: Puppet Architecture, The Puppet Server, setting up the Puppet Agent, Performance Optimizations, Ansible: Ansible Architecture, Ansible and Infrastructure Management, Local Infrastructure Development: Ansible and Vagrant. Introduction to open-source tools for data gathering and management, AWS

Software Engineering and DevOPs Laboratory (RCP23CPC353L)

List of Laboratory Experiments

Suggested Experiments:(Any 08)

1. Prepare detailed statement of problem for the selected / allotted mini project and identify suitable process model for the same with justification.
2. Develop Software Requirement Specification (SRS) document in IEEE format for the project.
3. Perform various GIT operations on local and Remote repositories using GIT Cheat Sheet.
4. To understand Continuous Integration, install and configure Jenkins with Maven/Ant/Gradle to setup a build Job.
5. To Build the pipeline of jobs using Maven / Gradle / Ant in Jenkins, create a pipeline script to Test and deploy an application over the tomcat server.
6. To understand Jenkins Master-Slave Architecture and scale your Jenkins standalone implementation by implementing slave nodes.
7. To understand Docker Architecture and Container Life Cycle, install Docker and execute docker commands to manage images and interact with containers.
8. To learn Dockerfile instructions, build an image for a sample web application using Dockerfile.
9. To learn Software Configuration Management and provisioning using Puppet Blocks (Manifest, Modules, Classes, Function).

Any other experiment based on syllabus may be included which would help the learner to understand topic/concept.

Practical examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.

Text Books:

1. Roger Pressman, "Software Engineering: A Practitioner's Approach", McGraw-Hill Publications, 8^{th} Edition, 2015.
2. Ali Behfrooz and Frederick J. Hudson, "Software Engineering Fundamentals", Oxford University Press.
3. Karl Matthias and Sean P. Kane, "Docker: Up and Running", 3^{rd} Edition, O'Reilly Publication, 2022.
4. Craig Berg, "DevOps For Beginners: A Complete Guide To DevOps Best Practices", 2020.
5. Mikael Krief, "Learning DevOps: A Comprehensive Guide to Accelerating DevOps Culture Adoption with Terraform, Azure DevOps, Kubernetes, and Jenkins", 2^{nd} Edition, Packt Publication, 2022.

Reference Books:

1. Ugrasen Suman, "Software Engineering-Concepts and Practices", Cengage Learning, 2022.
2. Pankaj Jalote, "Software Engineering: A Precise Approach", Wiley Publications, 2010.
3. Rajib Mall, "Fundamentals of Software Engineering", Prentice Hall India, 4^{th} Edition, 2014.
4. Mark S. Merkow, "Practical Security for Agile and DevOps", CRC Press Taylor & Francis, 2022.
5. Emily Freeman, "DevOps for Dummies", 3^{rd} Edition, Wiley Publication, 2019.

Online Resources:

1. NPTEL: Software Engineering, By Prof. Rajib Mall, IIT Kharagpur.
   https://onlinecourses.nptel.ac.in/noc25_cs108/preview
2. Advanced Certification in DevOps & Cloud Computing by Dr. Noor Mohammad and Dr. Pilli Emmanuel Shubhakar, IIT Madras.

Program: Computer Engineering	T.Y. B.Tech.	Semester: VI
Big Data Analytics (RCP23CPE361)		
Big Data Analytics Laboratory (RCP23CPE361L)		

Prerequisites: Database Management Systems, Business Intelligence and Analytics

Course Objective(s):

1. Provide a comprehensive overview of Big Data analytics, its challenges, and the limitations of traditional data mining approaches.
2. Introduce key Big Data technologies and frameworks—such as Hadoop, NoSQL, and MapReduce—along with scalable and streaming-based analytical techniques.
3. Develop the ability to analyze diverse Big Data types (social media, web graphs, data streams) and apply analytical skills to solve complex real-world decision-support problems.

Course Outcomes:

On completion of the course, the learner will be able to:

CO	Course Outcomes	Blooms Level	Blooms Description
CO1	Identify the motivation for Big Data systems and major real-world sources of Big Data.	L2	Understand
CO2	Utilize Big Data frameworks such as Hadoop, NoSQL, and MapReduce to store, retrieve, and process large-scale datasets.	L6	Create
CO3	Apply advanced analytical techniques—including clustering, classification, association mining, and stream/web graph/social media analytics—to derive insights from Big Data.	L3	Apply
CO4	Develop data-driven solutions such as recommendation engines for real-world applications.	L6	Create

Big Data Analytics (RCP23CPE361)
Course Contents

Unit-I Introduction to Big Data and Big Data Analytics 04 Hrs.

Big Data: Definition, Characteristics, Drivers, Traditional vs Big Data, Types of Big Data, Big Data Challenges, Examples of Big Data in Real Life, Big Data Applications. Introduction to Big Data Analytics, Types of Big Data Analytics, Big Data Analytics Process Models, Advantages of Big Data Analytics, Challenges in Big Data Analytics.

Unit-II Big Data Frameworks 08 Hrs.

Hadoop: Hadoop Core Components, Hadoop Ecosystem, Hadoop Physical Architecture, Working of Hadoop, Advantages and Limitations of Hadoop HDFS: Overview of HDFS, HDFS Architecture, HDFS Commands

Hive: Features of Hive, Hive Architecture, Hive Vs Traditional Database, HiveQL

HBase: Introduction to HBase, HBase Architecture, HBase Vs RDBMS, HBase Schema, Indexing in HBase, HBase Shell Queries

Pig: Features, Execution Modes of Pig, Pig Vs SQL, Pig Architecture, Pig Data Model

Zookeeper: Features, Architecture of Zookeeper, Working of Zookeeper

Unit-III MapReduce Paradigm 08 Hrs.

The Map Tasks, Grouping by Key, The Reduce Tasks, Combiners, Details of MapReduce Execution, Coping With Node Failures. Algorithms Using MapReduce: Word count by MapReduce, Matrix-Vector Multiplication by MapReduce, 04 Relational-Algebra Operations, Computing Selections by MapReduce, Computing Projections by MapReduce, Union ,Intersection, and Difference by MapReduce.

Unit-IV Mining Big Data Streams 08 Hrs.

The Stream Data Model: A DataStream-Management System, Examples of Stream Sources, Issues in Stream Processing, Sampling Data in a Stream: Sampling Techniques, Data Ingestion

Spark: Features of Spark, Components of Spark, Architecture of Spark, RRD in Spark, SparkQL, Schedulers in Spark, Shared Variables in Spark

Kafka: Features of Kafka, Components of Kafka, Kafka Cluster Architecture, Kafka Workflow

Filtering Streams: The Bloom Filter Counting Distinct Elements in a Stream: The Count-Distinct Problem, The Flajolet-Martin Algorithm,

Counting Ones in a Window: The Cost of Exact Counts, The Datar Gionis-Indyk, Motwani Algorithm, Query Answering in the DGIM Algorithm.

Unit-V Big Data Mining Algorithms 08 Hrs.

Frequent Pattern Mining: Handling Larger Datasets in Main Memory Basic Algorithm of Park, Chen, and Yu., The SON Algorithm and MapReduce.

Clustering Algorithms: CURE Algorithm. Canopy Clustering, Clustering with MapReduce

Classification Algorithms: Overview SVM classifiers, Parallel SVM, K Nearest Neighbor classifications for Big Data, One Nearest Neighbour

Unit-VI Big Data Analytics Applications 06 Hrs.

Link Analysis: PageRank Definition, Structure of the web, dead ends, Using Page rank in a search engine, Efficient computation of Page Rank, Topic sensitive Page Rank, link Spam, Hubs and Authorities, HITS Algorithm.

Mining Social- Network Graphs: Social Networks as Graphs, Types , Clustering of Social Network Graphs, Direct Discovery of Communities.

Recommendation Engines: A Model for Recommendation Systems, Content-Based Recommendations, Collaborative Filtering

Big Data Analytics Laboratory (RCP23CPE361L)

List of Laboratory Experiments

Suggested Experiments:(Any 8)

1. Installation of Hadoop on a single node cluster and execute HDFS commands.
2. Implementation of Map Reduce program to count words in a text file, matrix multiplication.
3. Execute HIVE commands to load, insert, retrieve, update, or delete data in the tables.
4. Execute HBASE commands to perform basic CRUD operations and joins.
5. Implement matrix-vector multiply using MapReduce and evaluate correctness and scaling.
6. To create RDD, perform various operations and find occurrence of each word.
7. To create SparkQL and execute various SQL commands.
8. Perform Sentiment analysis using Spark Streaming.
9. Implementation of DGIM algorithm.
10. Implementation of PCY, SON algorithm.
11. Implementation of PageRank, HITS algorithm.
12. Implementation of Recommendation System.

Any other experiment based on syllabus may be included which would help the learner to understand topic/concept.

Oral examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.

Text Books:

1. Seema Acharya, Subhashini Chellappan, Big Data and Analytics , 2^{nd} Edition, Wiley, 2022.
2. Raj Kamal and Preeti Saxena, "BIG DATA ANALYTICS - Introduction to Hadoop, Spark, and Machine-Learning", McGraw Hill Education, 2019.
3. Alex Holmes, "Hadoop in Practice", 2^{nd} Edition, Manning Publications Co., 2014.

Reference Books:

1. Anand Rajaraman, Jure Leskovec, and Jeffrey D. Ullman, "Mining of Massive Datasets", 3^{rd} Edition, Cambridge University Press, 2014.
2. C.S.R. Prabhu, Aneesh Sreevallabh Chivukula, Aditya Mogadala, Rohit Ghosh, L.M. Jenila Livingston, "Big Data Analytics: Systems, Algorithms, Applications", Springer, 2019.

Online Resources:

1. https://nptel.ac.in/courses/106104189
2. https://www.coursera.org/specializations/big-datacourses

Program: Computer Engineering	T.Y. B.Tech.	Semester: VI
UI/UX Design (RCP23CPE362)		
UI/UX Design Laboratory (RCP23CPE362L)		

Prerequisite: Design thinking.

Course Objective(s):

1. Explore essential visual design & interaction principles applicable to user interfaces.
2. Conduct user research and apply user-centered methodologies for design.
3. Create wireframes, interactive prototypes, and evaluate usability through testing methods.

Course Outcomes:

On completion of the course, the learner will be able to:

COs	Course Outcomes	Blooms Level	Blooms Description
CO1	Analyze the principles of user interface and user experience.	L4	Analyze
CO2	Apply the UI design elements in building user-friendly applications.	L3	Apply
CO3	Analyze user behavior and business requirements.	L4	Analyze
CO4	Develop user-centric design solutions using wireframes and prototypes.	L6	Create

UI/UX Design (RCP23CPE362)
Course Contents

Unit-I The Importance of UI and Characteristics of Graphical and Web User Interfaces: 06 Hrs.

Defining the User Interface; The Importance of Good Design; Interaction Styles, Graphical User Interface, Web User Interface; Merging Graphical Business Systems and the Web; Principles of User Interface Design.

Unit-II User Interface Design Process: 09 Hrs.

Obstacles and Pitfalls in the Development Path.

Understanding your User or Client: Understanding How People Interact with Computers, Important Human Characteristics in Design, Human Considerations in the Design of Business Systems, Human Interaction Speeds.

Understand the Business Function: Business Definition and Requirements Analysis, Determining Basic Business Functions

Unit-III Principles of Good Interface and Screen Design: 08 Hrs.

Human Considerations in Interface and Screen Design; Technological Considerations in Interface Design; Structures, Functions, Content and Formatting of Menus; Types of Graphical Menus; Window Characteristics; Components of a window; Types of Windows.

Unit-IV Interaction Devices and Screen-Based Controls: 07 Hrs.

Input Devices; Output Devices; Operable Controls; Text Entry/Read Only Controls; Selection Controls; Combination Entry/Selection Controls; Other Operable Controls; Presentation Controls.

Unit-V Effective Text, Feedback, Guidance, Internationalization and Accessibility: 06 Hrs.

Words, Sentences, Messages, and Text; Content and Text for Web Pages; Providing the Proper Feedback; Guidance and Assistance; International Considerations; Accessibility

Unit-VI Creating Meaningful Graphics, Images and Organizing Windows and Pages: 06 Hrs.

Icons: Types, Characteristics, Icon Design Process; Multimedia and Graphics; Choosing the Proper Colors: Color Uses, Possible Problems with Color, Color and Human Vision; Organizing and Laying Out Screens; Iterative Testing.

UI/UX Design Laboratory (RCP23CPE362L)

List of Laboratory Experiments

Suggested Experiments:(Any 08)

1. Identify and analyze the UI and UX of two digital products.
2. Design a Mobile app/ Website that can teach mathematics to children of 4-5 years age.
3. ATM machine/KIOSK screen design for rural people.
4. Design a UI application for Institute event management.
5. Designing user interfaces for the systems using various interaction styles.
6. Design appropriate icons pertaining to a given domain.
7. Redesign of a user interface (Suggest and implement changes in Existing User Interface).
8. Design an interface for home appliances.
9. Design UI for Motor paralysis for disabled people.
10. Mini-project on KIOSK design for hospital/school/educational campus/National Institute.

Any other experiment based on syllabus may be included which would help the learner to understand topic/concept.

Oral examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.

Text Books:

1. Joel Marsh, "UX for Beginners", O'Reilly, 2022.
2. Jon Yablonski, "Laws of UX using Psychology to Design Better Product & Services", O'Reilly, 2021.

Reference Books:

1. Jenifer Tidwell, Charles Brewer, Aynne Valencia, "Designing Interface", 3rd Edition, O'Reilly, 2020.
2. Wilbert O. Galitz, "The Essential Guide to User Interface Design: An Introduction to GUI Design Principles and Techniques", Wiley Publishing, Inc., 2007.

Online Resources:

1. Google UX Design Professional Certificate
   Google UX Design Professional Certificate — Coursera

Program: Computer Engineering	T.Y. B.Tech.	Semester: VI
Virtual Reality (RCP23CPE363)		
Virtual Reality Laboratory (RCP23CPE363L)		

Prerequisites: Computer Graphics.

Course Objective(s):

1. Introduce the principles and technology of Virtual Reality (VR) systems.
2. Familiarize with current VR hardware, software frameworks, and interaction techniques.
3. Develop practical skills in creating immersive VR applications.
4. Explore real-world applications of VR in domains such as gaming, education, healthcare and industrial training.

Course Outcomes:

On completion of the course, the learner will be able to:

CO	Course Outcomes	Blooms Level	Blooms Description
CO1	Explain the key concepts, architecture, and components of contemporary VR systems.	L2	Understand
CO2	Demonstrate knowledge of modern VR hardware, display/input devices, and software frameworks.	L3	Apply
CO3	Apply 3D modelling, rendering and graphics concepts to design virtual environments.	L3	Apply
CO4	Develop interactive VR applications using modern engines and SDKs.	L6	Create
CO5	Evaluate VR applications based on performance, usability, comfort, and human factors.	L5	Evaluate
CO6	Analyze and propose VR applications across different domains (education, training, entertainment).	L4	Analyze

Virtual Reality (RCP23CPE363)
Course Contents

Unit-I Introduction to Virtual Reality: 06 Hrs.

Definitions & evolution of VR, What is virtual reality? ,The beginnings of VR , VR paradigms ,Collaboration, Virtual reality systems, Representation ,User interaction immersive systems, presence & immersion, VR vs AR vs MR, major application domains.

Unit-II VR Hardware & Software Frameworks: 07 Hrs.

Head-mounted displays (HMDs), motion tracking, controllers, sensors, input/output devices, VR system architecture, display latency, software frameworks (XR Interaction Toolkit, OpenXR, SteamVR, etc.).

Unit-III 3D Graphics & Modelling for VR: 07 Hrs.

Coordinate systems, transformations, 3D modelling tools, lighting & shading, textures/materials, rendering pipeline, scene optimization for VR.

Unit-IV Interaction & User Interface in VR: 07 Hrs.

Navigation techniques (teleportation, locomotion), selection/manipulation, gesture & hand-tracking, spatial audio & haptics, designing UI for VR, motion sickness and human comfort.

Unit-V VR Application Development: 08 Hrs.

Introduction to development platform (e.g., Unity with XR toolkit), setting up VR projects, camera & rig setup, controller integration, object interactions, scripting (C#), deploying VR applications (PC, standalone).

Unit-VI Applying Virtual Reality: 07 Hrs.

Virtual reality: the medium, Form and genre, What makes an application a good candidate for VR, Promising application fields, Demonstrated benefits of virtual reality , More recent trends in virtual reality application development, A framework for VR application development.

Virtual Reality Laboratory (RCP23CPE363L)

List of Laboratory Experiments

Suggested Experiments:(Any 08)

1. Study of VR Hardware and Setup – identify components, set up a VR headset and controllers.
2. Configuring a VR Project in Unity + XR Toolkit – create base VR scene, set up VR camera rig.
3. Building & Navigating a Simple 3D Virtual Environment – design scene, load assets, implement navigation.
4. Object Interaction in VR – implement grabbing, throwing, UI interactions using controllers/hand-tracking.
5. Spatial Audio & Lighting Effects – add 3D audio, lighting, optimize for immersion.
6. Developing a VR Mini-Game – design and implement a small interactive game in VR (e.g., object hunt, gallery tour).
7. Performance Optimization & Comfort Design – analyze frame-rate, latency, adjust lighting/meshes, implement teleportation vs smooth locomotion, evaluate motion sickness.
8. Mini Project: Domain-Specific VR Application – students pick a domain (education, training, simulation) and create a short VR experience; show demo and report.

Any other experiment based on syllabus may be included which would help the learner to understand topic/concept.

Oral examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.

Text Books:

1. Steven M. LaValle, "Virtual Reality", Cambridge University Press, 2016.
2. Jonathan Linowes, "Unity 2020 Virtual Reality Projects", Packt Publishing, 2020.
3. Jason Jerald, "The VR Book: Human-Centered Design for Virtual Reality", ACM Books, 2020.
4. William R. Sherman and Alan B. Craig, "Understanding Virtual Reality: Interface, Application and Design", The Morgan Kaufmann Series in Computer Graphics, Morgan Kaufmann Publishers, San Francisco, CA, 2002.
5. Alan B. Craig, William R. Sherman, and Jeffrey D. Will, "Developing Virtual Reality Applications: Foundations of Effective Design", Morgan Kaufmann, 2009.

Reference Books:

1. "Unity 2022 by Example: A Project-based Guide to Building 2D & 3D Games, Enhanced for AR, VR & MR Experiences", Packt, 2022.

2. "Ultimate Unity for Multiplayer VR Development: Focuses on Multiplayer VR with Unity, XR Toolkit and Photon Fusion", recent edition.

3. M. C. tom Dieck et al., "Augmented Reality and Virtual Reality: New Trends in Immersive Technology", Springer Cham, 2021.

4. D. W. Staat, "Virtual Reality in Higher Education: Instruction for the Digital Age", Bloomsbury, 2021.

Online Resources:

1. Introduction to Virtual Reality (by Goldsmiths, University of London on Coursera https://www.coursera.org/specializations/virtual-reality

2. VR Development With XR Interaction Toolkit (on LearnXR.io)

3. Unity VR Tutorial: Unity XR Interaction Toolkit 3.0 (on Udemy) https://www.udemy.com/course/unity-vr-game development/?couponCode=25BBPMXNVD35V2

4. Free tutorial in Virtual Reality https://www.udemy.com/course/how-to-create-vr-experiences-simlab-soft-vr studio/?srsltid=AfmBOopN4 jR29Pi7m5jzr8HB-UeF04y36V6ZrOTM355z ibRe6sUyjOg

Program: Computer Engineering	T.Y. B.Tech.	Semester:VI
Computer Vision(RCP23CPE371)		
Computer Vision Laboratory (RCP23CPE371L)		

Prerequisites: Computer Graphics.

Course Objective(s):

1. To introduce fundamental concepts of computer vision and equip students with foundational knowledge in image formation, photometric processing, and geometric transformations.
2. To develop practical skills in feature extraction, recognition methodologies, and morphological image processing techniques for object segmentation and detection.
3. To explore advanced applications in feature-based alignment, 3D vision, and recognition, including face detection, pose estimation, and scene understanding.

Course Outcomes:

On completion of the course, the learner will be able to:

COs	Course Outcomes	Blooms Level	Blooms Description
CO1	Apply key principles of image formation and photometric processing to analyze images and digital camera functions.	L3	Apply
CO2	Evaluate feature extraction techniques and morphological operations for conditioning, labeling, and matching objects in images	L5	Evaluate
CO3	Apply segmentation and detection algorithms to identify and isolate objects and regions of interest in complex images.	L3	Apply
CO4	Develop real-world computer vision applications employing in advanced feature-based alignment, 3D object recognition, and scene understanding.	L6	Create

Computer Vision(RCP23CPE371)
Course Contents

Unit-I Introduction 04 Hrs.

What is computer Vision, Image Formation: Geometric Primitives, 2D transformation, 3 D transformation, 3D rotation, 3D to 2D Projection, Lens Distortion

Photometric Image Formation: Lighting, reflectance and shading

The digital camera: sampling and quantizition.

Unit-II Recognition Methodology 06 Hrs.

Conditioning, Labeling, Grouping, Extracting, Matching.

Morphological Image Processing: Introduction, Dilation, Erosion, Opening, Closing,Hit-or-Miss transformation, Morphological algorithm operations on binary images, Morphological algorithm operations on gray-scale images, Thinning, Thickining , Region growing, region shrinking.

Unit-III Feature Extraction and Model Fitting: 08 Hrs.

Edges - Canny, LOG, DOG, Line detectors (Hough Transform), Corners – Harris and Hessian Affine, Orientation Histogram, SIFT, SURF, HOG, GLOH, Deformation, RANSAC, Scale-Space Analysis-Image Pyramids and Gaussian derivative filters, Gabor Filters and DWT.

Unit-IV Object Segmentation and Detection: 08 Hrs.

Active contours, Split and merge, Mean shift and mode finding, Normalized cuts, Graph cuts and energy-based methods, Application: Medical image segmentation.

Unit-V Feature-based alignment 08 Hrs.

2D and 3D feature-based alignment, Pose estimation, Geometric intrinsic calibration, Calibration patterns, Vanishing points, Application: Single view metrology, Rotational motion, Radial distortion.

Unit-VI Recognition: 08 Hrs.

Object detection, Face detection, Pedestrian detection, Face recognition, 3D shape models, Application: Personal photo collections, Instance recognition, Category recognition, Context and scene understanding.

Computer Vision Laboratory(RCP23CPE371L)

List of Laboratory Experiments

Suggested Experiments:(Any 08)

1. To perform basic and geometric image transformations such as rotation, scaling, translation, and affine transformations using computer vision techniques.
2. To apply various image de-noising and enhancement techniques for improving image quality using spatial and frequency-domain methods.
3. To apply image smoothing and sharpening filters for enhancing image details.
4. To perform segmentation of medical images for isolating regions of interest using.
5. To implement feature detection techniques such as edge detection, corner detection, and image pyramids using OpenCV.
6. To detect and localize objects in an image or video using classical or deep-learning based object detection methods.
7. To track the movement of objects across video frames using suitable object tracking algorithms.
8. To implement pattern recognition techniques for classifying or identifying patterns using machine learning or image-based features.
9. To implement a face recognition system using feature extraction and classification techniques or pre-trained deep learning models.
10. Mini Project Based Learning /Research Review Article.

Any other experiment based on syllabus may be included which would help the learner to understand topic/concept.

Oral examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.

Text Books:

1. Richard Szeliski, 'Computer Vision: Algorithms and Applications", 2^{nd} Edition, Springer Verlag London Limited, 2022.
2. Alexander Hornberg, "Handbook of Machine and Computer Vision", Wiley-VCH, 2017.

Reference Books:

1. M. Sonka, V. Hlavac, and R. Boyle, "Image Processing, Analysis, and Machine Vision, Thomson India Edition, 4^{th} Edition, 2017.

2. David Forsyth, Jean Ponce, "Computer Vision: A Modern Approach, Pearson Education", 2^{nd} Edition, 2015.

Online Resources:

1. https://onlinecourses.nptel.ac.in/noc25$_{c}$s143/*preview*

1. https://www.coursera.org/learn/introduction-computer-vision-watson-opencv

Program: Computer Engineering	T.Y. B.Tech.	Semester: VI
Development Frameworks (RCP23CPE372)		
Development Frameworks Laboratory (RCP23CPE372L)		

Prerequisites:: Java Programming, Object Oriented Programming Concept, Database Management System.

Course Objective(s):

1. To apply design patterns and SOLID principles for creating flexible and maintainable software.
2. To develop database-driven Java web applications using Collections, Servlets, JSP, and JDBC.
3. To build enterprise applications using ORM, Spring Core, Spring Boot, and MVC architecture.
4. To design secure RESTful services and microservices using Spring Boot, JWT/OAuth2, and service discovery tools.

Course Outcomes:

On completion of the course, the learner will be able to:

CO	Course Outcomes	Blooms Level	Blooms Description
CO1	Apply Object-Oriented Design Patterns and SOLID principles to build scalable and maintainable software solutions.	L3	Apply
CO2	Develop Java applications using Collections, Servlets, JSP, and JDBC for dynamic, database-driven web solutions.	L6	Create
CO3	Build enterprise-level applications using ORM tools and Spring Framework (Core, Boot) with dependency injection and MVC architecture.	L6	Create
CO4	Design and implement secure RESTful services and Microservices using Spring Boot, JWT/OAuth2, API Gateway, Load Balancing, and Service Discovery.	L6	Create

Development Frameworks (RCP23CPE372) Course Contents

Unit-I Design Patterns: 03 Hrs.

Introduction to design patterns and Implementation: Singleton, Factory, Observer, Strategy.

SOLID Principles: Understanding and applying SOLID principles for better design, Implementation of SOLID Principles.

Unit-II Collections: 04 Hrs.

List: ArrayList, LinkedList, Set: HashSet, Tree Set, Map: HashMap, LinkedHashMap.

Unit-III Servlets and JSP: 08 Hrs.

Introduction, Web application Architecture, Http Protocol & Http Methods, Web Server & Web Container, Servlet Interface, GenericServlet, HttpServlet, Servlet Life Cycle, Servlet Communication, Introduction, JSP LifeCycle, JSP Implicit Objects & Scopes, JSP Directives, JSP Scripting Elements, JSP Actions, connecting to databases using JDBC, Executing SQL queries and managing results.

Introduction to Object-Relational Mapping (ORM): Overview of Hibernate and JPA, Creating a simple application using Hibernate.

Unit-IV Introduction to Spring: 10 Hrs.

Overview of Spring Framework features, Inversion of Control (IoC) and Dependency Injection (DI). (JDBC Must be covered)

Spring Core: Understanding Beans, Application Context, and Bean Lifecycle Configuring Spring with XML and Java annotations

Unit-V Introduction to Spring Boot: 10 Hrs.

Introduction to MVC architecture, understanding its purpose and advantages over traditional Spring.

Setting Up Spring Boot Applications: Project structure and configuration, Annotations, Layers, Service Class, Repository Layer, Controllers, Model Object, Thymeleaf.

Building RESTful Web Services: Create REST APIs using Spring Boot. Spring Data JPA: Introduction to database interactions and repository patterns, JWT, OAuth2.

Unit-VI Microservices: 07 Hrs.

Fundamentals of Microservices, Microservices Architecture & Design Principles, Service Discovery, Service Registry, Working with Feign, Load Balancing, API Gateway.

Development Frameworks Laboratory (RCP23CPE372L)

List of Laboratory Experiments

Suggested Experiments:(Any 8)

1. Implementation of Design Patterns and SOLID Principles. (Applying Singleton, Factory, Strategy/Observer along with SOLID-based refactoring)
2. Develop an application to implement Java Collections Framework. (ArrayList, LinkedList, HashSet, TreeSet, HashMap, LinkedHashMap)
3. Developing a Basic Servlet Application. (Handling GET/POST requests and servlet lifecycle)
4. Servlet-to-JSP Communication with MVC Approach. (Forwarding requests, using JSP scopes, and displaying dynamic data)
5. JSP Application with JDBC Connectivity. (Performing CRUD operations using JSP, JDBC, and MySQL)
6. Building an ORM-Based Application using Hibernate/JPA. (Entity mapping and CRUD operations)
7. Spring Core Application. (Bean creation, Dependency Injection using XML and annotations)
8. Spring Boot MVC Application. (Controller, Service, Repository layers with Thymeleaf views)
9. Creating RESTful Web Services using Spring Boot. (CRUD APIs using Spring Data JPA)
10. Microservices Implementation. (Service Discovery, Feign Client, and API Gateway)
11. Mini-Project.

Any other experiment based on syllabus may be included which would help the learner to understand topic/concept.

The students are expected to complete 1 Mini-Project based on the syllabus.

Oral examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.

Text Books:

1. Craig Walls, "Spring in Action", 6th Edition, Manning Publications, 2022.
2. Craig Walls, "Spring Boot in Action", Manning Publications, 2016.
3. Greg L. Turnquist, "Learning Spring Boot 3.0 – Third Edition", O'Reilly Media, 2023.
4. Magnus Larsson, "Microservices with Spring Boot and Spring Cloud", O'Reilly Media, 2023.

Reference Books:

1. Christian Bauer and Gavin King, "Java Persistence with Hibernate", 2nd Edition, Manning Publications, 2020.
2. Ranga Rao Karanam, "Mastering Spring Boot 3.0", 1st Edition, Packt Publishing, 2023.
3. John Thompson, "The Complete Beginner's Guide to Spring Boot", Independently Published, 2022.
4. Mark Johnson, "Learn All About Spring Boot: Building Modern Java Applications", Packt Publishing, 2023.
5. Moisés Macero García, "Learn Microservices with Spring Boot", Apress, 2020.
6. Sam Newman, "Building Microservices: Designing Fine-Grained Systems", O'Reilly Media, 2015.

Online Resources:

1. Zero to Hero: Master Java SpringBoot JPA with Projects (Link: https://www.coursera.org/learn/packt-zero-to-hero-java-springboot-and-jpa-mastery with-real-project-1ee3f)
2. Master Spring Boot 3 & Spring Framework 6 with Java (Link: https://www.udemy.com/course/spring-boot-and-spring-framework-tutorial-for beginners)

Program: Computer Engineering	T.Y. B.Tech.	Semester: VI
High Performance Computing (RCP23CPE373)		
High Performance Computing Laboratory (RCP23CPE373L)		

Prerequisite: Operating System.

Course Objective(s):

1. To understand parallel computing concepts, architectures, and performance metrics.
2. To design and implement efficient parallel programs using OpenMP, MPI, and CUDA

Course Outcomes:

On completion of the course, the learner will be able to:

COs	Course Outcomes	Blooms Level	Blooms Description
CO1	Describe fundamentals of parallel computing, classify parallel architectures, and evaluate performance.	L2	Understand
CO2	Design and implement parallel algorithms using decomposition and load balancing.	L6	Create
CO3	Develop high-performance programs using OpenMP, MPI and CUDA.	L6	Create

High Performance Computing (RCP23CPE373) Course Contents

Unit-I Introduction to Parallel Computing 06 Hrs.

Parallelism, Levels of parallelism(instruction, transaction, task, thread, memory, function) Classification Models: Architectural Schemes(Flynn‘s, Shore‘s, Feng‘s, Handler‘s) Memory Access: Distributed Memory, Shared Memory, Hybrid Distributed Shared Memory Parallel Architecture: Pipeline Architecture: Arithmetic pipelines, Floating Point, Array Processor.

Unit-II Parallel Programming Platform and Algorithm Design: 10 Hrs.

Parallel Programming Platform: Physical Organization of Parallel Platforms, Communication Costs in Parallel Machines Algorithm Design: Preliminaries, Decomposition Techniques, Characteristics of Tasks and Interactions, Mapping Techniques for Load Balancing, Methods for Containing Interaction Overheads, Parallel Algorithm Models.

Unit-III Performance Measures: 04 Hrs.

Performance Measures: Speedup, execution time, efficiency, cost, scalability, Effect of granularity on performance, Scalability of Parallel Systems, Amdahls Law, Gustavsons Law, Performance Bottlenecks, The Karp Flatt Metric

Unit-IV HPC Programming : OpenMP and MPI: 07 Hrs.

HPC Programming: Programming Using the Message-Passing Paradigm: Principles of Message Passing Programming, The Building Blocks: Send and Receive Operations MPI: the Message Passing Interface, Topology and Embedding, Overlapping Communication with Computation, Collective Communication and Computation Operations, Introduction to OpenMP

Unit-V GPU and CUDA Programming: 08 Hrs.

Overview of GPU, evolution of GPU, CPU vs. GPU, overview of CUDA: Features, Benefits, Architecture. Programming Model CUDA: Kernels and kernel launches, Thread and block indexing, CUDA Memory Management: Memory Hierarchy and Memory Management, Case Studies: simulation, data analytics and machine learning.

Unit-VI Advanced Topics in GPU: 07 Hrs.

Introduction to GPU accelerated libraries, GPU computing frameworks (TensorFlow, PyTorch) and their integration with GPUs, Introduction to GPU clusters and distributed GPU computing, Clu

Setup & its Advantages. Case studies: Real-world applications for GPU computing.

High Performance Computing Laboratory (RCP23CPE373L)

List of Laboratory Experiments

Suggested Experiments:(Any 08)

1. Perform Linux based commands on remote machine.
2. Implement parallel program using OpenMP library and analyze its performance.
3. Implement parallel program using MPI platform and analyze its performance.
4. Set up the CUDA environment, install the CUDA Toolkit, and write a basic CUDA program to understand the CUDA development environment.
5. Implement vector addition using CUDA to introduce students to parallelism, thread management, and memory allocation in GPU programming.
6. Develop a CUDA program for matrix multiplication to understand parallelism and optimization techniques in GPU computing.
7. Apply CUDA for image processing tasks, like blurring and edge detection, to learn how to process images efficiently using GPU parallelism.
8. Implement parallel reduction operations (e.g., sum, min, max) to grasp the concept of efficient parallel reduction.
9. Explore parallel sorting algorithms using CUDA, comparing their performance with CPU based sorting and optimizing CUDA sorting.
10. Employ CUDA to solve real-world problems to understand the power of GPU parallelism.

Any other experiment based on syllabus may be included which would help the learner to understand topic/concept.

Oral examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.

Text Books:

1. Ananth Grama et al., "Introduction to Parallel Computing", 2^{nd} Edition, Addison Wesley, 2023.
2. Alexander Heifetz, "High Performance Computing for Drug Discovery and Biomedicine", Spı Nature, ISBN: 1071634496, 9781071634493, 2023.

3. Michael J. Quinn, "Parallel Programming in C with MPI and OpenMP", 1st Edition, McGraw-Hill, 2003.

4. Richard Ansorge, "Programming in Parallel with CUDA", Cambridge University Press, ISBN-13: 978-1108479530, 2022.

Reference Books:

1. Georg Hager and Gerhard Wellein, "Introduction to High Performance Computing for Scientists and Engineers", CRC Press, 2019.

2. Duane Storti and Mete Yurtoglu, "CUDA for Engineers", 1st Edition, Addison Wesley, 2016.

3. Kai Hwang and Naresh Jotwani, "Advanced Computer Architecture: Parallelism, Scalability, Programmability", 3rd Edition, McGraw-Hill, 2018.

Online Resource:

1. https://nptel.ac.in/courses/106108055

Program: Computer Engineering	T.Y. B.Tech.	Semester: VI
Ethical Hacking and Digital Forensics (RCP23CMD351)		
Ethical Hacking and Digital Forensics Laboratory (RCP23CMD351L)		

Prerequisites: Nil

Course Objective(s):

1. Introduce ethical hacking concepts, phases of cyber-attacks, and the legal and ethical responsibilities of cybersecurity professionals.
2. Develop the ability to perform reconnaissance, vulnerability assessment, exploitation, and penetration testing using standard tools and techniques.
3. Provide knowledge of digital forensics methodologies including evidence collection, preservation, analysis, and reporting as per legal standards.
4. Enable learners to analyze security breaches, perform system and network investigations, and apply forensic tools for cybercrime detection and incident handling.

Course Outcomes:

On completion of the course, the learner will be able to:

CO	Course Outcomes	Blooms Level	Blooms Description
CO1	Explain the phases of ethical hacking, legal frameworks, and apply professional ethics during security assessments.	L2	Understand
CO2	Perform footprinting, scanning, vulnerability assessment, and simulate system, web, and wireless attacks in a controlled environment.	L3	Apply
CO3	Apply digital forensic procedures for acquiring, preserving, examining, and analyzing digital evidence from storage, memory, mobile, or network sources.	L3	Apply
CO4	Prepare technical forensic or penetration testing reports, follow chain of custody, and recommend preventive and corrective security measures.	L3	Apply

Ethical Hacking and Digital Forensics (RCP23CMD351) Course Contents

Unit-I Introduction to Ethical Hacking & Cyber Laws: 04 Hrs.

Ethical hacking vs. malicious hacking, Hacker types: White hat, black hat, grey hat, hacktivists, Hacking phases: Reconnaissance, Scanning, Gaining Access, Maintaining Access, Covering Tracks, Footprinting – Passive & Active information gathering, Social engineering (Phishing, Pretexting, Vishing, Shoulder Surfing), Cyber laws in India: IT Act 2000 & Amendments, Evidence Act relevance, Legal & ethical responsibilities of an ethical hacker.

Unit-II Scanning, Enumeration & Vulnerability Assessment 04 Hrs.

Network scanning: ICMP scanning, TCP connect, SYN scan, NULL, FIN, Xmas scans, Tools: Nmap, Zenmap, Angry IP Scanner, Enumeration: SNMP, LDAP, FTP, SMTP, NetBIOS, Banner grabbing, OS fingerprinting, Vulnerability assessment vs. Penetration testing, Tools: Nessus, OpenVAS, Nikto.

Unit-III System & Network Hacking Techniques 06 Hrs.

Password cracking: Brute force, dictionary, rainbow tables, John the Ripper, Hashcat, Privilege escalation: Windows & Linux techniques, Keyloggers, spyware, Trojans (conceptual – no overlap with malware internals studied earlier), Hacking Windows systems: SAM file extraction, Mimikatz basics, Hacking Linux systems: /etc/shadow, SSH attacks, Sniffing & MITM attacks (Ettercap, Bettercap, ARP poisoning – practical focus), Session hijacking fundamentals.

Unit-IV Web Application & Wireless Hacking 05 Hrs.

Web application penetration testing methodology, Burp Suite basics: Intercepting, scanning, exploiting, Command Injection, Directory Traversal, File Upload attacks, Session hijacking & Cookie stealing, Wireless hacking: WEP cracking, WPA/WPA2 handshake capture & cracking (aircrack-ng), Evil Twin & Rogue Access Point attacks.

Unit-V Digital Forensics & Incident Handling 05 Hrs.

Digital forensics process: Identification, Preservation, Collection, Examination, Analysis, Reporting, Forensic readiness & chain of custody, Disk forensics: Imaging (dd, FTK Imager), Hashing (MD5/SHA-1 for integrity), Autopsy/Sleuth Kit usage, Volatile data acquisition: RAM dump (FTK Imager Lite, DumpIt), live response tools, File systems: FAT, NTFS artifacts (deleted files, metadata, MFT), Timestamp analysis, log analysis.

Unit-VI Network, Email, Mobile & Cloud Forensics 04 Hrs.

Network forensics: Packet reconstruction, PCAP analysis, Wireshark forensic techniques, Email forensics: Header analysis, tracing emails, spoofed email detection, Cloud forensics fundamentals: Data acquisition challenges, legal issues, hypervisor forensics, Report writing & forensic documentation standards.

Ethical Hacking and Digital Forensics Laboratory (RCP23CMD351L)

List of Laboratory Experiments

Suggested Experiments:(Any 8)

1. Perform passive OSINT reconnaissance on a mock target and prepare an evidence log suitable for a forensic timeline.
2. Use DNS/WHOIS and subdomain enumeration tools to map a target's public footprint; capture and store outputs with hashes.
3. Run safe vulnerability scans (in an isolated lab) using an industry scanner; validate two findings manually and document false positives.
4. Demonstrate safe, controlled exploitation of a deliberately vulnerable VM (e.g., privilege escalation on Linux or Windows) and record the post-exploit steps (no real targets).
5. Create and verify a forensic image of a disk using a forensic imaging tool; compute hashes and demonstrate evidence integrity verification.
6. Recover deleted files from a disk image and reconstruct a simple timeline of user activity.
7. Capture and analyze a memory image from a test VM; identify running processes, network connections, and suspicious loaded modules.
8. Simulate a basic SQL injection on a lab web application, extract controlled test data, and then apply a mitigation (parameterized queries) to demonstrate fix.
9. Demonstrate a stored/reflective XSS attack on an isolated test page and then implement Content Security Policy (CSP) mitigation.
10. Perform captive-portal/WPA2 test in a controlled wireless lab (assess AP configuration weaknesses) and recommend hardening steps.
11. Perform static malware analysis on a safe sample in an isolated sandbox (examine strings, headers, basic indicators) and summarize findings.
12. Conduct a basic dynamic malware behavior run in a sandbox environment (network, file, regi changes) and capture indicators of compromise.

13. Acquire a logical image from an Android or iOS test device (emulator or spare device), extract call/SMS/app artifacts, and document acquisition steps and limitations.

14. Produce a professional forensic/penetration test report from one of the above labs: include scope, methodology, evidence (hashed files), findings, severity, remediation, and an executive summary suitable for stakeholders.

Any other experiment based on syllabus may be included which would help the learner to understand topic/concept.

Oral examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.

Text Books:

1. Matt Walker, "CEH Certified Ethical Hacker All-in-One Exam Guide", 6^{th} Edition, McGraw Hill, 2023.

2. William Stallings, "Computer Security: Principles and Practice", 5^{th} Edition, Pearson, 2023.

3. Nelson, Phillips & Steuart, "Guide to Computer Forensics and Investigations", 6^{th} Edition, Cengage, 2023.

Reference Books:

1. Dafydd Stuttard & Marcus Pinto, "The Web Application Hacker's Handbook", 2^{nd} Edition, Wiley, 2023 (latest available).

2. Jason Luttgens, Matthew Pepe, Kevin Mandia, "Incident Response & Computer Forensics", 3^{rd} Edition, McGraw Hill, 2022.

3. Michael Sikorski & Andrew Honig, "Practical Malware Analysis: The Hands-On Guide to Dissecting Malicious Software", Reprint Edition, No Starch Press, 2023.

Online Resource:

1. MITRE ATT&CK Framework – Real-world hacking tactics & techniques
   https://attack.mitre.org

2. OWASP (Open Web Application Security Project) – Web & API security, Top 10 vulnerabilities, labs https://owasp.org

3. DFIR Training & Forensics Tools (SANS — Autopsy — Volatility Foundation)
   https://digital-forensics.sans.org
   https://www.autopsy.com
   https://www.volatilityfoundation.org

Program: Computer Engineering	T.Y. B.Tech.	Semester: VI
Project Stage-I (RCP23XEL351P)		

Course Objective(s):

1. To understand the basic concepts and principles of project development.
2. To formulate/identify the problem statement.
3. To implement the solution as per the problem statement.
4. To develop the team building, writing, logical reasoning and management skills.
5. To provide the connections between the designs and concepts across different disciplinary boundaries.
6. To encourage students to become independent personnel, critical thinkers and lifelong learners.

Course Outcomes:

On completion of the course, the learner will be able to:

COs	Course Outcomes	Blooms Level	Blooms Description
CO1	Analyze the problem statement and produce solution of the problem considering cultural, social, environmental and economic factors using appropriate tool and method.	L4	Analyze
CO2	Interpret project based learning that allows students to transfer existing ideas into new applications.	L2	Understand
CO3	Apply the ability to work in teams and manage to conduct the project development activity.	L3	Apply
CO4	Use different perspectives from relevant disciplines which help them to get internships, jobs, and admission for higher studies.	L3	Apply
CO5	Explain the project development in the form of technical writing, and interpret what constitutes plagiarism and the use of proper referencing styles.	L2	Understand

Syllabus:

Domain knowledge (any beyond) needed from the following areas for the effective implementation of the project:

Database Management System, Networking and Internet of Things, Embedded Systems, Data science and Big data, Web and Application Development, Robotics, AI and Machine Learning, etc.

The above areas can be updated based on the technological innovations and development needed for specific project.

Guidelines: The main purpose of this activity is to improve the students' technical skills, communication skills by integrating writing, presentation and teamwork opportunities.

- Each student shall work on project stage-I as approved by the departmental committee, a group of 03 to 04 students (max allowed: 5 students in extraordinary cases, subject to the approval of the departmental committee) shall be allotted for each Project stage-I. The departmental committee shall include Head of Department, project coordinator and guide(s).
- Each group shall submit at least 3 topics for the Project stage-I. The departmental committee shall finalize one topic for every group.
- Each group is required to maintain separate log book for documenting various activities of the project stage-I (Refer Table 4).
- Each group will be reviewed twice in a semester and marks will be allotted based on the various points mentioned in the evaluation scheme.
- In the first review of this semester, each group is expected to complete 30% of project stage-I.
- In the second review of this semester, each group is expected to complete 50% of project stage-I.
- Interaction with alumni mentor will also be appreciated for the improvement of project stage-I.

Assessment Criteria:

- At the end of the semester, after confirmation by the project guide, each project group will submit project completion report in prescribed format for assessment to the departmental committee.
- Assessment of the project (at the end of the semester) will be done by the departmental committee.
- Oral examination shall be conducted by Internal and External examiners. Students have to give presentation and demonstration based on their project stage-I.

Prescribed project report guidelines:

Size of report shall be of minimum 30 pages (excluding cover and front pages). Project stage-I report should include appropriate content for:

- **Abstract**
- **Introduction**
  - Background
  - Motivation
  - Problem Statement
  - Objectives
  - Scope
- **Literature Survey**
  - Review of Existing System(s)
  - Limitations of Existing System(s)
- **Proposed System**
  - Analysis/Framework/ Algorithm
  - Details of H/W and S/W required
  - Design details
  - Methodology (your approach to solve problem)
- **Implementation Plan for Project Stage-II**
- **Conclusion**
- **References**

Assessment criteria for the departmental committee for Continuous Assessment:

Guide will monitor weekly progress and marks allocation will be as per Table 5.

Assessment criteria for the departmental committee for End Semester Exam:

Departmental committee will evaluate project as per Table 6.

Table 4: Log Book Format

Sr	Week (Start Date:End Date)	Work Done	Sign of Guide	Sign of Coordinator
1				
2				

Table 5: Continuous Assessment Sheet

Sr	P.R.N.	Name of Student	Student Attendance (5)	Log Book Maintenance (5)	Literature Review (5)	Depth of Understanding (5)	Report (5)	Total (25)
			5	5	5	5	5	25

Table 6: Evaluation Sheet

Sr	P.R.N.	Name of Student	Project Stage - I Selection (5)	Design/ Simulation/ Logic (5)	PCB/ Hardware/ Programming (5)	Result Verification (5)	Presentation (5)	Total (25)
			5	5	5	5	5	25

Program: Computer Engineering	T.Y. B.Tech.	Semester: VI
Environmental Studies (RCP23XHS361T)		

Prerequisite: Interest in Environment and its impact on Human.

Course Objective(s):

1. To familiarise students with environment related issues such as depleting resources, pollution, ecological problems and the renewable energy scenario.
2. To give overview of Green Technology options.

Course Outcomes:

On completion of the course, the learner will be able to:

CO	Course Outcomes	Blooms Level	Blooms Description
CO1	Describe how human activities affect environment .	L2	Understand
CO2	Discuss the various technology options that can make a difference.	L2	Understand

Environmental Studies (RCP23XHS361T)
Course Contents

Unit-I Air Pollution: 02 Hrs.

Sources of Air pollution. Definition of Air Quality Index and how it is measured.

Unit-II Water Pollution: 02 Hrs.

Sources of water pollution. Ground water pollution and eutrophication.

Unit-III Noise Pollution: 01 Hrs.

Noise pollution and sources. Decibel limits for hospital, library, silence zone.

Unit-IV Biodiversity loss: 01 Hrs.

Value of Biodiversity. Endangered species.

Unit-V Deforestation: 02 Hrs.

Product and services provided by forests. Relationship between forests and climate change.

Unit-VI Renewable Energy sources: 02 Hrs.

Our energy needs and global energy crisis. Renewable energy sources.

Unit-VII Climate change: 02 Hrs.

Greenhouse gases and climate change.

Unit-VIII Green Technology 02 Hrs.

Data Center Energy Efficiency, Thin-Client and Energy Efficiency.

List of Tutorials:

1. Case study on Smog.
2. Qualitative and Quantitative methods for Air Pollution Monitoring
3. Presentation on Water Pollution (Industrial, Sewage) explaining any specific case.
4. General Techniques in Water Quality Monitoring
5. List effects of noise pollution on human health. Measure decibel level in college library, cant classroom

6. Case study on effect of Pollution on Biodiversity loss.
7. Radioactive and Hazardous Pollutants.
8. Debate for and against "To promote Economic growth, Deforestation is required."
9. Presentation on different Renewable Energy Technologies.
10. Green Hydrogen: The Energy of Future
11. Report on major impact of Global warming on Environment giving real examples.
12. Report on advantages and examples of Green Building for Sustainable development,
13. Sustainable Software Design: Reducing energy/carbon footprint (e.g., green coding, optimized AI/ML, efficient cloud usage)
14. Reducing AI's carbon footprint (efficient models, serverless) and applying AI for environmental good, like optimizing energy grids, smart buildings, precision agriculture (water/fertilizer use), waste management, and climate modeling.

Text Books:

1. R. Rajagopalan, "Environmental Studies: From Crisis to Cure", Oxford University Press, 4^{th} Edition, 2023.
2. Erach Bharucha, "Textbook of Environmental Studies For Undergraduate Courses", Universities Press (India) Pvt. Limited, 2021.
3. P. Narayanan, "Environmental Pollution: Principles, Analysis and Control", CBS Publishers, 2018.
4. Mohammad Dastbaz, Colin Pattinson, Babak Akhgar, "Green Information Technology: A Sustainable Approach", Morgan and Kaufman, Elsevier, 2015.

Reference Books:

1. Paulina Golinska, Marek Fortsch, Jorge Marx-Gómez, "Information Technologies in Environmental Engineering", New Trends and Challenges, Springer, 2011.

Online Resources:

1. CITES: www.cites.org
2. Convention on Biological Diversity: www.biodiv.org

3. Kalpvriksh: www.kalpvriksh.org

4. Water pollution: http://en.wikipedia.org/wiki/Water_pollution

4. Ecosan: www.eco-solutions.org

Program: Computer Engineering	T.Y. B.Tech.	Semester: VI
Generative Artificial Intelligence (RCP23CSC351)		

Prerequisites: Python Programming and Machine Learning Fundamentals.

Course Objective(s):

1. To introduce how AI, machine learning, and predictive analytics address environmental and sustainability challenges.
2. To develop the ability to apply data-driven AI models across climate, energy, agriculture, water, waste, and urban systems.
3. To enable the design of responsible, ethical AI-powered sustainability solutions aligned with global goals.

Course Outcomes:

On completion of the course, the learner will be able to:

CO	Course Outcomes	Blooms Level	Blooms Description
CO1	Describe the role of AI techniques in addressing core sustainability and environmental challenges.	L2	Understand
CO2	Apply machine learning models for climate change mitigation and sustainable energy systems.	L3	Apply
CO3	Develop AI-driven solutions for sustainable agriculture, water, and waste management.	L6	Create
CO4	Analyze AI applications for biodiversity conservation, environmental monitoring, and smart cities.	L4	Analyze
CO5	Design an end-to-end AI solution for a real-world sustainability challenge with ethical considerations.	L6	Create

Generative Artificial Intelligence (RCP23CSC351) Course Contents

Unit-I 05 Hrs.

Foundations of AI for Sustainability:

Overview of artificial intelligence, introduction to sustainability and its challenges, multi-agent systems; AI techniques for sustainability including supervised learning for environmental forecasting, unsupervised learning and dimensionality reduction, reinforcement learning for resource management, the environmental impact of AI, and designing energy-efficient AI models. Case studies: climate change prediction, air quality in Beijing, and the smart water grid in Barcelona.

Unit-II 05 Hrs.

AI for Climate Change Mitigation and Sustainable Energy:

Predictive models for weather and climate patterns, machine learning for climate forecasting, AI-based risk assessment, grid optimization and energy storage, demand forecasting, carbon footprint and emission monitoring, predictive maintenance, smart grids, wind and solar power management, renewable energy monitoring, load balancing, and battery optimization.

Unit-III 04 Hrs.

AI for Sustainable Agriculture and Water Resources:

AI for irrigation and efficient water use, soil health monitoring, crop yield prediction, crop disease identification, AI-powered drones, farm management automation, water demand forecasting, drought prediction, IoT and machine learning for water management, contamination detection, and water purification. Case studies: precision agriculture and smart irrigation systems.

Unit-IV 03 Hrs.

AI in Waste Management and the Circular Economy:

Machine learning for waste classification, AI-powered smart recycling, waste-to-energy conversion, predicting energy generation from waste, optimization of waste processing plants, material recovery, waste-minimizing product design, and circular economy models for industries. Case study: AI for waste sorting and recycling.

Unit-V 05 Hrs.

AI for Biodiversity, Environmental Monitoring and Smart Cities:

Satellite imagery analysis, deforestation and land-use change detection, remote sensing, species tr[illegible] ing, endangered species protection, anti-poaching monitoring, ecosystem modeling, energy-effic[illegible]

buildings, smart transportation and traffic management, urban air quality monitoring, autonomous vehicles, and real-time urban monitoring. Case studies: deforestation monitoring and urban air quality analysis.

Unit-VI 02 Hrs.

Capstone Project – Designing an AI Solution for a Sustainability Challenge:

Problem identification and data collection, designing and implementing AI models, performance evaluation and impact assessment, ethical considerations, and synthesis of AI techniques for solving real-world sustainability challenges.

Web Resources:

https://www.coursera.org/learn/generative-ai-for-sustainability