Shri Vile Parle Kelavani Mandal's

Dwarkadas J. Sanghvi College of Engineering

(Autonomous College Affiliated to the University of Mumbai)

Scheme and detailed syllabus (DJS22)

Second Year B.Tech

in

Computer Science and Engineering (Data Science)

(Semester III)

Shri Vile Parle Kelavani Mandal's
DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING
(Autonomous College Affiliated to the University of Mumbai)
NAAC Accredited with "A" Grade (CGPA : 3.18)

Proposed scheme for Second Year B.Tech Program for Department of Computer Science and Engineering (Data Science)
Semester III
(Academic Year 2023-2024)

Sr	Course Code	Course	Teaching Scheme (hrs.)				Continuous Assessment (A) (marks)			Semester End Assessment (B) (marks)					(A+B)	Total Credits	
			Th	P	T	Credits	Th	T/W	Total CA (A)	Th/Cb	O	P	O&P	Total SEA (B)			
1	DJS22DSC301	Mathematics for Intelligent Systems	3	--	--	3	35	--	35	65	--	--	--	65	100	3	4
	DJS22DST301	Mathematics for Intelligent Systems - Tutorial	--	--	1	1	--	25	25	--	--	--	--	--	25	1	
2	DJS22DSC302	Data Structures	3	--	--	3	35	--	35	65	--	--	--	65	100	3	4
	DJS22DSL302	Data Structures Laboratory	--	2	--	1	--	25	25	--	--	--	25	25	50	1	
3	DJS22DSC303	System Fundamentals	3	--	--	3	35	--	35	65	--	--	--	65	100	3	4
	DJS22DSL303	System Fundamentals Laboratory	--	2	--	1	--	25	25	--	--	--	25	25	50	1	
4	DJS22DSC304	Database Management System	3	--	--	3	35	--	35	65	--	--	--	65	100	3	4
	DJS22DSL304	Database Management System Laboratory	--	2	--	1	--	25	25	--	--	--	--	--	25	1	
5	DJS22DSL305	Python Laboratory	1	2	--	2	--	25	25	--	--	--	25	25	50	2	2
6	DJS22DSL306	Web Engineering Laboratory	--	4	--	2	--	50	50	--	--	--	--	--	50	2	2
7	DJS22ILLA1	Innovative Product Development-I	--	2	--	--	--	--	--	--	--	--	--	--	--	--	--
8	DJS22A2	Constitution of India	1	--	--	--	--	--	--	--	--	--	--	--	--	--	--
		Total	**14**	**14**	**1**	**20**	**140**	**175**	**315**	**260**	**0**	**0**	**75**	**335**	**650**	**20**	**20**

Th	Theory	**T/W**	Termwork
P	Practical	**O**	Oral
T	Tuturial	**Cb**	Computer based

Prepared by Checked by Head of the Department Principal

Continuous Assessment (A):

Course	Assessment Tools	Marks	Time (hrs.)
Theory	a. One Term test (based on 40 % syllabus)	20	1
	b. Second Term test (next 40 % syllabus) / presentation / assignment / course project / group discussion / any other.	15	1
	Total marks (a + b)	35	--
Audit course	Performance in the assignments / qui / power point presentation / poster presentation / group project / any other tool.	--	As applicable
Laboratory	Performance in the laboratory and documentation.	25	
Tutorial	Performance in each tutorial & / assignment.	25	
Laboratory &Tutorial	Performance in the laboratory and tutorial.	50	

The final certification and acceptance of term work will be subject to satisfactory performance upon fulfilling minimum passing criteria in the term work / completion of audit course.

Semester End Assessment (B):

Course	Assessment Tools	Marks	Time (hrs.)
Theory / * Computer based	Written paper based on the entire syllabus.	65	2
	* Computer based assessment in the college premises.		
Oral	Questions based on the entire syllabus.	25	as applicable
Practical	Performance of the practical assigned during the examination and the output / results obtained.	25	2
Oral & Practical	Project based courses - Performance of the practical assigned during the examination and the output / results obtained. Based on the practical performed during the examination and on the entire syllabus.	As per the scheme	2

Prepared by Checked by Head of the Department Principal

Program: B.Tech in Computer Science and Engineering (Data Science)	S.Y. B.Tech	Semester: III
Course: Mathematics for Intelligent Systems (DJS22DSC301)		
Course: Mathematics for Intelligent Systems - Tutorial (DJS22DST301)		

Pre-requisite:

1. Concepts of basic matrices
2. Partial derivatives
3. Basic probability

Objectives:

To build the strong foundation in learners of mathematics needed for building concepts of machine learning.

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

1. Analyze probability of random variable and probability distributions.
2. Demonstrate knowledge of linear algebra.
3. Apply concepts of matrix theory.
4. Demonstrate concepts of calculus.
5. Analyze different optimization techniques.

Mathematics for Intelligent Systems (DJS22DSC301)		
Unit	**Description**	**Duration**
1	**Probability, Random variables and Probability Distributions Probability:** Conditional probability, mutually and pair wise independent events, Bayes' theorem Random variables: discrete random variable, probability mass function, discrete distribution function, continuous random variable, probability density function, continuous distribution function, mathematical expectation, moment generating function, two-dimensional random variable and its joint probability mass and density function, marginal distribution function, conditional distribution functions, covariance, joint moments Probability distributions: discrete probability distribution: binomial distribution, poisson distribution, hypergeometric distribution, continuous probability distribution: uniform distribution, exponential distribution, normal distribution, beta distribution, gamma distribution, central limit theorem	**10**
2	**Linear algebra:** Vectors in n-dimensional vector space, properties, dot product, cross product, norm and distance, vector spaces over real field, properties of vector spaces over real field, subspaces, linear independence and dependence of vectors, span of vectors, basis of a vector space, dimension of a vector space, Cauchy Schwarz inequality, linear transformation, Norms and spaces, orthogonal compliments and projection operator, Kernel Hilbert spaces.	**08**
3	**Matrix theory:** Characteristic equation, Eigen values and Eigen vectors, properties of Eigen values and Eigen vectors, Cayley-Hamilton theorem, examples based on verification of Cayley Hamilton theorem, similarity of matrices, diagonalization of matrices, functions	**08**

	of square matrix, derogatory and nonderogatory matrices, least squared and minimum normed solutions	
4	**Calculus:** Gradient, directional derivatives, Jacobian, hessian, convex sets, convex functions and its properties.	**04**
5	**Optimization:** Unconstrained and Constrained optimization, convergence Unconstrained optimization techniques: newton's method, quasi newton method Constrained optimization techniques: gradient descent, stochastic gradient descent, penalty function method, lagrange multiplier method, Karush–Kuhn–Tucker method, simplex method, penalty and duality, dual simplex method, downhill simplex method.	**12**
	Total	**42**

Mathematics for Intelligent Systems - Tutorial (DJS22DST301)	
Exp.	**Suggested experiments**
1	To solve numerical on discrete probability distributions
2	To solve numerical on continuous probability distributions
3	To solve numerical on vector spaces (basis and dimension)
4	To solve numerical on cauchy-schwarz inequality and linear transformation
5	To solve numerical on diagonalizability using eigenvalues and eigenvectors
6	To solve numerical on minimal polynomial and functions of a matrix
7	To solve numerical on calculus
8	To solve numerical on Gradient descent and Lagrange's multiplier method
9	To solve numerical on KKT method
10	To solve numerical on all forms of simplex method

Minimum eight tutorials based on syllabus will be conducted. Mini project relevant to the subject may be included, which would help the learner to apply the concept learnt.

Books Recommended:

Text books:

1. Dr. B. S. Grewal, Higher Engineering Mathematics, Khanna Publication, 44th Edition, 1965.
2. Kanti B. Datta, Mathematical Methods in Science and Engineering, Cengage Learning India, 1st Edition, 2011.
3. Hamdy A. Taha, Operations Research - An Introduction, Pearson, 10th Edition, Publication Year 2010.
4. P. K. Gupta, Mohan Man, Operations Research, Kanti Swarup, S Chand Publication, 1st Edition, 2005.

Reference Books:

1. W. Cheney, Analysis for Applied Mathematics, New York: Springer Science Business Media, 1st Edition, 2001.
2. S. Axler, Linear Algebra Done Right, Springer International Publishing, 3rd Edition, 2015.
4. J. Nocedal and S. J. Wright, Numerical Optimization, New York: Springer Science Business

Media, 2nd Edition, 2006.

5. J. S. Rosenthal, A First Look at Rigorous Probability Theory, Singapore: World Scientific Publishing, 2nd Edition, 2006.
6. Seymour Lipschutz and Marc Lipson, Linear Algebra‖ Schaum‘s outline series, Mc-Graw Hill Publication, 4th Edition, 2009.
7. Erwin Kreysizg, Advanced Engineering Mathematics, John Wiley & Sons Inc, 10th Edition, 2000.

Program: B.Tech in Computer Science and Engineering (Data Science)	S.Y. B.Tech	Semester: III
Course: Data Structures (DJS22DSC302)		
Course: Data Structures Laboratory (DJS22DSL302)		

Pre-requisite:

1. C-Programming

Objectives:

To introduce and familiarize students with linear and non-linear data structures, their use in fundamental algorithms and design & implementation of these data structures. To introduce students to the basics of algorithms and time complexity. To familiarize students to various sorting and searching techniques, and their performance comparison.

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

1. Understand the concept of time and space complexity for algorithms.
2. Assimilate the concept of various linear and non-linear data structures.
3. Solve the problem using appropriate data structure.
4. Implement appropriate searching and sorting technique for a given problem.

Data Structures (DJS22DSC302)		
Unit	**Description**	**Duration**
1	**Basics of Algorithms:** Algorithms, Characteristics of an Algorithm, Time and Space Complexities, Order of Growth functions, Preliminary Asymptotic Notations. **Data Structures:** Introduction, need of Data Structures, Types of Data Structures, Abstract Data Types (ADT)	**04**
2	**Linear Data Structures – LIST:** List as an ADT, Array-based implementation, Linked List implementation, singly linked lists, circularly linked lists, doubly-linked lists, All operations (Insertion, Deletion, Merge, Traversal, etc.) and their analysis, Applications of linked lists - (Polynomial Addition).	**06**
3	**Linear Data Structure – STACK:** Stack as an ADT, Operations, Array and Linked List representation of Stack, Applications – Reversing data, Conversion of Infix to prefix and postfix expression, Evaluation of postfix and prefix expressions, balanced parenthesis, etc.	**04**
4	**Linear Data Structure – QUEUE:** Queue as an ADT, Operations, Implementation of Linear Queue, Circular and Priority Queue using arrays and Linked List, DEQueue, Applications – Queue Simulation.	**04**
5	**Non-Linear Data Structure – TREES:** Tree Terminologies, Tree as an ADT, Binary Tree - Operations, Tree Traversals, Binary Search Tree (BST) - Operations, Expression Trees **Height Balanced Tree**: Creation of AVL Tree, **Heap**: Operations on heap **Applications**: Huffman coding	**10**

6	**Non Linear Data Structure – GRAPHS:** Graph Terminologies, Types of Graphs, Representation of Graph using arrays and Linked List, Breadth-First Search (BFS), Depth–First Search (DFS), **Applications of Graphs** -Topological sorting.	**06**
7	**Searching**- Linear Search, Binary Search and Fibonacci search. **Sorting:** Bubble Sort, Selection Sort, Heap Sort, Insertion Sort, Radix Sort, Merge Sort, Quick Sort. Analysis of Searching and Sorting Techniques. **Hashing:** Hash Functions, Overflow handling, Collision & Collision Resolution Techniques, Linear hashing, Hashing with chaining, Separate Chaining, Open Addressing, Rehashing and Extendible hashing.	**08**
	Total	**42**

Data Structures Laboratory (DJS22DSL302)	
Exp.	**Suggested experiments**
1	Implementations of Linked List using menu driven approach.
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	Implementations of stack using menu driven approach.
5	Implementations of Infix to Postfix conversion.
6	Implementation of prefix and postfix evaluation using menu driven approach.
7	Implementation of parenthesis checker using stack.
8	Implementations of Linear queue using menu driven approach.
9	Implementations of circular queue using menu driven approach.
10	Implementations of double ended queue using menu driven approach.
11	Implementation of Priority queue program using array and Linked list.
12	Implementations of Binary Tree using menu driven approach.
13	Implementation of Binary Tree Traversal.
14	Implementations of BST.
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	Implementations of Graph traversal using menu driven program (DFS & BSF).
17	Implementations of Selection sort, Radix sort using menu driven.
18	Implementations of Heap & Heap Sort using menu driven program.
19	Implementations of Advanced Bubble Sort and Insertion Sort using menu driven Program.
20	Implementations of searching methods (Index Sequential, Fibonacci search, Binary Search) using menu driven program.
21	Implementation of hashing functions with different collision resolution techniques.

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

*The Term Work will be calculated based on Laboratory Performance (15m) and Quizzes (10m).

Books Recommended:

Text books:

1. R. F. Gilberg and B. A. Forouzan, Data Structures – A Pseudocode Approach with C, 2nd Edition, Cengage Learning, 2005.
2. Ellis Horowitz, Sartaj Sahni and Susan Anderson-Freed, Fundamentals of Data Structures in C, 2nd Edition, W. H. Freeman and Company 2008.

Reference Books:

1. Mark A. Weiss, Data Structures and Algorithm Analysis in C, 4th Edition, Pearson, 2014.
2. M. T. Goodritch, R. Tamassia, D. Mount, Data Structures and Algorithms in C++, Wiley, 2nd Edition, 2011.
3. Kruse, Leung, Tondo, Data Structures and Program Design in C, Pearson Education, 2nd Edition, 2013.
4. Tenenbaum, Langsam, Augenstein, Data Structures using C, Pearson, 2nd Edition, 2015.
5. Aho, Hopcroft, Ullman, Data Structures and Algorithms, Addison-Wesley, 2010.
6. Reema Thareja, Data Structures using C, Oxford, 2017.
7. Seymour Lipschutz, Data Structures, Schaum's Outline Series, Tata McGraw-Hill, 1st Edition, 2014.

Program: B.Tech in Computer Science and Engineering (Data Science)	S.Y. B.Tech	Semester: III
Course: System Fundamentals (DJ22DSC303)		
Course: System Fundamentals Laboratory (DJ22DSL303)		

Pre-requisite:

1. Basic Mathematics

Objectives:

To understand the structure, functions and characteristics of computer system and operating systems.

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

1. Describe the fundamental organization of a computer system.
2. Apply appropriate memory mapping, process scheduling and disk scheduling methods.
3. Identify the need of concurrency and apply appropriate method to solve the concurrency or deadlock problem.
4. Differentiate between various processor architecture.

System Fundamentals (DJS22DSC303)		
Unit	**Description**	**Duration**
1	**Introduction to System Fundamentals:** Realization of half adder and full adder using Logic Gates, Von Neumann model, Fixed point representation, Register Transfer and Micro-operations: Floating point representation, Arithmetic Micro- Operations, Arithmetic logical shift unit. Addition and subtraction, Multiplication Algorithms (Booth Multiplication Algorithm), Division Algorithms, Floating Point Arithmetic operations, Instruction Cycle with interrupt and DMA. **Operating System Architecture:** Basic functions and services, System calls, Types of Operating Systems: Batch, multiprogramming. Multitasking, time sharing, parallel, distributed & real-time O.S.	**08**
2	**Central Processing Unit:** Introduction, General Register Organization, Stack Organization, Instruction format, Addressing Modes, data transfer and manipulation, Program Control, Reduced Instruction Set Computer (RISC). **Process Management:** Process Concept, Process states, Process control Block, Threads, Uni-processor Scheduling: Types of scheduling: Pre-emptive, Non pre-emptive, Scheduling algorithms: FCFS, SJF, RR, Priority. Comparative study of process management in Windows, Linux and Android OS.	**06**
3	**Memory Organization:** Memory Hierarchy, Main Memory, Cache Memory, Memory Mapping, cache coherence, Pentium IV cache organization, ARM cache organization. **Memory Management:** Memory partitioning: Fixed and Variable Partitioning, Memory Allocation: Allocation Strategies (First Fit, Best Fit, and Worst Fit), Fragmentation, Swapping, Virtual Memory, Paging. Segmentation, Demand paging and Page replacement policies. Comparative study of memory management in Windows, Linux and Android OS.	**08**

4	**Concurrency control** **Concurrency:** Principles of Concurrency, Mutual Exclusion: S/W approaches, H/W Support, Semaphores, Monitors, Classical Problems of Synchronization: Readers-Writers and Producer Consumer problems and solutions. **Deadlock:** Principles of deadlock, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Dining Philosopher problem. Comparative study of concurrency control in Windows, Linux and Android OS.	**10**
5	**File and I/O management:** File access methods, I/O Devices, Organization of I/O functions, Operating System Design issues, I/O Buffering, Disk Scheduling (FCFS, SCAN, C-SCAN, SSTF), RAID, Disk Cache, Arbitration methods, Comparative study of file and I/O management in Windows, Linux and Android OS.	**04**
6	**Advance Computer Architecture:** Characteristics of Multiprocessors, Flynn's taxonomy, Parallel processing architectures and challenges, Hardware multithreading, Multicore and shared memory multiprocessors, Introduction to Graphics Processing Units, Introduction to Multiprocessor network topologies.	**06**
	Total	**42**

System Fundamentals Laboratory (DJS22DSL303)	
Exp.	**Suggested experiments**
1	Implement Booth's multiplication algorithm.
2	Implement CPU Non-Preemptive scheduling algorithms like FCFS, SJF, Priority etc.
3	Implement CPU Preemptive scheduling algorithms like SRTF, Round Robin, Preemptive priority etc.
4	Explore the internal commands of Linux.
5	Write shell scripts handling File, Directory, Networking and security aspects.
6	Implement Best Fit, First Fit and Worst Fit Memory allocation policy.
7	Implement Fully associative and set associative cache memory mapping.
8	Implement various cache/page replacement policies.
9	Implement order scheduling in supply chain using Banker's Algorithm.
10	Implement Disk Scheduling Algorithms.
	Study Experiments
11	Implement Restoring and Non-Restoring division algorithm.
12	Implement Solution to Producer Consumer Problem of Process Synchronization.
13	Implement Solution to Reader Writer Problem of Process Synchronization.
14	Implement Solution to Dinning Philosopher Problem of Process Synchronization.
15	Implementation of Multithreading using parent process and child process using UNIX calls like fork, exec and wait.

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

*The Term Work will be calculated based on Laboratory Performance (15m) and Assignments (10m).

Books Recommended:

Text books:

1. William Stallings, Computer Organisation and Architecture, Pearson publication, 11th Edition, 2018.
2. Greg Gagne, Abraham Silberschatz, John Wiley & Sons, Inc. Operating System Concepts, 10th Eedition Peter B. Galvin, 2018.

Reference Books:

1. John Hayes, Computer Architecture and Organization, McGrawHill, 3rd Edition, 2017.
2. M. Morris Mano, Computer System Architecture, Pearson, 2017.
3. Andrew S. Tanenbaum and Todd Austin, Structured Computer Organization, 6th Edition, PHI, 2016
4. M. Murdocca & V. Heuring, Computer Architecture & Organization, Wiley, 2017.
5. Andrew S. Tanenbaum, Modern Operating Systems, PHI, 2009
6. G. Meike, Lawrence Schiefer, Inside the Android OS: Building, Customizing, Managing and Operating Android System Services (Android Deep Dive), 2021.

Program: B.Tech in Computer Science and Engineering (Data Science)	**S.Y. B.Tech**	**Semester: III**
Course: Database Management System (DJS22DSC304)		
Course: Database Management System Laboratory (DJS22DSL304)		

Pre-requisite:

1. Computer Basics

Objectives:

To introduce the students to the management of database systems, with an emphasis on how to design, organize, maintain and retrieve information efficiently and effectively from a database.

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

1. Design an optimized database.
2. Construct SQL queries to perform operations on the database.
3. Demonstrate appropriate transaction management and recovery techniques for a given problem.
4. Apply indexing mechanisms for efficient retrieval of information from database.

Database Management System (DJS22DSC304)		
Unit	**Description**	**Duration**
1	**Introduction Database Concepts:** Introduction, Characteristics of databases, File system v/s Database system, Users of Database system, Schema and Instance, Data Independence, DBMS system architecture, Database Administrator.	**03**
2	**Relational Data Model:** **Entity–Relationship Model:** The Entity-Relationship (ER) Model: Entity types: Weak and strong entity sets, Entity sets, Types of Attributes, Keys, Relationship constraints: Cardinality and Participation, Extended Entity-Relationship (EER) Model: Generalization, Specialization and Aggregation. **Relational Model:** Introduction to the Relational Model, relational schema and concept of keys, Mapping the ER and EER Model to the Relational Model. **Relational Algebra:** Unary and Set operations, Relational Algebra Queries	**09**
3	**Structured Query Language (SQL):** Overview of SQL, Data Definition Commands, Data Manipulation commands, Integrity constraints - key constraints, Domain Constraints, Referential integrity, check constraints, Data Control commands, Transaction Control Commands, Set and String operations, aggregate function - group by, having, Views in SQL, joins, Nested and complex queries, Triggers, Security and authorization in SQL	**09**
4	**Relational–Database Design:** Pitfalls in Relational-Database designs, Concept of normalization, Function Dependencies, Normal Forms- 1NF, 2NF, 3NF, BCNF	**05**
5	**Transaction Management and Recovery:** Transaction Concept, ACID properties, Transaction States, Implementation of atomicity and durability, Concurrent Executions, Serializability, Concurrency Control Protocols: Lock-based, Timestamp based, Validation Based, Deadlock Handling,	**09**

	Recovery System: Failure classification, Log based recovery, Shadow Paging, ARIES recovery algorithm.	
6	**Indexing Mechanism:** Hashing techniques, Types of Indexes: Single Level Ordered Indexes, Multilevel Indexes, Overview of B-Trees and B+ Trees	**04**
	Total	**42**

Database Management System Laboratory (DJS22DSL304)	
Exp.	**Suggested experiments**
1	To draw an ER diagram for a problem statement.
2	Map the ER/EER to relational schema.
3	To implement DDL and DML commands with integrity constraints.
4	To access & modify Data using basic SQL.
5	To implement Joins and Views.
6	To implement Subqueries.
7	To implement triggers.
8	Examine the consistency of database using concurrency control technique (Locks)
9	To simulate ARIES recovery algorithm.
10	To implement B-trees/B+ trees.

Minimum eight experiments from the above-suggested list or any other experiment or mini project based on syllabus will be included, which would help the learner to apply the concept learnt.
*The Term Work will be calculated based on Laboratory Performance (15m) and Computer Based Assessment (10m).

Books Recommended:

Text books:

1. Korth, Silberschatz, Sudarshan, "Database System Concepts", McGraw – Hill, 7th Edition, 2019.
2. Elmasri and Navathe, "Fundamentals of Database Systems", Pearson Education, 7th Edition, 2021.
3. G. K. Gupta: "Database Management Systems", 3rd Edition, McGraw – Hill.

Reference Books:

1. Raghu Ramkrishnan and Johannes Gehrke, "Database Management Systems", TMH
2. Sharnam Shah, "Oracle for Professional", SPD.
3. Dr. P.S. Deshpande, "SQL and PL/SQL for Oracle 10g", Black Book, Dreamtech Press.
4. Patrick Dalton, "Microsoft SQL Server Black Book", Coriolis Group, U.S.
5. Lynn Beighley, "Head First SQL", O'Reilly Media.

Prepared by Checked by Head of the Department Principal

Program: B.Tech in Computer Science and Engineering (Data Science)	S.Y. B.Tech	Semester: III
Course: Python Laboratory (DJS22DSL305)		

Pre-requisite:

1. Programming Fundamental

Objectives:

1. To learn the basic and OOP concepts of Python.
2. To study various advance python concept like inheritance, exception handling, modules etc.
3. Learn to develop GUI based standalone and web application.

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

1. Demonstrate basic data types, data structures and the concepts of Object-oriented programming in python.
2. Implement file handling and text processing concepts in python.
3. Develop an application using Tkinter, database connectivity and client-server communication using python.
4. Apply various advance modules of Python for data analysis.

Python Laboratory (DJS22DSL305)		
Unit	**Description**	**Duration**
1	**Python basics:** • Operators, Input and Output, Control statements. • Arrays, String and Character.	**02**
2	**Functions and Collections in python:** • Functions in python, Calling a Function, Arguments, Arbitrary Arguments, *args • Keyword Arguments, Arbitrary Keyword Arguments, **kwargs.. • The pass Statement, Recursion • Collections in Python, List, Tuples and Dictionaries	**04**
3	**Introduction to OOP:** • Classes, Objects, and Constructor. • Methods and Abstraction. • Inheritance. • Magic Methods.	**04**
4	**Exception Handling in Python in python:** • Exception Handling. • Try and Except Statement for Catching Exceptions. • Try with Else Clause. • Try, Except and Finally Statement for Catching Exception.	**02**

5	**Advanced Python Concepts:** • Modules, Packages. • Python Collections Module. • for Opening and Reading Files and Folders. • Python OS Module, Python Date Time Module, Python Math and Random Modules. • Text Processing & Regular expression.	**04**
6	**Python Integration Primer:** • GUI (Graphical User Interface) using Tkinter. • Client Server architecture using socket programming.	**03**
7	**Python database Connectivity:** • Database connectivity using SQL lite • CRUD (create, read, update and delete) operations on database (SQLite/ MySQL	**03**
8	**Python Numpy Module:** • Construct Numpy arrays, Printing arrays. • Arithmetic Operations on matrix's using Numpy Module. • numpy zeros ()	**03**
9	**Python Pandas Module:** • Data Processing using Pandas. • Data structure using Pandas. • Data Frame using Pandas and perform basic operations	**03**
10	**Python Matplotlib Module:** • Install Matplolib module. • Perform basic visualization.	**02**
	Total	**30**

*The Term Work will be calculated based on Laboratory Performance (15m) and Quizzes (10m).

Books Recommended:

Text books:

1. Zed Shaw, Learn Python the Hard Way, Addison-Wesley, 3rd Edition, 2013.
2. Laura Cassell, Alan Gauld, Python Projects, Wiley, 1st Edition, 2015.

Digital resources:

1. The Python Tutorial: http://docs.python.org/release/3.0.1/tutorial/
2. Free and Open Source IITB Lab: http://spoken-tutorial.org
3. www.staredusolutions.org

Prepared by Checked by Head of the Department Principal

Program: B.Tech in Computer Science and Engineering (Data Science)	S.Y. B.Tech	Semester: III
Course: Web Engineering Laboratory (DJS22DSL306)		

Pre-requisite:

1. Programming Fundamentals

Objectives:

To provide the basic framework of web development (MERN Stack) and cloud computing.

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

1. Develop a website as per the requirements.
2. Apply the concepts of cloud computing to improve the efficiency of web development.
3. Evaluate the requirement of the problem and select appropriate method of web development.

Web Engineering Laboratory (DJS22DSL306)		
Unit	**Description**	**Duration**
1	**HTML:** 1. Create a static web page using HTML. 2. Create a class timetable using HTML. 3. Create a registration form using HTML. 4. Create a web page using HTML5 tags.	**02**
2	**CSS:** 1. Design a web page using External or Embedded Style Sheet. 2. Design a responsive web page using media queries and CSS3. 3. Design a web page using Bootstrap. 4. Design a resume using Bootstrap. 5. Design the admission form using Bootstrap.	**04**
3	**Client-Side Scripting:** 1. Programs based on objects in JavaScript. 2. Program to design a calculator using JavaScript. 3. Programs based on form validation.	**04**
4	**React JS:** 1. Create an application using React. **Introduction to Git and GitHub:** 1. Introduction to Version Control 2. Using Git Locally and Remotely 3. Collaboration	**08**
5	**Server-Side Scripting:** 1. Installation and Configuration of Node.js server 2. Program based on inbuilt functions in Node.js	**04**
6	**Express & MongoDB:** 1. Using Mongoose to make schemas in MongoDB. 2. Making API end points using Express and testing using postman. 3. Develop a website and integrate it with pre-defined API.	**06**

	4. Develop a website to consume user defined API. 5. Doing CRUD on database MongoDB using Express. 6. Writing tests using mocha and chai.	
7	**XML & XSL:** 1. Design XML using XML DTD and schema. 2. Implementing XSL elements in XML. 3. Validating XML data through DTD and storing in database.	**04**
8	**Concepts of Cloud Computing:** 1. Introduction to cloud computing. 2. NIST model 3. Service and Deployment models.	**04**
9	**Networking and Security:** 1. Identity and Access Management 2. Networking basics 3. VPC networking and security 4. Design a VPC 5. Build your own VPC and Launch a Web Server	**04**
10	**Compute Service:** 1. Compute Services overview 2. Elastic Computing 3. Serverless Compute service 4. Deploying and scaling web applications	**04**
11	**Storage Service:** 1. Cloud object storage 2. Cloud block storage 3. Elastic file system	**04**
12	**Database Service:** 1. Cloud Relational database services 2. Cloud NoSQL Databases 3. Elastic load balancing	**04**
	Total	**52**

*The Term Work will be calculated based on Laboratory Performance (15m), Mini Project (10m) and Computer Based Assessment (25m).

Books Recommended:

Text books:

1. Vasan Subramanian, Pro MERN Stack, 2nd Edition, Apress Publication, 2019.
2. Shama Hoque, Full-Stack React Projects, 2nd Edition, Pac kt Publication, 2020
3. Rajkumar Buyya, James Broberg, Goscinki Cloud Computing: Principles and Paradigms, Wiley, 2013.
4. Shalabh Aggarwal, Flask Framework Cookbook: Over 80 proven recipes and techniques for Python web development with Flask, Packt publication, 2nd Edition, 2019

Reference Books:

1. Benjamin LaGrone, HTML5 and CSS3 Responsive Web Design Cookbook, 1st Edition, Packt

Publishing, 2013.
2. DT Editorial Services, Web Technologies: Black Book, 1st Edition, Dreamtech Press, 2018.
3. Christopher Schmitt, Kyle Simpson, HTML5 Cookbook, 1st Edition, O'Reilly Media Inc., 2011.
4. Uttam K. Roy, Web Technologies, 1st Edition, Oxford University Press, 2010.
5. Greg Sidelnikov, React. Js Book: Learning React JavaScript Library from Scratch, 1st Edition, Independently Published, 2017.
6. DT Editorial Services, HTML5 Black Book, 2nd Edition, Dreamtech Press, 2016.
7. Ben Frain, Responsive Web Design with HTML5 and CSS3, 2nd Edition, Packt Publishing, 2015.
8. Steve Suehring, JavaScript Step by Step, 3rd Edition, Pearson Education, 2013.
9. Stoyan Stefanov, React Up Running Building Web Applications, 1st Edition, O'Reilly Media Inc., 2016.
10. Velte, Cloud Computing a Practical Approach, Tata McGraw-Hill Education.
11. Sandip Bhowmik, Cloud Computing, Cambridge University Press, 2017.
12. Miguel Grinberg, Flask Web Development, O'Reilly publication,2018
13. Sack Stouffer Daniel Gaspar, Mastering Flask Web Development, Packt Publication, 2018

Program: B.Tech in Computer Science and Engineering (Data Science)	S.Y. B.Tech	Semester: III
Course: Innovative Product Development-I (DJS22ILLA1)		

Course Objectives:

1. To acquaint the students with the process of identifying the need (considering a societal requirement) and ensuring that a solution is found out to address the same by designing and developing an innovative product.
2. To familiarize the students with the process of designing and developing a product, while they work as part of a team.
3. To acquaint the students with the process of applying basic engineering fundamentals, so as to attempt at the design and development of a successful value-added product.
4. To inculcate the basic concepts of entrepreneurship and the process of self-learning and research required to conceptualize and create a successful product.

Course Outcome:

Learner will be able to:

1. Identify the requirement for a product based on societal/research needs.
2. Apply knowledge and skills required to solve a societal need by conceptualizing a product, especially while working in a team.
3. Use standard norms of engineering concepts/practices in the design and development of an innovative product.
4. Draw proper inferences through theoretical/ experimental/simulations and analyze the impact of the proposed method of design and development of the product.
5. Develop interpersonal skills, while working as a member of the team or as the leader.
6. Demonstrate capabilities of self-learning as part of the team, leading to life-long learning, which could eventually prepare them to be successful entrepreneurs.
7. Demonstrate product/project management principles during the design and development work and also excel in written (Technical paper preparation) as well as oral communication.

Guidelines for the proposed product design and development:

- Students shall form a team of 3 to 4 students (max allowed: 5-6 in extraordinary cases, subject to the approval of the department review committee and the Head of the department).
- Students should carry out a survey and identify the need, which shall be converted into conceptualization of a product, in consultation with the faculty supervisor/head of department/internal committee of faculty members.
- Students in the team shall understand the effective need for product development and accordingly select the best possible design in consultation with the faculty supervisor.
- Students shall convert the best design solution into a working model, using various components drawn from their domain as well as related interdisciplinary areas.
- Faculty supervisor may provide inputs to students during the entire span of the activity, spread over 2 semesters, wherein the main focus shall be on self-learning.

- A record in the form of an activity log-book is to be prepared by each team, wherein the team can record weekly progress of work. The guide/supervisor should verify the recorded notes/comments and approve the same on a weekly basis.
- The design solution is to be validated with proper justification and the report is to be compiled in a standard format and submitted to the department. Efforts are to be made by the students to try and publish a technical paper, either in the institute journal, "Techno Focus: Journal for Budding Engineers" or at a suitable publication, approved by the department research committee/ Head of the department.
- The focus should be on self-learning, capability to design and innovate new products as well as on developing the ability to address societal problems. Advancement of entrepreneurial capabilities and quality development of the students through the yearlong course should ensure that the design and development of a product of appropriate level and quality is carried out, spread over two semesters, ie during the semesters III and IV.

Guidelines for Assessment of the work:

- The review/ progress monitoring committee shall be constituted by the Head of the Department. The progress of design and development of the product is to be evaluated on a continuous basis, holding a minimum of two reviews in each semester.
- In the continuous assessment, focus shall also be on each individual student's contribution to the team activity, their understanding and involvement as well as responses to the questions being raised at all points in time.
- Distribution of marks individually for the both reviews as well as for the first review during the subsequent semester shall be as given below:
    - Marks awarded by the supervisor based on log-book 20
    - Marks awarded by review committee 20
    - Quality of the write-up 10
- In the last review of the semester IV, the marks will be awarded as follows
    - Marks awarded by the supervisor (Considering technical paper writing) 30
    - Marks awarded by the review committee 20

NOTE: A candidate needs to secure a minimum of 50 % marks to be declared to have completed the audit course.

Review/Progress monitoring committee may consider the following points during the assessment.

- In the semester III, the entire design proposal shall be ready, including components/system selection as well as the cost analysis. Two reviews will be conducted based on the presentation given by the student's team.
    - First shall be for finalization of the product selected.
    - Second shall be on finalization of the proposed design of the product.
- In the semester IV, the expected work shall be procurement of components/systems, building of the working prototype, testing and validation of the results based on work completed in semester III.
    - First review is based on readiness of building the working prototype.

- Second review shall be based on a presentation as well as the demonstration of the working model, during the last month of semester IV. This review will also look at the readiness of the proposed technical paper presentation of the team.

The overall work done by the team shall be assessed based on the following criteria;

1. Quality of survey/ need identification of the product.
2. Clarity of Problem definition (design and development) based on need.
3. Innovativeness in the proposed design.
4. Feasibility of the proposed design and selection of the best solution.
5. Cost effectiveness of the product.
1. Societal impact of the product.
6. Functioning of the working model as per stated requirements.
7. Effective use of standard engineering norms.
8. Contribution of each individual as a member or the team leader.
9. Clarity on the write-up and the technical paper prepared.

The semester reviews (III and IV) may be based on relevant points listed above, as applicable.

Guidelines for Assessment of Semester Reviews:

- The write-up should be prepared as per the guidelines given by the department.
- The design and the development of the product shall be assessed through a presentation and demonstration of the working model by the student team to a panel of Internal and External Examiners, preferably from industry or any research organizations having an experience of more than five years, approved by the Head of the Institution. The presence of the external examiner is desirable only for the 2nd presentation in semester IV. Students are compulsorily required to present the outline of the technical paper prepared by them during the sfinal review in semester IV.

Program: B.Tech in Computer Science and Engineering (Data Science)	S.Y. B.Tech	Semester: III
Course: Constitution of India (DJS22A2)		

Course Objectives:

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

Course Outcome: On successful completion of this course, student should be able to:

1. Have general knowledge and legal literacy and thereby to take up competitive examinations.
2. Understand state and central policies, fundamental duties.
3. Understand Electoral Process, special provisions.
4. Understand powers and functions of Municipalities, Panchayats and Co- Operative Societies.
5. Understand Engineering ethics and responsibilities of Engineers.
6. Understand Engineering Integrity & Reliability.

Constitution of India (DJ22A3)		
Unit	**Description**	**Duration**
1	**Introduction to the Constitution of India:** The Making of the Constitution and Salient features of the Constitution. Preamble to the Indian Constitution Fundamental Rights & its limitations.	**02**
2	**Directive Principles of State Policy:** Relevance of Directive Principles State Policy Fundamental Duties. Union Executives – President, Prime Minister Parliament Supreme Court of India.	**03**
3	**State Executives:** Governor, Chief Minister, State Legislature High Court of State. Electoral Process in India, Amendment Procedures, 42nd, 44th, 74th, 76th, 86th & 91st Amendments.	**03**
4	**Special Provisions:** For SC & ST Special Provision for Women, Children & Backward Classes Emergency Provisions. **Human Rights:** Meaning and Definitions, Legislation Specific Themes in Human Rights- Working of National	**03**
5	**Scope & Aims of Engineering Ethics:** Responsibility of Engineers Impediments to Responsibility. Risks, Safety and liability of Engineers, Honesty, Integrity & Reliability in Engineering	**03**
	Total	**14**

Books Recommended:

Text books:

1. Durga Das Basu, Introduction to the Constitution on India, (Students Edition.) Prentice –Hall EEE, 19th / 20th Edition., 2001.
2. Charles E. Haries, Michael S Pritchard and Michael J. Robins, Engineering Ethics, Thompson Asia, 2003.

Reference Books:

1. M.V. Pylee, An Introduction to Constitution of India, Vikas Publishing, 2002.
2. M. Govindarajan, S. Natarajan, V. S. Senthilkumar, Engineering Ethics, Prentice – Hall of India Pvt. Ltd. New, Delhi, 2004.
1. Brij Kishore Sharma, Introduction to the Constitution of India, PHI Learning Pvt. Ltd., New Delhi, 2011.
2. Latest Publications of Indian Institute of Human Rights, New Delhi.

Website Resources:

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