KAMARAJ
COLLEGE OF ENGINEERING & TECHNOLOGY

(An Autonomous Institution - AFFILIATED TO ANNA UNIVERSITY, CHENNAI)
S.P.G.Chidambara Nadar - C.Nagammal Campus
S.P.G.C. Nagar, K.Vellakulam – 625 701 (Near VIRUDHUNAGAR).

B.TECH. INFORMATION TECHNOLOGY

REGULATIONS – 2020

AUTONOMOUS SYLLABUS

CHOICE BASED CREDIT SYSTEM

CURRICULUM AND SYLLABI

(Semester V and VI)

Vision of the Department

To make the department of Information Technology the unique of its kind in the field of Research and Development activities in this part of world

Mission of the Department

To impart highly innovative and technical knowledge in the field of Information Technology to the urban and unreachable rural student folks through Total Quality Education.

Program Educational Objectives (PEOs)

PEO1: Technical Knowledge : Graduates will be able to identify, analyze and create solutions for real life, industrial and societal needs by applying the principles and practices of Information Technology.

PEO2: Teamwork & Ethics : Graduates will be able to collaborate effectively and ethically in a multi-disciplinary team as a member &/ as a leader.

PEO3: Lifelong Learning : Graduates will be able to adopt the contemporary technologies in the field of Information Technology to provide solutions for challenging environments.

Program Specific Objectives (PSOs)

PSO 1: Demonstrate technical and interpersonal skills to design and develop IT enabled solutions to meet the real time industrial and societal needs

PSO2: Exhibit an ability to adapt to the evolutionary changes in computing

SEMESTER V

SL NO	COURSE CODE	COURSE TITLE	CATEGORY	TOTAL CONTACT PERIODS	L	T	P	C
THEORY								
1	IT1571	Computer Networks	PC	3	3	0	0	3
2	IT1501	Design and Analysis of Algorithms	PC	5	3	0	2	4
3	IT1502	Object Oriented Analysis and Design	PC	5	3	0	2	4
4	IT1503	Web Technology	PC	3	3	0	0	3
5	PE1	Professional Elective – I	PE	3	3	0	0	3
6	OE1	Open Elective I	OE	3	3	0	0	3
PRACTICALS								
7	IT1581	Computer Networks Laboratory	PC	4	0	0	4	2
8	IT1511	Web Technology Laboratory	PC	4	0	0	4	2
9	HS1521	Professional Communication	EEC	2	0	0	2	1
				TOTAL CREDITS				**25**

SEMESTER VI

SL NO	COURSE CODE	COURSE TITLE	CATEGORY	TOTAL CONTACT PERIODS	L	T	P	C
THEORY								
1	IT1601	Data Mining and Warehousing	PC	5	3	0	2	4
2	IT1602	Mobile Computing	PC	3	3	0	0	3
3	IT1671	Cryptography and Network Security	PC	3	3	0	0	3
4	PE2	Professional Elective – II	PE	3	3	0	0	3
5	PE3	Professional Elective – III	PE	4	2	0	2	3
ONLINE COURSES								
6	OL1	Online Course I	OL	2	2	0	0	2
AUDIT COURSE								
7	AUD2	Audit Course	AU	3	3	0	0	0
PRACTICALS								
8	IT1681	Cryptography and Network Security Laboratory	PC	4	0	0	4	2
9	CS1681	Mobile Application Development Laboratory	PC	4	0	0	4	2
							TOTAL CREDITS	**22**

SEMESTER V - PROFESSIONAL ELECTIVE – I

SL NO	COURSE CODE	COURSE TITLE	CATEGORY	TOTAL CONTACT PERIODS	L	T	P	C
1	IT1531	Agile Methodologies	PE	3	3	0	0	3
2	IT1532	Internet of Things and its Applications	PE	3	3	0	0	3
3	AD1372	Introduction to Artificial Intelligence	PE	3	3	0	0	3

SEMESTER VI - PROFESSIONAL ELECTIVE – II

SL NO	COURSE CODE	COURSE TITLE	CATEGORY	TOTAL CONTACT PERIODS	L	T	P	C
1	CS1535	Software Testing and Quality Assurance	PE	3	3	0	0	3
2	IT1631	Blockchain Technologies	PE	3	3	0	0	3
3	IT1632	Machine Learning and Deep Learning Techniques	PE	3	3	0	0	3

SEMESTER VI - PROFESSIONAL ELECTIVE – III

SL NO	COURSE CODE	COURSE TITLE	CATEGORY	TOTAL CONTACT PERIODS	L	T	P	C
1	IT1633	Advanced Python Programming	PE	4	2	0	2	3
2	IT1634	Software Defined Networks	PE	4	2	0	2	3
3	CS1631	Big Data Analytics - Tools and Techniques	PE	4	2	0	2	3

OPEN ELECTIVES

Candidates shall register for Open Electives offered by other departments of the Institute in the 5th and 7th semesters. If one or more professional electives (listed above) are offered by other departments, those courses are also registered by the candidates as open electives with the prior approval from HoD of Department of Information Technology and Dean (Academic Courses) taking into account of perquisite requirement.

Suggested List of Open Electives

OPEN ELECTIVE – SEMESTER V

SL NO	COURSE CODE	COURSE TITLE	CATEGORY	TOTAL CONTACT PERIODS	L	T	P	C
1	OIT151	Mobile App Development	OE	3	3	0	0	3
2	OIT152	Principles of Database Management Systems	OE	3	3	0	0	3

SEMESTER V

IT1571 COMPUTER NETWORKS

L	T	P	C
3	0	0	3

OBJECTIVES:

To enable the students to

- Infer the importance of computer networks
- Apply error checking and error correction mechanisms in data Link layer
- Make use of various routing protocols
- Distinguish the functionalities of TCP & UDP protocols in the transport layer
- Summarize the functionalities and real time usage of various application layer protocols

UNIT I PHYSICAL LAYER 8

Introduction: OSI – TCP/IP architectures – Networking: Topology – Types – Devices: Hubs, Bridges, Switches, Routers and Gateways – Transmission Media – Performance: jitter – delay – throughput – bandwidth X delay

UNIT II DATALINK LAYER 10

Layers: MAC and LLC – HDLC - Frame – Types of Frames – Framing - PPP – Services: Error Detection and Correction – Ethernet, IEEE 802.11, Bluetooth, Token Ring, FDDI, Link Layer Addressing: ARP

UNIT III NETWORK LAYER 10

IP address: IPv4 and IPv6 – Subnetting – CIDR - Packet switching – Services: Routing - Intra domain: Distance Vector – RIP, Link State - OSPF, Inter domain - BGP, Multicast, IP, DHCP, ICMP.

UNIT IV TRANSPORT LAYER 8

Services: Flow Control, Congestion Control – QoS - Port Addressing, Protocols: TCP and UDP, SCTP

UNIT V APPLICATION LAYER 9

Protocols: HTTP, DNS, SMTP, MIME, POP, IMAP3, FTP, S-FTP, TELNET, SSH, SNMP

TOTAL: 45 PERIODS

COURSE OUTCOMES:

Upon successful completion of course the students will be able to

CO1: Infer the importance of computer networks with OSI and TCP/IP architectures

CO2: Apply error checking and error correction mechanisms in data Link layer for error free data transmission

CO3: Make use of various routing protocols and their strategies in the network

CO4: Compare the functionalities of TCP & UDP protocols in the transport layer during data transmission

CO5: Summarize the functionalities of various application layer protocols and their real time usage.

TEXT BOOKS:

1. Behrouz A. Forouzan, *Data Communications and Networking*, Fifth Edition TMH, 2013.

REFERENCES:

1. Larry L. Peterson, Bruce S. Davie, "*Computer Networks: A Systems Approach*", Fifth Edition, Morgan Kaufmann Publishers Inc., 2011.
2. William Stallings, "*Data and Computer Communications*", Tenth Edition, Pearson Education, 2014.
3. Ying-Dar Lin, Ren-Hung Hwang, Fred Baker, "*Computer Networks: An Open Source Approach*", McGraw Hill, 2012

		L	T	P	C
IT 1501	**DESIGN AND ANALYSIS OF ALGORTIHMS**	**3**	**0**	**2**	**4**

OBJECTIVES:

To enable the students to

- Apply the knowledge of computing and mathematics to algorithm design
- Explain Brute force and Divide-and-Conquer techniques

- Identify the algorithm efficiency for Greedy and Dynamic programming techniques
- Be familiar with Iterative improvement techniques
- Understand the limitations of Algorithm power

UNIT I INTRODUCTION 9

Notion of an Algorithm - Fundamentals of Algorithmic Problem Solving - Important Problem

Types - Performance analysis - space and time complexity - Growth of function – Big-Oh, Omega, theta notation - Asymptotic Notations and its properties- Recurrent equations and the master theorem - Empirical Analysis - Mathematical analysis for Recursive and Non-recursive algorithms - Visualization

UNIT II BRUTE FORCE AND DIVIDE-AND-CONQUER 9

Brute Force – String Matching - Closest-Pair and Convex-Hull Problems- Exhaustive Search - Traveling Salesman-Problem - Knapsack Problem - Assignment problem-Divide and conquer methodology – Merge sort – Quick sort – Randomized version of quick sort and analysis – Heap Sort - Binary search – Strassen’s matrix multiplication - Closest pair and Convex hull problems

UNIT III GREEDY TECHNIQUE AND DYNAMIC PROGRAMMING 9

Greedy Technique – container loading problem Minimum cost spanning tree Prim’s algorithm Kruskal's Algorithm - Dijkstra's Algorithm- Job sequencing with deadlines - Huffman Trees - Dynamic programming – Principle of optimality – Coin changing problem-Computing a Binomial Coefficient – Warshall’s and Floyd’ algorithm – Optimal Binary Search Trees – 0/1 Knapsack Problem and Memory functions

UNIT IV ITERATIVE IMPROVEMENT 9

The Simplex Method-The Maximum-Flow Problem – Maximum Matching in Bipartite GraphsThe Stable Marriage Problem.

UNIT V COPING WITH THE LIMITATIONS OF ALGORITHM POWER 9

Lower-Bound Arguments-Decision Trees-P, NP and NP-Complete Problems- Coping with the Limitations - Backtracking – n-Queens problem – Hamiltonian Circuit Problem – Subset Sum Problem – Branch and Bound – Assignment

problem – Knapsack Problem – Traveling Salesman Problem- Approximation Algorithms for NP – Hard Problems – Traveling Salesman problem – Knapsack problem

THEORY : 45 PERIODS

LIST OF EXPERIMENTS:

1. Implementation of Knapsack problem using Brute Force technique.
2. Implement merge and quick sort algorithms and compare the time complexity for various of n values.
3. Implementation of Huffman Tree using Greedy technique.
4. Implementation of Warshall's and Floyd's algorithm using Dynamic Programming technique.
5. Implementation of Stable marriage problem using Iterative improvement technique.
6. Implement N-Queens problem using backtracking technique

PRACTICAL : 30 PERIODS

LIST OF EQUIPMENT FOR A BATCH OF 30 STUDENTS:

S. No.	Description of Equipment Quality	Quantity Required
1	Personal Computers (Intel Core i3, HDD 500 GB, 4 GB RAM)	30
2	Printer	1
3	Software: Python 3.6	Open Source

TOTAL : 75 PERIODS

COURSE OUTCOMES:

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

CO1 Identify the time and space complexity of computational problems

CO2 Make use of Bruteforce and Divide and Conquer techniques for sorting and searching Problems

CO3 Apply Greedy and Dynamic Programming techniques for Graph and Combinatorial

Problems

CO4 Identify the roles of iterative improvement technique to solve optimization problems.

CO5 Explain the use of Backtracking, Branch & Bound and approximation techniques to overcome the limitations of NP-Complete and NP-Hard Problems

TEXT BOOKS:

1. Anany Levitin, 2012, *Introduction to the Design and Analysis of Algorithms*, 3rd ed, Pearson Education.

2. Ellis Horowitz, Sartaj Sahni & Sanguthevar Rajasekaran, 2007, *Computer Algorithms/ C++*, 2nd ed, Universities Press.

REFERENCES:

1. Thomas H Cormen, Charles E Leiserson, Ronald L Rivest & Clifford Stein, 2012, *Introduction to Algorithms*, 3rd ed, PHI Learning Private Limited.

2. Alfred V Aho, John E Hopcroft & Jeffrey D Ullman, 2006, *Data Structures and Algorithms*, Pearson Education, Reprint.

IT1502 OBJECT ORIENTED ANALYSIS AND DESIGN

L	T	P	C
3	0	2	4

OBJECTIVES:

To enable the students to:

- Understand the fundamentals of Object Modeling, Unified Process Model
- Design with static with UML diagrams
- Design with UML dynamic and Implementation Diagrams
- Improve the software design with design patterns
- Test the software against its requirements specification

UNIT I INTRODUCTION & CASE STUDY - NEXTGEN POS SYSTEM **15**

Theory Component: **9**

Introduction to OOAD with OO Basics - Unified ProcessModel- Case study - The Next Gen POS system - Inception - Use case Modelling - Relating Use cases – include, extend and generalization

Lab Component: **6**

Suggested Domain for Mini Project

1. E-Passport automation system.
2. Book Shop Management System
3. Alumni Information System
4. Attendance Management System
5. Exam registration System
6. Boutique Store management system.
7. Online course reservation system
8. Airline/Railway reservation system
9. Online Recruitment system
10. Conference management system
11. Library management system
12. Student information system
13. Electricity Bill Payment System
14. Exam Seating Management System
15. Hospital Management System

Case Study:

- Draw standard UML diagrams using an UML modeling tool for a given case study and map design to code and implement a 3 layered architecture.
- Test the developed code and validate whether the SRS is satisfied.

1. Identify a software system that needs to be developed
2. Document the Software Requirements Specification (SRS) for the identified system.
3. Identify use cases and develop the Use Case model.

UNIT II STATIC UML DIAGRAMS **15**

Theory Component: **9**

Class Diagram - Elaboration - Domain Model - Finding conceptual classes and description classes - Associations - Attributes - Domain model refinement - Finding conceptual class Hierarchies - Aggregation and Composition - Relationship between sequence diagrams and use cases

Lab Component: 6

4. Identify the conceptual classes and develop a Domain Model.

5. Derive Class Diagram from the Domain Model.

UNIT III DYNAMIC AND IMPLEMENTATION DIAGRAMS 15

Theory Component: 9

Dynamic Diagrams -UML interaction diagrams -System sequence diagram -Collaboration diagram -State machine diagram and Modelling -Activity diagram - UML package diagram - Component and Deployment Diagrams

Lab Component: 6

6. Using the identified scenarios, find the interaction between objects and represent them using UML Sequence and Collaboration Diagrams

7. Draw relevant State Chart and Activity Diagrams.

8. Draw UML Package, Component and Deployment Diagrams.

UNIT IV Design Patterns and Mapping Design to Code 15

Theory Component 9

GRASP: Designing objects with responsibilities - Creator - Information expert Low Coupling - High Cohesion - Controller Design Patterns - creational - factory method - structural - Bridge - Adapter - behavioural - Strategy – Observer - Applying GoF design patterns - Mapping design to code

Lab Component 6

9. Implement the system as per the detailed design

(i) ER diagrams and Table creation

(ii) Application Logic (Authentication)

(iii) Application Logic (Main Business Logic)

(iv) User Interface Design

(v) Database Connectivity

UNIT IV Object Oriented Testing 15

Theory Component: 9

Impact of object orientation on Testing - Testing: Issues in OO Testing - Class Testing - OO Integration Testing - GUI Testing - OO System Testing. - Develop Test Cases and Test Plans

Lab Component: 6

10. Test cases and Test Suite

- Test the software system for all the scenarios identified as per the use case diagram - (Unit Testing, Integration Testing and System Testing)

11. Improve the reusability and maintainability of the software system by applying appropriate design patterns.

TOTAL: 75 PERIODS

SOFTWARE SPECIFICATION:

1. ArgoUML, Eclipse IDE, Visual Paradigm, Visual case, and Rational Suitesupports UML 1.4 and higher
2. jUnit – for Java unit Test
3. Selenium - is a suite of tools for automating web applications for software testing purposes, plugin for Firefox
4. HP QC - is the HP Web-based test management tool. It familiarizes with the process of defining releases, specifying requirements, planning tests, executing tests, tracking defects, alerting on changes, and analyzing results. It also shows how to customize project
5. IBM Rational - Rational software has a solution to support business sector for designing, implementing and testing software

COURSE OUTCOMES:

Upon successful completion of course the students will be able to

CO1: Construct appropriate Use Case diagrams for real world problem.

CO2: Draw Domain model and Class Diagrams for the chosen software system

CO3: Draw Implementation Diagrams for the given software system

CO4: Illustrate design patterns and develop code for the given software system

CO5: Create various test cases to check the consistency of the developed software system

TEXT BOOKS:

1. Craig Larman, *Applying UML and Patterns: An Introduction to object-oriented Analysis and Design and iterative development*, Third Edition, Pearson Education, 2012.
2. Paul C. Jorgensen, *Software Testing:- A Craftsman"s Approach, Third Edition, Auerbach Publications*, Taylor and Francis Group, 2008.

REFERENCES:

1. Mike O'Docherty, *Object-Oriented Analysis & Design: Understanding System Development with UML 2.0*, John Wiley & Sons, 2005.

2. James W- Cooper, Addison-Wesley, *Java Design Patterns – A Tutorial*, 2000.

3. MichealBlaha, James Rambaugh, *Object-Oriented Modeling and Design with UML*, Second Edition, Prentice Hall of India Private Limited, 2007.

4. Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides, *Design patterns: Elements of Reusable object-oriented software*, Addison-Wesley, 1995.

5. Simon Bennett, Steve Mc Robb and Ray Farmer, *Object Oriented Systems Analysis and Design Using UML*, Fourth Edition, Mc-Graw Hill Education, 2010.

6. Martin Fowler, *UML Distilled: A Brief Guide to the Standard Object Modeling Language*, Third edition, Addison Wesley, 2003.

WEB REFERENCES:

1. http://vlabs.iitkgp.ernet.in/se/ - Virtual Lab at IIT Kharagpur
2. http://codecourse.sourceforge.net/materials/The-Waterfall-Lifecycle-Model.html
3. http://www.agilemodeling.com/
4. http://www.uml.org/
5. http://www.uml-diagrams.org

6. http://www.tutorialspoint.com/uml
7. http://www.gofpatterns.com/
8. http://www.objectmentor.com/resources/articles/umlfsm.pdf
9. http://www.agilemodeling.com/essays/umlDiagrams.html

IT 1503 WEB TECHNOLOGY

L	T	P	C
3	0	0	3

OBJECTIVES:

To enable the students to:

- Understand about client-server communication and creating a static web pages using HTML tags.
- Design interactive web pages using Scripting languages.
- Learn server side programming using servlets and JSP.
- Develop web pages using XML/XSLT.
- Understand AJAX and web services & identify the interactions among web services.

UNIT I WEB SITE BASICS AND HTML 9

Introduction: Clients and Web client, Servers and Web server, Communication **— Basic Internet Protocols:** WWW, HTTP request message and response message – **HTML:** Introduction, **Basic formatting tags:** Grouping using div and span, **List:** Ordered list, Unordered list, Definition list, Image and image mapping, Hyperlinks, Tables, Forms – **HTML5.0:** Semantics elements, Web forms, Canvas API, SVG API, Media (Audio and Video) — XHTML.

UNIT II CSS AND CLIENT SIDE SCRIPTING 9

CSS: Introduction, Features, Core syntax, Style rule cascading and Inheritance, Text properties, Box model normal flow, Box layout **— CSS3.0:** Background image, 2D transformation, 3D transformation, Transition and Animation, Multi-column layout – **Client side scripting:** Introduction, **JavaScript:** History and Versions, Syntax, Variables and Data Types, Statements, Operators, Literals, Functions, Objects, Arrays, Built-in Objects, JavaScript Debuggers.

UNIT III SERVER SIDE SCRIPTING 9

Server-Side Programming: Introduction, **Java Servlets:** Architecture, Overview, Generating Dynamic Content, Life Cycle, Parameter Data, Sessions, Cookies, URL Rewriting, Other Capabilities, Data Storage Servlets and Concurrency, Databases and Java Servlets **— JSP:** JSP Technology Introduction, JSP and Servlets, Running JSP Applications Basic JSP, JavaBeans Classes and JSP, Tag Libraries and Files, Support for the Model-View-Controller Paradigm, Databases and JSP.

UNIT IV DOM AND XML 9

DOM: Browsers and the DOM, Introduction to the Document Object Model (DOM), History and Levels, Intrinsic Event Handling, Modifying Element Style, The Document Tree, DOM Event Handling, Accommodating Noncompliant Browsers Properties of window **— Representing Web Data: XML,** Documents and

Vocabularies, Versions and Declaration, Namespaces, DOM based XML processing, **Event-oriented Parsing:** SAX, Transforming XML Documents, **Selecting XML Data:** XPATH, **Template based Transformations:** XSLT, Displaying XML Documents in Browsers.

UNIT V AJAX AND WEB SERVICES 9

AJAX: Ajax Client Server Architecture, XML Http Request Object, Call Back Methods **— Web Services:** JAX **—** RPC Concepts, Writing a Java Web Service, Writing a Java Web Service Client, **Describing Web Services:** WSDL, UDDI, **Representing Data Types:** XML Schema – **Communicating Object Data:** SOAP Related Technologies, JSON, REST APIs, Software Installation, Storing Java Objects as Files.

TOTAL: 45 PERIODS

COURSE OUTCOMES

Upon successful completion of course the students will be able to

CO1: Identify the appropriate HTML tags for creating static websites in client server communication

CO2: Choose appropriate tags to format and validate the front end of web application

CO3: Make use of sever side scripting and database concepts for creating an interactive web application

CO4: Illustrate the concepts of XML DOM for storing and validating the data

CO5: Make use of AJAX concepts for creating web application and explore JAVA web services.

TEXT BOOKS:

1. Jeffrey C. Jackson, "*Web Technologies--A Computer Science Perspective*", Pearson Education, 2006.

REFERENCES:

1. Robert. W. Sebesta, "*Programming the World Wide Web*", Fourth Edition,Pearson Education, 2007.

2. Deitel, Deitel, Goldberg, "*Internet & World Wide Web How To Program*", ThirdEdition, Pearson Education, 2006.

3. Marty Hall and Larry Brown, —*Core Web Programming*‖*, Second Edition, Volume I and II*, Pearson Education, 2001.

4. Bates, —*Developing Web Applications*‖, Wiley, 2006

IT 1581 COMPUTER NETWORKS LABORATORY

L	T	P	C
0	0	4	2

OBJECTIVES:

To enables the students to

- Learn and use network commands.
- Understand socket programming.
- Familiarize with the implementation and analysis of various network protocols.
- Make use of simulation tools to analyze the performance of various network protocols.

LIST OF EXPERIMENTS:

1. Learn to use commands like tcpdump, netstat, ifconfig, nslookup and traceroute. Capture ping and traceroute PDUs using wireshark and examine.
2. Simulate bit stuffing concept.

3. Simulate error correction mechanism (like CRC).
4. Simulation of Distance Vector/ Link State Routing algorithm.
5. Performance evaluation of Routing protocols using Simulation tool.
6. Implementation of subnetting concepts.
7. Write a HTTP web client program to download a web page using TCP sockets.
8. Applications using TCP sockets like:
    - Echo client and echo server
    - Chat
    - File Transfer
9. Simulation of DNS using UDP sockets.
10. Write suitable codes for simulating ARP /RARP protocols.
11. Simulation of Congestion Control Algorithms using NS.
12. Performance Evaluation of TCP/UDP using Simulation tool.
13. Working with RMI using Java.

TOTAL: 60 PERIODS

SOFTWARE SPECIFICATIONS:

For capturing packets : Wireshark

Simulation Tools Preferred : NS-2, CISCO Packet Tracer, or equivalent

Language preferred : C, Java, Python

COURSE OUTCOMES

Upon successful completion of course the students will be able to

CO1: Make use of basic networking commands for capturing packets in live networks

CO2: Implement bit stuffing and error correction algorithms in a client/server environment.

CO3: Analyse the performance of network routing protocols and transport layer services through simulation tools.

CO4: Simulate the application protocols using TCP and UDP.

CO5: Build an RMI server/client for a real time application.

REFERENCES:

1. Larry L. Peterson, Bruce S. Davie, “Computer Networks: A Systems Approach”, Fifth Edition, Morgan Kaufmann Publishers Inc., 2011.
2. Behrouz A. Forouzan, Data Communications and Networking, Fifth Edition TMH, 2013.
3. https://ns2projects.org/ns2-tutorial-for-beginners/
4. https://www.netacad.com/courses/packet-tracer/introduction-packet-tracer

IT 1511 WEB TECHNOLOGY LABORATORY

L	T	P	C
0	0	4	2

OBJECTIVES:

To enable the students to,

- Use HTML and CSS for creating a simple web application
- Develop interactive web pages using Scripting languages
- Learn server side programming using servlets and JSP
- Develop web pages using XML/XSLT
- Understand the concept of AJAX and Web Services

LIST OF EXPERIMENTS:

1. **Create a responsive web page with the following using HTML.**
    a. To embed an image map in a web page.
    b. To fix the hot spots.
    c. Show all the related information when the hot spots are clicked
2. Create a web page with all types of Cascading Style Sheets.
3. Client side scripts for validating web form controls using DHTML.
4. **Write programs in Java to create three-tier applications using Servlet and Databases**
    a. For conducting on-line examination.
    b. Session Tracking.
5. Write programs in Java to create three-tier applications using JSP and Databases

a. For displaying student mark list. Assume that student information is available in a database which has been stored in a database server.

6. Implementation of EJB
7. Programs using DOM and SAX parsers.
8. Programs Using Xml – Schema – XSLT/XSL.
9. Programs using AJAX.
10. Consider a case where there are two web Services – an airline service and a travel agent and the travel agent is searching for an airline. Implement this scenario using Web Services and Data base.
11. Mini Project

TOTAL: 60 PERIODS

SOFTWARE SPECIFICATIONS:

1. Dream Weaver or Equivalent
2. MySQL or Equivalent
3. WAMP/XAMPP

COURSE OUTCOMES

Upon successful completion of course the students will be able to

CO1: Apply HTML and CSS Technologies for creating interactive webpage

CO2: Develop a dynamic web application using DHTML and JavaScript

CO3: Make Use of server side scripting like servlets and JSP to implement three tier web applications

CO4: Utilize XML Technologies for storing and retrieving data

CO5: Experiment with web services and AJAX concepts for developing real time applications

REFERENCES:

Kogent Learning Solutions Inc. , "*Web Technologies: HTML, JAVASCRIPT, PHP, JAVA, JSP, ASP.NET, XML and Ajax, Black Book*", Dreamtech Press, 2009.

1. https://www.w3schools.com/html/
2. https://www.w3schools.com/css/

3. https://www.w3schools.com/js/
4. https://www.javatpoint.com/servlet-tutorial
5. https://www.javatpoint.com/jsp-tutorial
6. https://www.javatpoint.com/ejb-tutorial
7. https://www.javatpoint.com/xml-tutorial
8. https://www.w3schools.com/xml/ajax_intro.asp
9. https://www.javatpoint.com/web-services-tutorial

HS1521 PROFESSIONAL COMMUNICATION

L	T	P	C
0	0	2	1

OBJECTIVES:

The course aims to

- Enhance the Employability and Career Skills of students
- Orient the students towards grooming as a professional
- Make them Employable Graduates
- Develop their confidence and help them attend interviews successfully

UNIT I

Introduction to Soft Skills– Hard skills & soft skills – employability and career Skills— Grooming as a professional with values—Time Management—General awareness of Current Affairs

UNIT II

Self-Introduction-organizing the material – Introducing oneself to the audience – introducing the topic – answering questions with clarity and appropriate pharases – individual presentation practice— presenting the visuals effectively – 5 minute presentations

UNIT III

Introduction to Group Discussion— Participating in group discussions – understanding group dynamics – brainstorming the topic -– questioning and clarifying –GD strategies- activities to improve GD skills

UNIT IV

Interview etiquette – dress code – body language – attending job interviews– telephone/skype interview -one to one interview &panel interview – FAQs related to jobinterviews

UNIT V

Recognizing differences between groups and teams- managing time-managing stress- networking professionally- respecting social protocols-understanding career management-developing a long-term career plan-making career changes

TOTAL: 30 PERIODS

Available Software:

1. Odyll

OUTCOMES:

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

- Make effective presentations
- Participate confidently in Group Discussions.
- Attend job interviews and be successful in them.
- Develop adequate Soft Skills required for the workplace

REFERENCES:

1. Butterfield, *Jeff Soft Skills for Everyone*. Cengage Learning: New Delhi, 2015
2. E. Suresh Kumar et al. *Communication for Professional Success*. Orient Blackswan:Hyderabad, 2015
3. *Interact English Lab Manual for Undergraduate Students*,. OrientBalckSwan: Hyderabad,2016.
4. Raman, Meenakshi and Sangeeta Sharma. *Professional Communication*. OxfordUniversity Press: Oxford, 2014
5. S. Hariharanet al. Soft Skills. MJP Publishers: Chennai, 2010.

SEMESTER VI

IT1601 DATA MINING AND WAREHOUSING

L	T	P	C
3	0	2	3

OBJECTIVES:

To enable the students to:

- Be familiar with the concepts of data warehouse,
- Be acquainted with the tools and techniques used for knowledge discovery indatabases.
- Apply data mining and association rule mining techniques
- Utilize classification and prediction algorithms
- Choose clustering algorithms for real time applications.

UNIT I DATA WAREHOUSING 9 + 2 - 11

Data Warehousing Components –Building a Data warehouse –- Mapping the Data Warehouse to a Multiprocessor Architecture – DBMS Schemas for Decision Support – Data Extraction, Cleanup, and Transformation Tools –Metadata.

Lab Exercises:

I) Write ETL scripts and implement using data warehouse tools

UNIT II BUSINESS ANALYSIS 9 + 0 – 9

Reporting and Query tools and Applications – Tool Categories – The Need for Applications – Cognos Impromptu – Online Analytical Processing (OLAP) – Need – Multidimensional Data Model – OLAP Guidelines – Multidimensional versus Multirelational OLAP – Categories of Tools – OLAP Tools and the Internet.

UNIT III DATA MINING & ASSOCIATION RULE MINING 11+ 8 - 19

Introduction – Data – Types of Data – Data Mining Functionalities – Interestingness of Patterns – Classification of Data Mining Systems – Data Mining Task Primitives – Integration of a Data Mining System with a Data Warehouse – Issues –Data Preprocessing – Mining Frequent Patterns, Associations and Correlations – Mining Methods – Mining various Kinds of Association Rules – Correlation Analysis – Constraint Based Association Mining

Lab Exercises:

II) Load each dataset into Weka and run Apriori algorithm with different support and

confidence values.

III) FP Growth Algorithm

UNIT IV CLASSIFICATION AND PREDICTION 7+12 - 19

Classification and Prediction - Basic Concepts - Decision Tree Induction - Bayesian Classification – Rule Based Classification – Classification by Back propagation – Support Vector Machines – Associative Classification – Lazy Learners – Other Classification Methods – Prediction.

Lab Exercises:

IV) Naïve-bayes classification.

V) Support Vector Machines

VI) Decision Tree

UNIT V CLUSTERING AND TRENDS IN DATA MINING 9+ 8 - 17

Cluster Analysis - Types of Data – Categorization of Major Clustering Methods – K-means– Partitioning Methods – Hierarchical Methods - Density-Based Methods –Grid Based Methods – Model-Based Clustering Methods – Clustering High Dimensional Data - Constraint – Based Cluster Analysis – Outlier Analysis – Data Mining Applications

Lab Exercises

VII) K- Means Partitioning

VIII) Hierarchical Clustering

IX) Web Mining

X) Text Mining

TOTAL: 75 PERIODS

COURSE OUTCOMES

Upon successful completion of course the students will be able to

CO1 : Infer the various Data warehousing Components.

CO2 : Explain operations of OLAP tools involved in Business Analysis

CO3 : Make use of rule mining to discover patterns

CO4 : Experiment with classification and prediction algorithms.

CO5 : Select the appropriate clustering algorithm for real word use cases.

TEXT BOOKS:

1. Alex Berson& Stephen J.Smith, 2008, *Data Warehousing, Data Mining and*

OLAP, Tata McGraw – Hill Edition.

2. Jiawei Han & Micheline Kamber, 2012, *Data Mining Concepts and Techniques*, 3rd ed, Elsevier.

REFERENCES:

1. Pang-Ning Tan, Michael Steinbach & Vipin Kumar, 2007, *Introduction to Data Mining*, Person Education.
2. K.P. Soman, Shyam Diwakar & V. Aja, 2006, *Insight into Data Mining Theory and Practice*, Eastern Economy Edition, Prentice Hall of India.
3. G. K. Gupta, 2006, *Introduction to Data Mining with Case Studies*, Eastern Economy Edition, Prentice Hall of India.
4. Daniel T.Larose, 2006, *Data Mining Methods and Models*, Wiley-Interscience.

IT 1602 MOBILE COMPUTING

L	T	P	C
3	0	0	3

OBJECTIVES:

The student should be made to:

- Understand the basic concepts of mobile computing.
- Be familiar with the network protocol stack.
- Learn the basics of mobile telecommunication system.
- Be exposed to Ad-Hoc networks.
- Gain knowledge about different mobile platforms and application development

UNIT I INTRODUCTION 9

Mobile Computing – Mobile Computing Vs wireless Networking – Mobile Computing Applications – Characteristics of Mobile computing – Structure of Mobile Computing Application. MAC Protocols – Wireless MAC Issues – Fixed Assignment Schemes – Random Assignment Schemes – Reservation Based Schemes.

UNIT II MOBILE INTERNET PROTOCOL AND TRANSPORT LAYER 9

Overview of Mobile IP – Features of Mobile IP – Key Mechanism in Mobile IP – route Optimization. Overview of TCP/IP – Architecture of TCP/IP- Adaptation of tCP Window – Improvement in TCP Performance.

UNIT III MOBILE TELECOMMUNICATION SYSTEM 9

Global System for Mobile Communication (GSM) – General Packet Radio Service (GPRS) –

Universal Mobile Telecommunication System (UMTS).

UNIT IV MOBILE AD-HOC NETWORKS 9

Ad-Hoc Basic Concepts – Characteristics – Applications – Design Issues – Routing – Essential of Traditional Routing Protocols –Popular Routing Protocols – Vehicular Ad Hoc networks (VANET) – MANET Vs VANET – Security.

UNIT V MOBILE PLATFORMS AND APPLICATIONS 9

Mobile Device Operating Systems – Special Constrains & Requirements – Commercial Mobile Operating Systems – Software Development Kit: iOS, Android, BlackBerry, Windows Phone – MCommerce – Structure – Pros & Cons – Mobile Payment System – Security Issues.

TOTAL: 45 PERIODS

OUTCOMES:

At the end of the course, the student should be able to:

- Explain the basics of mobile telecommunication system
- Choose the required functionality at each layer for given application
- Identify solution for each functionality at each layer
- Use simulator tools and design Ad hoc networks
- Develop a mobile application.

TEXT BOOK:

1. Prasant Kumar Pattnaik, Rajib Mall, “*Fundamentals of Mobile Computing*”, PHI Learning Pvt. Ltd, New Delhi – 2012.

REFERENCES:

1. Jochen H. Schller, “*Mobile Communications*”, Second Edition, Pearson Education, New Delhi, 2007.

2. Dharma Prakash Agarval, Qing and An Zeng, "*Introduction to Wireless and Mobile systems*", Thomson Asia Pvt Ltd, 2005.

3. Uwe Hansmann, Lothar Merk, Martin S. Nicklons and Thomas Stober, "*Principles of Mobile Computing*", Springer, 2003.

4. William.C.Y.Lee, "*Mobile Cellular Telecommunications-Analog and Digital Systems*", Second Edition, Tata Mc Graw Hill Edition ,2006.

5. C.K.Toh, "*AdHoc Mobile Wireless Networks*", First Edition, Pearson Education, 2002.

6. Android Developers : http://developer.android.com/index.html

7. Apple Developer : https://developer.apple.com/

8. Windows Phone Dev Center : http://developer.windowsphone.com

9. BlackBerry Developer : http://developer.blackberry.com/

IT 1671 CRYPTOGRPAHY AND NETWORK SECURITY

L	T	P	C
3	0	0	3

OBJECTIVES:

To enable the students to

- learn the fundamental concepts and techniques in cryptography
- illustrate the symmetric cryptosystem with respect to block ciphers and stream ciphers
- explore the Asymmetric cryptosystems
- understand the various message authentication principals and data integrity algorithms
- acquire background knowledge of various security practices and system level security

UNIT I CLASSICAL CRYPTOSYSTEM 9

OSI security architecture: Security services, Security mechanism, Security attacks

– Network Security model **— Introduction to cryptosystem:** Symmetric cryptosystem, Asymmetric cryptosystem – Perfect Secrecy – **Classical symmetric techniques: Substitution**: Caesar cipher, Shift cipher, Playfair cipher, Hill cipher, Vigenere cipher, Autokey cipher, One time pad, **Transposition**: Rail fence, Row columnar transposition – Introduction to

modern cryptosystem.

UNIT II MODERN SYMMETRIC CRYPTOSYSTEM 9

Mathematics for Symmetric model: Algebraic structures, Modular arithmetic, Congruence and matrices, Groups, Rings, Fields, Finite fields, Euclid's algorithm – **Mode of operations:** Electronics codebook, Cipher block chaining, Cipher feedback, Output feedback, Counter – Differential and linear cryptanalysis – **Block Ciphers:** Block cipher design principles, Standard Data Encryption Standard (SDES), Data Encryption Standard (DES), Advanced Encryption Standard (AES) **– Stream ciphers:** Stream cipher design principles, RC4, IDEA.

UNIT III ASYMMETRIC CRYPTOSYSTEM 9

Mathematics for asymmetric model: Primes, Primality Testing, Factorization, Euler_s totient function, Fermat_s and Euler_s Theorem, Chinese Remainder Theorem, Exponentiation and logarithm – **Asymmetric cryptosystem:** RSA – Key distribution – Diffie Hellman – ElGammal cryptosystem – Elliptic curve cryptosystem.

UNIT IV INTEGRITY AND MESSAGE AUTHENTICATION 9

Authentication requirements – Authentication function – Message Authentication Code (MAC) – Hash function – Secure Hash Algorithm (SHA) – HMAC and CMAC – Digital signature – DSS – **Entity authentication:** Biometrics, Passwords, Challenge Response protocols – Kerberos – X.509.

UNIT V SYSTEM SECURITY AND SECURITY PRACTICES 9

Intrusion Detection System (IDS) – **Malicious software:** Trapdoor, Trojan horses, Logic bombs, Viruses, Worms, etc. – **Firewalls:** Types of firewall, Firewall configuration – **E-mail security:** PGP, S/MIME – **IP security:** AH, ESP, IKE – **Websecurity:** SSL, TLS, SET.

TOTAL: 45 PERIODS

COURSE OUTCOMES

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

CO1	Select an appropriate classical symmetric cryptosystem to provide data security

CO2	Apply the mathematical concepts for symmetric block ciphers and stream ciphers
CO3	Choose an appropriate asymmetric cryptosystem and key management to ensure a secure transmission for a real world scenario
CO4	Utilize the hash functions and digital signatures to provide authentication and integrity to a cryptosystem
CO5	Discuss various real time practices that provide Email security, network security and system security

TEXT BOOKS:

1. William Stallings, —*Cryptography and network Security*‖, 7th edition Pearson,2017.

2. BehrouzA.Foruzan, *Cryptography and Network Security*, Tata McGraw Hill2007.

3. C K Shyamala, N Harini and Dr. T R Padmanabhan: *Cryptography and Network Security*, Wiley India Pvt. Ltd.

REFERENCES:

1. Wade Trappe, Lawrence C Washington, —*Introduction to Cryptography with coding theory*‖, 2nd edition, Pearson, 2007.

2. Charles P. Pfleeger, Shari Lawrence Pfleeger, —*Security in computing*‖, Prentice Hall of India, 3rd Edition, 2006

3. Charlie Kaufman, Radia Perlman, and Mike Speciner, *Network Security: PRIVATE Communication in a PUBLIC World*, Prentice Hall, ISBN 0-13- 046019-2

IT1681 CRYPTOGRAPHY AND NETWORK SECURITY LABORATORY

L	T	P	C
0	0	4	2

OBJECTIVES:

This course enables the students to

- implement various classical symmetric cryptosystem
- get familiarized with different modern symmetric and asymmetric cryptosystems
- utilize the key exchange algorithm to secure the private key
- build data integrity algorithms to provide data integrity and authentication
- make use of various open source tools like GnuPG, KF sensor, Snort, etc.

LIST OF EXPERIMENTS:

1. **Implementation of Classical Substitution Crypto Algorithms to provide Confidentiality**
    a. Implementation of Caesar Cipher and Shift Cipher
    b. Implementation of Playfair Cipher
    c. Implementation of Vigenere Cipher
    d. Implementation of Hill Cipher
2. **Implementation of Classical Transposition Crypto Algorithms to provide Confidentiality**
    a. Implementation of Row - Columnar Transformation
    b. Implementation of Rail fence
3. **Implementation of Modern Private Key Cryptosystem Algorithms to provide Confidentiality**
    a. Apply DES algorithm for practical applications
    b. Apply AES algorithm for practical applications
4. **Implementation of Public Key Cryptosystem to provide Confidentiality and Authentication**
    a. Implement RSA Algorithm
5. **Implement the Diffie-Hellman Key Exchange algorithm**
6. **Implementation of Cryptographic Algorithms to provide Data Integrity and Authentication**
    a. Calculate the message digest of a text using the SHA algorithm.
7. **Implement the SIGNATURE SCHEME - Digital Signature Standard.**
8. **Demonstration on Usage of Modern Tools in Securing a System'**

a. Demonstrate how to provide secure data storage, secure data transmission and for creating digital signatures (GnuPG)
b. Demonstrate intrusion detection system (ids) using any tool eg. Snort or any other s/w.
c. Setup a honey pot and monitor the honeypot on network (KF Sensor)
d. Automated Attack and Penetration Tools
e. Defeating Malware i) Building Trojans ii) Rootkit Hunter

TOTAL: 60 PERIODS

SOFTWARE SPECIFICATIONS:

1. C / Java / Python

COURSE OUTCOMES

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

CO1	Choose appropriate classical symmetric cryptosystem to provide data security.
CO2	Make use of modern symmetric and asymmetric cryptosystem to enhance data security.
CO3	Apply the key exchange algorithm to securely exchange symmetric keys.
CO4	Select an appropriate hash algorithm and digital signature to provide integrity and authentication.
CO5	Examine the security of the network system using open source tools.

REFERENCES:

1. Bruce Schneier, "*Applied Cryptography: Protocols, Algorithms, and Source Code in C*", Wiley India Pvt. Ltd. https://mrajacse.files.wordpress.com/2012/01/applied-cryptography-2nd-ed-b-schneier.pdf
2. Jonathan B. Knudsen, "*Java Cryptography*", O'Reilly Media, Inc. https://informatika.stei.itb.ac.id/~rinaldi.munir/Kriptografi/2010-2011/Java%20Cryptography.pdf

3. Seth James Nielson, Christopher K. Monson, "*Practical Cryptography in Python: Learning Correct Cryptography by Example*", Apress.
4. http://www.keyfocus.net/kfsensor/
5. https://www.snort.org/
6. https://www.gpg4win.org/download.html

CS1681 MOBILE APPLICATION DEVELOPMENT LABORATORY

L	T	P	C
0	0	4	2

LIST OF EXPERIMENTS:

1. Develop an application that uses GUI components, Styles
2. Develop an application that uses Layout Managers and event listeners.
3. Develop an application to draw basic graphical primitives on the screen.
4. Create an Animation using Multithreading
5. Develop an application that makes use of databases. (SQLite)
6. Implement an application that writes data to the SD card.
7. Develop an application that makes use of Notification Manager
8. Develop a native application that uses GPS location information
9. Implement an application that creates an alert upon receiving a message
10. Develop a mobile application to send an email.
11. Develop a mobile application with regional language support
12. Develop a mobile application to record audio
13. Develop a mobile application for sensing temperature using IOT
14. Develop a mobile application to retrieve and process data from cloud storage
15. Develop a Mini Project

TOTAL: 60 PERIODS

TEXT BOOK:

1. John Horton, "*Android Programming for Beginners*", Second Edition, Packt Publishing,2018.

REFERENCES:

1. Pradeep Kothari, Kogent Learning Solutions Inc.,‖*Android Application Development (With Kitkat Support), Black Book*‖, First Edition, Kindle Edition, 2014.

PROFESSIONAL ELECTIVES

IT1531 AGILE METHODOLOGIES

L	T	P	C
3	0	0	3

OBJECTIVES:

To enable the students to:

- provide students with a theoretical as well as practical understanding of agile software development practices and how small teams can apply them to create high-quality software.
- To provide a good understanding of software design and a set of software technologies and APIs.
- do a detailed examination and demonstration of Agile development and testing techniques.
- understand the benefits and pitfalls of working in an Agile team.
- understand Agile development and testing.

UNIT I AGILE METHODOLOGY 9

Theories for Agile Management - Agile Software Development - Traditional Model vs. Agile Model - Classification of Agile Methods - Agile Manifesto and Principles - Agile Project Management - Agile Team Interactions - Ethics in Agile Teams - Agility in Design, Testing - Agile Documentations - Agile Drivers, Capabilities and Values

UNIT II AGILE PROCESSES 9

Lean Production: SCRUM, Crystal, Feature Driven Development - Adaptive Software Development - Extreme Programming: Method Overview - Lifecycle - Work Products, Roles and Practices.

UNIT III AGILITY AND KNOWLEDGE MANAGEMENT 9

Agile Information Systems - Agile Decision Making - Earl_S Schools of KM - Institutional Knowledge Evolution Cycle - Development, Acquisition, Refinement, Distribution, Deployment , Leveraging - KM in Software Engineering - Managing Software Knowledge - Challenges of Migrating to Agile Methodologies - Agile Knowledge Sharing - Role of Story-Cards - Story-Card Maturity Model (SMM).

UNIT IV AGILITY AND REQUIREMENTS ENGINEERING 9

Impact of Agile Processes in RE-Current Agile Practices - Variance - Overview of RE Using Agile - Managing Unstable Requirements - Requirements Elicitation - Agile Requirements Abstraction Model - Requirements Management in Agile Environment, Agile Requirements Prioritization - Agile Requirements Modeling and Generation - Concurrency in Agile Requirements Generation.

UNIT V AGILITY AND QUALITY ASSURANCE 9

Agile Product Development - Agile Metrics - Feature Driven Development (FDD) - Financial and Production Metrics in FDD - Agile Approach to Quality Assurance - Test Driven Development - Agile Approach in Global Software Development.

TOTAL: 45 PERIODS

COURSE OUTCOMES

Upon successful completion of course the students will be able to

CO1: Realize the importance of agile software development practices and how teams can apply them to create high-quality software.

CO2: Perform iterative software development processes.

CO3: Understand the importance of knowledge management.

CO4: Perform software process improvement in determining the requirements for a software system.

CO5: Show how agile approaches can be scaled up to the enterprise level.

TEXT BOOKS:

1. David J. Anderson and Eli Schragenheim, - *Agile Management for Software Engineering: Applying the Theory of Constraints for Business Results*‖, Prentice Hall, 2003.

2. Hazza and Dubinsky, - *Agile Software Engineering, Series: Undergraduate Topics in Computer Science*‖, Springer, 2009.

REFERENCES:

1. Craig Larman, - *Agile and Iterative Development: A Manager's Guide*, Addison-Wesley, 2004.

2. Kevin C. Desouza, - *Agile Information Systems: Conceptualization, Construction, and Management*, Butterworth-Heinemann, 2007.

IT1532 INTERNET OF THINGS AND ITS APPLICATIONS

L	T	P	C
3	0	0	3

OBJECTIVES

To enable the students to

- understand IoT architectures and smart objects for real world application
- learn about various IOT-related protocols
- build simple IoT Systems using Arduino and Raspberry Pi
- understand data analytics and cloud related toIoT
- develop IoT infrastructure for real time applications

UNIT I INTRODUCTION TO IoT 9

Evolution of Internet of Things - IoT Enabling Technologies - IoT Levels - IoT Architectures - OneM2M, IoT World Forum (IoTWF)- Simplified IoT Architecture and Core IoT Functional Stack - Fog, Edge and Cloud in IoT - Functional Blocks of an IoT EcosystemSensors, Actuators, Smart Objects and Connecting Smart Objects

UNIT II IoT PROTOCOLS 9

IoT Access Technologies: Physical and MAC Layers, Topology and Security of IEEE 802.15.4, 802.15.4g, 802.15.4e,802.11ah and LoRaWAN Network Layer: IP Versions, Constrained Nodes and Constrained Networks - Application Transport Methods: Supervisory Control and Data Acquisition - Application Layer Protocols: CoAP and MQTT

UNIT III DESIGN AND DEVELOPMENT WITH ARDUINO,RASPBERRY PI 9

Design Methodology - Embedded Computing Logic - Microcontroller, System on Chips - IoT System Building Blocks - Arduino - Board Details, IDE Programming - Logical Design using Raspberry PI

UNIT IV DATA ANALYTICS AND SUPPORTING SERVICES 9

Structured Vs Unstructured Data and Data in Motion Vs Data in Rest - Role of Machine Learning - No SQL Databases - Hadoop Ecosystem - Apache Kafka, Apache Spark - Edge Streaming Analytics and Network Analytics -Xively Cloud for IoT, Python Web Application Framework - Django - AWS for IoT

UNIT V CASE STUDIES AND REAL-WORLD APPLICATIONS 9

IoT Cloud Storage Models & Communication APIs - Cloud for IoT - Power Utility Industry - Smart Grid - Smart Agriculture - Smart and Connected Cities: Smart Lighting, Smart Parking, Smart Traffic Control and Commercial Building Automation

TOTAL: 45 PERIODS

Course Outcomes

Upon successful completion of course the students will be able to

CO1 Explain IoT architectures and its elements.

CO2 Summarize the various protocols for IoT.

CO3 Build simple IoT system using Raspberry Pi/Arduino.

CO4 Outline the concepts of data analytics and cloud for IoT applications

CO5 Relate the applications of IoT in real time scenario.

TEXTBOOK:

1. David Hanes, Gonzalo Salgueiro, Patrick Grossetete, Rob Barton and Jerome Henry, - IoT Fundamentals: Networking Technologies, Protocols and Use Cases for Internet of Things,Cisco Press, 2017

REFERENCES:

1. ArshdeepBahga, Vijay Madisetti, - *Internet of Things - A hands-on approach*, Universities Press, 2015

2. Olivier Hersent, David Boswarthick, Omar Elloumi, - *The Internet of Things - Key Applications and Protocols*, Wiley, 2012 (for Unit 2).

3. Jan Ho¨ ller, VlasiosTsiatsis , Catherine Mulligan, Stamatis , Karnouskos, Stefan Avesand.David Boyle, "*From Machine-to-Machine to the Internet of Things - Introduction to a New Age of Intelligence*", Elsevier, 2014.

4. Dieter Uckelmann, Mark Harrison, Michahelles, Florian (Eds), - *Architecting the Internet of Things*, Springer, 2011.

5. Michael Margolis, *Arduino Cookbook, Recipes to Begin, Expand, and Enhance Your Projects*, 2nd Edition, O'Reilly Media, 2011.

6.https://www.arduino.cc/

https://www.ibm.com/smarterplanet/us/en/?ca=v_smarterplanet

AD1372 INTRODUCTION TO ARTIFICIAL INTELLIGENCE

L T P C

3 0 0 3

OBJECTIVES

To enable the students to

- understand the various characteristics of Intelligent agents
- learn the different search strategies in Artificial Intelligence
- learn to represent knowledge in solving Artificial Intelligence problems
- understand the agent communication and Trust and Reputation
- know about the various applications of Artificial Intelligence**.**

UNIT I INTRODUCTION 9

Introduction–Definition – Future of Artificial Intelligence – Characteristics of Intelligent Agents–Typical Intelligent Agents – Problem Solving Approach to Typical AI problems.

UNIT II PROBLEM SOLVING METHODS 9

Problem solving Methods – Search Strategies- Uninformed – Informed – Heuristics – Local Search Algorithms and Optimization Problems – Searching with Partial Observations – Constraint Satisfaction Problems – Constraint

Propagation – Backtracking Search – Game Playing – Optimal Decisions in Games – Alpha – Beta Pruning – Stochastic Games

UNIT III KNOWLEDGE REPRESENTATION AND REASONING 9

First Order Predicate Logic – Prolog Programming – Unification – Forward Chaining-Backward Chaining – Resolution – Knowledge Representation – Ontological Engineering-Categories and Objects – Events – Mental Events and Mental Objects – Reasoning Systems for Categories –Reasoning with Default Information

UNIT IV SOFTWARE AGENTS 9

Architecture for Intelligent Agents – Agent communication – Negotiation and Bargaining –Argumentation among Agents – Trust and Reputation in Multi-agent systems**.**

UNIT V APPLICATIONS 9

AI applications – Language Models – Information Retrieval- Information Extraction – Natural Language Processing – Machine Translation – Speech Recognition – Robot – Hardware –Perception – Planning – Moving

TOTAL: 45 PERIODS

COURSE OUTCOME

Upon successful completion of course the students will be able to

CO1 Explain the various characteristics of Intelligent agents

CO2 Interpret appropriate search algorithms for Artificial Intelligence problem

CO3 Illustrate a Knowledge Representation using first order logic

CO4 Infer different ways of the agent communication and Trust and Reputation in Multi-agent systems

CO5 Summarize the various application of AI

TEXTBOOKS :

1. S. Russell and P. Norvig, "*Artificial Intelligence: A Modern Approach*‖, Prentice Hall, Third Edition, 2009
2. I. Bratko, ―*Prolog: Programming for Artificial Intelligence*‖, Fourth edition, Addison-Wesley Educational Publishers Inc., 2011.

REFERENCES :

1. M. Tim Jones, ―*Artificial Intelligence: A Systems Approach(Computer Science)*‖, Jones and Bartlett Publishers, Inc.; First Edition, 2008
2. Nils J. Nilsson, ―*The Quest for Artificial Intelligence*‖, Cambridge University Press, 2009
3. William F. Clocksin and Christopher S. Mellish,‖ *Programming in Prolog: Using the ISO Standard*‖, Fifth Edition, Springer, 2003
4. Gerhard Weiss, ―*Multi Agent Systems*‖, Second Edition, MIT Press, 2013.
5. David L. Poole and Alan K. Mackworth, ―*Artificial Intelligence: Foundations of Computational Agents*‖, Cambridge University Press, 2010.

CS1535 SOFTWARE TESTING AND QUALITY ASSURANCE

L	T	P	C
3	0	0	3

OBJECTIVES:

To enable the students to:

- Understand the need of testing, its principles and the origin of defects
- Learn the importance and types of black box and white box test case design strategies
- Be exposed with different levels and types of testing
- Understand the basic tenets of software quality factors, SQA architecture and SQA components
- Be exposed to how the SQA components can be integrated into the project life cycle.

UNIT I INTRODUCTION TO TESTING 9

Testing as an Engineering Activity - Testing Maturity Model - Basic Definitions - Software Testing Principles - Origins of Defects - Defect Classes, Defect Repository and Test Design: Case Study Example: Coin Problem - Introducing the Test Specialist: Skills Needed by a Test Specialist, Building a Testing Group - The Structure of Testing Group

UNIT II TEST CASE DESIGN 9

Test Case Design Strategies - Black Box Approach: Boundary Value Analysis, Equivalence Class Partitioning, State Based Testing, Cause-Effect Graph, Domain Testing, Requirements Based Testing - White Box Approach: Static Testing vs Structural Testing, Code Functional Testing, Coverage and Control Flow Graphs, Code Complexity Testing - Test Adequacy Criteria.

UNIT III LEVELS OF TESTING 9

The Need for Levels of Testing - Unit Testing: Unit Test Planning, Designing the Unit Test, The Test Harness, Running the Unit Tests and Recording Results - Integration Testing Designing Integration Test, Integration Test Planning, Scenario Testing, Defect Bash Elimination - System Testing: Performance Testing, Configuration Testing, Compatibility Testing, Usability and Accessibility Testing, Regression Testing, User Documentation Testing - Acceptance Testing: Alpha and Beta Testing.

UNIT IV INTRODUCTION TO SOFTWARE QUALITY & ARCHITECTURE 9

Need for Software Quality - Quality Challenges - Software Quality Assurance (SQA): Definition and Objectives - Software Quality Factors: McCall's Quality Model - SQA System and Architecture - Software Project Life Cycle Components - Pre Project Quality Components: Development and Quality Plans.

UNIT V SQA COMPONENTS AND PROJECT LIFE CYCLE 9

Integrating Quality Activities in the Project Life Cycle: Software Development Methodologies - Quality Assurance Activities in the Development Process - Verification and Validation - Reviews - Quality of Software Maintenance - Pre-Maintenance Software Quality Components - Maintenance Software Quality Assurance Tools - CASE Tool: Software Product Quality, Software Maintenance Quality, Project Management.

TOTAL: 45 PERIODS

COURSE OUTCOMES

Upon successful completion of course the students will be able to

CO1: Explain the importance of software testing, principles, origins and types of defects.

CO2: Summarize the relationship between white box and black box test case design strategies for valid and invalid input conditions

CO3: Describe the various levels of testing to be carried out in software development life cycle. CO4: Explain the importance of software quality factors, SQA architecture and SQA components.

CO5: Describe how the SQA components can be integrated into the project life cycle.

TEXT BOOKS:

1. Ilene Burnstein, "*Practical Software Testing*", Springer International Edition, 2003.
2. Daniel Galin, "*Software Quality Assurance*", Pearson Publication, 2009.

REFERENCES:

1. Srinivasan Desikan and Gopalaswamy Ramesh, "*Software Testing - Principles and Practices*", Pearson education, 2006.
2. Ron Patton, "*Software Testing*", Second Edition, Sams Publishing, Pearson Education, 2007.
3. Aditya P. Mathur, "*Foundations of Software Testing - Fundamental Algorithms and Techniques*", Dorling Kindersley (India) Pvt. Ltd., Pearson Education, 2008.
4. Alan C. Gillies, "*Software Quality: Theory and Management*", International Thomson Computer Press, 1997.
5. Mordechai Ben-Menachem "*Software Quality: Producing Practical Consistent Software*", International Thompson Computer Press, 1997.

IT1631	BLOCKCHAIN TECHNOLOGIES	L	T	P	C
		3	0	0	3

OBJECTIVES:

To enable the students to:

- understand the needs of cryptographic algorithms in blockchain technologies
- describe the operational and functional aspects of trading and mining
- know about the bitcoin consensus
- explain various algorithms that supports distributed consensus
- realize the usage of Hyper ledger fabric and ethereum in various fields

UNIT I INTRODUCTION TO BLOCKCHAIN 7

Blockchain- Public Ledgers, Blockchain as Public Ledgers -Bitcoin, Blockchain 2.0, Smart Contracts, Block in a Blockchain, Transactions-Distributed Consensus, The Chain and the Longest Chain - Cryptocurrency to Blockchain 2.0 - Permissioned Model of Blockchain, Cryptographic -Hash Function, Properties of a hash function-Hash pointer and Merkle tree.

UNIT II BITCOIN AND CRYPTOCURRENCY 9

A basic crypto currency, Creation of coins, Payments and double spending, FORTH – the precursor for Bitcoin scripting, Bitcoin Scripts , Bitcoin P2P Network, Transaction in Bitcoin Network, Block Mining, Block propagation and block relay, Consensus introduction, Distributed consensus in open environments-Consensus in a Bitcoin network.

UNIT III BITCOIN CONSENSUS 10

Bitcoin Consensus, Proof of Work (PoW)- HashcashPoW , BitcoinPoW, Attacks on PoW, monopoly problem- Proof of Stake- Proof of Burn - Proof of Elapsed Time - Bitcoin Miner, Mining Difficulty, Mining Pool-Permissioned model and use cases, Design issues for Permissioned Blockchains, Execute contracts-Consensus models for permissioned blockchain-Distributed consensus in closed environment-Paxos

UNIT IV DISTRIBUTED CONSENSUS 9

RAFT Consensus-Byzantine general problem, Byzantine fault tolerant system-Agreement Protocol, Lamport-Shostak-Pease BFT Algorithm-BFT over Asynchronous systems, Practical Byzantine Fault Tolerance.

UNIT V HYPER LEDGER FABRIC, ETHERUM AND BLOCKCHAIN APPLICATIONS 10

Hyper Ledger Fabric: Architecture of Hyperledger fabric v1.1-Introduction to hyperledger fabric v1.1, chain code- **Ethereum**:Ethereum network, EVM, Transaction fee, Mist Browser, Ether, Gas, Solidity, Smart contracts, Truffle-Design and issue Crypto currency, Mining, DApps, DAO **Blockchain Applications:** Internet of Things-Medical Record Management System-Blockchain in Government and Blockchain Security-Blockchain Use Cases – Finance.

TOTAL: 45 PERIODS

COURSE OUTCOMES

Upon successful completion of course the students will be able to

CO1: Demonstrate the need of cryptographic algorithms in blockchain technology

CO2: Explain the functional/operational aspects of trading and mining using crypto currencies.

CO3: Explain the Bitcoin census, Proof of work and its design issues

CO4: Paraphrase the algorithms used to provide distributed concensus

CO5: Outline the architectures of Hyperledger fabric, Ethereum and the impact and the challenges in implementing Blockchain in domains like IoT, governance, finance, Health care etc.

TEXT BOOKS

1. Imran Bashir, "*Mastering Blockchain: Distributed Ledger Technology, Decentralization, and Smart Contracts Explained*", Second Edition, Packt Publishing, 2018.

2. A. Narayanan, J. Bonneau, E. Felten, A. Miller, S. Goldfeder, "*Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction*" Princeton University Press, 2016.

REFERENCES:

1. ArshdeepBahga and Vijay Madisetti, "*Blockchain Applications : A Hands-On Approach*", 2017
2. Andreas Antonopoulos, Satoshi Nakamoto, "*Mastering Bitcoin*", O'Reilly Publishing, 2014.
3. Roger Wattenhofer, "*The Science of the Blockchain*" Create Space Independent Publishing Platform, 2016.
4. Alex Leverington, "*Ethereum Programming*" Packt Publishing Limited, 2017.

IT1632 MACHINE LEARNING AND DEEP LEARNING TECHNIQUES

L	T	P	C
3	0	0	3

OBJECTIVES

To enable the students

- To understand the learning methodologies of Machine learning
- To identify appropriate learning algorithms for various types of learning tasks in various domains
- To clustering, classification and reinforcement learning algorithms
- To understand instance based and genetic algorithms in machine learning
- To understand the fundamental concepts of Artificial Neural Networks.
- To experiment Deep Neural Network framework.

UNITI INTRODUCTION 8

Introduction to Machine Learning, supervised, unsupervised, reinforcement and semi-supervised modeling -Machine Learning pipeline- Linear regression, Logistic regression and Evaluation metrics

UNIT II LEARNING ALGORITHMS 9

Clustering: k-means, Hierarchical and Density Based clustering, Classification: Naive Bayes, Support Vector Machine, Decision Tree- -Random Forest-

Adaboost and Evaluation Metrics Reinforcement Learning: Task – Q-Learning – Temporal Difference Learning

UNIT III INSTANCE BASED LEARNING AND GENETIC ALGORITHMS 8

K- Nearest Neighbor Learning –Locally weighted Regression – Radial Bases Functions – Case Based Learning – Genetic Algorithms – Hypothesis Space Search – Models of Evolution and Learning- Parallelizing Genetic Algorithms

UNIT IV ARTIFICIAL NEURAL NETWORKS 8

Biological Neuron, Idea of computational units, McCulloch–Pitts unit, Single and Multi layer Perceptron learning Algorithms, Feedforward Networks, Backpropagation, Convolutional Neural Networks, Recurrent Neural Networks

UNIT V DEEP LEARNING 12

Deep Feed Forward network, regularizations, training deep models, dropouts, Gradient- Descent Strategies, Training Deep Neural Networks using Back Propagation, Training Deep Neural Networks using Convolution, Probabilistic Neural Network: Hopfield Net, Boltzman machine

TOTAL: 45 PERIODS

TEXT BOOKS:

1. Tom M. Mitchell, *Machine Learning*, McGraw Hill Education (India) Private Limited, 2013.

2. Ian Goodfellow and Yoshua Bengio and Aaron Courville, *Deep Learning*, MIT Press, 2016.

REFERENCES:

1. EthemAlpaydin, — *Introduction to Machine Learning (Adaptive Computation and Machine Learning)*, The MIT Press2004.

2. Stephen Marsland, —*Machine Learning: An Algorithmic Perspective*,CRC Press, 2009.

3. Michael Nielsen, "*Neural Networks and Deep Learning*", Determination Press, 2015.

http://neuralnetworksanddeeplearning.com/

IT1633 ADVANCED PYTHON PROGRAMMING

L	T	P	C
2	0	2	3

COURSE OBJECTIVES:

To enable the students to:

- experience the first class functions and flask micro framework
- understand the object oriented programming concepts
- apply the concepts of database with the Python programming
- demonstrate the Numpy and Pandas package to perform advanced computation
- represent the data set pictorially using the Python packages

UNIT I FUNTIONS AND MODULES 6

Comprehension, lambda, First class functions- Map, reduce, filter, Function Decorators and Closures, Iterables, Iterators and Generators, yield, context manager, magical methods, RegEx

Laboratory: 6

1. Given a list of numbers [1,2,3,4,5,6]. Filter the even numbers, map them to a square function and reduce it to 56 using filter(), map(), reduce() in python and not more than six LOCs to print the sum.
2. Write a python code to decorate arithmetic operators.
3. Write a python code to find the pattern “625706” in a text using RegEx.

UNIT II OBJECT ORIENTED PROGRAMMING CONCEPTS 7

Object Oriented Programming – Classes – Class Attributes – Instances – Instance Attributes – Binding – Method Invocation – Static Methods - Class Methods – Encapsulation – Composition – Subclassing and Derivation - Inheritance – Polymorphism – Built-in Functions – Special Methods.

Laboratory: 4

1. Write a python code to find the average height (feet and inches) of students’ in a class by overriding the magic methods of python in a “Height” class.
2. Inheritance
3. Polymorphism

UNIT III NUMPY 5

Introduction to Numpy – Ndarray Object - Data Types - NumPy Arrays – Array Manipulation - Indexing and slicing - Broadcasting – Binary operators – String functions – Statistical functions - Comparisons, Masks, Sorting Arrays - Structured Data: NumPy's Structured Arrays.

Laboratory: 4

1. Write a python code to add elements in two arrays of same dimension.
2. Read a CSV file and fill the missing numeric values with mean and categorical values with mode.

UNIT IV PANDAS 12

Introducing Pandas Objects – Data Frames – Reading data – Cleansing data - Operating on Data in Pandas - Hierarchical Indexing - Combining Datasets: Concat and Append - Combining Datasets: Merge and Join -Aggregation and Grouping - Pivot Tables - Vectorized String Operations - Working with Time Series – High Performance Pandas: eval() and query()

Laboratory: 4

1. Write a python code to add elements in two arrays of same dimension.
2. Read a CSV file and fill the missing numeric values with mean and categorical values with mode.

UNIT V DATA VISUALIZATION 8

Simple Line Plots - Simple Scatter Plots – Boxplot- Visualizing Errors - Density and Contour Plots – Histograms – Binnings – Density - Customizing Plot Legends - Customizing Colorbars - Multiple Subplots - Text and Annotation – Customizing Ticks - Customizing Matplotlib: Configurations and Stylesheets - Three-Dimensional Plotting in Matplotlib - Geographic Data with Basemap - Visualization with Seaborn

Laboratory: 4

1. Analyse the given dataset through Data visualisation and identify the insights.

TOTAL: 60 PERIODS

COURSE OUTCOMES

Upon successful completion of course the students will be able to

CO1: Describe the important first class functions and flask micro web framework in Python

CO2: Outline the basic concepts of Object oriented programming in python

CO3: Make use of database concepts in Python programming to solve simple real time problems

CO4: Utilize the NumPy and SciPy package in Python for advanced computations

CO5: Visualise the dataset to get insights using Python packages.

TEXT BOOK

1. Steven F. Lott, "*Mastering Object-Oriented Python: Build powerful applications with reusable code using OOP design patterns and Python 3.7*", 2nd Edition, ISBN-10: 1789531365
2. Jake VanderPlas, 2016, *Python Data Science Handbook: Essential Tools for Working with Data*, 1 sted, O'Reilly Media, Inc

REFERENCES:

1. Jason Cannon, "*Python Programming for Beginners*" O,Reilly, 2010
2. David Beazley , Brain K Jones "*Python CookBook*" Third edition,2013
3. Tamara Munzner, "*Visualization Analysis and Design*", CRC Press, 2014.
4. Scott Murray, "*Interactive Data Visualization for the Web*", O'Reilly Media, First Edition, 2013.

IT1634 SOFTWARE DEFINED NETWORKS

L T P C

2 0 2 3

COURSE OBJECTIVES:

To enable the students to:

- Learn the fundamentals of software defined networks.
- Understand the concepts of open flow and SDN controllers
- Understand the use of SDN in Data Center Networks
- Study the fundamentals of SDN Programming.
- Implement topologies in SDN

UNIT I INTRODUCTION TO SDN 7

History of Software Defined Networking (SDN) – Modern Data Center – Traditional Switch Architecture – Why SDN – Evolution of SDN – How SDN Works – Centralized and Distributed Control and Date Planes

UNIT II OPEN FLOW & SDN CONTROLLERS 8

Open Flow Specification – Drawbacks of Open SDN, SDN via APIs, SDN via Hypervisor[1] Based Overlays – SDN via Opening up the Device – SDN Controllers – General Concepts

UNIT III DATA CENTERS 9

Multitenant and Virtualized Multitenant Data Center – SDN Solutions for the Data Center Network – VLANs – EVPN – VxLAN – NVGRE

UNIT IV SDN PROGRAMMING 6

Programming SDNs: Northbound Application Programming Interface, Current Languages and Tools, Composition of SDNs – Network Functions Virtualization (NFV) and Software Defined Networks: Concepts, Implementation and Applications

UNIT V SDN – OPEN FLOW COMMUNICATION LABORATORY

(30 lab hours)

1. Configuration of Mininet and basic Mininet Commands Execution - 2 hours
2. Classic Topologies simulation with Mininet - 2 Hours
3. Custom Topologies creation with Mininet and Python API - 2 Hours
4. Configuration of ONOS controller with ONOS GUI - 2 Hours
5. ONOS CLI - 1 Hour
6. Visualization of Network Topologies using ONOS GUI - 2 Hours
7. Creating SDN environment with OpenDaylight, Mininet, PuTTY and Xming - 4 Hours

TOTAL: 60 PERIODS

TEXT BOOKS:

1. Paul Goransson and Chuck Black, —*Software Defined Networks: A Comprehensive Approach*, First Edition, Morgan Kaufmann, 2014.

2. Thomas D. Nadeau, Ken Gray, —*SDN: Software Defined Networks*, O'Reilly Media, 2013.

REFERENCES:

1. Siamak Azodolmolky, —*Software Defined Networking with Open Flow*, Packet Publishing, 2013.
2. Vivek Tiwari, —*SDN and Open Flow for Beginners*‖, Amazon Digital Services, Inc., 2013.
3. Fei Hu, Editor, —*Network Innovation through Open Flow and SDN: Principles and Design*, CRC Press, 2014.

CS1631 BIGDATA ANALYTICS – TOOLS AND TECHNIQUES

L	T	P	C
2	0	2	3

COURSE OBJECTIVES:

To enable the students to:

- Know the fundamental concepts of Big data analytics and its applications
- Describe various techniques for Big Data Analytics
- Learn and apply different recommendation systems for large volumes of data
- Learn to use various techniques for mining data stream.
- Know about the tools such as HIVE, HBase to process large amount of data

UNIT I INTRODUCTION TO BIG DATA 6

Theory Component: 6

Introduction to Big Data : Evolution of Big data - Best Practices for Big data Analytics – Types of Data - Big data characteristics - Validating - The Promotion of the Value of Big Data – Challenges with Big data - Big Data Use Cases- Applications

UNIT II HADOOP AND MAPREDUCE FRAMEWORK 6

Theory Component: 6

A General Overview of High-Performance Architecture – Introducing Hadoop – RDBMS versus Hadoop - Hadoop Overview – HDFS (Hadoop Distributed File System) – Introduction to Map-Reduce and YARN - Map Reduce Programming Model.

UNIT III RECOMMENDATION SYSTEM 15

Theory Component: 6

Recommendation System: Collaborative Recommendation- Content Based Recommendation - Knowledge Based Recommendation- Hybrid Recommendation Approaches.

Lab Component: 9

1. Implement content Based Recommender System using Cosine Similarity
2. Implement collaborative Based Recommender System using different Similarity measures.

UNIT IV STREAM MEMORY 18

Theory Component: 6

Introduction to Streams Concepts – Stream Data Model and Architecture - Stream Computing, Sampling Data in a Stream – Filtering Streams – Counting Distinct Elements in a Stream – Estimating moments – Counting oneness in a Window – Decaying Window – Real time Analytics Platform(RTAP) applications - Case Studies - Real Time Sentiment Analysis, Stock Market Predictions.

Lab Component: 12

3. Implement Bloom Filter for filter on Stream Data in C++/java.
4. Implement Flajolet-Martin Algorithm for counting distinct elements in Stream Data.
5. Implement Stock Market Prediction

UNIT V NOSQL DATA MANAGEMENT FOR BIG DATA AND VISUALIZATION **15**

Theory Component: **6**

Using Graph Analytics for Big Data: Graph Analytics - NoSQL Databases : Schema-less Models: Increasing Flexibility for Data Manipulation-Key Value Stores- Document Stores - Tabular Stores - Object Data Stores - Graph Databases Hive - Sharding –- Hbase.

Lab Component: **9**

6. Implement Spectral Clustering.
7. Implement real time sentiment analysis – Tweets.
8. Perform NoSQL database using mongodb to create, update and insert.

TOTAL: 60 PERIODS

COURSE OUTCOMES

Upon successful completion of course the students will be able to

CO1 : Discuss the fundamental concepts of Big data and its applications

CO2 : Apply the concepts of MapReduce Framework

CO3 : Build an appropriate recommendation system for real time data

CO4 : Experiment with various algorithms for mining data streams

CO5 : Utilize NO SQL database for managing huge volume of data

TEXT BOOKS:

1. Hurwitz JS, Nugent A, Halper F, Kaufman M.” *Big data for dummies*”, John Wiley & Sons; 2013.
2. Alan Gates & Daniel Dai ,” *Pig :Data flow Scripting with Hadoop*”, O’Reilly Media, 2016.
3. Jason Rutherglen, Dean Wampler, Edward Capriolo , “*Programming Hive*”, First edition, O’Reilly Media, 2012

REFERENCES:

1. Judith Hurwitz, Alan Nugent, Dr. Fern Halper and Marcia Kaufman, “*Big Data*”, Wiley Publications, 2014.

2. Mark Van Rijmenam, "*Think Bigger: Developing a Successful Big Data Strategy for Your Business* ", Amazon, First edition, 2014.
3. Hurwitz JS, Nugent A, Halper F, Kaufman M. *Big data for dummies*. John Wiley & Sons; 2013.
4. Tom White, "*Hadoop: The Definitive Guide*", O'Reilly Publications, 2011. Dayong Du, "*Apache Hive Essentials*", Packet Publishing, 2015.
5. Hanish Bansal, Saurabh Chauhan ,"*Apache Hive Cookbook*", Packt publishing, 2016

OPEN ELECTIVE

OIT151	**MOBILE APP DEVELOPMENT**	**L**	**T**	**P**	**C**
		3	**0**	**0**	**3**

COURSE OBJECTIVES:

To enable the students to:

- Understand the architecture and framework of android environment
- Design android applications with appropriate user interface
- Develop android applications using database
- Develop applications using messaging and networking concepts
- Deploy android applications

UNIT I ANDROID FUNDAMENTALS 9

Mobile Application development and trends – Android overview and Versions – Android open stack, features – Setting up Android environment (Eclipse, SDK, AVD)- Simple Android application development – Anatomy of Android applications – Activity and Life cycle – Intents, services and Content Providers

UNIT II ANDROID USER INTERFACE 9

Layouts: Linear, Absolute, Table, Relative, Frame, Scrollview, Resize and reposition - Screen orientation – Views: Textview, EditText, Button, ImageButton, Checkbox, ToggleButton, RadioButton, RadioGroup, ProgressBar, AutocompleteText, Picker, Listviews and Webview– Displaying pictures with views: Gallery and ImageView, ImageSwitcher, Gridview – Displaying Menus: Helper methods, Option and Context

UNIT III DATA PERSISTENCE 9

Shared User preferences – File Handling: File system, System partition, SD card partition, user partition, security, Internal and External Storage – Managing data using SQLite –User defined content providers

UNIT IV MESSAGING, NETWORKING AND SERVICES 9

SMS Messaging: Sending and Receiving – Sending email and networking – Downloading binary and text data files – Access Web services – Developing android services: creating services, performing long running task in a service-performing repeated task in a service

UNIT V LOCATION ACCESS & APPLICATION DEPLOYMENT 9

Location based services: Display map, zoom control, view and change, Marking, Geocoding, Get location - Publish Android applications and Deployment

TOTAL: 45 PERIODS

COURSE OUTCOMES

Upon successful completion of the course, the students will be able to

CO1: Understand the architecture and framework of android environment

CO2: Design android applications with appropriate user interface

CO3: Develop android applications using database

CO4: Develop applications using messaging and networking concepts

CO5: Deploy android applications

TEXT BOOKS:

1. WeiMeng Lee (2012), "*Beginning Android Application Development*", Wrox Publications (John Wiley, New York)

REFERENCES:

1. Ed Burnette (2010), "*Hello Android: Introducing Google's Mobile Development Platform*", The Pragmatic Publishers, 3rd edition, North Carolina USA

2. Reto Meier (2012), "*Professional Android 4 Application Development*", Wrox Publications (John Wiley, New York).
3. ZigurdMednieks, Laird Dornin, Blake Meike G, Masumi Nakamura (2011), "*Programming Android: Java Programming for the New Generation of Mobile Devices*", OReilly Media, USA

OIT152 PRINCIPLES OF DATABASE MANAGEMENT SYSTEMS

L	T	P	C
3	0	0	3

OBJECTIVES:

To enable the students to:

- Make the students to learn the fundamentals of data models and to representa database system using ER diagrams.
- Study SQL and relational database design.
- Understand the internal storage structures using different file and indexingtechniques which will help in physical DB design.
- Understand the fundamental concepts of transaction processing-concurrencycontrol techniques and recovery procedures.
- Learn about file organization and query processing

UNIT I DBMS AND CONCEPTUAL DATA MODELING 9

Purpose of Database System – Data independence - Data Models – Database System Architecture – Conceptual Data modeling: ER models - Enhanced-ER Model. Introduction to relational databases – Relational Model – Keys – ER-to-Relational Mapping. Modeling of a library management system.

UNIT II DATABASE QUERYING 9

Relational Algebra – SQL: fundamentals – DDL – Specifying integrity constraints - DML – Basic retrieval queries in SQL - Complex SQL retrieval queries – nested queries – correlated queries – joins - aggregate functions. Creating a table,

populating data, adding integrity constraints, querying tables with simple and complex queries.

UNIT III DATABASE PROGRAMMING 9

Database programming with function calls, stored procedures - views – triggers. Embedded SQL. ODBC connectivity with front end tools. Implementation using ODBC/JDBC and SQL/PSM, implementing functions, views, and triggers in MySQL / Oracle.

UNIT IV DATABASE DESIGN 9

Functional Dependencies – Design guidelines – Normal Forms: first, second, third – Boyce/Codd Normal Form – Normalization algorithms. Design of a banking database system / university database system.

UNIT V ADVANCED TOPICS 9

Database security issues – Discretionary access control – role based access – Encryption and public key infrastructures – challenges. Information Retrieval: IR Concepts, Retrieval Models, Queries in IR systems.

TOTAL: 45 PERIODS

COURSE OUTCOMES

Upon successful completion of course the students will be able to

CO1 : Infer the basic concepts of database system and model ER diagram for forreal time applications

CO2 : Use appropriate SQL commands to store and access data from RelationalDatabase..

CO3 : Construct normalized database for real world scenario using functionaldependencies.

CO4 : Illustrate the importance of transaction and concurrency control to maintainconsistency in a database..

CO5 : Interpret the mechanism incorporated in file organization and QueryProcessing.

TEXTBOOKS:

1.Ramez Elmasri, Shamkant B. Navathe, 2011, *Fundamentals of Database Systems*, 6th ed , Pearson.

2.Abraham Silberschatz, Henry F. Korth, S. Sudharshan, 2011, *Database System Concepts*, 6th ed, Tata McGraw Hill.

REFERENCES:

1. C.J.Date, A.Kannan, S.Swamynathan, 2006, *An Introduction to Database Systems*, 8th ed, Pearson Education.

2. Raghu Ramakrishnan, 2015, *Database Management Systems,* 4th ed, McGraw-Hill Publications.