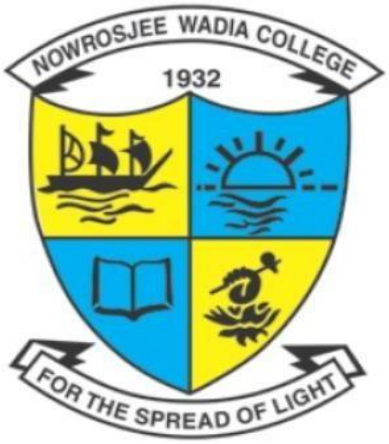

Modern Education Society's

Nowrosjee Wadia College(Autonomous)

An Autonomous college affiliated to Savitribai Phule Pune University

Three Year B.Sc. Degree Program in Computer Science
(Faculty of Science & Technology)

T.Y.B.Sc. (Computer Science)
(Syllabus under Autonomy)

Choice Based Credit System

To be implemented from Academic Year 2024-2025

Preamble:

Nowrosjee Wadia College has undergone autonomy from the academic year 2022- 2023. Taking into consideration the rapid changes in computer science and technology and new approaches in different areas of computer science and related subjects the syllabus of T.Y.B.Sc.(Computer Science) is prepared. To develop the syllabus the U.G.C. Model curriculum is followed.

i) Give the students a sufficient knowledge of fundamental principles, methods and a clear perception of in numerous power of computational ideas and tools and know how to use them by modeling, analyzing and programming.

ii) Reflecting the broad nature of the subject and developing software for continuing further study in various fields of science.

iii)Enhancing students' overall development and to equip them with software modeling abilities, programming, problem analyzing and solving skills, creative talent and power of communication necessary for various kinds of employment.

iv) Enabling students to develop a positive attitude towards computer science as aninteresting and valuable subject of study.

Objectives:

i) A student should be able to recall basic facts about computer science and should be able to display knowledge such as syntax, programming languages, algorithms, terminology and build logic, develop software building skills resulting from their studies.

ii) A student should get a computational understanding of computer science concepts and concerned programming ideas, and should be able to follow the ways of software development.

iii) A student should get adequate exposure to global and local concerns that explore them many aspects of Computer Sciences.

iv) A student should be able to apply their skills and knowledge, that is, translate information presented verbally into programming form, select and use appropriatealgorithms in order to process the information and develop the relevant software.

v) A student should be made aware of importance of computer science and hence of its past, present, and future role as part of our culture.

Eligibility for the Course

XIIth Science or its equivalent examination with Mathematics. All other criteria are same as per Savitribai Phule Pune University rules.

EVALUATION PATTERN :

Note : The Department follows all rules, regulations and procedure related to the examination decided by Examination Section of college.

(i) Each course shall be evaluated with Continuous Evaluation (CE) and Semester-end Examination (SEE) mechanism.

(ii) Theory courses: Continuous Evaluation shall be of 15 marks and Final Assessment shall be of 35 marks.

(iii) Practical courses: Continuous Evaluation shall be of 15 marks and Final Assessment shall be of 35 marks.

(iv) To pass a course of 2 credits, a student has to earn minimum 20 marks out of 50, provided that he/she should earn minimum 6 marks in Continuous Evaluation (out of 15) and minimum 14 marks (out of 35) in End-Semester Examination. That is passing criterion is minimum 40% marks in the examination.

(v) For Continuous Evaluation (out of 15 marks), There has to be one written test of 10 marks (Mid-Semester Examination). The remaining 05 marks shall be based on the continuous evaluation consisting of tutorial, viva, seminars, home-assignments, mini project, survey, group discussion etc.(on approval of Head of the Department) and performance and attendance in the lectures and labs.

(vi) There shall be revaluation of the answer scripts of End-Semester Examination (out of 35 marks) of theory papers only, but not of internal assessment papers as per Ordinance No. 134 A and B.

PATTERN OF THE QUESTION PAPER

Continuous Evaluation (Theory papers)

(1) As a part of Internal Evaluation, there shall be written test (Mid-Semester Examination) of 10 marks. Pattern of the question paper is as follows.

(2) Continuous Evaluation (Mid-Semester Examination of 10 marks, Duration: 45 minutes)

Question No.	Total Marks	No. Of questions	Remarks
Q. 1.	05	Solve any 5 out of 7	Short answer / objective type of questions. Each question carry 1 mark.
Q. 2.	05	Solve any 1 out of 2 questions	Each question carry 5 marks

(3) Semester-End Examination for B. Sc. courses, out of 35 marks, shall be of 2 hours duration. The pattern of the question paper will be as decided by the examination section of the college.

AWARD OF GRADES AND GRADE POINTS

The mapping of percentage to letter grade and grade point (for each course) is given in the following Table.

Sr. No.	Grade Letter	Grade Point	Marks
1.	O (Outstanding)	10	45 ≤ Marks ≤ 50
2.	A+ (Excellent	9	40 ≤ Marks ≤ 44
3.	A (Very Good)	8	35 ≤ Marks ≤ 39
4.	B+ (Good)	7	27.5 ≤ Marks ≤ 34
5.	B (Above Average)	6	25 ≤ Marks < 27.5
6.	C (Average)	5	22.5 ≤ Marks ≤ 24
7.	D (Pass)	4	20 ≤ Marks < 22.5
8.	F (Fail)	0	Marks < 20
9.	Ab (Absent)	0	

CGPA :The CGPA is the weighted average of the grade points obtained in all courses (theory and Practicals) by a student in all the courses in 6 semesters.

Based on the performance of the student in the Semester Examinations, Nowrosjee Wadia College will declare the results and issue the Semester Grade sheets. Also, the College will declare the results and issue the Grade sheets at the end of the course. The class will be awarded to a student on the basis of CGPA. The award of the class shall be as per the following table

Sr. No.	CGPA	Class of the degree awarded
1	9.50 or more than 9.50	OUTSTANDING (O)
2	8.50 or more but less than 9.50	EXCELLENT (A+)
3	7.50 or more but less than 8.50	VERY GOOD (A)
4	6.25 or more but less than 7.50	GOOD (B+)
5	5.25 or more but less than 6.25	ABOVE AVERAGE (B)
6	4.75 or more but less than 5.25	AVERAGE (C)
7	4.00 or more but less than 4.75	PASS (D)

Semester V (Total credits=22)

Course type	Paper Code	Paper title	Credits		Evaluation		
			T	P	CE	SEE	TOTAL
DSEC - I	CS-351	Operating Systems II	2		15	35	50
	CS-352	Computer Networks II	2		15	35	50
	CS-357	Practical course on CS-351		2	15	35	50
DSEC - II	CS-353	Web Technologies I	2		15	35	50
	CS-354	Foundations of Data Science	2		15	35	50
	CS-358	Practical course on CS-353 and CS-354		2	15	35	50
DSEC - III	CS-355	Object Oriented Programming using Java	2		15	35	50
	CS-356	Theoretical Computer Science	2		15	35	50
	CS-359	Practical Course on CS 355 and Project Design		2	15	35	50
SECC - I	CS-3510	Elective I	2	0	15	35	50
SECC - II	CS-3511	Elective II	2	0	15	35	50

Elective Group I:

1. Python Programming
2. Git and Github Programming
3. UI / UX design

Semester VI (Total credits=22)

Course type	Paper Code	Paper title	Credits		Evaluation		
			T	P	CE	SEE	TOTAL
DSEC - I	CS-361	Fundamentals of Artificial Intelligence	2		15	35	50
	CS-362	Software Testing	2		15	35	50
	CS-367	Practical course on CS-361 and CS-362		2	15	35	50
DSEC - II	CS-363	Web Technologies II	2		15	35	50
	CS-364	Data Analytics	2		15	35	50
	CS-368	Practical course on CS-363 and CS-364		2	15	35	50
DSEC - III	CS-365	Advanced Java Programming	2		15	35	50
	CS-366	Compiler Construction	2		15	35	50
	CS-369	Practical Course on CS 365 and Project implementation		2	15	35	50
SECC - I	CS-3610	Elective III	2	0	15	35	50
SECC - II	CS-3611	Elective IV	2	0	15	35	50

Elective Group II:

1. Mobile application development using Android.
2. Block chain technologies.
3. Cloud and Edge Computing.

Teaching Scheme :

- Each theory lecture time for T.Y. B. Sc Computer Science is of 50 min (3 lectures/ week for 2 credit course)
- Each practical session time for T.Y. B. Sc Computer Science is of 4 hrs 20 minutes (260 min)
- Practical batch size =12

Subject Code: CS-351
Subject Title: Operating System-II

Teaching Scheme	No. of Credits	Examination Scheme
36 Lectures	2	CE: 15 marks SEE: 35 marks

Course Objectives:

1. To understand the issue of Deadlocks in Process management.
2. To understand the concept of Files system management & disk scheduling
3. To study the concept of distributed and mobile operating systems

Course Contents

Unit 1 **Process Deadlock** **(8L)**

1.1 Introduction to Process Deadlock, Definition, Concepts
1.2 Deadlock Characterization–Necessary conditions, Resource allocation graph
1.3 Deadlock Handling Methods- Deadlock Prevention and Deadlock Avoidance-Safe state, Resource Allocation Graph Algorithm, Banker's Algorithm-Safety Algorithm, Resource Request Algorithm
1.4 Deadlock Detection-Single Instance Resource Type, Several Instances of Resource Type
1.5 Recovery from Deadlock–Process termination, Resource preemption

Unit 2 **File System Management** **(8L)**

2.1 Introduction to File System-Need for File System, File System Structure
2.2 Concepts of File System- File attributes, File Types and Operations
2.3 Access Methods–Sequential, Direct, Other access methods
2.4 Directory overview, Single level directory, two level directory, Tree structure directory, acyclic graph directory, General graph directory
2.5 Allocation Methods–Contiguous allocation, Linked allocation, Indexed allocation
2.6 Free Space Management–Bit vector, linked list, Grouping, Counting, Space map

Unit 3 **Disk Scheduling** **(6L)**

3.1 Introduction and Overview, Disk Structure
3.2 Concept of Disk Scheduling, Disk performance parameters, Disk Scheduling Algorithms-FCFS, SSTF, Scan, C-Scan, Look and C-Look
3.3 Disk Management

Unit 4 **Introduction to Distributed Operating Systems & Architecture** **(8L)**

4.1 What is a distributed system? Design goals
4.2 Types of distributed systems
4.3 Architectural Styles-Layered architectures, Object-based architectures, Resource- centered Architectures
4.4 System architecture–Centralized organization, Decentralized organizations, peer-to- peer systems, Hybrid architectures
4.5 Example Architectures-Network file system(NFS), Web-based distributed systems

Unit 5 **Mobile Operating Systems** **(6L)**

5.1 Introduction to Mobile Operating System- Features, Requirements of Mobile Operating System, Special Service Requirements
5.2 ARM & Intel architectures – Power management
5.3 Mobile OS architectures – Underlying OS, kernel structure & native level programming, Runtime issues, Approaches to power management
5.4 Commercial Mobile Operating Systems - Windows Mobile, iPhone OS (iOS), Android
5.5 A Comparative Study of Mobile Operating Systems (Palm OS, Android, Symbian OS, Blackberry OS, Apple iOS)

Course Outcomes:

On completion of this course, students will be able to:
CO1. Management of deadlocks and File System by operating system.
CO2. Scheduling storage or disk for processes.
CO3. Distributed Operating System and its architecture and the extended features in mobile Operating System.

Learning Resources:

1.Operating System Concepts - Avi Silberschatz, Peter Galvin, Greg Gagne, Student Edition, Wiley Asia.
2.Advanced Concepts in Operating Systems - M Singhal and NG Shivaratri, Tata McGraw Hill Inc, 2001 (Text Book).
3.Operating Systems: Internals and Design Principles- William Stallings, Prentice Hall of India.
4.Distributed Operating Systems Concepts and Design - Pradeep K. Sinha, PHI.
5. “Fundamentals of Mobile Computing” - Prasant Kumar Pattnaik, Rajib Mall, PHI Learning Pvt.Ltd, New Delhi – 2012.
6. “Modern Operating systems”- A. Tannenbum, Herbert Bos, Pearson Publication, 4th Edition.
7. “Distributed systems”- A. Tannenbum, Maarten van Steen, 3rd Edition.

Subject Code: CS-352
Subject Title: Computer Networks - II

Teaching Scheme	No. of Credits	Examination Scheme
36 Lectures	2	CE : 15 marks SEE: 35 marks

Course Objectives:

1. To understand different protocols of application layer.
2. To understand concepts of multimedia.
3. Explore the different methods used for Network/INTERNET security.

Course Contents

Unit 1 **Application Layer** **(10L)**

1.1 Domain Name System
- 1.1.1 Name Space-Flat name space, Hierarchical name space
- 1.1.2 Domain Name Space -Label, Domain name, FQDN, PQDN
- 1.1.3 Distribution of Domain Name Space-Hierarchy of name servers, zone, Root server, Primary and secondary servers
- 1.1.4 DNS in the Internet: Generic domains, Country domains, inverse domain
- 1.1.5 Resolution-Resolver, mapping names to address, mapping addresses to names, recursive resolution, iterative resolution, caching

1.2 Electronic Mail
- 1.2.1 Architecture-First scenario, second scenario, Third scenario, Fourth scenario
- 1.2.2 User agent-services of user agent, types of UA Format of e-mail
- 1.2.3 MIME-MIME header
- 1.2.4 Message transfer agent-SMTP
- 1.2.5 Message Access Agent: POP and IMAP

1.3 File Transfer
- 1.3.1 FTP-Communication over data control connection, File type, data structure, Transmission mode, anonymous FTP

Unit 2 **Multimedia** **(8L)**

2.1 Digitizing audio and video, Audio and Video compression

2.2 Streaming Stored audio/video
- 2.2.1 First approach
- 2.2.2 Second approach
- 2.2.3 Third approach
- 2.2.4 Fourth approach

2.3 Streaming live audio/video
- 2.3.1 Real time interactive audio/video- Characteristics, Time relationship, timestamp, Playback buffer, ordering multicasting, translation

2.4 RTP-Packet format

2.5 RTCP-Message types

2.6 Voice over IP-SIP,SIP sessionH.323

2.7 Architecture, Protocols

Unit 3 Cryptography and Network Security (9L)

3.1 Terminology: Cryptography, plain text and cipher text, cipher key, categories of cryptography-Symmetric key, asymmetric key
3.2 Encryption model
3.3 Symmetric key cryptography
- 3.3.1 Traditional ciphers – substitution cipher, shift cipher, Transposition cipher
- 3.3.2 Simple Modern ciphers-XOR, Rotation cipher, s-box, p-box
- 3.3.3 Modern round ciphers-DES
- 3.3.4Mode of operation-ECB, CBC, CFB, OFB

3.4 Asymmetric key cryptography-RSA
3.5 Security Services
- 3.5.1 Message confidentiality-With Symmetric key cryptography, with asymmetric key Cryptography
- 3.5.2 Message integrity -Document and fingerprint, message and message digest
- 3.5.3 Message authentication-MAC, HMAC
- 3.5.4 Digital signature
- 3.5.5 Entity Authentication-Passwords, Fixed passwords challenge-response

Unit 4 Security in the Internet (9L)

4.1 IPSecurity(IPSec)
- 4.1.1 Two modes
- 4.1.2 Two security protocols
- 4.1.3 Services provided by IPSec
- 4.1.4 Security association
- 4.1.5 Internet key exchange
- 4.1.6 Virtual private network

4.2 SSL/TLS
- 4.2.1 SSL services
- 4.2.2 Security parameters
- 4.2.3 Sessions and connections
- 4.2.4 Four protocols
- 4.2.5 Transport layer security

4.3 PGP
- 4.3.1 Security parameters
- 4.3.2 Services
- 4.3.3 PGP algorithms
- 4.3.4 Key rings
- 4.3.5 PGP certificates

4.4 Firewalls : Packet filter firewall , Proxy firewall

Course Outcomes :

On completion of the course, students will be able to–
CO1. Student will understand the different protocols of Application layer.
CO2. Develop understanding of technical aspect of Multimedia Systems.
CO3. Develop various Multimedia Systems applicable in real time.
CO4. Identify information security goals.

CO5. Understand, compare and apply cryptographic techniques for data security.

Learning Resources:

1.Data communications and networking - Behrouz Forouzan, McGraw Hill Pvt Ltd, 4th/5th Edition.
2. Computer Networks - Andrew S Tanenbaum, Pearson Education, 4 th/5th edition.
3. Cryptography and Network Security: Principles and Practice - William Stallings, Pearson Education, 7th edition.
4. Network Security Essentials: Applications and Standards (For VTU)-William Stallings, Pearson Education, 3rd edition.

Subject Code: CS-357

Subject Title: Practical course on CS-351(Operating System - II)

Teaching Scheme	No. of Credits:	Examination Scheme:
4 hrs 20 mins/week	2	CE: 15 marks SEE: 35 marks

Course Objectives:

1. To implement Banker's Algorithm for Deadlocks in Process management.
2. To study the concept of File System Management.
3. To study the implementation of various algorithms of Disk Scheduling.

Guidelines:

1. Operating system platform – Linux
2. Programming language – C

List of Assignments:

1.Simulation of Banker's algorithm of deadlock avoidance in processes of operating system (3 slots).

2.Simulation of File Allocation methods and free space management in storage – Contiguous allocation, Linked allocation, Indexed allocation. (4 slots)

3.Simulation of Disk Scheduling algorithms – FCFS, SSTF, Scan, Look (2 slots).

4. Assignment based on distributed and mobile OS (2 slots).

Course Outcomes

On completion of the course, students will be able to –

CO1. Implement practically the theoretical concepts of Operating System learnt.

CO2. Understand the management of deadlocks by operating system.

CO3. Learn the File System management.

CO4. Understand the Disk space management and scheduling for processes.

Subject Code: CS-353
Subject Title: Web Technologies I

Teaching Scheme	No. of Credits	Examination Scheme
36 Lectures	2	CE: 15 marks SEE: 35 marks

Course Objectives:

1. To Design dynamic and interactive Web pages.
2. To Learn Core-PHP.
3. To Learn PHP-Database handling.

Course Contents

Unit 1 **Introduction to HTML, HTTP and PHP** **(10L)**

1.1 Overview of HTML and Basic Tags
1.2 Creating Forms ,Tables, HTML5 Semantics
1.3 CSS basic concept , Three ways to use CSS, Box Model, Navigation Bar
1.4 Introduction to Web server and Web browser, HTTP basics
1.5 PHP Basics: Use of PHP, Lexical structure, Language basics

Unit 2 **Functions and Strings** **(8L)**

2.1 Defining and calling a function
2.2 Default parameters
2.3 Variable parameters, Missing parameters
2.4 Variable function, Anonymous function
2.5 Types of strings in PHP
  2.5.1 Single-Quoted Strings
  2.5.2 Double-Quoted Strings
  2.5.3 Variable Interpolation
  2.5.4 Here Documents
2.6 Printing Strings.
2.7 Encoding and escaping
2.8 Comparing strings
2.9 Manipulating and searching strings
2.10 Regular expression

Unit 3 **Arrays** **(6L)**

3.1 Arrays : Indexed , Associative
3.2 Identifying elements of an array
3.3 Storing data in array
3.4 Handling Arrays with Loops(For, while, For each, print_r)
3.5 Multidimensional arrays

3.6 Extracting multiple values
3.7 Converting between arrays and variables
3.8 Traversing arrays
3.9 Sorting
3.10 Action on entire array

Unit 4 Files and database handling (10L)

4.1 Working with files and directories
4.2 Opening and Closing, Getting information about file, Read/write to file
4.3 Splitting name and path from file, Rename and delete files
4.4 Reading and writing characters in file
4.5 Reading entire file
4.6 Random access to file data
4.7 Getting information on file
4.8 Ownership and permissions
4.9 Using PHP to access a database
    4.9.1 Database Connection.
    4.9.2 Record insertion, updation and deletion on tables
4.10 Relational databases and SQL
    4.10.1 PEAR DB basics
    4.10.2 Advanced database techniques

Unit 5 Handling email with PHP (2L)

5.1 Email background
5.2 Internet mail protocol
5.3 Structure of an email message
5.4 Sending email and validation of Email_id with php

Course Outcomes:

On completion of this course, students will be able to:
CO1. Understand how to develop dynamic and interactive Web pages.
CO2. Make database operations using Php.

Learning Resources:

1. Programming PHP - Rasmus Lerdorf and Kevin Tatroe, O'Reilly publication
2. Beginning PHP 5- Matt Doyle, Wrox publication
3. PHP web services – Lorna Jane Mitchell, Wrox publication
4. Mastering PHP- BPB Publication
5 .PHP cookbook - O'Reilly publication
6. PHP for Beginners - SPD publication
7. The Complete Reference PHP -Steven Holzner, Tata McGraw-Hill Publication

Reference Links:

1. www.php.net.in
2. www.W3schools.com
3. www.wrox.com

Subject Code: CS-354
Subject Title: Foundations of Data Science

Teaching Scheme	No. of Credits	Examination Scheme
36 Lectures	2	CE: 15 marks SEE: 35 marks

Course Objectives:

1. Provide students with knowledge and skills for data-intensive problem solving and scientific discovery.
2. Be prepared with a varied range of expertise in different aspects of data science such as data collection, visualization, processing and modeling of large data sets.
3. Acquire good understanding of both the theory and application of applied statistics and computer science based existing data science models to analyze huge data sets originating from diversified application areas.
4. Be better trained professionals to cater the growing demand for data scientists in industry.

Course Contents

Unit 1 **Introduction to Data Science** **(6L)**

1.1 Introduction to data science, The 3 V's: Volume, Velocity, Variety
1.2 Why learn Data Science?
1.3 Applications of Data Science
1.4 The Data Science Lifecycle
1.5 Data Scientist's Toolbox
1.6 Types of Data
- 1.6.1 Structured, semi-structured, Unstructured Data, Problems with unstructured data
- 1.6.2 Data sources
- 1.6.3 Open Data, Social Media Data, Multimodal Data, standard datasets
- 1.6.4 Data Formats
- 1.6.5 Integers, Floats, Text Data, Text Files, Dense Numerical Arrays, Compressed or Archived Data, CSV Files, JSON Files, XML Files, HTML Files , Tar Files, GZip Files, Zip Files, Image Files: Rasterized, Vectorized, and/or Compressed.

Unit 2 **Statistical Data Analysis** **(10L)**

2.1 Role of statistics in data science
2.2 Descriptive statistics
- 2.2.1 Measuring the Frequency
- 2.2.2 Measuring the Central Tendency: Mean, Median, and Mode
- 2.2.3 Measuring the Dispersion: Range, Standard deviation, Variance, Interquartile Range

2.3.Inferential statistics
- 2.3.1 Hypothesis testing, Multiple hypothesis testing, Parameter Estimation methods

2.4. Measuring Data Similarity and Dissimilarity
- 2.4.1 Data Matrix versus Dissimilarity Matrix
- 2.4.2 Proximity Measures for Nominal Attributes
- 2.4.3 Proximity Measures for Binary Attributes, Similarity and Dissimilarity, Distance measures: Euclidean, Manhattan, and Minkowski distance Proximity Measures for Ordinal Attributes.

2.5. Concept of Outlier, types of outliers, outlier detection methods.

Unit 3 Data Preprocessing (10L)

3.1 Data Objects and Attribute Types.
- 3.1.1 What Is an Attribute?, Nominal , Binary, Ordinal Attributes, Numeric Attributes, Discrete versus Continuous Attributes.

3.2 Data Quality: Why Preprocess the Data?

3.3 Data munging/wrangling operations

3.4 Cleaning Data - Missing Values, Noisy Data (Duplicate Entries, Multiple Entries for a Single Entity, Missing Entries, NULLs, Huge Outliers, Out-of- Date Data, Artificial Entries, Irregular Spacings, Formatting Issues - Irregular between Different Tables/Columns, Extra Whitespace, Irregular Capitalization, Inconsistent Delimiters, Irregular NULL Format, Invalid Characters, Incompatible Datetimes).

3.5 Data Transformation:
Rescaling, Normalizing, Binarizing, Standardizing, Label and One Hot Encoding Data reduction Data discretization.

Unit 4 Data Visualization (10L)

4.1 Introduction to Exploratory Data Analysis

4.2 Data visualization and visual encoding

4.3 Data visualization libraries

4.4 Basic data visualization tools
- 4.4.1 Histograms, Bar charts/graphs, Scatter plots, Line charts, Area plots, Pie charts, Donut charts.

4.5 Specialized data visualization tools
Boxplots, Bubble plots, Heat map, Dendrogram, Venn diagram, Treemap, 3D scatter Plots.

4.6 Advanced data visualization tools- Word Clouds

4.7 Visualization of geospatial data

Course Outcomes:

On completion of the course, students will be able to–

CO1. Perform Exploratory Data Analysis.

CO2. Obtain, clean/process, and transform data.

CO3. Detect and diagnose common data issues, such as missing values, special values, outliers, inconsistencies, and localization.

CO4. Demonstrate proficiency with statistical analysis of data.

CO5. Present results using data visualization techniques.

CO6. Prepare data for use with a variety of statistical methods and models and recognize how the quality of the data and the means of data collection may affect conclusions.

Learning Resources:

1.Data Science Fundamentals and Practical Approaches- Gypsy Nandi, Rupam Sharma, BPB Publications, 2020.

2. The Data Science Handbook - Field Cady, John Wiley & Sons, Inc, 2017

3. Data Mining Concepts and Techniques - Jiawei Han, Micheline Kamber, Jian Pei, Morgan Kaufmann, 2012, Third Edition

4. A Hands-On Introduction to Data Science - Chirag Shah, University of Washington Cambridge University Press

Subject Code: CS-358
Subject Title: Practical course on CS-353(Web Technologies I) and CS-354 (Foundation of Data Science)

Teaching Scheme	No. of Credits	Examination Scheme
4 hrs 20 mins/week	2	CE: 15 marks SEE: 35 marks

Course Objectives:

1. To Design dynamic and interactive Web pages.
2. To Learn Core-PHP.
3. To Learn PHP- Database handling.
4. To apply statistical, data preprocessing and visualization techniques on data sets.

Guidelines

- Operating Environment for web technologies: HTML 5.0, PHP 5.0, Web Server
- Operating Environment for Data Science: Linux + python

Course Contents

List of Assignments on web technologies I

1. Assignment Using HTML and CSS.

2. Assignment Using Bootstrap.

3. Assignment Using Function and String.

4. Assignment Using Arrays.

5. Assignment Using Files and Databases (PHP-PostgreSQL).

Suggested Assignments for Foundations of Data Science

Assignment 1: The Data Science environment
Getting introduced to Python IDLE, command line, online tools like google colaboratory and essential packages like NumPy, SciPy, pandas, scikit-learn, matplotlib, jupyter, beautiful-soup, etc.

Assignment 2: Loading the dataset
Select a dataset from a list of publicly available datasets at UCI Machine Learning Repository and load it using Pandas. (Import different data format files like .CSV,.htm,.json etc. Briefly describe what the dataset is about and size of the dataset (e.g. number of tables, number of instances and attributes, etc.).

Assignment 3: Basic statistical operations
Apply basic statistical operations on a dataset. For example - compute the mean, median, mode, range, quartiles, and variance for one or more attributes.

Assignment 4: Data preprocessing
Apply data preprocessing techniques that are likely required for the dataset.

1) Partition them into appropriate number of bins by equal-frequency as well as equal-width partitioning.

2) Use smoothing by bin means to smooth the data based on the above partitioning.
3) Normalize the attribute based on min-max normalization and z-score normalization. Comment on which method you would prefer to use for partitioning, smoothing, and normalization for the given attribute.

Assignment 5: Data Visualization with matplotlib

View the data using various 2-D, 3-D plots and charts, setting styles, saving the figures, customizing the legends, multiple subplots,

Course Outcomes:

On completion of the course, students will be able to–

CO1. Understand how to develop dynamic and interactive Web Page

CO2. Prepare data for use with a variety of statistical methods and recognize how the quality of the data may affect conclusions.

CO3. Perform exploratory data analysis

Subject Code: CS-355
Subject Title: Object Oriented Programming using Java

Teaching Scheme	No. of Credits	Examination Scheme
36 Lectures	2	CE: 15 marks SEE: 35 marks

Course Objectives:

1. To learn Object Oriented Programming language.
2. To study various java programming concept like Interface, File and Exception Handling etc.
3. To design User Interface using Swing and AWT.

Course Contents

Unit 1 **An Introduction to Java** **(6L)**

1.1 Object Oriented Programming Concepts
1.2 A short history of Java
1.3 Features OR Buzzwords of Java
1.4 Java Environment
1.5 Simple Java Program
1.6 Java Tools – jdb, javap, javadoc
1.7 Types of Comments
1.8 Data Types
1.9 Final Variable
1.10 Declaring 1D, 2D Array, Accepting Input (Command Line Arguments, BufferedReader, Scanner)

Unit 2 **Objects and Classes** **(7L)**

2.1 Defining your own classes
2.2 Access Specifiers (public, protected, private, default)
2.3 Array of Objects
2.4 Constructors, Overloading Constructors and Use of ‘this’ keyword
2.5 static block, static fields And methods
2.6 Predefined Classes
  2.6.1Object Class, Methods (equals(), toString(),hashcode(), getClass()
  2.6.2String Class And StringBufferClass,Formatting String data using format() method
2.7 Creating , Accessing And Using Packages
2.8 Wrapper Classes

Unit 3 **Inheritance and Interface** **(8L)**

3.1 Inheritance Basics (extends Keyword) and Types of Inheritance
3.2 Superclass, Subclass and use of Super Keyword
3.3 Method Overriding and runtime polymorphism
3.4 Use of final keyword related to method and class
3.5 Use of abstract class and abstract methods
3.6 Defining and Implementing Interfaces
3.7 Runtime polymorphism using interface
3.8 Concept of Marker and Functional Interfaces

Unit 4 Exception and File Handling (5L)

4.1 Dealing with errors , Exception class, Checked And Unchecked ExceptionCatching
4.2 Exceptions, Multiple Catch Block, Nested try block
4.3 Creating User Defined Exception
4.4 Introduction to Files And Streams
4.5 Input-OutputStream : FileInput/OutputStream, BufferedInput/OutputStream, DataInput/OutputStream
4.6 Reader-Writer : FileReader/Writer, BufferedReader/Writer, InputStreamReader, OutputStreamWriter

Unit 5 User Interface with AWT and Swing (10L)

5.1 What is AWT? What is Swing? Difference between AWT and Swing
5.2 The MVC Architecture And Swing
5.3 Layouts And Layout Managers
5.4 Containers And Components – JFrame, JButton, JLabel, JText, JTextArea, JCheckBox and JRadioButton, JList, JComboBox, JMenu And related Classes
5.5 Dialogs (Message, Confirmation, Input), JFileChooser, JColorChooser
5.6 Event Handling: Event Sources, Listeners
5.7 Adapters and Anonymous Inner Class

Course Outcomes:

On completion of the course, students will be able to:
CO1. Understand the concept of classes, object, packages and Collections.
CO2. To develop GUI based application.

Learning Resources:

1. Complete reference Java by Herbert Schildt(5th edition).
2. Java 2 programming black books, Steven Horlzner.
3. Programming with Java , A primer ,Forth edition , By E. Balagurusamy .
4. Core Java Volume-I-Fundamentals, Eighth Edition, Cay S. Horstmann, Gary Cornell, Prentice Hall, Sun Microsystems Press.

Subject Code: CS-356
Subject Title: Theoretical Computer Science

Teaching Scheme	No. of Credits	Examination Scheme
36 Lectures	2	CE: 15 marks SEE: 35 marks

Course Objectives:

1. To understand the Finite Automata, Pushdown Automata and Turing Machine.
2. To understand the Regular Language, Context Free Language, Context Sensitive Language and Unrestricted Language.
3. To understand the relation between Automaton and Language.

Course Contents

Unit 1 **Finite Automaton** **(10L)**

1.1 Introduction: Symbol, Alphabet, String, Prefix & Suffix of Strings, Formal Language, Operations on Languages
1.2 Deterministic finite Automaton – Definition, DFA as language recognizer, DFA as pattern recognizer
1.3 Nondeterministic finite automaton – Definition and Examples
  1.3.1 NFA To DFA (Myhill Nerode Method)
  1.3.2 NFA with ε- transitions Definition and Examples
  1.3.3 NFA with ε-Transitions to DFA & Examples
1.4 Finite automaton with output – Mealy and Moore machine, Definition and Examples
1.5 Minimization of DFA, Algorithm & Problem using Table Method

Unit 2 **Regular Expressions and Languages** **(6L)**

2.1 Regular Expressions (RE): Definition & Example
  2.1.1 Regular Expressions Identities
  2.1.2 Regular language-Definition and Examples
2.2 Conversion of RE to FA-Examples
2.3 Pumping lemma for regular languages and applications
2.4 Closure Properties of regular Languages

Unit 3 **Context-Free Grammars and Languages** **(10L)**

3.1 Grammar - Definition and Examples
3.2 Derivation-Reduction - Definition and Examples
3.3 Chomsky Hierarchy
  3.3.1 CFG: Definition & Examples. LMD, RMD, Parse Tree
  3.3.2 Ambiguous Grammar: Concept & Examples
  3.3.3 Simplification of CFG: Removing Useless Symbols, Unit Production, ε-production and Nullable Symbol
3.4 Normal Forms: Greibach Normal Form (GNF) and Chomsky Normal Form (CNF)
3.5 Regular Grammar: Definition
  3.5.1 Left linear and Right Linear Grammar-Definition and Example
  3.5.2 Equivalence of FA & Regular Grammar

3.5.3 Construction of regular grammar equivalent to a given DFA
3.5.4 Construction of a FA from the given right linear grammar

Unit 4 Push Down Automata (5L)

4.1 Definition of PDA and examples
4.2 Construction of PDA using empty stack and final State method: Examples using stack method
4.3 Definition DPDA & NPDA, their correlation and Examples of NPDA
CFG (in GNF) to PDA: Method and examples

Unit 5 Turing Machine (5L)

5.1 The Turing Machine Model, Definition and Design of TM
5.1.1 Problems on language recognizers
5.1.2 Language accepted by TM
5.2 Types of Turing Machines (Multitrack TM, Two-way TM, Multitape TM, Non-deterministic TM)
5.3 Introduction to LBA (Basic Model) & CSG. (Without Problems)

Course Outcomes:

On completion of this course, students will be able to:
CO1. Understand the use of automata during language design.
CO2. Relate various automata and Languages.

Learning Resources:

1. Introduction to Automata Theory, Languages and Computation- John E. Hopcraft, Rajeev Motwani, Jeffrey D. Ullman, Third Edition, Pearson Education Publication, 2008
2. Introduction to Automata theory, Languages and computation - John E. Hopcroft and JeffreyUllman – Narosa Publishing House, 1995
3. Theory of Computer Science Automata, Languages and Computation - K.L.P. Mishra, N. Chandrasekaran, Publication- Prentice Hall of India, 2008
4. Introduction to Computer Theory Daniel I. A. Cohen – 2nd edition – John Wiley & Sons,1996
5. Introduction to Languages and The Theory of Computation - John C. Martin, The McGraw- Hill, Fourth Edition, 2011

Subject Code: CS -359

Subject Title: Practical course on CS 355(Object Oriented Programming using Java) and Project Design

Teaching Scheme	No. of Credits	Examination Scheme
4 hrs 20 mins/week	2	CE: 15 marks SEE: 35 marks

Course Objectives

1. Students are able to learn basic programming of core java.
2. Learn the Basic Designing of Project using Different tools, techniques and Data Models.

Guidelines

- Operating Environment: Linux, Java, JDK 1.7 and above.

Course Contents

List of Assignments

Assignment 1: Java Tools and IDE, Simple java programs

Program to Use of java tools like java, javac, jdb and javadoc,Use of IDE – Eclipse (demo) ,creating simple classes and creating objects.

Assignment 2: Array of Objects and Packages

Program to Creating an array of objects, Creating a package(Using package command), Using packages (Using import command),use of Static method, Static variable and Static class.

Assignment 3: Inheritance and Interfaces

Programs to implement inheritance in java, to define abstract classes, To define and use interfaces, Use predefined interfaces like Cloneable.

Assignment 4: Exception and File Handling

Demonstrate exception handling mechanism in java, Defining user defined exception classes

Use of try, catch, throw, throws and finally keyword, I/O streams, Byte and Character stream classes, The File class, Performing Input/Output operations using console and files.

Assignment 5: GUI and Event handling

To demonstrate GUI creation using Swing package and Layout managers, AWT and Swing concept, Layout managers in java Containers and Components, Adding components to containers.

Project Design

1. **Project Synopsis**
2. **Analysis Modelling –** Use Case Model, Class Model, Deployment Model, Artifact Model.
3. **Design Model** – Sequence Diagram, State Machine Model, Activity Model, User Interface design.

Course Outcomes:

On completion of this course, students will be able to:

CO1. To develop GUI based application using AWT and Swing.

CO2. Learn the Basic Designing of Project using Different tools and Data Models.

Subject Code:

Subject Title: Python Programming (Elective)

Teaching Scheme	No. of Credits	Examination Scheme
36 Lectures	2	CE: 15 marks SEE: 35 marks

Course Objectives:

1. To introduce programming concepts using python.
2. Student should be able to develop Programming logic using python.
3. To develop basic concepts and terminology of python programming.
4. To test and execute python programs.

Course Contents

Unit 1 **An Introduction to Python** **(3L)**

1.1 Introduction to Python
  1.1.1The Python Programming Language
  1.1.2 History, features, Applications, Installing Python
1.2 Running Simple Python program
1.3 Basics of Python
1.4 Standard data types - basic, none, Boolean (true & False), numbers, Variables, Constants, Python identifiers and reserved words, Lines and indentation, multi-line statements and Comments, Input/output with print and input, functions Declaration, Operations on Data such as assignment, arithmetic, relational, logical and bitwise operations, dry run, Simple Input and output etc.

Unit 2 **Control Statements** **(4L)**

2.1 Sequence Control – Precedence of operators, Type conversion
2.2 Conditional Statements: if, if-else, nested if-else
2.3 Looping- for, while, nested loops, loop control statements (break, continue, pass)
2.4 Strings: declaration, manipulation, special operations, escape character, string formatting operator, Raw String, Unicode strings, Built-in String methods

Unit 3 **Lists, functions, tuples and dictionaries, Sets** **(7L)**

3.1 Python Lists: Concept, creating and accessing elements, updating & deleting lists, traversing a List, reverse Built-in List Operators, Concatenation, Repetition, In Operator, built-in List functions and methods
3.2 Functions: Definitions and Uses, Function Calls, Type Conversion Functions, Math Functions, Composition, Adding New Functions, Flow of Execution, Parameters and Arguments, Variables and Parameters, Stack Diagrams, Void Functions, Anonymous functions Importing with from, Return Values, Boolean Functions, More Recursion, Functional programming tools - filter(), map(), and reduce(),recursion, lambda forms
3.3 Tuples and Dictionaries: Tuples, Accessing values in Tuples, Tuple Assignment, Tuples as return values, Variable-length argument tuples, and Basic tuples operations, Concatenation, Repetition, in Operator, Iteration, Built-in tuple functions, indexing, slicing and matrices. Creating a Dictionary, Accessing Values in a dictionary, Updating Dictionary, Deleting Elements from Dictionary, Properties of Dictionary keys, Operations

in Dictionary, Built-In Dictionary Functions, Built-in Dictionary Methods. Sets- Definition, transaction of set(Adding, Union, intersection), working with sets.

Unit 4 Modules, Working with files, Exception handling (4L)

4.1 Modules: Importing module, Creating & exploring modules, Math module, Random module, Time module

4.2 Packages: Importing package, creating package, examples

4.3 Working with files: Creating files and Operations on files (open, close, read, write), File object attributes, file positions, Listing Files in a Directory, Testing File Types, Removing files and directories, copying and renaming files, splitting pathnames, creating and moving directories

4.4 Regular Expression- Concept of regular expression, various types of regular expressions, using match function

4.5 Exception Handling: Built-in Exceptions, Handling Exceptions, Exception with Arguments, User-defined Exceptions

Unit 5 Demonstration Programming Assignments (18L)

Out of 36 lectures, 18 are assigned for demonstration. Teacher should give demonstration of various programs mentioned below in the classroom or in the laboratory as per their convenience.

Programming assignments should be done individually by the student in their respective login from the list given in Lab book. The codes should be uploaded on either the local server, Moodle, Github or any LMS.

Assignment 1 - Python Basics.

Assignment 2 – Arrays, Strings, and Functions.

Assignment 3 - List, Tuples, Sets, and Dictionary.

Assignment 4 - File Handling and Date-Time.

Assignment 5 - Exception handling and Regular expression.

Course Outcomes: On completion of this course, students will be able to:

CO1. Develop logic for problem solving.

CO2. Determine the methods to create and develop Python programs by utilizing the data structures like lists, dictionaries, tuples and sets.

CO3. To be familiar about the basic constructs of programming such as data, operations, conditions, loops, functions etc.

CO4.To write python programs and develop a small application project.

Learning Resources:

1. An Introduction to Computer Science using Python 3 - Jason Montojo, Jennifer Campbell, Paul Gries, The pragmatic bookshelf-2013.

2. “Beginning Python: Using Python and Python 3.1- James Payne, Wrox Publication

3. Introduction to Computer Science Using Python- Charles Dierbach, Wiley Publication Learning with Python “, Green Tea Press, 2002

4. Introduction to Problem Solving with Python - E Balgurusamy, TMH publication-2016

5. Beginning Programming with Python for Dummies Paperback – 2015- John Paul Mueller

6. Object-oriented Programming in Python - Michael H. Goldwasser, David Letscher, Pearson Prentice Hall-2008

Subject Code:

Subject Title: Git and Github Programming (Elective)

Teaching Scheme	No. of Credits	Examination Scheme
36 Lectures	2	CE:15 marks SEE: 35 marks

Course Objectives:

1. To create and manage multiple versions of source code.
2. To collaborate with the other coders.
3. To configure and use Git locally.
4. Link local Git repositories to remote Github account.
5. Describe the benefits of an automated version control system.

Course Contents

Unit 1 **Introduction** **(11L)**

1.1 Version Control System.
- 1.1.1 What is a Version Control System (VCS)?
- 1.1.2 Distributed vs Non-distributed VCS.

1.2 What is Git and where did it come from?
1.3 Alternatives to Git.
1.4 Cloud-based solutions (Github, Gitlab, BitBucketetc)
1.5 Installation and Configuration.
1.6 Key Terminology: Clone, Working Tree, Checkout, Staging area, Add, Commit, Push, Pull, Stash.

Unit 2 **Git - Repository Actions** **(8L)**

2.1 Git - Local Repository Actions.
2.2 Git - Remote Repository Actions: Removing, adding, checking, Creating, Cloning, Updating
2.3 Tagging in Git and its types.

Unit 3 **Branching in Git** **(8L)**

3.1 What is a branch?
3.2 Listing branches.
3.3 Creating new branch.
3.4 Checkout branch.
3.5 Pushing branches.
3.6 Pulling branches.

Unit 4 **Merging and workflow in Git** **(9L)**

4.1 Fetching Changes (git fetch).
4.2 Rebasing (git rebase).
4.3 Git Pull.
4.4 Git Workflows.
- 4.4.1 Different ways of using Git.
- 4.4.2 Centralised.

4.4.3 Feature Branch.
4.4.4 Gitflow Workflow.
4.4.5 Forking Workflow.

Course Outcomes:

On Completion of this course, students will be able to –

CO1. A complete understanding of what a version control system is, including Git and GitHub theory.
CO2. Understand Theoretical and Practical concepts covering the differences between Git & GitHub, and their purposes.
CO3. Get Practical hands on guide to GitHub core concepts.
CO4. Learn to create and manage code repositories.
CO5. Create Github repositories and pushing the local repository to the Remote repository.

Learning Resources:

1. Pro Git, Scott Chacon and Ben Straub, Second Edition, Apress, 2023
2. Introduction to Git and Github, Bobby Ilive, (E-Book), 2023.
3. Version Control with Git and Github, Alex Magana and Joseph Muli, Packt Publishing, 2018.
4. Beginning Git and Github, Mariot Tsitoara, Apress , 2019.
5. Git Pocket Guide, Richard E. Silverman, O'Reilly Media, Inc, 2013.

Reference link:

1. https://www.obooko.com/free-IT-and-computer-programming-books/introduction-to-git-and-github/
2. https://docs.github.com/en/get-started/quickstart/hello-world.
3. https://www.freecodecamp.org/news/git-and-github-for-beginners/
4. https://www.w3schools.com/git/git_intro.asp?remote=github
5. https://www.geeksforgeeks.org/ultimate-guide-git-github/
6. https://www.datacamp.com/tutorial/github-and-git-tutorial-for-beginners
7. https://product.hubspot.com/blog/git-and-github-tutorial-for-beginners

Subject Code:
Subject Title: UI/UX Design (Elective)

Teaching Scheme	No. of Credits	Examination Scheme
36 Lectures	2	CE: 15 marks SEE: 35 marks

Course Objectives:

1. Learning the importance of UI/UX design.
2. Exploring building blocks of UI.
3. Exploring UI/UX Design common web features.
4. Exploring UI/UX Design from real life applications.
5. A case study with some of the best web sites.

Course Contents

Unit 1 **UI/UX design Introduction** **(9L)**

1.1 Introduction to User Interface Design (UI).
1.2 The Relationship between UI and UX.
1.3 Roles in UI /UX.
1.4 Approaches to Screen Based UI, Template Vs Content, Formal and Active elements interface Design, Composing the elements of Interface Design.
1.5 UI Design Process, Visual Communication designs, Components in Interface Design.
1.6 UI Vs UX design-Core Stages of Design Thinking- Divergent and Convergent Thinking- Brainstorming and Game storming - Observational Empathy.

Unit 2 **UX Design Process** **(6L)**

2.1 Introduction to User Experience.
2.2 Understanding User Experience.
2.3 Defining the UX Design Process and its Methodology.
2.4 Research in User Experience Design.
2.5 Tools and Method used for Research.
2.6 User Needs and its Goals.

Unit 3 **Wireframing, Prototyping And Testing** **(9L)**

3.1 Sketching Principles, Sketching Red Routes.
3.2 Responsive Design.
3.3 Wire framing-Creating Wire flows.
3.4 Building a Prototype, Building High -Fidelity Mockups.
3.5 Designing Efficiently with Tools (Figma, Imotions, Morae).
3.6 Interaction Patterns.
3.7 Conducting Usability Tests.
3.8 Synthesizing Test Findings.
3.9 Prototype Iteration.

Unit 4 Research, Designing, Ideating and Information Architecture (8L)

4.1 Writing problem statements and identifying appropriate research methods.
4.2 Creating Personas.
4.3 Solution Ideation.
4.4 Creating User Stories and Creating Scenarios.
4.5 Flow Diagrams Flow Mapping.
4.6 Information Architecture.

Unit 5 Great Website Designs (4L)

5.1 Case studies on great website designs
apple.com, airbnb.com, about.instagram.com, stripe.com, revolut.com

Course Outcomes:
On completion of this course, students will be able to:
CO1. Build UI for user Applications.
CO2. Evaluate UX design of any product or application.
CO3. Demonstrate UX skills in product development.
CO4. Implement Sketching principles.

Learning Resources:
1. Wire framing Essentials - Matthew J. Hamm Packt Publishing Limited, Illustrated edition (24 January 2014)
2. Hands on UX design for developers -Elvis Canziba, Packt Publishing
3. Handbook of Design-Thinking eBook.
4. “UX for Beginners” - Joel Marsh, O’Reilly , 2022

Reference Links:
1. https://www.interaction_design.org/literature
2. https://www.nngroup.com/art

Subject Code: CS-361
Subject Title: Fundamentals of Artificial Intelligence

Teaching Scheme	No. of Credits	Examination Scheme
36 Lectures	2	CE : 15 marks SEE: 35 marks

Course Objectives:

1. To understand the core concepts of Artificial Intelligence (AI).
2. To learn the primary algorithms and techniques that power AI.
3. To understand fundamentals of Game Theory.
4. To acquire the ability to identify real-world problems that can be solved or enhanced Using AI techniques and algorithm.

Course Contents

Unit 1 Introduction to Artificial Intelligence (5L)

1.1 What is AI?
1.2 The Foundations of Artificial Intelligence
1.3 The history of Artificial Intelligence
1.4 Applications of AI
1.5 Early work in AI and related fields
1.6 AI problems and Techniques.

Unit 2 Problem solving and Searching Techniques (12L)

2.1 Defining AI problems as a State Space Search: example
2.2 Production Systems
2.3Search and control strategies-BFS,DFS
2.4 Problem Characteristics
2.5 Production System characteristics
2.6 Issues in Design of Search Programs
2.7 Additional problems
2.8 Heuristic search techniques:
- 2.8.1. Generate and test
- 2.8.2. Hill Climbing
- 2.8.3. Best-First search
- 2.8.4. Problem Reduction
- 2.8.5. Constraint Satisfaction
- 2.8.6. Means-Ends Analysis
- 2.8.7. A*,AO* algorithm.

Unit 3 Knowledge Representation (12L)

3.1 Representations and Mappings
3.2 Approaches to Knowledge Representation
3.3 Representing Simple facts in Logic

3.4 Representing instance and ISA Relationships
3.5 Knowledge representation method
3.6 Propositional Logic
3.7 Predicate logic
3.8 Computable Functions and Predicates
3.9 Resolution
3.10 Forward and backward chaining
3.11 Game Playing-Minimax Search Procedures
3.12 Adding alpha-beta cutoffs.

Unit 4 Slot and Filter Structures (7L)

4.1 Weak structures
4.2 Semantic networks
4.3 Frames
4.4 Strong Structures
4.5 Conceptual Dependency
4.6 Scripts
4.7 CYC

Course Outcomes:

On completion of the course, students will be able to:
CO1. Identify real-world problems that can be solved or enhanced using AI techniques and algorithms.
CO2. Apply basic AI techniques to solve real-world problems,
CO3. Stay informed about the latest advancements and future trends in the field of AI

Learning Resources:

1. Artificial Intelligence, By Elaine Rich and Kevin Knight , 3rd Edition Tata McGraw Hill,
2. Introduction to Artificial Intelligence and Expert System, By Dan Patterson , 2nd Printing, Prentice Hall of India Pvt. Ltd., New Delhi, 1997

Subject Code: CS-362
Subject Title: Software Testing

Teaching Scheme:	No. of Credits:	Examination Scheme:
36 Lectures	2	CE:15marks SEE: 35marks

Course Objectives:

1. To provide the knowledge of software testing techniques.
2. To understand how testing methods can be used as an effective tool in quality assurance of software.
3. To provide skills to design test case plan for testing software.
4. To provide knowledge of latest testing methods.

Course Contents

Unit 1 **Introduction to Software Testing** **(5L)**

1.1 Basics of Software Testing – faults, errors and failures Testing objectives
1.2 Principles of testing
1.3 Testing and debugging
1.4 Testing metrics and measurements
1.5 Verification and Validation
1.6 Testing Life Cycle

Unit 2 **Software Testing Strategies & Techniques** **(10L)**

2.1 Testability - Characteristics lead to testable software
2.2 Test characteristics
2.3 Test Case Design for Desktop, Mobile, Web application using Excel
2.4 White Box Testing - Basis path testing, Control Structure Testing
2.5 Black Box Testing- Boundary Value Analysis, Equivalence partitioning. Differences between BBT & WBT

Unit 3 **Levels of Testing** **(10L)**

3.1 A Strategic Approach to Software Testing
3.2 Test strategies for conventional Software
3.2 Unit testing
3.3 Integration testing–Top-Down, Bottom-up integration
3.4 System Testing–Acceptance, performance, regression, Load/Stress testing, Security testing, Internationalization testing
3.5 Alpha, Beta Testing
3.6 Usability and accessibility testing
3.7 Configuration, compatibility testing

Unit 4 **Testing Web Applications** **(6L)**

4.1 Dimension of Quality,
4.2 Error within a Web App Environment
4.3 Testing Strategy for Web App Test Planning
4.4 The Testing Process–an overview

Unit 5 **Agile Testing** **(5L)**

5.1 Agile Testing
5.2 Difference between Traditional and Agile testing
5.3 Agile principles and values
5.4 Agile Testing Quadrants
5.5 Automated Tests.

Course Outcomes:

On completion of the course, students will be able to:
CO1. Understand different Software Testing Strategies and techniques.
CO2. Approach testing tasks with a structures and systematic approach.

Learning Resources:

1. Software Engineering – A Practitioners Approach, Roger S. Pressman, 7th Edition, TataMc GrawHill, 20.
2. Effective Methods of Software Testing, William E Perry, 3rdEdition, Wiley Publishing Inc.
3. Managing the Testing Process: Practical Tools and Techniques for Managing Hardware and Software Testing, Rex Black, MicrosoftPress,1999.
4. Agile Testing : A Practical Guide for Testers and Agile Teams, Lisa Crispin and Janet Gregory,1stEdition,Addison-WesleyProfessional,2008.
5. Software Testing Principles and Practices By Srinivasan Desikan, Gopala swamy Ramesh, Pearson.

Subject Code:CS-367

Subject Title: Practical course on CS-361(Fundamentals of Artificial Intelligence) and CS-362(Software Testing)

Teaching Scheme:	No. of Credits:	Examination Scheme:
4 hrs 20 mins/week	2	CE:15marks SEE: 35marks

Course Objectives:

1. Provide students with practical experience in implementing AI algorithms and techniques.
2. Assignments on TestProject can cover topics related to test automation, including creating and executing tests, working with different testing frameworks, integrating with various technologies, and more.

Course Contents

Practical Based On CS-361 Artificial Intelligence

Assignment 1: Based on Heuristic Search Techniques.
Assignment 2: Based on Knowledge Representation Techniques.
Assignment 3: Based on Gaming Techniques.
Assignment 4: A case study.

Practical Based On CS-362 Software Testing Tools

Assignment 5: Creating Tests Offline and Local Test Execution on TestProject.
Assignment 6: Based on Parameterization and Data-Driven Testing.
Assignment 7: Based on Offline Test Reports and Custom Addons Offline.
Assignment 8: CI/CD Integration in an Offline Environment and Exploratory Testing in an Isolated Environment.

Course Outcomes:

On completion of this course students will be able to :

CO1. Students will be able to implement and apply various AI algorithms and techniques to solve real-world problems.

CO2. Test Project documentation for any specific features or functions that may have limitations in an offline mode. Then also by using it students can do the above assignment 5 to assignment 8 in an offline mode.

Subject Code: CS-363
Subject Title: Web Technologies II

Teaching Scheme	No. of Credits	Examination Scheme
36 Lectures	2	CE: 15 marks SEE: 35 marks

Course Objectives:

1. To Learn different technologies used at client-Side Scripting Language
2. To Learn XML and XML parsers.
3. To Learn Java Script to program the behavior of web pages.
4. To Learn to develop web-based application.
5. To Learn AJAX to make our application more dynamic.

Course Contents

Unit 1 **Introduction to Web Techniques** **(6L)**

1.1 Variables
1.2 Server information
1.3 Processing forms
1.4 Setting response headers
1.5 Maintaining state
1.6 PHP error handling

Unit 2 **XML** **(6L)**

2.1 What is XML?
2.2 XML document Structure
2.3 PHP and XML
2.4 XML parser
2.5 The document object model
2.6 The simple XML extension
2.7 Changing a value with simple XML

Unit 3 **JavaScript and Jquery** **(10L)**

3.1 Overview of JavaScript
3.2 Object Orientation and JavaScript Basic Syntax(JS datatypes, JS variables) Primitives, Operations and Expressions , Screen Output and keyboard input (Verification and Validation)
3.3 JS Control statements and JS Functions
3.4 JavaScript HTML DOM Events (onmouseup, onmousedown, onclick, onload, onmouseover, onmouseout).
3.5 JS Strings and JS String methods
3.6 JS popup boxes (alert, confirm, prompt)
3.7 Jquery library, Including jquery library in page Jquery selector, DOM manipulation using jquery.

Unit 4 AJAX (6L)

4.1 Introduction of AJAX
4.2 AJAX web application model
4.3 AJAX –PHP framework
4.4 Performing AJAX validation
4.5 Handling XML data using php and AJAX
4.6 Connecting database using php and AJAX

Unit 5 PHP framework CodeIgniter (8L)

5.1 CodeIgniter - Overview, Installing CodeIgniter, Application Architecture
5.2 MVC Framework , Basic concept of CodeIgniter, Libraries
5.3 Working with databases
5.4 Load external JS and CSS page & redirecting from controller , Adding JS and CSS ,
5.5 Page redirection.
5.6 Loading dynamic data on page & session management, cookies management

Course Outcomes:

On completion of this course, students will be able to :
CO1. Build a dynamic website.
CO2. Using MVC based framework easy to design and handling the errors in dynamic website.

Learning Resources:

1 Programming PHP By Rasmus Lerdorf and Kevin Tatroe, O'Reilly publication
2.Beginning PHP 5, Wrox publication
3.Mastering PHP, BPB Publication
4. JQuery CookBook, O'reilly Publication.
5.Ajax Programming for the Absolute Beginner- Jerry Lee Ford, Jr, Course Technology PTR
6. Professional Codeigniter By Thomas Myer ,Wrox Publication.
7. Codeigniter 2 CookBook By Rob Foster , PACKT Publication.

Reference Links:

1. www.php.net.in
2. www.W3schools.com
3. https://www.tutorialspoint.com/codeigniter/index.htm
4. https://api.jquery.com/
5. http://codeigniter.com/docs

Subject Code: CS-364
Subject Title: Data Analytics

Teaching Scheme	No. of Credits	Examination Scheme
36 Lectures	2	CE: 15 marks SEE: 35 marks

Course Objectives:

1. Deploy the Data Analytics Lifecycle to address data analytics projects.
2. Develop in depth understanding of the key technologies in data analytics.
3. Apply appropriate analytic techniques and tools to analyze data, create models, and identify insights that can lead to actionable results.

Course Contents

Unit 1 Introduction to Data Analytics (6L)

1.1 Concept of data analytics
1.2 Data analysis vs Data analytics
1.3 Types of analytics
Diagnostic Analytics, Predictive Analytics, Prescriptive Analytics, Exploratory Analysis, Mechanistic Analysis
1.4 Mathematical models - Concept
Model evaluation: metrics for evaluating classifiers - Class imbalance - AUC, ROC (Receiver-Operator Characteristic) curves, Evaluating value prediction models.

Unit 2 Machine Learning Overview (6L)

2.1 Introduction to Machine Learning, deep learning, Artificial intelligence
2.2 Applications for machine learning in data science
2.2.1 The modeling process
Engineering features and selecting a model, Training the model, Validating the model, Predicting new observations
2.2.2 Types of machine learning
Supervised learning, Unsupervised learning, Semi-supervised learning, ensemble Techniques
2.3 Regression models
2.3.1 Linear Regression
2.3.2 Polynomial Regression
2.3.3 Logistic Regression
2.4. Concept of classification, clustering and reinforcement learning

Unit 3 Mining Frequent Patterns, Associations, and Correlations (12L)

3.1 What kind of patterns can be mined
Class/Concept Description: Characterization and Discrimination, Mining Frequent Patterns, Associations, and Correlations, Classification and Regression for Predictive Analysis, Cluster Analysis, Outlier Analysis.

3.2 Mining frequent patterns - Market Basket Analysis.
3.3 Frequent Itemsets, Closed Itemsets, and Association Rules
3.4 Frequent Itemset Mining Methods
3.5 Apriori Algorithm
3.6 Generating Association Rules from Frequent Itemsets
3.7 Improving efficiency of apriori algorithm
3.8 Frequent pattern growth (FP-growth) algorithm

Unit 4 Social Media and Text Analytics (12L)

4.1 Overview of social media analytics
  4.1.1 Social Media Analytics Process
  4.1.2 Seven layers of social media analytics
  4.1.3 accessing social media data
4.2 Key social media analytics methods
4.3 Social network analysis
  4.3.1 Link prediction, Community detection, Influence maximization, Expert finding, Prediction of trust and distrust among individuals
4.4 Introduction to Natural Language Processing
4.5 Text Analytics: Tokenization, Bag of words, Word weighting: TF-IDF, n- Grams, stop words, Stemming and lemmatization, synonyms and parts of speech tagging
4.6 Sentiment Analysis
4.7 Document or text summarization
  4.7.1 Trend analytics
4.8 Challenges to social media analytics

Course Outcomes:

On completion of the course, student will be able to–
CO1. Use appropriate models of analysis, assess the quality of input, and derive insight from results.
CO2. Analyze data, choose relevant models and algorithms for respective applications.
CO3. Understand different data mining techniques like classification, prediction, clustering and association rule mining.
CO4. Apply modeling and data analysis techniques to the solution of real world business problems.

Learning Resources:

1. Data Science Fundamentals and Practical Approaches, Gypsy Nandi, Rupam Sharma, BPB Publications, 2020.
2. The Data Science Handbook, Field Cady, John Wiley & Sons, Inc, 2017
3. Data Mining Concepts and Techniques, Jiawei Han, Micheline Kamber, Jian Pei, Morgan Kaufmann, Third Edition, 2012.
4. A Hands-On Introduction to Data Science, Chirag Shah, University of Washington Cambridge University Press
5. The Data Science Design Manual, Steven S. Skiena, Springer, 2017
6. Introducing data science: big data, machine learning, and more, using Python tools, Cielen D, Meysman A. D., & Ali M., Manning Publications Co., 2016

Subject Code: CS-368
Subject Title: Practical Course on CS-363(Web Technologies II) and CS-364(Data Analytics)

Teaching Scheme	No. of Credits	Examination Scheme
4 hrs 20 mins/week	2	CE : 15 marks SEE: 35 marks

Course Objectives:

1. To Learn different technologies used at client Side Scripting Language.
2. To Learn XML and XML parsers.
3. To Learn PHP framework for effective design of web application.
4. To Learn Java Script to program the behavior of web pages.
5. To Learn AJAX to make our application more dynamic.
6. Framework has some utility features that make easy to write API in more efficient way than Core PHP.

Guidelines

- Operating Environment :Linux, HTML, PHP5.0 and above, Python

Course Contents

List of Assignments based on Web Technologies- II

1. Assignment on Self Processing Forms, Sticky Forms, File Upload.

2. Assignment on COOKIES and SESSIONS.

3. Assignment on XML documents and DOM

4. Assignment on JavaScript

5. Assignment on Ajax

List of Assignments for Data Analytics

Assignment 1: Frequent itemset and association rule mining
Load Transactional data set. Do the needful data preprocessing. Display the set of frequent 2-itemsets and 3-itemsets. Repeat the process for different min_sup value.

Assignment 2: Linear and Logistic regression
For Given dataset predict the value of specific attribute.

Assignment 3: Text Analytics
Take text file as input. Create bag of words. Find frequent item sets. Display word cloud.

Assignment 4: Sentiment analysis

Course Outcomes:

CO1. Build dynamic website.
CO2. MVC based framework makes it easy to design and handle the errors in dynamic websites.

Subject Code: CS-365

Subject Title: Advanced Java Programming

Teaching Scheme	No. of Credits	Examination Scheme
36 Lecture	2	CE: 15 marks SEE: 35 marks

Course Objectives:

1. To learn database programming using Java.
2. To study web development concept using Servlet and JSP.
3. To develop a game application using multithreading.

Course Contents

UNIT 1 **Collections** **(8L)**

1.1 Introduction to the Collection framework
1.2 List - ArrayList, LinkedList
1.3 Set - Set- HashSet, TreeSet
1.4 Map - HashMap and TreeMap
1.5 Interfaces such as Comparator, Iterator, ListIterator, Enumeration

UNIT 2 **Multithreading** **(7L)**

2.1 What are threads?
2.2 Life cycle of thread Creating threads - Thread class, Runnable interface
2.3 Thread priorities
2.4 Running multiple threads
2.5 Synchronization and interthread communication

UNIT 3 **Database Programming** **(7L)**

3.1 The design of jdbc
3.2 Types of drivers
3.3 Executing sql statements, query execution
3.4 Scrollable and updatable Resultset

UNIT 4 **Servlets and JSP** **(8L)**

4.1 Introduction to Servlet and Hierarchy of Servlet Life cycle of servlet Handing get and post request (HTTP)
4.2 Handling data from HTML to servlet Retrieving data from database to servlet
4.3 Session tracking – User Authorization, URL rewriting, Hidden form fields, Cookies and HttpSession
4.4 Introduction to JSP, Life cycle of JSP
4.5 Implicit Objects Scripting elements - Declarations, Expressions, Scriplets, Comments
4.6 JSP Directives - Page Directive, include directive Mixing Scriplets and HTML JSP Actions - jsp:forward , jsp:include, jsp:useBean, jsp:setProperty and jsp:getProperty

UNIT 5 **Spring Framework** **(6L)**

5.1 Introduction of Spring framework
5.2 Spring Modules / Architecture
5.3 Spring Applications Spring MVC

5.4 Spring MVC Forms, Validation

Course Outcomes:

On completion of the course, students will be able to-

CO1. To access open database through Java programs using Java Data Base Connectivity (JDBC) and develop the application.

CO2. Understand and create dynamic web pages, using Servlets and JSP.

CO3. Work with basics of framework to develop secure web applications.

Learning Resources:

1. Complete reference Java - Herbert Schildt (5th edition).
2. Java 2 programming black books- Steven Horlzner.
3. Programming with Java, A primer - By E. Balagurusamy,,Fourth edition ,
4. Core Java Volume-I-Fundamentals- Cay S. Horstmann, Gary Cornell, Prentice Hall, Sun Microsystems Press, Eighth Edition,
5. Core Java Volume-II-Advanced Features - Cay S. Horstmann, Gary Cornell, Prentice Hall, Sun Microsystems Press, Eighth Edition,
6. Getting started with Spring Framework: covers Spring 5 - J Sharma and Ashish Sarin.
7. Spring 4 for Developing Enterprise Applications: An End-to-End Approach - Henry H. Liu.

Subject Code: CS-366
Subject Title: Compiler Construction

Teaching Scheme	No. of Credits	Examination Scheme
36 Lectures	2	CE: 15 marks SEE: 35 marks

Course Objectives:

1. To understand design issues of a lexical analyzer and use of LEX tool.
2. To understand design issues of a parser and use of YACC tool.
3. To understand and design code generation and optimization techniques.

Course Contents

Unit 1 **Introduction** **(4L)**

1.1 Definition of Compiler, Aspects of compilation
1.2 The structure of Compiler
1.3 Phases of Compiler – Lexical Analysis, Syntax Analysis, Semantic Analysis, Intermediate Code generation, code optimization, Code generation
1.4 Error Handling
1.5 Introduction to one pass & Multi pass compilers
1.6 cross compiler
1.7 Bootstrapping

Unit 2 **Lexical Analysis (Scanner)** **(4L)**

2.1 Review of Finite automata as a lexical analyzer
  2.1.2 Applications of Regular Expressions and Finite Automata (lexical analyzer, searching using RE)
2.2 Input buffering
2.3 Recognition of tokens
2.4 LEX: A Lexical analyzer generator (Simple Lex Program)

Unit 3 **Syntax Analysis (Parser)** **(14L)**

3.1 Definition, Types of Parsers
3.2 Top-Down Parser –
  3.2.1 Top-Down Parsing with Backtracking: Method & Problems
  3.2.2 Drawbacks of Top-Down parsing with backtracking
  3.2.3 Elimination of Left Recursion (direct & indirect)
  3.2.4 Need for Left Factoring & examples
  3.2.5 Recursive Descent Parsing: Definition
  3.2.6 Implementation of Recursive Descent Parser Using Recursive Procedures
  3.2.7 Predictive [LL (1)] Parser (Definition, Model)
  3.2.8 Implementation of Predictive Parser [LL (1)]
  3.2.9 FIRST & FOLLOW
  3.2.10 Construction of LL (1) Parsing Table
  3.2.11 Parsing of a String using LL (1) Table
3.3 Bottom-Up Parser –
  3.3.1 Operator Precedence Parser -Basic Concepts

3.3.1.1 Operator Precedence Relations form Associativity & Precedence
3.3.1.2 Operator Precedence Grammar
3.3.1.3 Algorithm for LEADING & TRAILING (with ex.)
3.3.1.4 Algorithm for Operator Precedence Parsing (with ex.)
3.3.1.5 Precedence Functions
3.3.2 Shift Reduce Parser
3.3.2.1 Reduction, Handle, Handle Pruning
3.3.2.2 Stack Implementation of Shift Reduce Parser (with examples)
3.3.3 LR Parser:
3.3.3.1 Model
3.3.3.2 Types [SLR (1), Canonical LR, LALR]-Method & examples.
3.4 YACC (from Book 3) –program sections, simple YACC program for expression Evaluation

Unit 4 Syntax Directed Definition (7L)

4.1 Syntax Directed Definitions (SDD)
4.1.1 Inherited & Synthesized Attributes
4.1.2 Evaluating an SDD at the nodes of a Parse Tree, Example
4.2 Evaluation Orders for SDD's
4.2.1 Dependency Graph
4.2.2 Ordering the Evaluation of Attributes
4.2.3 S-Attributed Definition
4.2.4 L-Attributed Definition
4.3 Application of SDT
4.3.1 Construction of syntax trees,
4.3.2 The Structure of a Type
4. 4 Translation Schemes
4.4.1 Definition, Postfix Translation Scheme

Unit 5 Code Generation and Optimization (7L)

5.1 Compilation of expression
5.1.1 Concepts of operand descriptors and register descriptors with example.
5.1.2 Intermediate code for expressions – postfix notations,
5.1.3 Triples, Quadruples and Expression trees
5.2 Code Optimization
5.2.1 Optimizing transformations – compile time evaluation, elimination of common sub expressions, dead code elimination, frequency reduction, strength reduction
5.3 Three address code
5.3.1 DAG for Three address code
5.3.2 The Value-number method for constructing DAG's
5.4 Definition of basic block, Basic blocks, and flow graphs
5.5 Directed acyclic graph (DAG) representation of basic block
5.6 Issues in design of code generator

Course Outcomes:

On completion of this course, students will be able to:

1. Understand the process of scanning and parsing of source code.
2. Learn the conversion code written in source language to machine language.
3. Understand tools like LEX and YACC.

Learning Resources:

1. Compilers: Principles, Techniques, and Tools - Alfred V. Aho, Ravi Sethi, Jeffrey D. Ullman, 2004
2. Principles of Compiler Design -Alfred V. Aho, Jeffrey D. Ullman, Narosa Publication House, 2002
3. LEX & YACC, O'reilly Publication, 2012, 2nd edition

Subject Code: CS-369

Subject Title: Practical Course on CS 365 and Project Implementation

Teaching Scheme	No. of Credits	Examination Scheme
4 hrs 20 mins/week	2	CE : 15 marks SEE: 35 marks

Course Objectives:

1. To learn Database programming using Java.
2. To study web development concept using Servlet and JSP.
3. Learn the Development and Deployment of Project.

Guidelines

- Operating Environment :Linux, Java , Pl/Pgsql , Tomcat Server , JDK 1.7 and above.

Course Contents

List of Assignments

Assignment 1: Collections

Program to Use various collections classes and interfaces in the Collections framework, Concept of iterator, Creating and using collections objects.

Assignment 2: Multithreading

Program To create and use threads in java, To demonstrate multithreading and use of Runnable interface, use of Thread methods .

Assignment 3: JDBC

Program To communicate with a database using java,To execute queries on tables,To obtain information about the database and tables,Statement, PreparedStatement, ResultSet, DatabaseMetaData and ResultSetMetaData.

Assignment 4: Servlets and JSP

Program To understand server-side programming ,Defining and executing servlets Handling Get and Post requests (HTTP),Data Handling using Servlet,Creating Cookies,Session Tracking using HTTP Servlet, JSP Directives, Scriptingelements, Actions in JSP.

Project Implementation

1. Project Documentation and Project Report generation

Course Outcomes:

CO1. To Build dynamic website using Java.

CO2. Learning of framework to develop secure web applications.

Subject Code:

Subject Title: Mobile application development using Android (Elective)

Teaching Scheme	No. of Credits	Examination Scheme
36 Lectures	2	CE : 15 marks SEE: 35 marks

Course Objectives:

1.This course covers the basics of Android along with database connectivity required programming codes for developing necessary programming skills for mobile applications.
2.The course content should be implemented with the aim to develop required skills in the students so that they are able to develop GUI base Mobile applications.

Course Contents

Unit 1 **Introduction to Android** **(6L)**

1.1 Introduction to Android - Overview and evolution of Android, Open Handset Alliance, What does Android run On – Android Internals?, Use Android for mobile app development.
1.2 Android Development Environment setup, Framework - Android-SDK, Eclipse, JD
1.3 Emulators – What is an Emulator/Android AVD?, First Android Application (hello World program).

Unit 2 **Android: Architecture and Framework** **(7L)**

2.1 Components of an Android Application
2.2 Linux Kernel, Libraries, Android Runtime, Application Framework
2.3 Applications, Android Startup and Zygote, Android Debug bridge, Android Permission model, Android Manifest File
2.4 Android Activity
2.5 Service Lifecycle

Unit 3 **Android: Activity and GUI Design Concepts** **(8L)**

3.1 Android application components Intent , Activity, Fragments, Fragments,Views
3.2 Broadcast receivers, Create Application and new Activities, Design Demands for Android Application
3.3 Expressions and Flow control, Simple UI -Layouts and Layout properties, Introducing Layouts, Creating new Layouts, Drawable Resources
3.4 XML Introduction to GUI objects viz.: Edit Text, Text View, Button, Toggle Button , Padding etc.

Unit 4 **Advanced Android GUI Programming** **(8L)**

4.1 Event driven Programming in Android (Text Edit, Button clicked etc.)
4.2 Creating a splash screen, Threads in Android, Menu: Custom Vs. System Menus
4.3 Creating and Using Handset menu Button (Hardware), Android Themes, Dialog, create an Alter Dialog, Toast in Android.
4.4 List & Adapters, Android Manifest.xml File
4.5 Accessing Phone Service(Call, SMS, MMS)
4.6 Location based services

Unit 5 Work with SQLite Database (7L)

5.1 Basic operation of SQLite Database, Android application Priorities
5.2 SQLite: Open Helper and create database, Open and close a database.
5.3 CRUD Operation
5.4 JSON Parsing

Course Outcomes:

On completion of the course, students will be able to–
CO1. Identify and understand the concepts of open-source mobile technology.
CO2. Understand Android architecture frame work.
CO3. Understand GUI Design concepts and design Android GUI Layout.
CO4. Develop and design event driven programming with menus and dialog boxes.
CO5. Design and develop applications with database

Learning Resources:

1. Android Programming: The Big Nerd Ranch Guide (Big Nerd Ranch Guides), 2013 - Bill Phillips and Brian Hardy.
2. Professional Android 4 Application Development, 2012 - Reto Meier.
3. Android Application Development in 24 Hours, Sams Teach Yourself 2015 - Carmen Delessio and Lauren Darcey 4th edition.
4. Beginning Android Application Development - Wei-Meng Lee Wiley, Wrox publication, Second edition.
5. Android cookbook - Ian F Darwin, O'Reilly publication, Second Edition.

Subject Code:
Subject Title: Blockchain technologies (Elective)

Teaching Scheme	No. of Credits	Examination Scheme
36 Lectures	2	CE : 15 marks SEE: 35 marks

Course Objectives:

1. Understand what and why of blockchain technology.
2. Explore major components of blockchain.
3. Learn about Bitcoin, Crypto currency and Ethereum.
4. To learn blockchain programming using Python, Flask Web Framework, and HTTP client Postman.

Course Contents

Unit 1 **Introduction to Blockchain** **(7L)**

1.1 Foundational Computing Concepts (Client-Server systems vs Peer to Peer Systems)
1.2 Evolution of Blockchain
1.3 Blockchain Vs Database
1.4 Essentials of Blockchain (Blockchain generations, types of blockchain, benefits and
1.5 Challenges of blockchain usage)
1.6 Types of Networks
1.7 Layered Architecture of Blockchain Ecosystem
1.8 Components of blockchain
1.9 Cryptography (private and public keys, Hashing & Digital Signature)
1.10 Consensus Mechanisms
1.11 Cryptocurrency, Digital Currency Bitcoin and Ethereum
1.12 Smart Contracts
1.13 Blockchain use cases

Unit 2 **How Blockchain Works?** **(5L)**

2.1 Understanding SHA256 Hash
2.2 Immutable Ledger
2.3 Distributed P2P Network
2.4 How Mining Works? (The NONCE and Cryptographic Puzzle)
2.5 Byzantine Fault Tolerance
2.6 Consensus Protocols: Proof of Work, Proof of State, Défense Against Attackers, Competing Chains
2.7 Blockchain Demo

Unit 3 **Smart Contracts** **(6L)**

3.1 Ethereum Network
3.2 What is a Smart Contract?
3.3 Ethereum Virtual Machine, Ether, Gas
3.4 DApps
3.5 Decentralized Autonomous Organizations (DAO)

3.6 Hard and Soft Forks
3.7 Initial Coin Offerings
3.8 Demo of Smart Contracts

Demonstration Programming Assignments (18L)

Out of 36 lectures, 18 are assigned for demonstration. Teachers should give demonstrations of various programs mentioned below in the classroom or in the laboratory as per their convenience.

Assignment 1 –Demonstration of Blockchain
https://andersbrownworth.com/blockchain

Assignment 2 – Installation of Ganache, Flask and Postman

Assignment 3 –Write a Simple Python program to create a Block class that contains index, timestamp, and previous hash. Connect the blocks to create a Blockchain.

Assignment 4 –Demo of Remix-Ethereum IDE https://remix.ethereum.org and Test Networks

Assignment 5 –Write a Simple Smart Contract for the Bank with withdrawal and deposit functionality.

Assignment 6 – Write a Smart Contract for storing and retrieving information of Degree Certificates.

Course Outcomes:

On completion of the course, students will be able to–
CO1. Learn the fundamentals of Blockchain Technology.
CO2. Learn Blockchain programming
CO3. Basic knowledge of Smart Contracts and how they function.

Learning Resources:

Textbook:

1. Beginning Blockchain : A Beginner's Guide to Building Blockchain Solutions By Bikramaditya Singhal, Gautam Dhameja, Priyanshu Sekhar Panda, Apress Media , 2018.

Reference Books:

1. Mastering Blockchain by Imran Bashir, Third Edition, Packt Publication , 2020.
2. The Science of the Blockchain by Roger Wattenhofer , Createspace Independent Pub, 2016.
3. Mastering Ethereum:Building Smart Contracts and DAPPS, by Andreas Antonopoulos, Dr. Gavin Wood, O'reilly Publication, 1st Edition, 2018.

Article:

1. Bitcoin: A Peer-to-Peer Electronic Cash System by Satoshi Nakamoto , 2008.

Reference Web Links

1. https://www.investopedia.com/terms/b/blockchain.asp

Subject Code:
Subject Title: Cloud and Edge Computing(Elective)

Teaching Scheme	No. of Credits	Examination Scheme
36 Lectures	2	CE: 15 marks SEE: 35 marks

Course Objectives:

1. To introduce concepts of Cloud and Edge Computing.
2. To study Cloud and Edge Computing technologies, architecture and applications.
3. To study key concepts of virtualization.
4. To Study the different types of Open Source and Commercial Clouds.

Course Contents

Unit 1 Fundamentals of Cloud Computing (6L)

1.1 Overview of cloud computing
Origin of Cloud Computing, challenges, benefits, limitations, Cloud Storage.
1.2 Types of cloud Computing- Private cloud, Public cloud, Hybrid cloud and multi cloud
1.3 Open Source and Commercial Clouds
1.4 Applications of Cloud Computing
1.5 Cloud Computing Architecture

Unit 2 Virtualization (10L)

2.1 What is virtualization
2.2 Needs and Benefits of virtualization, imitations
2.3 Types of virtualization
Hardware Virtualization: Full virtualization - partial virtualization - para virtualization
Software virtualization – Memory virtualization - Storage virtualization
Data virtualization - Network Virtualization
2.4 VM Resource Allocation, Management and Monitoring
2.5 Virtualization technologies

Unit 3 Edge Computing (8L)

3.1 Introduction to Edge computing
3.2 Edge Computing VS Cloud Computing
3.3 Architecture and Components of Edge computing
Edge devices and sensors, edge computing infrastructure.
3.4 Networking in Edge computing- communication protocol, edge-to- cloud communication, edge-to-edge communication
3.5 Security and Privacy in Edge Computing-Threats and vulnerabilities, security protocols, Privacy considerations
3.6 Advantages and Disadvantages of Edge computing

Unit 4 Security (6L)

4.1 Overview of Cloud Security Cloud Security Threads
4.2 Cloud Security Challenges and Risks
4.3 Edge Data Security
4.4 Data confidentiality and types of encryption and Authentication
4.5 Edge based attack detection and prevention

Unit 5 Emerging Trends (6L)

5.1 Multi-Cloud Vs Omni-Cloud
5.2 Integrated Block chain technology
5.3 Kubernetes
5.4 Cloud AI
5.5 Intelligent SaaS
5.6 Kubernetes Supremacy
5.7 Containerization by Industry Giants

Course Outcomes:

On completion of this course, students will be able to:

CO1. To understand the principles and paradigm of Cloud Computing and Edge Computing.
CO2. To understand the major components of Edge and Cloud computing architectures.
CO3. To understand the concept of virtualization.
CO4. To understand an emerging trend of Open Source and Commercial Clouds.

Learning Resources:

1. Fundamentals of Cloud Computing - Prasanta Kumar, Manas Ranjan Kabat, Souvik Pal, Vikas publishing House 2014,1st Edition
2. Edge Computing Fundamentals, Advances and Applications - M. Kumari, K. Anitha , Sadasivam, G. Sudha ,Dharani, D. , Niranjanamurthy (Author, Contributor)**,**Taylor & Francis Ltd; 1st edition (23 December 2021)
3. Edge Cloud Operations - Bruce Davie, Larry L Peterson, Scott Baker A Systems Approach, System Approach (16th June 2022)