

EPARTMENT OF INFORMATION TECHNOLOGY

II B.TECH II SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
	3	0	0	30	70	100	3
SUBCODE: R23CC2208	OPTIMIZATION TECHNIQUES						

COURSE OBJECTIVES:

1. To define an objective function and constraint functions in terms of design variables, and then state the optimization problem.
2. To state single variable and multi variable optimization problems, without and with constraints.
3. To explain linear programming technique to an optimization problem, define slack and surplus variables, by using Simplex method.
4. To state transportation and assignment problem as a linear programming problem to determine Simplex method.
5. To study and explain nonlinear programming techniques, unconstrained or constrained, and define exterior and interior penalty functions for optimization problems.

COURSE OUTCOMES:

After successful completion of this course, the students will be able to:

- CO1.** **Formulate** the optimization problem, without and with constraints, by using design variables from an engineering design problem.
- CO2.** **Apply** classical optimization techniques to minimize or maximize a multi-variable objective function, without or with constraints, and arrive at an optimal solution.
- CO3.** Apply and Solve transportation and assignment problem by using Linear programming Simplex method.
- CO4.** **Apply** gradient and non-gradient methods to nonlinear optimization problems and use interior or exterior penalty functions for the constraints to derive the optimal solutions
- CO5.** **Formulate** Dynamic programming technique to inventory control, production planning, engineering design problems etc. to reach a final optimal solution from the current optimal solution.

UNIT I:**INTRODUCTION AND CLASSICAL OPTIMIZATION TECHNIQUES:**

Statement of an Optimization problem, design vector, design constraints, constraint surface, objective function, objective function surfaces, classification of Optimization problems.

CLASSICAL OPTIMIZATION TECHNIQUES: Single variable Optimization, multi variable Optimization without constraints, necessary and sufficient conditions for minimum/maximum, multivariable Optimization with equality constraints. Solution by method of Lagrange multipliers, multivariable Optimization with inequality constraints, Kuhn – Tucker conditions

UNIT II:

LINEAR PROGRAMMING: Standard form of a linear programming problem, geometry of linear programming problems, definitions and theorems, solution of a system of linear simultaneous equations, pivotal reduction of a general system of equations, motivation to the simplex method, simplex algorithm.

UNIT III:

TRANSPORTATION PROBLEM: Finding initial basic feasible solution by north – west corner rule, least cost method and Vogel's approximation method, testing for optimality of balanced transportation problems, Special cases in transportation problem.

UNIT IV:

NONLINEAR PROGRAMMING: Unconstrained cases, One – dimensional minimization methods: Classification, Fibonacci method, Univariate method, steepest descent method. Constrained cases– Characteristics of a constrained problem, Classification, Basic approach of Penalty Function method, Basic approaches of Interior and Exterior penalty function methods

UNIT V:

DYNAMIC PROGRAMMING: Dynamic programming multistage decision processes, types, concept of sub optimization and the principle of optimality, computational procedure in dynamic programming, examples illustrating the calculus method of solution, examples illustrating the tabular method of solution.

TEXTBOOKS:

1. "Engineering optimization: Theory and practice", Singiresu. S.Rao, New Age International (P) Limited, 3rd enlarged edition, 2013.
2. "Introductory Operations Research", H.S. Kasene& K.D. Kumar, Springer (India),Pvt.LTd.

REFERENCE BOOKS:

1. "Optimization Methods in Operations Research and systems Analysis", by K.V. Mital and C. Mohan, New Age International (P) Limited, Publishers, 3rd edition, 1996.
2. Operations Research, Dr.S.D.Sharma, Kedarnath, Ramnath& Co.

II B.TECH II SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
	3	0	0	30	70	100	3
SUBCODE: R23CC2202	PROBABILITY AND STATISTICS						

COURSE OBJECTIVES:

- To familiarize the students with the foundations of probability and statistical methods.
- To impart probability concepts and statistical methods in various applications engineering.

COURSE OUTCOMES:

After completion of this course, the students would be able to

CO1: Classify the concepts of data science and its importance [K2].

CO2: Interpret the association of characteristics and through correlation and regression tools [K4]

CO3: Apply discrete and continuous probability distributions [K3]

CO4: Design the components of a classical hypothesis test [K3]

CO5: Infer the statistical inferential methods based on small and large sampling tests [K4].

UNIT – I: Descriptive statistics and methods for data science:

Data science, Statistics Introduction, Population vs Sample, Collection of data, primary and secondary data, Type of variable: dependent and independent Categorical and Continuous variables, Data visualization, Measures of Central tendency, Measures of Variability , Skewness, Kurtosis.

UNIT – II: Correlation and Regression:

Correlation: Correlation coefficient, Rank correlation.

Linear Regression: Straight line, Multiple Linear Regression, Regression coefficients and properties.

Curvilinear Regression: Parabola, Exponential, Power curves.

UNIT – III: Probability and Distributions:

Probability, Conditional probability and Baye's theorem.

Random variables: Discrete and Continuous random variables.

Distribution functions: Probability mass function, Probability density function and Cumulative distribution functions, Mathematical Expectation and Variance, Binomial, Poisson, Uniform and Normal distributions.

UNIT – IV: Sampling Theory:

Introduction , Population and Samples, Sampling distribution of Means and Variance (definition only), Point and Interval estimations, Maximum error of estimate, Central limit theorem (without proof), Estimation using t- test.

UNIT – V: Tests of Hypothesis:

Introduction, Hypothesis, Null and Alternative Hypothesis, Type I and Type II errors, Level of significance, One tail and two-tail tests . Test of significance for large samples and

EPARTMENT OF INFORMATION TECHNOLOGY

Small Samples: Single and difference means, Single and two proportions, Student's t- test, F-test, χ^2 -test.

TEXT BOOKS:

1. Miller and Freund's, Probability and Statistics for Engineers,7/e, Pearson, 2008.
2. S. C. Gupta and V.K. Kapoor, Fundamentals of Mathematical Statistics, 11/e, Sultan Chand & Sons Publications, 2012.

REFERENCE BOOKS:

1. Shron L. Myers, Keying Ye, Ronald E Walpole, Probability and Statistics Engineers and the Scientists,8th Edition, Pearson 2007.
2. Jay l. Devore, Probability and Statistics for Engineering and the Sciences, 8th Edition, Cengage publishers.
3. Sheldon M. Ross, Introduction to probability and statistics Engineers and the Scientists, 4th Edition, Academic Foundation, 2011.
4. Johannes Ledolter and Robert V. Hogg, Applied statistics for Engineers and Physical Scientists, 3rd Edition, Pearson, 2010.

EPARTMENT OF INFORMATION TECHNOLOGY

II B.TECH II SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
	3	0	0	30	70	100	3
SUBCODE: R23CC2203	OPERATING SYSTEMS						

COURSE OBJECTIVES:

- Understand the basic concepts and principles of operating systems, including process management, memory management, file systems, and Protection
- Make use of process scheduling algorithms and synchronization techniques to achieve better performance of a computer system.
- Illustrate different conditions for deadlock and their possible solutions.

COURSE OUTCOMES:

After completion of this course, the students would be able to

CO 1: Classify various operating system generations, functions and services. [K2]

CO 2: Analyze process scheduling, management and synchronization. [K4]

CO 3: Analyze deadlock prevention, detection, avoidance and recovery techniques [K4]

CO 4: Analyze various memory management and storage management techniques [K4].

CO 5: Analyze the concepts of file system [K2]

UNIT - I

Operating Systems Overview: Introduction, Operating system functions, Operating systems operations, Computing environments, Free and Open-Source Operating Systems

System Structures: Operating System Services, User and Operating-System Interface, system calls, Types of System Calls, system programs, Operating system Design and Implementation, Operating system structure, Building and Booting an Operating System, Operating system debugging

UNIT - II

Processes: Process Concept, Process scheduling, Operations on processes, Inter-process communication.

Threads and Concurrency: Multithreading models, Thread libraries, Threading issues.

CPU Scheduling: Basic concepts, Scheduling criteria, Scheduling algorithms, Multiple processor scheduling.

UNIT – III

Synchronization Tools: The Critical Section Problem, Peterson's Solution, Mutex Locks, Semaphores, Monitors, Classic problems of Synchronization.

Deadlocks: system Model, Deadlock characterization, Methods for handling Deadlocks, Deadlock prevention, Deadlock avoidance, Deadlock detection, Recovery from Deadlock.

UNIT - IV

Memory-Management Strategies: Introduction, Contiguous memory allocation, Paging, Structure of the Page Table, Swapping.

Virtual Memory Management: Introduction, Demand paging, Copy-on-write, Page replacement, Allocation of frames, Thrashing

UNIT - V

Storage Management: Overview of Mass Storage Structure, HDD Scheduling.

File System: File System Interface: File concept, Access methods, Directory Structure; File system Implementation: File-system structure, File-system Operations, Directory implementation, Allocation method, Free space management; File-System Internals: File-System Mounting, Partitions and Mounting, File Sharing.

Text Books:

1. Operating System Concepts, Silberschatz A, Galvin P B, Gagne G, 10th Edition, Wiley, 2018.
2. Modern Operating Systems, Tanenbaum A S, 4th Edition, Pearson , 2016

Reference Books:

1. Operating Systems -Internals and Design Principles, Stallings W, 9th edition, Pearson, 2018
2. Operating Systems: A Concept Based Approach, D.M Dhamdhere, 3rd Edition, McGraw- Hill, 2013

Online Learning Resources:

1. <https://nptel.ac.in/courses/106/106/106106144/>
2. <http://peterindia.net/OperatingSystems.html>

II B.TECH II SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
	3	0	0	30	70	100	3
SUBCODE: R23CC2204	DATABASE MANAGEMENT SYSTEMS						

COURSE OBJECTIVES:

- Introduce database management systems and to give a good formal foundation on the relational model of data and usage of Relational Algebra
- Introduce the concepts of basic SQL as a universal Database language
- Demonstrate the principles behind systematic database design approaches by covering conceptual design, logical design through normalization
- Provide an overview of physical design of a database system, by discussing Database indexing techniques and storage techniques

COURSE OUTCOMES:

After completion of this course, the students would be able to

CO 1: Interpret the fundamentals of DBMS. [K2]

CO 2: Analyze relational database designing. [K4]

CO 3: Develop queries in RDBMS [K3]

CO 4: Analyze database design methodology and normalization process [K4].

CO 5: Analyze transaction concepts and File indexing. [K2]

UNIT I:

Introduction: Database system, Characteristics (Database Vs File System), Database Users, Advantages of Database systems, Database applications. Brief introduction of different Data Models; Concepts of Schema, Instance and data independence; Three tier schema architecture for data independence; Database system structure, environment, Centralized and Client Server architecture for the database.

Entity Relationship Model: Introduction, Representation of entities, attributes, entity set, relationship, relationship set, constraints, sub classes, super class, inheritance, specialization, generalization using ER Diagrams.

UNIT II:

Relational Model: Introduction to relational model, concepts of domain, attribute, tuple, relation, importance of null values, constraints (Domain, Key constraints, integrity constraints) and their importance, Relational Algebra, Relational Calculus. BASIC SQL:Simple Database schema, data types, table definitions (create, alter), different DML operations (insert, delete, update).

UNIT III:

SQL:Basic SQL querying (select and project) using where clause, arithmetic & logical operations, SQL functions(Date and Time, Numeric, String conversion).Creating tables with relationship, implementation of key and integrity constraints, nested queries, sub queries, grouping, aggregation, ordering, implementation of different types of joins, view(updatable and non-updatable), relational set operations.

UNIT IV:

Schema Refinement (Normalization):Purpose of Normalization or schema refinement, concept of functional dependency, normal forms based on functional dependencyLossless join and dependency preserving decomposition, (1NF, 2NF and 3 NF), concept of surrogate key, Boyce-Coddnormal form(BCNF), MVD, Fourth normal form(4NF), Fifth Normal Form (5NF).

UNIT V:

Transaction Concept: Transaction State, ACID properties, Concurrent Executions, Serializability, Recoverability, Implementation of Isolation, Testing for Serializability, lock based, time stamp based, optimistic, concurrency protocols, Deadlocks, Failure Classification, Storage, Recovery and Atomicity, Recovery algorithm.

Introduction to Indexing Techniques: B+ Trees, operations on B+Trees, Hash Based Indexing:

TEXT BOOKS:

- 1) Database Management Systems, 3rd edition, Raghurama Krishnan, Johannes Gehrke, TMH (For Chapters 2, 3, 4)
- 2) Database System Concepts, 5th edition, Silberschatz, Korth, Sudarsan, TMH (For Chapter 1 and Chapter 5)

REFERENCE BOOKS:

- 1) Introduction to Database Systems, 8th edition, C J Date, Pearson.
- 2) Database Management System, 6th edition, Ramez Elmasri, Shamkant B. Navathe, Pearson
- 3) Database Principles Fundamentals of Design Implementation and Management, Corlos Coronel, Steven Morris, Peter Robb, Cengage Learning.

WEB-RESOURCES:

- 1) <https://nptel.ac.in/courses/106/105/106105175/>
- 2) https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01275806667282022456_shared/overview

II B.TECH II SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
	3	0	0	30	70	100	3
SUBCODE: R23CC2205	SOFTWARE ENGINEERING						

COURSE OBJECTIVES:

The objectives of this course are to introduce

- Software life cycle models, Software requirements and SRS document.
- Project Planning, quality control and ensuring good quality software.
- Software Testing strategies, use of CASE tools, Implementation issues, validation & verification procedures.

COURSE OUTCOMES:

After completion of this course, the students would be able to

CO 1: Analyze Software Life Cycle models. [K4]

CO 2: Analyze the importance of software requirement and project management [K4]

CO 3: Analyze various types of software design techniques [K4]

CO 4: Analyze Software testing and quality management [K4].

CO 5: Analyze various CASE tools and software maintenance process models. [K4]

UNIT I:

Introduction: Evolution, Software development projects, Exploratory style of software developments, Emergence of software engineering, Notable changes in software development practices, Computer system engineering.

Software Life Cycle Models: Basic concepts, Waterfall model and its extensions, Rapid application development, Agile development model, Spiral model.

UNIT II:

Software Project Management: Software project management complexities, Responsibilities of a software project manager, Metrics for project size estimation, Project estimation techniques, Empirical Estimation techniques, COCOMO, Halstead's software science, risk management.

Requirements Analysis And Specification: Requirements gathering and analysis, Software Requirements Specification (SRS), Formal system specification, Axiomatic specification, Algebraic specification, Executable specification and 4GL.

UNIT III:

Software Design: Overview of the design process, How to characterize a good software design? Layered arrangement of modules, Cohesion and Coupling.approaches to software design.

Agility: Agility and the Cost of Change, Agile Process, Extreme Programming (XP)

Function-Oriented Software Design: Overview of SA/SD methodology, Structured analysis, Developing the DFD model of a system, Structured design, Detailed design, and Design Review.

User Interface Design: Characteristics of a good user interface, Basic concepts, Types of user interfaces, Fundamentals of component-based GUI development, and user interface design methodology.

UNIT IV:

Coding And Testing: Coding, Code review, Software documentation, Testing, Black-box testing, White-Box testing, Debugging, Program analysis tools, Integration testing, Testing object-oriented programs, Smoke testing, and Some general issues associated with testing.

Software Reliability And Quality Management: Software reliability. Software quality, Software quality management system, ISO 9000. SEI Capability maturity model.

UNIT V:

Computer-Aided Software Engineering (Case): CASE and its scope, CASE environment, CASE support in the software life cycle.

Software Maintenance: Characteristics of software maintenance, Software reverse engineering, Software maintenance process models and Estimation of maintenance cost.

TEXT BOOKS:

1. Fundamentals of Software Engineering, Rajib Mall, 5th Edition, PHI.
2. Software Engineering A practitioner's Approach, Roger S. Pressman, 9th Edition, Mc-Graw Hill International Edition.

REFERENCE BOOKS:

1. Software Engineering, Ian Sommerville, 10th Edition, Pearson.
2. Software Engineering, Principles and Practices, Deepak Jain, Oxford University Press.

WEB RESOURCES:

- 1) <https://nptel.ac.in/courses/106/105/106105182/>
- 2) https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01260589506387148827_shared/overview
- 3) https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013382690411003904735_shared/overview

EPARTMENT OF INFORMATION TECHNOLOGY

II B.TECH II SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
	0	0	3	30	70	100	1.5
SUBCODE: R23IT22L4	OPERATING SYSTEMS & SOFTWARE ENGINEERING LAB						

COURSE OBJECTIVES

The main objectives of the course are to

- Provide insights into system calls, file systems, semaphores,
- Develop and debug CPU Scheduling algorithms, page replacement algorithms, thread implementation
- Implement Bankers Algorithms to Avoid the Dead Lock

Experiments covering the Topics:

- UNIX fundamentals, commands & system calls
- CPU Scheduling algorithms, thread processing
- IPC, semaphores, monitors, deadlocks
- Page replacement algorithms, file allocation strategies
- Memory allocation strategies

COURSE OUTCOMES

After completion of the course, the students would be able to:

CO 1: Experiment with various Unix Commands and system calls [K3]

CO 2: Experiment with various operating system concepts such as scheduling algorithms, page replacement algorithms. IPC mechanism, memory allocation, file allocation and handling deadlocks.[K3]

CO 3: Design various S/W applications using E-R diagrams, DFD, CFD, Structured charts UML diagrams etc and estimate the effort using COCOMO model and FP oriented estimation model.[K5]

CO 4: Design the test cases for e-Commerce and Mobile applications[K5]

Sample Experiments in Operating System:

1. Practicing of Basic UNIX Commands.
2. Write programs using the following UNIX operating system calls fork, exec, getpid, exit, wait, close, stat, opendir and readdir
3. Simulate UNIX commands like cp, ls, grep, etc.,
4. Simulate the following CPU scheduling algorithms
 - a) FCFS b) SJF c) Priority d) Round Robin
5. Control the number of ports opened by the operating system with
 - a) Semaphore b) Monitors.
6. Write a program to illustrate concurrent execution of threads using pthreads library.
7. Write a program to solve producer-consumer problem using Semaphores.
8. Implement the following memory allocation methods for fixed partition

EPARTMENT OF INFORMATION TECHNOLOGY

- a) First fit b) Worst fit c) Best fit
- 9. Simulate the following page replacement algorithms
 - a) FIFO b) LRU c) LFU
- 10. Simulate Paging Technique of memory management.
- 11. Implement Bankers Algorithm for Dead Lock avoidance and prevention
- 12. Simulate the following file allocation strategies
 - a) Sequential b) Indexed c) Linked
- 13. Download and install nachos operating system and experiment with it.

Sample Experiments in Software Engineering:

- 1) Perform the following, for the following experiments:
 - i. Do the Requirement Analysis and Prepare SRS
 - ii. Draw E-R diagrams, DFD, CFD and structured charts for the project.
 - a. Course Registration System
 - b. Students Marks Analyzing System
 - c. Online Ticket Reservation System
 - d. Stock Maintenance
- 2) Consider any application, using COCOMO model, estimate the effort.
- 3) Consider any application, Calculate effort using FP oriented estimation model.
- 4) Draw the UML Diagrams for the problem a, b, c, d.
- 5) Design the test cases for e-Commerce application (Flipcart, Amazon)
- 6) Design the test cases for a Mobile Application (Consider any example from Appstore)
- 7) Design and Implement ATM system through UML Diagrams.

REFERENCE BOOKS:

- 1. Operating System Concepts, Silberschatz A, Galvin P B, Gagne G, 10th Edition, Wiley, 2018.
- 2. Modern Operating Systems, Tanenbaum A S, 4th Edition, Pearson, 2016
- 3. Operating Systems -Internals and Design Principles, Stallings W, 9th edition, Pearson, 2018
- 4. Operating Systems: A Concept Based Approach, D.M Dhamdhere, 3rd Edition, McGraw- Hill, 2013

ONLINE LEARNING RESOURCES:

- 1. <https://www.cse.iitb.ac.in/~mythili/os/>
 - 2. <http://peterindia.net/OperatingSystems.html>
 - 3. www.cs.washington.edu/~tom/nachos
-

DEPARTMENT OF INFORMATION TECHNOLOGY

II B.TECH II SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
	0	0	3				1.5
SUBCODE: R23CC22L1	DATABASE MANAGEMENT SYSTEMS LAB						

COURSE OBJECTIVES:

This Course will enable students to

- Populate and query a database using SQL DDL/DML Commands
- Declare and enforce integrity constraints on a database
- Writing Queries using advanced concepts of SQL
- Programming PL/SQL including procedures, functions, cursors and triggers

Experiments covering the topics:

- DDL, DML, DCL commands
- Queries, nested queries, built-in functions,
- PL/SQL programming- control structures
- Procedures, Functions, Cursors, Triggers,
- Database connectivity- ODBC/JDBC

COURSE OUTCOMES:

After Completion of this course student must be able to

CO1: Apply SQL commands like DDL, DML, DCL and Indexing to perform different Database operations [K3].

CO2: Develop PL/SQL block statements, control statements and cursors. [K3]

CO3: Develop PL/SQL programs using functions and procedures. [K3]

CO4: Develop PL/SQL programs using packages and Triggers. [K3]

CO5: Develop a Java Program to connect to a database. [K3].

Sample Experiments:

1. Creation, altering and dropping of tables and inserting rows into a table (use constraints while creating tables) examples using SELECT command.
2. Queries (along with sub Queries) using ANY, ALL, IN, EXISTS, NOTEXISTS, UNION, INTERSET, Constraints. Example:- Select the roll number and name of the student who secured fourth rank in the class.
3. Queries using Aggregate functions (COUNT, SUM, AVG, MAX and MIN), GROUP BY, HAVING and Creation and dropping of Views.
4. Queries using Conversion functions (to_char, to_number and to_date), string functions (Concatenation, lpad, rpad, ltrim, rtrim, lower, upper, initcap, length, substr and instr), date functions (Sysdate, next_day, add_months, last_day, months_between, least, greatest, trunc, round, to_char, to_date)

EPARTMENT OF INFORMATION TECHNOLOGY

5.
 - i. Create a simple PL/SQL program which includes declaration section, executable section and exception –Handling section (Ex. Student marks can be selected from the table and printed for those who secured first class and an exception can be raised if no records were found)
 - ii. Insert data into student table and use COMMIT, ROLLBACK and SAVEPOINT in PL/SQL block.
6. Develop a program that includes the features NESTED IF, CASE and CASE expression. The program can be extended using the NULLIF and COALESCE functions.
7. Program development using WHILE LOOPS, numeric FOR LOOPS, nested loops using ERROR Handling, BUILT –IN Exceptions, USE defined Exceptions, RAISE-APPLICATION ERROR.
8. Programs development using creation of procedures, passing parameters IN and OUT of PROCEDURES.
9. Program development using creation of stored functions, invoke functions in SQL Statements and write complex functions.
10. Develop programs using features parameters in a CURSOR, FOR UPDATE CURSOR, WHERE CURRENT of clause and CURSOR variables.
11. Develop Programs using BEFORE and AFTER Triggers, Row and Statement Triggers and INSTEAD OF Triggers
12. Create a table and perform the search operation on table using indexing and non-indexing techniques.
13. Write a Java program that connects to a database using JDBC
14. Write a Java program to connect to a database using JDBC and insert values into it
15. Write a Java program to connect to a database using JDBC and delete values from it

TEXT BOOKS/SUGGESTED READING:

1. Oracle: The Complete Reference by Oracle Press
2. Nilesh Shah, "Database Systems Using Oracle", PHI, 2007
3. Rick F Vander Lans, "Introduction to SQL", Fourth Edition, Pearson Education, 2007

DEPARTMENT OF INFORMATION TECHNOLOGY

II B.TECH II SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
	0	1	2				2
R23IT22L2	PYTHON WITH DJANGO (SKILL ENHANCEMENT COURSE)						

COURSE OBJECTIVES:

The main objectives of the course are to

- Design and build static as well as dynamic web pages and interactive web-based applications
- Web development using Django framework.
- Analyze and create functional website in Django and deploy Django Web Application on Cloud

COURSE OUTCOMES:

After completion of the course, students are able to:

CO1: Develop Web Applications using Python Libraries. [K3]

CO2: Create applications using MVC architecture, interacting bootstrap, tables, grids and carousels of Django.[K5]

CO3: Create applications using interacting accounts and authentication on Django.[K5]

CO4: Create Apps by SQLitewith Django.[K5]

CO5: Develop websiteand deploy Djangoweb applications on cloud.[K3]

UNIT-I : Python libraries for web development :

Collections-Container datatypes, Tkinter-GUI applications, Requests-HTTP requests, BeautifulSoup4-web scraping, Scrapy, Zappa, Dash, CherryPy, Turbo Gears, Flask, Web2Py, Bottle, Falcon, Cubic Web, Quixote, Pyramid.

Sample Experiments:

1. Write a Python GUI program to import Tkinter package and create a window. Set its title and add a label to the window.
2. Write a Python program that designs a simple login form with labels and Entry widgets, arranging them in a grid using the Grid geometry manager.
3. Write a program using BeautifulSoup4 library for web scraping for a given URL
4. Develop a sample Hello World page using Flask framework
5. Develop a sample web page using CherryPy / Web2Py / Bottle Framework

UNIT-II: Introduction to Django Framework

Understanding Django environment, Features of Django and Django architecture, MVC and MTV, URLs and Views, Mapping the views to URLs, Django Template, Template inheritance Django Models, Creating model for site, Converting the model into a table, Fields in Models, Integrating Bootstrap into Django, Creating tables, Creating grids, Creating carousels.

Sample Experiments:

6. Create a Sample “Hello World” Application using Django
7. Create a Login and Registration Page using MVC architecture in Django Framework
8. Create a sample page in Django by integrating BootStrap.
9. Create an application with Tables, grids in Django
10. Create a Django App with Carousels feature.

UNIT-III : Integrating Accounts & Authentication on Django

Introduction to Django Authentication System, Security Problem & Solution with Django
Creating Registration Form using Django, Adding Email Field in Forms, Configuring email settings, Sending emails with Django, Adding Grid Layout On Registration Page, Adding Page Restrictions, Login Functionality Test and Logout.

Sample Experiments:

11. Create a registration page using Authentication System
12. Create an application in Django to send emails using email settings and Grid Layout
13. Create an application in Django using page restriction / authentication with Login and Logout Functionality
14. Create a sample form using Django Forms

UNIT-IV: Connecting SQLite with Django

Database Migrations, Fetch Data From Database, Displaying Data On Templates, Adding Condition On Data, Sending data from url to view, Sending data from view to template, Saving objects into database, Sorting objects, Filtering objects, Deleting objects, Difference between session and cookie, Creating sessions and cookies in Django.

Sample Experiments:

15. Create an app in Django which fetches data from database and show as list and also save objects in database
16. Create an app in Django for performing CRUD operations on records in a database
17. Create an app in Django which uses session management and cookies to store and manage user sessions.

UNIT-V: Deploying Django Web Application on Cloud

Creating a functional website in Django, Four Important Pillars to Deploy, registering on Heroku and GitHub, Push project from Local System to GitHub, working with Django Heroku, Working with Static Root, Handling WSGI with gunicorn, setting up Database & adding users.

Sample Experiments:

18. Create a website in Django with login, and registration page.
19. Register on GitHub, and Heroku and deploy the website on Heroku with all the functionalities developed.

20. Configure Django to handle static files.

Text books:

1. Martin C.Brown, "Python: The Complete Reference Paper back", 4th Edition 2018, McGraw Hill Education.
2. Reema Thareja, "Python Programming: Using Problem Solving Approach", 3rd Edition 2017,Oxford.
3. Daniel Rubio, Apress, "Beginning Django Web Application Development and Deployment with Python", 2nd Edition 2017, Apress.

Reference Books:

1. Tom Aratyn, "Building Django 2.0 Web Applications: Create enterprise-grade, scalable Python web applications easily with Django 2.0", 2nd Edition 2018, Packt Publishing.
2. Harry Percival, "Test-Driven Development with Python: Obey the Testing Goat: Using Django, Selenium and JavaScript",2nd Edition 2019, Kindle Edition.

DEPARTMENT OF INFORMATION TECHNOLOGY

II B.TECH II SEMESTER	L	T	P	INTERNAL MARKS	EXTERNAL MARKS	TOTAL MARKS	CREDITS
	1	0	2	30	70	100	2
SUBCODE: R23CC22L3	DESIGN THINKING & INNOVATION						

COURSE OBJECTIVES:

- Bring awareness on innovative design and new product development.
- Explain the basics of design thinking.
- Familiarize the role of reverse engineering in product development.
- Train how to identify the needs of society and convert into demand.
- Introduce product planning and product development process

COURSE OUTCOMES

After completion of the course, students are able to:

CO1: Define the concepts related to design thinking. [K1].

CO2: Infer the fundamentals of Design Thinking and innovation. [K2].

CO3: Apply the design thinking techniques for solving problems various sectors. [K3].

CO4: Analyze to work in a multidisciplinary environment.[K4].

CO5: Evaluate the value of creativity. [K4]

UNIT – I Introduction to Design Thinking

Introduction to elements and principles of Design, basics of design-dot, line, shape, form as fundamental design components. Principles of design. Introduction to design thinking, history of Design Thinking, New materials in Industry.

UNIT - II Design Thinking Process

Design thinking process (empathize, analyze, idea & prototype), implementing the process in driving inventions, design thinking in social innovations. Tools of design thinking - person, costumer, journey map, brainstorming, product development

Activity: Every student presents their idea in three minutes, Every student can present design process in the form of flow diagram or flow chart etc. Every student should explain about product development.

UNIT - III Innovation

Art of innovation, Difference between innovation and creativity, role of creativity and innovation in organizations. Creativity to Innovation. Teams for innovation, Measuring the impact and value of creativity.

Activity: Debate on innovation and creativity, Flow and planning from idea to innovation, Debate on value-based innovation.

UNIT - IV Product Design

Problem formation, introduction to product design, Product strategies, Product value, Product planning, product specifications. Innovation towards product design Case studies.

Activity: Importance of modeling, how to set specifications, Explaining their own product design.

DEPARTMENT OF INFORMATION TECHNOLOGY

UNIT – V Design Thinking in Business Processes

Design Thinking applied in Business & Strategic Innovation, Design Thinking principles that redefine business – Business challenges: Growth, Predictability, Change, Maintaining Relevance, Extreme competition, Standardization. Design thinking to meet corporate needs. Design thinking for Startups. Defining and testing Business Models and Business Cases. Developing & testing prototypes.

Activity: How to market our own product, about maintenance, Reliability and plan for startup.

TEXTBOOKS:

1. Tim Brown, Change by design, 1/e, Harper Bollins, 2009.
2. Idris Mootee, Design Thinking for Strategic Innovation, 1/e, Adams Media, 2014.

REFERENCE BOOKS:

1. David Lee, Design Thinking in the Classroom, Ulysses press, 2018.
2. Shrrutin N Shetty, Design the Future, 1/e, Norton Press, 2018.
3. William lidwell, Kritinaholden, &Jill butter, Universal principles of design, 2/e, Rockport Publishers, 2010.
4. Chesbrough.H, The era of open innovation, 2003.

ONLINE LEARNING RESOURCES:

1. <https://nptel.ac.in/courses/110/106/110106124/>
2. <https://nptel.ac.in/courses/109/104/109104109/>
3. https://swayam.gov.in/nd1_noc19_mg60/preview
4. [https://onlinecourses.nptel.ac.in/noc22_de16/preview.](https://onlinecourses.nptel.ac.in/noc22_de16/preview)