



VIGNAN'S INSTITUTE OF MANAGEMENT AND TECHNOLOGY FOR WOMEN

(An Autonomous Institution)

[Sponsored by Lavu Educational Society, Affiliated to JNTUH & Approved by AICTE, New Delhi]
Kondapur (V), Ghatkesar (M), Medchal - Malkajgiri (D) - 501 301. Phone: 96529 10002/3



COMPUTER SCIENCE AND ENGINEERING (AI&ML)

COURSE STRUCTURE – VR24

II B.Tech. – I Semester

S.No.	Course Code	Course	L	T	P	Credits
1	ML301PC	Software Engineering	3	0	0	3
2	ML302PC	Data Structures	3	0	0	3
3	ML303PC	Computer Organization and Architecture	3	0	0	3
4	MA304BS	Mathematical and Statistical Foundations	3	1	0	4
5	ML305PC	Operating Systems	3	0	0	3
6	ML306PC	Introduction to Data Structures Lab	0	0	2	1
7	ML307PC	Operating Systems Lab	0	0	2	1
8	ML308PC	Software Engineering Lab	0	0	2	1
9	ML309PC	Node JS/ React JS/ Django	0	0	2	1
10	*MC310	Constitution of India	3	0	0	0
		Total	18	0	10	20

II B.Tech. - II Semester

S.No.	Course Code	Course	L	T	P	Credits
1	ML401PC	Discrete Mathematics	3	0	0	3
2	ML402PC	Automata Theory and Compiler Design	3	0	0	3
3	ML403PC	Database Management Systems	3	0	0	3
4	ML404PC	Introduction to Artificial Intelligence	3	0	0	3
5	ML405PC	Object Oriented Programming through Java	3	0	0	3
6	ML406PC	Database Management Systems Lab	0	0	2	1
7	ML407PC	Java Programming Lab	0	0	2	1
8	ML408PC	Real-time Research Project/Field-Based Research Project	0	0	4	2
9	ML409PC	Prolog/ Lisp/ Pyswip	0	0	2	1
10	*MC410	Gender Sensitization Lab	0	0	2	0
		Total	15	0	12	20

Note: L - Theory T - Tutorial P - Practical C - Credits

B.Tech. CSE(AI&ML)	L	T	P	C
II Year - I Semester	3	0	0	3

SOFTWARE ENGINEERING

Pre-requisite: Computer Science Fundamentals

Course Objectives:

- The aim of the course is to provide an understanding of the working knowledge of the techniques for estimation, design, testing and quality management of large software development projects.
- Topics include process models, software requirements, software design, software testing, software process/ product metrics, risk management, quality management and UML diagrams

Course Outcomes

- Define a software process, Acquire knowledge about different software development processes and their usability indifferent problem domains.
- Identify different architectural styles and explain the procedure to design an architecture using requirement specifications.
- Assess the quality of the product using a few quality standards and metrics.
- Test the software system by applying various types of testing strategies and testing techniques.
- Estimate the software system by applying various metrics and estimation techniques.

UNIT-I

Introduction to Software Engineering: The evolving role of software, changing nature of software, software myths.

A Generic view of process: Software engineering- a layered technology, a process framework, the capability maturity model integration (CMMI).

Process models: The waterfall model, Spiral model and Agile methodology

UNIT- II

Software Requirements: Functional and non-functional requirements, user requirements, system requirements, interface specification, the software requirements document.

Requirements engineering process: Feasibility studies, requirements elicitation and analysis, requirements validation, requirements management.

UNIT- III

Design Engineering: Design process and design quality, design concepts, the design model. Creating an architectural design: software architecture, data design, architectural styles and patterns, architectural design, conceptual model of UML, basic structural modeling, class diagrams, sequence diagrams, collaboration diagrams, use case diagrams, component diagrams.

UNIT-IV

Testing Strategies: A strategic approach to software testing, test strategies for conventional software, black-box and white-box testing, validation testing, system testing, the art of debugging.

Metrics for Process and Products: Software measurement, metrics for software quality.

UNIT-V

Risk management: Reactive Vs proactive risk strategies, software risks, risk identification, risk projection, risk refinement, RMMM.

Quality Management: Quality concepts, software quality assurance, software reviews, formal technical reviews, statistical software quality assurance, software reliability, the ISO 9000 quality standards.

TEXTBOOKS:

1. Software Engineering, A practitioner's Approach-Roger S. Pressman, 6th edition, McGraw Hill International Edition.
2. Software Engineering-Sommerville, 7th edition, Pearson Education.

REFERENCEBOOKS:

1. The unified modeling language user guide Grady Booch, James Rumbaugh, Ivar Jacobson, Pearson Education.
2. Software Engineering, an Engineering approach-James F.Peters, Witold Pedrycz, John Wiley.
3. Software Engineering principles and practice-Waman S Jawadekar, The McGraw-Hill Companies.
4. Fundamentals of object-oriented design using UML Meiler page

B.Tech. CSE(AI&ML)	L	T	P	C
II Year - I Semester	3	0	0	3

DATA STRUCTURES

Pre-requisites: Programming for Problem Solving

Course Objectives:

- Exploring basic data structures such as stacks and queues.
- Introduces a variety of data structures such as hash tables, search trees, tries, heaps, graphs.
- Introduces sorting and pattern matching algorithms
- Strengthen the ability to identify and apply the suitable data structures for the given real-world problem.
- Gain knowledge in practical applications of data structures

Course Outcomes:

- Developing ADT is necessary for solving problems based on Stacks and Queues.
- Implement binary trees, general tree structures, advanced search trees, heaps, graphs.
- Implement hash functions and handle collisions.
- Implement various kinds of sorting techniques and apply appropriate techniques for solving a given problem.
- Understand Pattern matching and tries in application oriented

UNIT-I

Introduction to Data Structures, abstract data types, Linear list — singly linked list implementation, insertion, deletion and searching operations on linear list, Stacks-Operations, array and linked representations of stacks, stack applications, Queues-operations, array and linked representations.

UNIT- II

Dictionaries: linear list representation, skip list representation, operations - insertion, deletion and searching.

Hash Table Representation: hash functions, collision resolution-separate chaining, open addressing- linear probing, quadratic probing, double hashing, rehashing, extendible hashing.

UNIT- III

Search Trees: Binary Search Trees, Definition, Implementation, Operations-Searching, Insertion and Deletion, B-Trees, B+ Trees, AVL Trees, Definition, Height

of an AVL Tree, Operations— Insertion, Deletion and Searching, Red– Black, Splay Trees.

UNIT-IV

Graphs: Graph Implementation Methods. Graph Traversal Methods.

Sorting: Quick Sort, Heap Sort, External Sorting-Model for external sorting, Merge Sort.

UNIT-V

Pattern Matching and Tries: Pattern matching algorithms-Brute force, the Boyer– Moore algorithm, the Knuth-Morris-Pratt algorithm, Standard Tries, Compressed Tries, Suffix tries.

TEXTBOOKS:

1. Fundamentals of Data Structures in C, 2nd Edition, E. Horowitz, S. Sahni and Susan Anderson Freed, Universities Press.
2. Data Structures using C– A. S. Tanenbaum, Y. Langsam, and M. J. Augenstein, PHI/Pearson Education.

REFERENCE BOOK:

1. Data Structures: A Pseudocode Approach with C, 2nd Edition, R. F. Gilberg and B. A. Forouzan, Cengage Learning.

B.Tech. CSE(AI&ML)

L T P C

II Year - I Semester

3 0 0 3

COMPUTER ORGANIZATION AND ARCHITECTURE

Pre-requisite: A Course on “Digital Electronics”.

Course Objectives:

- The purpose of the course is to introduce principles of computer organization and the basic architectural concepts.
- It begins with basic organization, design, and programming of a simple digital computer and introduces simple register transfer language to specify various computer operations.
- Topics include computer arithmetic, instruction set design, microprogrammed control unit, pipelining and vector processing, memory organization and I/O systems, and multiprocessors

Course Outcomes:

- Understand the basics of instruction sets and their impact on processor design.
- Demonstrate an understanding of the design of the functional units of a digital computer system.
- Evaluate cost performance and design trade-offs in designing and constructing a computer processor including memory.
- Design a pipeline for consistent execution of instructions with minimum hazards.
- Recognize and manipulate representations of numbers stored in digital computers

UNIT-I

Digital Computers: Introduction, Block diagram of Digital Computer, Definition of Computer Organization, Computer Design and Computer Architecture.

Register Transfer Language and Microoperations: Register Transfer language, Register Transfer, Bus and memory transfers, Arithmetic Micro operations, logic micro-operations, shift micro operations, Arithmetic logic shift unit.

Basic Computer Organization and Design: Instruction codes, Computer Registers Computer instructions, Timing and Control, Instruction cycle, Memory Reference Instructions, Input – Output and Interrupt.

UNIT- II

Microprogrammed Control: Control memory, Address sequencing, micro program example, design of control unit.

Central Processing Unit: General Register Organization, Instruction Formats, Addressing

modes, Data Transfer and Manipulation, Program Control.

UNIT- III

Data Representation: Data types, Complements, Fixed Point Representation, Floating Point Representation.

Data Representation: Addition and subtraction, multiplication Algorithms, Division Algorithms, Floating —point Arithmetic operations. Decimal Arithmetic unit, Decimal Arithmetic operations.

UNIT-IV

Input-Output Organization: Input-Output Interface, Asynchronous data transfer, Modes of Transfer, Priority Interrupt Direct memory Access.

Memory Organization: Memory Hierarchy, Main Memory, Auxiliary memory, Associate Memory, Cache Memory.

UNIT-V

Reduced Instruction Set Computer: CISC Characteristics, RISC Characteristics.

Pipeline and Vector Processing: Parallel Processing, Pipelining, Arithmetic Pipeline, Instruction Pipeline, RISC Pipeline, Vector Processing, Array Processor.

Multi Processors: Characteristics of Multiprocessors, Interconnection Structures, Inter-processor arbitration, Inter-processor communication and synchronization, Cache Coherence.

TEXTBOOK:

1. **Computer System Architecture**— M.Morris Mano, Third Edition, Pearson/PHI.

REFERENCE BOOKS:

1. Computer Organization—Carl Hamacher, Zvonks Vranesic, SafeaZaky, VthEdition, McGraw Hill.
2. Computer Organization and Architecture— William Stallings Sixth Edition, Pearson/PHI.
Structured Computer Organization— Andrew S.Tanen

B.Tech. CSE(AI&ML)	L	T	P	C
II Year - I Semester	3	1	0	4

MATHEMATICAL AND STATISTICAL FOUNDATIONS

Pre-requisites: Basic Calculus and Linear Algebra

Course Objectives:

- The Number Theory basic concepts useful for cryptography etc.
- Linear Regression and Correlation into a model
- The theory of Probability, and probability distributions of single and multiple random variables
- The sampling theory and testing of hypothesis and making inferences
- Familiarize students with the fundamental concepts of stochastic processes, emphasizing their role in modeling systems influenced by randomness.

Course Outcomes: The student will learn.

- Apply the number theory concepts to cryptography domain
- Apply the concepts of probability and distributions to some case studies
- Correlate the material of one unit to the material in other units
- Resolve the potential misconceptions and hazards in each topic of study.
- Describe the Markov property and its significance in the context of stochastic modeling.

UNIT - I

Greatest Common Divisors and Prime Factorization: Greatest common divisors, The Euclidean algorithm, The fundamental theorem of arithmetic, Factorization of integers and the Fermat numbers

Congruences: Introduction to congruences, Linear congruences, The Chinese remainder theorem, Systems of linear congruences

UNIT - II

Simple Linear Regression and Correlation: Introduction to Linear Regression, The Simple Linear Regression Model, Least Squares and the Fitted Model, Properties of the Least Squares Estimators, Inferences Concerning the Regression Coefficients, Prediction, Simple Linear Regression Case Study

Random Variables and Probability Distributions: Concept of a Random Variable, Discrete Probability Distributions, Continuous Probability Distributions, Statistical Independence.

Discrete Probability Distributions: Binomial Distribution, Poisson distribution.

UNIT - III

Continuous Probability Distributions: Normal Distribution, Areas under the Normal Curve, Applications of the Normal Distribution, Normal Approximation to the Binomial

Fundamental Sampling Distributions: Random Sampling, Sampling Distributions, Sampling Distribution of Means and the Central Limit Theorem, Sampling Distribution of S^2 , t-Distribution, F-Distribution.

UNIT - IV

Estimation & Tests of Hypotheses: Introduction, Statistical Inference, Classical Methods of Estimation. Estimating the Mean, Standard Error of a Point Estimate, Prediction Intervals, Tolerance Limits, Estimating the Variance, Estimating a Proportion for single mean, Difference between Two Means, between Two Proportions for Two Samples and Maximum Likelihood Estimation.

UNIT - V

Stochastic Processes and Markov Chains: Introduction to Stochastic processes-Markov process. Transition Probability, Transition Probability Matrix, First order and Higher order Markov process, n-step transition probabilities, Markov chain, Steady state condition, Markov analysis.

.

TEXT BOOKS:

1. Kenneth H. Rosen, Elementary number theory & its applications, sixth edition, Addison- Wesley, ISBN 978 0-321-50031-1
2. Ronald E. Walpole, Raymond H. Myers, Sharon L. Myers, Keying Ye, Probability & Statistics for Engineers & Scientists, 9th Ed. Pearson Publishers.
3. S.D.Sharma, Operations Research, Kedarnath and Ramnath Publishers, Meerut, Delhi

REFERENCE BOOKS:

1. S C Gupta and V K Kapoor, Fundamentals of Mathematical statistics, Khanna publications
2. T.T. Soong, Fundamentals of Probability And Statistics For Engineers, John Wiley & Sons Ltd, 2004.
3. Sheldon M Ross, Probability and statistics for Engineers and scientists, Academic Press.

B.Tech. CSE(AI&ML)	L	T	P	C
II Year - I Semester	3	0	0	3

OPERATING SYSTEMS

Prerequisites:

1. A course on “Computer Organization and Architecture”.

Course Objectives:

- Understand the working of computer system and the basic concepts of operating System and the services provided by it.
- Understand the functions and management of different resources of the operating system (Processor, I/O, and Memory etc).
- Understand process management concepts including scheduling, synchronization, Deadlocks.
- Learn the mechanisms involved in memory management and I/O subsystems of an Operating system.
- Understand issues of protection and security.

Course Outcomes:

- Explain the fundamental concepts and functions of operating system.
- Understand process scheduling in a multi-programming environment and implement Process scheduling algorithms.
- Write application and system calls related programs for managing processes, memory, I/O and inter-process Communication related system calls.
- Understand memory management, disk management techniques, including virtual memory and file system structure.
- Explain protection and security related issues of the computer system.

UNIT-I

Operating System - Introduction, Structures - Simple Batch, Multi-programmed, Time-shared, Personal Computer, Parallel, Distributed Systems, Real-Time Systems, System components, Operating System services, System Calls

Process - Process concepts and scheduling, Operations on processes, Cooperating Processes, Threads

UNIT- II

CPU Scheduling - Scheduling Criteria, Scheduling Algorithms, Multiple -Processor Scheduling. System call interface for process management-fork, exit, wait, wait pid, exec

Deadlocks- System Model, Deadlocks Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, and Recovery from Deadlock

UNIT- III

Process Management and Synchronization-The Critical Section Problem, Synchronization Hardware, Semaphores, and Classical Problems of Synchronization, Critical Regions, Monitors

Interprocess Communication Mechanisms: IPC between processes on a single computer system, IPC between processes on different systems, using pipes, FIFOs, message queues, shared memory.

UNIT-IV

Memory Management and Virtual Memory - Logical versus Physical Address Space, Swapping, Contiguous Allocation, Paging, Segmentation, Segmentation with Paging, Demand Paging, Page Replacement Algorithms.

UNIT-V

File System Interface and Operations- Access methods, Directory Structure, Protection, File System Structure, Allocation methods, Free-space Management. Usage of open, create, read, write, close, lseek, stat, ioctl system calls.

TEXTBOOKS:

1. Operating System Principles-Abraham Silberchatz, Peter B.Galvin, GregGagne 7th Edition, John Wiley.
2. Advanced programming in the UNIX environment, W. R. Stevens, Pearson education.

REFERENCE BOOKS:

1. Operating Systems-Internals and Design Principles, William Stallings, Fifth Edition–2005, Pearson Education/PHI
2. Operating System A Design Approach-Crowley, TMH.
3. Modern Operating Systems, Andrew S.Tanenbaum 2nd edition, Pearson/PHI
4. UNIX programming environment, Kernighan and Pike, PHI/Pearson Education
5. UNIX Internals-The New Frontiers, U.Vahalia Pearson Education.

B. Tech. CSE(AI&ML)	L	T	P	C
II Year – I Semester	0	0	2	1

INTRODUCTION TO DATA STRUCTURES LAB

Prerequisites: A Course on “Programming for problem solving”.

Course Objectives:

- It covers various concepts of C programming language
- It introduces searching and sorting algorithms
- It provides an understanding of data structures such as stacks and queues.
- Strengthen the ability to identify and apply the suitable data structures for the given real-world problem
- Gain knowledge in practical applications of data structures.

Course Outcomes:

- Implement string matching and various data structures using arrays, linked lists.
- Developing ADT is necessary for solving problems based on Stacks and Queues.
- Implement binary trees, general tree structures, advanced search trees, heaps, graphs.
- Implement hash functions and handle collisions.
- Implement various kinds of sorting techniques and apply appropriate techniques for solving a given problem.
- Ability to Implement searching and sorting algorithms

Note: The lab experiments will be like the following experiment examples.

List of Experiments:

1. Write a program that uses functions to perform the following operations on a single linked list.: i) Creation ii) Insertion iii) Deletion iv) Traversal
2. Write a program that uses functions to perform the following operations on a doubly linked list.: i) Creation ii) Insertion iii) Deletion iv) Traversal
3. Write a program that uses functions to perform the following operations on circular linked list.: i) Creation ii) Insertion iii) Deletion iv) Traversal
4. Write a program that implements stack (its operations) using i) Arrays ii) Pointers
5. Write a program that implements Queue (its operations) using i) Arrays ii) Pointers
6. Write a program that implements the following sorting methods to sort a given list of integers in ascending order
i) Quick sort ii) Heap sort iii) Merge sort
7. Write a program to implement the tree traversal methods (Recursive and Non-Recursive).

8. Write a program to implement
 - i) Binary Search tree ii) B Trees iii) B+ Trees iv) AVL trees v) Red - Black trees
9. Write a program to implement the graph traversal methods.
10. Implement a Pattern matching algorithms using Boyer- Moore, Knuth-Morris-Pratt

TEXTBOOKS:

1. Fundamentals of Data Structures in C, 2nd Edition, E. Horowitz, S. Sahni and Susan Anderson Freed, Universities Press.
2. Data Structures using C – A. S. Tanenbaum, Y. Langsam, and M. J. Augenstein, PHI/Pearson Education.

REFERENCE BOOK:

1. Data Structures: A Pseudocode Approach with C, 2nd Edition, R. F. Gilberg and B. A. Forouzan, Cengage Learning

B. Tech. CSE(AI&ML)	L	T	P	C
II Year – I Semester	0	0	2	1

OPERATING SYSTEMS LAB

Prerequisites: A course on “Programming for Problem Solving”, A course on “Computer Organization and Architecture”.

Course Objectives:

- Understand the usage of various Linux commands.
- Understand the usage of threads management and Inter process communication System calls.
- Implement various page replacement, CPU scheduling algorithms and disk Scheduling algorithms.
- Understand and implement semaphores and Dead lock avoidance.
- Analyse and understand Disk Scheduling Algorithms.

Course Outcomes:

- Explore the LINUX low level I/O and construct applications using process management and file management System calls.
- Demonstrate how threads can be created and simultaneously handled in LINUX POSIX environment.
- Understand possible Inter-Process Communication implementations using LINUX IPC Constructs.
- Assess the working behaviour of various synchronization approaches used in Deadlock management.
- Analyse the performance of process scheduling algorithms, page replacement algorithms, and Disk scheduling algorithms.

List of Experiments:

1. Write C programs to simulate the following CPU Scheduling algorithms a) FCFS
b) SJF c) Round Robin d) priority
2. Write programs using the I/O system calls of UNIX/LINUX operating system
(open, read, write, close, fcntl, seek, stat, opendir, readdir)
3. Write a C program to simulate Bankers Algorithm for Deadlock Avoidance and Prevention.
4. Write a C program to implement the Producer – Consumer problem using semaphores using UNIX / LINUX system calls.
5. Write C programs to illustrate the following IPC mechanisms
a) Pipes b) FIFOs c) Message Queues d) Shared Memory
6. Write C programs to simulate the following memory management techniques
a) Paging b) Segmentation

7. Write C programs to simulate Page replacement policies

- a) FCFS b) LRU c) Optimal

TEXTBOOKS:

1. Operating System Principles- Abraham Silberchatz, Peter B. Galvin, Greg Gagne 7th Edition, John Wiley Advanced programming in the Unix environment, W.R.Stevens, Pearson education.

REFERENCE BOOK:

1. Operating Systems – Internals and Design Principles, William Stallings, Fifth
2. Edition–2005, Pearson Education/PHI
3. Operating System - A Design Approach-Crowley, TMH.
4. Modern Operating Systems, Andrew S Tanenbaum, 2nd edition, Pearson/PHI
5. UNIX Programming Environment, Kernighan and Pike, PHI/Pearson Education
6. UNIX Internals: The New Frontiers, U. Vahalia, Pearson Education

B. Tech. CSE(AI&ML)	L	T	P	C
II Year – I Semester	0	0	2	1

SOFTWARE ENGINEERING LAB

Prerequisites: A course on “Programming for Problem Solving”.

Course Objectives:

- To understand the software engineering methodologies for project development.
- To gain knowledge about open-source tools for Computer Aided Software Engineering (CASE).
- To develop test plans and test cases to perform various testing.

Course Outcomes: Student will be able to

- Ability to translate end-user requirements into system and software requirements.
- Ability to generate a high-level design of the system from the software requirements.
- Will have experience and/or awareness of testing problems and will be able to develop a simple testing report.
- Translate design specifications into executable code using appropriate programming languages and tools, followed by rigorous testing to ensure quality and reliability.
- Employ project management techniques, including version control, task scheduling, and team collaboration, to oversee software development projects effectively.

Note: The lab experiments will be like the following experiment examples.

List of Experiments:

Do the following seven exercises for any two projects given in the list of sample projects or any other Projects:

1. Development of problem statements.
2. Preparation of Software Requirement Specification Document, Design Documents and Testing Phase related documents.
3. Preparation of Software Configuration Management and Risk Management related documents.
4. Study and usage of any Design phase CASE too
5. Performing the Design by using any Design phase CASE tools.
6. Develop test cases for unit testing and integration testing
7. Develop test cases for various white box and black box testing techniques.

Sample Projects:

1. Passport automation System
2. Book Bank
3. Online Exam Registration
4. Stock Maintenance System
5. Online course reservation system
6. E-ticketing
7. Software Personnel Management System
8. Credit Card Processing
9. E-book management System.
10. Recruitment system

TEXT BOOKS:

1. Software Engineering, A practitioner's Approach- Roger S. Pressman, 6th edition, McGraw Hill International Edition.
2. Software Engineering- Sommerville, 7th edition, Pearson Education.
3. The unified modeling language user guide Grady Booch, James Rumbaugh, Ivar Jacobson, Pearson Education.

REFERENCE BOOKS:

1. Software Engineering, an Engineering approach- James F. Peters, Witold Pedrycz, John Wiley.
2. Software Engineering principles and practice- Waman S Jawadekar, The McGraw-Hill.

B. Tech. CSE(AI&ML)	L	T	P	C
II Year – I Semester	0	0	2	1
NODE JS/ REACT JS/ DJANGO Lab				

Pre-requisite: Basic HTML Tags, Java

Course Objectives:

- To implement the static web pages using HTML and do client-side validation using JavaScript.
- To design and work with databases using Java
- To develop an end-to-end application using java full stack.
- To introduce Node JS implementation for server-side programming.
- To experiment with single page application development using React.

Course Outcomes: At the end of the course, the student will be able to,

- Build a custom website with HTML, CSS, Bootstrap and little JavaScript.
- Demonstrate Advanced features of JavaScript and learn about JDBC
- Develop Server – side implementation using Java technologies like
- Develop the server – side implementation using Node JS.
- Design a Single Page Application using React.

Note: The lab experiments will be like the following experiment examples.

Exercises:

1. Build a responsive web application for shopping cart with registration, login, catalog and cart pages using CSS3 features, flex and grid.
2. Make the above web application responsive web application using Bootstrap framework
3. Use JavaScript for doing client – side validation of the pages implemented in experiment 1 and experiment 2.
4. Explore the features of ES6 like arrow functions, callbacks, promises, async/await. Implement an application for reading the weather information from openweathermap.org and display the information in the form of a graph on the web page.
5. Develop a java standalone application that connects with the database (Oracle / mySql) and perform the CRUD operation on the database tables
6. Create an xml for the bookstore. Validate the same using both DTD and XSD
7. Design a controller with servlet that provides the interaction with application developed in experiment 1 and the database created in experiment 5.
8. Maintaining the transactional history of any user is very important. Explore the various session tracking mechanism (Cookies, HTTP Session)

9. Create a custom server using the http module and explore the other modules of Node JS like OS, path, event.
10. Develop an express web application that can interact with REST API to perform CRUD operations on student data. (Use Postman)
11. For the above application create authorized end points using JWT (JSON Web Token).
12. Create a react application for the student management system having registration, login, contact, about pages and implement routing to navigate through these pages.
13. Create a service in react that fetches the weather information from openweathermap.org and the display the current and historical weather information using graphical representation using chart.js
14. Create a TODO application in react with necessary components and deploy it into github

REFERENCE BOOKS:

1. Jon Duckett, Beginning HTML, XHTML, CSS, and JavaScript, Wrox Publications, 2010
2. Bryan Basham, Kathy Sierra and Bert Bates, Head First Servlets and JSP, O'Reilly Media, 2nd Edition, 2008.
3. Vasam Subramanian, Pro MERN Stack, Full Stack Web App Development with Mongo, Express, React, and Node, 2nd Edition, A Press.

B.Tech.	L	T	P	C
II Year - I Semester	3	0	0	0

CONSTITUTION OF INDIA
(COMMON TO CSE, CSE(DS), CSE(AI&ML), IT & ECE)

Pre-requisite: Nil

Course Objectives: Students will be able to:

- To educate the students on Democratic values and Principles articulated in the Constitution.
- To help the students understand the structure of the Union, State and local governments.
- To create awareness among the students about the Fundamental Rights and Duties.
- To make the students understand the nature and characteristics of relations between union and state governments.
- To divulge the students about statutory institutions and policies.

Course Outcomes: Students will be able to:

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

Unit-I:

History of Making of the Indian Constitution-History of Drafting Committee.

Unit-II:

Philosophy of the Indian Constitution-Preamble Salient Features

Unit-III

Contours of Constitutional Rights & Duties- Fundamental Rights

- Right to Equality
- Right to Freedom

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

Unit-IV

Organs of Governance: Parliament, Composition, Qualifications and Disqualifications, Powers and Functions, Executive, President, Governor, Council of Ministers, Judiciary, Appointment and Transfer of Judges, Qualifications, Powers and Functions

Unit - V

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

Panchayatraj: Introduction, PRI: Zila Panchayat. Elected officials and their roles, CEO

Zila Panchayat: Position and role. Block level: Organizational Hierarchy (Different departments), Village level: Role of Elected and Appointed officials, Importance of grass root democracy

Unit - VI

Election Commission: Election Commission: Role and Functioning. Chief Election Commissioner and Election Commissioners.

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

Suggested Reading:

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