



**Maharashtra Education Society's
ABASAHEB GARWARE COLLEGE
(Autonomous)**

(Affiliated to Savitribai Phule Pune University)

**Two Years Post-Graduation Programme in Computer Application
(Faculty of Science & Technology)**

Syllabi under Autonomy

M.Sc. I (Computer Application)

Choice Based Credit System Syllabus

To be implemented from Academic Year 2022-2023

Title of the Course: M.Sc. (Computer Application)

Preamble

The M.Sc. (Computer Applications) program is aimed to impart Computer knowledge and skill sets in non-computer background students and enhance the skill sets of students having computer background.

The program has the advantage that students will have the domain knowledge of the subject in which they have done their graduation and they will learn the computer skills in this course and so, will be able to apply these computer skills in their respective domains.

It is a combination of basic as well as advanced computer skill sets. The course introduces different programming languages, basic and advanced database concepts, web designing, system analysis and design tools and Software Testing techniques. Also, it provides the students to select their path of career by providing the elective courses in each semester. The full-time internship in the last semester, makes it mandatory for each student to work in IT industry to complete his/her post-graduation course. This gives a student the exposure to the environment in IT industry, make students familiar for working in team and give a chance to students to learn recent technologies used in IT industries.

Learning Outcomes:

The main objective of the Program is to produce trained software professionals with hands-on experience on state-of-the art technologies who will be able to handle software challenges in industry as well as academia. In the context with information Technology industry, the objectives of M.Sc. (Computer Applications) course are: -

- To produce knowledgeable and skilled human resources that is employable in IT and ITES.
- To impart knowledge required for planning, designing and building Complex Application Software
- Systems as well as to provide support for automated systems or applications.
- To produce entrepreneurs

Eligibility:

A Bachelor degree in Science / Technology / Engineering with minimum 50% marks or equivalent for students belonging to open category and minimum 45% or equivalent for students belonging to reserved category

Structure of the Course: M. Sc. (Computer Application)

Year	Semester	Course Type	Course Code	Course Title	Remark	Credit	No. of Lectures /Practical to be conducted	
1	I	CC	PSCA-111	Web Technology		4	48	
		CC	PSCA-112	Advanced Databases		4	48	
		CC	PSCA-113	Data Structures and Algorithms		4	48	
		CBO	PSCAELE-114A	Object Oriented Programming - I		2	30	
		CBO	PSCAELEP-114A	Object Oriented Programming – I Laboratory		2	12	
		OR						
		CBO	PSCAELE-114B	Python Programming		2	30	
		CBO	PSCAELEP-114B	Python Programming Laboratory		2	12	
		OR						
		CBO	PSCAELE-114C	Computer Networks		2	30	
		CBO	PSCAELEP-114C	Computer Networks Laboratory		2	12	
		CC	PSCAP-115	Web Technology, Data Structures and Algorithms Laboratory		4	24	
		ACC	PSCYS1-11	Cyber Security – I		1	-	
		ACC	PSHR1-11	Human Rights – I		1	-	

Year	Semester	Course Type	Course Code	Course Title	Remark	Credit	No. of Lectures /Practical to be conducted	
1	II	CC	PSCA-121	Full Stack Development - I		4	48	
		CC	PSCA-122	Operating System		4	48	
		CC	PSCA-123	Machine Learning		4	48	
		CBO	PSCAELE-124A	Java Programming		2	30	
		CBO	PSCAELEP-124A	Java Programming Laboratory		2	12	
		OR						
		CBO	PSCAELE-124B	Project Design		2	30	
		CBO	PSCAELEP-124B	Project Implementation and Presentation		2	12	
		OR						
		CBO	PSCAELE-124C	Cryptography and Network Security		2	30	
		CBO	PSCAELEP-124C	Cryptography and Network Security Laboratory		2	12	
		CC	PSCAP-125	Full Stack Development - I and Operating System Laboratory		4	24	
		ACC	PSCYS2-12	Cyber Security - II		1	-	
		ACC	PSHR2-12	Human Rights - II		1	-	

Year	Semester	Course Type	Course Code	Course Title	Remark	Credit	No. of Lectures /Practical to be conducted		
2 2	III	CC	PSCA-231	Full Stack Development - II		4	48		
		CC	PSCA-232	Big Data		4	48		
		CC	PSCA-233	Software Testing		4	48		
		CBO	PSCAELE-234A	Web Services		2	30		
		CBO	PSCAELEP-234A	Web Services Laboratory		2	12		
		OR							
		CBO	PSCAELE-234B	Project Design		2	30		
		CBO	PSCAELEP-234B	Project Implementation and Presentation		2	12		
		OR							
		CBO	PSCAELE-234C	Django		2	30		
		CBO	PSCAELEP-234C	Django Laboratory		2	12		
		CC	PSCAP-125	Full Stack Development - II and Software Testing Laboratory		4	24		
		ACC	PSCYS3-23	Cyber Security - III		1	-		
		ACC	PSSD1-23	Skill Development - I		1	-		
		ACC	PSIC-23	Introduction to Constitution		2	-		

Year	Semester	Course Type	Course Code	Course Title	Remark	Credit	No. of Lectures /Practical to be conducted
2	IV	CC	PSCAP-241	Industrial Training / Industrial Project		4	02/ week
		ACC	PSCYS4-24	Cyber Security – IV		1	-
		ACC	PSSD2-24	Skill Development – I		1	-

FIRST YEAR / SEMESTER-I**Course Code and Title: PSCA-111 Web Technology****Lectures: 48 (Credits- 04)****Prerequisite:**

- Fundamentals of Computers

Course Objectives:

1. To learn HTML, CSS, Java Scripts basics of Web designing.
2. To learn basics of PHP.
3. To understand use of framework for web-based applications development.

Learning Outcomes:

1. Design a website using HTML and CSS.
2. Design web application using PHP framework.

Unit 1: Introduction to HTML and CSS**08**

1.1 Introduction to Internet-Basic

1.1.1 Evolution of Web

1.1.2 Introduction to Web 3.0

1.2 Internet Protocols (WWW, HTTP, FTP, IP)

1.3 HTML tags like Image, List, Tables, Frames and frame and attributes

1.4 HTML form designing and HTML 5

1.4.1 Designing of Forms using text box, text area, buttons, List box, radio, checkbox

1.4.2 GET and POST methods

1.5 CSS

1.5.1 Introduction to Style Sheet

1.5.2 Ways to apply CSS to HTML

1.5.3 CSS Border, margin, Positioning, color, text, link, background, list, table, padding, image, display properties, z-index, opacity

1.5.4 Use of Id and classes in CSS, <div> and

1.5.5 Introduction of CSS3: Gradients, Transitions, Animations, multiple columns

Unit 2: JavaScript**08**

2.1 Introduction to Javascript

2.2 Variables, identifiers and operators, control structures

2.3 Functions

2.4 Event Handling in Javascript

2.5 Concept of array, Types of an array

2.6 Math and date object, String object and predefined String functions

2.7 DOM concept in Javascript, DOM objects

2.8 Validations in Javascript

Unit 3: PHP - Functions, Strings, Arrays and OOP	08
3.1 Introduction to PHP Variables, Data Types, Strings, Operators, Loops	
3.2 Functions – Definition and Call, Default parameters, Variable parameters, Missing parameters, Variable function, Anonymous function	
3.3 String- Type of String, printing functions, comparing strings, Manipulating and searching strings Regular expressions	
3.4 Array- Types, storing data in arrays, Multidimensional arrays, extracting multiple values, Converting between arrays and variables, Traversing arrays, Sorting, Action on entire array	
3.5 OOP-Classes, Objects, Introspection, Inheritance, Interfaces	
Unit 4: Introduction to Laravel	03
4.1 Overview and Installation of Laravel	
4.2 Laravel Features	
4.3 Application Structure of Laravel	
4.3.1 Root Directory	
4.3.2 App Directory	
4.4 Configuration	
4.4.1 Basic Configuration	
4.4.2 Environmental Configuration	
4.4.3 Database Configuration	
4.4.4 Naming the Application	
4.4.5 Maintenance Mode	
Unit 5: Routing and controllers	06
5.1 Routing	
5.1.1 Basic Routing	
5.1.2 Routing Parameters	
5.2 Middleware	
5.2.1 Define Middleware	
5.2.2 Register Middleware	
5.2.3 Middleware Parameters	
5.2.4 Terminable Middleware	
5.3 Controller	
5.3.1 Basic Controllers	
5.3.2 Controller Middleware	
5.3.3 Restful Resource Controllers	
5.3.4 Implicit Controllers	
5.3.5 Constructor Injection	
5.3.6 Method Injection	
Unit 6: Request and Responses	06

6.1 Request

6.1.1 Retrieving the Request URI

6.1.2 Retrieving Input

6.2 Cookie

6.2.1 Creating Cookie

6.2.2 Retrieving Cookie

6.3 Response

6.3.1 Basic Response

6.3.2 Attaching Headers

6.3.3 Attaching Cookies

6.3.4 JSON Response Object-Oriented Programming

Unit 7: Views**06**

7.1 Understanding Views

7.2 Passing Data to Views

7.3 Sharing Data with all Views

7.4 Blade Templates

7.5 Redirecting to Named Routes

7.6 Redirecting to Controller Actions

Unit 8: Database and Session**04****8.1 Working with Database**

8.1.1 Connecting to Database

8.1.2 Insert Records

8.1.3 Retrieve Records

8.1.4 Update Records

8.1.5 Delete Records

8.2 Forms**8.3 Session**

8.3.1 Accessing Session Data

8.3.2 Storing Session Data

8.3.3 Deleting Session Data

Reference Books

1. HTML Black Book, Steven Holzner, Steven Holzner
2. HTML & CSS: The Complete Reference, Fifth Edition, Thomas A. Powell, Mc Graw Hill
3. Web Technologies, Black Book, Kogent Learning Solutions Inc., Dreamtech Press
4. Programming PHP, Rasmus Lerdorf and Kevin Tatroe, O'Reilly publication
5. Laravel: Up & Running A Framework for Building Modern PHP Apps, Matt Stauffer, O'Reilly publication

Course Code and Title: PSCA-112 Advanced Databases**Lectures: 48 (Credits- 04)****Prerequisite:**

- Basic Knowledge of Files
- Basics of Set Theory

Course Objectives:

1. To understand the fundamental concepts of relational database.
2. To understand the need and fundamental concepts of distributed databases.
3. To understand various issues in distributed databases and how distributed database management system handles them.

Learning Outcomes:

After completion of this course students will be able to -

1. understand advantages of distributed databases over centralized database management system.
2. understand how to design distributed database.
3. understand the complexities in managing transactions in distributed databases and solutions for the problems encountered in transaction management.

Unit 1: Overview of Relational DBMS	08
1.1 Concept of relational database	
1.2 Normalization	
1.3 Integrity Rules	
1.4 Relational data languages	
1.4.1 Relational Algebra	
1.4.2 Relational Calculus	
1.5 Relational DBMS	
1.6 Concept of Centralized database	
Unit 2: Introduction to Distributed database	02
2.1 What is distributed database system	
2.2 Distributed Vs Centralized database	
2.3 Promises of Distributed database	
2.4 Complicating factors	
2.5 Problem areas	
Unit 3: Distributed DBMS architecture	04
3.1 DBMS standardization	
3.2 Architectural models for Distributed DBMS	
3.3 Different architectures	
3.3.1 Client-server architecture	

3.3.2 Peer-to-Peer architecture

3.3.3 MDBS architecture

Unit 4: Distributed Database Design **10**

- 4.1 Alternative design strategies
- 4.2 Design issues in distribution
- 4.3 Fragmentation
 - 4.3.1 Horizontal fragmentation
 - 4.3.2 Vertical fragmentation
 - 4.3.3 Hybrid fragmentation
- 4.4 Allocation problem
 - 4.4.1 Information required for allocation
 - 4.4.2 Allocation model
 - 4.4.3 Solution methods

Unit 5: Overview of query processing **06**

- 5.1 Query processing problem
- 5.2 Objectives of query processing
- 5.3 Complexities of relational algebra operations
- 5.4 Characterization of query processing
- 5.5 Layers of query processing
- 5.6 Brief introduction of each layer of query processing

Unit 6: Introduction to transaction management **02**

- 6.1 Definition of transaction
- 6.2 Properties of transaction
- 6.3 Types of transactions

Unit 7: Distributed concurrency control **08**

- 7.1 Serializability theory
- 7.2 Taxonomy of concurrency control algorithms
- 7.3 Lock-based concurrency control algorithms
- 7.4 Timestamp-based concurrency control algorithms
- 7.5 Optimistic concurrency control algorithm
- 7.6 Deadlock management

Unit 8: Distributed DBMS reliability **08**

- 8.1 Reliability concept and measures
- 8.2 Failures and fault tolerance in distributed systems
- 8.3 Failures in distributed DBMS
- 8.4 Local reliability protocols
- 8.5 Distributed reliability protocols
- 8.6 Dealing with site failures
- 8.7 Dealing with network partitioning

Reference Books

1. Principles of Distributed Database Systems, M. Tamer Ozsü, Patrick, Valduriez, Pearson Education Asia.
2. Distributed Database; Principles & Systems, Stefano Ceri, Giuseppe, Pelagatti, McGraw-Hill International Editions.

Course Code and Title: PSCA-113 Data Structures and Algorithms

Lectures: 48 (Credits- 04)

Prerequisite:

- Sound knowledge of algorithms and any programming language

Course Objectives:

1. To understand the fundamental concepts of relational database.
2. To understand the need and fundamental concepts of distributed databases.
3. To understand various issues in distributed databases and how distributed database management system handles them.

Learning Outcomes:

After completion of this course students will be able to -

1. choose appropriate data structures depending upon the nature of the problem-in-hand.
2. apply suitable algorithm design strategy to create efficient solutions for solving complex problems.
3. find optimal solutions for solving hard problems.

Unit 1: Basics of Algorithms 08

- 1.1 Algorithm Definition and desirable properties
- 1.2 Characterizing Run Time, Use of asymptotic notation, Big-Oh Notation, Little-Oh, Big Omega, Little-Omega, and Theta Notations
- 1.3 Analyzing Iterative and Recursive Algorithms, Recurrence relations, Specifying runtime of recursive algorithms

Unit 2: Linear Data Structures 08

- 2.1 Stacks: Concept and applications
- 2.2 Queues: Concept and applications, Types of Queues
- 2.3 Linked Lists: Concept and applications

Unit 3: Non-Linear Data Structures 04

- 3.1 Trees, binary trees, binary search trees and their traversal techniques
- 3.2 Heaps, heap sort
- 3.3 Types of Graphs and its traversals

Unit 4: Greedy Method 06

- 4.1 Control Abstraction
- 4.2 Fractional Knapsack Problem
- 4.3 Task Scheduling Problem
- 4.4 Optimal merge patterns

- 4.5 Minimum Spanning Tree: Kruskal's and Prim's algorithm
- 4.6 Shortest Path Problem - Dijkstra's Algorithm

Unit 5: Divide and Conquer**06**

- 5.1 Control abstraction
- 5.2 Integer Multiplication Problem (Booth's Algorithm)
- 5.3 Sorting Problem (Merge Sort, Quick Sort)
- 5.4 Searching Problem and Binary Search Algorithm

Unit 6: Dynamic Programming**16**

- 6.1 Principle of optimality
- 6.2 Matrix Chain Product Problem
- 6.3 0/1 Knapsack Problem (function method and merge-and-purge)
- 6.4 All-pairs Shortest Path Problem (Floyd-Warshall's algorithm)
- 6.5 Bellman-Ford Algorithm
- 6.6 Longest Common Subsequence
- 6.7 String Editing
- 6.8 Traveling Salesperson Problem

Reference Books:

1. Fundamentals of Data Structures, Horowitz and Sahani, Galgotia Publications.
2. Fundamentals of Computer Algorithms, Horowitz and Sahani, Galgotia Publications.
3. Introduction to Algorithms, TH Cormen, CE Leiserson, RL Rivest, C Stein, PHI.

**Course Code and Title: PSCAELE-114A Object Oriented Programming
with C++**

Lectures: 30 (Credits- 02)

Prerequisite:

- No prior knowledge is required.

Course Objectives:

1. To learn the fundamental programming concepts and methodologies which are essential to building good C++ programs.
2. To introduce object-oriented programming concepts.
3. To understand early and late binding in C++.

Learning Outcomes:

After completion of this course students will be able to -

1. understand the difference between the top-down and bottom-up approach.
2. demonstrate the object-oriented programming approach in connection with C++.
3. apply virtual and pure virtual function & complex programming situations.

Unit 1: Introduction to C++

04

- 1.1 Basics of C++
- 1.2 Structure of C++ Program, keywords in C++,
- 1.3 Data types hierarchy in C++,
- 1.4 Operators in C++: Scope resolution operator, Insertion and Extraction Operator New and Delete operators
- 1.5 Reference variable
- 1.6 Manipulators function: endl, setw, set fill, set precision

Unit 2: Introduction to C++

06

- 2.1 Basics of C++
- 2.2 Structure of C++ Program, keywords in C++
- 2.3 Data types hierarchy in C++
- 2.4 Operators in C++: Scope resolution operator, Insertion and Extraction Operator New and Delete operators
- 2.5 Reference variable
- 2.6 Manipulators function: endl, setw, set fill, set precision

Unit 3: Function in C++

04

- 3.1 Call by reference, Return by reference
- 3.2 Function overloading and default arguments
- 3.3 Inline function

3.4 Passing and returning objects from functions	
3.5 Static class members	
3.6 Friend Concept – Function, Class	
Unit 4: Constructors and Destructors	02
4.1 Constructor	
4.2 Types of constructors: Default, parameterized, copy	
4.3 Destructors	
Unit 5: Operator overloading	03
5.1 Introduction, rules of operator overloading	
5.2 Operator overloading	
5.3 Unary and binary operator	
5.4 Comparison, arithmetic assignment operator	
Unit 6: Inheritance	04
6.1 Introduction	
6.2 Types of Inheritance: Single inheritance, Multiple inheritance, Multilevel inheritance Hierarchical inheritance Hybrid inheritance.	
6.3 Derived Class Constructor and Destructors	
6.4 Ambiguity in multiple Inheritances, virtual base classes, Abstract base class	
Unit 7: Virtual Function & Polymorphism	03
7.1 Introduction, Pointer to object, Pointer to derived Class	
7.2 Overriding member functions, Virtual function	
7.3 Rules for virtual functions, pure virtual function	
Unit 8: Working with files	03
8.2 File operations – Text files, Binary files	
8.2 File stream class and methods	
8.3 File Updation with random access	
Unit 9: Introduction to Exception Handling	01

Reference Books:

1. Object- Oriented Programming with C++, E. Balaguruswamy
2. C/C++ Programmer's Reference, Herbert Schildt

**Course Code and Title: PSCAELEP-114A Object Oriented Programming
with C++ Laboratory**

Practical: 12 (Credits- 02)

List of Assignments

1. Basic programs on arithmetic operations
2. Programs on functions, constructors and destructors
3. Programs on operator overloading
4. Programs on inheritance
5. Programs on inheritance
6. Programs on virtual functions and polymorphism
7. Program on file handling
8. Program on exception handling

Course Code and Title: PSCAELE-114B Python Programming

Lectures: 30 (Credits- 02)

Prerequisite:

- No prior programming language knowledge is required

Course Objectives:

1. To understand basics of programming and learn the python language.
2. To explore Python's object-oriented features.
3. To learn different modules and packages.

Learning Outcomes:

After completion of this course students will be able to -

1. solve basic problems by writing programs.
2. apply object-oriented concepts.
3. use modules and packages for project work.

Unit 1: Introduction

04

- 1.1. The Programming Cycle for Python
- 1.2. Python IDE, Interacting with Python Programs
- 1.3. Variables, Operators, Expressions
- 1.4. Primitive Data Types, Type Conversion

Unit 2: Strings, Python Data Structure

06

- 2.1 Defining strings
- 2.2 Operations on Strings
- 2.3 Concept of Indexing and Slicing of Strings.
- 2.4 Tuples, Unpacking Sequences, Lists
- 2.5 Mutable Sequences, List Comprehension
- 2.6 Sets, Dictionaries

Unit 3: Conditionals, Loops and Functions

06

- 3.1 Conditional statement in Python
- 3.2 if-else statement, its working and execution
- 3.3 Nested-if statement and Elif statement, pass
- 3.4 Purpose and working of loops
- 3.5 while loop including its working, For Loop
- 3.6 Nested Loops
- 3.7 Break and Continue
- 3.8 Concept, Arguments
- 3.9 Scope Rules
- 3.10 User-defined functions

3.11 Recursion

Unit 4: Modules & Packages **04**

- 4.1 Introduction to Data Science
- 4.2 Introduction to Modules and Packages
- 4.3 NumPy, SciPy, Pandas
- 4.4 Matplotlib
- 4.5 TensorFlow

Unit 5: Object Oriented Programming **06**

- 5.1 Concept of object-oriented programming
- 5.2 Defining Classes (Attributes Methods)
- 5.3 Creating Instance Objects
- 5.4 Accessing Attributes and Methods
- 5.5 Constructor
- 5.6 Concept of "Self"

Unit 6: Inheritance **04**

- 6.1 Concept of Base Class and Derived Class
- 6.2 Single Multilevel and multiple Inheritance
- 6.3 Overriding Methods Using Super() in Derived Class to Invoke Init()
- 6.4 Overridden Methods of Parent Class.

Reference Books:

1. Python Cookbook, David Beazley & Brian K. Jones, O'REILLY
2. An Introduction to Python – Revised and updated for Python 3.2, Guido van Rossum and Fred L. Drake Jr. , Network Theory Ltd., 2011.
3. Introduction to Programming Using Python, Y. Daniel Liang, Pearson

**Course Code and Title: PSCAELEP-114B Python Programming
Laboratory**

Practical: 12 (Credits- 02)

List of Assignments

1. Programs on data structures in Python
2. Basic programs on conditional statements, looping statements
3. Basic Programs on modules and packages
4. Advanced Programs on modules and packages
5. Basic Programs based on concept of OOP
6. Advanced Programs based on concept of OOP

Course Code and Title: PSCAELE-114C Computer Networks**Lectures: 30 (Credits- 02)****Prerequisite:**

- Basics of Digital Electronics
- Concept of Algorithms

Course Objectives:

1. To become familiar with layered communication architectures (OSI and TCP/IP).
2. To get understanding of the concepts of reliable data transfer and how TCP implements these concepts.
3. To learn basics required for Cryptography and Network Security

Learning Outcomes:

After completion of this course students will be able to -

1. have a good understanding of the OSI and TCP/IP Reference Models
2. understand working of various protocols
3. learn Cryptography and Network Security Concepts easily.

Unit 1: Data Communication**04**

- 1.1 Characteristics of data communication
- 1.2 Components, data representation, data flow
- 1.3 Computer Networks: Distributed processing, Physical structure Point to Point, Broadcast
- 1.4 Categories of topology (mesh, star, ring, bus etc.)
- 1.5 LAN, MAN and WAN
- 1.6 Protocols and Standards
- 1.7 Introduction of Reference Models

Unit 2: The Physical Layer**04**

- 2.1 Bit rate, bit length
- 2.2 Base band Transmission Impairments – attenuation, distortion
- 2.3 Noise Data Rate Limits – Nyquist's bit rate formula for noiseless channel and Shannon's law Performance of the Network: Bandwidth, Throughput, Latency (Delay), Bandwidth –Delay Product, Jitter
- 2.4 Line Coding digital to digital conversion: Characteristics, Line Coding Schemes Unipolar, NRZ, RZ, Manchester and Differential Manchester

Unit 3: The Data Link Layer	04
3.1 Framing: Character Count, Byte Stuffing, Bit Stuffing	
3.2 Physical Layer Coding Violations	
3.3 Error Control: Hamming Code and CRC	
Unit 4: The Network Layer	06
4.1 Design Issues	
4.2 Logical Addressing	
4.3 Classful Addressing, Classless Addressing, Network Address Translation (NAT)	
4.4 IPV6 Addresses – Addressing Structure, Address Space	
4.5 Transition From IPV4 to IPV6: Dual Stack, Tunnelling, Header Translation	
4.6 Routing Concepts: Properties of routing algorithm, Comparison of Adaptive and Non-Adaptive Routing Algorithms	
Unit 5: The Transport Layer	04
5.1 Process-to-Process Delivery: Client Server Paradigm	
5.2 Connectionless Vs Connection-Oriented Service	
5.3 Reliable Vs Unreliable	
5.4 User Datagram Protocol	
5.5 Transmission Control Protocol	
Unit 6: The Application Layer	04
6.1 Domain Name System, TELNET: Timesharing Environment	
6.2 Logging, NVT, Embedding, Options	
6.3 HTTP: HTTP Transaction	
6.4 Persistent and Non-persistent Connection	
6.5 Proxy Server	
Unit 6: Firewalls	04
7.1 Concept and Need of Firewalls	
7.2 Characteristics of Firewall	
7.3 Types of Firewalls	
7.4 Firewall Basing	
7.5 Location and Configuration	

Reference Books:

1. Computer Networks, Tanenbaum, Pearson
2. Data Communication and Networking, Behrouz Forouzan, TATA McGraw Hill

Course Code and Title: PSCAELEP-114C Computer Networks Laboratory

Practical: 12 (Credits- 02)

Assignments

1. Networking commands in LINUX.
2. Study of LAN Environment.
3. Introduction and basics of any network analysis tool.
4. Few activity-based assignments on firewall tool.
5. Study of Packet Tracer Tool
6. Case Study (can be done in a group of 2-3 students).

Course Code and Title: PSCAP-115 Web Technology and DSA Laboratory

Practical: 24 (Credits- 04)

List of Web Technology Assignments

Assignment 1

1. Basic HTML Tags - headings, paragraphs, line break, colors, fonts, links, Image
2. Creating forms by using HTML and HTML5 Tags
3. Styling HTML pages using CSS

Assignment 2

1. JavaScript basic concepts and functions
2. Field and Form validation using JavaScript

Assignment 3 PHP Function, Strings

1. User defined functions in PHP
2. String operations and use of built-in functions

Assignment 4 Arrays and OOP

1. Array manipulation and use of different built-in function
2. PHP script with Class, object and inheritance

Assignment 5

Basic Laravel

Assignment 6

Laravel Assignments

List of Data Structures and Algorithms Assignments

Student can do assignment in any programming language such as C/Java/ Python.

Assignment 1

Linked list

Assignment 2

Stack and Queue

Assignment 3

Tree and Graphs

Assignment 4

Algorithms implementation such as BST, Spanning Trees algorithms

Assignment 5

Implementation of Shortest path algorithms

FIRST YEAR / SEMESTER-II**Course Code and Title: PSCA-121 Full Stack Development - I****Lectures: 48 (Credits- 04)****Prerequisites:**

- Basics of Scripting Language
- Concept of client and server
- Concept of Web Server

Course Objectives:

1. To understand and learn modern web technologies.
2. To understand server-side programming.
3. To learn developing basic interactive web applications.

Learning Outcomes:

On the completion of this course students will be -

1. Able to write programs that are server-side scripting.
2. Familiar with latest web development tools and techniques in IT industry.
3. Ready for advanced level of Full Stack Development Course.

Unit 1: Introduction**04**

- 1.1 How Web works?
- 1.2 What is Full Stack Development?
- 1.3 What is ECMA Script?
- 1.4 Concept of Client-Side Scripting
- 1.5 Concept of Server-Side Scripting
- 1.6 Concept of Framework

Unit 2: Basics of JavaScript**10**

- 2.1 JavaScript Variables and Operators
- 2.2 JavaScript Arrays and Functions
- 2.3 Importance and need of JavaScript Objects
- 2.4 JavaScript Events
- 2.5 Async Functions
- 2.6 Promises and Callback

Unit 3: Introduction to Node.js**10**

- 3.1 Traditional Web Server Model
- 3.2 Concept of Node.js and Event Loop
- 3.3 Installing Node.js
- 3.4 Components of Node.js Application
- 3.5 Modules, Types of Modules

- 3.6 Concept and examples of export
- 3.7 Create a basic application

Unit 4: Node Package Manager **06**

- 4.1 What is Node Package Manager (NPM)?
- 4.2 Packages in Node.js
- 4.3 Local and global Packages
- 4.4 Dependencies in packages
- 4.5 Managing dependencies
- 4.6 Updating packages

Unit 5: Express.js **12**

- 5.1 REST, RESTful services
- 5.2 Introduction to Express Frameworks
- 5.3 Working with Express Framework
- 5.4 Routing
- 5.5 Routes, Views and Public Assets
- 5.6 Handling HTTP GET/POST request
- 5.7 Calling Endpoints using Postman
- 5.8 Helpers and Module
- 5.9 Form Validation
- 5.10 Fetching Data

Unit 6: Working with Databases **06**

- 6.1 Connection String
- 6.2 Configuring
- 6.3 Working with Select command
- 6.4 Various database operations
- 6.5 MongoDB
- 6.6 Mongoose ODM
- 6.7 Mongoose Schema
- 6.8 Mongoose Model
- 6.9 Querying with Mongoose
- 6.10 Aggregations in mongodb

Reference Books:

1. Mastering Nod.js, Sandro Pasquali, Packt
2. Node.js complete reference guide, Velentin Bojinov, David Herron, Dioge Resende, Packt
3. Smashing Node.js, Java Script Everywhere , Guillermo Rauch, WILEY

Course Code and Title: PSCA-122 Operating System

Lectures: 48(Credits- 04)

Prerequisites:

- Working knowledge of C programming.
- Basic Computer Architecture concepts.
- Basic algorithms and data structure concepts.

Course Objectives:

1. To learn Advanced Operating Systems Concepts using Unix/Linux.
2. To understand the programming interface to the Unix/Linux system - the system call interface.
3. To grasp the concepts underlying in the design and implementation of Operating Systems.

Learning Outcomes:

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

1. implement advanced operating systems concepts in C.
2. develop efficient system software.
3. easier to understand internals of other operating systems and their functionalities.

Unit 1: UNIX System Overview

03

- 1.1 Introduction, UNIX Architecture
- 1.2 Logging In, Files and Directories
- 1.3 Input and Output
- 1.4 Programs and Processes
- 1.5 Error Handling
- 1.6 User Identification, Signals
- 1.7 Time Values System Calls and Library Functions

Unit 2: File I/O

08

- 2.1 File Descriptors
- 2.2 Functions: open and openat, creat, close
- 2.3 lseek Function, read Function, write Function
- 2.4 I/O Efficiency, File Sharing, Atomic Operations
- 2.5 dup and dup2 Functions
- 2.6 sync, fsync, and fdatasync Functions
- 2.7 fcntl Function
- 2.8 ioctl Function, /dev/fd

Unit 3: Files and Directories

12

- 3.1 stat, fstat, fstatat, and lstat Functions

3.2 File Types, File Access Permissions, access and faccessat Functions	
3.3 Set-User-ID and Set-Group-ID	
3.4 Ownership of New Files and Directories	
3.5 mask Function, chmod, fchmod, and fchmodat Functions	
3.6 Sticky Bit	
3.7 chown, fchown, fchownat, and lchown Functions	
3.8 File Size, File Truncation, File Systems	
3.9 link, linkat, unlink, unlinkat, and remove Functions, rename and rename at Functions	
3.10 Symbolic Links, Creating and Reading Symbolic Links	
3.11 File Times, futimens, utimensat, and utimes Functions	
3.12 mkdir, mkdirat, and rmdir Functions	
3.13 Reading Directories	
3.14 chdir, fchdir, and getcwd Functions	
3.15 Device Special Files	
Unit 4: Process Environment	05
4.1 Process termination	
4.2 Environment list	
4.3 Memory layout of a C program	
4.4 shared libraries	
4.5 environment variables	
4.6 setjmp and longjmp	
4.7 getrlimit and setrlimit	
Unit 5: Process Control	06
5.1 Process identifiers	
5.2 fork, vfork, exit, wait and waitpid, waitid,	
5.3 wait3 and wait4	
5.4 Race conditions, exec	
5.5 Changing user IDs and group IDs	
5.6 User identification, process times	
Unit 6: Memory Management	04
6.1 The Process Address Space, Allocating Dynamic Memory, Managing Data Segment, Anonymous Memory Mappings, Advanced Memory Allocation, Debugging Memory Allocations	
6.2 Stack-Based Allocations, choosing a Memory Allocation Mechanism, Manipulating Memory, Locking Memory, Opportunistic Allocation	

Unit 7: Signal Handling**10**

- 7.1 Signal concepts, signal function, unreliable signals
- 7.2 interrupted system calls, reentrant functions
- 7.3 SIGCLD semantics, reliable-signal technology
- 7.4 kill and raise, alarm and pause, signal sets, sigprocmask, sigpending, sigsetjmp and siglongjmp, sigsuspend, sigaction, sigqueue
abort, system function revisited, sleep, nanosleep, and clock_nanosleep Functions

Reference Books

1. Advanced Programming in the UNIX Environment, Richard Stevens, Addison-Wesley
2. Linux System Programming, Robert Love, O'Reilly

Course Code and Title: PSCA-123 Machine Learning

Lectures: 48 (Credits- 04)

Prerequisites:

- Familiarity with Probability Theory, Multivariable Calculus, Linear Algebra
- Programming in Python (NumPy, SciPy, Pandas, Matplotlib, Seaborn, SciKit-Learn, Stats Model)

Course Objectives:

1. To introduce the basic concepts and techniques of Machine Learning.
2. To apply Machine Learning techniques to solve problems from real world.
3. To understand the functionality of neural networks.
4. To help build career in the field of Data Science.

Learning Outcomes: After completion of the course, students will be able to:

1. apply machine learning toolboxes effectively.
2. recognize the characteristics of machine learning that make it useful to real-world problems.
3. understand the concept behind neural networks for learning non-linear functions.

Unit 1: Introduction

10

- 1.1 Introduction to Machine Learning Data Science, Artificial Intelligence and Machine Learning
- 1.2 Why Learn and What is Learning, What is Machine Learning, Traditional Programming Vs. Machine Learning, Machine Learning Process
- 1.3 Types of Data, Key Elements of Machine Learning (Representation, Evaluation and Optimization), Dimensionality Reduction (Feature Reduction)
- 1.4 Descriptive and Inferential Statistics: Probability, Distribution, Distance Measures (Euclidean and Manhattan)
- 1.5 Correlation and Regression, Hypothesis Testing, Creating own dataset, Importing the dataset, Handling Missing Data
- 1.6 Splitting the dataset into the Training set and Test set, Feature Scaling

Unit 2: Machine Learning Models

06

- 2.1 Machine Learning Models
- 2.2 Type of Learning - Supervised, Unsupervised and Semi- Supervised Learning
- 2.3 Components of Generalization Error (Bias, Variance, underfitting, overfitting)
- 2.4 A Learning System Cycle and Design Cycle
- 2.5 Metrics for evaluation viz. accuracy, scalability, squared error, precision and recall, likelihood, posterior probability
- 2.6 Classification Accuracy and Performance

Unit 3: Regression Models	10
3.1 Linear Regression - Simple, Multiple, Polynomial	
3.2 Non-linear Regression – Support Vector, Random Forest	
3.3 Logistic Regression	
Unit 4: Classification Models	08
4.1 Naive Bayes Theorem, k-Nearest Neighbours (k-NN)	
4.2 Support Vector Machine	
4.3 Decision Forest Classification	
4.4 Random Tree Classification	
Unit 5: Clustering Models	04
5.1 k-means	
5.2 Hierarchical Clustering (Agglomerative, Divisive) Dendrogram	
5.3 Selecting optimal number of clusters: Within Clusters Sum of Squares (WCSS) by Elbow Method	
Unit 6: Reinforcement Learning	04
6.1 Upper Confidence Bound	
6.2 Thompson Sampling	
6.3 Q-Learning	
Unit 7: Deep Learning	06
7.1 Artificial Neural Network	
7.2 Convolutional Neural Network	
7.3 Recurrent neural networks	

Reference Books

1. Machine learning, Mitchell, Tom M., WCB
2. Neural networks: A classroom Approach, Satish Kumar, Mc-Graw Hill
3. A first course in machine learning, Rogers, Simon, and Mark Girolami, CRC Press
4. Reinforcement learning: An introduction, Sutton, Richard S., and Andrew G. Barto, MIT press

Course Code and Title: PSCAELE-124A Java Programming

Lectures: 30 (Credits- 02)

Prerequisites:

- No any prior knowledge is required.
- Having knowledge of any other programming language is additional benefit.

Course Objectives:

1. To learn implementation of object-oriented concepts with Java.
2. To understand collection classes and interfaces.
3. To know the process of application development using Graphical User Interface (GUI).
4. To acquire knowledge about handling databases using Java.
5. To study web components for developing web applications.

Learning Outcomes:

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

1. Identify classes, objects, class members and relationships for a given problem.
2. Design end to end applications using object oriented constructs.
3. Apply collection classes for storing java objects.
4. Use Java APIs for program development.
5. Handle abnormal termination of a program using exception handling.

Unit 1: Introduction to Java 02

- 1.1 History and Features of Java
- 1.2 JDK, JRE, JIT, Bytecode and JVM
- 1.3 Simple java program
- 1.4 Data Types Variable: final, static
- 1.5 Array

Unit 2: Objects and Classes 04

- 2.1 Definition of Class
- 2.2 Access Specifiers, Constructors
- 2.3 Use of “this” keyword
- 2.4 String, String Buffer and Wrapper class
- 2.5 Introduction to Packages
- 2.6 Garbage Collection (finalize () Method)

Unit 3: Inheritance and Collection 06

- 3.1 Inheritance Basics, Types of Inheritance
- 3.2 Use of 'super' and 'Final' Keyword
- 3.3 Usage of abstract class and abstract methods
- 3.4 Interface

3.5 Introduction to Collection – List, Set	
Unit 4: Exception Handling and I/O	04
4.1 Introduction to Exception handling	
4.2 Exception types, Exception class	
4.3 User defined exception	
4.4 Introduction to Java.io package	
4.5 Byte streams, Character streams	
4.6 File IO basics	
4.7 Object serialization – Reader and Writer	
Unit 5: Swing	06
5.1 What is Swing?	
5.2 The MVC Architecture and Swing	
5.3 Layout Manager and Layouts, The JComponent class	
5.4 Components – JLabel, JButton, JText, JTextArea, JCheckBox, JRadioButton, JList, JComboBox, JMenu and JPopupMenu Class, JMenuItem Dialogs (Message, confirmation, input), JFileChooser	
5.5 Event Handling: Event sources, Listeners – ActionListener, ItemListener	
5.6 Mouse and Keyboard Event Handling	
5.7 Adapters – MouseAdapter, KeyAdapter	
5.8 Anonymous inner class	
Unit 6: Database Programming	04
6.1 Introduction to JDBC: JDBC Drivers	
6.2 Connectivity, JDBC statement Result Set and types	
6.3 JDBC Metadata	
Unit 7: JSP	04
7.1 Simple first JSP program	
7.2 Life cycle of JSP	
7.3 Implicit Objects	
7.4 Scripting elements –Declarations, Expressions, Scriptlets, Comments	
7.5 JSP Directives –Page Directive, include directive	
7.6 Mixing Scriptlets and HTML	
7.7 Example of forwarding contents from database to servlet servlet to JSP and displaying it using JSP scriplet tag	
Reference Books:	
1. Complete reference Java (5th edition), Herbert Schildt	
2. Java 2 programming black books, Steven Horlzner	
3. Java EE Project using EJB 3, JPA and struts 2 for beginners, Shah, SPD	
4. Core Java Volume-II Advanced Features, Eighth Edition, Cay S.Horstmann, Gary Cornell, Prentice Hall, Sun Microsystems Press	

Course Code and Title: PSCAELEP-124A Java Programming Laboratory
Practicals: 12 (Credits- 02)

List of Assignments

1. Basic programs on Classes and Objects
2. Based on Inheritance, Interface and Collection
3. Based on Exception Handling, File Handling
4. Based on Swing
5. Based on DB and JSP

Course Code and Title: PSCAELEPA-124B Project Implementation and Presentation

Practicals: 12 (Credits- 02)

Guidelines:

1. Project groups will work on actual development and/or implementation of proposed idea/topic.
2. Record of progress will also be maintained by keeping track sheet.
3. At the end of the project, the group should prepare a report which should conform to international academic standards. The report should follow the style in academic journals and books, with clear elements such as: abstract, background, aim, design and implementation, testing, conclusion and full references, Tables and figures should be numbered and referenced to in the report.
4. Minimum 2 demos will be conducted for the project work.
5. The final project presentation with demonstration (EE) will be evaluated.

Internal Assessment - 15M

- Seminar for the work done or presentation of the project work in the conference by writing research paper.
- Overall weekly work done by the project group.

External Assessment - 35M

- Report writing 10 marks
- Demonstration 25 marks

Course Code and Title: PSCAELE-124C Cryptography and Network Security

Lectures: 30 (Credits- 02)

Prerequisites:

- Basics of Computer Networks
- Familiarity with Number Theory
- Knowledge of C/Java

Course Objectives:

1. To enable students to get sound understanding of Info-Sys-Security, Net-Security, Cryptography.
2. To equip with knowledge and skills necessary to support for their career in Information Security.

Learning Outcomes:

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

1. Able to develop secure software.
2. Able to build career in Network Security.

Unit 1: Introduction 04

- 1.1 Need, Principles, Policy
- 1.2 Types of attacks
- 1.3 Basic Network Security Terminologies
- 1.4 Definition and Goals of Cryptography Techniques

Unit 2: Symmetric Key Cryptography 10

- 2.1 Stream Ciphers
- 2.2 Block Ciphers, Algorithm Types and Modes (Electronic code book, Cipher block chaining, Cipher feedback, Output Feedback)
- 2.3 Computer based Symmetric Key Cryptographic Algorithms (Data Encryption Standard and variations, International Data Encryption Algorithm, RC5, Blowfish)
- 2.4 Asymmetric Key Cryptography (Public Key Cryptography)
- 2.5 Diffie-Hellman Key exchange algorithm, RSA algorithm, One way hash function, Digital Signature, MD5, Secure hash algorithm
- 2.6 Digital Certificates

Unit 3: Introduction to Network Security 06

- 3.1 IP Security, IPsec protocol
- 3.2 Internet Key exchange protocol
- 3.3 Authentication header, Encapsulating Security Payload
- 3.4 Socket layer, Secure hypertext transfer Protocol

3.5 Secure electronic transaction, Pretty Good Privacy, S/MIME

Unit 4: Authentication **06**

- 4.1 User Authentication, Password based Authentication
- 4.2 Certificate based authentication, Biometric authentication, Kerberos, Ticket granting approach
- 4.3 Authentication Model, Kerberos and Public key cryptography, Applications of Kerberos, X.509 authentication service

Unit 5: Introduction to blockchain **04**

- 5.1 Foundational Computing Concepts (Client-Server systems vs Peer to Peer Systems)
- 5.2 Evolution of Blockchain
- 5.3 Blockchain Vs Database
- 5.4 Essentials of Blockchain (Blockchain generations, types of blockchain, benefits and challenges of blockchain usage)
- 5.5 Types of Networks
- 5.6 Layered Architecture of Blockchain Ecosystem
- 5.7 Components of blockchain
- 5.8 Cryptography (private and public keys, Hashing & Digital Signature)
- 5.9 Consensus Mechanisms
- 5.10 Cryptocurrency, Digital Currency Bitcoin and Ethereum
- 5.11 Smart Contracts
- 5.12 Blockchain use cases

Reference Books

1. Cryptography and Network Security, Atul Kahate, Tata McGraw-Hill Publishing Company Limited
2. Cryptography and Information Security, V.K. Pachghare, PHI Learning Private Limited
3. Applied Cryptography Protocols, Algorithms, and Source Code in C, Bruce Schneier, Wiley India
4. Mastering Blockchain by Imran Bashir, Third Edition, Packt Publication

**Course Code and Title: PSCAELEP-124C Cryptography and Network
Security Laboratory**

Practical: 12 (Credits- 02)

List of Assignments

1. Programs based on Ciphers
2. Based on symmetric cryptography
3. Based on asymmetric cryptography
4. Based on message digest
5. Based on advanced topics

**Course Code and Title: PSCAP-125 Full Stack Development – I and
Operating Systems Laboratory**

Practicals: 24 (Credits- 04)

List of Assignments for Full Stack Development – I

Assignment 1

JavaScript

Assignment 2

Basics of Node.js

Assignment 3

Based on NPM concept

Assignment 4

Node.js programs with DB

Assignment 5

Based on Express.js

**List of Assignments for Operating Systems
(Programs are to be done in 'C' language)**

Assignment 1

Basic programs

Assignment 2

Based on File Subsystem

Assignment 3

Based on process control

Assignment 4

Based on Signal Handling