



# Parvatibai Chowgule College of Arts and Science Autonomous

Accredited by NAAC with Grade 'A+'  
Best Affiliated College-Goa University Silver Jubilee Year Award

## SEMESTER V

## COMPUTER SCIENCE

**Course Title: Operating Systems**

**Course Code: COM-V.C-7**

**Marks: 75**

**Credits: 3**

**Duration:45 Hrs**

**Course Prerequisites:** Nil

### Course Objectives:

- To understand different functions of an operating system.
- To study various aspects of operating system like Process Management, Memory Management, Storage management etc.
- To understand different algorithms used for CPU scheduling, Memory allocation.

### Course Outcomes:

Upon completion of the course students will be able to:

**CO1:** Identify the functions of operating system.

**CO2:** Describe process management and process coordination.

**CO3:** Implement various CPU scheduling algorithms and compare their performance.

**CO4:** Explain deadlock handling and recovery from deadlock.

**CO5:** Implement various Memory Allocation algorithms.

**CO6:** Describe implementation of Virtual memory.

**CO7:** Describe storage management and storage structure.

### SYLLABUS:

#### UNIT I : Introduction to Operating System and Process Management

[15HRS]

Introduction to Operating System: Basic elements of a computer system: Processor, Main Memory, I/O Modules, System Bus, Instruction Execution; Operating Systems: Definition, Operating system Structure, operating system operations, Relationship between Kernel, OS, and Hardware, Operating system services, System calls, Types of system calls, System programs.

Process Management: Process Definition, Process Control Block, Process States, Operations on Process; Inter process communication, Threads and Microkernels: Definition, Multi-threading Model Process Scheduling, Scheduling Criteria, Scheduling Algorithms, Multi-Processor Scheduling.

#### UNIT II : Process Coordination

[15HRS]

##### Process Coordination

Process Synchronization, Principles, Mutual Exclusion, The Critical-Section Problem, Petersons Solution, Semaphores, Monitors, Readers/Writers Problem; Classic Problems of Synchronization, Dining Philosopher's problem.

Deadlocks- system models, Deadlock characterization, Deadlock Handling Methods, Prevention, Avoidance, Detection, Recovery From Deadlock.

### **UNIT III: Memory Management and Storage Management**

**[15HRS]**

Memory Management: Introduction, Swapping, Contiguous Memory Allocation, Paging, Page Table, Segmentation, Virtual Memory: Introduction, Demand Paging, Page Replacement, Allocation of Frames, Thrashing

Storage Management : File System, Concepts, File Organization and Access Methods, Directory and Disk Structure.

Secondary Storage Structure : Overview, disk structure, Disk attachment, Disk scheduling  
Reading.

#### **MANDATORY:**

1. Silberchatz, A., Galvin, , & Gagne. (2008). Operating System Concepts (8th ed.). Wiley publication.

#### **SUPPLEMENTARY:**

1. Stallings, W. (2001). Operating systems: Internals and design principles.(6<sup>th</sup> ed.). Upper Saddle River, N.J: Prentice Hall.

#### **WEB BASED:**

1. <https://www.geeksforgeeks.org/operating-systems/>

2. Video Links and Animations

3. <https://www.youtube.com/watch?v=WP3uDgIbPiI>

4. <http://williamstallings.com/OS-Animation/Animations.html>

5. Linux Tutorials for Practical

6. <https://www.tutorialspoint.com/unix/index.htm>

### **Practical : Operating Systems**

**Credit : 1**

**Marks : 25**

**Duration:30 Hrs**

#### **Any 8 from the following can be done.**

1. Demo/Review of Installing Linux / Windows Operating System, Partitioning and formatting disk, Installing applications device drivers, working with files, mounting file systems, checking system space, creating, modifying and deleting user accounts.
2. Study of Basic commands of Linux.
3. Shell Programming in Unix/Linux, arithmetic operations, loops.
4. Shell Programming – advanced.
5. Menu Driven Shell scripting.
6. Filters and Pipes in LINUX.
7. Implementation of Inbuilt Linux/UNIX commands like cp, rename etc.
8. Implementation of CPU scheduling policies.
9. Implementation of Memory allocation techniques.
10. Implementation of Banker's algorithm. (Resource Allocation Graph).

**Course Title: Embedded Systems**  
**Course Code: COM-V.E-9**  
**Marks: 75**  
**Credits: 3**  
**Duration: 45Hours**

---

**Course Prerequisites:**

- Knowledge of Programming

**Course Objectives:**

- To have a thorough understanding of major components of Embedded Systems.
- Implement small modules of programs to solve well defined problems in an Embedded Systems.
- To familiarize with the tools to develop applications in an Embedded Environment.

**Course Outcomes:**

Upon completion of the course students will be able to:

**CO1:** Describe Embedded Systems and its characteristics.

**CO2:** Classify the Embedded processors and their design metrics.

**CO3:** Summarize the performance of ARM processors and various components of Embedded Systems.

**CO4:** Classify Sensors and Actuators and identify their functions and applications.

**CO5:** Categorize I/O devices, I/O Interfacing and Communication protocols along with their functions.

**CO6:** Generalize the functionality of IoT and RTOS.

**CO7:** Design and develop Embedded / IoT Applications using Arduino/Raspberry-Pi boards.

**SYLLABUS:**

**UNIT I: Introduction to Embedded Systems and System Hardware**

**[15 HRS]**

Introduction to Embedded Systems, Microprocessors and Micro-controllers. Components of Embedded System & its Classification, Characteristic of embedded system. Introduction to embedded processor, Digital Signal Processor, Application Specific System Processor, Design Process in Embedded System, Design metrics, Steps in design process. Challenges in Embedded System design, Design Examples. Advances in Embedded Systems.

System hardware, Interrupt structure and Applications, ARM Processor - Architecture, Programmer's model, Modes of operation, Interrupt, Handling Interrupts, Comparison of ARM7 & ARM9.

Components of Embedded Systems-Management of Power Supply, Clocking Unit, Real Time Clock and Timers, Reset Circuitry and Watchdog Timer. Processor and Memory Selection, Memory Map of Embedded System, Interfacing Processors, Memories and I/O – Analog vs Digital. Overview of Arduino, Intel Edison and Raspberry Pi boards.

**UNIT II: Input/output Devices and Internet of Things**

**[15 HRS]**

I/O interfacing and Communication Buses, Serial vs Parallel Communication, Serial Data Communication RS-232/UART.

I/O devices, ADC / DAC, Optical Devices such as LED / LCD Display devices, Opto-Isolator, Relays, DC motor, Stepper motor, Timers/Counters, Parallel ports - Device interfacing. Serial communication

Protocols - UART Protocols, I2C, CAN, USB & ZigBee – Protocol Architecture, Topology, Packets, Communication Cycle, Arbitration, Applications and comparison.  
Introduction to IoT, M2M to IoT-The Vision-Introduction, M2M towards IoT- the global context, IoT Architectural Overview, Potential and Challenges.

### **UNIT III: Sensors, Actuators and RTOS**

**[15 HRS]**

**Sensors / Transducers:** Principles – Classification – Parameters – Characteristics –Environmental Parameters (EP) – Characterization.

**Thermal Sensors:** Introduction – Gas thermometric Sensors – Thermal Expansion Type Thermometric Sensors.

**Magnetic sensors:** Introduction – Sensors and the Principles Behind – Magneto-resistive Sensors, Semiconductor Magneto resistors.

**Smart Sensors:** Introduction – Primary Sensors – Excitation – Amplification – Filters – Converters – Compensation – Information Coding/Processing - Data Communication – Standards for Smart Sensor Interface – The Automation.

**Sensors Applications:** Introduction – On-board Automobile Sensors (Automotive Sensors) – Home Appliance Sensors – Aerospace Sensors — Sensors for Manufacturing – Sensors for environmental Monitoring.

**Actuators:** Overview of Pneumatic and Hydraulic Actuation Systems, Mechanical Actuation Systems, Electrical Actuation Systems.

**RTOS:** Introduction to RTOS, architecture of kernel, task and task scheduler, interrupt service routines in RTOS Environment.

### **REFERENCES:**

#### **MANDATORY:**

- 1.Rajkamal, (2010), Embedded Systems – Architecture, Programming and Design, (2<sup>nd</sup> ed.), Tata McGraw Hill.
- 2.D. Patranabis, (2013), Sensors and Actuators, (2nd ed), PHI.

#### **SUPPLEMENTARY:**

1. David E Simon, (2004), An Embedded Software Primer, (1<sup>st</sup> ed.), Pearson India.
2. Tammy Noergaard, (2012), Embedded Systems Architecture, (2<sup>nd</sup> ed.), Elsevier
3. D. Patranabis, (2010), Sensors and Transducer, (2<sup>nd</sup> ed.), PHI Learning Private Limited.
4. Jan Holler, VlasiosTsiatsis, Catherine Mulligan, Stefan Avesand, Stamatis Karnouskos, David Boyle, (2014), From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence, (1st ed), Academic Press,.

#### **WEB BASED:**

- 1.[https://www.tutorialspoint.com/internet\\_of\\_things/index.htm](https://www.tutorialspoint.com/internet_of_things/index.htm)
- 2.<https://www.arduino.cc/>
- 3.<http://www.zdnet.com/article/m2m-and-the-internet-of-things-a-guide>
- 4.<https://www.raspberrypi.org/>
- 5.[www.thingspeak.com](http://www.thingspeak.com)
- 6.<https://blynk.io/>

## **Practicals: Embedded Systems**

**Credit: 1**

**Marks: 25**

**Duration: 30 Hrs**

Programs to be executed on some of the embedded boards like Arduino, Intel Edison, Raspberry Pi, Bolt, etc that covers the following tasks (Any 10 experiments):

- 1) Interfacing sensors (3P)
- 2) Interfacing output devices (1P)
- 3) Interfacing input devices (1P)
- 4) Interfacing actuators (1P)
- 5) Building obstacle avoiding Robot (1P)
- 6) Line Following Robot (1P)
- 7) Programming with Raspberry Pi (2P)
- 8) Monitoring Data over Cloud (1P)
- 9) Building Web app to control devices (1P)
- 10) Mini Project.

**Course Title: Data Base Management System - II**

**Course Code: COM-V.E-17**

**Marks: 75**

**Credits: 3**

**Duration:45 Hrs**

---

**Course Prerequisites:**

- Knowledge of Data Base Management Systems.

**Course Objectives:**

- To provide advance database solutions.

**Course outcome:**

Upon completion of the course students will be able to:

**CO1:** Formulate complex queries for database updation.

**CO2:** Implement stored procedures and Functions.

**CO3:** Understand concurrent transactions and Recovery mechanisms.

**CO4:** Develop a full database application.

**CO5:** Understand and appreciate the NoSQL databases.

**SYLLABUS:**

**1. Advanced SQL**

**[15HRS]**

SQL data types and schemas, Integrity constraints, Authorization, Embedded SQL, Dynamic SQL, Triggers, Stored Procedures, views

**2. Indexing and Hashing**

**[7HRS]**

Basic concepts, Ordered Indices, Dense and Sparse Indices. B and B+ trees Hashing – Static hashing, Dynamic Hashing, Extendable hashing, Comparison of Ordered Indexing and Hashing.

**3. Transaction, Concurrency Control, Recovery System.**

**[10HRS]**

Transaction: Transaction concept, Transaction state, Implementation of Atomicity and Durability, concurrency. Serializability, conflict serializability. Concurrency Control : Lock-Based Protocol  
Recovery System: Failure Classification, Storage structure, Stable storage implementation, Recovery and Atomicity: Log-Based Recovery.

**4. Introduction to Big data and NoSQL**

**[13HRS]** Introduction to the Big Data problem. Current challenges, trends, and applications Comparison between SQL and NOSQL Databases Types and examples of NoSQL databases- Column, Document, Key-value, Graph, Multi-model. Introduction to Document type NoSQL database such as Mongo Db. - Introduce concepts of collection and documents, Advantages, Data types, Projections, indexing, Sharding .

**MANDATORY:**

1.A Silberschatz, H F Korth, S Sudarshan, Database system concepts, McGraw-Hill ,sixth Edition

**SUPPLEMENTARY:**

Ramakrishan, J Gehrke, —Database management systems”, McGraw-Hill , 3rd edition R Elmasri, S B Navathe, —Fundamentals of database Systems”, Pearson Education , 5th Edition Kristina Chodorow MongoDB : The Definitive Guide (English) O’Reilly 2nd Edition.

**Practicals: Database Management Systems II****Credits: 1****Marks: 25**

1. SQL Revision
2. Advance SQL- Dynamic SQL, Triggers Advance SQL- Stored Procedures
3. Using ODBC API for insertion of record into database. Using ODBC API for deletion of record. Using ODBC API for modification of data. Using ODBC API for data retrieval.
4. Installing and Creating a document using MongoDB concept Performing Indexing using MongoDB.
5. Performing aggregation functions on MongoDB Implementation of Master-Slave approach.
6. Connection of Mongo Db using Java Insertion, modification, deletion using Mongo Db Data retrieval using Mongo Db Sharding using Java and Mongo Db.
7. Mini Project

**Course Title: Web Development Framework**

**Course Code: COM-VI. E-18**

**Marks: 75**

**Credits: 03**

**Duration: 45 HRS**

---

**Course Prerequisites:**

- Knowledge of Web Design, Object Oriented Paradigm and Database Management System.

**Course Objective:-**

- Use Web Frameworks and Libraries to develop interactive web applications.

**Course Outcomes:**

On completion of the course student will be able to:

**CO1:** Use ReactJS to build rich and interactive front end applications.

**CO2:** Use NodeJS to develop back end application to accept POST, GET, PUT, DELETE requests.

**CO3:** Develop REST API's using NodeJS.

**CO4:** Write non-blocking and blocking JavaScript code.

**CO5:** Explain Framework and Libraries with respect Web Development.

**UNIT I**

**[15HRS]**

**ReactJS**

History of front end libraries, Motivation for using React, Thinking in React, One way binding, JSX + CSS modules, Virtual DOM, ES6

ReactJS components

Component life cycle, Component API, Render functions, State, Props, Mix ins.

**UNIT II**

**[15HRS]**

**ReactJS : Interaction between components**

Passing data from parent to child, Passing data from child to parent, Passing data between 2 components at the same level, Forms, Refs, React-Router, API integration.

NodeJS: Introduction. Brief overview on the benefits of using Node.js and how Node.js is used in modern web development, Node and NPM, Introduction to setting up a Node.js project, Importing modules using npm, Using core modules to make HTTP requests and manipulate the file system.

**UNIT III**

**[15HRS]**

**NodeJS: Express framework**

Set up a web server, Implementing API routing, Implementing middle-ware, Implementing URL parameters.

**NodeJS: MySQL module**

Setting up a database and connecting it to a Nodel server ,Storing and retrieving data from the database.



## REFERENCES

### MANDATORY:

1. Brett McLaughlin (2011). What Is Node?(1st ed) O'Reilly Media
2. Alex Banks (2017). Learning React.(1sted) Shroff /O'Reilly.

### REFERENCES:

1. Mario Casciaro (2016). Node.js Design Patterns (2nd ed) Packt Publishing Limited

### WEB BASED:

1. <https://www.tutorialspoint.com/nodejs/index.htm>
2. <https://reactjs.org/docs/getting-started.html>
3. <https://www.youtube.com/watch?v=Ke90Tje7VS0>
4. <https://ict.iitk.ac.in/node-js-books/>
5. <https://freefrontend.com/nodejs-books/>

## Practical: Web Development Framework

**Marks: 25**

**Duration: 30 hrs**

**Credits: 01**

### List of Practicals

1. Creating a simple web server. (1P)
2. Connect to MySQL database. (1P)
3. CRUD using MySQL database API's. (4P)
4. Fetch data from a form, validate and insert in the database. Deleted data in the database.
5. Updated in the database.
6. Display data from the database.
7. Uploading files. (1P)
8. Login functionality using sessions. (1P)
9. Using cookies to store website data. (1P)
10. Completion of Mini project.

**Course Title: Software Testing**

**Course Code: COM-V.E-12**

**Marks: 75**

**Credits: 3**

**Duration:45 Hrs**

---

**Course Prerequisites:**

- Knowledge of Software Engineering

**Course Objectives:**

- To understand the fundamental concepts in software testing
- To study the various software testing strategies
- To learn how to design test cases and execute them.

**Course Outcomes:**

Upon completion of the course students will be able to:

**CO1:** Understand the different software testing strategies.

**CO2:**Apply testing strategies to live projects.

**CO3:** Design test cases.

**CO4:** Execute test cases using software testing tools.

**SYLLABUS:**

**UNIT I:**

**[15HRS]**

Software testing principles - Software Testing- Need for testing, Psychology of testing ,Testing economics, SDLC and Testing, Verification & Validation. Quality Assurance, Quality Control.

Testing strategies and types - White box testing techniques - Statement coverage, Branch Coverage , Condition coverage, Decision/Condition coverage , Multiple condition coverage ,Dataflow coverage, Automated code coverage analysis, Inspections, Walkthroughs Code Review Black box testing techniques - Boundary value analysis, Robustness testing ,Equivalence partitioning, Syntax testing, Finite state testing, Levels of testing, Unit, Integration and System Testing, Compatibility Testing, Domain Testing, Adhoc Testing ,Use of Requirement, Traceability Matrix.

**UNIT II:**

**[20HRS]**

Integration Testing Waterfall - Top-down ,Bottom up ,Big bang, Sandwich System and Performance Testing - Types of system testing ,Functional and non-functional testing Acceptance Testing ,Setting entry and exit criteria for phases and typical product release scenarios ,Basic factors governing performance testing, Methodology for performance testing, Tools for performance testing.

Regression Testing - Purpose ,Timing, Choice of tests ,Smoke tests ,Best practices Internationalization and Localization testing - Preliminary concepts, Adhoc testing, Pair testing, Extreme testing, Agile testing, Exploratory testing, Defect seeding.

Usability Testing - Factors in usability testing, Aesthetics testing, Accessibility testing, Tools for usability testing.

Testing object oriented software - Definitions and Challenge differences from testing non-OO Software, Class testing strategies Class Modality, State-based Testing, Message Sequence Specification.

**UNIT III:**

**[10HRS]**

People and organizational issues in testing - Common people issues and myths in testing, Providing career paths in testing, Organizational structures for testing teams, Geographically distributed testing teams and success factors.

Test Management and Automation- Test

Planning, Test Management, Test Process, Test Reporting, Test Automation, Factors to consider in automation, Challenges in test automation, Test Metrics, Product Metrics, Process Metrics, Progress Metrics. Use of metrics in ascertaining product release.

**REFERENCES:**

**MANDATORY:**

1. Srinivasan D.,Gopalaswamy R.:(2009) Software Testing- Principles and Practices, 4th Edition: Pearson Publication.

**SUPPLEMENTARY:**

1. Jalote P., (2010)An Integrated Approach to Software Engineering, 3rd Edition :Narosa Publishing House
2. Pressman R., (2017) ,Software Engineering: A Practitioners Approach, 6th Edition :McGraw Hill Publication.

**WEB BASED:**

1.<https://www.guru99.com>

**Practicals: Software Testing**

**Credit: 01**

**Marks: 25**

**Duration: 30 Hrs**

1. Planning Test Cases
2. Generating Test Cases/Test Suite
3. Enhancing Tests
4. Debugging Tests
5. Running Tests
6. Analysing Results
7. Reporting Defects

## SEMESTER VI

**Course Title: Computer Networks**

**Course Code: COM-VI. C-8**

**Marks: 75**

**Credits: 3**

**Duration: 45 Hrs**

---

**Course Prerequisites:** Nil

### **Course Objectives:**

- To understand the basic concepts of Computer Networking.
- To understand the layered architecture of computer networks.
- To understand various transmission media used for networking.
- To understand working of various protocols in different layers.

### **Course outcomes:**

Upon completion of the course students will be able to:

**CO1:** Understand the need for Network and various layers of OSI and TCP/IP reference model.

**CO2:** Explain various Data Communications media.

**CO3:** Identify the different types of network topologies and Switching methods.

**CO4:** Describe various Data link Layer Protocols.

**CO5:** Identify the different types of network devices and their functions within a network.

**CO6:** Differentiate between various Classless and Class full IP addresses with Sub netting concept.

**CO7:** Analyze and interpret various Network and Transport Layer protocols.

**CO8:** Explain different application layer protocols.

### **SYLLABUS:**

#### **UNIT I**

**[20HRS]**

##### **Introduction**

Basics of Computer Networks, Classification: transmission technology, scale; Applications; Data Communications: data, signal, bandwidth, bit interval and bit rate, Modes of Communication. Layered network architecture, Networks models: OSI model, TCP / IP protocol suite; Guided and Unguided Transmission media, Multiplexing: FDM, TDM. Switching: Circuit switching, message switching, Packet Switching.

##### **Data link layer**

Data link control: Framing: Character Count, Character Stuffing, Bit Stuffing; , Error Detection and correction, Flow and error control, HDLC; Multiple access: Random access – Controlled access, ALOHA, CSMA, CSMA/CD and CSMA/CA; Ethernet : IEEE standards, standard Ethernet, Fast Ethernet, Gigabit Ethernet.

#### **UNIT II**

**[15 HRS]**

## **Network layer**

Connecting devices: repeater/hub, bridge, router and gateway, Backbone networks - Virtual LANS. Functions of Network layer; Network Service types: Virtual Circuits, Datagrams; Logical addressing: IPv4, private and public IP addressing, special IP addresses, subnetting, IPV6 addressing Internet Protocol: Internetworking: IPv4, Fragmentation and reassembly , Address mapping : ARP, RARP, BOOTP, DHCP, ICMP . Routing: classification of routing, Shortest path routing, Distance Vector routing, Link State routing;

## **UNIT III: Transport layer, Application layer and Wireless network [10HRS]**

Transport layer: Process-to-Process delivery: User Datagram Protocol (UDP), Transmission Control Protocol (TCP), Quality of services (QoS)

Application Layer: Domain Name System (DNS), E-mail, FTP, HTTP.

Basics of Wireless Networking

## **REFERENCES:**

### **MANDATORY:**

1. Tanenbaum A., W.(2010) Computer Networks (5<sup>th</sup> ed.) . Pearson Education.

### **SUPPLEMENTARY:**

1. Behrouz A. Forouzan B.(2017) Data communication and Networking(5<sup>th</sup> ed.). McGraw Hill Education
2. Kurose J, K.(2017) Computer Networking – A Top-Down Approach (6<sup>th</sup> ed.) Pearson Education,
3. Mir N. (2006) Computer and Communication Networks, Prentice Hall Publishers.

### **WEB BASED:**

1. <https://www.youtube.com/watch?v=tj7f244tubM>
  2. <https://www.youtube.com/watch?v=vFypCugyFoM>
  3. <https://www.geeksforgeeks.org/computer-network-tutorials/>
  4. Slides of the Book . Andrew S. Tanenbaum, David J. Wetherall“Computer Networks”, Prentice-Hall, 5<sup>th</sup> Edition. :<https://www.pearson.com/us/higher-education/product/Tanenbaum-Power-Point-Lecture-Slides-for-Computer-Networks-5th-Edition/9780132127066.html?tab=downloadable-resources>
  5. [https://www.tutorialspoint.com/data\\_communication\\_computer\\_network/index.htm](https://www.tutorialspoint.com/data_communication_computer_network/index.htm)
  6. [https://www.cisco.com › training-events › netacad › course\\_catalog › docs](https://www.cisco.com › training-events › netacad › course_catalog › docs)
- Slides for the book Behrouz A. Forouzan, “Data communication and Networking”, Tata McGraw – Hill, 2011, 4<sup>th</sup> Edition: 8. <http://www.mhhe.com/engcs/compsci/forouzan/dcn/index.mhtml>

**Practicals: Computer Networks**

**Credits: 1**

**Marks: 25**

**Duration: 30 Hrs**

Practical (Any 6 practical):

1. Setting up of LAN Network (2P)
2. IP address manipulation -Extract network id and Host id given netmask (2P)/Mini Project
3. Configuring routing tables
4. TCP Socket programming (2P)
5. UDP Socket programming (2P)
6. Mini Project / Simulation of IP fragmentation
7. Mini Project/Configuring E-Mail/DNS
8. Installing virtual machines, Ethernet cabling

**Course Title: Network Security**  
**Course Code: COM-VI. E-13**  
**Marks: 75**  
**Credits: 3**  
**Duration: 45 Hrs**

---

**Course Prerequisites:**

- Knowledge of Programming

**Course Objectives:**

- To understand the theory and concepts of Network Security.
- To be able to secure a message over insecure channel by various means.
- To learn about how to maintain the Confidentiality, Integrity and Availability of a data.

**Course Outcomes:**

Upon completion of the course students will be able to:

- CO1:** classify the symmetric encryption techniques
- CO2:** Illustrate various Public key cryptographic techniques
- CO3:** Evaluate the authentication and hash algorithms.
- CO4:** Discuss authentication applications.
- CO5:** Summarize the intrusion detection and its solutions to overcome the attacks.
- CO6:** Basic concepts of system level security

**SYLLABUS**

**UNIT I:**

**[15HRS]**

**Concepts of Security & Classical Encryption Techniques:**

Introduction, The need for security, Security Approaches, Security Attacks, Security Services, Security Mechanisms, A Model for Network Security, Classical Encryption Techniques :Substitution techniques, Transposition techniques, Steganography.

**Design Principle of Block Cipher:**

Block Cipher Operation: Electronic Code Book, Cipher Block Chaining, Cipher Feedback, Output Feedback, Counter, Feistel Cipher, The Data Encryption Standard.

**Cryptography:**

Mathematical Tools

Introduction to Number Theory, Modular Arithmetic, Prime Numbers, Euler's Totient Function.

**UNIT II:**

**[15HRS]**

**Public Key Cryptography:**

Principles of Public Key Cryptosystems, The RSA Algorithm, Other Public key cryptosystems, Diffie Hellman Key Exchange.

**Cryptographic Hash Functions:**

Applications of Cryptographic Hash Functions, Two Simple Hash Functions, Hash Functions Based on Cipher Block Chaining, MD5 Message Digest Algorithm, Secure Hash Algorithm SHA 512.

**Message Authentication Codes and Digital Signatures:**

Message Authentication Requirements – Message Authentication Functions –Requirements for Security of MACs ,MACs Based on Hash Functions, HMAC, MACs Based on Block Ciphers, Data Authentication Algorithm. Digital Signatures, Digital Signature Standard.

**UNIT III:****[15 HRS]****Key Management & Distribution And User Authentication:**

Introduction, Digital Certificate, Private key Management, The PKIX Model, Public key cryptographic standards ,XML, PKI and security.

**Program Security:**

Flaws, Malicious code: viruses, Trojan horses, worms, Program flaws: buffer overflows, time-of check to time-of-use flaws, incomplete mediation.

**Firewall and Virtual Private Network:**

Introduction to network security techniques: IP Security, firewalls, virtual private networks.

**REFERENCES:****MANDATORY:**

1. William.S,(2017).Cryptography and Network Security – Principles and Practices(7th ed.).Prentice Hall of India.

**SUPPLEMENTARY:**

- 1.Charles P. Pfleeger and Shari L. Fleeger(2015).Security in Computing(7th ed.). Prentice-Hall.
- 2.Atul.K,(2007).Cryptography and Network Security(2nd ed.).Tata McGraw-Hill.
- 3.Menezes A. J., P.C. Van Oorschot and S.A. Vanstone(1997) —Handbook of Applied Cryptography(2nd ed.).Jaypee medical.

**WEB BASED:**

1. Cryptography and Network Security - Nptel:  
<https://nptel.ac.in/courses/106/105/106105031/>
2. William Stallings, —Cryptography and Network Security – Principles and Practices, Prentice Hall of India:  
<http://www.amorena.com.ar/PAG%20DE%20MATERIAS%20Y%20LIBROS/LIBROS%20TODOS/CRYPTOGRAPHY%20AND%20NETWORK%20SECURITY,%20PRINCIPLES%20AND%20PRACTICE.pdf>
3. Cryptography Tutorial:  
<https://www.tutorialspoint.com/cryptography/index.htm>
4. IPSec, VPN, and Firewall Concepts:



[http://www.cs.unh.edu/~it666/reading\\_list/Networking/firewall\\_concept\\_terms.pdf](http://www.cs.unh.edu/~it666/reading_list/Networking/firewall_concept_terms.pdf)

5. Implementation of Hill Cipher:

<https://www.tutorialspoint.com/cplusplus-program-to-implement-the-hill-cypher>

**Practicals: Network Security**

**Credits: 1**

**Marks: 25**

**Duration: 30 Hrs**

1. Implementation of Caesar Cipher
2. Implementation of One-Time Pad
3. Implementation of Playfair Cipher
4. Implementation of Hill Cipher
5. Implementation of Data Encryption Standard Algorithm
6. Implementation of Image Steganography
7. Implementation of RSA Algorithm
8. Implementation of Digital Signatures using RSA Algorithm
9. Design Network protocol analyser tool to analyse network traffic.
10. Mini Project/ Case Study

**Course Title: Cloud Computing**

**Course Code: COM-VI.**

**E-14**

**Marks: 75**

**Credits: 3**

**Duration:45 Hrs**

**Prerequisite Courses:**

- Operating Systems(COM-V.C-7)

**Course Objectives:**

- To make students understand the key elements of cloud computing.
- To understand the difference between deploying applications on the cloud and the local infrastructure.
- To understand various cloud service models.

**Course Outcomes:**

At the end of the course students will be able to :

**CO1:** Explain the core concepts of the cloud computing paradigm.

**CO2:** Characterize the different cloud services ie. Infrastructure, Platform and Software as a Service (IaaS, PaaS, SaaS).

**CO3:** Deploy application in a production environment.

**CO4:** Host a cloud platform like Apache OwnStack and Owncloud.

## **SYLLABUS:**

### **UNIT I: Introduction to Cloud Computing [15HRS]**

Recent trends in Computing- Grid Computing, Cluster Computing, Distributed Computing, Utility Computing, Cloud Computing. Cloud Computing definition, History of Cloud Computing, How Cloud Computing Works, 5-4-3 Principles of Cloud Computing, Cloud Ecosystem, Benefits and challenges of cloud computing, Issues for Cloud Computing.

### **UNIT II: Cloud Computing Architecture [15HRS]**

Comparison with traditional computing architecture (client/server), Cloud Architecture - Benefits of Cloud Architecture, Cloud Computing Architectural components. Cloud Computing Service Models- IaaS, PaaS and SaaS. Deployment Models- Public cloud, Private cloud, Hybrid cloud and Community cloud, Key drivers to adopting cloud, Impact of cloud on users, Governance in the cloud. Introduction to Virtualization, Different approaches to Virtualization, Hypervisors, Types of Virtualization, Machine Image, Virtual Machine(VM).

### **UNIT III: Cloud Service Providers [15HRS]**

**Amazon Web Services (AWS)**- Business and Technical benefits of Cloud Services, Elasticity, IAM, Security Groups, VPC, EC2, Traditional IP v/s Elastic IP, Amazon Machine Image, Elastic Load Balancing, CloudWatch, Elastic Block Storage, S3, SQS, SNS, RDS, Dynamo DB, Ops Works, Server less Architecture- API Gateways, Lambda. Hadoop- MapReduce, HDFS, Hadoop on AWS.

**Microsoft Windows Azure:** Azure Virtual Machines, Cloud Services, Application scenarios.

### **REFERENCES:**

**MANDATORY:**

1. Chandrasekaran, K.. Essentials of Cloud Computing. United Kingdom, CRC Press, 2014.
2. Tim mather, subra kumarswamry and sharhed Latif, "Cloud Computing Security and Privacy", O'Reilly publication.
3. Richard Hill, Laurie Hirsch, Peter Lake, Siavash Moshiri, "Guide to Cloud Computing Principles and Practices", Springer.
4. Buyya, R., Broberg, J., & Goscinski, A. M. (Eds.). (2010). Cloud computing: Principles and paradigms (Vol. 87). John Wiley & Sons.

**SUPPLEMENTARY:**

1. Nikos Antonopoulos, Lee Gillam "Cloud Computing: Principles, Systems and Applications", Springer.
2. Vines, R. L. K. R. D., & Krutz, R. L. (2010). Cloud security: A comprehensive guide to secure cloud computing (pp. 35-41). Wiley Publishing, Inc.

**WEB BASED:**

1. <https://azure.microsoft.com/>
2. <https://aws.amazon.com/what-is-cloud-computing/>
3. <https://cloud.google.com/appengine>
4. <https://www.ibm.com/cloud/learn/cloud-computing>
5. <https://www.salesforce.com/in/learning-centre/tech/cloudcomputing/>

## **Practicals: Cloud Computing**

**Credit: 1**

**Marks: 25**

**Duration: 30 Hr**

### 1. Virtualization

**[3P]**

- a. Install Oracle VirtualBox/VMware and create two VMs on your laptop.
- b. Install VirtualBox/VMware Workstation with Linux and windows OS on top of windows.
- c. Install a C compiler in the virtual machine created using virtual box and execute Simple Programs
- d. Find a procedure to transfer the files from one virtual machine to another virtual machine.
- e. Test ping command to test the communication between the guest OS and Host OS

### 2. Docker container

**[3P]**

- a. Installation and Setup of Docker container Linux/Windows
- b. Containerizing the Python/Java/Node.js application
- c. Creating a persistent Data Base.
- d. Logging into Docker Hub
- e. Pushing the container to the Docker hub.
- f. Running the application from the container in pushed Docker hub

### 3. Own Cloud(Iaas)

**[2P]**

- a. Installation and setup of Own Cloud

- b. Explore the features of Own Cloud
  - c. Create, Manage and groups User accounts in own Cloud by Installing Administrative Features.
  - d. Federated sharing
4. Google App Engine.
- [2P]**
- a. Install and configure Google app engine
  - b. Create hello world app and other simple web applications using python/java.
  - c. Use GAE launcher to launch the web applications.
5. Heroku(PaaS)
- [2P]**
- a. Deploy a website on Heroku
  - b. Integrate Postgresql and deploy website on heroku
6. Hadoop
- [3P]**
- a. Install Hadoop single node setup.
  - b. Develop hadoop application to count no of characters, no of words and each character frequency.
  - c. Develop hadoop application to process given data and produce results such as finding the year of maximum usage, year of minimum usage
  - d. Install Hadoop single node cluster and run simple applications like wordcount.

**Course Title: Introduction to Data Science**

**Course Code: COM-V.E-11**

**Marks: 75**

**Credits: 3**

**Duration:45 Hrs**

---

**Course Prerequisites:**

- Statistical Methods
- Basic probability and statistics.

**Course Objectives:**

- Become familiar with methods of data science and their practical usefulness.
- To learn, understand, and practice machine learning approaches.
- To analyse large and unstructured data with different tools.

**Course outcomes:**

Upon completion of the course students will be able to:

**CO1:** Describe what Data Science is and the skill sets needed to be a data scientist.

**CO2:** Explain in basic terms what Statistical Inference means. Identify probability distributions commonly used as foundations for statistical modelling. Fit a model to data.

**CO3:** Explain the significance of exploratory data analysis (EDA) in data science. Apply basic tools(plots, graphs, summary statistics) to carry out EDA.

**CO4:** Describe the Data Science Process and how its components interact.

**CO5:** Apply basic machine learning algorithms for predictive modelling.

**CO6:** Identify common approaches used for Feature Generation. Identify basic Feature Selection.

**CO7:** Reason around ethical and privacy issues in data science conduct and apply ethical practices.

**CO8:** Create effective visualization of given data (to communicate or persuade).

**CO9:** Use of Mining Social-Network Graphs in Data science out basic statistical modelling and analysis.

**SYLLABUS:**

**UNIT I:**

**[10HRS]**

**Introduction to Data Science**

What is Data Science? Big Data and Data Science hype -and getting past the hype,  
Why now? –Datafication, Current landscape of perspectives, Skill sets needed.

**Statistical Inference:**

Populations and samples, Statistical modelling, probability distributions, fitting a model,  
Intro to R.

**UNIT II:**

**[20HRS]**

**Exploratory Data Analysis and the Data Science Process:**

Basic tools (plots, graphs and summary statistics) of EDA, Philosophy of EDA, The Data Science Process, Case Study: Real Direct (online real estate firm).

**Three Basic Machine Learning Algorithms:**

Linear Regression, k-Nearest Neighbours (k-NN),k-means.

**Feature Generation and Feature Selection (Extracting Meaning From Data):**

Motivating application: user (customer) retention, Feature Generation (brainstorming, role of domain expertise, and place for imagination), Feature Selection algorithms, Filters; Wrappers; Decision Trees; Random Forests.

**UNIT III:**

**[15HRS]**

**Mining Social-Network Graphs:**

Social networks as graphs, Clustering of graphs, Direct discovery of communities in graphs, Partitioning of graphs, Neighbourhood properties in graphs

**Data Visualization:**

Basic principles, ideas and tools for data visualization, Examples of inspiring (industry) projects, Exercise: create your own visualization of a complex data set.

**Data Science and Ethical Issues:**

Discussions on privacy, security, ethics, A look back at Data Science, Next-generation data scientists

**REFERENCES:**

**MANDATORY:**

1.O'Neil, C., & Schutt, R. (2013). Doing data science: Straight talk from the frontline. " O'Reilly Media, Inc."

**SUPPLEMENTARY:**

1. Jure.L., Anand. R,Jeffrey.U(2014). Mining of Massive Datasets v2.1(2nd ed.).Cambridge University Press.
2. Kevin P. Murphy,(2012).Machine Learning: A Probabilistic Perspective .MIT Press.
- 3.Provost, F., & Fawcett, T. (2013). Data Science for Business: What you need to know about data mining and data-analytic thinking. " O'Reilly Media, Inc."
- 4.Hastie, T., Tibshirani, R., & Friedman, J. (2009). The elements of statistical learning: data mining, inference, and prediction. Springer Science & Business Media..
- 5.Blum, A., Hopcroft, J., & Kannan, R. (2020). Foundations of data science. Cambridge University Press.
6. Zaki, M. J., Meira Jr, W., & Meira, W. (2014). Data mining and analysis: fundamental concepts and algorithms. Cambridge University Press.
7. Han, J., Pei, J., &Kamber, M. (2011). Data mining: concepts and techniques. Elsevier.

**WEB BASED:**

1. Cathy O'Neil and Rachel Schutt. Doing Data Science, Straight Talk From The Frontline, O'Reilly <https://classroom.google.com/u/0/w/NDA4ODg4MTQ3MjZa/t/all>
2. Data Science E-Learning Course:

[https://onlinecourses.nptel.ac.in/noc19\\_cs60/unit?unit=5&lesson=6](https://onlinecourses.nptel.ac.in/noc19_cs60/unit?unit=5&lesson=6)

3. Simple Linear Regression Example:

<https://www.spss-tutorials.com/simple-linear-regression/>

4. <https://www.kaggle.com/pavansanagapati/a-simple-tutorial-on-exploratory-data-analysis>

5. Data visualization

6. <https://paldhous.github.io/ucb/2016/dataviz/week2.html>

7. <https://www.targetprocess.com/articles/visual-encoding/>

### **Practicals : Introduction to Data Science**

**Credit: 1**

**Marks: 25**

**Duration: 30 Hrs**

1. Implementation of probability distribution
2. Sampling and re-sampling.
3. Linear Models
4. K-Nearest neighbour
5. K-Means
6. Feature Selection Algorithm
7. Filters and Wrappers
8. Decision Trees

**All the experiments will be implemented using Excel /R-Tool/ or equivalent.**



**Course Title: Web Development with Laravel**

**Course Code: COM-E-19**

**Marks: 75**

**Credits: 03**

**Duration: 45hrs**

**Course Prerequisite: Object Oriented Programming**

**Course Objectives:**

1. Provide an in depth understanding of a server side language, and use it to develop applications
2. Design and implement basic server-side scripts.
3. Create responsive and interactive web applications using frameworks

**Course Outcomes:**

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

CO1: Explain the core features and functionalities of PHP

CO2: Design interactive web application using core PHP

CO3: Develop basic server side script to interact with users and the database

CO4: Build a web application using laravel framework

CO5: Utilize MVC model

**UNIT 1: [15 Hrs]**

Static vs. Dynamic web pages, Need for Server Side technologies, Multi Tier Web Architecture, file architecture of a web server, Use of PHP Tags, Tag Styles, include, Processing GET and POST request, uploading files to server, cookies, sessions, Difference between MySQLi and PDO, CRUD - Create, Read, Update, Delete, records in database, pagination, login.

## **UNIT 2 [15 Hrs]**

Introduction to Laravel, Routing in Laravel, MVC in Laravel, Caching in Laravel, Event subscribers in Laravel, Package Development, Templates, XML and AJAX, defining AJAX array, parse AJAX using JQuery, AJAX request, AJAX response.

## **UNIT 3 [15 Hrs]**

Creating an Application, Database Configuration, Helpers in Laravel, Laravel Pagination, Laravel Security, Authentication Facade, Validation in Laravel, Eloquent ORM, Artisan Command Line Interface, Deploy Application using Laravel.

## **REFERENCE**

### **Mandatory Reading:**

1. Leon A, Zee S(2004), *Core PHP Programming* (3rd ed.).Prentice Hall Professional
2. Stauffer, M. (2019). *Laravel: Up & Running: A Framework for Building Modern PHP Apps*. O'Reilly Media.

### **Supplementary Reading:**

1. Williams, H. E., & Lane, D. (2004). *Web Database Applications with PHP and MySQL: Building Effective Database-Driven Web Sites*. O'Reilly Media, Inc.
2. Brinzarea, B., & Hendrix, A. (2009). *Ajax and PHP: Building modern Web applications*. Packt Publishing Ltd.

### **Web References:**

1. <https://www.w3schools.com/php/>
2. <https://www.tutorialspoint.com/php/index.htm>
3. <https://laravel.com/docs/6.x> <https://www.tutorialspoint.com/laravel/index.htm>

**Practical** :Web Development with Laravel

**Marks:** 75

**Duration:** 45hrs

**Credits:** 03

1. 1. PHP Classes and instances, PHP Controls Structures [1P]
2. PHP Array Programming, Inheritance [1P]
4. CRUD using PHP database API's. [3P]
5. Fetch data from a form, validate and insert in the database, Delete data in the database ,Update data in the database, Display data from the database .Uploading files and session management. [1P]
6. Implementing MVC [2P]
7. Migrations in Laravel [1P]
8. Using Forms and Gathering Input in Laravel [1P]
9. Creating a registration & user login form in Larvael [1P]
10. Using Controllers and Routes for URLs and APIs in Laravel [1P]
11. Eloquent ORM in Laravel [1P]
12. Creating and Using Composer Packages [1P]
13. Security & Session [ 1P]

## **SKILL ENHANCEMENT COURSE (SEC)**

**Course Title: Graphic Design**

**Course Code: COM-SEC4**

**Marks: 100**

**Credits: 4**

**Duration: 60 Hrs**

**Course Prerequisites: Nil**

### **Course Objectives:**

To understand the basic concept of Multimedia.

To develop skills for designing graphical images.

To develop skills for audio and video editing.

To acquire skills in using audio/video editing software.

### **Course Outcomes:**

Upon completion of the course students will be able to:

**CO1:** Understand the concept of Multimedia – Team members and their roles.

**CO2:** Identify and describe the function of the general skill sets in the multimedia industry.

**CO3:** Classify and realize the types of Authoring tools and their functions.

**CO4:** Identify basic components of a multimedia project.

**CO5:** Analyze the requirements of Multimedia product.

**CO6:** Assemble and deliver multimedia projects

### **SYLLABUS:**

#### **UNIT I: Introduction to Multimedia and Computer Graphics: [15HRS]**

Commonly used terms associated with multimedia like CDROM, Storyboard, Script and Authoring tools. Stages of a Multimedia Project-Planning and Costing, Designing and Producing, Testing and Delivering. The Multimedia team and their roles- Project Manager, Writer, Video specialist, Audio specialist and Multimedia programmer. Multimedia Software. Multimedia Hardware. Social & Ethical considerations, Digital Representations & Standards. A survey of Computer Graphic:

Computer Aided Design, Presentation Graphics, Computer art, Entertainment, Education and

Training, Visualization, Image Processing, Graphical User Interface

Multimedia Authoring Tools:

Types of authoring tools; Card and Page based tools; Icon-based; Time-based; Object-Oriented Tools.

TEXT: Text in graphics, character set, fonts, layout.

Character Attributes, Text Attributes, Marker Attributes, Bundled Text Attributes, Bundled Marker Attributes.

Vector graphics fundamentals, shapes, transforms and filters, Bitmapped graphics: resolution, image compression, manipulation, Geometrical transformations.

Activity: Design a Brochure for a given product, give details. Learn about different Image file Formats.

## **UNIT II: Audio, Video and Color [15HRS]**

Basic Sound Concepts, Digitizing and processing sound, Music, Speech, Compression, formats, MIDI and Digital Audio Human vision, Camera systems, Gamma correction, Color matching, different Color models – RGB, CYMK, Transformations among color model.

Video: Analog and Digital Video, video standards, Video on PC. Introduction to graphics accelerator cards, Video Broadcast Standards - NTSC, PAL, SECAM, HDTV. Introduction to video capturing, Media & Instrument – Videodisk, DVCAM, Camcorder.

Recording Formats like S-VHA Video, Component (YUV), Component Digital, Composite Digital, and Video Hardware Resolutions.

Video Tips like shooting platforms, Lighting, Chroma Key or Blue Screen.

### **Activity:**

1. Record and Edit the Sound file and Learn about Effects and Filters of sound
2. Prepare Video content with title and special effects.

**UNIT III: Animation [15HRS]** Basics of animation, Principle and use of animation in multimedia, Animation techniques-persistence of vision, animation file formats, Computer animation kinematics and Morphing. Design of Animation Sequence, General Computer Animation Functions, Computer Animation Languages, Key frame System, Particle systems – particle generation, modeling water, fire, explosions.

Effect of resolutions, pixel depth, Images size on quality and storage. Overview of 2-D and 3-D animation techniques and software. Animation on the Web – features and limitations, creating simple animations for the Web. Animation file

formats.

**Activity:** Animate 2D cartoon characters in real-time.

#### **UNIT IV: Multimedia on the Web [15HRS]**

Bandwidth relationship, broadband technologies, Text in the web – Dynamic and embedded font technology, Audio on the Web – Real Audio and MP3/MP4, Audio support in HTML, Graphics – HTML safe color palate, Interlaced V/s Non interlaced model, Graphics support in HTML, Image Map, Video on the Web – Streaming video.

**Activity:**

1. Prepare multimedia content for the web.

#### **REFERENCES:**

##### **MANDATORY:**

1. Chapman, N., & Chapman, J. (2005). Digital multimedia. John Wiley & Sons, Inc.
2. Li, Z. N., Drew, M. S., & Liu, J. (2004). Fundamentals of multimedia (pp. 253-265). Upper Saddle River (NJ): Pearson Prentice Hall.
3. Vaughan, Tay; Multimedia: Making it Work; Tata McGraw-Hill, 9th edition.

##### **SUPPLEMENTARY:**

1. Jeffcoate, J. (1995). Multimedia in practice. Technology and Applications. Great Britain: Prentice Hall.

##### **WEB BASED:**

1. <https://www.gimp.org/>
2. <https://www.audacityteam.org/>
3. [https://swayam.gov.in/nd2\\_ugc19\\_hs42/](https://swayam.gov.in/nd2_ugc19_hs42/)
4. <https://www.tutorialspoint.com/multimedia>
5. <https://libguides.bc.edu/>

**Course Title: Programming in Python**

**Course Code: COM-SEC5**

**Marks: 75**

**Credits: 3**

**Duration: 45 HRS**

**Course prerequisite:** Nil

**Course Objectives:**

To provide skills of data analysis using Python

programming language.

**Course Outcomes:**

At the end of the course students should be able to :

**CO1:** Understand syntax of Python Programming.

**CO2:** Write program using conditional statements, loops.

**CO3:**Apply required List function.

**CO4:** Write Python program specific to the domain of the

given problem.

**SYLLABUS:**

**UNIT I: Introduction to Python [15HRS]** Motivation, programming paradigms, What Python can do, Python's technical strength, Python interpreter, Program execution, Execution model variations, How to run programs.

Basic Syntax

Variable Data Types, Operator, Conditional

Statements-if,if-else, Nested If else.Looping–For, While, Nested loops. Control Statements–Break, Continue, Pass.

## **UNIT II: String Manipulation, Tuple, Lists and Dictionaries:**

### **[15HRS]**

Accessing Strings,Basic Operations,String Slices,Function and Methods. Tuple and Lists

Introduction, Accessing list, Operations, Working with lists, Function and Methods. Introduction Accessing tuples, Operations, Working, Functions and Methods.

## **UNIT III: Functions, Modules and Input-Output [15HRS]**

Introduction, Accessing values in dictionaries, Working with dictionaries, Properties, Functions. Functions. Defining a function, Calling a function, Types of functions, Function Arguments, Anonymous functions, Global and local variables.

### **Modules**

Importing module. Math module. Random Module.

### **Exception Handling**

Exception.ExceptionHandling-Exceptclause,Try,except,finallyclause.UserDefined Exceptions

### **Input-Output**

Printing Onscreen,Reading data from Keyboard,Opening and closing file,Reading and writing files, Functions. Printing Onscreen,Reading data from Keyboard,Opening and closing file,Reading and writing files, Functions.

## **REFERENCES:**

### **MANDATORY:**

1. Mark Lutz, Learning Python, O'Reilly Media, Fifth Edition.

### **SUPPLEMENTARY:**

1. Alex Martelli,(2006)Python–A Nutshell, O'Reilly Media, Second Edition. 2. WesMc Kinney,(2012)Python for Data Analysis ,O'Reilly Media.



**WEB BASED:**

1. <https://www.w3schools.com>
2. <https://www.tutorialspoint.com>
3. <https://www.javatpoint.com>
4. <https://www.geeksforgeeks.org>
5. <https://www.guru99.com>

**Practicals: Programming in Python****Credit: 01****Marks: 25****Duration:30 Hrs****List of Experiments using Python Language:**

1. Write a python program to compute a given formula.
2. Write a python program to implement an if else statement.
3. Write a python program to implement nested if else statements.
4. Implement for and while loop in python. [2P]
5. Write a string manipulation in python. [2P]
6. Write a program to implement lists in python.
7. Write a python program to implement tuple.
8. Write a python program to implement a dictionary.
9. Write a python program to implement a function.
10. Implement Module in python
11. Write a python program to implement exception handling.
12. Write a python program to implement Input-Output File operations in python. [2P]

**Paper Title : Web Development with FLASK**

**Paper Code : COM-SEC6**

**Marks : 75**

**Credits : 3**

**Hours:45Hrs**

**Course Prerequisites:** Programming in Python, Database Management System

**Course Objectives:**

To develop skills to build dynamic web applications, ensuring security and scalability. Train students to deploy web applications and configure the same for a development and production environment.

**Course Outcome:**

At the end of the course students will be able to:

**CO1:** Understand the basic of server client architecture on how request are made, handling those requests and providing an appropriate response.

**CO2:** Understand how to scale a web application.

**CO3:** Secure a web application from CSRF, Sql injection and XSS .

**CO4:** Handle dependencies management across multiple web application using python virtual environment.

**CO5:** Configure a web application for development and production.

**CO6:** Build a dynamic web site / web app.

**CO7:** Create API .

**CO8:** Deploying a web application

**Syllabus:-**

**UNIT I : Introduction to Web Development: [15 HRS]** Client-Server architecture, Requests: GET, POST. Web addresses and URL, Dynamic Websites, Static Websites, Database Servers and Web Servers. MVC and MVT.

Introduction to Flask. Flask and Other web frameworks. Introduction to python modules and pip. Python venv module, Installing modules using pip, dependency management.

Configuration and Deployment Flask configuration basic , Built in Configuration Values: SECRET\_KEY, SESSION\_COOKIE\_PATH, LOGGER\_NAME,

APPLICATION\_ROOT Development / Production configuration, Configuration Best Practices. Deployment on Apache HTTP server.

## **Jinja 2.**

Introduction to Jinja, Jinja Setup, Standard Context , Standard Filters , Controlling Auto escaping. Registering Filters, Context Processors.

### **UNIT II : Building a Minimal Application: [15 HRS]**

Routing, Starting a flask server, URL building, Rendering Templates, Debugging, Accessing request data,HTML methods: GET POST, Cookies Sessions Handling. Redirecting and Errors. Message Flashing. Logging. Building a salable project structure.

### **FLASK WT Forms and FLASK SQL Alchemy (ORM):**

WT Form Introduction, Creating Forms, Validating Forms , Securing Forms, File Uploads, CSRF protection. SQL Alchemy setup and installation, Creating database using Sql Alchemy, Simple Relationship, One to many and many to many relationship, Inserting records , Deleting records , Editing Records , Querying Records

### **UNIT III: Flask Security [15 HRS]**

Session based authentication, Role management , Password hashing , Basic HTTP authentication Token based authentication ,Token based account activation ,Token based password recovery / resetting , User registration, Login tracking , JSON/Ajax Support

### **Flask Admin:**

Getting Started Authorization & Permissions, Customizing Built-in Views Adding Your Own Views Working With the Built-in Templates .

### **REFERENCES:**

#### **MANDATORY:**

1.Miquel Grinberg , “Flask Web Development “, O’REILLY

#### **SUPPLEMENTARY:**

1.Gareth Dwyer “Flask By Example “, Packt Publishing Limited

#### **WEB BASED:**

1. <https://flask.palletsprojects.com/en/2.1.x/>
2. <https://dl.acm.org/doi/10.5555/2621997>

3. <https://www.fullstackpython.com/flask.html>

4.

[https://coddyschool.com/upload/Flask\\_Web\\_Development\\_Developing.pdf](https://coddyschool.com/upload/Flask_Web_Development_Developing.pdf) 5.

<https://www.packtpub.com/product/mastering-flask-web-development-second-edition/9781788995405>

### **Practicals: Web Development with FLASK**

**Credit: 01**

**Marks: 25**

**Duration:30 Hrs**

- 1) Creating a basic application. (1P)
- 2) Working with the requests and jinja. (1P)
- 3) Form creating and validation with WTForms. (1P)
- 4) Creating an admin panel with FLASK Admin. (1P)
- 5) Customizing FLASK admin panel. (1P)
- 6) Connecting database through SQL Alchemy. (1P)
- 7) Insert/Update/Delete/Query data using SQL Alchemy. (4P)
- 8) Securing your application with FLASK Security. (1P)
- 9) Creating a scalable project structure. (1P)
- 10) Build your own API (REST API). (2P)
- 11) Deploying and Configuring FLASK. (1P)