Master of Computer Application

Name of Program	**Master of Computer Application**
Abbreviation	**MCA**
Duration	**2 Years**
Eligibility Criteria	Passed BCA/ Bachelor Degree in Computer Science Engineering or equivalent Degree. OR Passed B.Sc./ B.Com./ B.A. with Mathematics at 10+2 Level or at Graduation Level (with additional bridge Courses as per the norms of the concerned University).
Objective of Program	The core objective of the MCA programme is to prepare the students for productive career in software industry and academia by providing an outstanding environment of teaching and research in the core and emerging areas of the discipline.
Program Outcome	**PO1 : Fundamental Knowledge Enrichment** Program trains students with the core computer science and Information Technology (IT) knowledge domains. It also makes students capable of using core concepts in the conceptualization of domain specific application development. **PO2 : Critical Thinking Development** The program develops the skills of critical thinking, problem solving, evaluative learning of various techniques, and understanding the essence of the problem. **PO3 : Advanced Emerging Technology Awareness** The program trains students with the latest technologies that is being used in the industry. The continuous syllabi review adds value to the program for the outgoing students and make them ready to face challenging demands of the industry. **PO4 : Advanced Tools Usage** The program teaches the students to apply the advanced tools to solve real world problems. **PO5 : Nurturing Project Planning and Management Capabilities** The program trains students for designing and conceptualizing the software architecture, planning and managing the product development process of complex and live software projects. It also makes students understand the decision making for selection of an appropriate project management capabilities. **PO6 : Real World Problem / Project Development** Real world project provides the candidates exposure to work in the challenging and demanding environment of the industry. The project development training makes students employable and industry ready. **PO7 : Team Work and Leadership Development** Trains students to work in a team and also to take leadership of the of the project management team.
Program Specific Outcomes	PSO1 : Develop and strengthen the fundamental core concepts that are required to solve complex problems PSO2 : Develop the professional and entrepreneurship skills that needs independent logical and analytical thinking, teamwork and leadership PSO3 : Nurture the students to investigate for the design and development of a workable solution for a real world problem PSO4 : Develop students for self-learning and practicing challenging

	problem solution PSO5 : Train students to apply managerial skills to develop business applications. PSO6 : Train students to use recent computer science and application domain specific knowledge PSO7 : Train students to take-up the real world challenges to develop workable solution to a domain specific problem PSO8 : Inculcate the passion for continuous learning and doing research for making a successful professional career

Mapping between POs and PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8
PO1								
PO2								
PO3								
PO4								
PO5								
PO6								
PO7								

Medium of Instruction: English

Program Structure: Semester 1

Course Code	Title	Teaching per week		Course Credits	University Examination		Internal Marks	Total Marks
		Theory	Practical		Duration	Marks		
101	Relational Database Management System	4	0	4	3 Hrs	70	30	100
102	Object Oriented Programming Methodology	4	0	4	3 Hrs	70	30	100
103	Cloud Computing	4	0	4	3 Hrs	70	30	100
104	Computer Network	4	0	4	3 Hrs	70	30	100
105	Data Structures and Design and Analysis of Algorithms	4	0	4	3 Hrs	70	30	100
106	Programming Skills - I	0	3	3	2 Hrs	70	30	100
107	Programming Skills – II	0	3	3	2 Hrs	70	30	100
108	Programming Skills – III	0	4	4	2 Hrs	70	30	100
	Total	20	10	30	21 Hrs	560	240	800

Program Structure: Semester 2

Course Code	Title	Teaching per week		Course Credits	University Examination		Internal Marks	Total Marks
		Theory	Practical		Duration	Marks		
201	Artificial Intelligence	4	0	4	3 Hrs	70	30	100
202	Front End Technologies	4	0	4	3 Hrs	70	30	100
203	Programming in .NET	4	0	4	3 Hrs	70	30	100
204	Python Programming Language	4	0	4	3 Hrs	70	30	100
205	iOS/Android	4	0	4	3 Hrs	70	30	100
206	Programming Skills – IV	0	3	3	2 Hrs	70	30	100
207	Programming Skills – V	0	2	2	2 Hrs	70	30	100
208	Programming Skills – VI	0	2	2	2 Hrs	70	30	100
209	Programming Skills – VII	0	3	3	2 Hrs	70	30	100
	Total	20	10	30	23 Hrs	630	270	900

MCA 1st Semester

Course: 101: Relational Database Management System

Course Code	**101**
Course Title	**Relational Database Management System**
Credit	4
Teaching per Week	4 Hrs
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Review / Revision	June 2020
Purpose of Course	Give fundamental knowledge of Database Fundamentals like Keys & Normalisation, Oracle Database Server Architecture and Working knowledge of SQL & PL/SQL in Oracle.
Course Objective	To acquaint the students with Database Fundamentals like Keys & Normalisation in general and Oracle Architecture in particular. Also, to get working knowledge of SQL and PL/SQL programming
Course Outcome	CO1: Students will be able to understand and evaluate the role of database management systems in IT applications within an organization. CO2: Students will be able to normalise any real-life database and hence they will be able to design and implement properly structured databases of real-world applications. CO3: Students will be able to write queries using Structured Query Language (SQL) and programs using PL/SQL. CO4: Students will understand how Oracle Database works and the importance of various components of Oracle Database. CO5: Students will understand administration and security enforcement of Oracle Database. This will help them in pursuing higher studies and career in Database Administration.
Mapping between COs with PSOs	
Pre-requisite	Nil
Course Content	**Unit 1: Database Computing Models** 1.1. Notion of Keys (Super Key, Candidate Key, Primary Key, Foreign Key) 1.2. Referential Integrity Constraint 1.3. Functional Dependencies 1.4. Normalization using Functional Dependencies 1.5. Normalization using Multivalued Dependencies 1.6. Normalization using Join Dependencies **Unit 2: Overview of Oracle Database Server Architecture** 2.1. Architecture of Oracle Database and Oracle Instance 2.2. Overview of Physical and Logical Structures 2.3. Oracle Server Start-up and Shutdown 2.4. Creating Database **Unit 3: Oracle Tools and Utilities** 3.1. SQL 3.2. Pl/SQL Procedural Extension. 3.2.1. Overview, PL/SQL data types & Control Structures 3.2.2. Cursors 3.2.3. Stored Procedures & Functions 3.2.4. Database Triggers 3.2.5. Package Creation

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8
CO1	■		■		■		■	
CO2	■	■	■		■		■	
CO3	■		■	■		■	■	
CO4			■	■	■	■	■	■
CO5		■	■	■	■	■	■	■

	Unit 4: Database Administration 4.1. Managing Users 4.1.1. User Authentication Methods 4.1.1.1. Password Authentication 4.1.1.2. O.S Authentication 4.1.2. User Configuration Setup 4.1.2.1. Profiles 4.1.2.2. Default Table space 4.1.2.3. Temporary Table space 4.1.3. Resource Management 4.1.3.1. Quotas 4.1.4. Working with user database account 4.1.4.1. Creating, Modifying and deleting user account 4.1.4.2. Changing password 4.2. Backup & Recovery **Unit 5: Database Security** 5.1. Authentication 5.2. Privileged Accounts & Privileges 5.3. Object Security 5.4. System security 5.5. Database Roles 5.6. Database Auditing [Self-Study] Export & Import Tools, Overview of Grid Based Database **Computing, Calling External Routines from PL/SQL
Reference Books	1. Oracle 9i PL/SQL Programming -Scott Urman- Oracle Press 2. Oracle DBA Fundamentals-I - Oracle Press 3. Effective PL/SQL: - Apress 4. Expert Oracle Database Architecture 9i and 10g-Tom Kyte- Apress 5. Effective Oracle by Design - Peter Norton - Tom Kyte-Oracle Press 6. Oracle 24 X 7 Tips and Techniques - Venkat Devraj– Oracle Press 7. Expert Oracle Database 11g Administration – Alpati- Wiley Student Edition 8. Fundamentals of Database Management System- Gilleneon-Wiley Student Edition 9. SQL & PL/SQL for Oracle 11g Black Book - Deshpande-McGraw Hill 10. Beginning Oracle Database 11g Administration from novice to professional-Iggy Fernandez - Apress/Springer 11. Oracle PL/SQL-Benjamin Rosenweig & Elena Silvestrova-4/e, Pearson 12. Database Systems Using Oracle: A simplified guide to SQL & PL/SQL-Shah Nilesh- PHI 13. Learning Oracle SQL & PL/SQL: A Simplified Guide- Chatterjee, Rajeeb C-PHI
Teaching Methodology	Class Work, Discussion, Self Study, Seminars and/or Assignment
Evaluation Method	30% Internal assessment 70% External Assessment

Course: 102: Object Oriented Programming Methodology

Course Code	**102**
Course Title	**Object Oriented Programming Methodology**
Credit	4
Teaching per Week	4 Hrs
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Review / Revision	June 2020
Purpose of Course	This course introduces the concepts of object-oriented programming and skills necessary for developing programs in C++.
Course Objective	1. To make students understand concepts of object-oriented paradigm 2. To make students develop C++ programs 3. To make students learn capabilities of an object-oriented programming language
Course Outcome	CO1- Articulate the principles of Object Oriented Problem solving and programming. CO-2-To demonstrate the differences between traditional imperative design and object Oriented Design CO-3-Outline the essential features and elements of C++ programming language. CO-4- To grasp and apply the concepts of class, method, constructor, abstraction, inheritance and Static Polymorphism. CO-5- To understand and apply Dynamic Polymorphism in real world applications. CO-6- To implement Genericity through the usage of Templates. Mapping between Cos and PSOs

Mapping between COs with PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8
CO1	■		■	■	■			
CO2	■			■	■	■	■	
CO3	■					■		
CO4		■	■	■		■	■	
CO5		■			■	■	■	■
CO6		■			■	■	■	■

Pre-requisite	Nil
Course Content	**Unit 1: C++ Basics** 1.1 Data Types 1.2 Pointers 1.2.1 Pointer Arithmetic 1.2.2 Array of Pointers 1.2.3 Dynamic Array 1.3 ios Class 1.4 Input and Output 1.5 Manipulators **Unit 2: Introduction to Object Oriented Programming** 2.1 Structure, classes and Objects 2.2 Encapsulation & Data Hiding 2.3 Constructors 2.4 Friend Functions 2.5 Inline Functions 2.6 Dynamic Object Creation & Destruction 2.7 Static Members 2.8 this Pointer 2.9 Destructors **Unit 3: Object Oriented Properties** 3.1 Introduction to Object Oriented Properties 3.2 Abstraction

	3.3 Polymorphism 3.3.1 Operator Overloading 3.3.2 Function Overloading & Type Conversions 3.4 Inheritance 3.4.1 Types of Inheritance 3.4.2 Constructor & Destructor calls during Inheritance 3.5 Dynamic Polymorphism 3.5.1 Overriding 3.5.2 Virtual Functions 3.5.3 Abstract Class **Unit 4: Data Files and Exception Handling** 4.1 Streams 4.2 File Types and Modes 4.3 File Pointers & their manipulations 4.4 Sequential Input & Output operations 4.5 Random access 4.6 Error handling during File operations 4.7 Exception Handling **Unit 5: Generic Programming and C++ Standard Template Library (STL)** 5.1 Template Classes 5.2 Template Functions 5.3 Implementation of Object-Oriented Properties on Template Classes 5.4 STL 5.4.1 Algorithms 5.4.2 Containers 5.4.3 Functions 5.4.4 Iterators
Reference Books	1. The C++ Programming Language, Stroustrup, Addison Wesley 2. The Complete Reference C++, Schildt, Tata McGraw Hill 3. OOP in Turbo C++, Robert Lafore, Galgotia Publication 4. C++ Primer, Lippman, Addition Wesley 5. Object Oriented Programming with ANSI and Turbo C++, Kamthane, Pearson Education 6. Thinking in C++, Bruce Eckel, Pearson 7. Object Oriented Modelling & Design, Rumbaugh, PHI 8. Object Oriented Analysis & Design with Application, Grady Booch, LPE 9. Standard C++ with Object Oriented Programming, Paul S. Wang, Thomson 10. C++ Primer Plus, Stephan Prata, Addison Wesley 12. Programming with ANSI C++, Bhushan Trivedi, Oxford University Press
Teaching Methodology	Class Work, Discussion, Self Study, Seminars and/or Assignment
Evaluation Method	30% Internal assessment 70% External Assessment

Course: 103: Cloud Computing

Course Code	**103**
Course Title	**Cloud Computing**
Credit	4
Teaching per Week	4 Hrs.
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Review / Revision	June 2020
Purpose of Course	The purpose of the course is to make student capable of implementing the concepts, methods and tools of Cloud Computing
Course Objective	The objective of the course is to provide comprehensive and in-depth knowledge of Cloud Computing Concepts, technologies, architecture, applications and implementation.
Course Outcome	CO1 : Explain students about the cloud and cloud computing, History & Evolution ,Properties & Characteristics, Advantages & Disadvantages of cloud computing. CO2: Explain students about various service models and deployment models CO3: To provide students a foundation of different cloud service models IAAS,PAAS and SAAS so that they are able to use Cloud Computing Services in real world problem CO4:Understanding the concepts of cloud infrastructure security, data security and storage, Access control and authentication in cloud. CO5: Train students to use AWS and Microsoft Azure CO6: Explain students in brief about BigTable and Firebase
Mapping between COs with PSOs	
Pre-requisite	Basics of DBMS, Web Development & HTML, Networking
Course Out come	To give basic knowledge of cloud computing, its architecture and its benefits and how to deploy applications on well-known cloud platforms
Course Content	**Unit 1: Introduction to Cloud & its architecture** 1.1 Introduction & Definitions 1.2 Cloud Computing (NIST) 1.2.1 History & Evolution 1.2.2 Properties & Characteristics 1.2.3 Advantages & Disadvantages 1.3 Cloud Architecture overview **Unit 2: Cloud Computing Models** 2.1 Cloud computing Stack 2.1.1 Comparison with traditional architecture 2.2 Service Models 2.2.1 Infrastructure as a Service (IaaS) 2.2.2 Platform as a Service (PaaS) 2.2.3 Software as a Service (SaaS) 2.3 Deployment Models 2.3.1 Public Cloud

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8
CO1	■							
CO2		■	■	■	■	■		
CO3		■	■		■	■	■	
CO4	■	■	■			■		
CO5		■	■	■	■	■	■	■
CO6	■	■						

	2.3.2 Private Cloud 2.3.3 Hybrid Cloud 2.3.4 Community Cloud **Unit 3: Cloud Service Models** 3.1 Infrastructure as a Service (IAAS) 3.1.1 Introduction to Virtualization 3.1.1.1 Hypervisors, Virtual Machine, Machine Image 3.1.2 Resource Virtualization 3.1.2.1 Server, Storage, Network 3.1.3 Amazon EC2, Eucalyptus 3.2 Platform as a Service (PAAS) 3.2.1 Introduction to SOA 3.2.2 Cloud Platform 3.2.2.1 Computing 3.2.2.2 Storage 3.2.3 Introduction to Microsoft Azure 3.2.4 Introduction to Salesforce's Force.com 3.3 Software as a Service (SAAS) 3.3.1 Introduction 3.3.2 Web Service & Web OS **Unit 4: Cloud Security** 4.1 Infrastructure Security 4.2 Data Security and Storage 4.3 Identity and Access Management (IAM) 4.4 Access Control 4.5 Authentication in Cloud **Unit 5: Cloud Databases (DBaaS)** 5.1 AWS SimpleDB & RDS 5.2 Azure Table Service & SQL Azure 5.3 Introduction to BigTable 5.4 Introduction to Firebase
Reference Books	1. Cloud Computing Principles and Paradigms (Wiley) Rajkumar Buyya, James Broberg, Andrzej M. Goscinski 2. Cloud Computing: Principles, Systems and Applications Nikos Antonopoulos, Lee Gillam (Springer) 3. Enterprise Cloud Computing: Technology, Architecture, Applications Gautam Shroff - Cambridge University Press 4. Cloud and Virtual Data Storage Networking Greg Schulz - Auerbach 5. Cloud Security: A Comprehensive Guide to Secure Cloud Computing Ronald L Krutz, Russel Dean Vines (John Wiley & Sons) 6. Cloud Computing (David Crookes - TMH Education) 7. Cloud Computing Bible Barrie Sosinsky (Wiley India) 8. Cloud Computing: Implementation, Management and Security (James F Ransome, John W Rittinghouse - CRC Press)

	9. Amazon Cloud Computing with Java (Aditya Yadav - Lulu.com) 10. Grid and Cloud Database Management Fiore, Sandro, Aloisio, Giovanni - Springer 11. Building a Database Cloud for Dummies Michael Wessler John Wiley & Sons
Teaching Methodology	Class work, Discussion, Self-Study, Seminars and/or Assignment
Evaluation Method	30 % internal assessment and 70% external assessment

Course: 104: Computer Network

Course Code	**104**
Course Title	**Computer Network**
Credit	4
Teaching per Week	4 Hrs
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Review / Revision	June 2020
Purpose of Course	This course aims towards learning fundamentals of computer network. The course teaches students about the various network technologies and popular network protocols
Course Objective	1.To make students learn about computer network fundamentals 2.To make students familiar with services offered at each layer of the network protocol stack 3.To make students learn various protocols at data link layer, network layer, and transport layer of network.
Course Outcome	CO1 : Understand students the fundamental aspects of the computer networks. CO2 : Explain and help students to learn fundamentals network protocols at data link layer, network layer and transport layer. CO3 : Explore students the services offered at each layer of the network protocol stack. CO4 : Train students to implement various error control, flow control, routing algorithms and security algorithms fall under data link layer, network layer and transport layer. CO5 : Explore students the concepts of Security, digital certificate, Public key Infrastructure, and similar security schemes.

Mapping between COs with PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8
CO1								
CO2								
CO3								
CO4								
CO5								

Pre-requisite	Nil
Course Content	**Unit 1: Introduction to Data Communication** 1.1 Introduction to networks, Internet and its application 1.2 Network Structure 1.3 Network Architecture 1.4 The OSI Reference model & services 1.5 The TCP/IP Reference model and Comparison with OSI Model 1.6 Concepts of data transmission 1.6.1 Guided and unguided Transmission media. PSTN 1.7 Multiplexing & switching techniques 1.8 ISDN (Integrated Service Digital Network) **Unit 2: Data Link Layer** 2.1 MAC Sub layer 2.1.1 Multiple Access Protocols 2.1.2 Ethernet 2.1.3 LAN protocols & IEEE standards for LAN 2.1.4 Fibre Optic & Satellite networks 2.2 Data Link Layer protocols 2.3 Error detection & correction **Unit 3: Upper Layers**

	3.1 Network 3.1.1 Routing Algorithms 3.1.2 Congestion Control Algorithm 3.1.3 Internetworking 3.2 Transport Layer 3.2.1 Connection Management 3.3 Concepts of Session Layer **Unit 4: The Presentation Layer** 4.1 Data Compression Technique 4.2 Cryptography 4.3 Symmetric Key Algorithms 4.4 Public – Key Algorithms & management of Public Keys 4.5 Digital Signatures and Communications security **Unit 5: The Application Layer** 5.1 Electronic Mail 5.2 Virtual Terminals General Purpose Applications [Self Study] Virtual LAN
Reference Books	1. Networking Complete - 1st Edition 2002, BPB Publication (Text Book) 2. Data Communication and Networking: Forouzan, TMH 3. Computer Networks - A. S. Tanenbaum - Prentice-Hall 4. Computer Networks and Distributed Processing - Martin J. - Pretice-Hall 5. Local Area Networks: An Introduction - Stalling, William - Mc-Millan Publishing Co. 6. Computer Networks: Protocols, Standards and Interfaces - Black – Prentice-Hall 7. Data Networks: Concepts Theory and Practices - Black - PHI 8. N/W Architecture - Comer - Prentice-Hall
Teaching Methodology	Class Work, Discussion, Self Study, Seminars and/or Assignment
Evaluation Method	30% Internal assessment 70% External Assessment

Course: 105: Data Structures and Design and Analysis of Algorithms

Course Code	**105**
Course Title	**Data Structures and Design and Analysis of Algorithms**
Credit	4
Teaching per Week	4 Hrs
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Review / Revision	June 2020
Purpose of Course	This course introduces the various data structures and algorithms involving these data structures and their logical implementation. Students also will be able to understand complex data structures like trees and their use in various Applications
Course Objective	1. To learn fundamental data structures like arrays, stacks, lists. 2. To learn complex data structures like trees. 3. To learn and compare various sorting techniques. 4. To learn analysis of algorithms
Course Outcome	CO1. Explain the futures of various non-primitive data structure and their applications. Train students for algorithms to create, insert, delete and traversing various data structure. CO2. Explain the basics of computational complexity analysis and big O & theta notation that help for fundamental research in algorithm analysis. Students can select the efficient algorithm after analyzing a problem and identifying it's the computing requirements. CO3. Explain insight of basic technique for algorithm such as Divide & Conquer, Greedy method, Backtracking & Branch & Bound CO4. Students will be able to work with various searching and sorting techniques and compare its computational complexity. CO5. Train students with to work with Hash structure, proper hash method and collision detection techniques.

Mapping between COs with PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8
CO1	■	■	■	■			■	
CO2		■			■			■
CO3	■	■	■	■		■	■	■
CO4	■	■	■	■			■	
CO5	■	■						

Pre-requisite	C programming Language
Course Content	**Unit 1: Non-Primitive Data structures.** 1.1 Arrays - its storage structures & operations 1.2 Stacks - operations and its applications in Recursion, Polish expressions etc. 1.3 Queues - Types of queues, operations and its applications. 1.4 Linked lists - Types of linked list, operations and its applications. 1.5 Trees - Concept and Definitions, Operations, linked & threaded storage representation of Binary Trees. 1.6 Applications of Trees - The manipulation of Arithmetic expressions, Symbol-table construction, Syntax Analysis etc. **Unit 2: Analysis of Algorithms** 2.1 Asymptotic: Big-O and Theta **Unit 3: Basic techniques & example algorithms for** 3.1 Divide & Conquer method 3.2 Greedy method

	3.3 Backtracking 3.4 Branch & Bound **Unit 4: Searching and Sorting** 4.1 Sequential, Binary 4.2 Search Trees:- Height, Balanced tree, 2-3, tree, red-black trees weight-balanced trees 4.3 Sorting 4.3.1 Internal sorting - Insertion, Selection, Quick, 2-way merge and Heap 4.3.2 External sorting - k-way merging, Balanced merge and poly phase Merge **Unit 5: Hashing** 5.1 Hash Tables 5.2 Hash functions 5.2.1 Division method 5.2.2 Multiplication method [Self Study] Graphs – Creation and Traversal
Reference Books	1. An Introduction to Data Structures with applications - Trembley – McGraw Hill 2. Theory and Problems of Data Structure – Lipschutz Semour – McGraw Hill 3. Algorithms + Data Structure Programs - Wirth, Niclaus - PHI. 4. Fundamentals of Data Structures,Horwitz, E. and Sahni S. - Computer Science Press. 5. The Art of Computer Programming, Vols. 1-2, Knuth D. - Addison Wesley. 6. Data Structures and Algorithms - Aho A.V., Hopcroft and Ullman - Addison Wesley 7. Data Structure & "C" Programming - Vanwyte C J - Addison Wesley. 8. Data Structures, Algorithms And Object Oriented Programming – Tata McGraw Hill edition Geogory L. Heileman. 9. Data Structures and the Standard Template Library - William J. Collins, Tata McGraw Hill edition. 10. Programming with C++ and Data Structures - Maria Litvin & Gary Litvin, Vikas Publishing House Pvt. ltd. 11. Data Structures using C & C++ - Y. Langsam Moshe J. Angensterin & A.M. Tenenbaum 12. Data Structures and Algorithms in C++ - Adam Drozdek, Thomson Learning 13. Data Structures & Program Design in C - Robert Kruse, C.L. Tondo, Brnceleing PHI Pvt Ltd. 14. Data Structures and Algorithms in Java, Lafore, Pearson 15. Data Structures and Algorithm Analysis in Java, Mark Allen Weiss, Pearson 16. Data Structures and Algorithms in Java, Micheal T Goodrich, Roberto Tamassia, Wiley
Teaching Methodology	Class Work, Discussion, Self Study, Seminars and/or Assignment
Evaluation Method	30% Internal assessment 70% External Assessment

Course: 106: Programming Skills - I

Course Code	**106**
Course Title	**Programming Skills – I**
Credit	5
Teaching per Week	5 Hrs
Minimum weeks per Semester	15 (Including lab work, examination, preparation, holidays etc.)
Review / Revision	June 2020
Purpose of Course	Give fundamental knowledge of Database Models, Oracle Database Server Architecture and Working knowledge of SQL & PL/SQL in Oracle.
Course Objective	To acquaint the students with Client Server Architecture in general and Oracle Architecture in particular. Also, to get working knowledge of SQL and PL/SQL programming
Pre-requisite	None
Course Outcome	After studying the course, students will be able to understand how Oracle Database works and the importance of various components of Oracle. This course will also help students to appreciate the role of a database administrator. After successful completion, students will be able to manage Oracle database and will be able to write codes in SQL & PL/SQL necessary for an application.
Course Content	Practical based on paper no: 101 (RDBMS)
Reference Books	None
Teaching Methodology	Lab. Work
Evaluation Method	30% Internal assessment and 70% External Assessment

Course: 107: Programming Skills - II

Course Code	**107**
Course Title	**Programming Skill – II**
Credit	3
Teaching per Week	3 Hrs
Minimum weeks per Semester	15 (Including lab work, examination, preparation, holidays etc.)
Review / Revision	June 2020
Purpose of Course	This course helps students practically implement the concepts of object oriented programming using C++.
Course Objective	1. To make students practically learn concepts of object-oriented paradigm 2. To make students develop and code C++ programs.
Pre-requisite	Nil
Course Outcome	After studying the course, students will be able to practically solve common and complex programming problems using object-oriented paradigm. This course also helps students learn practical implementation of data files and operations upon them using object-oriented approach.
Course Content	Practical based on paper no: 102 (OOPM)
Reference Books	None
Teaching Methodology	Lab. Work
Evaluation Method	30% Internal assessment and 70% External Assessment

Course: 108: Programming Skills - III

Course Code	**108**
Course Title	**Programming Skill – III**
Credit	3
Teaching per Week	3 Hrs
Minimum weeks per Semester	15 (Including lab work, examination, preparation, holidays etc.)
Review / Revision	June 2020
Purpose of Course	This course introduces the various data structures and algorithms involving these data structures and their practical implementation using JAVA programming language. Students also will be able to understand and write JAVA programs using complex data structures like trees.
Course Objective	1. To practically learn implementation of fundamental data structures like arrays, stacks, lists using JAVA 2. To learn implementing complex data structures like trees using JAVA 3. To learn and compare various sorting techniques practically 4. To learn analysis of algorithms practically.
Pre-requisite	C programming Language
Course Outcome	After studying the course, students will be able to use data structures and their application in sorting, searching and comparison of algorithms. Students will also learn analysis of the algorithms.
Course Content	Practical based on paper no: 105 (Data Structures and Design and Analysis of Algorithms)
Reference Books	None
Teaching Methodology	Lab. Work
Evaluation Method	30% Internal assessment and 70% External Assessment

MCA 2nd Semester

Course: 201: Artificial Intelligence

Course Code	**201**
Course Title	**Artificial Intelligence**
Credit	4
Teaching per Week	4 Hrs
Medium of Instruction	English
Minimum weeks per Semester	15 (Including Classwork, examination, preparation, holidays etc.)
Effective From	June 2020
Purpose of Course	The purpose of the course is to make the student capable of implementing the concepts, methods, and tools of Artificial Intelligence and learn their implementation in Knowledge-Based Systems Course Objective To acquaint students with concepts of Artificial Intelligence and its applications.
Course Objective	To make students acquainted with concepts of Artificial Intelligence and its applications.
Course Outcome	CO1 : Explain students the insight of the historical and fundamental aspects the artificial intelligence. CO2 : Train students to represent declarative knowledge in the form of symbolic knowledge through various Knowledge Representation (KR) techniques like First Order Predicate Logic (FOPL), Semantic Network, Conceptual Graphs, Scripts, and Frames. CO3 : Train students to apply various searching algorithms fall under informed and uninformed search methods to solve complex problem of AI domain. CO4 : Explain and train students to deal with the uncertainty that inherently lies within many AI problem. CO5 : Expose the students with the analysis and development process of the knowledge based system development. CO6 : Explain students to utilize the AI problem solving techniques in the advanced AI problem domain like Natural Language Processing (NLP) and Computer Vision (CV)

Mapping between COs with PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8
CO1	■							
CO2	■		■					
CO3	■	■	■		■			
CO4		■	■	■		■	■	■
CO5	■		■		■	■		
CO6	■	■		■		■	■	■

Pre-requisite	Basics of Mathematics, Data Structures
Course Content	**Unit 1:** 1.1 Introduction to Artificial Intelligence 1.1.1 Definition of Artificial Intelligence 1.1.2 History of Artificial Intelligence 1.1.3 Application of Artificial Intelligence 1.1.4 Introduction to Knowledge-Based System 1.2 Turing Problem 1.3 Knowledge Representation 1.3.1 knowledge and Knowledge Base 1.3.2 First Order Predicate Logic (FOPL) 1.3.3 Inference Rules **Unit 2:** 2.1 Structured Knowledge Representation 2.1.1 Associative network and Conceptual graphs 2.1.2 Frames and Scripts 2.1.3 Conceptual Dependencies

	2.2 Searching 2.2.1 Search Problem 2.2.2 Initial State, action, transition model, goal test, the cost function 2.2.3 Uninformed Search 2.2.3.1 Depth First Search 2.2.3.2 Breadth-First Search 2.2.3.3 Iterative Deepening Search 2.2.4 Informed Search 2.2.4.1 Heuristics 2.2.4.2 A* Search 2.2.4.3 Minimax 2.2.4.5 Hill-Climbing Method 2.2.4.6 Constraint Satisfaction Search **Unit 3:** 3.1 Uncertainty 3.1.1 Probability 3.1.2 Conditional Probability 3.1.3 Baye's Rule 3.1.4 Joint Probability 3.1.5 Probability Rules 3.2 Introduction to Hidden Markov Model **Unit 4:** 4.1 Knowledge Acquisition 4.1.1 Knowledge gathering 4.1.2 Learning Models 4.1.2.1 Introduction to Supervised Learning 4.1.2.2 Introduction to Unsupervised Learning 4.1.2.3 Reinforcement Learning 4.1.3 Performance of Learning Model **Unit 5:** 5.1 Expert System 5.2 Characteristics of Expert System 5.3 Architecture of Expert System 5.4 Application of AI in Natural Language Processing 5.5 Application of AI in Computer Vision
Reference Books	1. Artificial intelligence, 3rd Edition, Kevin Knight, Elaine Rich, B. Shivashankar Nair, McGraw Hill 2. Russell Stuart Jonathan and Norvig Peter, Artificial Intelligence: A Modern Approach, 3rd Edition, Prentice-Hall, 2010 3. A First Course in Artificial Intelligence, Deepak Khemani, McGraw Hill 4. Introduction to artificial intelligence, Akerkar, Rajendra, PHI Learning 5. Foundation of Artificial Intelligence and Expert Systems by V.S. Janakiraman, K. Sarukesi, P. Gopalakrishnan, Mc Millan 6. Expert Systems Principles and Programming (3rd Edition) by Giarratano & Riley, Thomson (Vikas Publishing House)
Teaching Methodology	Classwork, Discussion, Self-Study, Seminars and/or Assignment
Evaluation Method	30% Internal assessment based on class attendance, participation, class test, quiz, assignment, seminar, internal examination, etc. 70% External based on semester end University examination

Course: 202: Front-End Technologies

Course Code	**202**
Course Title	**Front-end Technologies**
Credit	4
Teaching per Week	4 Hrs
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Review / Revision	June 2020
Purpose of Course	To understand the concepts of HTML, CSS, Front-end Scripting technologies
Course Objective	To teach the concepts of Front-end Scripting and its practical applications.
Course Outcome	CO1 : Explain students the insight of the fundamental aspects of the static web technology like HTMl , CSS and Javascript CO2 : Train students to represent declarative knowledge in the Boorstrap and it's components , with help of this students can learn static web designing . CO3 : Train students to understand various Javascript based technologies like Jquery , JSON and AJAX. CO4 : Explain and train students to understand basic concepts of Angular JS. CO5 : Expose the students to the various UI technology and real world use of it.. CO6 : After completion of this course, the student will be able to design and develop web pages and Interactive UI for Web Applications
Mapping between COs with PSOs	
Pre-requisite	Basic concepts of HTML, Web & Programming skills
Course Content	**Unit 1: Fundamentals of Web Technology** 1.1 HTML 1.1.1 Basic HTML tags 1.1.2 HTML Forms 1.2 HTML5 1.2.1 HTML5 new elements 1.2.2 HTML5 Form elements 1.2.3 HTML5 Attributes 1.2.4 Canvas 1.2.5 Video and Audio 1.2.6 Web storage 1.2.7 Geolocation 1.2.8 HTML 5 APIs 1.3 CSS3 1.3.1 Introduction to CSS3 1.3.2 Selectors and Classes 1.3.3 Font and Text effect 1.3.4 Colors, Gradients, Background Images, and Masks 1.3.5 Border and Box effects etc. 1.3.6 Embedding Media 1.4 JavaScript 1.4.1 Fundamentals of JavaScript 1.4.2 Syntax of JavaScript 1.4.3 Use of JavaScript in HTML 1.4.4 Validation using JavaScript 1.4.5 DOM **Unit 2: Introduction to Bootstrap** 2.1 What is Bootstrap

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8
CO1	■							
CO2	■		■					
CO3	■	■	■		■			
CO4		■	■	■		■	■	■
CO5	■		■		■	■		
CO6	■	■		■		■	■	■

	2.1.1 What is Responsive web page 2.1.2 Advantages and features of Bootstrap 2.1.3 Setup Environment 2.1.4 Apply bootstrap to Application 2.2 Bootstrap with CSS 2.2.1 Grid system 2.2.2 Typography 2.2.3 Code, table, forms, buttons, image, responsive utilities etc. 2.3 Bootstrap components 2.3.1 What is Bootstrap components 2.3.2 Advantages of components 2.3.3 Types of Bootstrap components 2.3.3.1 Glyphicons, Drop downs, button group, input groups navigation, pagination etc. **Unit 3: JQuery** 3.1 Introduction to JQuery 3.1.1 Syntax, Attributes, Selectors, Events 3.2 JQuery Effects 3.2.1 Hide/Show, Fade, Slide, Animation etc. 3.2.2 JQuery with HTML 3.3 Traversing 3.4 JQuery and AJAX **Unit 4: AJAX and JSON** 4.1 Ajax Basics 4.1.1 HTTP Request and Response Fundamentals 4.1.2 The XMLHttpRequest Object XMLHttpRequest Methods 4.1.3 XMLHttpRequest Properties 4.1.4 Cross-Browser Usage Sending a Request to the Server 4.1.5 Server-Side Processing Expanding and Contracting Content 4.1.6 Form Validation 4.1.7 Ajax-Based Database Querying using any one server site scripting language 4.2 JSON 4.2.1 JSON Basics 4.2.2 Syntax 4.2.3 Datatype, Parse, Stringify, Objects, Array 4.2.4 Use of JSON using any one server site scripting **Unit 5: Introduction to Angular JS** 5.1 Introduction to AngularJS 5.2 Directives, Expressions, Controllers, Filters, Tables, Html DOM 5.3 Modules, Forms, Includes, Views 5.4 Angular SQL 5.5 AngularJS Applications
Reference Books	1. Responsive Web Design with HTML5 and CSS3 By Ben Frain - Packt Publishing Ltd. 2. HTML, CSS & JavaScript Web Publishing in One Hour a Day, Sams Teach Yourself by Laura Lemay, Rafe Colburn, Jennifer Kyrnin – Sams Publication 3. Training Guide Programming in HTML5 with JavaScript and CSS3 (MCSD): 70-480 by Glenn Johnson - Pearson Education 4. Learning Bootstrap by Aravind Shenoy, Ulrich Sossou - Packt Publishing Ltd. 5. Professional AngularJS by Valeri Karpov, Diego Netto - John Wiley & Sons 6. Ajax: Creating Web Pages with Asynchronous JavaScript and XML - Edmond Woychowsky - Prentice Hall

Teaching Methodology	Class work, Discussion, Self-Study, Seminars and/or Assignment
Evaluation Method	30 % internal assessment and 70% external assessment

Course: 203: Programming in .NET

Course Code	**203**
Course Title	**Programming in .NET**
Credit	4
Teaching per Week	4 Hrs.
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Review / Revision	June 2020
Purpose of Course	This course is an introduction to students to understand fundamentals of .NET technology. The course also gives students an idea about VB.NET Programming. The course also explains the concept of ASP.NET
Course Objective	1. To make students understand .NET Technology 2. To make students understand VB.NET Programming 3. To make students understand the importance of ASP.NET
Course Outcome	CO-1 Study .Net Architecture. CO-2- Design and develop console and window based .NET application. CO-3-To learn basic syntax and implement small applications in C# programming language. CO-4-Create and manipulate GUI components in C#. CO-5-Create applications in C# using Object Oriented Properties. CO-6-Design and implement Database connectivity using ADO.Net. CO-7-Identify and resolver problems in C# applications.
Mapping between COs with PSOs	
Pre-requisite	Nil
Course Content	**Unit1: Overview of Microsoft .NET Platform** 1.1 Introduction to Building Blocks of .Net Platform 1.2 Overview of .Net Assemblies 1.3 Common Type System 1.4 Common Language Specification 1.5 Common Language Runtime 1.6 Exploring an Assembly (ildasm) 1.7 Platform Independent Nature of .Net 1.8 Base Class Libraries **Unit2: Overview of C#** 2.1 Literals, Variables, Data Types 2.2 Operators 2.3 Expressions and Looping 2.4 Constants, Arrays, Array Class, List 2.5 String, String Builder 2.6 Boxing and UnBoxing 2.7 Events, Errors and Exceptions **Unit3: Object Oriented Aspects of C#** 3.1 Creating Classes, Encapsulation, Object Construction & Destruction 3.2 Inheritance 3.3 Polymorphism 3.4 Abstraction 3.4 Interfaces and Abstract Classes 3.5 Delegates

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8
CO1	■		■	■			■	
CO2	■		■	■		■	■	
CO3	■	■		■		■	■	
CO4	■			■		■		
CO5				■	■	■	■	
CO6			■	■		■	■	
CO7		■	■	■	■		■	■

	Unit4: Application Development 4.1 Creating Windows Forms with Events and Controls 4.2 Menu Creation 4.3 Inheriting Windows Forms 4.4 SDI and MDI Application 4.5 Dialog Boxes (Modal and Modeless) 4.6 Validating Controls **Unit5: Accessing Data** 5.1 ADO.Net 5.1.1 Data Adapter 5.1.2 Data Set 5.1.3 Typed Data Set 5.2 Using Stored Procedures 5.3 Handling Exceptions 5.4 LINQ [Self Study] Report Generation, Deployment
Reference Books	1. .NET Framework Essentials, Hoand Lam, Thuan L. Thai, O'REILLY 2. Microsoft .NET Framework 4.5 Quickstart Cookbook, Jose Luis Latorre Millas, PACKT Publishing 3. Pro C# 5.0 and the .NET 4.5 Framework, Andrew Troelsen, Apress 4. C# IN DEPTH, Jon Skeet, Manning Publications 5. Beginning C# 7 Programming with Visual Studio 2017, Benjamin Perkins, wrox 6. Illustrated C#, Daniel Solis, Cal Schrotenboer, Apress 7. The C# Programmer's Study Guide, Ali Asad, Hamza Ali, Apress
Teaching Methodology	Class work, Discussion, Self-Study, Seminars and/or Assignment
Evaluation Method	30 % internal assessment and 70% external assessment

Course: 204: Python Programming Language

Course Code	**204**
Course Title	**Python Programming Language**
Credit	4
Teaching per Week	4 Hrs
Medium of Instruction	English
Minimum weeks per Semester	15 (Including Classwork, examination, preparation, holidays, etc.)
Effective From	June 2020
Purpose of Course	The Python language is used popularly among the people working in the area of Machine Learning (ML), Data Analytics, Artificial Intelligence, Web Application, and even the people working on Desktop Applications. This course imparts to the students understanding of Python programming language.
Course Objective	1. To make students understand Python Language 2. To make students understand various components of language and its Working 3. To prepare students to understand the use of language in the area of AI, ML, Data Analytics etc. 4. To make students understand the important
Course Outcome	CO1: Students will be able to Write, Test and Debug Python Programs. CO2: Students will be able to Implement Conditionals and Loops, use functions and represent Compound data using Lists, Tuples and Dictionaries in Python programs. CO3: Students will be able to Read and write data from & to files in Python and develop Real World Application. CO4: Students will be able to Design and implement programs to solve real-world problems using Python Programming Language. CO5: Students will learn essential packages like NumPy and Matplotlib, which are necessary for Machine Learning, Data Analytics, and AI.

Mapping between COs with PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8
CO1	■							
CO2				■			■	
CO3			■	■		■	■	
CO4			■			■	■	■
CO5	■		■	■		■	■	

Pre-requisite	Basic programming skills
Course Content	**Unit 1: Fundamentals of Python** 1.1 Features of Python 1.2 Python's Integrated Development and Learning Environment (IDLE) 1.3 Python identifiers 1.4 Python Operators 1.5 Python Datatypes 1.5.1 Numeric: integer, float, complex 1.5.2 Sequence: list, tuple, range 1.5.3 Sets 1.5.4 Texts 1.5.5 Binary: bytes, bytearray 1.5.6 Iterator 1.5.7 Mapping: dictionary 1.6 Input/Output in Python **Unit 2: Program Logic** 2.1 Control Structures 2.1.1 If, if..else, nested if, shorthand if, shorthand if..else 2.1.2 while loop 2.1.3 for loop 2.1.4 break, continue, pass 2.2 Functions in Python

	2.2.1 Function declaration 2.2.2 Passing arguments to function 2.2.3 Return values 2.2.4 Variable scope and name space 2.2.5 Lambda function 2.2.6 Recursive function 2.2.7 In-built function
	Unit 3: Files and Modules 3.1 File handling 3.1.1 Reading and writing to a file 3.1.2 Creation of new file 3.1.3 Deletion of a file 3.2 Python Modules 3.2.1 Creation of module 3.2.2 Importing a module 3.2.3 Date & time module **Unit 4: Exceptions, Class and Objects** 4.1 Exception Handling 4.1.1 try, catch, finally 4.2.2 Multiple error handling: except 4.2.3 Throwing a particular error: raise 4.2 Classes and Objects 4.2.1 Creation of class and object 4.2.2 The __init__() function 4.2.3 Self parameter 4.2.4 Modifying the property of a class 4.2.5 Inheritance & Encapsulation **Unit 5: Python Packages** 5.1 NumPy 5.1.1 Installing numpy 5.1.2 Numpy Array: dtype, shape, reshape, ndim, itemsize, empty, zeros, ones, fromiter, arrange, linespace 5.1.3 Indexing and slicing, broadcasting 5.1.4 Array manipulation: changing shapes, transpose, changing dimension, joining and splitting arrays, adding and removing elements 5.1.5 Mathematical functions and matrix library 5.2 Introduction to Matplotlib 5.2.1 Installing Matplotlib 5.2.2 Components of a plot 5.2.3 Drawing a plot 5.2.4 Drawing scatter diagram
Reference Books	1. Python Programming, Anurag Gupta, G Biswas,, Mc Graw Hill 2. Exploring Python, Timothy A. Budd, McGraw Hill Publication 3. Core Python Programming, R. Nageswara Rao, Dreamtech Press 4. Learning Python, 5th Edition, Mark Lutz, O'Reilly Media 5. Python Projects, Laura Cassell, Alan Gauld, Wrox Publication 6. NumPy: Beginner's Guide, 3rd Edition, Ivan Idris, Packt Publishing 7. NumPy Essentials, Leo Chin, Tanmay Dutta, Packt Publishing 8. Matplotlib 2.x By Example, Allen Yu, Claire Chung, Aldrin Yim, Packt Publishing
Teaching Methodology	Class work, Discussion, Self-Study, Seminars and/or Assignment
Evaluation Method	30% Internal assessment based on class attendance, participation, class test, quiz, assignment, seminar, internal examination, etc. & 70% External based on semester end University examination

Course: 205: iOS Development using Swift

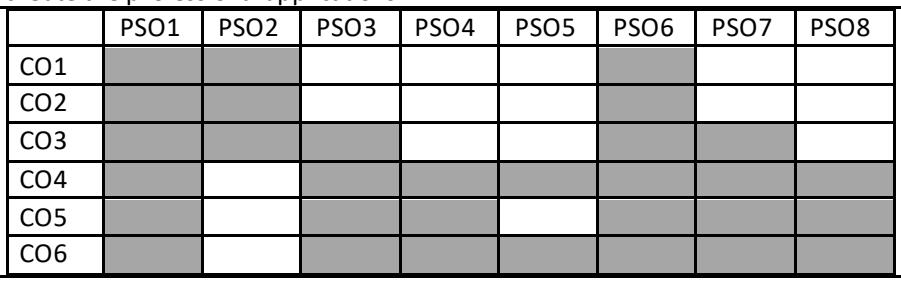
Course Code	**205**
Course Title	**iOS Development using Swift**
Credit	4
Teaching per Week	4 Hrs.
Minimum weeks/ Semester	15 (Including Class work, examination, preparation, holidays etc.)
Review / Revision	June 2020
Purpose of Course	This course will help the students to understand the fundamental as well as advanced concepts of iOS Programming. The course also provides them the skills necessary to develop an iOS Application from scratch to deploying it on the App Store.
Course Objective	The objective of the course is - 1. To understand the iOS ecosystem and tools for creating iOS applications 2. To explain advanced level concepts in iOS application design and development 3. To impart knowledge of Swift programming language
Course Outcome	CO1: Understand the iOS ecosystem and Xcode IDE. Understand the life cycle of iOS application and how to implement it in MVC. Understand Foundation framework in iOS. CO2: Understand the syntax, and semantics of the Swift programming language. Expose the students to CLI applications with Swift. CO3: Understand the UIKit framework in iOS. Understand the usage and working of UI elements in iOS application. Understanding various types of design and their implementation. CO4: Understand data persistence in mobile application. Understand working with files in iOS. Expose students with implementation and usage of database in an iOS application. CO5: Understand the usage and data extraction of sensors in iPhone. Expose the students with Location and MapKit Framework in iOS to build map-based applications. Expose the students with ad-hoc and App Store application deployment.
Mapping between COs with PSOs	

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8
CO1								
CO2								
CO3								
CO4								
CO5								

Pre-requisite	Knowledge of C, C++ and SQL
Course Content	**Unit 1: Introduction to iOS ecosystem** 1.1. Introduction to Xcode IDE 1.1.1. Environment setup 1.1.2. Editors, Storyboard and Simulator 1.2. Application Life-Cycle 1.3. View Controller Life-Cycle 1.4. Info.plist and App Permissions 1.5. MVC in iOS 1.6. Introduction to iOS App Frameworks 1.6.1. Foundation Framework 1.6.2. UIKit Framework 1.6.3. Swift and SwiftUI **Unit 2: Introduction to Swift Programming Language** 2.1. Simple Values – Constant and Variable 2.2. Control Flow 2.3. Functions and Closures 2.4. Objects and Classes 2.5. Enumerations and Structures

	2.6. Protocols and Extensions 2.7. Error Handling 2.8. Generics **Unit 3: UIKit: View Controllers, Views and Controls** 3.1. Text Views: UILabel, UITextField, UITextView 3.2. Controls: UIButton, UIDatePicker, UIPageControl, UISegmentedControl, UISlider, UIStepper, UISwitch 3.3. Content Views: UIActivityIndicatorView, UIImageView, UIPickerView, UIProgressView 3.4. Bars: UINavigationBar, UISearchBar, UIToolbar, UITabBar 3.5. Images and Video: UIImagePickerController 3.6. Container View Controllers: UINavigationController, UITabBarController 3.7. Container Views: Table Views, Collection Views 3.8. Alerts: UIAlertController 3.9. Gestures: UITapGestureRecognizer, UIPinchGestureRecognizer, UIRotationGestureRecognizer, UISwipeGestureRecognizer, UIPanGestureRecognizer **Unit 4: Data Persistence and Networking** 4.1. UserDefaults 4.2. FileManager 4.3. SQLite Framework 4.4. Core Data Framework 4.5. JSON Parsing 4.6. Working with URL and URL classes **Unit 5: App Services and App Deployment** 5.1. Core Motion – Accelerometer, Gyroscope, Pedometer, Magnetometer, Altitude 5.2. Core Location – CLLocationManager, CLLocation, Authorization 5.3. MapKit – Map Fundamentals, Map Coordinates, Annotations and Overlays 5.4. How to deploy an Ad-Hoc app – (diawi) 5.5. Publishing an app to the AppStore
Reference Books	1. Apple Documentation [developer.apple.com/documentation] 2. The Swift Programming Language by Apple Inc. [swift.org/documentation] 3. Hacking with Swift by Paul Hudson [hackingwithswift.com] 4. iOS 13 Programming Fundamental with Swift by Matt Neuberg, O'Reilly 5. Programming iOS 13 by Matt Neuberg, O'Reilly 6. Mastering Swift 5: Deep dive into the latest edition of the Swift programming language, 5th Edition, Packt Publishing Limited 7. Swiftui Essentials - IOS Edition: Learn to Develop IOS Apps Using Swiftui, Swift 5 and Xcode 11 by Neil Smyth, Payload Media, Inc. 8. Beginning iOS 13 & Swift App Development: Develop iOS Apps with Xcode 11, Swift 5, Core ML, ARKit and more by Greg Lim 9. Pro iPhone Development with Swift 5: Design and Manage Top Quality Apps by Wallace Wang, Apress
Teaching Methodology	Class work, Discussion, Self-study, Seminars and/or Assignment
Evaluation Method	30% Internal assessment is based on class attendance, participation, class test, quiz, assignment, seminar, internal examination etc. 70% assessment is based on semester end University External examination

Course: 205: Android Application Programming

Course Code	**205**
Course Title	**Android Application Programming**
Credit	4
Teaching per Week	4 Hrs.
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Review / Revision	June 2020
Purpose of Course	The purpose of the course is to make student capable of implementing the concepts, methods and tools of mobile applications development using Android.
Course Objective	To provide a thorough introduction to the Android environment and tools for creating Android applications.
Pre-requisite	Basic concepts of Operating Systems, Programming skills in core Java and Knowledge of object-oriented programming is desirable. Knowledge of XML format is helpful.
Course Outcome	CO1. Train students for installing and using the Android Developer's Toolkit such as SDK Manager, Android Virtual Device, Dalvik Debug Monitor Service (DDMS), Android Debug Bridge (ADB) and make them capable to develop, manage and maintain application(Apps) using Android and publish Apps on Google Play. CO2. Understand the Android software stack & program building blocks like activities, services and notifications to use them effectively to develop Android applications. CO3. Explain working with AndroidManifest, xml resources like layout and values and incorporate xml resources with Java code. CO4. Explain the use of java library for views, controls, menus, dialogs, graphics, media, storage, SQLiteDatabase etc… to make interactive applications. CO5. Train students to build Android app that access the database & other resources on web server using web services. CO6. Train students for managing Telephony and Message services. Make them capable of using Location Manager and Google MAP related APIs for App and create the professional applications.
Mapping between COs with PSOs	
Course Content	**Unit 1: Introduction to Android** 1.1 Evolution of Android and OHA 1.2 Architecture of Android OS 1.3 Introduction to Android SDK 1.4 Android Development tools: SDK Manager, Android Emulator, Android Virtual Device, Dalvik Debug Monitor Service (DDMS), Android Debug Bridge (ADB) 1.5 Anatomy of Android App: AndroidManifest.xml, Resources & R.java, Assets, Layouts & Drawable Resources **Unit 2: Working with User Interface in with Android Activity** 2.1 Widgets: Button, ImageButton, EditText, CheckBox, ToggleButton, RadioButton, RadioGroup, ProgressBar, Text Fields, ListView, Spinner 2.2 Designing UI Layouts: LinearLayout, RelativeLayout, TableLayout 2.3 Toast and Dialogs: AlertDialogs, TimePicker, DatePicker 2.4 Menus: Option menu, Context menu 2.5 Adapters for data binding: Array adapter, Cursor adapter 2.6 Event callback methods: onClick(), onLongClick() and onTouch()

	Unit 3: Android Application Components 3.1 Activity and Activity lifecycle 3.2 Intents and Intent Filters 3.3 Implicit Intent and Explicit Intent 3.4 Linking of Activity using Intent: startActivity() & startActivityForResult() 3.5 Fragments 3.6 Introduction to Service: life cycle, creation and destroy 3.7 Broadcast receiver & notification **Unit 4: Data Persistency in Android** 4.1 Shared preferences 4.2 File I/O Access: internal and external files 4.3 Working with SQLite Database - Performing insert, update, delete and query operations 4.4 Data access through web services (external databases) 4.5 Working with inbuilt Content Provider: CallLogs, Contacts **Unit 5: Telephony APIs, Sensors and Leveraging Google APIs** 5.1 Telephony APIs 5.1.1 Working with Telephony utilities 5.1.2 Sending and receiving SMS 5.2 Location and Map 5.2.1 Incorporating Location APIs 5.2.2 Incorporating Google map 5.2.3 Geocoding and reverse Geocoding
Reference Books	1. Beginning Android 4 Application Development, WEI-MENG LEE, WROX Publication-Wiley-India 2. Professional Android 4 Application Development by Reto Meier WROX Publication-Wiley-India 3. Android Programming Unleashed, B.M. Harwani, Sams Publishing 4. Beginning Android 4 Onur Cinar Apress Publication 5. Advanced Android Application Development, Fourth Edition, By Shane Conder, Lauren Darcey, Joseph Annuzzi Jr., Pearson
Teaching Methodology	Class work, Discussion, Self-Study, Seminars and/or Assignment
Evaluation Method	30 % internal assessment and 70% external assessment

Course: 206: Programming Skills IV

Course Code	**206**
Course Title	**Programming Skills IV**
Credit	2
Teaching per Week	2 Hrs.
Minimum weeks/ Semester	15 (Including Lab. work, examination, preparation, holidays etc.)
Review / Revision	June 2020
Purpose of Course	To understand the concepts of HTML, CSS, Front-end Scripting technologies
Course Objective	To teach the concepts of Front-end Scripting and its practical applications
Pre-requisite	Basic concepts of HTML, Web & Programming skills
Course Outcome	After completion of this course, the student will be able to design and develop web pages and Interactive UI for Web Applications
Course Content	Practical based on paper no 202 (Front End Technologies)
Reference Books	None
Teaching Methodology	Lab work, Practical Programming Exercises (to be documented in a separate journal), Self-study, and/or Assignment
Evaluation Method	30% Internal assessment & 70% External Assessment

Course: 207: Programming Skills V

Course Code	**207**
Course Title	**Programming Skills V**
Credit	2
Teaching per Week	2 Hrs.
Minimum weeks/ Semester	15 (Including Lab. work, examination, preparation, holidays etc.)
Review / Revision	June 2020
Purpose of Course	This course is an introduction to students to understand fundamentals of .NET technology. The course also gives students an idea about C#.NET Programming. The course also explains the concept of C#.NET
Course Objective	1. To make students understand .NET Technology 2. To make students understand C#.NET Programming 3. To make students understand the importance of C#.NET
Pre-requisite	Object Oriented Programming methodology
Course Outcome	After studying the course, students will be able to understand how .NET Technology works and the importance of object-oriented programming. This course will also help students to appreciate the C#.NET programming.
Course Content	Practical based on paper no 203 (Programming in .NET)
Reference Books	None
Teaching Methodology	Lab work, Practical Programming Exercises (to be documented in a separate journal), Self-study, and/or Assignment
Evaluation Method	30% Internal assessment & 70% External Assessment

Course: 208: Programming Skills VI

Course Code	**208**
Course Title	**Programming Skills VI**
Credit	2
Teaching per Week	2 Hrs.
Minimum weeks/ Semester	15 (Including Lab. work, examination, preparation, holidays etc.)
Review / Revision	June 2020
Purpose of Course	The purpose of the course is to make student capable of implementing the concepts, methods and tools of mobile applications development using Android/iOS.
Course Objective	To provide a thorough introduction to the Android/iOS environment and tools for creating Android/iOS applications.
Pre-requisite	Basic concepts of Operating Systems, Programming skills in core Java and Knowledge of object-oriented programming is desirable. Knowledge of XML format is helpful.
Course Outcome	After completion of this course, the student will be capable to develop, manage and maintain mobile device-based application using Android/iOS.
Course Content	Practical based on paper no 204 (Python Programming Language)
Reference Books	None
Teaching Methodology	Lab work, Practical Programming Exercises (to be documented in a separate journal), Self-study, and/or Assignment
Evaluation Method	30% Internal assessment & 70% External Assessment

Course: 209: Programming Skills VII

Course Code	**209**
Course Title	**Programming Skills VII**
Credit	2
Teaching per Week	2 Hrs.
Minimum weeks/ Semester	15 (Including Lab. work, examination, preparation, holidays etc.)
Review / Revision	June 2020
Purpose of Course	The purpose of the course is to make student capable of implementing the concepts, methods and tools of mobile applications development using Android/iOS.
Course Objective	To provide a thorough introduction to the Android/iOS environment and tools for creating Android/iOS applications.
Pre-requisite	Basic concepts of Operating Systems, Programming skills and knowledge of object-oriented programming is desirable
Course Outcome	After completion of this course, the student will be capable to develop, manage and maintain mobile device-based application using Android/iOS.
Course Content	Practical based on paper no. 205 (iOS Programming using Swift / Android Application Programming)
Reference Books	None
Teaching Methodology	Lab work, Practical Programming Exercises (to be documented in a separate journal), Self-study, and/or Assignment
Evaluation Method	30% Internal assessment & 70% External Assessment

Master of Computer Application

Name of Program	**Master of Computer Application**
Abbreviation	**MCA**
Duration	**2 Years**
Eligibility Criteria	Passed BCA/ Bachelor Degree in Computer Science Engineering or equivalent Degree. OR Passed B.Sc./ B.Com./ B.A. with Mathematics at 10+2 Level or at Graduation Level (with additional bridge Courses as per the norms of the concerned University).
Objective of Program	The core objective of the MCA programme is to prepare the students for productive career in software industry and academia by providing an outstanding environment of teaching and research in the core and emerging areas of the discipline.
Program Outcome	**PO1 : Fundamental Knowledge Enrichment** Program trains students with the core computer science and Information Technology (IT) knowledge domains. It also makes students capable of using core concepts in the conceptualization of domain specific application development. **PO2 : Critical Thinking Development** The program develops the skills of critical thinking, problem solving, evaluative learning of various techniques, and understanding the essence of the problem. **PO3 : Advanced Emerging Technology Awareness** The program trains students with the latest technologies that is being used in the industry. The continuous syllabi review adds value to the program for the outgoing students and make them ready to face challenging demands of the industry. **PO4 : Advanced Tools Usage** The program teaches the students to apply the advanced tools to solve real world problems. **PO5 : Nurturing Project Planning and Management Capabilities** The program trains students for designing and conceptualizing the software architecture, planning and managing the product development process of complex and live software projects. It also makes students understand the decision making for selection of an appropriate project management capabilities. **PO6 : Real World Problem / Project Development** Real world project provides the candidates exposure to work in the challenging and demanding environment of the industry. The project development training makes students employable and industry ready. **PO7 : Team Work and Leadership Development** Trains students to work in a team and also to take leadership of the of the project management team.
Program Specific Outcomes	PSO1 : Develop and strengthen the fundamental core concepts that are required to solve complex problems PSO2 : Develop the professional and entrepreneurship skills that needs independent logical and analytical thinking, teamwork and leadership PSO3 : Nurture the students to investigate for the design and development of a workable solution for a real world problem

PSO4 : Develop students for self-learning and practicing challenging problem solution
PSO5 : Train students to apply managerial skills to develop business applications.
PSO6 : Train students to use recent computer science and application domain specific knowledge
PSO7 : Train students to take-up the real world challenges to develop workable solution to a domain specific problem
PSO8 : Inculcate the passion for continuous learning and doing research for making a successful professional career

Mapping between POs and PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8
PO1								
PO2								
PO3								
PO4								
PO5								
PO6								
PO7								

Medium of Instruction: English

Program Structure: Semester 3 (Web Group)

Course Code	Title	Teaching per week		Course Credits	University Examination		Internal Marks	Total Marks
		Theory	Practical		Duration	Marks		
301	Internet of Things / Machine Learning	4	0	4	3 Hrs	70	30	100
302	Design Patterns	4	0	4	3 Hrs	70	30	100
303	Advanced Web Technologies / Advanced Java Technologies	4	0	4	3 Hrs	70	30	100
304	Full Stack Technology	4	0	4	3 Hrs	70	30	100
305	Open Source Web Based Programming	4	0	4	3 Hrs	70	30	100
306	Programming Skills VIII	0	3	3	2 Hrs	70	30	100
307	Programming Skills IX	0	3	3	2 Hrs	70	30	100
308	Programming Skills X	0	3	3	2 Hrs	70	30	100
309	Programming Skills XI	0	3	3	2 Hrs	70	30	100
	Total	20	12	32	23 Hrs	630	270	900

Program Structure: Semester 3 (Database Group)

Course Code	Title	Teaching per week		Course Credits	University Exam		Internal Exam	Total Marks
		Theory	Practical		Duration	Marks		
301	Internet of Things / Machine Learning	4	0	4	3 Hrs	70	30	100
302	Design Patterns	4	0	4	3 Hrs	70	30	100
303	ERP Using SAP / NoSQL Databases	4	0	4	3 Hrs	70	30	100
304	Advanced Database Administration	4	0	4	3 Hrs	70	30	100
305	Data Warehousing and Data Mining / Big Data	4	0	4	3 Hrs	70	30	100
306	Programming Skills XI	0	2	2	2 Hrs	70	30	100
307	Programming Skills XII	0	3	3	2 Hrs	70	30	100
308	Programming Skills XIII	0	2	2	2 Hrs	70	30	100
309	Programming Skills XIV	0	3	3	2 Hrs	70	30	100

Program Structure	Semester 3 (Network Group)						
Course Code	**Title**	**Teaching per week**		**Course Credits**	**University Exam**		**Internal Exam**	**Total Marks**
		Theory	**Practical**		**Duration**	**Marks**		
301	**Internet of Things / Machine Learning**	**4**	**0**	**4**	**3 Hrs**	**70**	**30**	**100**
302	**Design Patterns**	**4**	**0**	**4**	**3 Hrs**	**70**	**30**	**100**
303	**Network Essential & its Security**	**4**	**0**	**4**	**3 Hrs**	**70**	**30**	**100**
304	**Network Administration**	**4**	**0**	**4**	**3 Hrs**	**70**	**30**	**100**
305	**Wireless Network & Mobile Computing**	**4**	**0**	**4**	**3 Hrs**	**70**	**30**	**100**
306	**Programming Skills XI**	**0**	**2**	**2**	**2 Hrs**	**70**	**30**	**100**
307	**Programming Skills XII**	**0**	**3**	**3**	**2 Hrs**	**70**	**30**	**100**
308	**Programming Skills XIII**	**0**	**2**	**2**	**2 Hrs**	**70**	**30**	**100**
309	**Programming Skills XIV**	**0**	**3**	**3**	**2 Hrs**	**70**	**30**	**100**

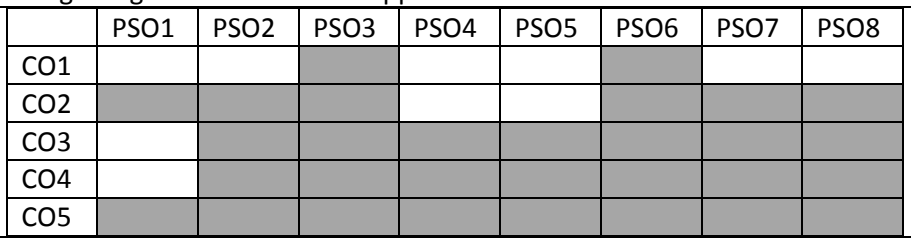
Program Structure	Semester 3 (General Group)						
Course Code	**Title**	**Teaching per week**		**Course Credits**	**University Exam**		**Internal Exam**	**Total Marks**
		Theory	**Practical**		**Duration**	**Marks**		
301	**Internet of Things / Machine Learning**	**4**	**0**	**4**	**3 Hrs**	**70**	**30**	**100**
302	**Design Patterns**	**4**	**0**	**4**	**3 Hrs**	**70**	**30**	**100**
303	**Network Essential & its Security**	**4**	**0**	**4**	**3 Hrs**	**70**	**30**	**100**
304	**Advanced Database Administration**	**4**	**0**	**4**	**3 Hrs**	**70**	**30**	**100**
305	**Open Source Web Based Programming**	**4**	**0**	**4**	**3 Hrs**	**70**	**30**	**100**
306	**Programming Skills XI**	**0**	**2**	**2**	**2 Hrs**	**70**	**30**	**100**
307	**Programming Skills XII**	**0**	**3**	**3**	**2 Hrs**	**70**	**30**	**100**
308	**Programming Skills XIII**	**0**	**2**	**2**	**2 Hrs**	**70**	**30**	**100**
309	**Programming Skills XIV**	**0**	**3**	**3**	**2 Hrs**	**70**	**30**	**100**

Program Structure	Semester 4			
Course Code	**Title**	**Course Credit**	**University Exam Marks**	**Internal Marks**	**Total Marks**
401	**Seminar**	6	70	30	100
402	**Project**	24	280	120	400

MCA 3rd Sem. (Web Group)

Course: 301: **Internet of Things (IoT)**

(Elective)

Course Code	301
Course Title	**Internet of Things (IoT)**
Credit	4
Teaching per Week	4 Hrs.
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Review / Revision	June 2021
Purpose of Course	This course is an introduction for students to IoT. The course also gives students an idea about various components of IoT and explain its working. The course also explains the role of embedded systems in IoT ecosystem.
Course Objective	The objective of the course is - 1. To make student understand IoT 2. To understand the working of Micro-Controller & Micro-Computer 3. To explain various types of sensors 4. To introduce students with Programming in IoT
Course Outcome	CO1: Understand the IoT ecosystem and architecture. Understand IoT standards and protocols. Understand the privacy, security and governance issues in IoT applications. Exposing students with IoT botnet and the risks involved with IoT based applications. CO2: Understand the overview and working of the various sensors used in IoT applications. Introduce and Explain various network communication protocols, standards and IoT data Protocols. Understanding the Wireless Sensor Network and how IoT devices communicate with each other. CO3: Understand Micro-Controller and its architecture. Understand the usage of Micro-Controller in IoT applications. Understand how Arduino and NodeMCU interact with sensors and communicate over the network. CO4: Understand Micro-Computer and its architecture. Understand the difference between a Micro-Controller and Micro-Computer. Understand the usage of Micro-Computer in IoT applications. Understand how Raspberry Pi interact with sensors and communicate over the network. CO5:Expose the students with Server-side development in IoT applications. Understand how to develop and deploy applications in Arduino and NodeMCU. Understand how Arduino and NodeMCU communicate among themselves, sensors and the server. Understand how to build a full IoT app by integrating them with mobile applications.
Mapping between COs with PSOs	
Pre-requisite	C , C++
Course Content	**Unit 1: Introduction to IoT** 1.1 What is IoT 1.2 IoT Applications 1.3 IoT Privacy and Security 1.3.1 Identification in Distributed Environment 1.3.2 Device Authentication 1.4 IoT Botnet **Unit 2: Networking and Communication** 2.1 Basics of Wireless Networking 2.1.1 CSMA/CA

	2.2 IoT Network Protocols 2.2.1 BLE, Zigbee, LoRaWAN, RFID 2.3 IoT Data Protocols 2.3.1 CoAP, MQTT, XMPP, DDS **Unit 3: Sensors** 3.1 Introduction to Sensors 3.2 Types of Sensors & their working 3.3 Wireless Sensor Network 3.3.1 Introduction to WSN 3.3.2 Applications 3.3.3 Characteristics 3.3.4 Challenges 3.3.5 Components 3.4 Wireless Adhoc Network Vs Wireless Sensor Network **Unit 4: Micro-Controller: Arduino, NodeMCU** 4.1 Introduction to Microcontrollers 4.2 Arduino IDE 4.3 Arduino Architecture 4.4 Arduino Pin Diagram 4.5 Introduction to NodeMCU 4.6 NodeMCU Specifications and Applications 4.7 NodeMCU ESP8266 Pinout **Unit 5: IoT App Interaction & Introduction to Raspberry Pi** 5.1 Uploading sensor data to server 5.2 Reading sensor data from server 5.3 Controlling IoT device and components from Mobile or Web 5.4 Introduction to Microcomputers 5.5 Raspberry Pi Architecture 5.6 Raspberry Pi Pinout
Reference Books	1) Getting Started with Internet of Things – By Cuno Pfister, O'Reilly 2) Learning Internet of Things – By Peter Waher , Packt Publication 3) Internet of Things : A Hands-on Approach – By Arshdip Bahga and Vijay Madisetti 4) IoT Governance, Privacy and Security Issues, IERC 5) IoT Fundamentals: Networking Technologies, Protocols and Use Cases for the Internet of Things, Cisco Press 6) Fundamentals of IoT Communication Technologies, Springer 7) Microcontrollers – Architecture, Programming, Interfacing and system design – By Raj Kamal , Pearson 8) Exploring C for Microcontrollers : A hands on approach, Springer 9) Arduino for Dummies, Wiley 10) Make: Getting Started With Arduino - The Open Source Electronics Prototyping Platform, Shroff/Maker Media 11) ESP8266: Get Started With ESP8266 Programming NodeMCU Using Arduino IDE, Createspace Independent Pub 12) Internet of Things Projects with ESP32, Packt Publishing Limited 13) Microprocessor Architecture, Programming and Applications with the 8085 - By Ramesh Gaonkar , Penram International Publishing 14) Raspberry Pi for Dummies , Wiley 15) Raspberry Pi User Guide – By Eben Upton and Garath Halfacree, Wiley
Teaching Methodology	Class work, Discussion, Self-Study, Seminars and/or Assignment
Evaluation Method	30% Internal assessment is based on class attendance, participation, class test, quiz, assignment, seminar, internal examination etc. 70% assessment is based on semester end University External examination

Course: 301: **Machine Learning**
(Elective)

Course Code	301
Course Title	**Machine Learning (ML)**
Credit	4
Teaching per Week	4 Hrs.
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Review / Revision	June 2021
Purpose of Course	This course is an introduction for students to ML. The course also gives students an idea about various methods and algorithms of Machine Learning and application development of ML.
Course Objective	The objective of the course is – 1. To make student understand ML 2. To understand the various Machine Learning method 3. To explain various algorithms used in Machine learning 4. To introduce students with Programming in ML
Course Outcome	CO1 : Explain to the students the fundamental know how like the types of machine learning algorithms, applications and various required libraries, model selection etc. required to implement machine learning algorithms. CO2 : Train students with can utilize various data wrangling techniques, data cleaning, data transformation, data reduction, data discretization, feature selection, and data visualization CO3 : Train students who can implement supervised learning algorithms utilizing regression and classification algorithm on the real world dataset. CO4 : Train student to have understanding of Artificial Neural Network and its working. Also, to make them capable of implementing ANN for solving real world problems using it. CO5 : Explain to the students to use clustering and association rules as unsupervised learning method to solve complex problems. CO6 : Train students to use machine learning techniques to solve real life complex problems.
Mapping between COs with PSOs	
Pre-requisite	Basics of Linear Algebra, Statistics and Mathematics, Python Programming
Course Content	**Unit 1 : Introduction** 1.1. Definition of Machine Learning 1.2 Types of Machine Learning : Supervised , Unsupervised and Semi-supervised 1.3 Applications and tools of Machine Learning (Scikit learn library) 1.4 Data Pre-processing, Selecting a model and training a model 1.5 Evaluating a performance of model and improving performance **Unit 2 : Data Wrangling** 2.1 Definition and goal of Data Wrangling 2.2 Importance of Data Wrangling 2.3 Data Pre-processing and Data Cleaning

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8
CO1	■							■
CO2	■	■	■			■	■	
CO3	■	■	■			■	■	
CO4	■	■	■			■	■	
CO5		■	■			■	■	
CO6		■	■	■	■	■	■	■

	2.3.1 Data Cleaning 2.3.2 Data Transformation 2.3.3 Data Reduction 2.3.4 Data Discretization 2.3.5 Feature Selection 2.4 Data Visualization **Unit 3 : Supervised Learning** 3.1 Supervised Learning : Classification and Regression 3.2 Regression 3.2.1 Simple and Multiple Regression 3.2.2 Linear Regression 3.2.3 Gradient Decent 3.2.4 Logistic Regression 3.3 Classification Algorithms : 3.3.1 K-nearest Neighbour 3.3.2 Support Vector Machines 3.3.3 Decision Trees 3.3.4 Naïve Bayes Classifier 3.4 Introduction to Support Vector Machine **Unit 4 : Neural Network** 4.1 Introduction to Neural Network 4.2 Architecture of Neural Network 4.3 Feedforward network and Backpropagation with example 4.4 Applications of Neural Network **Unit 5 : Unsupervised Learning** 5.1 Introduction to Unsupervised learning 5.2 Clustering 5.2.1 Selection of Clusters 5.2.2 Algorithms : 5.2.2.1 K – means clustering 5.2.2.2 Hierarchical Clustering 5.3 Association Rule Learning 5.3.1 Algorithms : 5.3.1.1 FP- Growth 5.3.1.2 Apriori Algorithm
Reference Books	1. “Machine Learning” by Tom M. Mitchell, McGraw Hill 2. “Understanding Machine Learning” by Shai Shalev-Shwartz, Shai Ben-David 3. “Machine Learning” by Anuradha Srinivasaraghavan, Vincy Joseph 4. “Machine Learning using Python” by U Dinesh Kumar Manaranjan Pradhan 5. “Real-World Machine Learning” by Henrik Brink, Joseph Richards, Mark Fetherolf 6. “Python Machine Learning” by Sebastian Raschka and Vahid Mirjalili 7. “Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems” by Aurelien Geron 8. “Machine Learning in Action” by Peter Harrington 9. “Introduction to Machine Learning with Python : A Guide for Data Scientists” by Andreas C. Muller, Sarah Guido
Teaching Methodology	Class work, Discussion, Self-Study, Seminars and/or Assignment
Evaluation Method	30% Internal assessment is based on class attendance, participation, class test, quiz, assignment, seminar, internal examination etc. 70% assessment is based on semester end University External examination

Course: 302: **Design Patterns**

<table>
<tr><td>Course Code</td><td>302</td></tr>
<tr><td>Course Title</td><td>Design Patterns</td></tr>
<tr><td>Credit</td><td>4</td></tr>
<tr><td>Teaching per Week</td><td>4 Hrs.</td></tr>
<tr><td>Minimum weeks per Semester</td><td>15 (Including Class work, examination, preparation, holidays etc.)</td></tr>
<tr><td>Review / Revision</td><td>June 2021</td></tr>
<tr><td>Purpose of Course</td><td>The purpose of the course is to make student understand how Patterns can be implemented in various object oriented programming languages to solve real world problems.</td></tr>
<tr><td>Course Objective</td><td>The objective of the course is -
1. To study various Design Patterns
2. How these Patterns can be used to design better systems through Object Oriented Programming Languages</td></tr>
<tr><td>Course Outcome</td><td>CO1: Explain students about the various design patterns; their categories, and purpose.
CO2: Explain the creational design patterns.
CO3: Explain the structural design patterns.
CO4: Explain the behavioural design patterns.
CO5: Explain some more design patterns used in IT industry currently.
CO6: Make students understand the applicability of design patterns practiced by IT companies and how effectively combine these patterns for effective software development.</td></tr>
<tr><td>Mapping between COs with PSOs</td><td>
<table>
<tr><th></th><th>PSO1</th><th>PSO2</th><th>PSO3</th><th>PSO4</th><th>PSO5</th><th>PSO6</th><th>PSO7</th><th>PSO8</th></tr>
<tr><td>CO1</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO2</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO3</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO4</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO5</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO6</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
</table>
</td></tr>
<tr><td>Pre-requisite</td><td>Object Oriented Programming, Software Engineering</td></tr>
<tr><td>Course Content</td><td>Unit -1 Creational Patterns
1.1 Singleton Pattern
1.2 Prototype Pattern
1.3 Builder Pattern
1.4 Factory Method Pattern
1.5 Abstract Factory Pattern

Unit-2 Structural Patterns
2.1 Proxy Pattern
2.2 Decorator Pattern
2.3 Adapter Pattern
2.4 Façade Pattern
2.5 Flyweight Pattern
2.6 Composite Pattern
2.7 Bridge Pattern

Unit-3 Behavioural Pattern
3.1 Visitor Pattern
3.2 Observer Pattern
3.3 Strategy Pattern
3.4 Template Method Pattern
3.5 Command Pattern
3.6 Iterator Pattern
3.7 Memento Pattern
3.8 State Pattern</td></tr>
</table>

	3.9 Mediator Pattern 3.10 Interpreter Pattern **Unit-4 Additional Design Patterns** 4.1 Simple Factory Pattern 4.2 Null Object Pattern 4.3 MVC Pattern **Unit-5 Pattern Applicability** 5.1 Security Patterns Repository 5.2 Patterns for Agile Development 5.3 Restful Service Patterns 5.4 Solution with semaphore 5.5 Patterns and Pattern combination in practice 5.6 Big Ball of Mud **Self-Study :** Pattern Languages
Reference Books	1. Design Patterns: Elements of Reusable Object-Oriented Software, Erich Gamma, Richard Helm, Ralph, John, Addision Wesley 2. Head First Design Patterns, Eric Freeman, O'Reilly 3. Design Patterns in C#, Vaskaran Sarcar, Apress 4. Design Patterns in Modern C++, Reusable Approaches for Object-Oriented Software Design, Dmitri Nesteruk, Apress 5. Modern C++ design: generic Programming and design patterns applied, Alexendrescu, Andrei, Addison-Wesley 6. Java Design Patterns: A Hands-on Experience with Real-World Examples, Vaskaran Sarcar, Apress
Teaching Methodology	Class work, Discussion, Self-Study, Seminars and/or Assignment
Evaluation Method	30 % internal assessment and 70% external assessment

Course: 303: **Advanced Web Technologies**

(Elective)

Course Code	303
Course Title	**Advanced Web Technologies**
Credit	4
Teaching per Week	4 Hrs.
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Last Review / Revision	June 2021
Purpose of Course	To teach ASP .Net for web application development
Course Objective	To impart knowledge of web application development using ASP .Net
Course Outcome	CO1 : Explain students the fundamental aspects of .NET framework and ASP.NET. CO2 : Train students to use various tools and controls available in ASP.NET in web application development, how to integrate them and to get them work as a part of one single unit. CO3 : Train students to work with database using ADO.NET through design alternatives and through coding as well, also learn the concept of Language Integrated Query. CO4 : Explain and train students to work with MVC architecture to adopt the requirements of modern days application development. Also learn component based development through web services and APIs. CO5 : Expose the students to the new era of .NET Core to understand the upgrades and developments in .NET architecture. CO6 : Explain students to utilize tools & techniques available in ASP.NET for web based application development and server side component development.
Mapping between COs with PSOs	

	PSO1	PSO2	PSO3	POS4	PSO5	PSO6	PSO7	PSO8
CO1	■							
CO2	■		■					
CO3	■	■	■			■		
CO4		■	■	■	■	■	■	■
CO5		■	■	■	■	■	■	■
CO6	■	■		■	■		■	■

Pre-requisite	Fundamental of .Net framework, HTML and C# desirable
Course Content	**Unit : 1 : Introduction to ASP.NET** 1.1. Overview of ASP.NET 1.2. Page Class and Page Life Cycle 1.3. Web Configuration files 1.4. Exception Handling 1.5. Error Pages 1.6. ASP.NET State Management 1.7. Introduction to Caching **Unit : 2 : ASP.NET Controls** 2.1 Web Controls 2.1.1 Common Web Server Controls 2.1.2 Specialized Web Server Controls 2.1.3 Table, Image, FileUpload 2.1.4 PostBack / Auto PostBack 2.2 Validation and Rich Controls 2.3 Website Navigation Controls - Sitemap, Treeview, Menu Controls 2.4 ASP.NET AJAX Controls 2.4.1 Introduction 2.4.2 Server Callbacks / Script Manager 2.4.3 ASP.NET AJAX Server Controls 2.4.4 UpdatePanel

	Unit : 3 : ASP .NET Web Application with Database 3.1. ADO.NET Architecture 3.2. Direct Data Access 3.3. Disconnected Data Access 3.4. Data Binding & Data Controls 3.4.1. Single-view, Repeated-Value, Data Source 3.4.2. Grid view ,Detail View, Form View 3.4.3. Data Repeater Control 3.5. LINQ 3.5.1. LINQ Language Features 3.5.2. LINQ to Objects 3.5.3. LINQ to SQL **Unit : 4 : ASP .NET - MVC , WCF Services & API** 4.1. ASP.NET MVC 4.1.1. MVC Architecture 4.1.2. URL Routing Engine 4.1.3. Wiring Controller, Model, and View 4.1.4. Data Access 4.1.5. Introduction to Entity Framework 4.2. WCF Services 4.2.1. Introduction to Web Services 4.2.2. RESTful API 4.2.3. Working with WCF Services 4.3. API 4.3.1. Introduction to JSON 4.3.2. Web API 4.3.3. API Creation and Consumption **Unit : 5 : ASP.NET CORE** 5.1. Overview of C#.NET CORE 5.2. .NET CORE Assemblies and Libraries 5.3. Pattern Matching 5.4. Tuples and Deconstruction 5.5. Local/Nested Functions 5.6. NuGet Package
Reference Book	1. Professional ASP.NET, Wrox Publication 2. ASP.NET – From Novice to Professional, Wrox Publication 3. ASP.NET Bible, By Mridula Parihar 4. Beginning ASP.NET 4.5, Wrox Publication 5. ASP.NET MVC with Entity Framework and CSS, APress 6. Programming Microsoft ASP.NET, Microsoft Press 7. Beginning AJAX with ASP.NET, Wrox Publication 8. Professional ASP.NET MVC 5, Wrox Publication 9. Professional C# 7 and .NET Core 2.0 , Wrox Publication 10. ASP.NET Core 2 Fundamentals, Packt Publication 11. Pro ASP.NET MVC 5, Apress 12. Programming ASP.NET Core, Microsoft Press 13. Pro C# 7 with .NET and .NET Core, Apress 14. Pro ASP.NET Core MVC by Adam Freeman, Springer, 2016
Teaching Methodology	Classroom, seminar and assignment
Evaluation Method	As per University rules

Course: 303: **Advanced Java Programming**

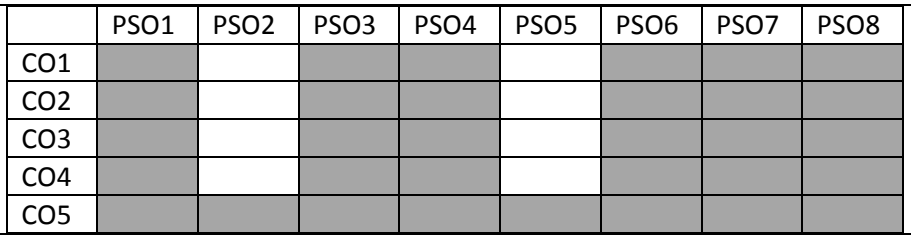
(Elective)

Course Code	303
Course Title	**Advanced Java Programming**
Credit	4
Teaching per Week	4 Hrs.
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Last Review / Revision	June 2021
Purpose of Course	This course is advance level java course to learn web & web enabled application development using Java Technologies.
Course Objective	To develop web application skills using Java web technology
Course Outcome	CO1 : Explain students the insight of the various aspects the Java web technologies. CO2 : learn to access database through Java programs, using Java Data Base Connectivity (JDBC). CO3 : Create dynamic web pages, using Servlets and JSP, make a reusable software component, using Java Bean. CO4 : Understand the multi-tier architecture of web-based enterprise applications using Enterprise JavaBeans (EJB), develop Session and Entity Beans CO5 : Expose the students with the analysis and development process of the web application development using Java. CO6 : Map Java classes and object associations to relational database tables with JPA
Mapping between COs with PSOs	
Pre-requisite	Core Java, Object oriented Programming
Course Content	**Unit 1: Advance JDBC (Java Database Connectivity)** 1.1 Types of JDBC Drivers 1.2 Connecting to databases like Access, MySQL, SqlServer, Oracle 1.3 Interacting with Database using SQL Queries 1.4 JDBC Objects: Connection, Resultset, Statement, Metadata 1.5 More JDBC Objects: DataSource, RowSet, RowSet events 1.6 Calling Stored Procedures 1.7 Managing Transactions 1.8 JDBC Connection Pooling 1.9 Handling Errors/Warning **Unit 2: Java Servlets** 2.1 Introduction to Servlets 2.2 Servlet Lifecycle 2.3 Handling HTTP GET and POST requests 2.4 Invoking other web resources 2.5 Maintaining client state 2.6 Servlet Annotations 2.7 Servlet Filter 2.8 File Upload

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8
CO1	■							
CO2	■		■					
CO3	■	■			■			
CO4		■	■	■		■		
CO5	■		■		■			
CO6	■	■		■		■	■	■

	Unit 3: Java Server Pages(JSP), JSTL (Standard Tag Library) & EL 3.1 Introduction to JSP, page lifecycle 3.2 JSP Elements – directives, scriplet, action 3.3 Implicit JSP objects 3.4 Using JavaBeans in JSP, Session Tracking 3.5 JSTL – Using Java Standard Tag Library 3.6 JSTL Core & Database tags 3.7 Introduction to EL (Expression Language) 3.8 EL implicit objects **Unit 4: Web Services with XML & JSON** 4.1 Introduction to Web Services 4.2 Building XML based web services with JAX-WS 4.3 Building Restful web services with JAX-RS 4.4 Reading/Writing XML files in Java (JAXP) 4.5 Introduction to AJAX **Unit 5: JPA, EJB & MVC Introduction** 5.1 Introduction to Java Persistence API (JPA) 5.2 Entity Beans & Session Beans 5.3 Overview of MVC Framework 5.4 Spring Architecture 5.5 Spring XML Configuration 5.6Aspect oriented programming
Reference Books	1. Java EE Tutorial Basic Concepts by Oracle Corporation 2. Beginning Java™ EE Platform with GlassFish™ : From Novice to Professional by Antonio Goncalves 3. Beginning EJB 3 Application Development From Novice to Professional by Raghu R.Kodali and Jonathan Wetherbee with Peter Zadrozny, Apress Publication 4. Pro JPA 2: Mastering the Java™ Persistence API 5. Head First Servlets and JSP By: Bryan Basham, Kathy Sierra, BertBates Publisher: 'Reilly Media 6. Core Servlets and Javaserver Pages: Author Marty Hall , LarryBrown, Sun Micro System 7. Java Servlet & JSP Cookbook by Bruce W. Perry O;reilly Publication 8. Beginning JSP™, JSF™ and Tomcat™ Web Development: FromNovice to Professional by Giulio Zambon and Michael Sekler 9. JAVA Complete Reference , TMH Publication 10. Professional Java Development with Spring Framework , WroxPublication
Teaching Methodology	Discussion, Independent Study, Seminars and Assignment
Evaluation Method	30% Internal assessment is based on class attendance, participation,class test, quiz, assignment, seminar, internal examination etc. 70% assessment is based on semester end University Externalexamination

Course: 304: **Full Stack Technology**

Course Code	304
Course Title	**Full Stack Technology**
Credit	4
Teaching per Week	4 Hrs.
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Review / Revision	June 2021
Purpose of Course	The purpose of the course is to make the students capable of developing full stack web applications.
Course Objective	The objective of the course is to make student understand full stack development
Course Outcome	CO1 : Understand the syntax, and semantics of the JavaScript programming language. Manipulate DOM elements with the help of JavaScript. CO2: Understand the design of single-page applications and how React facilitates their development. Understand advantages and disadvantages of using React. Understand functional components, state components, parent & child components, lifecycle, hooks, routing, and state management in React. CO3: Understanding the working of Node environment and Express Framework. Understand Server-side Web Application development and Server-side routing. CO4: Understanding Mongo as a data store. Understanding common use-cases and architectures of Mongo. Performing database operations using Mongo's query and update languages. CO5 : Expose the students with the combined development process of the full stack application. Understand connecting React and Node. Understand Github and CI/CD. Understand the deployment of full stack application using Netlify / Heroku.
Mapping between COs with PSOs	
Pre-requisite	HTML, CSS, Front-end Scripting
Course Content	**Unit 1: Introduction** 1.1 JavaScript 1.1.1 Execution Context and Call Stack 1.1.2 Hoisting in JavaScript 1.1.3 Spread Operator 1.1.4 Scope Chain, Temporal Dead Zone 1.1.5 Block Scope, Shadowing 1.1.6 Closures 1.2 Full Stack Technology **Unit 2: React.js** 2.1 React Introduction 2.1.1 What is React 2.1.2 What is a Component 2.1.3 JSX Overview 2.2 create-react-app 2.2.1 Understanding basics of react app 2.3 Understanding virtual DOM, SPA 2.4 Components 2.4.1 Class Components 2.4.2 Functional Components 2.4.3 Parent, Child Components

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8
CO1								
CO2								
CO3								
CO4								
CO5								

	2.4.4 Conditional Rendering 2.4.5 State, setState Method 2.4.6 Props 2.5 Event Handling in React 2.5.1 Event Handling in Class Components 2.5.2 Event Handling in Functional Components 2.6 Lifecycle 2.6.1 Class Component Life Cycle Methods 2.7 React Hooks 2.7.1 What is a React Hook 2.7.2 useState Hook 2.7.3 useEffect Hook 2.8 Building forms in React 2.9 React Router 2.10 Controlled vs Uncontrolled Components 2.11 State Management 2.11.1 Single Source of Truth 2.11.2 Lifting State Up 2.11.3 Prop Drilling 2.11.4 useContext 2.11.5 Redux 2.12 HTTP Methods 2.12.1 Fetch 2.12.2 Axios **Unit 3: Node.js & Express.js** 3.1 Introduction to Node.js 3.2 Creating a Simple Server 3.3 Response types - HTML, JSON 3.4 Modules 3.5 NPM 3.6 Introduction to Express.js 3.7 Express Params and Query String 3.8 Express Router **Unit 4: Mongo DB** 4.1 SQL/NoSQL landscape 4.2 Document Vs. Other types of Storage 4.3 MongoDB feature set 4.4 Introduction to BSON and JSON 4.5 Simple Queries 4.6 Connecting with Node JS 4.6.1 Inserts and Retrievals 4.6.2 Updates and Deletes **Unit 5: MERN & Deployment** 5.1 Connecting React and Node 5.2 Building an application in MERN 5.3 Github and CI/CD 5.4 Deploy using Netlify / Heroku
Reference Books	1. Eloquent JavaScript: A Modern Introduction to Programming, No Starch Press 2. You Don`t Know JS, Shroff/O'Reilly 3. The Road to Learn React: Your Journey to Master Plain Yet Pragmatic React.js, Zaccheus Entertainment 4. React Explained: Your Step-by-Step Guide to React, OS Training, LLC 5. Beginning React, Greg Lim

	6. Learning React: Functional Web Development with React and Redux, Shroff/O'Reilly 7. Learn React Hooks: Build and refactor modern React.js applications using Hooks, Packt Publishing Limited 8. Pro React, Apress 9. Web Development with Node and Express: Leveraging the JavaScript Stack, O'Reilly Media 10. Express in Action: Writing, building, and testing Node.js applications, Manning Publications 11. Beginning Node.js, Express & MongoDB Development, Greg Lim 12. MongoDB: The Definitive Guide - Powerful and Scalable Data Storage, Shroff/O'Reilly; Third edition 13. Full-Stack React Projects: Learn MERN stack development by building modern web apps using MongoDB, Express, React, and Node.js, Packt Publishing Limited 14. Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node, Apress
Teaching Methodology	Class work, Discussion, Self-Study, Seminars and/or Assignment
Evaluation Method	30% Internal assessment is based on class attendance, participation, class test, quiz, assignment, seminar, internal examination etc. 70% assessment is based on semester end University External examination

Course: 305: **Open Source Web Based Programming**

<table>
<tr><td>Course Code</td><td>305</td></tr>
<tr><td>Course Title</td><td>Open Source Web Based Programming</td></tr>
<tr><td>Credit</td><td>4</td></tr>
<tr><td>Teaching per Week</td><td>4 Hrs.</td></tr>
<tr><td>Minimum weeks/ Semester</td><td>15 (Including Class work, examination, preparation, holidays etc.)</td></tr>
<tr><td>Review / Revision</td><td>June 2021</td></tr>
<tr><td>Purpose of Course</td><td>This course helps students to understand fundamentals of Open Source web based Programming. The course also imparts students learning about Open source web based scripting language PHP and Mysql database. It also includes MVC or Three tier architecture of web based programming.</td></tr>
<tr><td>Course Objective</td><td>Student will learn fundamentals and advance topics of Open source Web technology</td></tr>
<tr><td>Course Outcome</td><td>CO1 : Explain students the fundamental as well as Advanced aspects of the Open Source Web based Technology.
CO2 : Train students about react JS and difference between React JS and React Native.
CO3 : Train students to understand MVC structure and it’s benefits.
CO4 : Explain and train students to deal with possible problem while developing websites and it’s solution.
CO5 : Expose the students with the analysis and development process of Websites.
CO6 : After studying the course, students will be able to understand how Open source web technology works. They will also be able to create database driven Professional Websites.</td></tr>
<tr><td>Mapping between COs with PSOs</td><td>
<table>
<tr><th></th><th>PSO1</th><th>PSO2</th><th>PSO3</th><th>PSO4</th><th>PSO5</th><th>PSO6</th><th>PSO7</th><th>PSO8</th></tr>
<tr><td>CO1</td><td>■</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO2</td><td>■</td><td></td><td>■</td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO3</td><td>■</td><td>■</td><td>■</td><td></td><td>■</td><td></td><td></td><td></td></tr>
<tr><td>CO4</td><td></td><td>■</td><td>■</td><td>■</td><td></td><td>■</td><td>■</td><td>■</td></tr>
<tr><td>CO5</td><td>■</td><td></td><td>■</td><td></td><td>■</td><td>■</td><td></td><td></td></tr>
<tr><td>CO6</td><td>■</td><td>■</td><td></td><td>■</td><td></td><td>■</td><td>■</td><td>■</td></tr>
</table>
</td></tr>
<tr><td>Pre-requisite</td><td>Knowledge of HTML and SQL</td></tr>
<tr><td>Course Content</td><td>Unit 1 : Introduction to Open source Web based Programming
1.1 Introduction to PHP & MySql
1.2 Installation of PHP and MySql
1.3 Language Characteristics & Features
1.4 Operators and Variables, Control Structures, Looping and Error handling
1.5 PHP functions
1.5.1 String Functions
1.5.2 Array Functions
1.5.3 Mathematical Functions
1.5.4 Graphics Library (GD Support)
1.5.6 Date and Time Functions
1.5.7 Misc. Function
1.6 State management Techniques
1.7 Object Oriented Features of PHP
1.7.1 Classes and Objects
1.7.2 Use of constructors
1.7.3 Serialization
1.7.4 Inheritance

Unit 2 : MySQL database server</td></tr>
</table>

	2.1 Configuring the MySQL Server 2.2 MySQL Tables, Displaying MySQL Database ,Adding and removing user access 2.3 Database connection and data processing functions **Unit 3 : Advance PHP** 3.1 Ajax Basics 3.1.1 HTTP Request and Response Fundamentals 3.1.2 The XMLHttpRequest Object XMLHttpRequest Methods 3.1.3 XMLHttpRequest Properties 3.1.4 Cross-Browser Usage Sending a Request to the Server 3.1.5 PHP and Ajax Client-Driven Communication 3.1.6 Server-Side Processing Expanding and Contracting Content 3.1.7 Form Validation 3.1.8 Ajax-Based Database Querying 3.2 XML 3.3 Web services **Unit 4 : MVC** 4.1 Introduction to MVC 4.2 Introduction to Laravel, Architecture Concepts 4.3 Routing 4.4 Middleware 4.5 Controllers 4.6 Request & Response 4.7 View 4.8 URL generation 4.9 Validation 4.10 Session & Cookie 4.11 Form & File uploading 4.12 Error Handling 4.13 Security 4.14 Database **Unit 5 : Advance Concepts of MVC** 5.1 Blade Templates 5.2 Mail 5.3 Authentication 5.4 Authorization 5.5 Encryption
Reference Books	1. Beginning PHP, Apache, MySQL Web Development - Elizabeth Naramore, Jason Gerner, Yann Le Scouarnec, Jeremy Stolz, Michael K. Glass, Gary Mailer – Wrox Publication 2. Professional PHP Programming - Jesus Castagnetto ,Wrox Press Ltd 3. Beginning PHP and MySQL: From Novice to Professional - W. Jason Gilmore, Apress 4. Php: The Complete Reference - Steven Holzner, Tata Mcgraw Hill Education Private Limited 5. AJAX and PHP: Building Responsive Web Applications - Bogdan Brinzarea, CristianDarie packtpub 6. Php manual – www. Php.com 7. Beginning Laravel Build Websites with Laravel 5.8 - Sanjib Sinha · 2019 - Apress 8. Laravel The Ultimate Beginner's Guide to Learn Laravel Step by Step, 2nd Edition - Mem Lnc, Rufus Stewart
Teaching Methodology	Class work, Discussion, Self-Study, Seminars and/or Assignment

Evaluation Method	30% Internal assessment is based on class attendance, participation, class test, quiz,assignment, seminar, internal examination etc. 70% assessment is based on semester end University External examination

Course: 306: **Programming Skills VIII**

Course Code	306
Course Title	**Programming Skills VIII**
Credit	3
Teaching per Week	2 Hrs.
Minimum weeks/ Semester	15 (Including Lab. work, examination, preparation, holidays etc.)
Review / Revision	June 2021
Purpose of Course	Learn practically IoT / ML
Course Objective	Learn IoT practically, understand the working of Micro-Controller & Micro-Computer and using various types of sensors and its Programming. OR Implement various algorithms used in Machine learning and introduce students with Programming in ML.
Pre-requisite	C/C++ for IoT OR Python Programming for ML
Course Outcome	After successful completion, students will be able to work with different types of Micro-Controllers, Micro-Computers and sensors for their IoT based application development. OR After successful completion, students will be able to work with different types of ML algorithms and ML based application development.
Course Content	Practical based on paper no 301. (IoT/ML) Separate journal to be prepared for this subject 301.
Reference Books	-----
Teaching Methodology	Lab work, Practical Programming Exercises (to be documented in a separate journal), Self-study, and/or Assignment
Evaluation Method	30% Internal assessment is based on Lab attendance, practical test, practical internal examination etc. 70% assessment is based on semester end University External practical examination

Course: 307: **Programming Skills IX**

Course Code	307
Course Title	**Programming Skills IX**
Credit	3
Teaching per Week	2 Hrs.
Minimum weeks/ Semester	15 (Including Lab. work, examination, preparation, holidays etc.)
Review / Revision	June 2021
Purpose of Course	This course helps students to implement the advanced concepts of .NET/Java practically.
Course Objective	Learning to implement the advanced topics of .NET/Java practically.
Pre-requisite	Practical programming in basic .NET/Java.
Course Outcome	After studying the course, students will be able to practically work on advanced technology platforms of .NET/Java.
Course Content	Practical based on paper no 303. Separate journal to be prepared for this subject 303.
Reference Books	-----
Teaching Methodology	Lab work, Practical Programming Exercises (to be documented in a separate journal), Self-study, and/or Assignment
Evaluation Method	30% Internal assessment is based on Lab attendance, practical test, practical internal examination etc. 70% assessment is based on semester end University External practical examination

Course: 308: **Programming Skills X**

Course Code	308
Course Title	**Programming Skills X**
Credit	3
Teaching per Week	2 Hrs.
Minimum weeks/ Semester	15 (Including Lab. work, examination, preparation, holidays etc.)
Review / Revision	June 2021
Purpose of Course	Learn practically, developing full stack web applications.
Course Objective	Practically learn full stack development
Pre-requisite	HTML, CSS, Front-end Scripting
Course Outcome	After completion of this course, the students will be able to design and develop Full Stack web applications
Course Content	Practical based on paper no 304. (Full Stack Technology) Separate journal to be prepared for this subject 304.
Reference Books	-----
Teaching Methodology	Lab work, Practical Programming Exercises (to be documented in a separate journal), Self-study, and/or Assignment
Evaluation Method	30% Internal assessment is based on Lab attendance, practical test, practical internal examination etc. 70% assessment is based on semester end University External practical examination

Course: 309: **Programming Skills XI**

Course Code	309
Course Title	**Programming Skills XI**
Credit	3
Teaching per Week	2 Hrs.
Minimum weeks/ Semester	15 (Including Lab. work, examination, preparation, holidays etc.)
Review / Revision	June 2021
Purpose of Course	This course helps students to implement the basic and advanced concepts of PHP/MySql practically.
Course Objective	Learning to develop and deploy websites using PHP/MySql practically.
Pre-requisite	Basic scripting, programming, html.
Course Outcome	After studying the course, students will be able to practically develop dynamic websites using PHP/MySql.
Course Content	Practical based on paper no 305. Separate journal to be prepared for this subject 305.
Reference Books	-----
Teaching Methodology	Lab work, Practical Programming Exercises (to be documented in a separate journal), Self-study, and/or Assignment
Evaluation Method	30% Internal assessment is based on Lab attendance, practical test, practical internal examination etc. 70% assessment is based on semester end University External practical examination

MCA 3rd Sem

(Database Group)

Course: 301: **Internet of Things (IoT)**

(Elective)

<table>
<tr><td>Course Code</td><td>301</td></tr>
<tr><td>Course Title</td><td>Internet of Things (IoT)</td></tr>
<tr><td>Credit</td><td>4</td></tr>
<tr><td>Teaching per Week</td><td>4 Hrs.</td></tr>
<tr><td>Minimum weeks per Semester</td><td>15 (Including Class work, examination, preparation, holidays etc.)</td></tr>
<tr><td>Review / Revision</td><td>June 2021</td></tr>
<tr><td>Purpose of Course</td><td>This course is an introduction for students to IoT. The course also gives students an idea about various components of IoT and explain its working. The course also explains the role of embedded systems in IoT ecosystem.</td></tr>
<tr><td>Course Objective</td><td>The objective of the course is -
5. To make student understand IoT
6. To understand the working of Micro-Controller & Micro-Computer
7. To explain various types of sensors
8. To introduce students with Programming in IoT</td></tr>
<tr><td>Course Outcome</td><td>CO1: Understand the IoT ecosystem and architecture. Understand IoT standards and protocols. Understand the privacy, security and governance issues in IoT applications. Exposing students with IoT botnet and the risks involved with IoT based applications.
CO2: Understand the overview and working of the various sensors used in IoT applications. Introduce and Explain various network communication protocols, standards and IoT data Protocols. Understanding the Wireless Sensor Network and how IoT devices communicate with each other.
CO3: Understand Micro-Controller and its architecture. Understand the usage of Micro-Controller in IoT applications. Understand how Arduino and NodeMCU interact with sensors and communicate over the network.
CO4: Understand Micro-Computer and its architecture. Understand the difference between a Micro-Controller and Micro-Computer. Understand the usage of Micro-Computer in IoT applications. Understand how Raspberry Pi interact with sensors and communicate over the network.
CO5:Expose the students with Server-side development in IoT applications. Understand how to develop and deploy applications in Arduino and NodeMCU. Understand how Arduino and NodeMCU communicate among themselves, sensors and the server. Understand how to build a full IoT app by integrating them with mobile applications.</td></tr>
<tr><td>Mapping between COs with PSOs</td><td>
<table>
<tr><td></td><td>PSO1</td><td>PSO2</td><td>PSO3</td><td>PSO4</td><td>PSO5</td><td>PSO6</td><td>PSO7</td><td>PSO8</td></tr>
<tr><td>CO1</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO2</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO3</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO4</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO5</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
</table>
</td></tr>
<tr><td>Pre-requisite</td><td>C , C++</td></tr>
<tr><td>Course Content</td><td>Unit 1: Introduction to IoT
1.1 What is IoT
1.2 IoT Applications
1.3 IoT Privacy and Security
1.3.1 Identification in Distributed Environment
1.3.2 Device Authentication
1.4 IoT Botnet</td></tr>
</table>

	Unit 2: Networking and Communication 2.1 Basics of Wireless Networking 2.1.1 CSMA/CA 2.2 IoT Network Protocols 2.2.1 BLE, Zigbee, LoRaWAN, RFID 2.3 IoT Data Protocols 2.3.1 CoAP, MQTT, XMPP, DDS **Unit 3: Sensors** 3.1 Introduction to Sensors 3.2 Types of Sensors & their working 3.3 Wireless Sensor Network 3.3.1 Introduction to WSN 3.3.2 Applications 3.3.3 Characteristics 3.3.4 Challenges 3.3.5 Components 3.4 Wireless Adhoc Network Vs Wireless Sensor Network **Unit 4: Micro-Controller: Arduino, NodeMCU** 4.1 Introduction to Microcontrollers 4.2 Arduino IDE 4.3 Arduino Architecture 4.4 Arduino Pin Diagram 4.5 Introduction to NodeMCU 4.6 NodeMCU Specifications and Applications 4.7 NodeMCU ESP8266 Pinout **Unit 5: IoT App Interaction & Introduction to Raspberry Pi** 5.1 Uploading sensor data to server 5.2 Reading sensor data from server 5.3 Controlling IoT device and components from Mobile or Web 5.4 Introduction to Microcomputers 5.5 Raspberry Pi Architecture 5.6 Raspberry Pi Pinout
Reference Books	1) Getting Started with Internet of Things – By Cuno Pfister, O'Reilly 2) Learning Internet of Things – By Peter Waher , Packt Publication 3) Internet of Things : A Hands-on Approach – By Arshdip Bahga and Vijay Madisetti 4) IoT Governance, Privacy and Security Issues, IERC 5) IoT Fundamentals: Networking Technologies, Protocols and Use Cases for the Internet of Things, Cisco Press 6) Fundamentals of IoT Communication Technologies, Springer 7) Microcontrollers – Architecture, Programming, Interfacing and system design – By Raj Kamal , Pearson 8) Exploring C for Microcontrollers : A hands on approach, Springer 9) Arduino for Dummies, Wiley 10) Make: Getting Started With Arduino - The Open Source Electronics Prototyping Platform, Shroff/Maker Media 11) ESP8266: Get Started With ESP8266 Programming NodeMCU Using Arduino IDE, Createspace Independent Pub 12) Internet of Things Projects with ESP32, Packt Publishing Limited

	13) Microprocessor Architecture, Programming and Applications with the 8085 - By Ramesh Gaonkar , Penram International Publishing 14) Raspberry Pi for Dummies , Wiley 15) Raspberry Pi User Guide – By Eben Upton and Garath Halfacree, Wiley
Teaching Methodology	Class work, Discussion, Self-Study, Seminars and/or Assignment
Evaluation Method	30% Internal assessment is based on class attendance, participation, class test, quiz, assignment, seminar, internal examination etc. 70% assessment is based on semester end University External examination

Course: 301: **Machine Learning**

(Elective)

<table>
<tr><td>Course Code</td><td>301</td></tr>
<tr><td>Course Title</td><td>Machine Learning (ML)</td></tr>
<tr><td>Credit</td><td>4</td></tr>
<tr><td>Teaching per Week</td><td>4 Hrs.</td></tr>
<tr><td>Minimum weeks per Semester</td><td>15 (Including Class work, examination, preparation, holidays etc.)</td></tr>
<tr><td>Review / Revision</td><td>June 2021</td></tr>
<tr><td>Purpose of Course</td><td>This course is an introduction for students to ML. The course also gives students an idea about various methods and algorithms of Machine Learning and application development of ML.</td></tr>
<tr><td>Course Objective</td><td>The objective of the course is –
1. To make student understand ML
2. To understand the various Machine Learning method
3. To explain various algorithms used in Machine learning
4. To introduce students with Programming in ML</td></tr>
<tr><td>Course Outcome</td><td>CO1 : Explain to the students the fundamental know how like the types of machine learning algorithms, applications and various required libraries, model selection etc. required to implement machine learning algorithms.
CO2 : Train students with can utilize various data wrangling techniques, data cleaning, data transformation, data reduction, data discretization, feature selection, and data visualization
CO3 : Train students who can implement supervised learning algorithms utilizing regression and classification algorithm on the real world dataset.
CO4 : Train student to have understanding of Artificial Neural Network and its working. Also, to make them capable of implementing ANN for solving real world problems using it.
CO5 : Explain to the students to use clustering and association rules as unsupervised learning method to solve complex problems.
CO6 : Train students to use machine learning techniques to solve real life complex problems.</td></tr>
<tr><td>Mapping between COs with PSOs</td><td>
<table>
<tr><th></th><th>PSO1</th><th>PSO2</th><th>PSO3</th><th>PSO4</th><th>PSO5</th><th>PSO6</th><th>PSO7</th><th>PSO8</th></tr>
<tr><td>CO1</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO2</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO3</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO4</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO5</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO6</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
</table>
</td></tr>
<tr><td>Pre-requisite</td><td>Basics of Linear Algebra, Statistics and Mathematics, Python Programming</td></tr>
<tr><td>Course Content</td><td>Unit 1 : Introduction
1.1. Definition of Machine Learning
1.2 Types of Machine Learning : Supervised , Unsupervised and Semi-supervised
1.3 Applications and tools of Machine Learning (Scikit learn library)</td></tr>
</table>

	1.4 Data Pre-processing, Selecting a model and training a model 1.5 Evaluating a performance of model and improving performance **Unit 2 : Data Wrangling** 2.1 Definition and goal of Data Wrangling 2.2 Importance of Data Wrangling 2.3 Data Pre-processing and Data Cleaning 2.3.1 Data Cleaning 2.3.2 Data Transformation 2.3.3 Data Reduction 2.3.4 Data Discretization 2.3.5 Feature Selection 2.4 Data Visualization **Unit 3 : Supervised Learning** 3.1 Supervised Learning : Classification and Regression 3.2 Regression 3.2.1 Simple and Multiple Regression 3.2.2 Linear Regression 3.2.3 Gradient Decent 3.2.4 Logistic Regression 3.3 Classification Algorithms : 3.3.1 K-nearest Neighbour 3.3.2 Support Vector Machines 3.3.3 Decision Trees 3.3.4 Naïve Bayes Classifier 3.4 Introduction to Support Vector Machine **Unit 4 : Neural Network** 4.1 Introduction to Neural Network 4.2 Architecture of Neural Network 4.3 Feedforward network and Backpropagation with example 4.4 Applications of Neural Network **Unit 5 : Unsupervised Learning** 5.1 Introduction to Unsupervised learning 5.2 Clustering 5.2.1 Selection of Clusters 5.2.2 Algorithms : 5.2.2.1 K – means clustering 5.2.2.2 Hierarchical Clustering 5.3 Association Rule Learning 5.3.1 Algorithms : 5.3.1.1 FP- Growth 5.3.1.2 Apriori Algorithm
Reference Books	1. “Machine Learning” by Tom M. Mitchell, McGraw Hill 2. “Understanding Machine Learning” by Shai Shalev-Shwartz, Shai Ben-David 3. “Machine Learning” by Anuradha Srinivasaraghavan, Vincy Joseph

	4. “Machine Learning using Python” by U Dinesh Kumar Manaranjan Pradhan 5. “Real-World Machine Learning” by Henrik Brink, Joseph Richards, Mark Fetherolf 6. “Python Machine Learning” by Sebastian Raschka and Vahid Mirjalili 7. “Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems” by Aurelien Geron 8. “Machine Learning in Action” by Peter Harrington 9. “Introduction to Machine Learning with Python : A Guide for Data Scientists” by Andreas C. Muller, Sarah Guido
Teaching Methodology	Class work, Discussion, Self-Study, Seminars and/or Assignment
Evaluation Method	30% Internal assessment is based on class attendance, participation, class test, quiz, assignment, seminar, internal examination etc. 70% assessment is based on semester end University External examination

Course: 302: **Design Patterns**

Course Code	302
Course Title	**Design Patterns**
Credit	4
Teaching per Week	4 Hrs.
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Review / Revision	June 2021
Purpose of Course	The purpose of the course is to make student understand how Patterns can be implemented in various object oriented programming languages to solve real world problems.
Course Objective	The objective of the course is - 3. To study various Design Patterns 4. How these Patterns can be used to design better systems through Object Oriented Programming Languages
Course Outcome	CO1: Explain students about the various design patterns; their categories, and purpose. CO2: Explain the creational design patterns. CO3: Explain the structural design patterns. CO4: Explain the behavioural design patterns. CO5: Explain some more design patterns used in IT industry currently. CO6: Make students understand the applicability of design patterns practiced by IT companies and how effectively combine these patterns for effective software development.
Mapping between COs with PSOs	See table below
Pre-requisite	Object Oriented Programming, Software Engineering
Course Content	**Unit -1 Creational Patterns** 1.1 Singleton Pattern 1.2 Prototype Pattern 1.3 Builder Pattern 1.4 Factory Method Pattern 1.5 Abstract Factory Pattern **Unit-2 Structural Patterns** 2.1 Proxy Pattern 2.2 Decorator Pattern 2.3 Adapter Pattern 2.4 Façade Pattern 2.5 Flyweight Pattern 2.6 Composite Pattern 2.7 Bridge Pattern **Unit-3 Behavioural Pattern** 3.1 Visitor Pattern 3.2 Observer Pattern 3.3 Strategy Pattern 3.4 Template Method Pattern

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8
CO1	■							
CO2	■	■	■			■	■	
CO3	■	■	■			■	■	
CO4	■	■	■			■	■	
CO5	■		■	■	■	■	■	
CO6							■	■

	3.5 Command Pattern 3.6 Iterator Pattern 3.7 Memento Pattern 3.8 State Pattern 3.9 Mediator Pattern 3.10 Interpreter Pattern **Unit-4 Additional Design Patterns** 4.1 Simple Factory Pattern 4.2 Null Object Pattern 4.3 MVC Pattern **Unit-5 Pattern Applicability** 5.1 Security Patterns Repository 5.2 Patterns for Agile Development 5.3 Restful Service Patterns 5.4 Solution with semaphore 5.5 Patterns and Pattern combination in practice 5.6 Big Ball of Mud **Self-Study :** Pattern Languages
Reference Books	7. Design Patterns: Elements of Reusable Object-Oriented Software, Erich Gamma, Richard Helm, Ralph, John, Addision Wesley 8. Head First Design Patterns, Eric Freeman, O'Reilly 9. Design Patterns in C#, Vaskaran Sarcar, Apress 10. Design Patterns in Modern C++, Reusable Approaches for Object-Oriented Software Design, Dmitri Nesteruk, Apress 11. Modern C++ design: generic Programming and design patterns applied, Alexendrescu, Andrei, Addison-Wesley 12. Java Design Patterns: A Hands-on Experience with Real-World Examples, Vaskaran Sarcar, Apress
Teaching Methodology	Class work, Discussion, Self-Study, Seminars and/or Assignment
Evaluation Method	30 % internal assessment and 70% external assessment

Course: 303: **ERP Using SAP**

Course Code	303
Course Title	**ERP Using SAP**
Credit	4
Teaching per Week	4 Hrs.
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Last Review / Revision	June 2020
Purpose of Course	This course imparts fundamental as well as detailed Enterprise Resource Planning using SAP technology platform.
Course Objective	Learn ERP applications and its programming
Course Outcome	CO1 : Explain students the insight of the fundamental aspects of the ERP and SAP. CO2 : Train students to represent declarative knowledge of Sap Application Server and 3-Tier Architecture. CO3 : Train students to understand concepts of Data Dictionary and data structure in ABAP. CO4 : Explain and train students to understand basic concepts of Modularization Technique, Module pool programing, BDC & LSMW and Selection-Screen programming. CO5 : The course emphasizes on teaching SAP implementation as the whole process of transforming ERP business procedures to organization wide requirements. CO6 : After studying this students will be able to understand how to work with ERP modules. After successful completion, students will be able to develop ERP application using SAP.
Mapping between COs with PSOs	

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8
CO1								
CO2								
CO3								
CO4								
CO5								
CO6								

Pre-requisite	Fundamentals of ERP, DBMS
Course Content	**Unit 1: Introduction** 1.1 Introduction to ERP 1.2 Introduction to SAP 1.3 Example: How SAP works in an Organization **Unit 2: Architecture of SAP Application Server** 2.1 3-Tier Architecture 2.2 Application Servers 2.3 Work processes and its Type **Unit 3: Data Dictionary & Data Structures in ABAP** 3.1 Introduction to Data dictionary 3.2 Different Types of Data structures 3.3 Internal Tables and its operation

	Unit 4: Modularization Techniques 4.1 Include Programs 4.2 Subroutines 4.3 Function Module. 4.4 Types of Function Module(Simple, RFC enabled, BAPI) **Unit 5: List Report, ALV Report** 5.1 Simple List Report 5.2 Interactive List Report 5.3 Events in List Reports 5.4 Field catalog generation in ALV 5.5 Operation on ALV(Sorting, Filtering, Totals, Subtotals, Download, Hide Columns) **Unit 6: Module pool programming / Screen Programming** 6.1 Screen Elements(Simple & Complex) 6.2 Screen Events(PBO/PAI) 6.3 Transactions **Unit 7: Selection-Screen programming** 7.1 Defining Selection Screen. 7.2 User Actions on Selection Screen. 7.3 Events of Selection Screen **Unit 8: Smartform/Sapscript** 8.1 Form printing with smartform 8.2 Form printing with Sapscript **Unit 9: BDC & LSMW** 9.1 Data upload through BDC 9.2 Data upload through LSMW **Unit 10: Enhancement(Exits & BADI)** 10.1 What is Enhancement 10.2 User-Exits 10.3 BADI(Business Add-in)
Reference Books	1. ABAP Cookbook by James Wood 2. BC - ABAP Programming from SAP-AG 3. Teach Yourself ABAP/4 in 21 Days by Ken Greenwood, SAMS 4. SAP Smart Forms by Christoph Wachter, Werner Hertleif 5. SAPscript by Michaelson Buchanan 6. Developing Sap's R/3 Application with Abap/4 7. Data Migration Made Easy - R/3 Simplications Group, SAP Labs 8. ABAP Development for SAP NetWeaver BW: Exits, BAdIs, and Enhancements by Dirk Herzog

	9. Next Generation ABAP Development (2nd Edition) by Rich Heilman and Thomas Jung
Teaching Methodology	Discussion, Independent Study, Seminars and Assignment
Evaluation Method	30% Internal assessment is based on class attendance, participation, class test, quiz, assignment, seminar, internal examination etc. 70% assessment is based on semester end University External examination

Course: 303: **NoSQL Databases**

Course Code	303
Course Title	**NoSQL Databases**
Credit	4
Teaching per Week	4
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Last Review / Revision	June 2020
Purpose of Course	To teach the emerging trends in NoSQL databases
Course Objective	To impart knowledge of NoSQL Databases
Course Outcome	CO1 : Explain students about history, concept , characteristics and types of NoSQL databases CO2: Explain students about the differences between relational databases and NoSQL databases, Advantages and disadvantages of NoSQL databases and application of NoSQL databases CO3: Explain students about fundamentals of MongoDB, MongoDB feature set and Architecture CO4: Train student to create document, collection and databases in MongoDB, use of simple and complex queries to insert, update and view data. CO5: Explain and train student to use MongoDB restful API and applying security CO6: Explain students about Cassandra Architecture, Data modelling in Cassandra and integration of Cassandra with Hadoop CO7: Train student to create and use keyspaces and databases in Cassandra, Also explain and train student about Cassandra administration
Mapping between COs with PSOs	

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8
CO1	■							
CO2	■	■	■					
CO3	■							
CO4		■	■	■	■	■	■	
CO5		■	■	■	■	■	■	■
CO6	■	■	■	■				
CO7		■	■	■	■	■	■	■

Pre-requisite	Fundamentals of DBMS
Course Content	**Unit 1: Introduction** 1.1 History 1.2 Concepts and Characteristics of NoSQL databases 1.3 Primary benefits of NoSQL databases **Unit 2: MongoDB** 2.1 SQL/NoSQL landscape 2.2 Document Vs. Other types of Storage 2.3 MongoDB feature set 2.4 Introduction to BSON and JSON 2.5 MongoDB Architecture 2.6 Documents and Collections 2.6.1 Creating Documents

	2.6.2 Managing Documents in collections 2.6.3 Iterating over Documents 2.7 Queries 2.7.1 Simple Queries 2.7.2 Complex Queries 2.7.2.1 Existential field values 2.7.2.2 Aggregations and groups 2.7.2.3 Aggregations and groups in hierarchical data 2.8 Updates and Deletes 2.9 Updates and Arrays 2.10 Indexing 2.11 MongoDB RESTful API 2.12 MongoDB Security 2.13 MongoDB Replication and Sharing 2.14 Introduction to MapReduce **Unit 3: Cassandra** 3.1 Cassandra Architecture 3.1.1 Cassandra P2P Architecture 3.1.2 Clustering Structures- Nodes 3.1.3 Rings 3.1.4 Virtual Nodes 3.1.5 Consistency & Hashing 3.1.6 Gossip Protocol 3.1.7 Data Replication 3.1.8 Replication Factors & Indexes 3.1.9 Tunable Consistency 3.1.10 High & Rapid Scalability Memtables, SStables & Commitlogs 3.1.11 Repairs 3.1.12 Tombstones 3.1.13 Repairs 3.1.14 Replication Factors 3.1.15 Compaction and Anti-Entropy 3.1.16 Bloom Filters 3.2 Data Modelling in Cassandra 3.3 Cassandra Administration 3.4 CQL3 3.5 Integration with Hadoop
Reference Books	1. Chodorow, K. (2013). MongoDB: The Definitive Guide (2nd ed.). Upper Saddle River, NJ: Pearson Education, Inc. ISBN-13: 978-1449344689 ISBN-10: 1449344682. 2. Shashank Tiwari, Professional NoSQL, Sierra Nevada Books, ISBN-13: 978-0470942246 3. Amol Nayak, Instant MongoDB, Packt Publishing Limited, 2013, ISBN-13**:** 978-1782169703 4. Kristina Chodorow, MongoDB Definitive Guide 2e, O'Reilly, 2013, ISBN-13: 978-1449344689

	5. Eben Hewitt, Cassandra Definitive Guide, O'Reilly, 2010, ISBN:ISBN 10:1-4493-9041-2
Teaching Methodology	Classroom, seminar and assignment
Evaluation Method	30% Internal assessment is based on class attendance, participation, class test, quiz, assignment, seminar, internal examination etc. 70% assessment is based on semester end University External examination

Course: 304: **Advanced Database Administration**

Course Code	304
Course Title	**Advanced Database Administration**
Credit	4
Teaching per Week	4 Hrs.
Minimum weeks/ Semester	15 (Including Class work, examination, preparation, holidays etc.)
Review / Revision	June 2020
Purpose of Course	Understanding advanced database administration
Course Objective	To learn advanced database administration, database tuning and maintenance
Course Outcome	CO1 : To provide strong foundation in Advanced Database Administration concepts from an industry perspective. CO2 : To have thorough understanding of Oracle Database Management System internal architecture. CO3 : To understand the security aspects and user management. CO4 : To apply and learn various Oracle utilities CO5 : To learn how to practically tune the database to optimize the overall performance. CO6 : To learn and implement Backup and Recovery.
Mapping between COs with PSOs	

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8
CO1								
CO2								
CO3								
CO4								
CO5								
CO6								

Pre-requisite	RDBMS
Course Content	**Unit 1. Oracle10g Instance creation and management** 1.1 Oracle Instance 1.2 Installing Oracle 1.3 Oracle Optimal Flexible Architecture (OFA) 1.4 Locating initialization, listener.ora & sqlnet.ora files 1.5 Finding the alert log 1.6 Common environment variables 1.7 Structures in an Oracle Instance 1.8 Oracle Memory Structures, SGA and PGA 1.9 Oracle Processes and their purposes 1.10 Startup, nomount, mount and open database commands **Unit 2. Oracle10g Database Architecture** 2.1 Oraclde10g management framework 2.1 Using the Database Creation Assistant (DBA) 2.3 Creating and dropping a database 2.4 Tablespaces 2.5 Tables and Indexes 2.6 Clusters 2.7 Partitioning of Tables and Indexes 2.8 Gathering and applying patches **Unit 3. Concurrency Management** 3.1 Transactions, serialization, locks and latches 3.2 Lock modes 3.3 Detecting and resolving lock conflicts

	3.4 Managing deadlocks **Unit 4. Interfacing with Oracle** 4.1 Oracle transaction management 4.2 Using SQL*Plus and iSQL*Plus 4.3 Using embedded Oracle with Pro*C & Java 4.4 PL/SQL & Triggers 4.5 Pining PL/SQL packages & compiling PL/SQL 4.6 System-level triggers – startup trigger, logon trigger, PL/SQL error trigger **Unit 5. Oracle*Net** 5.1 Basic Network structure 5.2 Oracle*Net Files 5.3 Multi-threaded server 5.4 Create additional listeners 5.5 Create Oracle Net service aliases 5.6 Configure connect time failover 5.7 Oracle*Net names resolution **Unit 6. Tablespace Management Overview** 6.1 Dictionary Managed Tablespaces 6.2 Locally Managed Tablespaces 6.3 Automatic Segment Space Management 6.4 Moving tablespaces online and offline **Unit 7. UNDO Tablespace Management** 7.1 Use of undo segments 7.2 Creating an undo tablespace 7.3 User managed undo tablespaces 7.4 Automatic undo management 7.5 Monitor & Configure undo retention 7.6 Use the Undo Advisor 7.7 Size the undo tablespace **Unit 8. Oracle Utilities** 8.1 Datapump - Import/export 8.2 SQL*Loader 8.3 Oracle Streams 8.4 Automatic Database Diagnostic Monitor 8.5 Automatic Tuning Optimizer 8.6 Automatic Shared Memory Tuning **Unit 9. Oracle Performance Tuning** 9.1 Locate invalid and unusable objects 9.2 Gather SQL optimizer statistics with dbms_stats 9.3 Basic Oracle performance metrics 9.4 Use OEM and dbms_alert to set warning and critical alert thresholds 9.5 The SQL Tuning Advisor 9.6 The SQL Access Advisor 9.6 Interpreting server generated alerts 9.7 Oracle advisory utilities vdb_cache_advice, vshared pool_advice, v$pga_aggregate_target_advice 9.8 Using OEM performance screens

	9.9 Fixing performance issues **Unit 10. User Management** 10.1 Creating Users 10.2 Altering users 10.3 User Profiles 10.4 User resource groups 10.5 Granting privileges & roles 10.6 Auditing user activity with dbms_audit **Unit 11. Oracle Security** 11.1 Password use in Oracle, Password encryption and password aging, External authentication, Using Single sign-on (SSO) 11.2 Object security 11.3 Virtual Private Databases (VPD) in Oracle 11.4 Oracle "grant execute" security 11.5 Use of Roles in Oracle 11.6 Register for security updates **Unit 12. Backup & Recovery** 12.1 Oracle backup & recovery planning 12.2 Parallel instance recovery 12.3 Basics of checkpoints, redo log files, and archived log files 12.4 Using ARCHIVELOG mode 12.5 Creating consistent Oracle backups 12.6 Online hot backups 12.7 Incremental Oracle backups 12.8 Automating database backups with dbms_scheduler 12.9 Monitor the flash recovery area 12.10 Recovering from loss of a Control file 12.11 Recovering from loss of a Redo log file 12.12 Recovering from loss of a system-critical data file 12.13 Recovering from loss of a non system-critical data file
Reference Books	1. Essentials : Oracle Database 10g by Rick Greenwald, Robert Stackowiak, Jonathan Stern, O'Reilly 2. Oracle High Performance Tuning for 9i and 10g by Gavin Powell, Digital Press 3. Oracle Database 10g, DBA Handbook by Loney, Kevin, Bryla, Bob, Oracle Press 4. Oracle Database 10g - The Complete Reference by Loney, Kevin, Oracle Press 5. Oracle Database 10g: A Beginner's Guide by Micheal Abbey, Ian Abramson Osborne, Oracle Press Series
Teaching Methodology	Class work, Discussion, Self-Study, Seminars and/or Assignment
Evaluation Method	30% Internal assessment is based on class attendance, participation, class test, quiz, assignment, seminar, internal examination etc. 70% assessment is based on semester end University External examination

Course: 305: **Data Warehousing & Data Mining**

<table>
<tr><td>Course Code</td><td>305</td></tr>
<tr><td>Course Title</td><td>Data Warehousing & Data Mining</td></tr>
<tr><td>Credit</td><td>4</td></tr>
<tr><td>Teaching per Week</td><td>4 Hrs.</td></tr>
<tr><td>Minimum weeks/ Semester</td><td>15 (Including Class work, examination, preparation, holidays etc.)</td></tr>
<tr><td>Review / Revision</td><td>June 2020</td></tr>
<tr><td>Purpose of Course</td><td>Understanding Data Warehousing and Data Mining</td></tr>
<tr><td>Course Objective</td><td>To understand data warehousing and data mining - concepts and methods, and apply them in practice</td></tr>
<tr><td>Course Outcome</td><td>CO1. Explores Differences between Online Transaction Processing and Online Analytical processing System & describe Multidimensional schemas suitable for data warehousing
CO2. Explains Data warehousing architectures and tools for organizing able to voluminous data of online processing systematically in Data warehouse/Data Mart and use those data for making strategic decisions
CO3. Explains various data pre-processing methods via data reduction, data cleaning, data integration, data transformation etc...
CO4. Trains students to extract knowledge using data mining techniques & to discover interesting patterns from large amounts of data for predictions and classification
CO5. Trains students to develop a data mining application for data analysis using various algorithms & tools.</td></tr>
<tr><td>Mapping between COs with PSOs</td><td>
<table>
<tr><th></th><th>PSO1</th><th>PSO2</th><th>PSO3</th><th>PSO4</th><th>PSO5</th><th>PSO6</th><th>PSO7</th><th>PSO8</th></tr>
<tr><td>CO1</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO2</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO3</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO4</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO5</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
</table>
</td></tr>
<tr><td>Pre-requisite</td><td>RDBMS, Basics of statistics</td></tr>
<tr><td>Course Content</td><td>Unit 1: Data warehouse: Introduction
1.1 Data Warehouse characteristics
1.2 Data Marts
1.3 OLTP and OLAP systems
1.4 Star, Snowflakes, and Fact Constellations Schemas for Multi-dimensional Databases
1.5 OLAP Operations in the Multidimensional data model
1.6 Type of OLAP servers

Unit 2: Developing Data Warehouse
2.1 Building a Data Warehouse
2.2 Three-Tier Data Warehouse Architecture
2.3 Metadata Repository

Unit 3: Data Pre-processing
3.1 Descriptive Data Summarization: central tendency, disperation of data
3.2 Data Cleaning : missing values, noisy data
3.3 Data Integration & Transformation
3.4 Data Reduction: Attribute selection
3.5 Data Discretization & Concept Hierarchy Generation</td></tr>
</table>

	Unit 4: Data Mining: Introduction 4.1 Knowledge discovery and Data Mining. 4.2 Basic Introduction to Data Mining Functionalities: 4.2.1 Concept/Class Description Characterization & Discrimination 4.2.2 Mining Frequent Patterns, Associations, and Correlations 4.2.3 Classification & Prediction 4.3.4 Cluster Analysis 4.2.5 Outlier Analysis 4.2.6 Evolution analysis **Unit 5: Mining Frequent Patterns, Associations, and Correlations** 5.1 Basic concepts: Frequent Itemsets & Closed Itemsets, Association Rules 5.2 The Apriory algorithm: Finding Frequent Itemsets Using Candidate Generation 5.3 FP-growth: Finding Frequent Itemsets without Candidate Generation 5.4 Generating Association Rules from Frequent Itemsets 5.5 Introduction to multilevel and multidimensional Association rules **Unit 6: Classification & Prediction** 6.1 Introduction to Classification & Prediction? 6.2 Prediction: Linear Regression, Nonlinear Regression 6.3 Decision Tree Algorithm 6.3.1 Decision Tree Induction 6.3.2 Attribute Selection Measures- Information Gain and Gain Ratio 6.3.3 Tree Pruning 6.4 Bayesian Classification 6.4.1 Bayes' Theorem 6.4.2 Naïve Bayesian Classification 6.5 Accuracy and Error Measures for classification **Unit 7: Cluster Analysis** 7.1 Classification vs. clustering 7.2 What is Partitioning & Hierarchical Clustering Methods 7.3 Classical Partitioning Methods: k-Means **Unit 8: Application and Trends in Data Mining**
Reference Books	1. Data Mining: Concepts & Techniques by Han & Kamber , Morgan Kaufmann Publishers 2. Introduction to Data Mining with Case Studies by G. K. Gupta, PHI 3. Data Mining Introductory and Advanced Topics by Dunha, Pearson 4. Data Warehouse Toolkit by R. Kinball, John Wiley & Sons 5. Data Warehouses and OLAP: Concepts, Architectures, and Solutions by Robert Wrembel, Christian Koncilia I, GI 6. Data Mining Techniques: For Marketing, Sales, and Customer Relationship Management by Gordon S. Linoff, Michael J. A. Berry, Wiley 7. Data Preparation for Data Mining by Dorian Pyle, Morgan Kaufmann Publishers 8. Data Warehousing Fundamentals: A Comprehensive Guide for IT Professionals by Paulraj Ponniah, Wiley 9. Data Warehousing: Concepts, Techniques, Products and Applications by C.S.R. Prabhu, PHI Learning 10. Advanced Data Mining Techniques by David Louis Olson, Dursun Delen, Springer

Teaching Methodology	Class work, Discussion, Self-study, Seminars and/or Assignment
Evaluation Method	30% Internal assessment is based on class attendance, participation, class test, quiz, assignment, seminar, internal examination etc. 70% assessment is based on semester end University External examination

Course: 305: **Big Data**

Course Code	305
Course Title	**Big Data**
Credit	4
Teaching per Week	4 Hrs.
Minimum weeks/ Semester	15 (Including Class work, examination, preparation, holidays etc.)
Review / Revision	June 2020
Purpose of Course	Understanding distributed computing, Big Data and Hadoop
Course Objective	To learn Big Bata and Hadoop
Course Outcome	CO1 : Student should be able to understand concept and the building blocks of Big Data CO2 : Student should be able to articulate the programming aspects of cloud computing (map Reduce etc.) CO3 : Access and Process Data on Distributed File System, and to understand big data with the help of different big data applications CO4 : Student must be able to represent the analytical aspects of Big Data CO5 : Students will have understanding of distributed computing and will have hands-on experience on Hadoop CO6 : Student shall know the recent trends related to Hadoop File System, MapReduce etc.
Mapping between COs with PSOs	
Pre-requisite	RDBMS
Course Content	**Unit 1: Introduction** 1.1 Evolution of Big Data 1.2 Structuring Big Data 1.3 Elements of Big Data(V's) 1.4 Big Data Analytics 1.5 Commercial use of Big Data **Unit 2: Big Data Technology** 2.1 Distributed and Parallel Computing 2.2 Introducing Hadoop 2.3 HDFS and MapReduce 2.4 Cloud Computing and Big Data 2.5 In-Memory Computing **Unit 3: Hadoop** 3.1 HDFS Architecture 3.2 Blocks 3.3 Name Nodes and Data Nodes 3.4 Using HDFS Files 3.5 Hadoop Specific File System Types 3.6 HDFS Commands 3.7 org.apache.hadoop.io package 3.8 MapReduce Architecture 3.9 Hadoop YARN

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8
CO1	■							
CO2		■	■					
CO3	■		■					
CO4			■	■	■	■		
CO5	■	■	■					
CO6		■		■		■	■	■

	3.10 HBase 3.11 Combining HBase 3.12 Hive 3.13 Pig and Pig Latin 3.14 Sqoop **Unit 4: Technology Foundations** 4.1 Big Data Stack 4.2 Virtualization and Big Data **Unit 5: Storing Data in Databases and Processing of Data** 5.1 RDBMS and Big Data 5.1.1 CAP Theorem 5.3 NoSQL Databases 5.4 Polygot Persistence 5.5 Integrating Big Data with traditional Data Warehouses 5.6 Big Data Analytics 5.7 Processing Data with MapReduce 5.8 Customizing MapReduce Execution and implementing MapReduce Program 5.9 Testing and Debugging MapReduce Applications 5.10 Analytical Approaches and Tools to Analyze Data
Reference Books	1. D T Editorial services, Big Data Black book, Dreamtech Press, ISBN 978-93-5119-931 2. Alex Holmes, Hadoop in Practice, Manning Publication company, 2014, ISBN 1617292222, 9781617292224 3. Kuan-Ching, Li Hai Jiang, Laurence T. Yang Alfredo Cuzzocrea, Big Data : Algorithms, Analytics and Applications. CRC Press 4. Hu, Wen Chen, Big Data Management, Technologies and Applications, IGI Global 5. Tom White, Hadoop The Definitive Guide, o'Reilly
Teaching Methodology	Class work, Discussion, Self Study, Seminars and/or Assignment
Evaluation Method	30% Internal assessment is based on class attendance, participation, class test, quiz, assignment, seminar, internal examination etc. 70% assessment is based on semester end University External examination

Course: 306: **Programming Skills XI**

Course Code	306
Course Title	**Programming Skills XI**
Credit	2
Teaching per Week	3 Hrs.
Minimum weeks/ Semester	15 (Including Lab. work, examination, preparation, holidays etc.)
Review / Revision	June 2020
Purpose of Course	This course helps students to implement the Unix Internals with shell programming/IOT practically.
Course Objective	Learning to implement fundamentals and advanced topics of Unix Internals with Shell Scripting/IOT practically
Pre-requisite	Practical programming in desktop environment / Embedded Technology
Course Outcome	After studying the course, students will be able to practically work on advanced technology platforms of Unix Internals with Shell Scripting /IOT.
Course Content	Practical based on paper no 501. Separate journal to be prepared for this subject based on 501.
Reference Books	-----
Teaching Methodology	Lab work, Practical Programming Exercises (to be documented in a separate journal), Self-study, and/or Assignment
Evaluation Method	30% Internal assessment is based on Lab attendance, practical test, practical internal examination etc. 70% assessment is based on semester end University External practical examination

Course: .307: **Programming Skills XII**

Course Code	307
Course Title	**Programming Skills XII**
Credit	3
Teaching per Week	3 Hrs.
Minimum weeks/ Semester	15 (Including Lab. work, examination, preparation, holidays etc.)
Review / Revision	June 2020
Purpose of Course	This course helps students to implement the concepts of SAP/NoSQL practically
Course Objective	Learning to implement the ERP using SAP/NoSQL databases practically
Pre-requisite	DBMS
Course Outcome	After studying the course, students will be able to practically work on SAP/NoSQL Databases
Course Content	Practical based on paper no 503. Separate journal to be prepared for this subject 503.
Reference Books	-----
Teaching Methodology	Lab work, Practical Programming Exercises (to be documented in a separate journal), Self-study, and/or Assignment
Evaluation Method	30% Internal assessment is based on Lab attendance, practical test, practical internal examination etc. 70% assessment is based on semester end University External practical examination

Course: 308: **Programming Skills XIII**

Course Code	308
Course Title	**Programming Skills XIII**
Credit	2
Teaching per Week	3 Hrs.
Minimum weeks/ Semester	15 (Including Lab. work, examination, preparation, holidays etc.)
Review / Revision	June 2020
Purpose of Course	This course helps students to handle advanced database administration activities
Course Objective	Advanced database administration
Pre-requisite	Practically learning advanced database administration
Course Outcome	After studying the course, students will be able to handle database administration, tuning and maintenance in various fields
Course Content	Practical based on paper no 504. Separate journal to be prepared for this subject 504.
Reference Books	-----
Teaching Methodology	Lab work, Practical Programming Exercises (to be documented in a separate journal), Self-study, and/or Assignment
Evaluation Method	30% Internal assessment is based on Lab attendance, practical test, practical internal examination etc. 70% assessment is based on semester end University External practical examination

Course: 309: **Programming Skills XIV**

Course Code	309
Course Title	**Programming Skills XIV**
Credit	3
Teaching per Week	3 Hrs.
Minimum weeks/ Semester	15 (Including Lab. work, examination, preparation, holidays etc.)
Review / Revision	June 2020
Purpose of Course	This course helps students to implement Data warehousing & data mining / Big Data practically.
Course Objective	Learn to use Data warehousing and data mining techniques in various practical environments / use Big data practically.
Pre-requisite	Databases, SQL, Advanced SQL
Course Outcome	After studying the course, students will be able to understand Data warehousing and data mining/Big Data practically
Course Content	Practical based on paper no 505. Separate journal to be prepared for this subject 505.
Reference Books	-----
Teaching Methodology	Lab work, Practical Programming Exercises (to be documented in a separate journal), Self-study, and/or Assignment

Evaluation Method	30% Internal assessment is based on Lab attendance, practical test, practical internal examination etc. 70% assessment is based on semester end University External practical examination

MCA 3rd Sem

(Network Group)

Course: 301: **Internet of Things (IoT)**

(Elective)

<table>
<tr><td>Course Code</td><td>301</td></tr>
<tr><td>Course Title</td><td>Internet of Things (IoT)</td></tr>
<tr><td>Credit</td><td>4</td></tr>
<tr><td>Teaching per Week</td><td>4 Hrs.</td></tr>
<tr><td>Minimum weeks per Semester</td><td>15 (Including Class work, examination, preparation, holidays etc.)</td></tr>
<tr><td>Review / Revision</td><td>June 2021</td></tr>
<tr><td>Purpose of Course</td><td>This course is an introduction for students to IoT. The course also gives students an idea about various components of IoT and explain its working. The course also explains the role of embedded systems in IoT ecosystem.</td></tr>
<tr><td>Course Objective</td><td>The objective of the course is -
9. To make student understand IoT
10. To understand the working of Micro-Controller & Micro-Computer
11. To explain various types of sensors
12. To introduce students with Programming in IoT</td></tr>
<tr><td>Course Outcome</td><td>CO1: Understand the IoT ecosystem and architecture. Understand IoT standards and protocols. Understand the privacy, security and governance issues in IoT applications. Exposing students with IoT botnet and the risks involved with IoT based applications.
CO2: Understand the overview and working of the various sensors used in IoT applications. Introduce and Explain various network communication protocols, standards and IoT data Protocols. Understanding the Wireless Sensor Network and how IoT devices communicate with each other.
CO3: Understand Micro-Controller and its architecture. Understand the usage of Micro-Controller in IoT applications. Understand how Arduino and NodeMCU interact with sensors and communicate over the network.
CO4: Understand Micro-Computer and its architecture. Understand the difference between a Micro-Controller and Micro-Computer. Understand the usage of Micro-Computer in IoT applications. Understand how Raspberry Pi interact with sensors and communicate over the network.
CO5:Expose the students with Server-side development in IoT applications. Understand how to develop and deploy applications in Arduino and NodeMCU. Understand how Arduino and NodeMCU communicate among themselves, sensors and the server. Understand how to build a full IoT app by integrating them with mobile applications.</td></tr>
<tr><td>Mapping between COs with PSOs</td><td>
<table>
<tr><td></td><td>PSO1</td><td>PSO2</td><td>PSO3</td><td>PSO4</td><td>PSO5</td><td>PSO6</td><td>PSO7</td><td>PSO8</td></tr>
<tr><td>CO1</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO2</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO3</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO4</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO5</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
</table>
</td></tr>
<tr><td>Pre-requisite</td><td>C , C++</td></tr>
<tr><td>Course Content</td><td>Unit 1: Introduction to IoT
1.1 What is IoT
1.2 IoT Applications
1.3 IoT Privacy and Security
1.3.1 Identification in Distributed Environment
1.3.2 Device Authentication
1.4 IoT Botnet</td></tr>
</table>

	Unit 2: Networking and Communication 2.1 Basics of Wireless Networking 2.1.1 CSMA/CA 2.2 IoT Network Protocols 2.2.1 BLE, Zigbee, LoRaWAN, RFID 2.3 IoT Data Protocols 2.3.1 CoAP, MQTT, XMPP, DDS **Unit 3: Sensors** 3.1 Introduction to Sensors 3.2 Types of Sensors & their working 3.3 Wireless Sensor Network 3.3.1 Introduction to WSN 3.3.2 Applications 3.3.3 Characteristics 3.3.4 Challenges 3.3.5 Components 3.4 Wireless Adhoc Network Vs Wireless Sensor Network **Unit 4: Micro-Controller: Arduino, NodeMCU** 4.1 Introduction to Microcontrollers 4.2 Arduino IDE 4.3 Arduino Architecture 4.4 Arduino Pin Diagram 4.5 Introduction to NodeMCU 4.6 NodeMCU Specifications and Applications 4.7 NodeMCU ESP8266 Pinout **Unit 5: IoT App Interaction & Introduction to Raspberry Pi** 5.1 Uploading sensor data to server 5.2 Reading sensor data from server 5.3 Controlling IoT device and components from Mobile or Web 5.4 Introduction to Microcomputers 5.5 Raspberry Pi Architecture 5.6 Raspberry Pi Pinout
Reference Books	1) Getting Started with Internet of Things – By Cuno Pfister, O'Reilly 2) Learning Internet of Things – By Peter Waher , Packt Publication 3) Internet of Things : A Hands-on Approach – By Arshdip Bahga and Vijay Madisetti 4) IoT Governance, Privacy and Security Issues, IERC 5) IoT Fundamentals: Networking Technologies, Protocols and Use Cases for the Internet of Things, Cisco Press 6) Fundamentals of IoT Communication Technologies, Springer 7) Microcontrollers – Architecture, Programming, Interfacing and system design – By Raj Kamal , Pearson 8) Exploring C for Microcontrollers : A hands on approach, Springer 9) Arduino for Dummies, Wiley 10) Make: Getting Started With Arduino - The Open Source Electronics Prototyping Platform, Shroff/Maker Media 11) ESP8266: Get Started With ESP8266 Programming NodeMCU Using Arduino IDE, Createspace Independent Pub 12) Internet of Things Projects with ESP32, Packt Publishing Limited

	13) Microprocessor Architecture, Programming and Applications with the 8085 - By Ramesh Gaonkar , Penram International Publishing 14) Raspberry Pi for Dummies , Wiley 15) Raspberry Pi User Guide – By Eben Upton and Garath Halfacree, Wiley
Teaching Methodology	Class work, Discussion, Self-Study, Seminars and/or Assignment
Evaluation Method	30% Internal assessment is based on class attendance, participation, class test, quiz, assignment, seminar, internal examination etc. 70% assessment is based on semester end University External examination

Course: 301: **Machine Learning**

(Elective)

<table>
<tr><td>Course Code</td><td>301</td></tr>
<tr><td>Course Title</td><td>Machine Learning (ML)</td></tr>
<tr><td>Credit</td><td>4</td></tr>
<tr><td>Teaching per Week</td><td>4 Hrs.</td></tr>
<tr><td>Minimum weeks per Semester</td><td>15 (Including Class work, examination, preparation, holidays etc.)</td></tr>
<tr><td>Review / Revision</td><td>June 2021</td></tr>
<tr><td>Purpose of Course</td><td>This course is an introduction for students to ML. The course also gives students an idea about various methods and algorithms of Machine Learning and application development of ML.</td></tr>
<tr><td>Course Objective</td><td>The objective of the course is –
1. To make student understand ML
2. To understand the various Machine Learning method
3. To explain various algorithms used in Machine learning
4. To introduce students with Programming in ML</td></tr>
<tr><td>Course Outcome</td><td>CO1 : Explain to the students the fundamental know how like the types of machine learning algorithms, applications and various required libraries, model selection etc. required to implement machine learning algorithms.
CO2 : Train students with can utilize various data wrangling techniques, data cleaning, data transformation, data reduction, data discretization, feature selection, and data visualization
CO3 : Train students who can implement supervised learning algorithms utilizing regression and classification algorithm on the real world dataset.
CO4 : Train student to have understanding of Artificial Neural Network and its working. Also, to make them capable of implementing ANN for solving real world problems using it.
CO5 : Explain to the students to use clustering and association rules as unsupervised learning method to solve complex problems.
CO6 : Train students to use machine learning techniques to solve real life complex problems.</td></tr>
<tr><td>Mapping between COs with PSOs</td><td>
<table>
<tr><th></th><th>PSO1</th><th>PSO2</th><th>PSO3</th><th>PSO4</th><th>PSO5</th><th>PSO6</th><th>PSO7</th><th>PSO8</th></tr>
<tr><td>CO1</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO2</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO3</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO4</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO5</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO6</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
</table>
</td></tr>
<tr><td>Pre-requisite</td><td>Basics of Linear Algebra, Statistics and Mathematics, Python Programming</td></tr>
<tr><td>Course Content</td><td>Unit 1 : Introduction
1.1. Definition of Machine Learning
1.2 Types of Machine Learning : Supervised , Unsupervised and Semi-supervised
1.3 Applications and tools of Machine Learning (Scikit learn library)</td></tr>
</table>

	1.4 Data Pre-processing, Selecting a model and training a model 1.5 Evaluating a performance of model and improving performance **Unit 2 : Data Wrangling** 2.1 Definition and goal of Data Wrangling 2.2 Importance of Data Wrangling 2.3 Data Pre-processing and Data Cleaning 2.3.1 Data Cleaning 2.3.2 Data Transformation 2.3.3 Data Reduction 2.3.4 Data Discretization 2.3.5 Feature Selection 2.4 Data Visualization **Unit 3 : Supervised Learning** 3.1 Supervised Learning : Classification and Regression 3.2 Regression 3.2.1 Simple and Multiple Regression 3.2.2 Linear Regression 3.2.3 Gradient Decent 3.2.4 Logistic Regression 3.3 Classification Algorithms : 3.3.1 K-nearest Neighbour 3.3.2 Support Vector Machines 3.3.3 Decision Trees 3.3.4 Naïve Bayes Classifier 3.4 Introduction to Support Vector Machine **Unit 4 : Neural Network** 4.1 Introduction to Neural Network 4.2 Architecture of Neural Network 4.3 Feedforward network and Backpropagation with example 4.4 Applications of Neural Network **Unit 5 : Unsupervised Learning** 5.1 Introduction to Unsupervised learning 5.2 Clustering 5.2.1 Selection of Clusters 5.2.2 Algorithms : 5.2.2.1 K – means clustering 5.2.2.2 Hierarchical Clustering 5.3 Association Rule Learning 5.3.1 Algorithms : 5.3.1.1 FP- Growth 5.3.1.2 Apriori Algorithm
Reference Books	1. “Machine Learning” by Tom M. Mitchell, McGraw Hill 2. “Understanding Machine Learning” by Shai Shalev-Shwartz, Shai Ben-David 3. “Machine Learning” by Anuradha Srinivasaraghavan, Vincy Joseph

	4. “Machine Learning using Python” by U Dinesh Kumar Manaranjan Pradhan 5. “Real-World Machine Learning” by Henrik Brink, Joseph Richards, Mark Fetherolf 6. “Python Machine Learning” by Sebastian Raschka and Vahid Mirjalili 7. “Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems” by Aurelien Geron 8. “Machine Learning in Action” by Peter Harrington 9. “Introduction to Machine Learning with Python : A Guide for Data Scientists” by Andreas C. Muller, Sarah Guido
Teaching Methodology	Class work, Discussion, Self-Study, Seminars and/or Assignment
Evaluation Method	30% Internal assessment is based on class attendance, participation, class test, quiz, assignment, seminar, internal examination etc. 70% assessment is based on semester end University External examination

Course: 302: **Design Patterns**

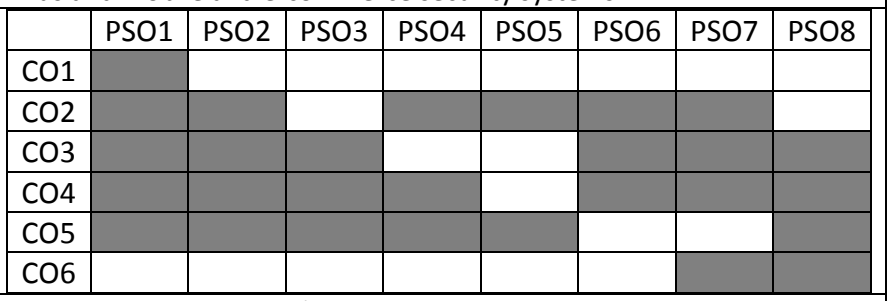
<table>
<tr><td>Course Code</td><td>302</td></tr>
<tr><td>Course Title</td><td>Design Patterns</td></tr>
<tr><td>Credit</td><td>4</td></tr>
<tr><td>Teaching per Week</td><td>4 Hrs.</td></tr>
<tr><td>Minimum weeks per Semester</td><td>15 (Including Class work, examination, preparation, holidays etc.)</td></tr>
<tr><td>Review / Revision</td><td>June 2021</td></tr>
<tr><td>Purpose of Course</td><td>The purpose of the course is to make student understand how Patterns can be implemented in various object oriented programming languages to solve real world problems.</td></tr>
<tr><td>Course Objective</td><td>The objective of the course is -
5. To study various Design Patterns
6. How these Patterns can be used to design better systems through Object Oriented Programming Languages</td></tr>
<tr><td>Course Outcome</td><td>CO1: Explain students about the various design patterns; their categories, and purpose.
CO2: Explain the creational design patterns.
CO3: Explain the structural design patterns.
CO4: Explain the behavioural design patterns.
CO5: Explain some more design patterns used in IT industry currently.
CO6: Make students understand the applicability of design patterns practiced by IT companies and how effectively combine these patterns for effective software development.</td></tr>
<tr><td>Mapping between COs with PSOs</td><td>
<table>
<tr><th></th><th>PSO1</th><th>PSO2</th><th>PSO3</th><th>PSO4</th><th>PSO5</th><th>PSO6</th><th>PSO7</th><th>PSO8</th></tr>
<tr><td>CO1</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO2</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO3</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO4</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO5</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO6</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
</table>
</td></tr>
<tr><td>Pre-requisite</td><td>Object Oriented Programming, Software Engineering</td></tr>
<tr><td>Course Content</td><td>Unit -1 Creational Patterns
1.1 Singleton Pattern
1.2 Prototype Pattern
1.3 Builder Pattern
1.4 Factory Method Pattern
1.5 Abstract Factory Pattern

Unit-2 Structural Patterns
2.1 Proxy Pattern
2.2 Decorator Pattern
2.3 Adapter Pattern
2.4 Façade Pattern
2.5 Flyweight Pattern
2.6 Composite Pattern
2.7 Bridge Pattern

Unit-3 Behavioural Pattern
3.1 Visitor Pattern
3.2 Observer Pattern</td></tr>
</table>

	3.3 Strategy Pattern 3.4 Template Method Pattern 3.5 Command Pattern 3.6 Iterator Pattern 3.7 Memento Pattern 3.8 State Pattern 3.9 Mediator Pattern 3.10 Interpreter Pattern **Unit-4 Additional Design Patterns** 4.1 Simple Factory Pattern 4.2 Null Object Pattern 4.3 MVC Pattern **Unit-5 Pattern Applicability** 5.1 Security Patterns Repository 5.2 Patterns for Agile Development 5.3 Restful Service Patterns 5.4 Solution with semaphore 5.5 Patterns and Pattern combination in practice 5.6 Big Ball of Mud **Self-Study :** Pattern Languages
Reference Books	13. Design Patterns: Elements of Reusable Object-Oriented Software, Erich Gamma, Richard Helm, Ralph, John, Addision Wesley 14. Head First Design Patterns, Eric Freeman, O'Reilly 15. Design Patterns in C#, Vaskaran Sarcar, Apress 16. Design Patterns in Modern C++, Reusable Approaches for Object-Oriented Software Design, Dmitri Nesteruk, Apress 17. Modern C++ design: generic Programming and design patterns applied, Alexendrescu, Andrei, Addison-Wesley 18. Java Design Patterns: A Hands-on Experience with Real-World Examples, Vaskaran Sarcar, Apress
Teaching Methodology	Class work, Discussion, Self-Study, Seminars and/or Assignment
Evaluation Method	30 % internal assessment and 70% external assessment

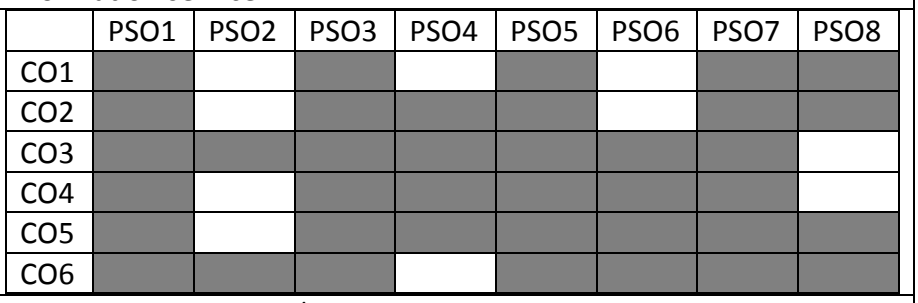
Course: 303: **Network Essentials and its Security**

Course Code	303
Course Title	**Network Essentials and its Security**
Credit	4
Teaching per Week	4 Hrs.
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Review / Revision	June 2020
Purpose of Course	To make students learn Network essentials and various security measures for the challenges to which the IT industry is exposed
Course Objective	To understand Network Management and its security
Course Outcomes	CO1: Explain students familiar about network essentials. Make them familiar with various network devices like repeaters, bridge router, and gateway. CO2: Explain how to administer computer network. Make them understand various wide area network techniques, and explain network security. CO3: To understand cryptography, PKI, and digital signatures. CO4: Familiarize with various security services, and how certification and key management is handled in PKI. CO5: To make students understand various network security applications; covering internet protocol security, web security, email security, network management and its security. CO6: Make students understand access control, authentication schemes, firewalls, and virtual private network. Also to learn intrusion detection, virus and mobile and e-commerce security systems.
Mapping between COs with PSOs	
Pre-requisite	Computer Network, TCP/IP
Course Content	**Unit 1: Network Essentials** 1.1 Repeaters & Bridges 1.1.1 LAN Expansion 1.1.2 Repeaters 1.1.3 Bridges 1.1.4 How Bridges Work 1.1.5 Creating the routing table 1.1.6 Segmenting Network Traffic 1.1.7 Remote Bridges 1.1.8 Differentiating between bridges and repeaters 1.2 Routers & Gateways 1.2.1 Routers 1.2.2 How routers work 1.2.3 Routing benefits 1.2.4 Routing protocols 1.2.5 Routing V/S Bridging 1.2.6 B Routers 1.2.7 Gateways

	1.2.8 How Gateways work 1.3 Network Administration 1.3.1 Bottlenecks 1.3.2 Simple Network Management Protocol 1.3.3 Data Protection 1.3.4 Backup Methods 1.3.5 Testing and Storage 1.3.6 Implementing a Backup System 1.3.7 Uninterruptible Power Suppliers 1.3.8 How Gateways work 1.3.9 Implementing Fault Tolerant Systems 1.3.10 RAID 1.3.11 Sector Sparing 1.4 Advance WAN Transmission 1.4.1 Overview 1.4.2 Multiplexing, Packet and Circuit Switching Networks 1.4.3 X.25 1.4.4 Asynchronous Transfer Mode (ATM) 1.4.5 ISDN 1.4.6 SONET 1.4.7 SMDS **Unit 2: Introduction to Network Security** **Unit 3: Cryptography Techniques** 3.1 Classical Cryptography 3.2 Conventional Cryptography 3.2.1 DES 3.3 Public – key Cryptography 3.3.1 RSA 3.4 Digital Signatures 3.4.1 DSA **Unit 4: Security Services** 12.1 Message Integrity 12.2 Confidentiality and Authentication 12.3 Certification and Key Management 4.3.1 PKI **Unit 5: Network Security Applications** 5.1 IP Security 5.1.1 IPsec 5.2 Web Security 5.2.1 SSL, TLS, SET 5.3 Electronic Mail Security 5.3.1 PGP, S/MIME 5.4 SNMP Security **Unit 6: Access Control in Computer Networks** 6.1 Authentication Protocols and Services 6.1.1 Kerberos and X.509 6.2 Firewalls 6.3 Virtual Private Networks (VPNs) **Unit 7: System Security** 7.1 Intrusion detection

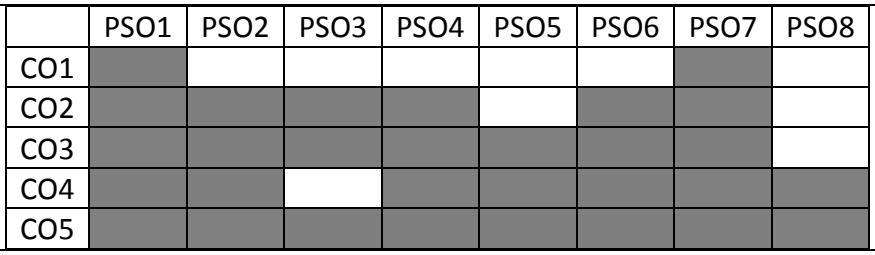
	7.2 Viruses **Unit 8: Mobile System & E-Commerce Securities** 8.1 3G Security 8.2 E-Payment Systems 8.3 Fair Data Exchange
Reference Books	1) Cryptography and Network Security, 2/e, ISBN: 0-13-869017-0 - W. Stallings - Pearson Education, 1999 2) Network Security Essentials: Applications and Standards, 1/e, ISBN: 0-13-016093-8 - W. Stallings - Pearson Education, 2000 3) SSL and TLS: designing and building secure systems, ISBN: 0-201-61598-3 - E. Rescorla - Addison-Wesley, 2001 4) Implementing Secure Intranets and Extranets, ISBN: 0-89006-447-4 - K M Phaltankar - Artech House Publishers, 2000 5) Secure Electronic Commerce: Building the Infrastructure for Digital Signature and Encryption, ISBN: 0-13-027276-0 - W. Ford, and M. Baum - Prentice Hall, 2001 6) Security in Computing, ISBN: 0-13-185794-0, 2/e - C. P. Pfleeger - Prentice Hall, 1997 7) Building Internet Firewalls, 2/e, ISBN: 1-56592-871-7 - E. D. Zwicky, et al - O'Reilly, 2000 8) CDMA Cellular Mobile Communications & Network Security, ISBN: 0-13-598418-1 - M. Y. Rhee, - Prentice Hall, 1998 9) Journal of Computer Security 10) ACM Transactions on Information and System Security 11) ACM Conference on Computer and Communications Security 12) IEEE Symposium on Security and Privacy 13) Internet documents - RFCs (Request for Comments) 14) Guide to Networking Essentials, Fourth Edition - Greg Tomsho, et al 15) Computer Networking Essentials - Debra Littlejohn Shinder 16) Networking Essentials: Hands-On, Self-Paced Training for Supporting Local and Wide Area Networks - Microsoft Corporation (Corporate Author) 17) Computer Network - A. S. Tanenbaum
Teaching Methodology	Class work, Discussion, Self-Study, Seminars and/or Assignment
Evaluation Method	30% Internal assessment is based on class attendance, participation, class test, quiz, assignment, seminar, internal examination etc. 70% assessment is based on semester end University External examination

Course: 304: **Network Administration**

Course Code	304
Course Title	**Network Administration**
Credit	4
Teaching per Week	4 Hrs.
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Review / Revision	June 2020
Purpose of Course	Understanding advanced network administration
Course Objective	To learn advanced network administration, its configuration and maintenance
Course Outcomes	CO1: Understand networking fundamentals and networking using tcp/ip protocol. CO2: Learn network management using dhcp, dns, and nfs. CO3: Learn network management using Linux os; and also remote administration. CO4: Understand administration services, task automation and cron daemon on Linux. CO5: Understand samba service and its configuration CO6: Learn to configure and use system wide logging and network information service
Mapping between COs with PSOs	
Pre-requisite	Computer Network, TCP/IP
Course Content	**Unit 1: Networking and TCP/IP on Linux** a. Fundamentals of Linux Networking b. Fundamentals of TCP/IP on the Linux Operating System c. Advanced Linux TCP/IP Concepts d. Introduction to Dial-up Technologies **Unit 2: Dynamic Host Configuration Protocol** 2.1 Introduction to BOOTP and DHCP 2.2 Installing and Examining a Linux DHCP server 2.3 Examining Additional DHCP Options and Configurations **Unit 3: Domain Name System** 3.1 Introduction to the Domain Name System 3.2 Installing and Configuring DNS **Unit 4: The Network File System** 4.1 Introduction to the Network File System 4.2 Configuring NFS **Unit 5: Linux Remote Administration** 5.1 Introduction to Remote Administration 5.2 The Telnet Protocol 5.3 The open secure Shell protocol

	Unit 6: The Cron Daemon 6.1 Introduction to Automation 6.2 Configuring the Cron Daemon **Unit 7: Samba** 7.1 Introduction to Samba 7.2 Cross-Platform Connectivity 7.3 Installing and Configuring Samba **Unit 8: Linux System-Wide Logging** 8.1 Introduction to System-wide Logging 8.2 Configuring System-Logging **Unit 9: The Network Information Service** 9.1 Introduction to NIS 9.2 Setting Up and Configuring an NIS server
Reference Books	1) TCP/IP Network Administration - Craig Hunt - O'Reilly & Associates 2) Managing NFS and NIS - Hal Stern - O'Reilly & Associates 3) DNS and BIND - Albitz/Liu - O'Reilly & Associates 4) Sendmail - Bryan Costales/Eric , Allman/Neil Rickert - O'Reilly & Associates 5) UNIX System Administration Handbook - Second Edition - Nemeth/Snyder/Seebass - Prentice Hall 6) Red Hat Linux Networking and System Administration – Terry Collings, Kurt Wall 7) Red Hat Linux 9 Bible - Christopher Negus 8) Official Red Hat Linux User's Guide - Red Hat Inc. 9) Official Red Hat Linux Administrator's Guide - Red Hat Inc. 10) Red Hat Linux Security and Optimization - Mohammad J. Kabir
Teaching Methodology	Class work, Discussion, Self-Study, Seminars and/or Assignment
Evaluation Method	30% Internal assessment is based on class attendance, participation, class test, quiz, assignment, seminar, internal examination etc. 70% assessment is based on semester end University External examination

Course: 305: **Wireless Network and Mobile Computing**

Course Code	305
Course Title	**Wireless Network and Mobile Computing**
Credit	4
Teaching per Week	4 Hrs.
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Review / Revision	June 2020
Purpose of Course	Understanding Wireless Network and Mobile Computing
Course Objective	To understand various aspects related to Wireless Network technologies, mobile networks and its computing
Course Outcomes	CO1: Make students learn fundamentals of wireless technologies. CO2: Make students learn W-LAN technologies, and explain its implementation. CO3: To understand hardware, its implementation and protocols for wireless network. CO4: Explain MANET, its applications, protocols associated, and routing algorithms. CO5: Learn mobile computing and its architecture. Also learn GSM, GPRS and WAP.
Mapping between COs with PSOs	
Pre-requisite	Computer Network
Course Content	**Unit 1: Introduction to Wireless Network Technologies** 1.1 Introduction 1.2 Standards 1.3 Emerging Technologies 1.4 OSI Basics 1.5 LAN Basics 1.6 LAN & WAN Protocols 1.7 Internet Protocol **Unit 2: Wireless LAN Technologies, Implementation and Layers** 2.1 Frequency Hopping Spread Spectrum 2.2 Direct Sequence Spread Spectrum (DSSS) 2.3 Interference 2.4 RF Math 2.5 Service Sets 2.6 Mobile IP 2.7 Appropriate use and design of wireless NetworkingMultipath 2.8 Co-location 2.9 Power-over-Ethernet (PoE) 2.10 Modulation and Bit Coding 2.11 Fragmentation 2.12 SIFS / PIFS / DIFS / EIFS **Unit 3: Hardware Configuration, Implementation and Protocols** 3.1 Access Points

	3.2 Bridges 3.3 Workgroup bridges 3.4 Wireless Residential Gateways 3.5 Host Connectivity 3.6 Antennas, Cables, & Connectors 3.7 MAC and Routing Protocols for IEEE 802.11 3.8 Wireless Mesh Networks **Unit 4: MANET** 4.1. Various applications of MANET 4.2. Destination- Sequenced Distance Vector protocol 4.3. Dynamic Source Routing protocol 4.4. Ad Hoc On-Demand Distance-Vector protocol 4.5. Link Reversal Routing a. Gafni-Bertsekas algorithm b. Lightweight mobile routing algorithm. 4.6. Temporally ordered routing algorithm **Unit 5: Introduction to Mobile Computing, GSM, GPRS & WAP** 5.1 Mobility, Nomadic, Mobile and Ubiquitous computing 5.2 Mobile Computing Architecture 5.3 Mobile Computing Technologies (Hardware, Software, Communication) 5.4 Introduction to GSM 5.5 GSM Architecture, Mobility Management, Network Signaling 5.6 GPRS Architecture 5.7 Network Nodes 5.8 Mobile Internet Standards, WAP Gateway and Protocols 5.9 WML
Reference Books	1) Wireless Local Area Network Fundamentals - Pejman Roshan, Jonathan Leary 2) Wireless Networks First Step (First-step series) - Jim Geier 3) 802.11 wireless network site surveying and installation - Bruce Alexander 4) Introduction to Wireless and Mobile Systems by Cengage Learning (Thompson) 5) J. Schiller, Mobile Communications, Addison –Wesley, 2003 6) Wi-Fi Security - Stewart Miller 7) Wireless and Mobile Network - Architecture Yi-Bing Lin & Imrich Chlamtac - John Wiley & Sons, 2001 8) Mobile and Wireless Design Essentials by Martyn Mallick, John Wiley & Sons 9) Guide to Designing and Implementing wireless LANs - Mark Ciampa - Thomson learning , Vikas Publishing House, 2001 10) Wireless Web Development - Ray Rischapter - Springer publishing, 2000
Teaching Methodology	Class work, Discussion, Self-Study, Seminars and/or Assignment
Evaluation Method	30% Internal assessment is based on class attendance, participation, class test, quiz, assignment, seminar, internal examination etc. 70% assessment is based on semester end University External examination

Course: 306: **Programming Skills XI**

Course Code	306
Course Title	**Programming Skills XI**
Credit	2
Teaching per Week	3 Hrs.
Minimum weeks/ Semester	15 (Including Lab. work, examination, preparation, holidays etc.)
Review / Revision	June 2020
Purpose of Course	This course helps students to implement the Unix Internals with shell programming/IOT practically.
Course Objective	Learning to implement fundamentals and advanced topics of Unix Internals with Shell Scripting/IOT practically
Pre-requisite	Practical programming in desktop environment / Embedded Technology
Course Outcome	After studying the course, students will be able to practically work on advanced technology platforms of Unix Internals with Shell Scripting /IOT.
Course Content	Practical based on paper no 501. Separate journal to be prepared for this subject based on 501.
Reference Books	-----
Teaching Methodology	Lab work, Practical Programming Exercises (to be documented in a separate journal), Self-study, and/or Assignment
Evaluation Method	30% Internal assessment is based on Lab attendance, practical test, practical internal examination etc. 70% assessment is based on semester end University External practical examination

Course: 307: **Programming Skills XII**

Course Code	307
Course Title	**Programming Skills XII**
Credit	3
Teaching per Week	3 Hrs.
Minimum weeks/ Semester	15 (Including Lab. work, examination, preparation, holidays etc.)
Review / Revision	June 2020
Purpose of Course	This course helps students to understand the Networking and its security practically
Course Objective	Learning to implement Network and its security practically.
Pre-requisite	Network fundamentals
Course Outcome	After studying the course, students will be able to practically implement network and its security
Course Content	Practical based on paper no 503. Separate journal to be prepared for this subject 503.
Reference Books	-----
Teaching Methodology	Lab work, Practical Exercises (to be documented in a separate journal), Self-study, and/or Assignment
Evaluation Method	30% Internal assessment is based on Lab attendance, practical test, practical internal examination etc. 70% assessment is based on semester end University External practical examination

Course: 308: **Programming Skills XIII**

Course Code	308
Course Title	**Programming Skills XIII**
Credit	2
Teaching per Week	3 Hrs.
Minimum weeks/ Semester	15 (Including Lab. work, examination, preparation, holidays etc.)
Review / Revision	June 2020
Purpose of Course	This course helps students to learn network administration
Course Objective	Students will learn administration of network practically
Pre-requisite	Networking fundamentals
Course Outcome	After studying the course, students will be able to practically perform administrative tasks of networks practically
Course Content	Practical based on paper no 504. Separate journal to be prepared for this subject 504.
Reference Books	-----
Teaching Methodology	Lab work, Practical Exercises (to be documented in a separate journal), Self-study, and/or Assignment
Evaluation Method	30% Internal assessment is based on Lab attendance, practical test, practical internal examination etc. 70% assessment is based on semester end University External practical examination

Course: 309: **Programming Skills XIV**

Course Code	309
Course Title	**Programming Skills XIV**
Credit	3
Teaching per Week	3 Hrs.
Minimum weeks/ Semester	15 (Including Lab. work, examination, preparation, holidays etc.)
Review / Revision	June 2020
Purpose of Course	This course helps students to implement the basic and advanced concepts of wireless network and mobile computing
Course Objective	Learning wireless protocols and its implementation practically
Pre-requisite	Computer Network, C/C++ programming
Course Outcome	After studying the course, students will be able to practically develop/enhance wireless protocols and find better solutions application to the various industries dependent upon them
Course Content	Practical based on paper no 505. Separate journal to be prepared for this subject 505.
Reference Books	-----
Teaching Methodology	Lab work, Practical Exercises (to be documented in a separate journal), Self-study, and/or Assignment
Evaluation Method	30% Internal assessment is based on Lab attendance, practical test, practical internal examination etc. 70% assessment is based on semester end University External practical examination

MCA 3rd Sem. (General Group)

Course: 301: **Internet of Things (IoT)**

(Elective)

<table>
<tr><td>Course Code</td><td>301</td></tr>
<tr><td>Course Title</td><td>Internet of Things (IoT)</td></tr>
<tr><td>Credit</td><td>4</td></tr>
<tr><td>Teaching per Week</td><td>4 Hrs.</td></tr>
<tr><td>Minimum weeks per Semester</td><td>15 (Including Class work, examination, preparation, holidays etc.)</td></tr>
<tr><td>Review / Revision</td><td>June 2021</td></tr>
<tr><td>Purpose of Course</td><td>This course is an introduction for students to IoT. The course also gives students an idea about various components of IoT and explain its working. The course also explains the role of embedded systems in IoT ecosystem.</td></tr>
<tr><td>Course Objective</td><td>The objective of the course is -
13. To make student understand IoT
14. To understand the working of Micro-Controller & Micro-Computer
15. To explain various types of sensors
16. To introduce students with Programming in IoT</td></tr>
<tr><td>Course Outcome</td><td>CO1: Understand the IoT ecosystem and architecture. Understand IoT standards and protocols. Understand the privacy, security and governance issues in IoT applications. Exposing students with IoT botnet and the risks involved with IoT based applications.
CO2: Understand the overview and working of the various sensors used in IoT applications. Introduce and Explain various network communication protocols, standards and IoT data Protocols. Understanding the Wireless Sensor Network and how IoT devices communicate with each other.
CO3: Understand Micro-Controller and its architecture. Understand the usage of Micro-Controller in IoT applications. Understand how Arduino and NodeMCU interact with sensors and communicate over the network.
CO4: Understand Micro-Computer and its architecture. Understand the difference between a Micro-Controller and Micro-Computer. Understand the usage of Micro-Computer in IoT applications. Understand how Raspberry Pi interact with sensors and communicate over the network.
CO5:Expose the students with Server-side development in IoT applications. Understand how to develop and deploy applications in Arduino and NodeMCU. Understand how Arduino and NodeMCU communicate among themselves, sensors and the server. Understand how to build a full IoT app by integrating them with mobile applications.</td></tr>
<tr><td>Mapping between COs with PSOs</td><td>
<table>
<tr><td></td><td>PSO1</td><td>PSO2</td><td>PSO3</td><td>PSO4</td><td>PSO5</td><td>PSO6</td><td>PSO7</td><td>PSO8</td></tr>
<tr><td>CO1</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO2</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO3</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO4</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO5</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
</table>
</td></tr>
<tr><td>Pre-requisite</td><td>C , C++</td></tr>
<tr><td>Course Content</td><td>Unit 1: Introduction to IoT
1.1 What is IoT
1.2 IoT Applications
1.3 IoT Privacy and Security
1.3.1 Identification in Distributed Environment
1.3.2 Device Authentication
1.4 IoT Botnet</td></tr>
</table>

	Unit 2: Networking and Communication 2.1 Basics of Wireless Networking 2.1.1 CSMA/CA 2.2 IoT Network Protocols 2.2.1 BLE, Zigbee, LoRaWAN, RFID 2.3 IoT Data Protocols 2.3.1 CoAP, MQTT, XMPP, DDS **Unit 3: Sensors** 3.1 Introduction to Sensors 3.2 Types of Sensors & their working 3.3 Wireless Sensor Network 3.3.1 Introduction to WSN 3.3.2 Applications 3.3.3 Characteristics 3.3.4 Challenges 3.3.5 Components 3.4 Wireless Adhoc Network Vs Wireless Sensor Network **Unit 4: Micro-Controller: Arduino, NodeMCU** 4.1 Introduction to Microcontrollers 4.2 Arduino IDE 4.3 Arduino Architecture 4.4 Arduino Pin Diagram 4.5 Introduction to NodeMCU 4.6 NodeMCU Specifications and Applications 4.7 NodeMCU ESP8266 Pinout **Unit 5: IoT App Interaction & Introduction to Raspberry Pi** 5.1 Uploading sensor data to server 5.2 Reading sensor data from server 5.3 Controlling IoT device and components from Mobile or Web 5.4 Introduction to Microcomputers 5.5 Raspberry Pi Architecture 5.6 Raspberry Pi Pinout
Reference Books	1) Getting Started with Internet of Things – By Cuno Pfister, O'Reilly 2) Learning Internet of Things – By Peter Waher , Packt Publication 3) Internet of Things : A Hands-on Approach – By Arshdip Bahga and Vijay Madisetti 4) IoT Governance, Privacy and Security Issues, IERC 5) IoT Fundamentals: Networking Technologies, Protocols and Use Cases for the Internet of Things, Cisco Press 6) Fundamentals of IoT Communication Technologies, Springer 7) Microcontrollers – Architecture, Programming, Interfacing and system design – By Raj Kamal , Pearson 8) Exploring C for Microcontrollers : A hands on approach, Springer 9) Arduino for Dummies, Wiley 10) Make: Getting Started With Arduino - The Open Source Electronics Prototyping Platform, Shroff/Maker Media 11) ESP8266: Get Started With ESP8266 Programming NodeMCU Using Arduino IDE, Createspace Independent Pub 12) Internet of Things Projects with ESP32, Packt Publishing Limited

	13) Microprocessor Architecture, Programming and Applications with the 8085 - By Ramesh Gaonkar , Penram International Publishing 14) Raspberry Pi for Dummies , Wiley 15) Raspberry Pi User Guide – By Eben Upton and Garath Halfacree, Wiley
Teaching Methodology	Class work, Discussion, Self-Study, Seminars and/or Assignment
Evaluation Method	30% Internal assessment is based on class attendance, participation, class test, quiz, assignment, seminar, internal examination etc. 70% assessment is based on semester end University External examination

Course: 301: **Machine Learning**

(Elective)

<table>
<tr><td>Course Code</td><td>301</td></tr>
<tr><td>Course Title</td><td>Machine Learning (ML)</td></tr>
<tr><td>Credit</td><td>4</td></tr>
<tr><td>Teaching per Week</td><td>4 Hrs.</td></tr>
<tr><td>Minimum weeks per Semester</td><td>15 (Including Class work, examination, preparation, holidays etc.)</td></tr>
<tr><td>Review / Revision</td><td>June 2021</td></tr>
<tr><td>Purpose of Course</td><td>This course is an introduction for students to ML. The course also gives students an idea about various methods and algorithms of Machine Learning and application development of ML.</td></tr>
<tr><td>Course Objective</td><td>The objective of the course is –
1. To make student understand ML
2. To understand the various Machine Learning method
3. To explain various algorithms used in Machine learning
4. To introduce students with Programming in ML</td></tr>
<tr><td>Course Outcome</td><td>CO1 : Explain to the students the fundamental know how like the types of machine learning algorithms, applications and various required libraries, model selection etc. required to implement machine learning algorithms.
CO2 : Train students with can utilize various data wrangling techniques, data cleaning, data transformation, data reduction, data discretization, feature selection, and data visualization
CO3 : Train students who can implement supervised learning algorithms utilizing regression and classification algorithm on the real world dataset.
CO4 : Train student to have understanding of Artificial Neural Network and its working. Also, to make them capable of implementing ANN for solving real world problems using it.
CO5 : Explain to the students to use clustering and association rules as unsupervised learning method to solve complex problems.
CO6 : Train students to use machine learning techniques to solve real life complex problems.</td></tr>
<tr><td>Mapping between COs with PSOs</td><td>
<table>
<tr><th></th><th>PSO1</th><th>PSO2</th><th>PSO3</th><th>PSO4</th><th>PSO5</th><th>PSO6</th><th>PSO7</th><th>PSO8</th></tr>
<tr><td>CO1</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO2</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO3</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO4</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO5</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO6</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
</table>
</td></tr>
<tr><td>Pre-requisite</td><td>Basics of Linear Algebra, Statistics and Mathematics, Python Programming</td></tr>
<tr><td>Course Content</td><td>Unit 1 : Introduction
1.1. Definition of Machine Learning
1.2 Types of Machine Learning : Supervised , Unsupervised and Semi-supervised
1.3 Applications and tools of Machine Learning (Scikit learn library)</td></tr>
</table>

	1.4 Data Pre-processing, Selecting a model and training a model 1.5 Evaluating a performance of model and improving performance **Unit 2 : Data Wrangling** 2.1 Definition and goal of Data Wrangling 2.2 Importance of Data Wrangling 2.3 Data Pre-processing and Data Cleaning 2.3.1 Data Cleaning 2.3.2 Data Transformation 2.3.3 Data Reduction 2.3.4 Data Discretization 2.3.5 Feature Selection 2.4 Data Visualization **Unit 3 : Supervised Learning** 3.1 Supervised Learning : Classification and Regression 3.2 Regression 3.2.1 Simple and Multiple Regression 3.2.2 Linear Regression 3.2.3 Gradient Decent 3.2.4 Logistic Regression 3.3 Classification Algorithms : 3.3.1 K-nearest Neighbour 3.3.2 Support Vector Machines 3.3.3 Decision Trees 3.3.4 Naïve Bayes Classifier 3.4 Introduction to Support Vector Machine **Unit 4 : Neural Network** 4.1 Introduction to Neural Network 4.2 Architecture of Neural Network 4.3 Feedforward network and Backpropagation with example 4.4 Applications of Neural Network **Unit 5 : Unsupervised Learning** 5.1 Introduction to Unsupervised learning 5.2 Clustering 5.2.1 Selection of Clusters 5.2.2 Algorithms : 5.2.2.1 K – means clustering 5.2.2.2 Hierarchical Clustering 5.3 Association Rule Learning 5.3.1 Algorithms : 5.3.1.1 FP- Growth 5.3.1.2 Apriori Algorithm
Reference Books	1. “Machine Learning” by Tom M. Mitchell, McGraw Hill 2. “Understanding Machine Learning” by Shai Shalev-Shwartz, Shai Ben-David 3. “Machine Learning” by Anuradha Srinivasaraghavan, Vincy Joseph

	4. “Machine Learning using Python” by U Dinesh Kumar Manaranjan Pradhan 5. “Real-World Machine Learning” by Henrik Brink, Joseph Richards, Mark Fetherolf 6. “Python Machine Learning” by Sebastian Raschka and Vahid Mirjalili 7. “Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems” by Aurelien Geron 8. “Machine Learning in Action” by Peter Harrington 9. “Introduction to Machine Learning with Python : A Guide for Data Scientists” by Andreas C. Muller, Sarah Guido
Teaching Methodology	Class work, Discussion, Self-Study, Seminars and/or Assignment
Evaluation Method	30% Internal assessment is based on class attendance, participation, class test, quiz, assignment, seminar, internal examination etc. 70% assessment is based on semester end University External examination

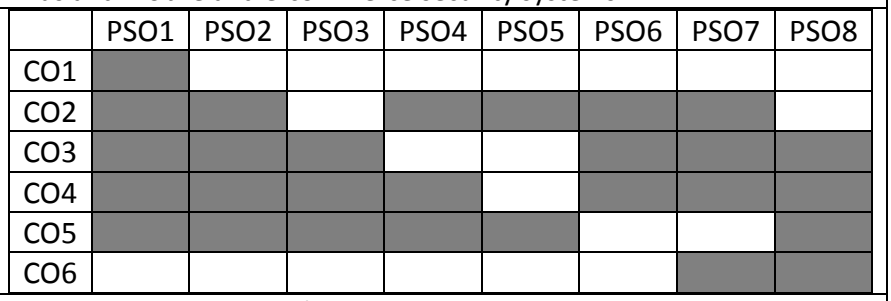
Course: 302: **Design Patterns**

Course Code	302
Course Title	**Design Patterns**
Credit	4
Teaching per Week	4 Hrs.
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Review / Revision	June 2021
Purpose of Course	The purpose of the course is to make student understand how Patterns can be implemented in various object oriented programming languages to solve real world problems.
Course Objective	The objective of the course is - 7. To study various Design Patterns 8. How these Patterns can be used to design better systems through Object Oriented Programming Languages
Course Outcome	CO1: Explain students about the various design patterns; their categories, and purpose. CO2: Explain the creational design patterns. CO3: Explain the structural design patterns. CO4: Explain the behavioural design patterns. CO5: Explain some more design patterns used in IT industry currently. CO6: Make students understand the applicability of design patterns practiced by IT companies and how effectively combine these patterns for effective software development.
Mapping between COs with PSOs	See table below
Pre-requisite	Object Oriented Programming, Software Engineering
Course Content	**Unit -1 Creational Patterns** 1.1 Singleton Pattern 1.2 Prototype Pattern 1.3 Builder Pattern 1.4 Factory Method Pattern 1.5 Abstract Factory Pattern **Unit-2 Structural Patterns** 2.1 Proxy Pattern 2.2 Decorator Pattern 2.3 Adapter Pattern 2.4 Façade Pattern 2.5 Flyweight Pattern 2.6 Composite Pattern 2.7 Bridge Pattern **Unit-3 Behavioural Pattern** 3.1 Visitor Pattern 3.2 Observer Pattern 3.3 Strategy Pattern 3.4 Template Method Pattern

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8
CO1	■							
CO2	■	■	■			■	■	
CO3	■	■	■			■	■	
CO4	■	■	■			■	■	
CO5	■		■	■	■	■	■	
CO6							■	■

	3.5 Command Pattern 3.6 Iterator Pattern 3.7 Memento Pattern 3.8 State Pattern 3.9 Mediator Pattern 3.10 Interpreter Pattern **Unit-4 Additional Design Patterns** 4.1 Simple Factory Pattern 4.2 Null Object Pattern 4.3 MVC Pattern **Unit-5 Pattern Applicability** 5.1 Security Patterns Repository 5.2 Patterns for Agile Development 5.3 Restful Service Patterns 5.4 Solution with semaphore 5.5 Patterns and Pattern combination in practice 5.6 Big Ball of Mud **Self-Study :** Pattern Languages
Reference Books	19. Design Patterns: Elements of Reusable Object-Oriented Software, Erich Gamma, Richard Helm, Ralph, John, Addision Wesley 20. Head First Design Patterns, Eric Freeman, O'Reilly 21. Design Patterns in C#, Vaskaran Sarcar, Apress 22. Design Patterns in Modern C++, Reusable Approaches for Object-Oriented Software Design, Dmitri Nesteruk, Apress 23. Modern C++ design: generic Programming and design patterns applied, Alexendrescu, Andrei, Addison-Wesley 24. Java Design Patterns: A Hands-on Experience with Real-World Examples, Vaskaran Sarcar, Apress
Teaching Methodology	Class work, Discussion, Self-Study, Seminars and/or Assignment
Evaluation Method	30 % internal assessment and 70% external assessment

Course: 303: **Network Essentials and its Security**

Course Code	303
Course Title	**Network Essentials and its Security**
Credit	4
Teaching per Week	4 Hrs.
Minimum weeks per Semester	15 (Including Class work, examination, preparation, holidays etc.)
Review / Revision	June 2020
Purpose of Course	To make students learn Network essentials and various security measures for the challenges to which the IT industry is exposed
Course Objective	To understand Network Management and its security
Course Outcome	CO1: Explain students familiar about network essentials. Make them familiar with various network devices like repeaters, bridge router, and gateway. CO2: Explain how to administer computer network. Make them understand various wide area network techniques, and explain network security. CO3: To understand cryptography, PKI, and digital signatures. CO4: Familiarize with various security services, and how certification and key management is handled in PKI. CO5: To make students understand various network security applications; covering internet protocol security, web security, email security, network management and its security. CO6: Make students understand access control, authentication schemes, firewalls, and virtual private network. Also to learn intrusion detection, virus and mobile and e-commerce security systems.
Mapping between COs with PSOs	
Pre-requisite	Computer Network, TCP/IP
Course Content	**Unit 1: Network Essentials** 1.1 Repeaters & Bridges 1.1.1 LAN Expansion 1.1.2 Repeaters 1.1.3 Bridges 1.1.4 How Bridges Work 1.1.5 Creating the routing table 1.1.6 Segmenting Network Traffic 1.1.7 Remote Bridges 1.1.8 Differentiating between bridges and repeaters 1.2 Routers & Gateways 1.2.1 Routers 1.2.2 How routers work 1.2.3 Routing benefits 1.2.4 Routing protocols 1.2.5 Routing V/S Bridging 1.2.6 B Routers 1.2.7 Gateways

1.2.8 How Gateways work

1.3 Network Administration

1.3.1 Bottlenecks

1.3.2 Simple Network Management Protocol

1.3.3 Data Protection

1.3.4 Backup Methods

1.3.5 Testing and Storage

1.3.6 Implementing a Backup System

1.3.7 Uninterruptible Power Suppliers

1.3.8 How Gateways work

1.3.9 Implementing Fault Tolerant Systems

1.3.10 RAID

1.3.11 Sector Sparing

1.4 Advance WAN Transmission

1.4.1 Overview

1.4.2 Multiplexing, Packet and Circuit Switching Networks

1.4.3 X.25

1.4.4 Asynchronous Transfer Mode (ATM)

1.4.5 ISDN

1.4.6 SONET

1.4.7 SMDS

Unit 2: Introduction to Network Security

Unit 3: Cryptography Techniques

3.1 Classical Cryptography

3.2 Conventional Cryptography

3.2.1 DES

3.3 Public – key Cryptography

3.3.1 RSA

3.4 Digital Signatures

3.4.1 DSA

Unit 4: Security Services

16.1 Message Integrity

16.2 Confidentiality and Authentication

16.3 Certification and Key Management

4.3.1 PKI

Unit 5: Network Security Applications

5.1 IP Security

5.1.1 IPsec

5.2 Web Security

5.2.1 SSL, TLS, SET

5.3 Electronic Mail Security

5.3.1 PGP, S/MIME

5.4 SNMP Security

Unit 6: Access Control in Computer Networks

6.1 Authentication Protocols and Services

6.1.1 Kerberos and X.509

6.2 Firewalls

6.3 Virtual Private Networks (VPNs)

Unit 7: System Security

7.1 Intrusion detection

	7.2 Viruses **Unit 8: Mobile System & E-Commerce Securities** 8.1 3G Security 8.2 E-Payment Systems 8.3 Fair Data Exchange
Reference Books	1. Cryptography and Network Security, 2/e, ISBN: 0-13-869017-0 - W. Stallings - Pearson Education, 1999 2. Network Security Essentials: Applications and Standards, 1/e, ISBN: 0-13-016093-8 - W. Stallings - Pearson Education, 2000 3. SSL and TLS: designing and building secure systems, ISBN: 0-201-61598-3 - E. Rescorla - Addison-Wesley, 2001 4. Implementing Secure Intranets and Extranets, ISBN: 0-89006-447-4 - K M Phaltankar - Artech House Publishers, 2000 5. Secure Electronic Commerce: Building the Infrastructure for Digital Signature and Encryption, ISBN: 0-13-027276-0 - W. Ford, and M. Baum - Prentice Hall, 2001 6. Security in Computing, ISBN: 0-13-185794-0, 2/e - C. P. Pfleeger - Prentice Hall, 1997 7. Building Internet Firewalls, 2/e, ISBN: 1-56592-871-7 - E. D. Zwicky, et al - O'Reilly, 2000 8. CDMA Cellular Mobile Communications & Network Security, ISBN: 0-13-598418-1 - M. Y. Rhee, - Prentice Hall, 1998 9. Journal of Computer Security 10. ACM Transactions on Information and System Security 11. ACM Conference on Computer and Communications Security 12. IEEE Symposium on Security and Privacy 13. Internet documents - RFCs (Request for Comments) 14. Guide to Networking Essentials, Fourth Edition - Greg Tomsho, et al 15. Computer Networking Essentials - Debra Littlejohn Shinder 16. Networking Essentials: Hands-On, Self-Paced Training for Supporting Local and Wide Area Networks - Microsoft Corporation (Corporate Author) 17. Computer Network - A. S. Tanenbaum
Teaching Methodology	Class work, Discussion, Self-Study, Seminars and/or Assignment
Evaluation Method	30% Internal assessment is based on class attendance, participation, class test, quiz, assignment, seminar, internal examination etc. 70% assessment is based on semester end University External examination

Course: 304: **Advanced Database Administration**

Course Code	304
Course Title	**Advanced Database Administration**
Credit	4
Teaching per Week	4 Hrs.
Minimum weeks/ Semester	15 (Including Class work, examination, preparation, holidays etc.)
Review / Revision	June 2020
Purpose of Course	Understanding advanced database administration
Course Objective	To learn advanced database administration, database tuning and maintenance
Course Outcome	CO1 : To provide strong foundation in Advanced Database Administration concepts from an industry perspective. CO2 : To have thorough understanding of Oracle Database Management System internal architecture. CO3 : To understand the security aspects and user management. CO4 : To apply and learn various Oracle utilities CO5 : To learn how to practically tune the database to optimize the overall performance. CO6 : To learn and implement Backup and Recovery.
Mapping between COs with PSOs	

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8
CO1								
CO2								
CO3								
CO4								
CO5								
CO6								

Pre-requisite	RDBMS
Course Content	**Unit 1. Oracle10g Instance creation and management** 1.1 Oracle Instance 1.2 Installing Oracle 1.3 Oracle Optimal Flexible Architecture (OFA) 1.4 Locating initialization, listener.ora & sqlnet.ora files 1.5 Finding the alert log 1.6 Common environment variables 1.7 Structures in an Oracle Instance 1.8 Oracle Memory Structures, SGA and PGA 1.9 Oracle Processes and their purposes 1.10 Startup, nomount, mount and open database commands **Unit 2. Oracle10g Database Architecture** 2.1 Oraclde10g management framework 2.1 Using the Database Creation Assistant (DBA) 2.3 Creating and dropping a database 2.4 Tablespaces 2.5 Tables and Indexes 2.6 Clusters 2.7 Partitioning of Tables and Indexes 2.8 Gathering and applying patches **Unit 3. Concurrency Management** 3.1 Transactions, serialization, locks and latches 3.2 Lock modes 3.3 Detecting and resolving lock conflicts

	3.4 Managing deadlocks **Unit 4. Interfacing with Oracle** 4.1 Oracle transaction management 4.2 Using SQL*Plus and iSQL*Plus 4.3 Using embedded Oracle with Pro*C & Java 4.4 PL/SQL & Triggers 4.5 Pining PL/SQL packages & compiling PL/SQL 4.6 System-level triggers – startup trigger, logon trigger, PL/SQL error trigger **Unit 5. Oracle*Net** 5.1 Basic Network structure 5.2 Oracle*Net Files 5.3 Multi-threaded server 5.4 Create additional listeners 5.5 Create Oracle Net service aliases 5.6 Configure connect time failover 5.7 Oracle*Net names resolution **Unit 6. Tablespace Management Overview** 6.1 Dictionary Managed Tablespaces 6.2 Locally Managed Tablespaces 6.3 Automatic Segment Space Management 6.4 Moving tablespaces online and offline **Unit 7. UNDO Tablespace Management** 7.1 Use of undo segments 7.2 Creating an undo tablespace 7.3 User managed undo tablespaces 7.4 Automatic undo management 7.5 Monitor & Configure undo retention 7.6 Use the Undo Advisor 7.7 Size the undo tablespace **Unit 8. Oracle Utilities** 8.1 Datapump - Import/export 8.2 SQL*Loader 8.3 Oracle Streams 8.4 Automatic Database Diagnostic Monitor 8.5 Automatic Tuning Optimizer 8.6 Automatic Shared Memory Tuning **Unit 9. Oracle Performance Tuning** 9.1 Locate invalid and unusable objects 9.2 Gather SQL optimizer statistics with dbms_stats 9.3 Basic Oracle performance metrics 9.4 Use OEM and dbms_alert to set warning and critical alert thresholds 9.5 The SQL Tuning Advisor 9.6 The SQL Access Advisor 9.6 Interpreting server generated alerts 9.7 Oracle advisory utilities vdb_cache_advice, vshared pool_advice, v$pga_aggregate_target_advice 9.8 Using OEM performance screens

	9.9 Fixing performance issues **Unit 10. User Management** 10.1 Creating Users 10.2 Altering users 10.3 User Profiles 10.4 User resource groups 10.5 Granting privileges & roles 10.6 Auditing user activity with dbms_audit **Unit 11. Oracle Security** 11.1 Password use in Oracle, Password encryption and password aging, External authentication, Using Single sign-on (SSO) 11.2 Object security 11.3 Virtual Private Databases (VPD) in Oracle 11.4 Oracle "grant execute" security 11.5 Use of Roles in Oracle 11.6 Register for security updates **Unit 12. Backup & Recovery** 12.1 Oracle backup & recovery planning 12.2 Parallel instance recovery 12.3 Basics of checkpoints, redo log files, and archived log files 12.4 Using ARCHIVELOG mode 12.5 Creating consistent Oracle backups 12.6 Online hot backups 12.7 Incremental Oracle backups 12.8 Automating database backups with dbms_scheduler 12.9 Monitor the flash recovery area 12.10 Recovering from loss of a Control file 12.11 Recovering from loss of a Redo log file 12.12 Recovering from loss of a system-critical data file 12.13 Recovering from loss of a non system-critical data file
Reference Books	1. Essentials : Oracle Database 10g by Rick Greenwald, Robert Stackowiak, Jonathan Stern, O'Reilly 2. Oracle High Performance Tuning for 9i and 10g by Gavin Powell, Digital Press 3. Oracle Database 10g, DBA Handbook by Loney, Kevin, Bryla, Bob, Oracle Press 4. Oracle Database 10g - The Complete Reference by Loney, Kevin, Oracle Press 5. Oracle Database 10g: A Beginner's Guide by Micheal Abbey, Ian Abramson Osborne, Oracle Press Series
Teaching Methodology	Class work, Discussion, Self-Study, Seminars and/or Assignment
Evaluation Method	30% Internal assessment is based on class attendance, participation, class test, quiz, assignment, seminar, internal examination etc. 70% assessment is based on semester end University External examination

Course: 305: **Open Source Web Based Programming**

<table>
<tr><td>Course Code</td><td>305</td></tr>
<tr><td>Course Title</td><td>Open Source Web Based Programming</td></tr>
<tr><td>Credit</td><td>4</td></tr>
<tr><td>Teaching per Week</td><td>4 Hrs.</td></tr>
<tr><td>Minimum weeks/ Semester</td><td>15 (Including Class work, examination, preparation, holidays etc.)</td></tr>
<tr><td>Review / Revision</td><td>June 2020</td></tr>
<tr><td>Purpose of Course</td><td>This course helps students to understand fundamentals of Open Source web based Programming. The course also imparts students learning about Open source web based scripting language PHP and Mysql database. It also includes MVC or Three tier architecture of web based programming and Javascript technology like ReactJS.</td></tr>
<tr><td>Course Objective</td><td>Student will learn fundamentals and advance topics of Open source Web technology</td></tr>
<tr><td>Course Outcome</td><td>CO1 : Explain students the fundamental as well as Advanced aspects of the Open Source Web based Technology.
CO2 : Train students about react JS and difference between React JS and React Native.
CO3 : Train students to understand MVC structure and it’s benefits.
CO4 : Explain and train students to deal with possible problem while developing websites and it’s solution.
CO5 : Expose the students with the analysis and development process of Websites.
CO6 : After studying the course, students will be able to understand how Open source web technology works. They will also be able to create database driven Professional Websites.</td></tr>
<tr><td>Mapping between COs with PSOs</td><td>
<table>
<tr><th></th><th>PSO1</th><th>PSO2</th><th>PSO3</th><th>PSO4</th><th>PSO5</th><th>PSO6</th><th>PSO7</th><th>PSO8</th></tr>
<tr><td>CO1</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO2</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO3</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO4</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO5</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td>CO6</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
</table>
</td></tr>
<tr><td>Pre-requisite</td><td>Knowledge of HTML, Javascript and SQL</td></tr>
<tr><td>Course Content</td><td>Unit 1 : Introduction to Open source Web based Programming
1.1 Introduction to PHP & MySql
1.2 Installation of PHP and MySql
1.3 Language Characteristics & Features
1.4 Operators and Variables, Control Structures, Looping and Error handling
1.5 PHP functions
1.5.1 String Functions
1.5.2 Array Functions
1.5.3 Mathematical Functions
1.5.4 Graphics Library (GD Support)
1.5.6 Date and Time Functions
2.5.7 Misc. Function
1.6 State management Techniques
1.7 Object Oriented Features of PHP
1.7.1 Classes and Objects
1.7.2 Use of constructors
1.7.3 Serialization</td></tr>
</table>

	1.7.4 Inheritance **Unit 2 : MySQL database server** 2.1 Configuring the MySQL Server 2.2 MySQL Tables, Displaying MySQL Database , Adding and removing user access 1.3 Database connection and data processing functions **Unit 3 : Advance PHP** 3.1 Ajax Basics 3.1.1HTTP Request and Response Fundamentals 3.1.2 The XMLHttpRequest Object XMLHttpRequest Methods 3.1.3 XMLHttpRequest Properties 3.1.4 Cross-Browser Usage Sending a Request to the Server 3.1.5 PHP and Ajax Client-Driven Communication 3.1.6 Server-Side Processing Expanding and Contracting Content 3.1.7 Form Validation 3.1.8 Ajax-Based Database Querying 3.2 XML 3.3 Web services **Unit 4 : MVC** 4.1 Introduction to MVC 4.2 CodeIgniter: Introduction, Features and Application Flow Chart 4.3 Controller 4.4 Views 4.5 Models 4.6 Helpers 4.7 Creating and Usage of Libraries and Helpers 4.8 URL Routing 4.9 Error Handling 4.10 Profiling Application **Unit 5 : Introduction to React JS** 5.1 What is React JS 5.2 Environment Setup 5.3 JSX and ES6 5.4 Components 5.5 Props and State 5.6 Components API and Lifecycle 5.7 Forms and Events 5.8 Difference between React JS and React Native
Reference Books	1. Beginning PHP, Apache, MySQL Web Development - Elizabeth Naramore, Jason Gerner , Yann Le Scouarnec,Jeremy Stolz,Michael K. Glass, Gary Mailer – Wrox Publication 2. Professional PHP Programming - Jesus Castagnetto ,Wrox Press Ltd 3. Beginning PHP and MySQL: From Novice to Professional - W. Jason Gilmore, Apress 4. Php: The Complete Reference - Steven Holzner, Tata Mcgraw Hill Education Private Limited

	5. AJAX and PHP: Building Responsive Web Applications - Bogdan Brinzarea, Cristian Darie packtpub 6. CodeIgniter for Rapid PHP Application Development - David Upton ,packtpub 7. Professional CodeIgniter- Thomas Myer, Wrox Press Ltd 8. Learning React - Kirupa Chinnathambi , Paperback – 2018 9. Mastering React- Adam Horton and Ryan Vice, packtpub 10. Php manual – www. Php.com
Teaching Methodology	Class work, Discussion, Self Study, Seminars and/or Assignment
Evaluation Method	30% Internal assessment is based on class attendance, participation, class test, quiz, assignment, seminar, internal examination etc. 70% assessment is based on semester end University External examination

Course: 306: **Programming Skills XI**

Course Code	306
Course Title	**Programming Skills XI**
Credit	2
Teaching per Week	3 Hrs.
Minimum weeks/ Semester	15 (Including Lab. work, examination, preparation, holidays etc.)
Review / Revision	June 2020
Purpose of Course	This course helps students to implement the Unix Internals with shell programming/IOT practically.
Course Objective	Learning to implement fundamentals and advanced topics of Unix Internals with Shell Scripting/IOT practically
Pre-requisite	Practical programming in desktop environment / Embedded Technology
Course Outcome	After studying the course, students will be able to practically work on advanced technology platforms of Unix Internals with Shell Scripting /IOT.
Course Content	Practical based on paper no 501. Separate journal to be prepared for this subject based on 501.
Reference Books	-----
Teaching Methodology	Lab work, Practical Programming Exercises (to be documented in a separate journal), Self-study, and/or Assignment
Evaluation Method	30% Internal assessment is based on Lab attendance, practical test, practical internal examination etc. 70% assessment is based on semester end University External practical examination

Course: 307: **Programming Skills XII**

Course Code	307
Course Title	**Programming Skills XII**
Credit	3
Teaching per Week	3 Hrs.
Minimum weeks/ Semester	15 (Including Lab. work, examination, preparation, holidays etc.)
Review / Revision	June 2020
Purpose of Course	This course helps students to understand the Networking and its security practically
Course Objective	Learning to implement Network and its security practically.
Pre-requisite	Network fundamentals
Course Outcome	After studying the course, students will be able to practically implement network and its security
Course Content	Practical based on paper no 503. Separate journal to be prepared for this subject 503.
Reference Books	-----
Teaching Methodology	Lab work, Practical Exercises (to be documented in a separate journal), Self-study, and/or Assignment
Evaluation Method	30% Internal assessment is based on Lab attendance, practical test, practical internal examination etc. 70% assessment is based on semester end University External practical examination

Course: 308: **Programming Skills XIII**

Course Code	308
Course Title	**Programming Skills XIII**
Credit	2
Teaching per Week	3 Hrs.
Minimum weeks/ Semester	15 (Including Lab. work, examination, preparation, holidays etc.)
Review / Revision	June 2020
Purpose of Course	This course helps students to learn network administration
Course Objective	Students will learn administration of network practically
Pre-requisite	Networking fundamentals
Course Outcome	After studying the course, students will be able to practically perform administrative tasks of networks practically
Course Content	Practical based on paper no 504. Separate journal to be prepared for this subject 504.
Reference Books	-----
Teaching Methodology	Lab work, Practical Exercises (to be documented in a separate journal), Self-study, and/or Assignment
Evaluation Method	30% Internal assessment is based on Lab attendance, practical test, practical internal examination etc. 70% assessment is based on semester end University External practical examination

Course: 309: **Programming Skills XIV**

Course Code	309
Course Title	**Programming Skills XIV**
Credit	3
Teaching per Week	3 Hrs.
Minimum weeks/ Semester	15 (Including Lab. work, examination, preparation, holidays etc.)
Review / Revision	June 2020
Purpose of Course	This course helps students to implement the basic and advanced concepts of PHP/MySql practically.
Course Objective	Learning to develop and deploy websites using PHP/MySql practically.
Pre-requisite	Basic scripting, programming, html.
Course Outcome	After studying the course, students will be able to practically develop dynamic websites using PHP/MySql.
Course Content	Practical based on paper no 505. Separate journal to be prepared for this subject based on 505.
Reference Books	-----
Teaching Methodology	Lab work, Practical Programming Exercises (to be documented in a separate journal), Self-study, and/or Assignment
Evaluation Method	30% Internal assessment is based on Lab attendance, practical test, practical internal examination etc. 70% assessment is based on semester end University External practical examination

MCA 4th Semester

Course: 401: **Seminar**

Course Code	401
Course Title	**Seminar**
Credit	6
Review / Revision	June 2021

- The students are required to prepare a seminar on a relevant topic concerning the subject of interest of the student; as well as latest technology.
- The students must prepare documentation of the seminar.
- At the end of the semester, the students have to submit the seminar reports in spiral bounded form to the institution.
- Seminar Completion Certificate issued by the institute is mandatory for appearing in Seminar Presentations.
- The Seminar Presentation will be conducted as per the University exam schedule.
  The students have to submit the following reports at the institution:

  1. Seminar Topic Chosen
  2. Institution Certificate for Seminar

Course: 402: **Project**

Course Code	402
Course Title	**Project**
Credit	24
Review / Revision	June 2021

- The students are required to carry out full time software development project in a company.
- The students must prepare documentation of the project completed as per the guidelines given by the institute.
- At the end of the semester, the students have to submit the project reports in bounded form to the institution.
- Project Completion Certificate issued by the institute is mandatory for appearing in Project Presentation and Viva – Voce.
- The Project Presentation and Viva – Voce will be conducted as per the University exam schedule.

  The students have to submit the following reports at the institution:

  1. Project Joining Report
  2. Progress Reports
  3. Project Completion Certificate from the company
  4. Institution Certificate for Project
  5. Non-disclosure of Source Code Certificate (In case the student is unable to submit project source code)