SES

SHIRPUR EDUCATION SOCIETY

॥ सा विद्या या विमुक्तये ॥

Shirpur Education Society's

R. C. Patel Institute of Technology, Shirpur

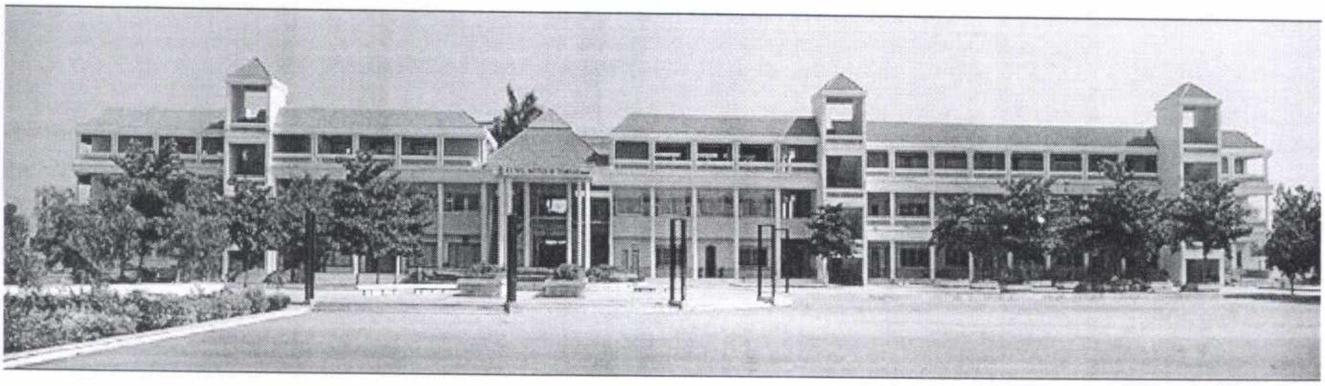
(An Autonomous Institute)

Course Structure and Syllabus

Third Year B. Tech

Artificial Intelligence and Machine Learning

With effect from Year 2025-26



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

Third Year B. Tech Artificial Intelligence and Machine Learning Semester-V (w.e.f. 2025-26)

Sr	Course Cate-gory	Course Code	Course Title	Teaching Scheme			Evaluation Scheme					Total	Credit	
							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	RCP23ACPC501	Machine Learning	3			25	15	15	15	60	100	3	4
	PC	RCP23ALPC501	Machine Learning Laboratory			2	25				25	50	1	
2	PC	RCP23ACPC502	Natural Language Processing	3			25	15	15	15	60	100	3	4
	PC	RCP23ALPC502	Natural Language Processing Laboratory			2	25				25	50	1	
3	PC	RCP23ACPC503	Image Processing and Computer Vision	3			25	15	15	15	60	100	3	4
	PC	RCP23ALPC503	Image Processing and Computer Vision Laboratory			2	25				25	50	1	
4	MD	RCP23ACMD504	Computer Network	3			25	15	15	15	60	100	3	3
5	MD	RCP23ALMD505	Programming Laboratory-III (Full Stack Development using NextJs)			2	25				25	50	1	1
6@	PE	RCP23ACPE511	Human Machine Interaction	3			25	15	15	15	60	100	3	4
		RCP23ALPE511	Human Machine Interaction Laboratory			2	25				25	50	1	
		RCP23ACPE512	Advanced Data Structures and Algorithms	3			25	15	15	15	60	100	3	
		RCP23ALPE512	Advanced Data Structures and Algorithms Laboratory			2	25				25	50	1	
		RCP23ACPE513	Recommendation Systems	3			25	15	15	15	60	100	3	
		RCP23ALPE513	Recommendation Systems Laboratory			2	25				25	50	1	
7	HS	RCP23ITHSX06	Environmental Science Tutorial		1		25					25	1	1
8	SC	RCP23IPSC501	Semester Project-III			2	25				25	50	1	1
9	SC	RCP23ALSC506	Employability Skill Development Program-I			2	25				25	50	1	1
			Total	15	1	14	325			75	475	875	23	

@Any 1 Programme Elective Course

Prepared by:
Ms. S. P. Salunkhe

Prof. Dr. U. M. Patil
BOS Chairman

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

Checked by:
Dr. P. S. Sanjekar

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

Prof. Dr. J. B. Patil
Director

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

Semester - V

Program: Artificial Intelligence and Machine Learning	**T. Y. B.Tech**	**Semester: V**
Machine Learning (RCP23ACPC501)		
Machine Learning Laboratory (RCP23ALPC501)		

Prerequisite: Knowledge of basic probability and statistics, Data Mining and Analytics concepts

Course Objective(s):

1. To understand key machine learning concepts: hypothesis formation, biasvariance trade-off, and model evaluation metrics
2. To master regression, classification, and clustering techniques
3. To apply machine learning algorithms to real-world datasets effectively

Course Outcomes:

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

CO	Course Outcomes	Blooms Level	Blooms Description
CO1	Explain the types of machine learning and issues in machine learning.	L2	Understand
CO2	Analyze model performance using evaluation metrics.	L4	Analyze
CO3	Implement and tune regression and classification algorithms.	L3	Apply
CO4	Apply knowledge of Bayesian learning principles.	L3	Apply
CO5	Apply association rule mining and clustering techniques.	L3	Apply
CO6	Explain the fundamentals of Neural Networks.	L2	Understand

Machine Learning (RCP23ACPC501)
Course Contents

Unit-I 05 Hrs.

Introduction to Machine Learning:

Types of Machine Learning, Issues in Machine Learning, Application of Machine Learning, Steps involved in developing a Machine Learning Application, Hypothesis and Inductive Bias, Bias-Variance Trade-off, Performance measures, Data Validation.

Evaluation & Selection: Metrics for Evaluating Classifier Performance, Holdout Method and Random Subsampling, Cross Validation, Bootstrap, Model Selection Using Statistical Tests of Significance, Comparing Classifiers Based on Cost–Benefit and ROC Curves.

Unit-II 08 Hrs.

Regression:

Linear Regression, Least Minimum Slope (LMS) algorithm, Gradient Descent, Lasso and Ridge Regression. Polynomial Regression. Logistic Regression, Maximum Likelihood Function.

Classification: Introduction to decision tree, Learning Decision tree using ID3 and Gini index; CART, Overfitting. Ensemble methods: Bagging (Random Forest) and Boosting (XG Boost).

Unit-III 08 Hrs.

Bayesian Learning:

Bayesian Learning, Naïve Bayes, Bayesian Network: Representation in Bayesian, Belief Network, Inference in Bayesian Network, Applications of Bayesian Network. Classification Model.

Unit-IV 05 Hrs.

Introduction to Support Vector Machine:

Support Vectors, Kernels: Linear, Polynomial and Radial Basis Function (RBF) Kernel Association Rule Mining: Market Basket Analysis, Apriori algorithm and measures of association.

Unit-V 08 Hrs.

Clustering:

Cluster Analysis and Requirements of Cluster Analysis Partitioning Methods: k-Means, k-Medoids Hierarchical Methods: Agglomerative, Divisive. Dimensionality Reduction: Dimensionality Reduction Techniques: Principal Component Analysis.

Unit-VI 05 Hrs.

Introduction to Neural Networks and Deep Learning:

Deep Learning applications, Association of biological neuron with artificial network, activation functions, weights, bias, threshold, learning rate, momentum factor.

McCulloch Pitts Neuron:

Theory and architecture; linear separability; Hebb Network: Theory and algorithm, ANN architectures. Hyperparameter tuning and batch normalization, Machine Learning vs Deep Learning.

Machine Learning Laboratory (RCP23ALPC501)

List of Laboratory Experiments

Suggested Experiments: (Any 08)

1. Perform Linear Regression.
    - Perform data cleaning
    - EDA
    - Data transformation
    - Model Training
    - Performance evaluation
2. Perform Logistic Regression.
    - Perform data cleaning
    - EDA
    - Data transformation
    - Model Training
    - Performance evaluation
3. Perform Decision Tree using GINI.
    - Data cleaning
    - EDA
    - Data transformation
    - Model Training, Visualize Decision Tree
    - Performance evaluation.
4. Perform CART decision tree algorithm
    - Data cleaning
    - EDA
    - Data transformation

- Model Training, Visualize Decision Tree
- Performance evaluation

5. Perform Ensemble methods
    - Data cleaning
    - EDA
    - Data transformation
    - Model Training
    - Performance evaluation
6. Perform Bayesian Classification
    - Data cleaning
    - EDA
    - Data transformation
    - Model Training
    - Performance evaluation
7. Compare performance of classification algorithms.
    - Model Training
    - Performance evaluation
    - Comparison of performance of different classification algorithms
8. Perform Support Vector Machine.
    - Data cleaning
    - EDA
    - Data transformation
    - Dimensionality reduction
9. Perform K-means/ K-Medoids clustering.
    - Data cleaning
    - EDA
    - Data transformation
    - Clustering
10. Study a machine learning patent.
11. 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. Mitchell, T. M., "Machine Learning", 1st Edition, McGraw Hill, 2017.
2. Ethem Alpaydın, "Introduction to Machine Learning", 4th Edition, MIT Press, 2020.
3. Peter Harrington, "Machine Learning In Action", 1st Edition, DreamTech Press, 2012.
4. Bruce, P., Bruce, A., & Gedeck, P., "Practical statistics for data scientists: 50+ essential concepts using R and Python", 2nd Edition, O'Reilly Media, 2020.

Reference Books:

1. "Data Mining for Business Analytics (An Indian Adaptation): Concepts, Techniques and Applications in Python", Cambridge University Press, ISBN NO. 978-1108727747, 2019.
2. Andreas C. Müller and Sarah Guido , "Introduction to Machine Learning with Python: A Guide for Data Scientists", O'reilly, 2016.
3. Stephen Marsland, "Machine Learning an Algorithmic Perspective", CRC Press, 2015.
4. Han Kamber, "Data Mining Concepts & Techniques", Morgann Kaufmann Publishers, 2012.
5. Kevin P. Murphy, "Machine Learning A Probabilistic Perspective", 2012.

Web Links:

1. https://onlinecourses.nptel.ac.in/noc21_cs06/preview
2. https://www.datacamp.com/tutorial/tableau-tutorial-for-beginners
3. https://www.kaggle.com/code/ekami66/detailed-exploratory-data-analysis-with-python

Program: Artificial Intelligence and Machine Learning	**T. Y. B.Tech**	**Semester: V**
Natural Language Processing (RCP23ACPC502)		
Natural Language Processing Laboratory (RCP23ALPC502)		

Prerequisite: Python Programming

Course Objective(s):

1. To introduce the fundamental concepts and techniques of Natural Language Processing for analyzing words based on Morphology and CORPUS.
2. To examine the NLP models and interpret algorithms for classification of NLP sentences by using both the traditional, symbolic and the more recent statistical approach.
3. To get acquainted with the algorithmic description of the main language levels that includes morphology, syntax, semantics, and pragmatics for information retrieval and machine translation applications.

Course Outcomes:

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

CO	Course Outcomes	Blooms Level	Blooms Description
CO1	Understand the Principles and Process of Natural Languages and real-world applications.	L2	Understand
CO2	Demonstrate understanding of state-of-the-art algorithms and techniques for text-based processing of natural language with respect to morphology.	L3	Apply
CO3	Perform POS tagging for a given natural language and select a suitable language modelling technique based on the structure of the language.	L6	Create
CO4	Check the syntactic and semantic correctness of sentences using grammars and labelling.	L5	Evaluate

Natural Language Processing (RCP23ACPC502) Course Contents

Unit-I

04 Hrs.

Introduction to Natural Language Processing:

Origin & History of NLP, Stages in NLP, Ambiguities and its types in English and Indian Regional Languages; Applications of NLP- Machine Translation, Information Retrieval, Question Answering System, Sentiment Analysis, Text Categorization , Text Summarization, Named Entity Recognition.

Unit-II

04 Hrs.

Computational tools for text analysis:

Basic Terms: Tokenization, Tokenization in the NLTK, Tokenizing text, Stemming, Lexicon free FST Porter Stemmer algorithm, Lemmatization, Natural Language Toolkit (NLTK): Corpora and other data resources. Uses of corpora: Lexicography, Grammar and syntax, Stylistics, Training and evaluation. Basic corpus analysis: Frequency distribution building and analyzing a corpus.

Unit-III

09 Hrs.

Word Level Analysis (statistical language model):

Inflectional Morphology, Derivational Morphology, Regular expression with types, Finite State Automata, NFA and DFA, Morphological Models: finite state morphology, Morphological parsing with FST (Finite State Transducer), Grams and its variation: Bigram, Trigram, Simple (Unsmoothed) N-grams; N-gram Sensitivity to the Training Corpus, Evaluating N-grams: Perplexity, smoothing: Laplace Smoothing, Good-Turing Discounting .

Unit-IV

09 Hrs.

Syntax analysis:

Part-Of-Speech tagging (POS):, Tag set for English (Upenn Treebank), Difficulties/Challenges in POS tagging , Rule-based, Stochastic and Transformation-based tagging, Generative Model: Hidden Markov Model (HMM Viterbi) for POS tagging; Issues in HMM POS tagging, Discriminative Model: Maximum Entropy model, Conditional random Field (CRF), Syntax tree vs Parse tree, Parsers: Top down and Bottom up parsers, CYK.

Unit-V

08 Hrs.

Semantic Analysis:

Lexical Semantics; Corpus study: Study of Various language dictionaries like WorldNet, Babelnet, Attachment for fragment of English, Relations among lexemes & their senses – Homonymy, Polysemy, Synonymy, Hyponymy, Semantic Ambiguity, Word Sense Disambiguation (WSD), Knowledge based approach (Lesk's Algorithm), Supervised (Naïve Bayes, Decision List).

Unit-VI 05 Hrs.

Pragmatic & Discourse Processing:

Discourse: Reference Resolution, Reference Phenomena, Syntactic & Semantic constraint on coherence, Anaphora Resolution using Hobbs and Centering Algorithm, Discourse segmentation, Coreference resolution .

Natural Language Processing Laboratory (RCP23ALPC502)

List of Laboratory Experiments

Suggested Experiments: (Any 08)

1. Preprocessing steps in NLP: Tokenization, stop word Removal, Lemmatization, stemming using NLTK and SPACY .
2. Implement Named Entity Recognition for any given text.
3. Perform morphological analysis and word generation for any given text.
4. Implement Chunking for the given input text.
5. Build a POS tagger using HMM.
6. Similarity Detection in NLP.
7. Implement N-Gram model for the given text input.
8. Generate word cloud using Python.
9. Any application of NLP: Spell Check, Autocorrect, plagiarism detection, sentiment analysis, sarcasm detection or text analytics in any domain.

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. Raymond S. T. Lee, "Natural Language Processing: A Textbook with Python Implementation", 1st Edition, 2023.
2. Lewis Tunstall, Leandro von Werra, Thomas Wolf, "Natural Language Processing with Transformers", O'Reilly , 2022.

3. Thushan Ganegedara, Andrei Lopatenko, "Natural Language Processing with TensorFlow: The definitive NLP book to implement the most sought-after machine learning models and tasks", 2nd Edition, 2022.

4. Daniel Jurafsky, James H. and Martin, Speech and Language Processing, "An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition", Pearson, 2014

Reference Books:

1. Masato Hagiwara, "Real-World Natural Language Processing: Practical applications with deep learning", Mnaning, 2021.

2. Ashish Bansal, "Advanced Natural Language Processing with TensorFlow 2: Build effective real-world NLP applications using NER, RNNs, seq2seq models, Transformers, and more", Packt Publishing, 2021.

Web Links:

1. https://medium.com/data-science-in-your-pocket/pos-tagging-using-hidden-markov-models-hmm-viterbi-algorithm-in-nlp-mathematics-explained-d43ca89347c4

2. https://towardsdatascience.com/text-generation-using-n-gram-model-8d12d9802aa0

3. https://towardsdatascience.com/how-to-create-beautiful-word-clouds-in-python-cfcf85141214

4. https://medium.com/analytics-vidhya/best-nlp-algorithms-to-get-document-similarity-a5559244b23b

5. https://towardsdatascience.com/how-to-chunk-text-data-a-comparative-analysis-3858c4a0997a

6. https://towardsdatascience.com/how-to-chunk-text-data-a-comparative-analysis-3858c4a0997a

7. https://medium.com/@raghvendra.zarkar18/natural-language-processing-65f82c8dd7e0

Online Courses and Tutorials:

1. NPTEL Course : Natural Language Processing - Course (nptel.ac.in).

2. Cousera: Natural Language Processing Specialization [4 courses] (DeepLearning.AI) — Coursera

3. Udemy: NLP - Natural Language Processing with Python — Udemy

R. C. Patel Institute of Technology, Shirpur
SES
SHIRPUR EDUCATION SOCIETY
Autonomous

Program: Artificial Intelligence and Machine Learning	**T. Y. B.Tech**	**Semester: V**
Image Processing and Computer Vision (RCP23ACPC503)		
Image Processing and Computer Vision Laboratory (RCP23ALPC503)		

Prerequisite: Basic Mathematics, Python Programming

Course Objective(s):

1. To provide a fundamental understanding of image processing concepts, including image representation, enhancement techniques, and transformation methods
2. To equip students with the knowledge of image segmentation, morphological processing, feature detection, and motion analysis for practical applications in computer vision.
3. To introduce various image compression techniques and their significance in reducing storage and transmission costs while maintaining image quality

Course Outcomes:

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

CO	Course Outcomes	Blooms Level	Blooms Description
CO1	Apply image enhancement in spatial domain	L3	Apply
CO2	Analyse image in frequency domain through different transforms and enhancement in frequency domain	L4	Analyze
CO3	Apply different image segmentation, feature detection techniques on images	L3	Apply
CO4	Examine and apply different morphological operations on an image	L3	Apply
CO5	Apply geometric transformations on an image and Evaluate optical flow algorithms	L3	Apply
CO6	Implement and evaluate image compression techniques	L3	Apply

Image Processing and Computer Vision (RCP23ACPC503) Course Contents

Unit-I

08 Hrs.

Image Processing Fundamentals and Image Enhancement:

types, Image formats, Sampling & Quantization.

Spatial Domain Techniques:

Point Processing; Digital Negative, Contrast Stretching, Thresholding, Bit Plane Slicing, Power Law Transformation, Dynamic Range Compression. Histogram Modelling; Histogram Stretching & Histogram Equalization. Neighbourhood Processing; Noise, Smoothing (Low Pass Averaging Filter, Low Pass Median Filter), Sharpening Filters (High Pass Filtering & High Boost Filtering). Mean filters, Order statistic filters

Unit-II

08 Hrs.

Image Segmentation:

Connectivity of Pixels, Detection of discontinuities (Point, Line, Edge), Detection of Edges (Computing Gradients, 1^{st} order Derivative Filters, 2^{nd} order Derivative Filters, Laplacian of Gaussian). Region-based segmentation-Region Growing, Region Splitting, Region Merging, Region Split & Merge.

Unit-III

07 Hrs.

Image Transforms & Morphological Processing:

Fourier Transform, 1D DFT, Frequency domain techniques - 2D-DFT, Low pass Filter (Ideal, Butterworth, Gaussian), High pass Filter (Ideal, Butterworth, Gaussian). Hadamard Transform, Walsh Transform.

Morphological Operations: Dilation, Erosion, Opening, Closing

Unit-IV

05 Hrs.

Geometric Transformations: Translation, Rotation, Scaling, Shearing

Feature Detection & Description:

Interest or Corner Point Detectors- Harris and Hessian. Histogram of Oriented Gradients, Scale Invariant Feature Transform (SIFT), Speeded up Robust Features (SURF), Scale-Space Analysis- Image Pyramids

Unit-V

06 Hrs.

Object Segmentation & Detection:

Canny Edge Detection, Difference of Gaussian (DOG), Graph Technique, Distance Metrics, Global

Thresholding, Otsu's Thresholding, Watershed Algorithm

Motion Analysis & Optical Flow: Introduction to Motion Detection, Lucas-Kanade, Horn-Schunck

Unit-VI **05 Hrs.**

Image Compression: Run-Length Encoding (RLE), Huffman Coding, LZW (Lempel-Ziv-Welch), Improved Grey Scale coding (IGS)

Image Processing and Computer Vision Laboratory (RCP23ALPC503)

List of Laboratory Experiments

Suggested Experiments: (Any 08)

1. Point Processing Techniques (Digital Negative, Thresholding, Intensity Transformation,Contrast Stretching).
2. Sharpening & Smoothing filters
3. Histogram Modelling
4. Frequency Domain Filtering.
5. Edge detection.
6. Morphological Operations
7. Object Detection
8. Image assessment with NumPy and OpenCV
9. Image Transformation in OpenCV
10. Feature Detection using OpenCV- Corner
11. Image Arithmetic Operations
12. Image Compression
13. Motion analysis and Action detection
14. Project Based Learning
15. Research Article Review

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. Rafael C. Gonzalez, "Digital Image Processing", 4th Edition, 2018
2. Christopher M. Bishop, "Pattern Recognition and Machine Learning", 1st Edition, 2006.
3. DRichard Szeliski, "Computer Vision: Algorithms and Applications", 2nd Edition, 2022.

Reference Books:

1. Emanuele Trucco and Alessandro Verri, "Introductory Techniques for 3-D Computer Vision", 1st Edition, 1998.
2. William K. Pratt, "Digital Image Processing: PIKS Scientific Inside", 4th Edition, 2007
3. David Forsyth and Jean Ponce, "Computer Vision: A Modern Approach", 2nd Edition, 2011.
4. E. R. Davies, "Computer and Machine Vision: Theory, Algorithms, Practicalities", 4th Edition, 2012.
5. S. Jayaraman, S. Esakkirajan, and T. Veerakumar, "Digital Image Processing", 2nd Edition, 2019.
6. Adrian Kaehler and Gary Bradski, "Learning OpenCV: Computer Vision with the OpenCV Library", 1st Edition, 2008.

Web Links:

1. https://opencv.org/
2. https://staff.fnwi.uva.nl/r.vandenboomgaard/IPCV20162017/index.html
3. https://www.geeksforgeeks.org/computer-vision/
4. https://egyankosh.ac.in/handle/123456789/90205

Online Courses:

1. **Computer Vision and Image Processing – Fundamentals and Applications:**
    - `https://onlinecourses.nptel.ac.in/noc22_ee48/preview/`
    - `https://www.coursera.org/learn/introduction-computer-vision-watson-opencv`
2. Computer Vision:
    - `https://onlinecourses.nptel.ac.in/noc19_cs58/preview`

- https://www.coursera.org/specializations/computer-vision

3. **Digital Image Processing:**

    - https://onlinecourses.nptel.ac.in/noc20_ee75/preview
    - https://www.coursera.org/learn/introduction-image-processing

Program: Artificial Intelligence and Machine Learning	**T.Y. B.Tech**	**Semester: V**
Computer Network (RCP23ACMD504)		

Course Objective(s):

1. To Master the principles of computer networking across all layers, from physical to application, including modern technologies like 5G.
2. Develop proficiency in network design, protocols, and problem-solving techniques for various networking scenarios.

Course Outcomes:

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

CO	**Course Outcomes**	**Blooms Level**	**Blooms Description**
CO1	Understanding the Demonstrate the concepts of data communication at physical layer and compare ISO - OSI model and TCP/IP model.	L2	Understand
CO2	Understand the fundamental concepts of the Data Link Layer and analyze different MAC protocols.	L4	Analyze
CO3	Design IP addressing schemes and implement routing solutions.	L3	Explain
CO4	Analyze various transport layer, application layer protocols.	L4	Apply

Computer Network (RCP23ACMD504)
Course Contents

Unit-I Introduction to Networking 06 Hrs.

Introduction to computer network, network application, network software and hardware components, Network topology, design issues for the layers. Reference models: Layer details of OSI, TCP/IP models.

Unit-II Physical Layer 05 Hrs.

Introduction to Digital Communication System: Guided Transmission Media: Twisted pair, Coaxial, Fiber optics. Unguided media: Bluetooth. Data Encoding techniques.

Unit-III Data Link Layer 08 Hrs.

DLL Design Issues (Services, Framing, Error Control, Flow Control), Error Detection and Correction (Hamming Code, Parity, CRC, Checksum), Elementary Data Link protocols: Stop and Wait, Sliding Window (Go Back N, Selective Repeat), HDLC Medium Access Control Sublayer: Channel Allocation problem, Multiple Access Protocol (Aloha, Carrier Sense Multiple Access (CSMA/CA, CSMA/CD).

Unit-IV Network Layer 10 Hrs.

Network Layer: Network Layer design issues, Communication Primitives: Unicast, Multicast, Broadcast. IPv4 Addressing (Classfull and Classless), IPv4 Protocol, Network Address Translation (NAT), IPv6.

Routing algorithms: Link state routing, Distance Vector Routing.

Routing Protocols: ARP, RARP, ICMP, IGMP, RIP, OSPF.

Congestion control algorithms: Open loop congestion control, Closed loop congestion control, Token & Leaky bucket algorithms.

Unit-V Transport Layer 06 Hrs.

The Transport Service, Port Addressing, Transport service primitives, Berkeley Sockets, Connection management (Handshake, Teardown), UDP, TCP, TCP Congestion Control: Slow Start.

Unit-VI Application Layer 04 Hrs.

DNS: Name Space, Resource Record and Types of Name Server. HTTP, SMTP, Telnet, FTP, DHCP

Textbooks:

1. A.S. Tanenbaum, "Computer Networks", 6th Edition, Pearson Education, 2020.
2. B.A. Forouzan, "Data Communications and Networking with TCP/IP Protocol Suite", 6th

Edition, TMH, 2022.

3. James F. Kurose, Keith W. Ross, "Computer Networking", A Top-Down Approach Featuring the Internet, 6th Edition, Pearson, 2017.

4. David Hanes, Jerome Henry, Rob Barton, Gonzalo Salgueiro and Patrick Grossetete, "IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things", 1/e, 2018.

Reference Books:

1. Behrouz A. Forouzan, Firouz Mosharraf, "Computer Networks: A Top-Down Approach", Mc Graw Hill, 2023.

2. Dhanashree K. Toradmalle, "Computer Networks and Network Design", Wiley, 2020.

Web Links:

1. **Web Resources:**
   - (a) https://www.netacad.com/courses/networking/networking-essentials
   - (b) https://www.coursera.org/learn/computer-networking
   - (c) https://www.edx.org/course/introduction-to-networking

2. **Online Courses: NPTEL / Swayam:**
   - (a) https://nptel.ac.in/courses/106/105/106105081
   - (b) https://nptel.ac.in/courses/106105183
   - (c) https://onlinecourses.swayam2.ac.in/cec21_cs04/preview

R. C. Patel Institute of Technology, Shirpur
SES
SHIRPUR EDUCATION SOCIETY
Autonomous

Program: Artificial Intelligence and Machine Learning	**T. Y. B.Tech**	**Semester: V**
Programming Laboratory-III (Full Stack Development using NextJs) (RCP23ALMD505)		

Prerequisite: Basic knowledge of JavaScript, React.js, and fundamental database concepts

Course Objective(s):

1. Understand the fundamentals of Next.js and its advantages over traditional React applications.
2. Learn how to build full-stack applications using Next.js and integrate various databases.
3. Implement server-side rendering (SSR), static site generation (SSG), and API routes in Next.js.
4. Explore authentication, authorization, and state management techniques.
5. Deploy Next.js applications on cloud platforms like Vercel, Netlify, or AWS.
6. Optimize web applications for performance and SEO.
7. Work with relational (MySQL, PostgreSQL) and NoSQL (MongoDB, Firebase) databases for dynamic data handling.

Course Outcomes:

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

CO	Course Outcomes	Blooms Level	Blooms Description
CO1	Demonstrate an understanding of Next.js architecture and its role in modern web development.	L3	Apply
CO2	Develop full-stack applications integrating databases and API routes.	L6	Create
CO3	Implement authentication and authorization techniques for secure web applications.	L3	Apply
CO4	Apply relational (MySQL, PostgreSQL) and NoSQL (MongoDB, Firebase) databases for dynamic data handling	L3	Apply

Programming Laboratory-III (Full Stack Development using NextJs)(RCP23ALMD505) Course Contents

Unit-I 03 Hrs.

Introduction to Next.js:

Install Next.js and configure a basic project. Explore the file structure of a Next.js project. Develop a multi-page application using the Next.js routing system. Implement dynamic and nested routes.

Unit-II 05 Hrs.

Styling and UI Design:

Use CSS Modules for component-level styling. Integrate Tailwind CSS for responsive design. Create a global layout component for consistent UI across pages.

Unit-III 05 Hrs.

Data Fetching and API Routes:

Implement getStaticProps, getServerSideProps, and getStaticPaths. Compare SSR, SSG, and CSR. Create API routes in the pages/api/ directory. Implement GET, POST, PUT, and DELETE requests.

Unit-IV 05 Hrs.

Database Integration:

Set up a MySQL or PostgreSQL database using Prisma ORM. Perform CRUD operations. Connect to MongoDB using Mongoose. Perform CRUD operations on a products collection.

Unit-V 05 Hrs.

Authentication and Authorization:

Implement Google and GitHub authentication using NextAuth.js. Manage user sessions securely. Implement different user roles (Admin, User, Guest). Restrict access to certain pages based on roles.

Unit-VI 03 Hrs.

Performance Optimization and SEO and Deployment and Cloud Services:

Implement metadata using next/head. Optimize images using next/image. Deploy a Next.js project on Vercel and Netlify. Set up environment variables for security. Use Supabase or PlanetScale for database integration. Compare performance with traditional databases.

Suggested List of Experiments:

1. Develop a Multi-Page Application with Dynamic and Nested Routes in Next.js

2. Develop a Responsive Web Design with Tailwind CSS in Next.js

3. Build a Reusable Layout Component for Consistent UI

4. Implement API Routes in Next.js: Building CRUD Operations with GET, POST, PUT, and DELETE Requests

5. Set up a database, integrate Prisma ORM, and performing CRUD operations using Prisma.

6. Connect to MongoDB with Mongoose and Perform CRUD Operations on a Products Collection in Next.js

7. Implement Google and GitHub Authentication with NextAuth.js

8. Implement user roles (Admin, User, Guest), restricting access to pages based on roles, and managing secure user sessions.

9. Optimize the Performance and SEO in Next.js using next/head and next/image for Metadata and Image Optimization

10. Deploy a Next.js Application on Vercel and Netlify: Environment Variables and Database Integration with Supabase/PlanetScale

11. Mini Project

Minimum eight experiments and mini project from the above suggested list or any other experiment based on syllabus will be included, which would help the learner to apply the concept learnt.
Practical examination will be based on the entire syllabus, including the practicals performed during laboratory sessions.

Text Books:

1. Gianni Ciolli, Boriss Mejías, Jimmy Angelakos, Vibhor Kumar, Simon Riggs, "PostgreSQL 16 Administration Cookbook", 2023.

2. Luca Ferrari, Enrico Pirozzi, "Learn PostgreSQL: Use, Manage, and Build Secure and Scalable Databases with PostgreSQL 16", Packt Publishing Ltd, 2023.

3. Roberto Rescigno, "Tailwind CSS: A Guide to Using the Popular Utility-first CSS framework", 2023.

4. Fritz Schneider and Thomas Powell, "JavaScript: The Complete Reference", 2012.

Reference Books:

1. Michele Riva, "Real-World Next.js: Build scalable, high-performance, and modern web applications using Next.js, the React framework for production", 2022.

2. Kartik Bhat, "Ultimate Tailwind CSS Handbook: Build sleek and modern websites with immersive UIs using Tailwind CSS", 2023.

3. Noel Rappin, "Modern CSS with Tailwind: Flexible Styling Without the Fuss", 2021.

Web Links:

1. `https://nextjs.org/docs`

2. `https://www.coursera.org/learn/introduction-to-mongodb`

3. `https://www.coursera.org/learn/learn-tailwind-css?`

4. `https://www.coursera.org/learn/introduction-to-next-js`

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

Program: Artificial Intelligence and Machine Learning	**T. Y. B.Tech**	**Semester: V**
Human Machine Interaction (RCP23ACPE511)		
Human Machine Interaction Laboratory (RCP23ALPE511)		

Course Objective(s):

1. This course provides an opportunity to learn and apply the design principles of Human Machine Interaction.
2. Learners will learn the basic human psychology of everyday actions and will be able to design an UI prototype of an application.
3. This course covers the discussion on various interaction design concepts.
4. The laboratory experiments are designed to practice the concepts and to adopt the systematic approach for interface design using various UX tools.

Course Outcomes:

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

CO	Course Outcomes	Blooms Level	Blooms Description
CO1	Understand Foundational Concepts of Human-Machine Interaction.	L2	Understand
CO2	Illustrate User-Centered Design Principles and Processes	L4	Analyze
CO3	Analyze and Design Effective User Interfaces.	L4	Analyze
CO3	Discover the various UX prototyping tools and its applications.	L6	Create

Human Machine Interaction (RCP23ACPE511) Course Contents

Unit-I 06 Hrs.

Introduction:

The Human: History of User Interface Designing, I/O channels, Hardware, Software and Operating environments, The Psychopathology of everyday Things, Psychology of everyday actions, Reasoning and problem solving. The computer: Devices, Memory, processing and networks. Interaction: Models, frameworks, Ergonomics, styles, elements, interactivity, Paradigms.

Unit-II 08 Hrs.

Design & Software Process:

Mistakes performed while designing a computer system, Human interaction with computers, importance of human characteristics human consideration, Human interaction speeds. Interactive Design basics, process, scenarios, navigation, Iteration and prototyping. HMI in software process: software life cycle, usability engineering, Prototyping in practice, design rationale. Design rules: principles, standards, guidelines, rules. Recognize the goals, Goal directed design process. Evaluation Techniques: Universal Design.

Unit-III 07 Hrs.

Graphical User Interface and Web Interface:

The Graphical User Interface: Popularity of graphics, the concept of direct manipulation, characteristics of GUI, Web user Interface: Interface popularity, characteristics. The merging of graphical Business systems and the Web. Principles of user interface design.

Unit-IV 07 Hrs.

Design Guidelines & Interaction Styles:

Perception, Gestalt principles, visual structure, reading is unnatural, color, vision, memory, six behavioral patterns, recognition and recall, learning, factors affecting learning, and time. Interaction Styles: Menus, Windows, Device-based and Screen- based Controls. Communication: Text messages, Feedback, and Guidance, Icons, Multimedia, and colors.

Unit-V 06 Hrs.

Communication:

Text Messages, Feedback and Guidance, Icons, Multimedia, Color and Human Vision, Colors for Textual Graphics, Stastical Graphics Screen, and Web pages.

Case Study-Voice enabled Android Application for Vehicular Complaint System, Applying Human Computer Interaction to individual security using mobile application, Machine learning Applied to Human Learning, Educative Games for Autistic Children.

Unit-VI 05 Hrs.

UX tools:

Figma, Just In Mind, and any open-source tool for prototype designing like Penpot, MockFlow, Pencil Project. Mobile Ecosystem: Platforms, Application frameworks: Types of Mobile Applications: Widgets, Applications.

Human Machine Interaction Laboratory (RCP23ALPE511)

List of Laboratory Experiments

Suggested Experiments: (Any 08)

1. A literature survey on Human Machine Interaction (based on IEEE/Scopus-Indexed Publication)
2. To Study of open-source UX tools (Justinmind Prototype, Pidoco, Marvel ,Figma Prototype) and create a simple design for a given problem definition.
3. Know your client.
   - Design an app that can teach mathematics to children of 4-5 years age in schools in Rural Sector.
   - Design an app that can teach mathematics to children of 4-5 years age in schools in Urban Sector.
   - Design a site that can help people to sell their handmade products in metro cities.
   - Design a site that can connect housewives and keep them engaged. Note : Students should be able to do the following for any given problem statement
   - Analysis of user's/client's behavior eg their preferences, interests etc
   - What kind of interfaces will they like and why?
   - Existing apps - analyze and rate them.
   - What will be your choice of screen elements?
   - How will your app/web design be better than the existing one?
4. Goal-oriented design - Design an experience for passengers whose flight /train is delayed.

5. Design Principles - Understand principles of good UI design by heuristic evaluation. Design UI for a given problem statement.

6. Menus & Navigation - Redesign of a user interface (Suggest and implement changes in Existing User Interface) for a given problem statement.
   a. Windows & Screen controls - Design UI for a given problem statement.
   b. Design a navigator for a student new in your Institute.
   c. Design a navigator for a person new in tourist city/ village.
   d. Motor paralysis for differently able people.
   e. Vaccination App design with localization

7. Icons - Design appropriate icons pertaining to a given domain. (Eg. Greeting cards, Travelling, restaurants, Education, Medical, security at Airport, Malls etc)

8. Colors – Design a personal website for any socio-technical problem. Use color guidelines with statistical graphics for better visualization.

9. Design a Map-based UI(Web User) for the given problem statement. Example: Mumbai Dabbawallas with localization feature. Pet Care New Visitors to Hospital

10. Mini Project:

    SDG 3: Good Health and Well-being

    Design a mobile application that provides accessible mental health resources (e.g., guided meditations, stress management techniques, links to support organizations) with a focus on user-centered design for diverse users, including those with disabilities or limited digital literacy. Interactive System for Promoting Healthy Habits

    SDG 4: Quality Education

    Accessible and Engaging E-Learning Platform for Underserved Communities: Design an e-learning platform with a focus on accessibility for learners in underserved communities, considering factors like low bandwidth, limited device capabilities, and varying levels of digital literacy.

    SDG 11: Sustainable Cities and Communities

    Mobile Application for Promoting Sustainable Transportation Choices: Design a mobile app that encourages users to adopt sustainable transportation options like walking, cycling, and public transport by providing real-time information, route planning, gamified challenges, and rewards.

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. Taylor Royce, "When Machines Understand Us: The Human Side of Emotional AI", 2025.
2. Kalbande, Kanade, Iyer, "Galitz's Human Machine Interaction", 1st Edition, Wiley Publications, 2015.
3. Donald A. Normann, "Design of everyday things", Basic Books, 2nd Edition, 2013.

Reference Books:

1. Rogers Sharp Preece, "Interaction Design: Beyond Human Computer Interaction", Wiley, 2019.
2. Guy A. Boy, "The Handbook of Human Machine Interaction", Ashgate publishing Ltd., 2017.

Web Links:

1. NPTEL Course: https://onlinecourses.nptel.ac.in/noc25_cs38
2. https://www.coursera.org/learn/human-computer-interaction
3. https://www.edx.org/learn/human-computer-interaction
4. Human-Computer Interaction (HCI) Course for AI Systems Design

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

Program: Artificial Intelligence and Machine Learning	**T. Y. B.Tech**	**Semester: V**
Advanced Data Structures and Algorithms (RCP23ACPE512)		
Advanced Data Structures and Algorithms Laboratory (RCP23ALPE512)		

Prerequisite: Python Programming, Data Structures, Design & Analysis of Algorithms

Course Objective(s):

1. To provide conceptual and practical knowledge of Advance Data Structures and Algorithms.
2. To Cultivate algorithmic thinking and problem-solving skills through practice and exposure to diverse problem domains.
3. Develop strategies for breaking down complex problems into manageable sub problems and applying suitablc algorithms and data structurcs.
4. Develop skills to analyze problem complexity and choose appropriate data structures and algorithms for efficient solutions.

Course Outcomes:

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

CO	Course Outcomes	Blooms Level	Blooms Description
CO1	Understand the concept of time complexity and its importance in analyzing algorithms and to Explore the complexity analysis of popular machine learning algorithms..	L2, L4	Understand, Analyze
CO2	A Explore balanced search tree data structures and spatial data structures used in geometric and spatial applications.	L4	Analyze
CO3	Apply graph algorithms to solve real-world problems related to network flows, matching, and optimization.	L3	Apply
CO4	Understand the complexity classes NP, P, NP-complete, and NP-hard and their significance in algorithm classification and to explore the computational geometry algorithms.	L2, L4	Understand, Analyze

Advanced Data Structures and Algorithms (RCP23ACPE512) Course Contents

Unit-I 08 Hrs.

Analysis of Algorithm Based on Time:

i. Amortized Analysis: Aggregate Method, Accounting Method, Potential Method (for Stack data structure)

ii. Probabilistic and Randomized Algorithm: Probabilistic approach to algorithm and Randomized Analysis, Indicator Random Variable (IRV), Analysis of Hiring Problem

Complexity Analysis of Machine Learning Algorithms:

i. Training Time Complexity and Testing Time Complexity

ii. Train/Test Complexity of Linear Regression

iii. Train/Test Complexity of Naïve Bayes Classifier

Unit-II 10 Hrs.

Balanced Search Trees:

Red-Black Tree, Tango Tree,2-3 Tree, B Tree, B+ Tree, Splay Tree

Unit-III 06 Hrs.

Advanced Data Structures:

i. Spatial Data Structure: KD Tree, R Tree

ii. Probabilistic Data Structure: Bloom filter, LogLog and HyperLogLog, Count Min sketch, MinHash with Machine Learning context (Vector Representation)

iii. Functional Data Structures: Binomial Tree, Binomial Heap

Unit-IV 06 Hrs.

Graph Based Algorithms:

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

ii. Bipartite Matching: Maximum Bipartite Matching

Unit-V 07 Hrs.

Classification of Algorithms:

i. Algorithm Classes: P, NP, NP Hardness and NP Completeness

ii. Np Completeness Proofs: Satisfiability (3 sat), Reducibility, Cook's Theorem, Traveling Salesman Problem

iii. Approximation Algorithms: Vertex Cover Problem, Travelling Salesman problem

Unit-VI 02 Hrs.

Computational Algorithms – Computational Geometry:

Line Segment Properties, Convex Hull Graham's scan algorithm

Advanced Data Structures and Algorithms Laboratory (RCP23ALPE512)

List of Laboratory Experiments

Suggested Experiments: (Any 08)

1. Experiment on Amortized Analysis.
2. To perform and implement Hiring Problem.
3. Experiment on Randomized Algorithms (Randomized Quick Sort)
4. To implement Red Black Tree creation.
5. To implement Red Black Tree deletion.
6. To implement KD Tree.
7. To implement Ford Fulkerson Algorithm.
8. To implement Approximation Algorithms (Vertex Cover)
9. Experiment on Computational Geometry Algorithms (Graham Scan)

Minimum eight experiments from the above suggested list or any other experiment based on syllabus will be included, which would help the learner to apply the concept learnt.

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", 3rd Edition, 2009.
2. S. Sridhar, "Design and analysis of algorithms", 1st Edition, Oxford, 2014.
3. Horowitz, Sahani and Rajsekaran, "Fundamentals of Computer Algorithms', 2nd Edition, Galgotia, 1998.
4. Harsh Bhasin, "Algorithms Design and Analysis", 1st Edition, Oxford, 2015.

5. Giuseppe Bonaccorso, “Machine Learning Algorithms”, Packt, 2019.

Reference Books:

1. Rajeev Motwani, “Prabhakar Raghavan, Randomized Algorithm”, Cambridge University, 2004.
2. Vijay V. Vajirani, “Approximation Algorithms”, Springcr, 2003.
3. “Computational Complexity”, Stanford University, 2010.
4. Jason Brownlee, “Master Machine Learning Algorithms”, Machine Learning Mastery, 2020.

Web Links:

1. Train/Test Complexity and Space Complexity of Linear Regression — by Writuparna Banerjee — Level Up Coding (gitconnected.com) https://levelup.gitconnected.com/train-test-complexity-and-space-complexity-of-linear-regression-26b604dcdfa3
2. Computational Complexity of ML Models — by Paritosh Kumar — Analytics Vidhya l Medium https://medium.com/analytics-vidhya/time-complexity-of-ml-models-4ec39fad2770
3. Importance of Understanding the Complexity of a Machine Learning Algorithm — by Baran Köseoğlu — Towards Data Science https://towardsdatascience.com/importance-of-understanding-the-complexity-of-a-machine-learning-algorithm-9d0532685982
4. Probabilistic Data Structures Decoded: Enhancing Performance in Modern Computing — by Naman Agrawal — Towards Data Science https://towardsdatascience.com/probabilistic-data-structures-decoded-enhancing-performance-in-modern-computing-17f700e6ea47/
5. ML Security Pro Tips: Understanding MinHash in a Security Context — by Melanie Beck — AI/ML at Symantcc — Mcdium. https://mcdium.com/ai-ml-at-symantcc/ai-ml-sccurity-pro-tips-understanding-minhash-in-a-security-context-3dd0dd2ffe8
6. Bloom Filters and when to use them — by Janko Krstic — The Little Bit Ninja — Medium https://medium.com/the-little-bit-ninja/bloom-filters-and-when-to-use-them-ab64028996d4
7. The Power of Bloom Filters: A Comprehensive Guide — by Chiranjeet Baruah — Medium https://medium.com/@chiranjeetbaruah/the-power-of-bloom-filters-a-comprehensive-guide-e7d209774b5f

Program: Artificial Intelligence and Machine Learning	**T. Y. B.Tech**	**Semester: V**
Recommendation Systems (RCP23ACPE513)		
Recommendation Systems Laboratory (RCP23ALPE513)		

Prerequisite: Statistics for Data Science, and Machine Learning.

Course Objective(s):

This course aims to provide a comprehensive understanding of recommendation system techniques, including collaborative, content-based, knowledge-based, and hybrid models. It equips students with the skills to design, implement, and evaluate personalized recommender systems across various applications using real-world data.

Course Outcomes:

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

CO	**Course Outcomes**	**Blooms Level**	**Blooms Description**
CO1	Understanding the architecture and working of Collaborative Filtering, Content based recommendation systems.	L2	Understand
CO2	Analyze and differentiate various collaborative and content-based filtering techniques, including their algorithms, similarity functions, and limitations.	L4	Analyze
CO3	Apply knowledge-based and hybrid recommendation models to develop personalized recommendation solutions using constraint-based, case-based, or hybridization strategies.	L3	Apply
CO4	Evaluate the performance of different types of recommender systems using offline and online evaluation paradigms and appropriate metrics.	L5	Evaluate

Recommendation Systems (RCP23ACPE513) Course Contents

Unit-I 06 Hrs.

Introduction to Recommender System:

Introduction to Recommendation System, Framework of recommendation systems, Eliciting Ratings and other Feedback Contributions, Implicit and Explicit Ratings, Recommender system functions. Applications of recommendation systems, Issues with recommender system

Unit-II 08 Hrs.

Collaborative filtering-based Recommender System:

Architecture of Collaborative Filtering, User-based nearest neighbour recommendation: Similarity Function, User-Based Algorithms, Item-based nearest neighbour recommendation: Similarity Function, Item-Based Algorithms, Comparing User-Based and Item-Based recommendations, data drift and concept drift.

Unit-III 07 Hrs.

Content-based Recommender System:

Architecture of content-based systems, Content representation and content similarity, Item profiles, Discovering features of documents, Obtaining item features from tags, Representing item profiles, Methods for learning user profiles, Similarity based retrieval, The Role of User Generated Content in the Recommendation Process. Bayes classifier for recommendation, Regression based recommendation system. Advantages and drawbacks of content-based filtering.

Unit-IV 06 Hrs.

Knowledge based recommendation:

Knowledge representation and reasoning, Constraint based recommenders, Case based recommenders, Persistent Personalization in Knowledge-Based Systems, Conversational Recommendation. Search based recommendation, Navigation-based recommendation.

Unit-V 06 Hrs.

Hybrid Recommendation System:

Opportunities for hybridization, Monolithic hybridization design: Feature combination, Feature augmentation, Parallelized hybridization design: Weighted, Switching, Mixed, Pipelined hybridization design: Cascade Meta level, Limitations of hybridization strategies.

Unit-VI **06 Hrs.**

Evaluating Recommendation System:

Evaluation Paradigms, General Goals of Evaluation Design, Design Issues in Offline Recommender Evaluation, Online Recommender evaluation techniques. Comparison between evaluation design of classification model and recommendation system, Error metrics, Decision-Support metrics, User Centered metrics. Comparative analysis between different types of recommendation systems.

Recommendation Systems Laboratory (RCP23ALPE513)

List of Laboratory Experiments

Suggested Experiments: (Any 08)

1. Build a Recommendation Engine with Item-Based Collaborative Filtering.
2. Build a Recommendation Engine with User-Based Collaborative Filtering.
3. Build Content-based recommendation engine on different datasets.
4. Build recommender system using association rule mining.
5. Implement Recommendation System using K-Nearest Neighbours.
6. Build Context-Aware Recommender Systems.
7. Build Constraint-based Recommenders.
8. Implement knowledge-based recommender system.
9. Implement a Monolithic hybridization design.
10. Evaluate the recommendation system with evaluation matrix.
11. Compare the performance of different recommender systems.
12. Mini Projects:

    1) Build a recommender system that suggests SDG-related content (articles, actions, or products) based on user preferences or content similarity.

    2) Build a recommender system that suggests eco-friendly products or services aligned with specific SDGs.

    3) Build a recommender system that suggests books aligned with specific SDGs, helping users find educational or informative literature related to sustainability, equality, or climate action.

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. C.C. Aggarwal, "Recommender Systems: The Textbook", 1st Edition, Springer, 2016.
2. Jannach D., Zanker M. and FelFering A., "Recommender Systems: An Introduction", 1st Edition, Cambridge University Press, 2011.
3. Ricci, F., Rokach, L., & Shapira, B., "Introduction to Recommender Systems Handbook", Springer, Boston, MA 2011.

Reference Books:

1. P. Pavan Kumar, S. Vairachilai, Sirisha Potluri, "Recommender Systems: Algorithms and Applications", 1st Edition, CRC Press, 2021.
2. Kim Falk, "Practical Recommender Systems", 1st Edition, Manning, 2019.
3. Rounak Banik, "Hands-On Recommendation Systems with Python: Start building", Ingram short title, 2018.
4. M. D. Ekstrand, J.T. Riedl, J.A. Konstan, "Collaborative filtering recommender systems", 1st Edition, Now publishers, 2011.
5. P J. Leskovec, A. Rajaraman and J. Ullman, "Mining of massive datasets", 2nd Edition, Cambridge, 2012.
6. Rounak Banik, "Hands-On Recommendation Systems with Python: Start building powerful and personalized, recommendation engines with Python", 2018.

Web Links:

1. https://tryolabs.com/blog/introduction-to-recommender-systems
2. https://medium.com/@deepapandithu/recommender-system-user-collaborative-filtering-37613f0c6a9
3. https://www.analyticsvidhya.com/blog/2015/08/beginners-guide-learn-content-basedrecommender-systems/
4. https://www.researchgate.net/publication/2378325Knowledge-Based_Recommender_Systems
5. https://medium.com/analytics-vidhya/7-types-of-hybrid-recommendation-system3e4f78266ad8
6. https://www.researchgate.net/publication/226264572EvaluatingRecommendationSystems

Program: Artificial Intelligence and Machine Learning	**T.Y. B.Tech**	**Semester:V**
Environmental Science Tutorial (RCP23ITHSX06)		

Prerequisite: Interest in Environment and its impact on Human Calculus.

Course Objective(s):

1. Familiarise students with environment related issues such as depleting resources, pollution, ecological problems and the renewable energy scenario.
2. 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	Understand how human activities affect environment.	L2	Understand
CO2	Understand the various technology options that can make a differencc.	L2	Understand

Environmental Science Tutorial (RCP23ITHSX06) Course Contents

Unit-I Air Pollution 01 Hr.

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

Unit-II Water Pollution 01 Hr.

Sources of water pollution. Ground water pollution and eutrophication.

Unit-III Noise Pollution 01 Hr.

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

Unit-IV Biodiversity loss 01 Hr.

Value of Biodiversity. Endangered species.

Unit-V Deforestation 01 Hr.

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

Unit-VI Renewable Energy sources 01 Hr.

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

Unit-VII Climate change 01 Hr.

Greenhouse gases and climate change.

Unit-VIII Green Technology 01 Hr.

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

Textbooks:

1. R. Rajagopalan, "Environmental Studies from Crisis to Cure", 2^{nd} Edition, Oxford University Press.
2. Erach Bharucha, "Textbook of Environmental Studies for Undergraduate Courses" for University Grants Commission, New Delhi & Bharti Vidyapeeth Institute of Environment Education and Research.
3. P. Narayanan, "Environmental Pollution: Principles, Analysis and Control", CBS Publishers.

4. Mohammad Dastbaz, Colin Pattinson, Babak Akhgar, Morgan and Kaufman, Elsevier, "Green Information Technology: A Sustainable Approach".

Reference Books:

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

Websites:

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_polution
5. Ecosan: www.eco-solutions.org

List of Tutorials:

1. Case study on Smog.
2. Presentation on Water Pollution (Industrial, Sewage) explaining any specific case.
3. List effects of noise pollution on human health. Measure decibel level in college library, canteen, classroom.
4. Case study on effect of pollution on Biodiversity loss.
5. Debate for and against to promote Economic Growth Deforestation is required.
6. Presentation on different Renewable Energy Technologies.
7. Report on major impact of Global warming on Environment giving real examples.
8. Report on advantages and examples of Green Building for Sustainable development, Sustainable Software Design.

The final certification and acceptance of term work will be subject to satisfactory laboratory work performance and fulfilling minimum passing criteria in the term work.

Program: Artificial Intelligence and Machine Learning	**T. Y. B.Tech**	**Semester: V**
Semester Project-III (RCP23IPSC501)		

Course Objective(s):

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:

CO	**Course Outcomes**	**Blooms Level**	**Blooms Description**
CO1	Conduct a survey of several available literatures in the preferred field of study.	L4	Analyze
CO2	Demonstrate various/alternate approaches to complete a project.	L2	Understand
CO3	Ensure a collaborative project environment by interacting and dividing project work among team members.	L3	Apply
CO4	Present their project work in the form of a technical report / paper and thereby improve the technical communication skill.	L3	Apply
CO5	Demonstrate the ability to work in teams and manage the conduct of the research study.	L2	Understand

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 Third 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

R. C. Patel Institute of Technology, Shirpur
SES
SHIRPUR EDUCATION SOCIETY
Autonomous

- 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.

Each group shall present/publish a paper based on the semester project in reputed/peer reviewed Conference/Journal/TechFest/Magazine before end of the semester.

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: Artificial Intelligence and Machine Learning	**T. Y. B.Tech**	**Semester: V**
Employability Skill Development Program-I (RCP23ALSC506)		

Prerequisite: Basic knowledge of C programming.

Course Objectives:

1. To implement and apply linear and non-linear data structures to solve real-world problems.
2. To design, code, and test data structures and algorithms with efficiency in mind.
3. To analyze time and space complexity through hands-on problem-solving.
4. To implement and compare searching and sorting techniques for different datasets.
5. To apply data structures in practical tasks or mini-projects for software solutions.

Course Outcomes:

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

CO	Course Outcomes	Blooms Level	Blooms Description
CO1	Analyze and compute time and space complexity for various algorithms	L3	Apply
CO2	Implement linear and non-linear data structures in a programming language.	L3	Apply
CO3	Solve real-world coding problems using suitable data structures.	L4	Analyze
CO4	Implement and compare searching and sorting algorithms on different datasets.	L3	Apply

Employability Skill Development Program-I (RCP23ALSC506) Course Contents

Unit-I Time Complexity 03 Hrs.

- Introduction to time complexity
- Applying time complexity
- Log and square root time complexity
- Practice time complexity

Unit-II Linked Lists 04 Hrs.

- Introduction to Arrays
- Arrays Practice
- Singly linked list
- Deletion in linked lists
- Circular and doubly linked list

Unit-III Stacks and Queues 04 Hrs.

- How to implement a stack
- How to implement a queue
- Stack and Queue applications

Unit-IV Trees and Binary trees 05 Hrs.

- Basic tree concepts
- Trees: Practice problems
- Binary trees
- BT: Practice problems
- Binary search trees
- BST: Practice problems

Unit-V Graphs 05 Hrs.

- Basic concepts in Graphs

- Graph types
- Graph representation
- Graph Traversal
- Dijkstra's algorithm

Unit-VI Searching, Sorting and Hashing 05 Hrs.

- Searching
- Practice Searching
- Sorting
- Bubble sort
- Selection sort
- Insertion sort
- Merge sort
- Quick sort
- Radix sort
- Hashing

Suggested List of Experiments:

1. Implementation of Linked List operations.
2. Implementation of different operations on Linked List –copy, concatenate, split, reverse, count no. of nodes etc.
3. Implementation of polynomials operations (addition, subtraction) using Linked List.
4. Implementation of stack.
5. Implementation of Infix to Postfix conversion.
6. Implementation of Prefix and Postfix evaluation using stack.
7. Implementation of parenthesis checker using stack.
8. Implementation of Linear queue.
9. Implementation of Circular queue.
10. Implementation of Double ended queue.

11. Implementation of Priority queue program using array and Linked list.
12. Implementation of Binary Tree.
13. Implementation of Binary Tree Traversal.
14. Implementation of BST using following operations – create, delete, display.
15. Implementation of various operations on tree like – copying tree, mirroring a tree, counting the number of nodes in the tree, counting only leaf nodes in the tree.
16. Implementation of Graph traversal (DFS & BFS).
17. Implementations of Selection and Radix sort.
18. Implementation of Heap & Heap Sort.
19. Implementation of Merge Sort and Quick Sort.
20. Implementation of Bubble Sort.
21. Implementation of searching methods (Index Sequential, Fibonacci search, Binary Search)
22. Implementation of hashing functions with different collision resolution techniques.
23. Implementation of Insert and Delete operation in an array.
24. Implement the program to perform sum of array elements.
25. Implement the program to find maximum of array elements.
26. Implement the program to print all the leaders in the array.
27. Implement the program to find the sub array with the largest sum, and print its sum.
28. Implement the program to remove duplicates elements from Linked List.
29. Implement the program to Insert the node at the end in Circular Linked List.
30. Implement the program to delete the node from Circular Linked List.
31. Implement the program to insert the node in Double Linked List.
32. Implement the program to delete the node from Double Linked List.
33. Implement a stack using arrays to solve a classic problem: converting an integer from decimal to binary using the ”divisor-remainder” method.
34. Implement a program to print numbers in a specific pattern using a queue.
35. Implement a program that simulates a to-do list manager using a queue.

36. Implement a program to find whether a given string is palindrome or not using stack.

37. Implement a program to find whether a given string is palindrome or not using queue.

38. Implementation of Adjacency Matrix for Tree.

39. Implementation of Adjacency List for Tree.

40. Implementation of Depth-First Search (DFS) tree traversal algorithm.

41. Implementation of Breadth-First Search (BFS) tree traversal algorithm.

42. Implement a program find the number of leaf nodes using DFS algorithm.

43. Implement a program to find the height of the tree for undirected connected tree.

44. Implement a program to print all the nodes at the level K from root node in sorted order given undirected connected tree.

45. Implement a program to check if there exists a path from root to any leaf whose sum is equal to targetSum for undirected connected tree.

46. Implement a program to count the number of neighbours of the node v for undirected connected tree.

47. Implement a program to find the subtree sum of each node i for undirected connected tree.

48. Implement a program to find the length of the diameter of the tree for undirected connected tree.

49. Implement a program to find the lowest common ancestor (LCA) for undirected connected tree.

50. Implement a program to find the distance between two nodes for undirected connected tree.

51. Implement a program to find the Kth ancestor of the node v for undirected connected tree.

52. Implement a program to find the largest node value at each level for undirected connected tree.

53. Implement a program to determine for each node the number of distinct colours in the subtree of the node for an undirected connected tree with N coloured nodes.

54. Implement a program to determine the minimum time in minutes you have to spend to collect all coins in the tree, starting at root node 1 and coming back to it for an undirected connected tree with N nodes and some of the nodes has gold coins in it.

55. Implement a program to find the height of it given connected binary tree with N nodes.

56. Implement a program to determine binary tree is height-balanced.

57. Implement a program to check given two undirected binary trees are the identical or not.

58. Implement a program to print the level order traversal of binary tree nodes.

59. Implement a program to print the zigzag level order traversal of binary tree nodes.

60. Implement a program to print the top view of the binary tree.

61. Implement a program to print binary tree minimum depth.

62. Implement a program to construct and return a binary tree using two integer arrays, preorder and inorder.

63. Implement a program to find the sum of leaves at deepest level in a binary tree.

64. Implement a program to check if the second tree T2 exists as a subtree in the first tree T1 in the given two undirected binary trees T1 and T2.

65. Implement a program to find the lowest common ancestor of the three nodes in a given binary tree with three nodes.

66. Implement a program to retrieve all root-to-leaf paths where the sum of the node values along the path equals the given integer target. Print each such path on separate lines for a binary tree and an integer target.

67. Implement a program to search if X exists in the BST or not.

68. Implement a program to print its nodes in sorted order in a binary search tree.

69. Implement a program to find the maximum/largest node in a binary search tree.

70. Implement a program to return the minimum difference between the values of any two different nodes in a binary search tree.

71. Implement a program to check if the given binary tree is a valid Binary Search Tree (BST).

72. Implement a program to find the Kth largest node in BST.

73. Implement a program to determine the inorder predecessor of a given node X in a binary search tree.

74. Implement a program to check if there exist two elements in the BST such that their sum is equal to S.

75. Implement a program to recover the tree without changing its structure for given a binary search tree, where exactly two nodes of the tree were swapped by mistake.

76. Implement a program to find the LCA (lowest common ancestor) of these two nodes using BST.

77. Implementation of Adjacency Matrix for Graph.

78. Implementation of Adjacency List for Graph.

79. A Cloud Provider's network spans across n locations with m connections between them. Your task is to determine whether Chef can send a message to Chefina. If possible, you need to find the minimum number of servers on such a route.

80. Given an undirected and unweighted graph and two nodes x and y, Implement a program to find the length of the shortest path between the two nodes. If no path exists, return -1.

81. Given a string s1, a character c1, and an integer k, Implement a program to find and print the position of the kth occurrence of the character c1 in the string s1. If the kth occurrence does not exist, print -1.

82. Implement a program to find the smallest and largest elements in an array of integers.

83. Implement a program to find the element in an array with the smallest absolute difference from a given integer k. If there are multiple elements with the same minimum difference, print the smallest of these elements.

84. Implement a program to find and print all pairs of integers from a list of n pairs where the sum of each pair is divisible by k.

85. Implement a program that reads an integer n followed by n pairs of integers. Given two additional integers left and right, the program should print all pairs whose sum and product fall within the inclusive range [left, right].

86. Implementation a program to find frequency of each element in the array.

87. Implementation a program to find frequency of elements using Hashing.

88. Implementation a program to Count Beautiful Pairs using Hashing.

89. Implementation a Hashing function using Division method.

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

Reference Books:

1. R. F. Gilberg and B. A. Forouzan, "Data Structures – A Pseudocode Approach with C", 2^{nd} Edition, Cengage Learning, 2005.

2. Ellis Horowitz, Sartaj Sahni and Susan Anderson-Freed, "Fundamentals of Data Structures in C", 2^{nd} Edition, W. H. Freeman, and Company 2008.

3. Mark A. Weiss, "Data Structures and Algorithm Analysis in C", 4^{th} Edition, Pearson, 2014.

4. M. T. Goodritch, R. Tamassia, D. Mount, "Data Structures and Algorithms in C++", 2^{nd} Edition, Wiley, 2011.

5. Kruse, Leung, Tondo, "Data Structures and Program Design in C", 2nd Edition, Pearson Education, 2013.

6. Tenenbaum, Langsam, Augenstein, "Data Structures using C", 2nd Edition Pearson, 2015.

7. Reema Thareja, "Data Structures using C", Oxford, 2017.

8. Seymour Lipschutz, "Data Structures, Schaum's Outline Series", 1st Edition, Tata McGraw-Hill, 2014.