ANNEXURE A

Bachelor of Computer Applications / Bachelor of Computer Applications (Honours)

Programme Specific Outcomes (PSO)

PSO1. Explore concepts & processes of computer applications (logic & programming, software development, data analytics etc.) and experience a conducive environment in cultivating skills for thriving professional career and higher studies.

PSO2. Develop, evaluate and propose ideas and computer application solutions to real computing problems, culminating into a modern, easy to use tool, by a larger section of the society with longevity.

PSO3. Adapt to rapid changes in tools, technology & work environment with an understanding of societal responsibilities, professional ethics, and good interpersonal skills as an individual & team leader, relevant to computer application professionals.

PSO4. Pursue higher studies, undertake research, take up professional careers in the IT & ITeS sector, or become Entrepreneurs.

Goa University

Programme Structure for Semester I to VIII Under Graduate Programme - Computer Applications (Revised)

Semester	Major -Core	Minor	MC	AEC	SEC	I	D	VAC	Total Credits	Exit
I	Major- 1 CSA-100 (Problem Solving and Programming) (3T+ 1P)	Minor -1 CSA-111 (Computer System Fundamentals) (4T) OR CSA-112 (Open Source Software) (4T)	MC-1 CSA-131 (E-Commerce) (3T) OR MC-2 CSA-132 (Green Computing) (3T)	One course to be opted from the available ability enhancement courses approved by the Board of Studies in English	SEC-1 CSA-141 (Office Automation and PC Troubleshooting) (1T + 2P) OR CSA-142 (Python Programming) (1T + 2P)			Two courses to be opted from the available common value added courses approved by the respective Board of Studies.	20	
II				One course to be opted from the available ability enhancement courses approved by the Board of Studies in English	SEC-2 CSA-143 (Data Analytics using Spreadsheets) (1T + 2P) OR CSA-144 (2D Animation) (1T + 2P)			Two courses to be opted from the available common value added courses approved by the respective Board of Studies.	20	EXT-1 CSA-161 (PC Troubleshooting & Networking) (2T + 2P)

III	Major- 2 CSA-200 (Data Structures) (3T+1P) Major- 3 CSA-201 (Database Management Systems) (3T+1P)	Minor -3 CSA-211 (Reasoning Techniques) (3T + 1 Tutorial) OR CSA-212 (Techpreunership Development) (3T + 1 Tutorial) OR CSA-213 (Computer Organization & Architecture Fundamentals) (3T + 1P)	MC-3 CSA-231 (Cyber Law and Ethics) (3T) OR CSA-232 (Digital Ecosystem) (3T) OR CSA-233 (Website Design) (2T+1P) OR CSA-234 (ERP) (2T+1P) OR CSA-235 (Latex) (2T+1P) OR CSA-236 (Multimedia Essentials) (2T+1P)	One course to be opted from the available ability enhancement courses approved by the concerned Board of Studies	SEC-3 CSA-241 (Multimedia Applications) (1T + 2P) OR CSA-242 (Search Engine Optimization) (1T + 2P) OR CSA-243 (3D Animation) (1T + 2P)				20	
IV	Major-4 CSA-202 (Web App Development) (3P + 1Tutorial) Major-5 CSA-203 (Agile Methodologies) (3T+1P) Major-6 CSA-204 (Object Oriented Concepts) (3T+1P) Major-7 CSA-205 (Web Technology) (2T)	Minor-4 VET CSA-221 (Digital Marketing Fundamentals) (3T + 1P) OR CSA-222 (Data Analysis) (3T + 1P) OR CSA-223 (Advanced JavaScript) (3T + 1 P)		One course to be opted from the available ability enhancement courses approved by the concerned Board of Studies					20	EXT-2 CSA-261 (Digital Media Marketing & Analytics) (2T + 2P)

V	Major-8 CSA-300 (UI- UX Design) (3T+1P) Major- 9 CSA-301 (Full Stack Development) (3P + 1Tutorial) Major- 10 CSA-302 (Cloud Computing) (3T + 1P) Major- 11 CSA-303 (Internet Technologies) (2T)	Minor VET -5 CSA-321 (Internship) (4)				CSA-361 Summer Internship (2)			20	
VI	Major-12 CSA-304 (Cyber Security) (3T + 1P) Major- 13 CSA-305 (Mobile App Development) (3P + 1 Tutorial) Major- 14 CSA-306 (Machine Learning) (3T + 1P) Major- 15 CSA-307 (Project) (4)	Minor VET-6 CSA-323 (Social Media Marketing & Analytics) (3T + 1P) OR CSA-324 (E-Commerce Applications) (3T + 1P) OR CSA-325 (Modern Frameworks) (3T + 1P)							20	

VII	**Major-16 CSA-400 (Statistical Tools) (3T + 1P)** **Major- 17 CSA-401 (DevOps) (3P + 1 Tutorial)** **Major- 18 CSA-402 (Software Design Patterns) (3T + 1P)** **Major- 19 CSA-403 (NLP Applications) (3T + 1P)**	**Minor -7 CSA-411 (Project Management) (3T + 1P) OR CSA-412 (Dashboard Development) (3T + 1P) OR CSA-413 (Introduction to Quantum Computing) (3T + 1P)**							**20**	
VIII	**Major-20 CSA-404 (Introduction to Functional Programming) (3T + 1P)** **Major-21 CSA-405 (Information Systems Audit) (3T + 1 Tutorial)** **Major- 22 CSA-406 (Internet of Things) (3T + 1P)** **Major- 23 CSA-407 (Research Methodology) (3T + 1P)**	**Minor-8 CSA-414 (Interactive Media) (3T + 1P) OR CSA-415 (Game Design) (3T + 1P) OR CSA-416 (Educational Technology) (3T + 1P) OR CSA-417 (Blockchain Technology) (3T + 1P)**							**20**	

First Year - Semester I and Semester II

Name of the Programme: Bachelor of Computer Applications
Course Code: CSA-100
Title of the Course: Problem-Solving and Programming
Number of Credits: 4 (3T + 1P)
Effective from AY: 2024-25

Pre-requisites for the Course:	None	
Course Objectives:	1. To understand the concepts and techniques of problem-solving. 2. To analyze, understand, and build logic to solve basic problems. 3. To design Algorithms and flowcharts for better understanding and documentation for accurate implementation of the problem. 4. To code and implement a well-structured, robust programming logic using a suitable programming language.	
Units	**Content**	**No of hours 75 (45T+30P)**
I	**Introduction to Problem Solving** ● Problem-Solving Life Cycle — Understanding the Problem Statement, Analyzing the Problem, using Hierarchy charts, and Expressing Program logic using flowcharts / Pseudocode. ● Structured Programming concept ● Modular Programming-Top-Down Design, Bottom-Up Design, Stepwise Refinement **Understanding basic Problem Solving Tools** ● Algorithms: Definition and Attributes, Algorithm Constructs, Statements: Input-Output, Decision-making, and Looping, Examples ● Flowchart: Definition and its attributes, symbols, Statements: Input-Output, Decision-Making & Looping, Module representation, Drawing conventions and standards, Examples. ● Pseudo-code: Definition and its attributes, constructs, and Examples **Basic Program Structures** ● Data and its types (Integer, Floating-point, Character, String), Constants and variables, scope, instructions, and their types, how the computer stores data, Operators (Arithmetic, Assignment,	**15**

	Relational, Logical, etc), Expressions and Equations, Evaluation of expressions, and keywords. ● Local and Global Variables, Parameters, return values, naming conventions and standards, Understanding literals, syntax and semantics, functions, and modules.	
II	**Basic Sequential Instructions** ● Sequential statements using operators, constants, variables, operands, expressions, and equations. ● ***Activity:*** *Apply the concepts learn to design the algorithms of at least 2 basic problems. Represent it using flowchart and pseudo-code.* **Debugging & Documentation** ● Definition, Types, Need, and how to do it. **Problem-solving with Decisions** ● The Decision Logic Structure, Multiple If/Then/Else Instructions, Using Straight-Through Logic, Using Positive and Negative Logic, Logic Conversion, Decision Tables, and Case Logic Structure. ● ***Activity:*** *Apply the concepts learned to design the algorithms for at least 4 basic problems. Represent it using flowchart and pseudo-code.*	**15**
III	**Problem Solving with Loops** ● The Loop Logic Structure, Incrementing, Accumulating, While/While End, Repeat/Until, Automatic-Counter Loop, Nested Loops, Indicators (flags). ● Iterating, accessing, and modifying array elements. ● ***Activity:*** *Apply the concepts learnt to design the algorithms of at least 3 basic problems. Represent it using flowchart and pseudo-code.* **Problem Solving with Arrays** ● Arrays Concepts: One-dimensional Arrays, Creating, Concept of Strings, String as an array of characters. ● ***Activity:*** *Apply the concepts learnt to design the algorithms of at least 3 basic problems. Represent it using flowchart and pseudo-code.* **Understanding functions** ● Functions: Definition and its need and constructs, designing	**15**

	simpler functions, function communication using arguments, and return statements. scope of functions, function declaration and prototype, call by Value, and Call by reference. ● Concept of Recursive functions: why, when, and how. Designing recursive functions and recursive calls. Basecase and recursive case. ● *Apply the concepts learnt to design the algorithms of at least three basic problems. Represent it using flowchart and pseudo-code.*	
IV	**Practical work** Using any suitable programming language like C, the concepts learned in the units from I to III are required to be implemented practically. The broad area of practical problems is mentioned/ suggested below.	**Practical Hours(30)**
Week 1 & 2 [These practical's should be done using pen, paper,and using buddy learning strategy]	1. For each of the following tasks, write a set of numbered, step-by-step instructions (a solution) so complete that another person can perform the task without asking questions. Define the knowledge base of this person by listing what you expect the person to know to follow your directions. For example, for task "a" (below), make a cup of cocoa, the knowledge base might include such things as knowledge of milk or water, a refrigerator, pan, spoon, cocoa, cup, range top or microwave, and so forth. a. Make a cup of cocoa. b. Sharpen a pencil. c. Walk from the classroom to the student lounge, your dorm, or the cafeteria. d. Start a car(include directions regarding what to do if the car doesn't start). e. Get a glass of water from your kitchen. f. Start your computer. 2. Test your solution in problem 1 by giving your instructions to another person to see whether he or she can accomplish the task without your help. If they can't, modify your solution so that the person can accomplish the task. Check the solution again by Giving the instructions to another person.	**04**
Week3 &4	3. Basic Program Structures ● At least 10 basic programming problems related to Module II to be completed during the practical sessions. ● More programs may be given to the learners to complete and practice as part of their Practice Work.	**04**

Week5 &6	4. Basic Sequential Instructions ● At least 08 programming problems to be completed during the practical sessions. ● More programs may be given to the learners to complete and practice as part of their Practice Work. 5. Debugging & Documentation ● Debug & Document at-least 02 problems. ● More programs may be given to the learners to complete and practice as part of their Practice Work.	**04**
Week7, 8&9	6. Problem Solving with Decisions ● At least 08 programming problems to be completed during the practical sessions. ● Debug & Document at least 02 problems. ● More programs may be given to the learners to complete and practice as part of their Practice Work.	**06**
Week10 &11	6. Problem Solving with Loops ● At least 08 programming problems to be completed during the practical sessions. ● Debug & Document at least 02 problems. ● More programs may be given to the learners to complete and practice as part of their Practice Work.	**04**
Week12 &13	7. Understanding functions ● At least 08 programming problems to be completed during the practical sessions. ● Debug & Document at least 02 problems. ● More programs may be given to the learners to complete and practice as part of their Practice Work.	**04**
Week14 &15	8. Problem Solving with Arrays ● At least 08 programming problems to be completed during the practical sessions. ● Debug & Document at least 02 problems. ● More programs may be given to the learners to complete and practice as part of their Practice Work.	**04**
Pedagogy:	**Suggestedstrategiesforusetoacceleratetheattainmentofthevariouscourse outcomes.** 1. The lecture method need not be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. You may use a. Video/Animation to explain various concepts. b. Collaborative, Peer, Flipped Learning, etc. 2. Ask at least three HOT (Higher-Order Thinking) questions in the class, which promotes critical thinking.	

	3. Adopt Problem-Based Learning (PBL), which fosters students' Analytical skills, and develops design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it. 4. Introduce Topics in manifold representations. 5. Show the different ways to solve the same problem and encourage the students to come up with their own creative ways to solve them. 6. Discuss how every concept can be applied to the real world when that's possible, it helps improve the student's understanding 7. To promote self-learning, give at least one assignment (equivalent to 50% assignment weightage) where they can complete one MOOCs (certificate or equivalent) course out of lecture hour. Test their understanding through quizzes or presentations.
References/ Readings:	**Main Reading:** 1. Forouzan, B.A., & Gilberg, R.F. (2007).*A Structured Programming Approach Using C*. Cengage Learning India. 2. Kuppuswamy, S., Malliga, S., Kanimozhi Selvi, C.S., & Kousalya, K. (2019). Problem Solving and Programming. Tata McGraw Hill. 3. Sprankle, M., & Hubbard, J.(2013).*Problem-solving and Programming Concepts*. Pearson Education India. **Additional Reading:** 1. K. N. King (2008). C Programming: A Modern Approach, 2nd Edition 2nd Edition, W. W. Norton & Company 2. Perry Greg, Miller Dean (2013). C Programming Absolute Beginner's Guide 3rd Edition, Kindle Edition. Que Publishing.
Course Outcomes:	On completion of the course, students will be able to: **CO1.** Remember the basic concepts and terminologies of problem-solving, algorithms, flowcharts, pseudo-code, language syntax, and debugging. **CO2.**Understand basic computing concepts, algorithm design, flowchart design, pseudo-code, programming constructs, and debugging. **CO3.**Apply problem-solving and programming concepts and design solutions to simpler problems using algorithms, flowcharts, and pseudocode. **CO4.**Code, debug, and analyze well-structured programming logic using suitable Programming language/s.

Name of the Programme: Bachelor of Computer Applications Course
Code: CSA-111
Title of the Course: Computer System Fundamentals
Number of Credits: 4T
Effective from AY: 2024-25

Prerequisites for the Course:	Nil	
Course Objectives:	1. To remember the basics of computers, Computer Organization, Number Systems, process management, memory management, I/O Management, and File management concepts. 2. To understand the concepts of process management, memory systems, I/O devices, and File Management Systems 3. To apply the concepts of process management in handling deadlock situations. 4. To analyze the appropriate type of memory for a given scenario.	
Units	**Content**	**No of hours 60**
I	**Fundamentals of Computer** ● **Evolution of Computer** Operating Systems – Definition, Introduction to Major Functions/Services, OS Structure, Relationship between Kernel, OS, Hardware, Block Diagram of computer, Evolution of Computers - Computer Generations ● **Computer Organization**: Input Unit, Output Unit, Structure and functions of Central Processing Unit, Arithmetic Logic Unit, and Control Unit, Von Neumann Machine Architecture, Computer Function – Top Level View, Instruction Cycle with and without interrupts (State diagram), Classes of Interrupts, Multiple interrupts, Interconnection structures, Bus Interconnection. ● **Number System** Conversion(Binary, Decimal, Octal, Hexa-Decimal), Data Representation, Binary Arithmetic, One's and Two's Complement.	**15**
II	**Processes & Process Management** ● **Process** Definition, Process Control Block, Process States, Operations on Process. ● **Threads** Processes and Threads, Multithreading, Types of Threads. ● **Process Scheduling**	**15**

	Introduction, Scheduling Criteria, Scheduling Algorithms. ● **Concurrency/Process Coordination** Synchronization Principles, Mutual Exclusion, The Critical-Section Problem, Peterson's Solution ● **Deadlock** Principles, Deadlock Handling Methods, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery from Deadlock	
III	**Memory Management** ● **Memory Management Concepts** Memory Partitioning (Fixed and dynamic), Swapping, Paging, and Segmentation. ● **Virtual Memory** Introduction, Demand Paging, Page Replacement- Algorithms, Thrashing. ● **Cache Memory** Characteristics of Memory Systems, Memory Hierarchy, Cache Memory Principles. ● **Internal Memory** Semiconductor main memory–SRAM, DRAM, Types of ROM. ● **External Memory** Magnetic Disk, SSD, Optical memory, Magnetic Tape	**15**
IV	**Input/Output and File Management** ● **I/O Management** I/O devices, Organization of I/O (programmed, interrupt driven and DMA), I/O Buffering, Disk Scheduling- Algorithms, RAID. ● **File Management** Overview–File and File Systems, File Structure, File Management System, File Organization and Access, File Directories, Directory Structure, File Sharing,	**15**
Pedagogy:	**Suggested strategies for use to accelerate the attainment of the various course outcomes.** 1. The lecture method need not be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. You may use a. Video/Animation to explain various concepts. b. Collaborative, Peer, Flipped Learning, etc. 2. Ask at least three HOT(Higher-Order Thinking) questions in class, which promotes critical thinking. 3. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, and develops design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it. 4. Introduce Topics in manifold representations. 5. Show the different ways to solve the same problem and encourage the	

	students to come up with their own creative ways to solve them. 6. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding 7. To promote self-learning, give at least one assignment where they can complete one MOOCs (certificate or equivalent) course out of lecture hour. Test their understanding through quizzes or presentations.
References / Readings:	**Main Reading:** 1. Stallings, W.(2012). *Operating Systems: Internals and Design Principles*. Pearson Education. 2. Stallings, W.(2013). *Computer Organization and Architecture: Designing for Performance*. Pearson Education. **Additional Reading:** 1. Sinha, P. ,& Sinha, P.(2016). *Computer Fundamentals*. BPB Publications. 2. Silberschatz, A., Galvin, P.B., & Gagne, G. (2006). *Operating System Principles*. Wiley India.
Course Outcomes:	On completion of the course, students will be able to: **CO1.** Remember the basics of computers, Computer Organization, Number Systems, process management, memory management, I/O Management, and File management concepts. **CO2.** Understand the concepts of process management, memory systems, I/O devices, and File Management Systems **CO3.** Apply the concepts of process management in handling deadlock situations. **CO4.** Analyse an Appropriate type of memory for a given scenario.

Name of the Programme: **Bachelor of Computer Applications**
Course code: CSA-112
Title of the Course: Open-Source Software
Number of Credits: 4T
Effective from AY:2024-25

Prerequisite for the Course:	None	
Course Objectives:	1. To remember the significance of Open-Source software practices and guidelines 2. To understand the Open-Source ecosystem, its use, impact, and importance. 3. To apply open-source methodologies, &case studies with real-life examples. 4. To collaborate and contribute to Open-Source Projects	
Unit	**Content**	**No of Hours 60**
I	**Introduction to Open-Source Software** ● Open Source, Free Software, Free Software vs. Open-Source Software, Public Domain Software, FOSS does not mean no cost. History: BSD, The Free Software Foundation, and the GNU Project. **Methodologies** ● Open-Source History, Initiatives, Principles, and methodologies. Philosophy: Software Freedom, Open Source Development Model Licenses and Patents: What Is A License, Important FOSS Licenses (Apache, BSD, GPL, LGPL), copyrights and copy lefts, Patents Economics of FOSS: Zero Marginal Cost, Income-generation opportunities, Problems with traditional commercial software, Internationalization.	**15**
II	**Social Impact** ● Open source vs. closed source, Open-source government, Open-source ethics. Social and financial impacts of open-source technology, Shared software, Shared source, and Open Source in Government. **Case studies** ● Example Projects: Apache web server, GNU/Linux, Android, Mozilla (Firefox), Wikipedia, Drupal, WordPress, GCC, GDB, GitHub, Open Office. Study: Understanding the developmental models, licensing, mode of funding, and commercial/non-commercial use. Opensource Hardware, Open-Source Design, Open-Source Teaching. Open-source media.	**15**

III	**Collaboration, Community, and Communication Contributing to Open-Source Projects** ● Introduction to Git Hub, interacting with the community on Git Hub, Communication and etiquette, testing open-source code, reporting issues, and contributing code. ● Introduction to Wikipedia, contributing to Wikipedia, or contributing to any prominent open-source project of the student's choice. ● Starting and Maintaining own Open-Source Project	**15**
IV	**Understanding Open-Source Ecosystem** ● Open-Source Operating Systems: GNU/Linux, Android, Free BSD, Open Solaris. Open-source hardware, Virtualization Technologies, Containerization Technologies: Docker, Development tools, IDEs, debuggers, Programming languages, LAMP, Open Source database technologies	**15**
Pedagogy:	1. Course delivery pattern, evaluation scheme, and prerequisite shall be discussed at the beginning. 2. Lectures preferably to be conducted with the aid of a multi-media projector, blackboard, group activities, charts, cases, etc. 3. One internal written exam would be conducted as a part of the internal theory evaluation. 4. One assignment based on the course content may be given to the students to evaluate how the learning of objectives was achieved.	
References:	**Main Reading:** 1. Fogel, K. (2009). *The Open Source Way: Openness and Collaboration Principles for Life*. O'Reilly Media. 2. Fogel, K. (2005). Producing Open Source Software: How to Run a Successful Free Software Project. O'Reilly Media. 3. Hassan, N. A. (2018). Open Source Intelligence Methods and Tools: A Practical Guide to Online Intelligence. Apress. 4. Raymond, E. S. (1999). The Cathedral & the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary. O'Reilly Media. **Additional Reading:** 1. Das, S. (2017).UNIX: Concepts and Applications. Tata McGraw Hill Education. 2. DiBona, C., Cooper, D., & Stone, M. (Eds.). (2005). Open Sources 2.0: The Continuing Evolution. O'Reilly Media. 3. Helmke, M.,Joseph ,E.K., Rey, J.A., Ballew, P., & Hill, B.M.(2014).The Official Ubuntu Book. Prentice Hall. 4. Whitehurst, J. (2015). The Open Organization: Igniting Passion and Performance. Harvard Business Review Press.	

Learning Outcomes:	**On completion of the course, students will be able to:** **CO1.**Remember the significance of Open-Source software practices and guidelines. **CO2.** Understand the Open-Source ecosystem, its use, impact, and importance. **CO3.** Apply Open-Source methodologies, and case studies with real-life examples. **CO4.**Collaborate and contribute to Open-Source Projects

Name of the Programme: Bachelor of Computer Applications
Course Code: CSA-131
Title of the Course: E-Commerce
Number of Credits: 3T
Effective from AY: 2024-25

Prerequisites for the Course:	None	
Course Objectives:	1. To give a fundamental understanding of e-commerce and online marketing 2. To instill ideas of Search Engine Optimization and Marketing, Applications of e-commerce and digital payments 3. To identify, define and differentiate the e-commerce models and risks of electronic commerce.	
Units	**Content**	**No. of Hours 45**
I	**Introduction to Electronic Commerce**: Meaning, Nature, and scope of e-commerce, History of e-commerce, Business applications of e-commerce, E-Commerce Models(B2B, B2C, C2C, B2G), Advantages and Disadvantages of e-commerce, Applications of M-Commerce. **E-Commerce Web-sites**: Websites as a marketplace, Role of the website in B2C e-commerce, Website design principles, Alternative methods of customer communication such as e-mail, Email etiquette, and e-mail security. **Online Marketing:** Online marketing and advertising, Push and pull approaches, Web counters, Web advertisements, Content marketing, Need of Digital Marketing for an e-commerce Business.	**15**
II	**Search Engine Optimization:** Search Engine Optimization (SEO), Search Engine Marketing (SEM), Social Media Marketing (SMM), Web Analytics. **Applications of E-commerce:** Applications of e-commerce to Supply chain management Applications of e-commerce to Customer Relationship Management, Product and service digitization, Remote servicing. **Electronic Payment System**: Types of payment systems, credit cards, debit cards, mobile, etc., Electronic Fund Transfer (EFT), Operational credit and legal risk of e-payment, and Risk management options for e-payment systems.	**15**

III	**Business to Consumer E-Commerce:** Cataloguing, Order planning and order generation, Cost estimation and pricing, Order receipt and accounting, Order selection and prioritization, Order scheduling, Order fulfilling, Order delivery, Order billing, Post sales service. **Business-to-Business E-Commerce:** Need and Models of B2B e-commerce, Using public and private computer networks for B2B trading; EDI and paperless trading, Characteristic features of EDI service arrangement, EDI architecture, and standards. **Security Issues in E-Commerce**: Risks of e-commerce, Types and sources of threats; Security tools, Risk management approaches.	**15**
Pedagogy:	1. PowerPoint, Tutorials, Hybrid learning.	
References/ Readings:	**Main Reading:** 1. Kalakota, Ravi, Andrew Whinston(2015). Frontiers of Electronic Commerce. Pearson Education. 2. P.T.Joseph(2015).E-Commerce: An Indian Perspective Paperback. PHI Learning. 3. V.Rajaraman(2015). Essentials of E-Commerce Technology. PHI Learning. **Additional Reading:** 1. C.S.V.Murthy (2015). *E-Commerce - Concepts, Models and Strategies.* Himalaya Publishing House.	
Course Outcomes:	**At the end of the course, students will be able to:** **CO1.**Understand the foundation of e-commerce, e-commerce websites and Online Marketing and Security Issues **CO2.**Explain the importance of Search Engine Optimization, Applications of E-commerce and Electronic Payment Systems. **CO3**.Compare B2B and B2C e-commerce models.	

Name of the Programme: Bachelor of Computer Applications

Course Code: CSA-132

Title of the Course: Green Computing

Number of Credits: 3T

Effective from AY: 2024-25

Pre-requisites for the Course:	None	
Course Objectives:	1. To remember the fundamentals of Green Computing and Green IT 2. To understand Green Hardware/Software and green Data Centers. 3. To devise a Green IT Strategy for an organization. 4. To implement Green IT initiatives.	
Units	**Content**	**No of hours 45**
I	**Trends and Reasons to Go Green** ● Overview and Issues ● Current Initiatives and Standards ● Consumption Issues-Minimizing Power Usage, Cooling **Introduction to Green IT** ● Green IT ● Holistic Approach to Greening IT ● Awareness to Implementation ● Green IT Trends ● Green Engineering **Greening by IT** ● Using RFID for Environmental ● Sustainability ● Smart Grids ● Smart Buildings and Homes ● Green Supply Chain and Logistics ● Enterprise-Wide Environmental Sustainability **Green Hardware and Software** ● **GreenHardware** ● Introduction ● Life Cycle of a Device or Hardware ● Reuse, Recycle, and Dispose ● **Green Software** ● Introduction ● Energy-Saving Software Techniques	**15**
II	**Green Data Centres and Storage Green Data Centres** ● Data Centre IT Infrastructure ● Data Centre Facility Infrastructure: Implications	**15**

	for energy efficiency ● IT Infrastructure Management ● Green Data Centre Metrics **Green Data Storage** ● Introduction ● Storage Media Power Characteristics ● Energy Management Techniques for Hard Disks ● System-Level Energy Management **Green Networks and Communications** ● Introduction ● Objectives of Green Network Protocols ● Green Network Protocols and Standards **Enterprise Green IT Strategy** ● Introduction ● Approaching Green IT strategies ● Business Drivers of Green IT Strategy ● Business Dimensions for Green IT Transformation ● Organizational Considerations in a Green IT Strategy ● Steps in Developing a Green IT Strategy ● Metrics and Measurements in Green Strategies ● Organizational and Enterprise Greening ● Greening the Enterprise: IT Usage and Hardware	
III	**Managing and Regulating Green IT** **Managing Green IT** ● Introduction and Approaches to Green ● Strategizing Green Initiatives ● Implementation of Green IT ● Information Assurance ● Communication and Social Media **Regulating Green IT** ● Introduction ● The Regulatory Environment and IT Manufacturers ● Non-regulatory Government Initiatives ● Industry Associations and Standards Bodies ● Green Building Standards ● Green Data Centres ● Social Movements and Greenpeace **The Future of Green IT** ● Green Computing and the Future ● Mega trends for Green Computing ● Tele-presence Instead of Travel ● Tele-commuting instead of Commuting	**15**

Pedagogy:	**Suggested strategies for use to accelerate the attainment of the various course outcomes.** 1. The lecture method need not be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. You may use a. Video/Animation to explain various concepts. b. Collaborative, Peer, Flipped Learning, etc. 2. Ask at least three HOT(Higher-Order Thinking)questions in the class, which promotes critical thinking. 3. Adopt Problem-Based Learning (PBL), which fosters students' Analytical skills, and develops design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it. 4. Introduce Topics in manifold representations. 5. Show the different ways to solve the same problem and encourage the students to come up with creative ways to solve them. 6. Discuss how every concept can be applied to the real world and when that's possible, it helps improve the students' understanding 7. To promote self-learning give at least one assignment (equivalent to 50% assignment weightage) where they can complete one MOOCs (certificate or equivalent) course out of lecture hour. Test their understanding through quizzes or presentations.
References/ Readings:	**Main Reading:** 1. San Murugesan, G.R.Gangadharan(2013).*Harnessing Green IT: Principles and Practices.* Wiley. 2. Toby Velte, Anthony Velte (2008). *Green IT: Reduce Your Information System's Environmental Impact While Adding to the Bottom Line*. McGraw Hill Education. **Additional Reading:** 1. BudE. Smith(2013). Green Computing- Tools and Techniques for saving energy, money and resources. Auerbach Publications. 2. MarkG. O'Neill(2011).Green IT for Sustainable Business Practice. BCS, The Chartered Institute for IT. 3. Mike Ebbers, Alvin Galea(2008). The Green Data Center: Steps for the Journey. International Business Machines Corporation 2008.
Course Outcomes:	On completion of the course, students will be able to: **CO1:** Recall the fundamental concepts of Green Computing and Green IT **CO2:** UnderstandfundamentalsofGreenComputingandGreenITanditsregulation. **CO3:** Apply Green IT Strategies for an organization. **CO4:**Analyze Green IT/Computing regulation and strategies.

Name of the Programme: Bachelor of Computer Applications

Course Code: CSA-141

Title of the Course: Office Automation and PC Troubleshooting

Number of Credits: 3 (1T + 2P)

EffectivefromAY:2024-25

Pre-requisites for the course:	Nil	
Course Objectives:	● To understand the basics of office automation software and its applications. ● To develop proficiency in using wordprocessing, spreadsheet, and presentation software. ● To diagnoseandtroubleshootcommonPCissuesandoptimizingtheperformance of a PC.	
Units	**Content**	**No of hours 75 (15T+60P)**
I	**Introduction to Office Automation** Understanding office automation software and its applications, Types of office automation software, Microsoft Office Suite, Google Workspace	**15**
	Word Processing Introduction to Microsoft Word, creating and formatting documents, working with templates, Mail merge and labels, Collaboration tools	
	Spreadsheets Introduction to Microsoft Excel, creating and formatting spreadsheets, working with formulas and functions, Charts and graphs, Collaboration tools	
	Presentation Software Introduction to Microsoft PowerPoint, creating and formatting presentations, working with images, videos, and animations, Collaboration tools	
	Email management & Internet and Web Browsers Introduction to Email, setting up and configuring email accounts, composing and sending emails, Managing Email Accounts Introduction to the Internet, Web browsers, searching the Internet, configuring web browser settings **PC Troubleshooting** Hardware Troubleshooting: Basic hardware components of a PC, Common hardware issues and their solutions, maintenance, and optimization of hardware	

	Software Troubleshooting: Common software issues and their solutions, Malware and virus removal, System recovery and backups, Network Troubleshooting	
II	**Practical:** list of suggested practical's	**60**
Week 1&2	1. Study of Google Workspace and its collaboration tools ● Create a Google form to build a questionnaire and collect responses. ● Use the tool to take surveys and generate reports on them.	**08**
Week 3&4	2. Experiments based on Word processing ● To create a document and apply basic formatting, creating a bulleted and numbered ist, applying headers and footers to the document, and page numbering. ● To study the creation of tables in MS Word and apply formatting to the table ● To insert pictures, shapes, and clipart in a document ● Prepare a bio-data in MS word using templates.	**08**
Week 5	3. Experiments based on Mail Merge ● Using Mail Merge to prepare letters, email messages, envelopes, and labels. ● Prepare ease-to-field trip notices using mail merge	**04**
Week 6to8	4. Practical on Spreadsheet ● Create a worksheet and perform basic formatting of cells, rows, and columns. ● Create a Student Mark Statement in MS Excel and calculate total, average, and percentage using Auto sum. ● Apply conditional formatting to the mark statement. ● Working with an advanced formulae ● Presenting data with charts	**10**
Week 8 to10	5. Practical Presentation software ● Usage of text, images, and animation for presentation ● Adding slide transition, custom animation, and setup show. ● Creating graphs in presentation. ● Design an advertisement in MS PowerPoint	**10**
Week 11	6. Email Management ● Experiment to setup and configure the email account ● Composeandsendanemailtoatleast5email addresses ● To manage the Email Accounts	**08**
Week 12 &13	7. Practical Internet browsing, downloading files, knowing secure browsing.	**04**

Week14 &15	8. PC troubleshooting ● Understanding PC components and PC assembling, formatting, fragmentation and installation of Operating systems and configuration of different types of software. ● To install different hardware devices, configure printers ● Identifying issues with hardware devices and troubleshooting. ● Network setup of two or more PCs. ● To install an antivirus software and understand the working of the firewall	**08**
Pedagogy:	**Suggested strategies to use to accelerate the attainment of the various course outcomes**. 1. Lecture methods need not be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. You may use a. Video/Animation to explain various concepts. b. Collaborative, Peer, Flipped Learning etc. 2. Ask at least three HOT (Higher-order Thinking) questions in the class, which promotes creative thinking. 3. Adopt Problem Based Learning(PBL),which fosters students' Analytical skills, develop design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it. 4. Introduce Topics in manifold representations. 5. Show the different ways to solve the same problem and encourage the students to come up with creative ways to solve them. 6. Discuss how every concept can be applied to the real world and when that's possible, it helps improve the students' understanding 7. To promote self-learning, give at least one assignment where they can complete atleast one MOOCs(certificateor equivalent) course out of lecture hour. Test their understanding through quizzes or presentations. 8. Activity/ Practical Based Learning (Suggested Activities in Class) a. Real-world problem solving using group discussion. E.g., designing posters for road safety etc., b. Demonstration of solution to a problem through design. 9. Demonstration of simple projects and motivating the students to develop similar type of projects.	

References/ Readings:	1. Andrews,J. (2019).A+ Guide to IT Technical Support (MindTap Course List). Cengage Learning. 2. Shelly,G.B.,& Vermaat,M.E.(2017). Microsoft Office365 & Office2016: Introductory. Cengage Learning 3. Vermaat,M.E.(2022).Discovering Computer: Digital Technology, Data, and Devices. Course Technology Inc.
Course Outcomes:	**On completion of the course, students will be able to:** **CO1.** Understand the basics of office automation software **CO2.** Demonstrate proficiency in creating and formatting documents, spreadsheets, and presentation **CO3.** Analyze the basic software and hardware issues & troubleshoot them.

Name of the Programme: Bachelor of Computer Applications

Course Code: CSA-142

Title of the course: Python Programming

Number of Credits: 3 (1T +2P)

Effective from AY: 2024- 25

Prerequisite for the course:	None	
Course Objectives:	1. To understand Python programming concepts. 2. To acquire proficiency in utilizing Python library functions and data structures. 3. To gain fundamental understanding of object-oriented programming (OOPS) concepts in Python.	
Units	**Content**	**No of Hours 75 (15T+60P)**
I	**Introduction to Python** ● Python interpreter/shell, indentation; identifiers and keywords; literals, numbers, and strings; operators(arithmetic operator, relational operator, Boolean operator, assignment, operator, ternary operator and bitwise operator) and expressions. **Program Flow Control** ● Input and output statements, defining functions, control statements (conditional statements, loop control statements, break, continue and pass, exit function.), default arguments, errors, and exceptions. **List,Tuple and Dictionary** ● Lists creation, traversal, slicing and splitting operations, passing list to a function. Tuple and Dictionaries. **OOPS Concepts** ● Introduction to Classes, Objects and Methods, Standard Libraries, File handling through libraries.	**15**
II	**Practical Work -I** Using any suitable python IDE or Interpreter.	**Practical Hours(28)**

Week1	1. Write a Python program to find the area and perimeter of a circle. 2. Write a Python program to generate the Fibonacci series. 3. Write a Python program to compute the GCD of two numbers. 4. Write a Python program to generate the first prime numbers. 5. Write a Python program to find the sum of squares of n natural numbers.	**4**
Week2 & week3	6. Program palindrome or not 7. Write a Python program to store strings in a list and print them. 8. Write a Python program to find the length of a list, reverse it, copy it, and then clear it. 9. Write a Python program to print the squares of numbers from 1 to 10 using loop control. 10. Write a Python program to count the number of even and odd numbers from a series of numbers. Sample numbers: numbers=(1,2,3,4,5,6,7,8,9) Expected Output: Number of even numbers: 5 Number of odd numbers: 4	**6**
Week4 & week5	11. Write a Python program that prints all the numbers from 0 to 6 except 3 and 6 Note: Use the' continue' statement. Expected Output: 0 1 2 4 5 12. Print the following pattern 1 12 123 1234 12345 13. Display numbers from -10to-1 using for loop 14. Print the following pattern * * * * * * * * * * 15. Write a Python function to sum all the numbers in a list Sample List: (8, 2, 3, 0, 7) Expected Output: 20	**8**
Week6 & week7	16.Write a Python program to reverse a string Sample String: "1234abcd"	**10**

	Expected Output: "dcba4321" 17. Write a Python function to calculate the factorial of a number (a non-negative integer). The function accepts the number as an argument 18. Write a Python program to print the even numbers from a given list. Sample List:[1,2,3,4,5,6,7,8,9] Expected Result:[2,4, 6,8] 19. Write a Python program to calculate the length of a string 20. Write a Python program to get a string from a given string where all occurrences of its first char have been changed to '$', except the first char itself.	
III	**Practical Work -II**	**Practical Hours (32)**
Week8 & week9	21. Write a Python program to change a given string to a new string where the first and last chars have been exchanged. 22. Write a Python program to count the occurrences of each word in a given sentence 23. Write a program to find the first and the last occurrence of the letter 'E' and character',' in "NEP IMPLEMENTATION, FOR BCA ". 24. Write a program to check if the word 'open' is present in the "This is open-source software". 25. Write a program to check if the letter 'e' is present in the word 'Welcome'.	**10**
Week10 & week11	26. Write a program in Python to delete first and last elements from a list 27. Write a Python program to check a list is empty or not 28. Write a Python program to remove duplicates from a list 29. Write a Python program to find the second smallest number in a list 30. Write a Python program to find common items from two lists 31. Let list=['a','b','c','d','e','f']. Find a)list[1:3]b)t[:4]c)t[3:] 32. Write a Python program to create a tuple with different data types.	**6**
Week12 & week13	33. Write a Python program to unpack a tuple in several variables 34. Write a Python program to read an entire text file 35. Write a Python program to append text to a file and display the text 36. Write a Python program to count the number of lines in a text file 37. Write a Python program to write a list to a file 38. Write a Python program to extract characters from various text files and puts them into a list	**6**

Week14 & week15	39. Write a function that reads a file file1 and copies only alternative lines to another file file2. Alternative lines copied should be the odd-numbered lines. 40. Write a function that reads a file file 1 and displays the number of words and the number of vowels in the file. 41. Consider a showroom of electronic products, where there are various salesmen. Each salesman is given a commission of 5%, depending on the sales made per month. In case the sale done is less than 50000, then the salesman is not given any commission. Write a function to calculate total sales of a salesman in a month, commission and remarks for the salesman. Sales done by each salesman per week is to be provided as input. Assign remarks according to the following criteria: Excellent:Sales>=80000 Good:Sales>=60000and<80000 Average:Sales>=40000and<60000 Work Hard: Sales < 40000	**10**
Pedagogy:	**Suggested strategies to use to accelerate the attainment of the various course outcomes:** 1. Lecture methods need not be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. You may use a. Video/Animation to explain various concepts. b. Collaborative, Peer, Flipped Learning etc. 2. Ask at least three HOT (Higher-order Thinking) questions in the class, which promotes critical thinking. 3. Adopt Problem-Based Learning (PBL), which fosters students' Analytical skills, and develops design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it. 4. Introduce Topics in manifold representations. 5. Show the different ways to solve the same problem and encourage the students to come up with their own creative ways to solve them. 6. Discuss how every concept can be applied to the real-world-and when that's possible, it helps improve the students' understanding To promote self-learning give at least one assignment where they can complete at least one MOOCs (certificate or equivalent) course out of lecture hour. Test their understanding through quizzes or presentations. 8. One internal practical exam will be conducted as a part of internal evaluation. 9. Practical shall be performed in the laboratory as indicated in the syllabus. 7. A Hand written Hard Copy or e-journal shall be maintained clearly mentioning the name of the experiment and other required information.	

References:	**Main Reading:** 1. Balagurusamy, E. (2017). *Introduction to Problem Solving with Python*. McGraw Hill Education India Private Limited. 2. Nageshwara Rao, R. (2018). *Core Python Programming*. Dreamtech Press. 3. Sedgewick, R., Wayne, K., & Dondero, R. (2016). *Introduction to Programming in Python: An Interdisciplinary Approach*. Pearson India Education Services Pvt. Ltd. 4. Yates, J. (2019). *Python Practical Python Programming For Beginners and Experts*. Packt Publishing. **Additional Books** 1. Dawson, M. (2020). *Python Programming for the Absolute Beginner*. No Starch Press. 2. Kumar, T. (2018). *Python Programming*. Wiley.. 3. Hoskins, A. (2017). *The Python Book: The ultimate guide to coding with Python*. Future Publishing Limited.. 4. Shovik, J. (2019). *Python All-In-One for Dummies*. For Dummies.
Course Outcomes	**On completion of the course, students will be able to:** **CO1.** Remember the basics of Python Programming **CO2.** Understand the concepts and constructs of Python programming. **CO3.** Apply Python library functions and data structures. **CO4.** Analyze the implementation of Python Programming

Name of the Programme: Bachelor of Computer Applications

Course Code: CSA-143

Title of the Course: Data Analytics using Spreadsheets

Number of Credits: 3P(1T +2P)

Effective from AY: 2024-25

Pre-requisites for the Course:	None	
Course Objectives:	1. Remember basic and advanced functions in spreadsheets. 2. Understand data analysis and data visualization with charts and pivot tables. 3. Implement dataanalysis tools and functions.	
Unit	**Content**	**No of Hours 75 (15T+60P)**
I	**Introduction to spreadsheets** ● Introduction to spreadsheets, understanding spreadsheet environment, cell addressing, cell references, absolute and relative cell references, named ranges, formatting using paste special, Data filters and sorting, worksheet and workbook protection	**15**
	Formulas and Functions, Advanced Functions ● Sum, Average, Min, Max, count, IF, nested IF, using IF with AND OR formulas, COUNTIF, SUMIF, AVERAGEIF formulas, TEXT functions ● Vlookup function, match function, index function, date and time functions, maths functions, financial functions	
	Data Analysis, Charts and Visualization ● Conditional formatting, What if analysis using data table, Goal seek, scenario manager, Linear regression ● Data storytelling tips, Introduction to charts, types of charts, uses and benefits, ● Understanding Pivot tables, Pivot table tips and tricks	
	DAX and Power Query ● Power query tips, Introduction to power pivot, Apply DAX in power pivot for analysis, introduction to types of joins in power query, full outerjoin and innerjoin in powerquery, left outer join and right outer join in power query, Left antijoin and right antijoin in power query	

	Dashboard reporting and Data Analysis tools ● Understanding how to create a dashboard in spreadsheets, a Sales Analytical Dashboard using Data Analysis Expressions (DAX) & Visualization, creating a simplified GANTT chart with AND function ● ANOVA, Correlation, Covariance, regression, sampling, t-test, z-test and histograms	
	PRACTICALS	**60 hours**
UNIT II	**List of suggested practicals**	**28**
Week1	Practical on introduction to a spreadsheet using simple tabular data and formatting using paste special, absolute, and relative cell references, calculating sum, average, min, max, count, and percentage.	4
Week2	Practical using IF, NESTEDIF, SUMFIF, AVERAGEIF, COUNTIF	4
Week 3 &4	Practical on advanced functions	8
Week5	Practical on conditional formatting, what-if analysis using Goal seek, scenario manager and linear regression	4
Week 6 &7	Practical on different types of charts and pivot table with suitable examples	8
UNIT III	**List of suggested practicals:**	**32**
Week8 to10	Practical on Powerquery, DAX, and different types of joins with suitable data.	12
Week 11 & 12	Creatingdashboardandganttchartinspreadsheetusingsuitableexamples	8
Week13to15	Excel data analysis Toolpak add-in covering ANOVA, Correlation, Covariance, Descriptive Statistical analysis, random number generation analysis, rank and percentile analysis, regression analysis, T-test, Z-test, Histogram	12
Pedagogy	Suggestedstrategiestousetoacceleratetheattainmentofthevariouscourse outcomes. 1. The lecture method need not be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. You may use a. Video/Animation to explain various concepts. b. Collaborative, Peer, Flipped Learning etc. 2. Ask at least three HOT (Higher-order Thinking) questions in the class, which promotes critical thinking. 3. Adopt Problem-Based Learning(PBL), which fosters students' Analytical skills, and develops design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it. 4. Introduce Topics in manifold representations. 5. Show the different ways to solve the same problem and encourage the students to come up with creative ways to solve them. 6. Discuss how every concept can be applied to the real world-and	

	when that's possible, it helps improve the students' understanding 7. To promote self-learning give at least one assignment where they can complete at least one MOOCs (certificate or equivalent) course out of lecture hour. Test their understanding through quizzes or presentations. 8. One assignment in the form of a mini-project collecting data and using analytic tools may be given to the students.	
References	**Main Reading:** 1. D.Whigham(2007).Business Data Analysis using Excel. New York: Oxford University Press. 2. Michael Alexander, Richard Kusleika, John Walkenbach. (2018). Excel 2019 Bible Paperback. Wiley 3. Stephen L. Nelson, Elizabeth C.Nelson,(January 2018).Microsoft Excel Data Analysis for Dummies. Wiley. 3ed	
Course Outcomes	**CO1.** Demonstrate basic and advanced functions in spreadsheet applications. **CO2.** Apply data analysis techniques and create visualizations using charts and pivot tables. **CO3.** Implement data analysis tools and functions for practical applications.	

Name of the Programme: Bachelor of Computer Applications
Course Code: CSA-144
Title of the Course:2DAnimation
Number of Credits: 3 (1T + 2P)
Effective from AY:2024-25

Pre-requisites fortheCourse:	None	
Course Objectives:	1. Familiarize with various approaches, methods and techniques of Animation 2. Study the Basics of Color Theory and Graphics. 3. Implement traditional & digital tools to produce still and moving images. 4. Develop expertise in life drawing and related techniques.	
Units	**Content**	**No. of hours 75 (45T+30P)**
I	**Introduction to Animation:** • Terms used in Animation, • Types of Animation-Cel(Celluloid) Animation, 2D Animation, 3D Animation, Motion Graphics, Stop Motion. • Animation Techniques used in 2D Animation: Hand-drawn animation, Cut-out animation, Model animation or Stop motion animation, Computer animation, or computer-generated imagery. • Equipment required for animation-Pentablet, Graphic tablet, Artist glove, Ergo stand, Flex arm. **Principles of Animation:** • Disney's twelve basic principles of animation- Squash and stretch, Anticipation, Staging, Straight ahead action and pose to pose, Follow through and overlapping action, Slow in and slow out, Arc, Secondary action, Timing, Exaggeration, Solid drawing, Appeal **Fundamentals of Drawing and Design** • Basic Shapes and Drawing techniques • Concepts of Visualization-Perspective drawing, Illustration, Shading, and Sketching techniques **Color Theory and Graphics** • Color fundamentals-primary colors, secondary colors, Tertiary Colors • Properties of color-Hue, Reflective Value, Tints, And Shades, Saturation, Color tone – Intensity	**15**

	• Additive Color System (RGB)-Subtractive Color System (CMYK). • Vector and Raster graphics **2D Animation tools processing** 2D animation software paradigms- scripting & Story boarding, Usage of tools for Digital Painting and vector drawings, developing a character and background creation.	
II	**Practical Work** Suggested list of Animation Tools: Pencil2d, Adobe Animate, Synfig studio, OpenToonz	**Practical Hours (60)**
Week1	Flipbook(on paper) Drawing simple flipbook with minimum 10 pages Flip Book (Digital) Create simple flipbook with minimum 10 frames	**(4)**
Week 2&3	Frame by frame animation Creating simple frame by frame animation for a short animation, demonstrating the concept of layering and onion skinning(maximum 20sec with color drawings and background.)	**(8)**
Week 4&5	Tween Create simple animations, using concepts of Grouping layers to create artwork, import images and apply tweening, Preview, and Render the animation in suitable format ● a classic Tweening: Create an E-card animation ● Motion tweening: Creating animation: Draw, Give Rotation effect, Time Loop demonstration ● Shape tweening: Demonstrate the animation	**(8)**
Week 6&7	Ball animation Drawing the ball with gradient color, Creating key frames for the animation sequence, Creating stretch and squash for the ball animation, Giving tween to the sequence of ball animation by connecting to path, duplicating waypoints,work with background image in the developed scene	**(8)**
Week 8&9	Character Animation Drawing simple character, Preparing the character for animation, dividing each body parts into symbol and creating motion	**(8)**
Week 10to12	Human/Animal walk cycle Drawing cycle sheet for an human/animal walk cycle, Creating four different types of walk cycle(jump,run,tip toe, crawl)	**(12)**

Week 13 to15	Mini project Prepare a storyboard and create short animation film using the concepts learnt in previous weeks	**(12)**
Pedagogy:	Suggested strategies for use to accelerate the attainment of the various course outcomes. 1. The lecture method need not be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. You may use a. Video/Animation to explain various concepts. b. Collaborative, Peer, Flipped Learning etc. 2. Ask at least three HOT (Higher-order Thinking) questions in the class, which promotes critical thinking. 3. Adopt Problem-Based Learning(PBL), which fosters students' Analytical skills, and develops design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it. 4. Introduce Topics in manifold representations. 5. Show the different ways to solve the same problem and encourage the students to come up with creative ways to solve them. 6. To promote self-learning, give at least one assignment where they can complete at least one MOOCs(certificate or equivalent)course out of lecture hour. Test their understanding through quizzes or presentations. 7. Mini-Project may be given as a part of the assessment	
References/ Readings:	**Main Reading:** 1. Chris Patmore(2003).The Complete Animation course. Barrons Educational Series. 2. Mary Murphy(2008). Beginner's Guide to Animation: Everything you need to know to get started. Watson-Guptill 3. Richard Williams (2012). The Animator's Survival Kit: A Manual of Methods, Principles, and Formulas for Classical, Computer, Games, Stop Motion, and Internet Animators. Farrar, Straus and Giroux. 4. Tony White(1988).The Animator's Workbook. Watson-Guptill	
Course Outcomes:	**On completion of the course, students will be able to:** **CO1.** Remember terminologies and aspects of computer animation. **CO2.** Apply the different principles of animation to produce still and moving images. **CO3.** Demonstrate and develop 2D animations using different tools. **CO4.** Integrate the concepts of drawing and color theory in animation.	

Name of the Programme: Bachelor of Computer Applications
Course Code: CSA-161
Title of the Course: PC Troubleshooting and Networking [Exit Internship Course]
Number of Credits: 4 (2T + 2P)
Effective from AY: 2024-25

Pre-requisites for the Course:	Knowledge of Personal Computer and Programming	
Course Objectives:	1. To understand the PC troubleshooting techniques. 2. To learn the basic concepts of networking. 3. To apply the PC troubleshooting techniques and networking concepts. 4. To analyze the cases of existing network setup and apply it.	
Units	**Content**	**No. of Hours 90 (30T + 60P)**
I	**PC Troubleshooting** 1. Hardware overview - CPU, RAM, Motherboard, storage devices, etc. 2. Peripherals overview - Monitors, Keyboards, Mouse, Printers, etc. 3. OS overview - OS environments: Windows and Unix / Linux, basic operations and navigation 4. Troubleshooting Fundamentals a. Identifying common PC issues: slow performance, hardware failures, software glitches, etc. b. Introduction to troubleshooting methodologies: isolation, testing, observation 5. Software Troubleshooting a. Diagnostic tools: Task Manager, Event Viewer, Resource Monitor, etc. b. Software installation and removal c. Managing updates and patches d. Web Browser Management e. Firewall & Anti-Virus 6. Hardware Troubleshooting a) Identifying hardware issues: RAM failures, hard drive errors, overheating, printers etc. b) Basic hardware maintenance: cleaning, replacing components c) Introduction to BIOS/UEFI settings	**10**
II	**Networking** 1. Introduction to Networking Basics	**20**

a) Overview of computer networks and their importance
b) Introduction to networking terminology and concepts
c) Understanding the TCP/IP models

2. Setting Up a Home Network
   a) Setting up a basic network environment using consumer-grade routers and switches
   b) Configuring IP addresses, subnet masks, and default gateways
   c) Connecting devices to the network (e.g., computers, smartphones, printers)
3. Introduction to Network Protocols
   a) Hands-on experience with common networking protocols (e.g., TCP, UDP, IP)
   b) Using packet sniffing tools to analyze network traffic
   c) Understanding the purpose and structure of Ethernet frames and IP packets
4. Wireless Networking Basics
   a) Configuring and securing Wi-Fi networks
   b) Understanding different wireless encryption methods (WEP, WPA, WPA2)
   c) Troubleshooting common Wi-Fi connectivity issues
5. Network Services Configuration
   a) Setting up and configuring network services such as DHCP, DNS, and FTP
   b) Configuring port forwarding and NAT (Network Address Translation)
   c) Implementing basic firewall rules to control network traffic
6. LAN Design and Troubleshooting
   a) Designing and implementing a small local area network (LAN)
   b) Troubleshooting common LAN connectivity issues (e.g., cable faults, IP conflicts)
   c) Using network diagnostic tools (e.g., ping, traceroute) to identify and resolve network problems
7. Introduction to Network Security
   a) Basic network security principles and best practices
   b) Securing network devices with strong passwords and access controls
   c) Implementing basic security measures such as MAC filtering

	and disabling SSID broadcast 8. Network Monitoring and Management a) Introduction to network monitoring tools (e.g., Wireshark, Nagios) b) Monitoring network performance metrics (e.g., bandwidth utilization, packet loss) c) Performing basic network troubleshooting and maintenance tasks 9. Introduction to Virtualization and Cloud Computing a) Setting up virtual networks using virtualization platforms (e.g., VMware, VirtualBox) b) Understanding cloud networking concepts and services (e.g., AWS, Azure)	
III	Practical Activities - To be carried out along in sync with the concepts mentioned in Unit I & II respectively. **PC Troubleshooting** 1) Boot Failure • Identify common causes of boot failure, such as hardware issues, corrupted system files, or misconfigured BIOS settings. • Troubleshoot boot failure by checking hardware connections, performing hardware diagnostics, and accessing BIOS settings to verify boot order and configuration. 2) Blue Screen of Death (BSOD) • Understand common causes of BSOD errors, including driver issues, hardware failures, and software conflicts. • Troubleshoot BSOD errors by analyzing error codes, checking device drivers, and performing memory and disk diagnostics. 3) Slow Performance • Identify factors contributing to slow PC performance, such as insufficient RAM, high CPU usage, or disk fragmentation. • Troubleshoot slow performance by checking resource usage in Task Manager, disabling unnecessary startup programs, and optimizing disk performance with disk cleanup and defragmentation. 4) Internet Connectivity Issues • Troubleshoot network connectivity issues by checking physical connections, verifying network settings, and testing connectivity with other devices.	**40**

	• Use command-line tools like ipconfig and ping to diagnose network problems and resolve issues with DNS resolution or IP address conflicts. 5) Hardware Malfunctions • Identify common hardware malfunctions such as overheating, noisy fans, or malfunctioning peripherals (e.g., keyboard, mouse). • Troubleshoot hardware issues by checking for loose connections, cleaning dust buildup, and replacing faulty components if necessary. 6) Software Errors • Troubleshoot software errors such as application crashes, freezes, or errors messages. • Use Event Viewer to analyze error logs, update software applications and drivers, and perform malware scans to detect and remove viruses or malware. 7) Peripheral Device Issues • Troubleshoot issues with peripheral devices such as printers, scanners, or external drives. • Check device connections, update drivers, and verify compatibility with the operating system. 8) Data Backup and Recovery • Develop a backup strategy to protect important data from loss due to hardware failure, software errors, or accidental deletion. • Practice data recovery techniques using backup software, file recovery tools, and cloud storage services. 9) System Maintenance • Perform routine system maintenance tasks to optimize PC performance and prevent issues. • Schedule regular updates for the operating system, antivirus software, and device drivers, and perform disk cleanup and defragmentation to maintain disk health. **Basic Networking** 10) Setting Up a Home Network • Configure a home router: Set up a router with DHCP enabled and configure wireless security. • Connect devices: Connect computers, smartphones, and	

printers to the network and ensure they can communicate with each other.

11) Introduction to Network Protocols
- Packet sniffing with Wireshark: Capture and analyze network traffic to understand protocols like TCP, UDP, and IP.
- Ethernet frame analysis: Use Wireshark to examine the structure of Ethernet frames and identify source and destination MAC addresses.

12) Wireless Networking Basics
- Wi-Fi setup and security: Configure a Wi-Fi network with WPA2 encryption and a strong passphrase. Test connectivity with various devices.
- Troubleshoot Wi-Fi issues: Troubleshoot common Wi-Fi problems such as signal interference or connectivity issues.

13) Network Services Configuration
- DHCP setup: Configure a DHCP server on a router or server and verify that clients receive IP addresses dynamically.
- DNS configuration: Set up a DNS server and configure DNS resolution for local and external domain names.

14) LAN Design and Troubleshooting
- LAN setup: Design and implement a small LAN with multiple devices connected through switches.
- Troubleshooting scenarios: Simulate LAN connectivity issues such as cable faults, misconfigured IP addresses, or DNS resolution problems.

15) Introduction to Network Security
- Password policies: Implement strong password policies on network devices and user accounts.
- Firewall setup: Configure basic firewall rules on a router or firewall appliance to control inbound and outbound traffic.

16) Network Monitoring and Management
- Bandwidth monitoring: Use network monitoring tools to measure bandwidth utilization and identify bandwidth-intensive applications.
- Network troubleshooting: Troubleshoot network issues using diagnostic tools like ping, traceroute, and netstat.

IV	**Case Studies** 1. Study the performance of any PC of the College lab, analyze and improve its performance. 2. Analyze any real-world existing networking scenario and case studies, like existing networking of your college labs. **Mini - Project** *Scenario:* You have been hired as a network administrator for a small business with approximately 15 employees. The company operates in a single office location and requires a reliable and secure network infrastructure to support its day-to-day operations. Develop a network design and implementation plan for a real-world scenario mentioned above, incorporating all aspects learned throughout the course. Simulate the above plan using a suitable free and open-source simulator like "GNS3" (Graphical Network Simulator-3) **OR** CISCO Packet Tracer ***Optional*** *- Prepare for industry-recognized certification (e.g., CompTIA Network+, Cisco CCNA) to enhance employability. Practice exams and hands-on labs to reinforce learning and prepare for certification exams*	**20**
Pedagogy:	Suggested strategies for use to accelerate the attainment of the various course outcomes. 1. A plan is to be developed by the student/s in consultation with the teacher incharge and to be approved. 2. One or methods mentioned below may be used for learning purposes. a. Intensive training / teaching b. Online or offline training (approved by the college or instructor) c. Approved MOOCS Courses d. Workshops - on-campus or off-campus e. Self-learning means & methods f. Enquiry-based learning 3. A work diary to be maintained where all the learning & work carried out to maintained and certified by the teacher incharges. 4. All deliverable & artifacts to be submitted in the college for evaluation and assessments.	

References/ Readings:	**Main Reading:** 1. Gookin, D. (2021). *Troubleshooting and Maintaining Your PC All-in-One For Dummies* (4th edition). For Dummies. 2. Kurose, J. F., & Ross, K. W. (2021). *Computer Networking: A Top-Down Approach* (8th ed.). Pearson Education Ltd. 3. Lowe, D. (2021). *Networking All-in-One For Dummies*(8th ed.). Wiley. 4. Meyers, M. (2019). *CompTIA A+ Certification All-in-One Exam Guide* (10th ed.). McGraw-Hill Education. **Additional Reading:** 1. Beasley, J. S., & Nilkaew, P. (2020). *Networking Essentials*. Pearson. 2. Donahue, G. A. (2015). *Network Warrior* (2nd ed.). O'Reilly Media. 3. Mueller, S. (2022). *Upgrading and Repairing PCs* (10th ed.). Que Publishing 4. Stevens, W. R. (1994). *TCP/IP Illustrated, Volume 1: The Protocols*(2nd ed.). Addison-Wesley.
Course Outcomes:	On completion of the course, student will be able to **CO1.** Understand the concepts and techniques of PC troubleshooting and basic networking. **CO2.** Apply troubleshooting and networking concepts & strategies and improve the performances. **CO3.** Analyze the performances of PCs and existing networks. **CO4.** Develop a network design for a small group of computers successfully.

Second Year - Semester III

Name of the Programme: Bachelor of Computer Applications

Course Code: CSA - 200

Title of the Course: Data Structures

Number of Credits: 4 (3T + 1P)

Effective from AY: 2024-25

Prerequisites for the Course:	Knowledge of C programming language	
Course Objectives:	1. To understand the concept of Algorithms. 2. To discuss linear and non-linear data structure 3. To implement data structure concepts	
Units	**Content**	**No of hours 75 (45T + 30P)**
I	**Algorithm Basics** – Algorithms and Data Structures, Pseudocode, Algorithm Features. **Data Structures:** Basic concepts, concepts of Linear and Non-Linear data structures, Array as data structure. Concept of ADT. **Searching and Sorting using array:** Searching (Linear & Binary) Sorting (Bubble Sort, Selection Sort & Insertion Sort).	**15**
II	**Stacks and Queues** (Using Arrays) Definition, Structure, Examples, Applications, and Basic Operations. **Linked Lists** (Linear and Doubly) Definition, Structure, Examples, Applications, and Basic Operations. Stacks and Queues using Linked List	**15**
III	**Trees:** Basic, Binary Tree and Binary Search Tree. **Graphs** – Graph Terminology, Representation, Traversals,	**15**
IV	**Practical Work** Using C programming language, data structure concepts to be covered in practicals are mentioned below.	**Practical Hours (30)**
Week 1 and 2	**Implement programs :** Array implementation - Creation, insertion, deletion	**04**

Week 3 to 5	**Searching and Sorting:** Searching (Linear & Binary) Sorting (Bubble Sort, Selection Sort & Insertion Sort).	**06**
Week 6 to 8	Stack & Queue data structure using arrays.	**06**
Week 9 to 12	Linked List data structure, Stack & Queue using linked list.	**08**
Week 13 to 15	Binary Search Tree.	**06**
Pedagogy:	**Suggested strategies for use to accelerate the attainment of the various course outcomes.** 1. The lecture method need not be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. You may use a. Video/Animation to explain various concepts. b. Collaborative, Peer, Flipped Learning, etc. 2. Ask at least three HOT (Higher-Order Thinking) questions in the class, which promotes critical thinking. 3. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, and develops design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it. 4. Introduce Topics in manifold representations. 5. Show the different ways to solve the same problem and encourage the students to come up with their creative ways to solve them. 6. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding 7. To promote self-learning, give at least one assignment (equivalent to 50% assignment weightage) where they can complete one MOOCs (certificate or equivalent) course out of lecture hour. Test their understanding through quizzes or presentations.	
References/ Readings:	**Main Reading :** 1. E. Balagurusamy.(2017). Data Structures using C. McGraw Hill Education. First Edition. 2. Yashavant Kanetkar(2019). Data Structures through C. BPB. Third Edition. **Additional Reading:** 1. Prabhakar Gupta (2011).Data Structures using C. Laxmi Publications.	
Course Outcomes:	**On completion of the course, students will be able to:** **CO1:** Remember the basic concepts of Data Structure. **CO2:** Understand the concept of linear and non-linear data structures. **CO3:** Analyze various data structures types and its implementation.	

Name of the Programme: Bachelor of Computer Applications

Course Code: CSA-201

Title of the Course: Database Management Systems

Number of Credits: 4 (3T + 1P)

Effective from AY: 2023-24

Prerequisites for the Course:	None	
Course Objectives:	1. To understand the basic concepts of database management systems and the process of database design using ERD, Schema design, and relational / table design. 2. To learn normalization concepts, basic relational operations and transaction processing and concurrency control concepts. 3. To learn To define and manipulate the relational databases in SQL using a suitable RDBMS system.	
Units	**Content**	**No of hours 75 (45T + 30P)**
I	**Introduction to DBMS** Data, Database, Database system, Database Management System, File oriented systems and its limitations; Three schema, levels of Data Abstraction, Database Architecture (Internal, Conceptual, View) and Data Independence Database Languages: Data Definition Language (DDL), Data Manipulation Language (DML), Data Control Language (DCL), Transaction Control Language (TCL) Database Users, DBMS functions, Advantages and Disadvantages Database Administration and Control: Functions Brief overview of Hierarchical, Network, Relational, Object-relational and Object-oriented data models **E-R Model** Data Modelling using Entity-Relationship Model ● ER Diagram Concepts & Terminologies ● Concept and Types of Entities, attributes, and relationship sets ● Key attribute, and domain of an attribute. ● Degree of a relationship set, cardinalities, ● Total and partial participation ● Generalization, specialization, aggregation ● integrity constraint, Referential integrity constraint and Key constraint.	**15**

	Activity: *Apply the concepts learned to design the ERD of at least 3 to 4 basic and different types of applications.*	
II	**Relational Data Model** Relational model concepts. Characteristics of relations; Types of keys-super key, candidate key, primary key, and foreign key Relational model constraints: Domain constraints, key constraints, primary and foreign key constraints, integrity constraints, and null values; Mapping Conceptual model into a normalized relational schema ***Activity:*** *Apply the concepts learned and convert the ERD designed in the previous Unit into a relational schema.* **Relational Operations** Basic/Fundamental Operations: Concept and Examples ● Select (σ) ● Project (∏) ● Union (∪) ● Set Difference (-) ● Cartesian product (X) ● Rename (ρ) Derived Operations: Concept and Examples ● Natural Join (⋈) ● Left, Right, Full outer join (⟕, ⟖, ⟗) ● Intersection (∩) ● Division (÷) Basic Concepts of Triggers, Views, and Procedures	**15**
III	**Normalization** Anomalies in a database Functional dependencies ● Armstrong's axioms/properties of functional dependencies ● Types of Functional dependencies Normalization Rules - 1NF,2NF, 3NF and Higher NF First Normal Form:1NF, Why convert to 1NF, Conversion to 1NF Second Normal Form: 2NF Functional Dependency and Fully Functional Dependency Why convert to 2NF, Conversion to 2NF Third Normal Form: 3NF Transitive Dependency why convert to 3NF, Conversion to 3NF Boyce- Codd NF, Convert to BCNF Normalization considerations: Good and bad decomposition **Activity:** *Apply the concepts learnt to show the step-wise normalization process of tables from 1NF till BCNF by outlining appropriate reasoning of at least 3 basic and different types of applications.*	**15**

	Transaction processing concepts ● Concept and state Diagram of Transactions ● ACID Properties ● Serializability: Conflict & View ● Schedule: Serial & Non- Serial ● Lock-based concurrency control ● Two-Phase Locking Protocol ● Transaction Recovery (log based)	
IV	**List of Practicals** To be done using any suitable RDBMS software like MYSQL	**Practical Hours (30)**
Week 1 & 2	1. Introduction and installation of DBMS Software 2. Database creation, alteration and deletion 3. Table creation, alteration, and Deletion 4. Identify and add appropriate data types to the fields 5. Add primary key and domain constraints to the table 6. Inserting data in the created tables 7. Data Manipulation language: Simple select query, Select with where clause	**04**
Week 3 to 7	8. Add Foreign key constraints to the table 9. Creating tables along with the primary key, foreign key, check, and other column constraints 10. To add rows in created tables, updating column(s) and performing deletions using truncate and delete should be done. 11. Group function and having clause 12. Operators 13. Aggregate Functions 14. Set operations 15. Sorting data	**10**
Week 8 to 10	16. Write SQL statements to perform operations using sub-queries for the following: ● Returning single-row ● Returning multiple rows ● Returning more than one column ● Correlated subquery	**06**
Week 11 to 13	17. Write SQL statements to implement the following types of SQL joins ● INNER JOIN ● LEFT OUTER JOIN ● RIGHT OUTER JOIN ● FULL OUTER JOIN	**06**

	Complex Queries using Joins, Aggregate Function and Correlated subqueries using set sub-queries & exist clause. 18. Write an SQL statement to show how VIEW can be created, altered, and dropped.	
Week 14 & 15	19. Demonstration and understanding on the following a. SQL statements to create simple triggers & stored procedures b. SQL statements to start a transaction, commit, rollback and define various save points in the queries. c. SQL statements to lock tables in read or write mode and also to perform unlock on the tables. d. SQL statements to assign and revoke privileges to/from users and user roles.	**04**
Pedagogy:	**Suggested strategies for use to accelerate the attainment of the various course outcomes.** 1. The lecture method need not be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. You may use a) Video/Animation to explain various concepts. b) Collaborative, Peer, Flipped Learning etc. 2. Ask at least three HOT (Higher-Order Thinking) questions in the class, which promotes critical thinking. 3. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, develop design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it. 4. Introduce Topics in manifold representations. 5. Show the different ways to solve the same problem and encourage the students to come up with their own creative ways to solve them. 6. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding 7. To promote self-learning, give at least one assignment where they can complete at least one MOOCs (certificate or equivalent) course out of lecture hour. 8. Test their understanding through quizzes or presentations.	
References/ Readings:	**Main Reading** 1. Elmasri, R., & Navathe, S. B. (2015). *Fundamentals of Database Systems* (7th ed.). Pearson Education. 2. Silberschatz, A., Korth, H., & Sudarshan, S. (2013). *Database System Concepts* (6th ed.). McGraw Hill. **Additional Reading** 1. An Introduction to Database systems, C.J. Date, A.Kannan, S.Swami Nadhan, Pearson, Eight Edition 2. Ramakrishnan, R., & Gehrke, J. (2002). *Database Management Systems* (6th ed.). McGraw Hill.	

Course Outcomes:	**On completion of the course, students will be able to:** **CO1.** Remember the basic concepts and terminologies of DBMS, ERD, Normalization, and Transaction Processing. **CO2.** Understand ER diagrams, Normalization, relational schema design, Relational Operations, Transaction Processing, and SQL concepts. **CO3.** Apply & discuss the concepts of ER Diagram, Relational Model and Normalization. **CO4.** Design relational database and formulate queries on the database and data using different SQL constructs mentioned in the syllabus.

Name of the Programme: Bachelor of Computer Applications

Course Code: CSA-211

Title of the Course: Reasoning Techniques

Number of Credits: 4 (3T+1 Tutorial)

Effective from AY: 2024-25

Prerequisite for the Course:	None	
Course Objectives:	1. To assess problem statement and make logical decisions 2. To interpret given data and derive conclusions 3. To understand Data interpretation and Data sufficiency 4. To solve problems using mathematical logic	
Units	**Content**	**No of Hours 60 (45T + 15 Tutorial)**
Tutorial Session Instructions	1. Tutorial lecture of 1 hour duration to be conducted each week. 2. Suggestive concepts/exercises needed to be discussed during tutorial hours every week are mentioned after Unit III. 3. These sessions may also be utilized for the doubt clearance	
I	**Statements & Arguments, Decision Making** ● Logic, Statements, Arguments, and Assumptions, Statements and Course of Action, Logical Venn Diagrams, Statements and Conclusions, Syllogism ● Seating Arrangement, Ranking & Time Sequence Test, Blood Relations, Direction Sense Test, Conditions & Grouping, Simple & Coded Inequality, Decision Making, Clocks and Calendar, Situation Reaction Test	**15**
II	**Data interpretation** ● Decision-making, Judgement, Problem-solving, Analogies, Analysis, Differences, Discrimination ● Arithmetic series, Similarities, Verbal & figure classification, Space visualization, Observation ● Simple Problems on Data interpretation and Data sufficiency	**15**
III	**Logic Building** ● Introduction, Statements, Logical Connectives and Compound Statements: Negation, Conjunction, Disjunction, Implication, Converse and Inverse, logical Equivalence, Tautologies: Contradiction, Contingency, Algebra of Propositions, Argument, Predicate and Quantifiers. ● Mathematical induction, deduction, proof by contradiction, program correctness.	**15**

Tutorial	List of suggested Tutorial Activities to be conducted in 15 weeks.	**15**
	● Solve Problems to be able to distinguish between Strong and Weak arguments. (Statement and Argument) ● Problems to assess a given statement and decide which of the given assumptions is implicit in the statement. (Statement and Assumptions) ● Problems to find out which of the conclusions definitely follow from a given statement. (Statement and Conclusions) ● Problem to analyse the statement and decide course of action. (Statement and Course of Action) ● Problem to analyse relation and decipher the relationship. (Blood Relations) ● Problems to ascertain the final direction or distance between two points (Direction Sense Test) ● Problems to analyse a given situation and choose the best response. (Situation Reaction Test) ● Problems to relate a given group of items and illustrate it diagrammatically. (Logical Venn Diagram) ● Problems on Data Interpretation, Data Sufficiency. (Data Interpretation) ● Problems based on fragmentation of a figure into sample parts, pattern rearrangement. (Data Interpretation) ● Problems on Induction, Deduction, Constructing and Understanding Truth Tables. (Mathematical Logic)	
Pedagogy:	1. Lecture methods need not be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. You may use a. Video/Animation to explain various concepts. b. Collaborative, Peer, Flipped Learning etc. 2. Ask at least three HOT (Higher-order Thinking) questions in the class, which promotes critical thinking. 3. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, develop design thinking skills such as the ability to design, evaluate, generalize, and analyse information rather than simply recall it. 4. Introduce Topics in manifold representations. 5. Show the different ways to solve the same problem and encourage the students to come up with their own creative ways to solve them. 6. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding 7. To promote self-learning, give atleast one assignment where they can complete at least one MOOCs (certificate or equivalent) course out of lecture hour. Test their understanding through quizzes or presentations.	
References/ Readings:	**Main Reading** 1. A.K. Gupta,. Logical and Analytical Reasoning. Ramesh Publishing House. 34th edition 2. Arun Sharma. How to Prepare for Logical Reasoning for the CAT. McGraw Hill Education (India) Private Ltd. 8th edition	

	Additional Reading 1. Peeyush Bhardwaj. Analytical & Logical Reasoning for CAT & Other Management Entrance Tests. Arihant Publications. 4th edition
Course Outcomes:	**On completion of the course, students will be able to:** **CO1** Remember basics rules of logic and reasoning **CO2** Understand various logic and reasoning concepts & techniques. **CO3** Apply the suitable reasoning techniques to solve real world problems **CO4** Analyze the obtained solution with suitable and relevant logic / reasoning.

Name of the Programme: Bachelor of Computer Applications

Course Code: CSA-212

Title of the Course: Techpreunership Development

Number of Credits: 4 (3T + 1 Tutorial)

Effective from AY: 2024-25

Prerequisites for the Course:	None	
Course Objectives:	1. To understand the basic concepts of Technopreneurship and experience the entrepreneurial process from the generation of creative ideas. 2. To understand the market needs or provide a solution to a key problem. 3. To discuss Intellectual Property strategy to protect inventions and innovations of new ventures. 4. To create and present a business plan for a technology idea.	
Units	**Content**	**No of hours 60 (45T+15 Tutorials)**
I	**a. Introduction to Techpreunership** • Concept of Technopreneurship • Technopreneur Vs Entrepreneur • Traits and characteristics of Technopreneur • Importance of Technopreneurship • Successful Global and Local Technopreneurs • Challenges in Technopreneurship **b. Idea, Innovation & Creativity** • Opportunity identification and idea generation – Case studies, Case scenarios • Basic concepts in Idea, Innovation & Creativity • Characteristics of an Innovative or a Creative Individual • Principles of Innovation • Types of innovation: Product, Process, and Business model • Importance of Creativity and Innovation • Factors that impact Innovation and Creativity	**15**
II	**Introduction to Intellectual Property** • Needs of Intellectual Property • Types of Intellectual Property • Procedure to register • Intellectual Property of a product • Importance of Intellectual Property in business • Copyright & trademarks regulations • Patents, trade secrets, contracts, non -disclosure and non -compete agreements	**15**

III	**Market Research & Customers Identification**
	• Customer Needs, Pain Points and Demographics
	• Market Research and Validation
	• The Decision-Making Process (Rational Decision Making)
	• Customer Profiling – STP (Segmentation, Targeting and Processes)
	
	Planning IT Business & Execution
	• Principles and concepts of business ownership
	• Types of business ownership
	• Factors that influence in starting a new entrepreneurial venture
	• Roadmap for research, development, and production
	• Develop IT Business Plan
	  ● Importance of a Business Plan
	  ● Criteria of a good Business Plan
	  ● Determine business plan outline
		15

IV	**Tutorial (case studies)** Tutorial lecture of 1 hour duration to be conducted each week.	**15 hours**
Week 1 & 2	**Case studies on successful Technopreneurs of Goa** • Analyze a specific case study(s) on successful technopreneurs, examining the key decisions, innovations, and challenges they faced. • Evaluate the impact of their entrepreneurial ventures on the technological landscape and the broader economy of the country.	**2**
Week 3 to 6	**Group Activities** ● Imagine you are a founder of a tech startup, and you're facing a common challenge in the industry. Your team is tasked with coming up with an innovative solution. Discuss and outline a step-by-step process you would follow to encourage creative thinking and generate unique ideas within your startup environment. ● Be sure to include specific methods, tools, or techniques you would employ, and explain how you would foster a culture of continuous innovation within your team. ● Additionally, consider potential obstacles and how you would address them in the pursuit of turning innovative ideas into successful implementations.	**4**

Week 7 & 8	**Report- How can emerging tech startups effectively utilize market research techniques/methods to gain a competitive edge and understand customer needs** ● Provide a detailed exploration of practical strategies, tools, and methodologies that tech startups can employ in their market research efforts to inform product development, target audience identification, and overall business strategy.	**2**
Week 9 & 10	**IPR Patent Filing Process Report:** ● Provide a detailed exploration of the practical aspects involved, including documentation requirements, legal considerations, potential challenges, and strategies for a successful patent filing	**2**
Week 11 & 12	**Case studies on India Government policies towards supporting entrepreneurship** ● Using a specific case study(s), analyze the effects of these policies on the development, challenges, and opportunities for entrepreneurs, highlighting key strategies and outcomes.	**2**
Week 13 to 15	**Business Plan Creation- Create a business plan for an IT company with the following key considerations.** ● Develop a comprehensive guide outlining the essential components, market analysis, financial projections, and strategic planning necessary to establish a robust business plan tailored to the specific needs and goals of the imaginary IT company of your choice."	**3**
Pedagogy	1. The lecture method need not be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. You may use a. Video/Animation to explain various concepts. b. Collaborative, Peer, Flipped Learning, etc. 2. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding. 3. Adopt Problem-Based Learning (PBL), which fosters students' Analytical skills, and develops design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it. 4. Show the different ways to solve the same problem and encourage the students to come up with their own creative ways to solve them. 5. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the student's understanding.	

References/ Readings:	**Main Reading** 1. Arya Kumar (2012). Entrepreneurship Creating And Leading An Entrepreneurial Organization. PEARSON INDIA. 2. Mathur, C. A. (2021). Taxmann's Entrepreneurship – Simple, Systematic Explanations along-with Comprehensive Coverage of the Concept & Theories). Taxmann Publications Private Limited. **Additional Reading** 1. Bruce R. Barringer, R.Duane Ireland (2020). Entrepreneurship: Successfully Launching New Ventures,Pearson Education. 2. Dr. Rakesh Kumar Singh, Arunabha Banerjee (2022). Intellectual Property Rights - A Textbook on IPR (Intellectual Property Rights). 3. Ramakrishna B & Anil Kumar H.S (2017). Fundamentals of Intellectual Property Rights : For Students, Industrialist and Patent Lawyers.
Course Outcomes:	**On completion of the course, students will be able to:** **CO1.** Understand the importance of idea, innovation and requirements in starting a business **CO2.** Explain the concepts of Intellectual Property Rights (IPR). **CO3.** Analyze the Opportunities of a potential scalable business through market research. **CO4.** Develop a business plan and implement their planning skills.

Name of the Programme: Bachelor of Computer Application
Course Code: CSA-213
Title of the Course: Computer Organization & Architecture Fundamentals
No. of credits: 4 (3T + 1P)
Effective from AY: 2024-25

Prerequisites for the Course:	None	
Course Objectives:	1. Conceptualize the basics of Computer Organizational and Architectural issues and classify the computers based on performance and machine instructions. 2. Learn various data transfer techniques and the I/O interfaces 3. Estimate and compare performances of various classes of memory 4. Understand the basics of ALU implementation, hardwired and micro-programmed control units, pipelining and parallel architectures	
Units	**Content**	**No of hours 75 (45T + 30P)**
I	**Data representation**: Data Type Representation, Number System, Signed number, fixed, floating point, character representation, Addition, Subtraction, Multiplication - Shift and Add, Booth's Algorithm, Division Pseudo-code: Definition and its attributes, constructs, and Examples **Introduction to Computer Architecture**: Introduction to Computer Architecture, Flynn's Classification of Computers, Performance Metrics (like Latency, throughput), Fundamental Blocks of Computer (like CPU, I/O subsystems, memory, control unit), computer function, interconnection structures, Bus interconnections	**15**
II	**Memory Hierarchy**: Hierarchical memory organization, Types of Memory-internal and external, Cache memory, Memory interleaving, **Peripheral devices:** Types of Peripheral Devices, I/O subsystem,programmed I/O, Interrupt-driven I/O, DMA, I/O channels and processors	**15**
III	**Instruction Set Architecture (ISA):** Introduction to Instruction Set, Types of ISA; RISC, CISC; Processor Organization, Registers organization, Instruction Execution Cycle, Instruction formats, Addressing Modes; Register Transfer Language (RTL), Assembly Language Programming, X86-Architecture, ARM Architecture	**15**
IV	**Practical Work** Writing assembly language programs in 8086 using MASM or compatible assembler either in Windows or Linux.	**Practical Hours (30)**

Week 1 & 2	1. Introduction to 8086 architecture and instruction set 2. Find the sum of 1 + 2 + 3 + ……+ n	**04**
Week 3 & 4	3. Display the multiplication table of a number 4. Store and retrieve numbers from memory	**04**
Week 5 & 6	5. Block Transfer 6. Block Transfer in reverse order	**04**
Week 7, 8 & 9	7. Sort the numbers stored in the memory(Any two methods) 8. Searching methods	**06**
Week 10 & 11	9. Masking of bits 10. Counting of number of bits	**04**
Week 12 & 13	11. Count the number of even or odd numbers from a given set of numbers 12. Check if the number is a palindrome	**04**
Week 14 & 15	13. Count the number of positive and negative numbers from a given set of numbers 14. Generate a series like 1,3,5,7,…….. up to n terms	**04**
Pedagogy:	**Suggested strategies for use to accelerate the attainment of the various course outcomes.** Lectures, Tutorials, Collaborative/peer learning, Hands-on assignments	
References/ Readings:	**Main Reading** 1. William Stallings. (9th Edition). Computer Organization and Architecture: Designing for performance. Prentice Hall of India. 2. John L. Hennessy & David Patterson. (5th Edition). Computer Architecture: A Quantitative Approach. Morgan Kaufmann.	
Course Outcomes:	On completion of the course, students will be able to: **CO1.** Recall the basic concepts & terminologies of Computer Organisation. **CO2.** Understand the concepts of data representation, computer & instruction set architecture, memory hierarchy, and peripheral devices. **CO3.** Apply the concepts of data representation, Assembly Language, and performance matrices in solving basic problems. **CO4.** Analyze multiplication & division algorithms at basic level and basic design issues in terms of speed, technology, cost, performance, CPU architecture.	

Name of the Programme: Bachelor of Computer Applications

Course Code: CSA 231

Title of the Course: Cyber Law and Ethics

Number of Credits: 3T

Effective from AY: 2024-25

Prerequisites for the Course:	None	
Course Objectives:	1. To understand the basic concepts of cyber law, cyber security, and the need for privacy protection and intellectual property protection. 2. To comprehend the importance of ethics for IT professionals and IT organizations.	
Units	**Content**	**No of hours 45**
I	**Overview of Ethics, Ethics for IT Workers and IT Users** Ethics, Ethics in the Business World; Corporate Social Responsibility; Fostering Corporate Social Responsibility and Good Business Ethics; Improving Business Ethics; Ethical Considerations in Decision Making; Ethics in Information Technology; Managing IT Worker Relationship; Encouraging Professionalism of IT Workers — Professional Codes of Ethics, Professional Organizations, Certifications and Licensing; Encouraging Ethical Use of IT Resources among Users. **Ethical Decision in Software Development and Ethics of IT Organizations:** Software Quality and its Importance; Strategies for Developing Quality Software; Use of Contingent Workers; H-IB Workers; Outsourcing; Whistle-Blowing.	**15**
II	**Cyberattacks, Cybersecurity, and Cyber Law:** Threat Landscape — Computer Incidents, Types of Exploits; CIA Security Triad; Confidentiality, Integrity, Availability, Implementing CIA at Organizational, Network, Application, and End-User Level; Response to Cyber Attack – Incident Notification Protection of Evidence and Activity Logs Incident Containment Eradication Incident Follow-Up Using an MSSP, and Computer Forensics; Cyber Law; Provision of Cyber, Overview of IT Act 2000, Code of conduct for computer professionals, Amendments and Limitations of IT Act.	**15**
III	**Privacy, Freedom of Expression, Intellectual Property and Organizational Ethics:** Privacy Protection and the Law – Information Privacy, Privacy Laws, Applications, and Court Rulings; Key Privacy and Anonymity Issues Consumer Profiling, Electronic Discovery,	**15**

	Workplace Monitoring, Surveillance; First Amendment Rights; Freedom Expressions: Key Issues; Social Networking Ethical Issues. **Intellectual Property:** Intellectual Property, Copyright; Patent; Trade Secrets; Intellectual Property Issues: Plagiarism, Reverse Engineering, Open Source Code, Competitive Intelligence, Trademark Infringement, and Cybersquatting.	
Pedagogy:	**Suggested strategies for use to accelerate the attainment of the various course outcomes.** 1. The lecture method need not be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. You may use a. Video/Animation to explain various concepts. b. Collaborative, Peer, Flipped Learning, etc. 2. Adopt Problem-Based Learning (PBL), which fosters students' Analytical skills such as the ability to evaluate, generalize, and analyze information rather than simply recall it. 3. Show the different ways to analyze cyber laws and crimes. 4. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding	
References/ Readings:	**Main Reading** 1. George W. Reynolds,(2012) Sixth Edition. Ethics in Information Technology. Course Technology, Cengage Learning 2. Herman T. Tavani, John Wiley and Sons, Fifth Edition, 2016. Ethics and Technology: Controversies, Questions, and Strategies for Ethical Computing. Wiley **Additional Reading** 1. Michael J. Quinn, Pearson, (2015) Eighth Edition. Ethics for Information Age. Pearson	
Course Outcomes:	**On completion of the course, students will be able to:** **CO1:** Understand the concepts of Cyber Law, Intellectual Property, and issues emerging in Cyberspace and the importance of Information Technology Act. **CO2:** Apply knowledge in implementing IT ethics for users and organizations	

Name of the Programme: Bachelor of Computer Applications

Course Code: CSA-232

Title of the Course: Digital Ecosystem

Number of Credits: 3T

Effective from AY: 2024-25

Prerequisites for the Course:	None	
Course Objectives:	1. To understand the fundamentals of the Digital Ecosystem. 2. To analyze digital workspace concepts and the design practices. 3. To comprehend the architecture and the future of the Digital Ecosystem.	
Units	**Content**	**No of hours 45**
I	**Introduction to Digital Ecosystem:** Introduction, key elements of a Digital Ecosystem, importance, Types of digital ecosystems, working, digital ecosystem mapping, Challenges in building and managing a Digital Ecosystem, Examples of successful digital ecosystems **Approaches to Digital Ecology:** Concept of Information Ecology, Information Ecology as a Research Model, Digital business ecosystem, Digital publicity platforms **Computing of Digital Ecosystems:** Multi-Agent Systems, Evolutionary Computing, Service-Oriented Architectures, Distributed Evolutionary Computing	**15**
II	**Architecture of Digital Ecosystem:** Trends and rise of Technological Ecosystem, Ecosystem Viewpoints **Digital Workspace Concepts:** Introduction, Human-Machine interface, Contextualization of objects, places and actions, Digital User Experience (DUX) and Customer Experience (CX), Evolution of software techniques, Data analytical software development and techniques, Digital workspaces **Design Practices in Digital Enterprise:** Introduction, Example of a digital business model using digital workspaces, Design practices in digital enterprise, Future of intelligent workspaces.	**15**

III	**Reference Architecture for Digital Ecosystem (RADE)** Components of a digital ecosystem, RADE, principles in different areas of architecture; Layers of RADE- environment, Context and niche, Interaction, Adaptation to goals, Species integration and User integration; Security principles in RADE. **Case Studies** Digital ecosystem for the environment, Digital health ecosystem, Facebook ecosystem, Google ecosystem, E-Governance **Future of Digital Ecosystem** Risks in the current environment, Building a digital ecosystem for Planet, overcoming the risks, Future aspects.	**15**
Pedagogy:	**Suggested strategies for use to accelerate the attainment of the various course outcomes.** 1. The lecture method need not be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. You may use a. Video/Animation to explain various concepts. b. Collaborative, Peer, Flipped Learning, etc. 2. Ask at least three HOT (Higher-Order Thinking) questions in the class, which promotes critical thinking. 3. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, and develops design thinking skills such as the ability to design, evaluate, generalize, & analyze information rather than simply recall it. 4. Introduce Topics in manifold representations. 5. Show the different ways to solve the same problem and encourage the students to come up with their own creative ways to solve them. 6. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding 7. To promote self-learning, give at least one assignment where they can complete one MOOCs (certificate or equivalent) course out of lecture hour. Test their understanding through quizzes or presentations.	
References/ Readings:	**Main Reading** 1. Alessandra Lazazzara, Francesca Ricciardi, Stefano Za. (2019) Exploring Digital Ecosystems: Organizational and Human Challenges. Springer International Publishing 2. Jaydip Sen. (2018) Digital Technologies in the Digital Enterprise, Internet of Things: Technology, Applications and Standardization. IntechOpen 3. Mark Skilton (2016) Building Digital Ecosystem Architectures: A Guide to Enterprise architecting. Springer	

	Additional Reading 1. Arnoud De Meyer, Peter J. Williamson, and Fiona H. Murray. (2020)Ecosystem Edge: Sustaining Competitiveness in the Face of Disruption. Stanford Business Books 2. Geoffrey G. Parker, Marshall W. Van Alstyne, and Sangeet Paul Choudary(2016) Platform Revolution: How Networked Markets Are Transforming the Economy—and How to Make Them Work for You. W. W. Norton & Company
Course Outcomes:	**On completion of the course, students will be able to:** **CO1.** Remember key elements, types and working of Digital Ecosystem **CO2.** Understand digital ecosystem fundamentals and computing concepts. **CO3.** Acquire the knowledge of digital workspace and design practices in a digital enterprise **CO4.** Analyze the architecture and the prospects of the digital ecosystem.

Name of the Programme: Bachelor of Computer Applications

Course Code: CSA-233

Title of the Course: Website Design

Number of Credits: 3 (2T+1P)

Effective from AY: 2024-25

Prerequisites for the Course:	None	
Course Objectives:	1. To understand the basic principles and syntax of HTML and CSS. 2. To Effectively address common styling challenges and achieve desired visual effects through skillful use of CSS techniques. 3. To apply CSS features to create dynamic and engaging user interactions that enhance web experiences that seamlessly adapt to diverse devices and screen sizes. 4. To design simple webpages using HTML and CSS.	
Units	**Content**	**No of hours 60 (30T + 30P)**
I	**Introduction to HTML** ● World Wide Web, URL, Domain, Text Editors used, Web Page and Website ● HTML Tags, Basic structure of an HTML document, Headings, Paragraphs, Line Breaks, Mark-up Tags ● Basic formatting tags, Hyperlinks, Images, and Multimedia, Marquee Elements ● Lists, Tables, Frames, Forms and controls (button, checkboxes, textboxes etc.), Audio and Video Tags	**10**
II	**Introduction to CSS** ● Creating Style Sheet, CSS Properties, inline and block elements ● CSS Selectors - Element Selector, ID Selector, Class Selector, Grouping Selectors, Universal Selector ● Text Properties - Letter-Spacing Property, Word-spacing Property, Text-align Property, Text-transform Property, Line-height Property, Text Decoration, and Font properties ● Table and List Properties **Advanced CSS Concepts** ● Box Model, Margins, Padding, Border, Color, Opacity ● Color Properties, Background Color, Layering Elements using Z-Index ● Animation using transitions ● Display - flexbox and grid	**20**

	● Absolute and Relative Positioning, Align, Pseudo class, Pseudo-element, Responsive design - Media Queries	
III	**List of experiments:**	**Practical Hours (30)**
Week 1	Create a simple HTML document with a title, heading, paragraph, list, and an image.	**02**
Week 2	Design a form with different types of input fields such as text, password, radio buttons, checkboxes, and a submit button.	**02**
Week 3	Style the HTML page created in Experiment 2 using CSS. Apply different font styles, sizes, and colors. Experiment with background colors and margins.	**02**
Week 4	Design a webpage with CSS focusing on text properties (letter-spacing, word-spacing, text-align, text-transform, line-height, text decoration, and font properties).	**02**
Week 5 & 6	Create an HTML document and apply CSS to style inline and block elements using various selectors (element, ID, class, grouping, universal). Experiment with color properties, background color, border color, opacity, margins, padding, and z-index.	**04**
Week 7 & 8	Implement basic animations using CSS transitions.	**04**
Week 9	Explore the use of Flexbox for layout design on a webpage.	**02**
Week 10	Create a webpage with a multi-column layout using CSS Grid. Experiment with grid properties to achieve different column structures and alignments.	**02**
Week 11	Experiment with absolute and relative positioning in CSS.	**02**
Week 12	Apply pseudo-classes and pseudo-elements to style specific states or parts of a webpage.	**02**
Week 13 to 15	Construct a webpage that adapts to different devices like desktops, tablets, and mobile phones based on screen sizes using media queries.	**06**
Pedagogy:	**Suggested strategies for use to accelerate the attainment of the various course outcomes.** 1. Lecture method need not be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. You may use a. Video/Animation to explain various concepts. b. Collaborative, Peer, Flipped Learning etc. 2. Ask at least three HOT (Higher-Order Thinking) questions in the class, which promotes critical thinking. 3. Adopt Problem-Based Learning (PBL), which fosters students' Analytical skills, and develops design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it. 4. Introduce Topics in manifold representations. 5. Show the different ways to solve the same problem and encourage the students to come up with their own creative ways to solve them.	

	6. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding 7. To promote self-learning give at least one assignment (equivalent to 50% assignment weightage) where they can complete at least one MOOCs (certificate or equivalent) course out of lecture hour. Test their understanding through quizzes or presentations.
References:	**Main Reading** 1. Jonathan Fielding (2014). Beginning Responsive Web Design with HTML5 and CSS3; Apress. 2. Robin Nixon (2022). HTML5 and CSS3 Masterclass. BPB Publications **Additional Reading** 1. Ed Tittel, Chris Minnick (2013). Beginning HTML5 and CSS3 For Dummies, 1st Edition. For Dummies 2. Joe Attardi (2020) Modern CSS: Master the Key Concepts of CSS for Modern Web Development; Apress.
Course Outcomes:	**On completion of this course, students will be able to:** **CO1**. Remember the basic concepts of HTML and CSS. **CO2**. Understand and apply different HTML text formatting, images, hyperlinks and CSS selectors to web pages. **CO3**. Apply CSS for styling and layout, ensuring a visually appealing and responsive design. **CO4**. Design static webpages using Flexbox and grid layouts.

Name of the Programme: Bachelor of Computer Applications
Course Code: CSA-234
Title of the Course: Enterprise Resource Planning (ERP)
Number of Credits: 3(2T+1P)
Effective from AY: 2024-25

Prerequisites for the Course:	None	
Course Objectives:	1. To study the basic concepts, evolution of ERP and its application in organization. 2. To study the life cycle/ activities of ERP. 3. To study various technologies related to ERP. 4. To analyze market trends on the usage of ERP and develop a process driven thinking towards business processes.	
Unit	**Content**	**No of hours 60 (30 T+30 P)**
I	**Introduction to ERP** ● Evolution of ERP ● What is ERP? ● Reasons for the Growth of ERP ● Modules of ERP ● Advantages and Disadvantages of ERP **An Overview of Enterprise** ● An Overview of Enterprise ● Management Information System ● Business Processes Integration ● Need of ERP for Small Business ● Business Process Mapping for ERP Module Design ● Implementation of ERP and concerns involving implementation **ERP and Information System** ● ERP and Information System ● Business Process and Business Process Reengineering (BPR) ● Management Information System (MIS) ● Executive Information System (EIS) ● Decision support System (DSS) ● Supply Chain Management ● Customer Relationship Management	**15**

II	**ERP Implementation Lifecycle** ● Issues in Implementing ERP Packages ● Pre-evaluation Screening ● Package Evaluation ● Project Planning Phase, Gap Analysis, Reengineering, Configuration, Implementation, Team Training, Testing, Going Live, End-User Training, Post Implementation (Maintenance Mode). **Advance Technologies** ● E-Procurement ● E-Logistics ● Internet Auctions ● E-markets ● Electronic Business Process Optimization ● Business Objects in SCM ● E commerce ● Customer Relationship Management	**15**
III	**Practicals** The concepts learned in the units from I and II are required to be implemented practically. The use of open source software (ERPNext, Odoo, Dolibarr, Tryton etc.) could be used to demonstrate the working of different modules used in ERP.	**Practical 30 hours**
Week 1 to 3	● Study and analyse need for Business Process re-engineering ● Case studies on ERP and their Functionalities	**06**
Week 4 to 6	● Solving Case studies/scenarios using ERP	**06**
Week 7 to 9	● Analyse, use and review any Open Source ERP softwares	**06**
Week 10 to 15	● Analyse and use the Open Source ERP System with the following modules: • Sales and Distribution (SD) • Materials Management (MM) • Production Planning (PP) • Financial Accounting (FI) • Human Capital Management (HCM) • Business Warehouse (BW)	**12**

Pedagogy:	Suggested strategies for use to accelerate the attainment of the various course outcomes. 1. Lecture methods need not be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. 2. You may use a. Video/Animation to explain various concepts. b. Collaborative, Peer, Flipped Learning etc. 3. Ask at least three HOT (Higher-Order Thinking) questions in the class, which promotes critical thinking. 4. Use of Case studies to illustrate concepts of ERP 5. Introduce Topics in manifold representations. 6. Discuss how every concept can be applied to the real world
References/ Readings:	1. Alexis Leon, (3rd or later Edition). ERP Demystified. Tata Mc Graw Hill. 2. Christian N. Madu. (July 2005) ERP and Supply Chain Management. Chi Pub.
Course Outcomes:	**On completion of the course, the students will be able to:** **CO1.** Recall the basic concepts and issues of ERP systems. **CO2.** Understand the concepts, techniques and processes of ERP System and its implementation. **CO3.** Apply the basic concepts to design the ERP implementation strategies. **CO4.** Analyse the strategic options for ERP identification and adoption.

Name of the Programme: Bachelor of Computer Applications

Course Code: CSA-235

Title of the Course: LaTex

Number of Credits: 3(2T+1P)

Effective from AY: 2024-25

Pre-requisites for the Course:	None	
Course Objectives:	1. Familiarize students with the installation process and graphical user interface (GUI) of widely used typesetting software, particularly in the field of Mathematics. 2. Acquire proficiency in the application of mathematical formulae, drawing, and designing using LaTeX. 3. Recognize the significance of this software in publishing research articles, papers, project reports, and books, fostering comfort and confidence in its use.	
Units	**Content**	**HOURS 60 (30T + 30P)**
I	**Installation of LaTeX** i. Installation of Kile and MikeTeX ii. Class and packages iii. Latex programming and commands, sample packages iv. Error messages: Some sample errors, list of LaTeX error messages **Formatting of output document** i. Fonts, symbols, indenting, paragraphs, line spacing, word spacing, titles and subtitles ii. Document class, page style, parts of the documents, table of contents iii) Command names and arguments, environments, declarations iii. Theorem like declarations, comments within text	**15**
II	**Mathematical formulae** i. Mathematical environments, math mode, mathematical symbols ii. Graphic package, multivalued functions, drawing matrices iii. Tables, tables with captions iv. References to figures and tables in text **Drawing with LaTeX** i. Picture environments ii. Extended pictures, other drawing packages iii. Preparing book, project report in LaTeX.	**15**

III	Practical Work	Practical Hours (30)
Week 1 to 3	Introduction to LaTeX i) Installation of LaTeX, Kile and MikeTeX ii) Class and packages iii) Latex programming and commands, sample packages iv) Error messages : Some sample errors, list of LaTeX error messages	**06**
Week 4 to 7	Formatting of output document 1. Fonts, symbols, indenting, paragraphs, line spacing, word spacing, titles and subtitles 2. Document class, page style, parts of the documents, table of contents 3. Command names and arguments, environments, declarations 4. Theorem like declarations, comments within text	**08**
Week 8 to 11	Mathematical formulae 1. Mathematical environments, math mode, mathematical symbols 2. Graphic package, multivalued functions, drawing matrices 3. Tables, tables with captions 4. References to figures and tables in text	**08**
Week 12 to 15	Drawing with LaTeX 1. Picture environments 2. Extended pictures, other drawing packages 3. Preparing book, project report in LaTeX.	**08**
Pedagogy:	PowerPoint, Tutorials, Hybrid learning, Peer Learning	
References / Readings:	**Main Reading** 1. Kopka, H., & Daly, P. W. (Year). *Guide to LaTeX* (4th Edition). Addison-Wesley. 2. Kumar, S. S. (2019). *LATEX - A Beginner Guide to Professional Documentation*. Laxmi Publications Pvt Ltd. **Additional Reading** 1. Swaminathan Murugan. (2022). *Latex For Beginners*. (1st edition). Notion Press	
Course Outcomes:	**At the end of the course, students will be able to:** **CO1.** Successfully install the software and navigated its GUI, gaining a foundational understanding of its features. **CO2.** Understand the role of LaTeX in academic publishing, and utilize the software for the preparation of scholarly documents. **CO3.** Demonstrate the ability to effectively use LaTeX for typesetting mathematical content, creating accurate formulae, and incorporating drawings and designs within documents.	

Name of the Programme: Computer Applications

Course Code: CSA-236

Title of the Course: Multimedia Essentials

Number of Credits: 3(2T+1P)

Effective from AY: 2024-25

Prerequisites for the Course:	None	
Course Objectives:	1. To make the students aware of Color Models and Color harmony 2. Study basics of animation and to learn about 2D/3D animations 3. Develop creative social media ready videos with visual effects. 4. Develop and learn best practices for elements of design,audio and video editing.	
Units	**Content**	**No of hours 60 (30T+30P)**
I	Multimedia - Introduction, Uses of Multimedia, Social & Ethical considerations, Digital Representation.	**15**
	Color Theory - Color Basics, Color Systems, Color Wheel, Complementary Colors, After Images, Color Combinations, Color & Contrast, Proportion & Intensity, Shades, Tones & Tints.	
	Introduction to Computer Graphics: Difference between Raster and Vector Graphics, Raster graphics: resolution, image compression, file formats, manipulation; Vector graphics fundamentals, file formats, shapes, transforms and filters	
	Text and Layout: character set, fonts & faces, using Text in Multimedia, Font Editing & Tools.	
II	Sound: Introduction, Digital Audio, MIDI Audio, Audio Codec & file formats, Making Digital Audio files.	**15**
	Animation: Principles of Animation, Types of Animation, Keyframe, Sprite, file formats.	
	Video: How Video Works and is Displayed, Aspect Ratio, Frame size, Frame Rate, Video Codec & File formats, Processing & Delivery.	
III	**Practical Work**	**Practical Hours (30)**
Week 1	1. Design a Brochure for given Product and details. Learn about different file formats	**2**
Week 2	2. Design a Brochure for given Product and details. Learn about different file formats	**2**
Week 3	3. Design a poster with given information and learn about image compression	**2**

Week 4 & 5	4. Edit the sound file and Learn about Effects and Filters of sound	**4**
Week 6 & 7	5. Record voice and learn about Audio Compression	**4**
Week 8 to 10	6. Learn Audio mixing and streaming of audio content	**6**
Week 11 to 13	7. Learn about Video editing. Prepare video with rough cut, Prepare video content with title and special effects.	**6**
Week 14 & 15	8. Record video content and learn about video compressions, Prepare Video content for vimeo / youtube. Note: -(Practical can be done using GIMP, Inkscape, Scribus, Photoshop, Illustrator, Flash, Blender, Audacity, Lightworks.)	**4**
Pedagogy:	1. Conventional Lecture method 2. Case based learning 3. Experiential Design Thinking 4. Formative and summative assessments 5. Live experimental projects	
References/ Readings:	**Main Reading:** 1. Chapman, N., & Chapman, J. (2004). *Digital Multimedia* (2nd ed.). Wiley. 2. Parekh, R. (2017). *Principles of Multimedia* (2nd ed.). McGraw Hill Education. 3. Tay, V. (2011). *Multimedia: Making it Work* (8th ed.). Tata McGraw-Hill.	
Course Outcomes:	**On completion of the course, students will be able to:** **CO1.** To remember the fundamentals and underlying theories of Multimedia. **CO2.** To understand animation and to design and develop 2D/3D animations **CO3.** To analyze the best practices for elements of design, audio and video editing. **CO4.** To create films, visual effects for the creative media.	

Name of the Programme: Bachelor of Computer Applications

Course Code: CSA-241

Title of the Course: Multimedia Applications

Number of Credits: 3 (1T + 2P)

Effective from AY: 2024-25

Pre-requisites for the Course:	None	
Course Objectives:	1. Introducing terminologies and technologies in multimedia. 2. Learning different types and forms of multimedia. 3. Learn storage and access mechanisms of each multimedia file type.	
Units	**Content**	**No of hours 75 (15+60)**
I	**Introduction to Multimedia & Graphic Design Fundamentals** • Definition and Characteristics of Multimedia • Evolution of Multimedia Technologies • Multimedia Elements: Text, Images, Audio, Video, Animation • Multimedia Hardware and Software • Principles of Graphic Design • Image Editing Techniques • Creating and Manipulating Vector Graphics **Audio ,Video Production and Animation Principles** ● Basics of Sound and Audio Editing ● Video Production Process ● Editing Techniques using Software ● Incorporating Sound and Music in Multimedia ● Basics of Animation ● 2D and 3D Animation Techniques **Virtual and Augmented Reality (VR/AR) & Multimedia in Social Media.** ● Basics of VR and AR Technologies ● Developing Multimedia Content for VR and AR ● Social Media Platforms and Trends ● Creating Multimedia-rich Content for Social Media ● Social Media Campaign Planning and Execution	**15**

II	Practical Work	Practical Hours (60)
Week 1 & 2	**Graphic Design** :- Practical exercises using graphic design software to create posters, banners, and digital artwork (task: designing a Banner for an event)	**8**
Week 3 to 5	**Audio Editing**:- Audio recording ,Audio storage and conversion , Audio mixing and rendering.	**12**
Week 6 to 9	**Video Editing** :- Video Capturing and Editing, Effects and transitions,color correction, Video composition and rendering.	**16**
Week 10 to 12	**Animation :**- introduction to animation software and practical animation exercises (task: short animation sequence using a 2D / 3D Sequence)	**12**
Week 13 to 15	**Social Media Content Creation:** :- Planning and executing a social media campaign using the components of multimedia.	**12**

Pedagogy:	**Suggested strategies for use to accelerate the attainment of the various course outcomes.** 1. The lecture method need not be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. You may use a. Video/Animation to explain various concepts. b. Collaborative, Peer, Flipped Learning, etc. 2. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, and develops design thinking skills . 3. Introduce Topics in manifold representations. 4. Show the different ways to solve the same problem and encourage the students to come up with their own creative ways to solve them. 5. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding 6. To promote self-learning, give at least one assignment (equivalent to 50% assignment weightage) where they can complete one MOOCs (certificate or equivalent) course out of lecture hour. 7. Practical shall be performed in the laboratory as indicated in the syllabus. 8. A softcopy of e-journal shall be maintained clearly mentioning the name of the experiment and other required information. 9. Mini-Project may be given as part of assessment
References/ Readings:	**Main Reading :** 1. Brie Gyncild. (2012) Adobe Photoshop CS6. Pearson Education 2. Mischeal Hammel,(2012) The Artist's Guide to GIMP, 2nd Edition,No Starch Press 3. Ranjan Parekh,(2017) Principles of Multimedia.2nd Edition. McGraw Hill

	Additional Reading 1. Douglas Spotter Eagle ,(2004) Using Soundtrack , 1st Edition .CMP Books **2.** Kusum Lata and Rishabh Anand (2015) ,Computer Graphics and Multimedia, Satya Prakashan
Course Outcomes:	**On completion of the course, students will be able to:** **CO1.** Remember the Multimedia elements **CO2.** Understand methods for integrating different types of media seamlessly into multimedia projects **CO3.** Apply design principles specific to multimedia , Ensuring visually appealing and effective communication **CO4.** Implement and Execute multimedia projects applying design principles ensuring practical application of visual and interactive design concepts.

Name of the Programme: Bachelor of Computer Applications
Course Code: CSA-242
Title of the Course: Search Engine Optimisation
Number of Credits: 3 (1T + 2P)
Effective from AY: 2024-25

Prerequisites for the course	None	
Course Outcomes :	1. Learn the concept of Search Engine, Search Engine Optimization and importance of Links in SEO. 2. Understand Web Analytics, Search Engine Optimization, and Search Engine Marketing. 3. Analyse data and assess reports on traffic to web sites; 4. Implement page ranking in order to improve website visibility in search engine listings.	
Units	**Content**	**No of hours 75 (15T+60P)**
I	**Introduction to SEO Basics** What is SEO and key factors determine the same, Components of SEO - onsite and off page, Keyword Planning, Long tail keywords; Art and science of tags - URL, title, meta, H1, alt text, etc, Write a good meta description; Page speed, All about links - broken, internal, Dealing with duplicate content, Robot.txt and Sitemap **Linking Strategies** Importance of Links, Inbound and Outbound, PageRank, Internal links and external links, Need to link to forum, blogs and social media sites link farm. **Content Design and Page Optimization** Correcting source code of the website, Mobile Optimization and responsiveness of a site, Choosing the best writing style, Creating unique content, building infographics, Rewriting content in avoid duplication or plagiarism issues to avoid Search engine penalization **Decompile a Competitor's Website** Ways to beat the competition, Using Google Chrome, Firefox, IE as a research tool, find your competition, Find why they have	**15**

	good search engine rankings, check the number of cached pages of the website, analyze their site architecture, find the keywords, finding who links to them. **SEO Tools** Setup and use a Google Webmaster Account, Verify your website, Setup and register a Google sitemap Produce and install a robots.txt file **SEM** Introduction to SEM, Link building, blogging, social media, Viral marketing, PPC, PPA campaigns, ad campaigns, Email marketing, Affiliate marketing, Podcasting,, Rich media, Managing Ad Campaign, Campaign Targeting, PPC management and SEO Major ad networks, "Content network" vs search advertising, Writing effective ads, Creating a landing page, Conversions and calls-to-action. A/B Testing.	
II	**List of Practicals:**	**Practical Hours (60)**
Week 1	Assign a website with significant traffic for analysis to Decompile a Competitor Website: ● How to beat the competition How to use Google Chrome as a research tool ● How to find your competition ● How to find why they have good search engine rankings ● How to check the number of cached pages ● How to analyse their site architecture ● How to find the keywords they use ● How to find who links to them	**4**
Week 2	Create a relevant website to host keeping in mind: ● CSS vs table-based design ● Understanding website frames ● How to choose the best domain name ● How to choose the best hosting company ● How to validate your website pages	**4**
Week 3 & 4	Improve poorly focused pages of the website: ● Take an existing site/page and begin to optimize it with enhanced content and design. ● optimize page and file names ● Choose the appropriate website theme ● structure your page content	**8**

	Correct the code, optimize Meta tags, optimize page title tags, optimize Meta description tags, optimize Meta keywords, optimize h tags, optimize li tags, optimize p tags, optimize alt tags, optimize title attribute tags, avoid the misuse of header tags ● Assess your site for calls-to-action ● optimize your keywords ● Rewrite the content, using longtail keywords ● integrate social media • Build Mobile responsive pages • Choosing the best writing style • Review for duplicate content • Avoid penalization	
week 5	Reviewing website for duplicate content issues across other sites to avoid penalization	**4**
Week 6	Apply robot controls (produce and install robots.txt file).	**4**
Week 7	Use Keyword tools to find relevant and niche keywords and analyze competitors' keywords.	**4**
Week 8	Create Inbound(backlinks) and Outbound links • Reviewing Page ranks so the best source links are utilized to build rank for your website(websites, forums, blogs, social media) • build a link farm	**4**
Week 9 & 10	Use Google Tag Manager to configure and deploy Google Analytics into your website Google. • Monitor traffic , and sessions and generate reports by analyzing the data, concentrating on different metrics used.	**8**
Week 11	Setup Google Search Console Tools and Yahoo! Site Explorer	**4**
Week 12	Setup and Register site to Google, Yahoo! And Bing: URL and Sitemaps	**4**
Week 13	Implement a comprehensive 301 redirect strategy to ensure smooth and SEO-friendly transitions when restructuring a website	**4**
Week 14 & 15	Improve load time of websites: Implement measures for Negative SEO attacks	**8**
Pedagogy	Course delivery pattern, evaluation scheme, prerequisite shall be discussed at the beginning. 1. Lectures preferably to be conducted with the aid of multimedia projector, black board, group activities, charts, cases, etc. 2. One internal written exam would be conducted as a part of internal theory evaluation. 3. One assignment based on the course content may be given to the students to evaluate how learning of objectives was achieved. It can incorporate designing of	

	problems and analysis of solutions submitted by the student's groups. E.g. 4. Give an individual Final semester Project to select/build a site built by students to apply analytics, SEO and SEM strategies. o Complete initial SEO of individual project site 5. Write a 1-page summary of organic traffic on group site. 6. Discuss the effect of designs on organic traffic. 7. Complete landing page Complete tweaks to site to improve your conversion rate 8. Track analytics	
References/ Readings:	**Main Reading:** 1. Danny Dover and Erik Dafforn; (2011) Search Engine Optimization (SEO) Secrets, Wiley Publication,1st edition 2. Peter Kent;(2015) Search Engine Optimization for Dummies , Wugnet Publications, 6th Edition. **Additional reading** 1. Eric Enge , Stephan Spencer, Jessie C. Stricchiola(2016), The Art of SEO: Mastering Search Engine Optimization 3rd Edition. Oreilly & Associates Inc 2. Peter Kent (2020).SEO For Dummies: Going Beyond the Buzzword to Continuously Drive Growth, Improve the Bottom Line, and Enact Change. 1st edition. For Dummies.	
Course Outcomes:	**On completion of the course, students will be able to:** **CO1.** Understand the concept of Search Engine, Search Engine Optimization and importance of Links in SEO. **CO2.** Apply Google Analytics and other metrics / tools to monitor progress in achieving search engine marketing goals and Create Pay-Per-Click Campaigns. **CO3.** Analyse websites and implement optimal Search Engine and marketing strategies for improved revenue generation. **CO4.** Create Web pages designed to be easily crawled and optimally indexed by search engines and Attract inbound Links from other Web Sites.	

Name of the Programme: Bachelor of Computer Applications

Course Code: CSA-243

Title of the Course: 3D Animation

Number of Credits: 3 (1T + 2P)

Effective from AY: 2024-25

Pre-requisites for the Course:	Basic concepts of animation	
Course Objectives:	1. Understand the basic concept of 3D animation and the its applications 2. Illustrate the importance of each process in 3D animation production pipeline . 3. Construct 3D models by employing textures, UVs, and shaders provided within a 3D modeling software. 4. Create an animation project by applying rigging, visual effects ,lighting, camera and rendering techniques provided within a 3D animation software.	
Units	**Content**	**No of hours 75 (15T + 60P)**
I	**Introduction to 3D Animation** • Defining 3D Animation • Exploring 3D animation Industry • The History of 3D Animation **Getting to Know the Production Pipeline** • Working in 3D Animation Preproduction : Idea/Story , Script/Screenplay, Storyboard, Animatic/Pre-visualization ,Design . • Working in 3D Animation Production : Layout , Research and Development ,Modeling ,Texturing , Rigging/Setup, Animation ,3D Visual Effects ,Lighting, Rendering. • Working in 3D Animation Postproduction: Compositing , 2D Visual Effects/Motion Graphics , Color Correction , Final Output Using Production Tools , Production Bible . **Understanding Modeling and Texturing** • Introduction to Modeling • Modeling Workflows : Primitive modeling, Box Modeling, Boolean Modeling • Texturing : Applying Textures • UVs : Unwrapping UVs & mapping texture • Shaders : Basic shader attributes- Color, Ambience, Transparency, Reflectivity, Refraction, Translucency,	**15**

	Specular highlights, Glow. **Rigging and Animation** ● Rigging - Parenting , Skeleton System ,Constraints. ● Animation – Keyframe, Timeline, Graph Editor , Function Curves, Dope Sheet , Tracking Marks and Ghosting. **Understanding Visual Effects, Lighting, Camera and Rendering** ● Visual Effects -- Particles , Hair and Fur , Fluids , Rigid Bodies , Soft Bodies (Cloth) ● Lighting -- Light Types : Spot, Point, Infinite, Area . Light Attributes – Color, Intensity, Shadows . Lighting Techniques - Three-Point Lighting, Two-Point Lighting One-Point Lighting. ● Camera – Camera View, Camera Attributes-Lens type: Perspective, Orthographic, Focal Length. ● Rendering – Render engines, Basic Rendering Methods	
II	**Practical Work** Using any suitable 3D Animation software like Blender, the concepts learned in the units are required to be implemented practically. The broad area of practical problems is mentioned below.	**Practical Hours (60)**
Week 1 & 2	Introduction to 3D Animation Software, exploring the Interface Basic Modeling Tools.	**8**
Week 3 & 4	Creating various 3D models with modeling tools, Editing Polygon Mesh, Curves and NURBS.	**8**
Week 5	Applying textures and materials to 3D Models.	**4**
Week 6	Working with UV maps	**4**
Week 7	Working with Shaders	**4**
Week 8	Working with Rigs and Constraints.	**4**
Week 9	Keyframe Animations.	**4**
Week 10	Working with Graph Editor ,Function Curves, Dope Sheet to create 3D animations .	**4**
Week 11	Working with Lights - Adding Lights to the scene, Light Types, World Settings and Attributes of Lights.	**4**
Week 12	Working with Cameras- Adding Cameras, Camera Navigation, Camera Properties, Animating and Switching cameras.	**4**

Week 13	Rendering – Explore Rendering Methods.	**4**
Week 14 & 15	Mini Project- Creating a short 3D Animation Scene.	**8**
Pedagogy:	**Suggested strategies for use to accelerate the attainment of the various course outcomes.** 1. The lecture method need not be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. You may use a. Video/Animation to explain various concepts. b. Collaborative, Peer, Flipped Learning, etc. 2. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, and develops design thinking skills . 3. Introduce Topics in manifold representations. 4. Show the different ways to solve the same problem and encourage the students to come up with their own creative ways to solve them. 5. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding 6. To promote self-learning, give at least one assignment (equivalent to 50% assignment weightage) where they can complete one MOOCs (certificate or equivalent) course out of lecture hour. 7. Practical shall be performed in the laboratory as indicated in the syllabus. 8. A softcopy of e-journal shall be maintained clearly mentioning the name of the experiment and other required information. 9. Mini-Project may be given as part of assessment	
References/ Readings:	**Main Reading:** 1. Beane, A. (2012). *3D Animation Essentials.* (1st ed.). John Wiley & Sons. 2. Kerlow, I. V. (2009). *The Art of 3D Computer Animation and Effects*. 3. Williams, R. E. (2009). *Animator's Survival Kit*. **Additional Reading:** 1. Park, J. E. (2004). *Understanding 3D Animation Using Maya*. 2. Blain, J. M. (2024). *The Complete Guide to Blender Graphics: Computer Modeling and Animation: Volume 1* (8th ed.).	
Course Outcomes:	**On completion of the course, students will be able to:** **CO1.** Understand various aspects of 3D Animation and understand the 3D animation production pipeline **CO2.** Apply 3D techniques that demonstrate characters with realistic motion **CO3.** Create sophisticated 3D models within a 3D environment **CO4.** Design and develop 3D animation scene	

Second Year - Semester IV

Name of the Programme: Bachelor of Computer Applications
Course Code: CSA-202
Title of the Course: Web App Development
Number of Credits: 4 (3P + 1 Tutorial)
from AY: 2024-25

Pte-requisites for the Course:	Basic Programming, Object-Oriented Concepts and DBMS Courses	
Course Objectives:	1. To understand the Fundamentals of client-side and server-side technologies 2. To understand dynamic and interactive web experiences using JavaScript and client-side frameworks. 3. To design web applications using server-side technologies and databases. 4. To apply secure web application deployment and maintenance.	
Units & Weeks	**Content**	**No of hours 105 (90P + 15 Tutorials)**
Tutorial Session Instructions	**Tutorial lecture of 1 hour duration to be conducted each week.** 1. Concepts needed for the conduct of Practical Sessions to be discussed. 2. These sessions may also be utilized for the doubt clearance 3. Suggestive client-side scripting language: JavaScript 4. Suggestive server-side scripting language: PHP 5. Suggestive frameworks for client-side scripting: Bootstrap, Zurb Foundation. 6. Suggestive frameworks for server-side scripting: Laravel, Code Igniter 7. Suggestive Database: MYSQL or MariaDB 8. Suggestive FTP Tool: FileZilla, cyberduck 9. Suggestive Control Panels: Plesk, CPanel 10. Suggestive Web server: Xampp, Wamp, EASYPHP	
I	**Client-side scripting language**	**35 (30 + 05)**
Week 1	● Introduction to client-side scripting language ● Naming convention for variables ● Operators ● Conditional statements	**7**
Week 2	● Loops ● Functions- named functions, anonymous functions, and arrow functions	**7**
Week 3	● DOM (Document Object Model) ● DOM Tree ● DOM Manipulation	**7**

	● Accessing elements using DOM	
Week 4	● Event Handling - Attaching events to HTML elements, Common events	**7**
Week 5	● AJAX- XMLHttpRequest Object, Working with Data Formats ● Cookie(get,set) ● Localstorage, ● Session storage	**7**
II	Client-side framework	**21**
Week 6	● Introduction to CSS frameworks ● Integrating Bootstrap into web application ● Understanding Bootstrap grid system	**7**
Week 7	● Bootstrap containers ● Bootstrap carousel, navbar, glyphicons	**7**
Week 8	● Bootstrap tables ● Bootstrap forms ● Bootstrap images ● Bootstrap typography ● Bootstrap color	**7**
III	**Server-side framework and Database connectivity**	**21**
Week 9	● Introduction to server-side scripting language ● Input/output statements ● Decision statements ● Looping statements	**7**
Week 10	● Database connectivity, CRUD (Create, Update, Read and Delete) ● Introduction to server-side frameworks ● Downloading and installing server-side framework ● Directory structure, modules, libraries ● APIs, configuring database connections	**7**
Week 11	● Handling database migrations and schema changes ● CRUD operations (Create, Read, Update, Delete) using framework	**7**
IV	**Data Representation and Web Hosting**	**28**
Week 12	● Data representation using XML ● Data representation using JSON	**7**
Week 13	● Web Hosting (Windows/Linux) ● Configuring Name Server ● Configuring email service ● Understanding Web Hosting file manager ● Cache Management ● Understanding and integrating SSL certificate into web application (OpenSSL)	**7**

Week 14 & 15	● Create a simple web application integrating client-side framework for styling and web interface, server-side scripting language and database connectivity with CRUD operations.	**14**
Pedagogy:	**Suggested strategies for use to accelerate the attainment of the various course outcomes.** 1. Lecture methods need not be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. You may use a. Video/Animation to explain various concepts. b. Collaborative, Peer, Flipped Learning etc. 2. Ask at least three HOT (Higher-order Thinking) questions in the class, which promotes critical thinking. 3. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, develop design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it. 4. Introduce Topics in manifold representations. 5. Show the different ways to solve the same problem and encourage the students to come up with their own creative ways to solve them. 6. Discuss how various concepts can be applied to the real world - and when that's possible, it helps improve the students' understanding 7. To promote self-learning give atleast one assignment (equivalent to 50% assignment weightage) where they can complete atleast one MOOCs (certificate or equivalent) course out of lecture hour. Test their understanding through quizzes or presentations. 8. One internal practical exam will be conducted as a part of internal evaluation. 9. Practical shall be performed in the laboratory as indicated in the syllabus. 10. A Hand written Hard Copy (or digital copy) of the journal shall be maintained clearly mentioning the name of the experiment and other required information.	
References:	**Main Reading :** 1. Harold, E. R., & Means, W. S. (2004). XML In A Nutshell (3rd ed.). O'Reilly. 2. Haverbeke, M. (2018). Eloquent JavaScript: A Modern Introduction to Programming (3rd ed.). No Starch Press. 3. Welling, L., & Thomson, L. (2016). PHP and MySQL Web Development (5th ed.). Pearson Education. **Additional Reading :** 1. Fielding, J. (2014). Beginning Responsive Web Design with HTML5 and CSS3. Apress. 2. Stauffer, M. (2023). Laravel: Up & Running: A Framework for Building Modern PHP Apps (3rd ed.). O'Reilly. 3. Sullivan, B., & Lui, V. (2012). Web Application Security, A Beginner's	

	Guide. McGraw-Hill Education. 4. Deitel, P. (2018). Internet and World Wide Web-How to Program (5th ed.). Pearson Education.
Course Outcomes:	**On completion of the course, students will be able to:** **CO1.** Understand and utilize JavaScript for dynamic web behaviors, including DOM manipulation and event handling. **CO2.** Apply a client-side framework for responsive, mobile-first web design components, and grid system to deliver visually appealing and user-friendly web experiences across various devices and screen sizes **CO3.** Compare and setup web hosting environments, generate and install SSL certificates, and integrate them with their websites. **CO4.** Design dynamic and interactive web applications to process user requests, interact with databases, manage server-side logic, and generate dynamic content.

Name of the Programme: Bachelor of Computer Applications

Course Code: CSA-203

Title of the Course: Agile Methodologies

Number of Credits: 4(3T+1P)

Effective from AY: 2024-25

Prerequisites for the Course:	None	
Course Objectives:	1. To remember the practices and philosophies of Agile methodologies. 2. To understand agile development and testing techniques. 3. To apply best practices of agile methodologies for software development and testing.	
Units	**Content**	**No of hours 75 (45T + 30P)**
I	**Introduction to Agile and Scrum** **Agile Methodology** Agile Software Development, Traditional Model vs. Agile Model, Classification of Agile Methods, Agile Manifesto and Principles, Agile Project Management, Agile Team Interactions, Ethics in Agile Teams, Agility in Design, Agile Documentations, Agile Drivers, Capabilities and Values. **Agile Processes:** Work Products, Roles, and Practices - SCRUM, SCRUM Meetings, SCRUM Artifacts, SCRUM Events, Scrum Ceremonies, Crystal, Feature Driven Development, Adaptive Software Development, Kanban, Extreme Programming, Lean Production.	**15**
II	**Agility and Knowledge Management:** Agile Information Systems, Agile Decision Making, KM in Software Engineering, Managing Software Knowledge, Challenges of Migrating to Agile Methodologies, Agile Knowledge Sharing, Role of Story-Cards, Story-Card Maturity Model (SMM). **Agility and Requirement Engineering:** Impact of Agile Processes in RE, Current Agile Practices, Variance, Overview of RE Using Agile, Managing Unstable Requirements, Requirements Elicitation, Agile Requirements Prioritization. Agile Product Development, Agile Metrics, feature-driven development (FDD).	**15**

III	**Extreme Programming :** Introduction, Values, Principles, Practices (Customer Testing, Refactoring, Pair Programming, Collective Ownership, TDD, Continuous Integration) **Agile Testing:** Testing - Aim and objectives, verification - validation: Testing Levels & Testing Strategies ● Behaviour Driven Testing ● Integration - top-down, bottom-up, bi-directional ● CI/CD Agile Approach to Quality Assurance, Test Driven Development, Agile Approach in Global Software Development.	**15**
IV	**Practical Work** Using suitable Agile Software Development tools (JIRA, Zephyr recommended), the concepts learned in the units are required to be implemented practically. The broad area of practical problems is mentioned/suggested below.	**Practical Hours (30)**
Week 1 & 2	To understand the background and driving forces for taking an Agile approach to Software Development.	**4**
Week 3	Understand the business value of adopting an agile approach.	**2**
Week 4 & 5	Installation, Configuration, and Understanding the various features of automated tools for Agile Software Development. (JIRA recommended)	**4**
Week 6 to 8	Agile workflow 1)Build a fitness tracker app that allows users to set fitness goals, track their progress, and receive personalized workout recommendations. Begin with features such as user registration, goal setting, and basic workout tracking. Iterate on the app by adding features like meal tracking, social sharing, and integration with wearable devices. 2)Develop an online learning platform. Start by creating user accounts, browsing courses, and enrolling in them. Implement features for course instructors to upload content and for students to interact through forums and quizzes. Enhance the platform with features like progress tracking, certificates upon completion, and peer-to-peer reviews. The above mentioned Projects to be created i. Creation of Project, SCRUM. ii. Creation of Backlog.	**6**
Week 9 & 10	iii. Creation of Sprint iv. Add stories to Sprint	**4**

Week 11 to 13	Test Management Activities i. Create a Test case for the above-mentioned projects. ii. Test Cases iii. Test Cycles iv. Update Test cases(passed/failed)	**6**
Week 14 & 15	i. Report Bugs ii. Reports	**4**
Pedagogy:	Suggested strategies for use to accelerate the attainment of the various course outcomes. 1. The lecture method need not be only a traditional lecture method, but alternative effective teaching methods could be adopted. You may use a. Video/Animation to explain various concepts. b. Collaborative, Peer, Flipped Learning etc. 2. Ask at least three HOT(Higher-Order Thinking) questions in the class that promote critical thinking. 3. Adopt problem-based learning(PBL), which fosters students' Analytical skills, and develops design thinking skills such as the ability to design, evaluate, generalize, and analyse information rather than simply recall it. 4. Introduce Topics in manifold representations. 5. Show the different ways to solve the same problem and encourage the students to come up with creative ways to solve them. 6. Discuss how every concept can be applied to the real world and when that's possible, it helps improve the students' understanding. 7. To promote self-learning, give at least one assignment where they can complete at least one MOOC (certificate or equivalent) course out of lecture hour. Test their understanding through quizzes or presentations.	
References/ Readings:	**Main Reading** 1. Anderson, D. J., & Schragenheim, E. (2003). Agile Management for Software Engineering: Applying the Theory of Constraints for Business Results. Prentice Hall. 2. Hazza, & Dubinsky. (2009). Agile Software Engineering, Series: Undergraduate Topics in Computer Science. Springer. **Additional Reading** 1. Desouza,K.C.,(2007). Agile Information Systems: Conceptualization, Construction, and Management. Butterworth-Heinemann. 2. Larman, C. (2004). Agile and Iterative Development: A Manager's Guide. Addison-Wesley.	
Course Outcomes:	**At the end of the course the students will be able to :** **CO1.** Remember the practices and philosophies of Agile methodologies. **CO2.** Understand agile development and testing techniques. **CO3.** Apply best practices of agile methodologies for software development and testing.	

Name of the Programme: Bachelor of Computer Applications
Course Code: CSA-204
Title of the Course: Object-Oriented Concepts
Number of Credits: 4 (3T + 1P)
Effective from AY: 2024-25

Prerequisites for the Course:	Knowledge of any basic Programming Language	
Course Objectives:	1. To remember Object-Oriented Programming concepts. 2. To understand object-oriented paradigms: abstraction, encapsulation, inheritance, polymorphism, and apply them in problem-solving 3. To apply object-oriented solutions for real-world problems. 4. To implement appropriate OO concepts in applications.	
Units	**Content**	**No of hours 75 (45T+30P)**
I	**Introduction to OO Programming** Introduction to Object-oriented programming Problems/ Limitations of Procedure-Oriented Programming Comparison of Procedure-Oriented and Object-Oriented Paradigms Object Oriented Programming Paradigms i. Classes & Objects ii. Inheritance iii. Polymorphism iv. Abstraction v. Encapsulation **Variables, scope, methods and Class Diagram** Introduction to variables, scope of variables-local, instance and class variables, Objects, Class, attributes, methods, static methods Relationship between Classes/ Objects using class diagrams and Aggregation	**15**
II	**Constructors, Destructors, and Polymorphism** **Constructors** Introduction, Types of Constructors and concepts used as Destructors, Compile and run time polymorphism **Operator and Function Overloading** Introduction Examples	**15**

	Inheritance Introduction, Base class and derived classes Private, Public and Protected members Types Of Inheritance i. Single Inheritance ii. Multilevel Inheritance iii. Multiple Inheritance iv. Hierarchical Inheritance v. Hybrid Inheritance	
III	**Method overriding** Virtual base classes (concept only) Abstract classes and Interfaces **Exception Handling** Introduction Types of errors Exception types-checked and unchecked Exception Handling Mechanism: Using try catch and multiple catch Nested try, throw, throws, and finally Creating user-defined Exceptions	**15**
IV	**Practical Work** The use of an object-oriented programming language for the concepts learned in the units from I to III is required to be implemented practically. The broad area of practical problems is mentioned below.	**Practical Hours (30)**
Week 1 to 3	**Introduction to Java** Application/Use of language, Simple Programs, arithmetic, logical and relational operators, Data types, Control statements, and Java Packages (Scanner, math), break and continue in loops. Predefined Java String and math functions Examples of programs: Create a simple program to print “Hello World” For if structure: -Using user input from the user check if an individual can vote or not For loop structure : -for, while, and do-while display the series 2,4,6,8,10 -Display Good Morning five times using a loop -Fibonacci series and Factorial of a number For menu-driven program : -display the area of squares, triangles, circles, and rectangles.	**06**

	-display appropriate object if a user selects a vowel (eg. A- apple, E-elephant). Use switch case and do-while loop. More programs may be given to the learners to complete and practice as part of their Practice Work.	
Week 4 & 5	Implementing Classes and objects, Array of Objects Examples of programs: Create a class dog with data members' breed, size, color, and age. Create 2 dog objects and display the details. Create a class book with data members' brands, pages, and prices. Use an array of objects. Create 6 books. Take user input. • More programs may be given to the learners to complete and practice as part of their Practice Work.	**04**
Week 6 to 8	Reading and writing data using methods, Modes of Parameter passing, and Return keyword. Examples of programs: create a class book with data members' brands, pages, and prices. using the concept of initializing by method to give values to the objects. Create 2 books. create a class purse with data members' color, brand, pockets, and price. using the concept of initializing by reference to give values to the objects. Create 2 purses. implement a program using the return keyword. More programs may be given to the learners to complete and practice as part of their Practice Work	**06**
Week 9 & 10	Constructors: Default, Parameterized, and Copy Examples of programs: Create a class rectangle with attributes length, breadth, and color. Create a rectangle using a default constructor. Create a class bag with attributes price, brand, color, and type(eg. college/office) Create 2 bags using one default and one parameterized constructor. Create a class shoe with data members' size, price, and color. create 3 shoes using default, parameterized, and copy constructors. More programs may be given to the learners to complete and practice as part of their Practice Work.	**04**

Week 11 & 12	Polymorphism: Function Overloading and function overriding, super keyword Examples of programs: Create class shapes with respective data members. Also, create classes of triangles and circles and calculate areas. Use the concept of polymorphism. Inheritance: Single, Multilevel, Multiple, Hierarchical, Hybrid, Method Abstract classes and interfaces Examples of programs: For single inheritance: Create a class vehicle with data members as the base class. Create a derived class motorbike from the vehicle. For multilevel inheritance: Create a class wristwatch with data members as the base class. Create a class custom_belt_wristwatch as the intermediary class. Create a class custom_bracelet_wristwatch as the derived class. More programs may be given to the learners to complete and practice as part of their Practice Work.	**04**
Week 13 to 15	**Exception Handling in Java** Syntax for Exception Handling, Throwing and Catching mechanism, rethrowing exceptions, multiple catch, Nested try, throw, throws, and finally User-defined Exceptions Examples of programs: Execute exceptions for arithmetic- division by zero, array index out of bounds, null pointer, string index out of bounds, etc. More programs may be given to the learners to complete and practice as part of their Practice Work	**06**
Pedagogy:	**Suggested strategies for use to accelerate the attainment of the various course outcomes.** 1. The lecture method need not be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. You may use a. Video/Animation to explain various concepts. b. Collaborative, Peer, Flipped Learning, etc. 2. Ask at least three HOT (Higher-Order Thinking) questions in the class, which promotes critical thinking. 3. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, and develops design thinking skills such as the ability to design,	

	evaluate, generalize, and analyze information rather than simply recall it. 4. Introduce Topics in manifold representations. 5. Show the different ways to solve the same problem and encourage the students to come up with their own creative ways to solve them. 6. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding 7. To promote self-learning, give at least one assignment (equivalent to 50% assignment weightage) where they can complete one MOOCs (certificate or equivalent) course out of lecture hour. Test their understanding through quizzes or presentations.
References/ Readings:	**Main Reading:** 1. Bhave, M., & Patekar, S. (2008). Programming with Java (1st ed.). Pearson. 2. Balagurusamy, E. (2010). Object-oriented programming with Java (4th ed.). Tata Mc Graw Hill Publishing House. 3. Schildt, H. (2017). The Complete Reference JAVA2 (10th ed.). Tata Mc Graw Hill Publishing House.
Course Outcomes:	**On completion of the course, students will be able to:** **CO1.** Remember Object-Oriented Programming concepts. **CO2.** Understand object-oriented paradigms: abstraction, encapsulation, inheritance, polymorphism, and apply them in problem-solving **CO3.** Apply object-oriented solutions for real-world problems. **CO4.** Implement appropriate OO concepts in applications.

Name of the Programme: Bachelor of Computer Applications
Course Code: CSA-205
Title of the Course: Web Technology
Number of Credits: 2T
Effective from AY: 2024-25

Prerequisites for the Course:	Basic understanding of using the internet and web browsers.	
Course Objectives:	1. To introduce the fundamentals of web technology, scripting languages, and web publication. 2. To create dynamic and interactive web experiences using JavaScript and client-side frameworks. 3. To apply client and server-side programming language that can be used to create websites and web applications. 4. To explore MVC Architecture for dynamic and interactive user interfaces using views and templates.	
Units	**Content**	**No of hours 30**
I	**Introduction to web technology** ● Internet, world wide web, web 2.0 ● Client/Server paradigm ● Protocols (TCP, IP, UDP, HTTP, HTTPS, FTP, TFTP, SMTP, MIME in brief) ● Functions and features of web servers and web browsers **Introduction to client-side scripting** ● Basics of JavaScript- syntax and data types ● DOM ● Accessing and modifying HTML elements with JavaScript ● Control structures (Conditional Statement, loops) ● Functions and events	**15**
II	**Introduction to server-side scripting** ● Overview of PHP, features ● PHP syntax and variables ● Input/Output statements ● Decision Statements ● Looping Statements ● Server-side validations Database Connectivity ● CRUD (Create, Update, Read and Update) operations ● Report Generation ● Session and cookies	**15**

	MVC Architecture ● Understanding the Model-View-Controller (MVC) architecture ● Role of Models, Views, and Controllers in web applications ● Views and templates: Creating dynamic and interactive user interfaces ● Implementing data models: Connecting to databases, retrieving and storing data **Web Publication** ● Hosting your Site ● ISP ● Domain Names ● Name Servers	
Pedagogy:	1. The lecture method need not be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. You may use a. Video/Animation to explain various concepts. b. Collaborative, Peer, Flipped Learning, etc. 2. Ask at least three HOT (Higher-Order Thinking) questions in the class, which promotes critical thinking. **3.** Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding.	
References/ Readings:	**Main Reading** 1. Luke Welling, Laura Thomson (2016). PHP and MySQL Web Development, 5th Edition, Pearson Education. 2. Paul Deitel (2018). Internet and World Wide Web- How to Program, 5th Edition, Pearson Education. **Additional Reading** 1. David Flanagan (2020). JavaScript: The Definitive Guide: Master the World's Most-Used Programming Language. 2. Prof. Satish Jain , M. Geetha Iyer (2020). O Level Made Simple – Web Designing & Publishing.	
Course Outcomes:	**On completion of the course, students will be able to:** **CO1.** Learn the fundamentals of web technology, scripting languages and web publication. **CO2.** Explain the concepts of creating dynamic and interactive web experiences using client-side scripting language. **CO3.** Apply client and server-side programming language that can be used to create websites and web applications. **CO4.** Analyze MVC Architecture for dynamic and interactive user interfaces using views and templates.	

Name of the Programme: Bachelor of Computer Applications

Course Code: CSA 221

Title of the Course: Digital Marketing

Number of Credits: 4 (3T + 1P)

Effective from AY: 2024-25

Prerequisites for the Course:	None	
Course Objectives:	1. To learn basic principles and concepts of digital marketing & advertising 2. To understand and familiarize the students with the concept of Digital Marketing and Search Engine Optimization. 3. to Analyze Marketing techniques like Adwords, search advertising, display advertising.	
Units	**Content**	**No of hours 75 (45T + 30P)**
I	**Fundamentals of Digital Marketing** Marketing in the digital world; Integrated marketing- The Phygital; Global trends in Digital Marketing; Digital channels- Paid, Owned and Earn; Fundamentals on the primary asset- your website; Careers in digital marketing; Skill development in digital marketing, Understanding Pay-per-click Advertisement; ; Keywords - planning, matching and combination , Keywords – significance and planning; Using Keyword Planner and other tools; Keyword matches and their usage.	**15**
II	**AdWords Fundamentals** Significance and evolution of AdWords in PPC, Bing Ads V/s Google Ads- overview; AdWords Certification- Overview, Benefits and Preparation; Google Ad Networks; Different Ad Formats, Campaign Structure and Organisation Quality, Rank and Relevance of Ads; Bidding and budget; Targeting Setting Extensions and their usage; Ad policies and approvals; Reports and Analysis, Metrics; Conversion Tracking; Campaign Optimisation **Search & Display Advertising with Adwords** Search with Adwords; Specifications of an Ad and how to put it to good use; Managing Invalid Clicks; Ad extensions and usage; Dynamic search ads; Landing page - your virtual front; AdWords APIs; AdWords editor- Benefits and usage; Managing multiple	**15**

	accounts. Display with Adwords, Google Display Network and Partnerships; Doubleclick Ad Exchange and AdSense, Campaign Creation and Structuring for display; Keyword and targeting through display network; Campaign Metrics: Analysis and optimization	
III	**SEO Basics** How search engines work; Different Search results and significance; Query types and significance; What is SEO and key factors determining the same; Components on SEO - onsite and off page; Keyword Planning; Using tools to get effective keywords; Long tail keywords - the hidden gems; Art and science of tags - URL, title, meta, H1, alt text, etc.; Write a good meta description; Page speed - its impact and improvement areas; All about links - broken, internal et al; Dealing with duplicate content; Robot.txt and Sitemap; Structured data and schema.org **SEO Advanced Concepts** Link building basics; Avoiding harmful links; Finding and leveraging link building opportunities; Creating a link building plan; Major Google updates and their implications on SEO; Using Search Console for SEO; KPIs of SEO; Tools for SEO; Moz SEO Products; SEMrush Competitive Research and Business Intelligence Software; Competition Analysis for SEO; Overall planning for SEO; Understanding nuances of local and international SEO; Accelerated mobile pages and SEO; Artificial Intelligence, Voice search and SEO – what to look forward	**15**
	List of Practicals	**30 Hours**
Week 1 & 2	1. Introduction to Digital Marketing and its Implementation in Business Scenarios. 2. Do a comparative analysis of their landing pages 3. Do a comparative analysis of their call to action (CTA) 4. Do a comparative analysis of website loading and website navigation 5. Find the rankings of Amazon, Flipkart, Snapdeal using Alexa. com	**04**
Week 3 & 4	6. Create the Digital Marketing Webpage 7. Go to any Web Hosting site and analyse the different kind of domain names, hosting options offered there. 8. Go to Wix.com and create a promotional web page in a shared hosting service	**04**

Week 5 & 6	9. Conducting Search Engine Optimization and Search Engine Marketing. 10. Use Google Adwords Keyword Planner - Select a Topic - Get Keyword Ideas	**04**
Week 7 to 9	11. Using Google Analytics to analyse website performance - Create a Google Analytics account - Install a tracking code in your Website. - Generate reports through Google Analytics - Unique Visitors, Sessions, Page Views, Referrer, Landing Page, Click through rate, Bounce rate and Exit rate, Conversion, Acquisition	**06**
Week 10 & 11	12. Creating Promotional banner through Canva. 13. Facebook Promotion using banners.	**04**
Week 12 & 13	14. Creating YouTube Channel for Markting 15. Email, YouTube and Instagram Marketing.	**04**
Week 14 & 15	16. Digital Marketing Analysis and Reports. - Analyze the change in ranking of your Web Promotion Page - Analyze the performance of your Facebook and Instagram Page - Analyze the performance of your YouTube Video,X and E-Mail Campaign - Create a comprehensive digital marketing strategy to reach out to your targeted customers in an effective manner.	**04**
Pedagogy:	**Suggested strategies for use to accelerate the attainment of the various course outcomes.** 1. The lecture method need not be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. You may use a. Video/Animation to explain various concepts. b. Collaborative, Peer, Flipped Learning, etc. 2. Ask at least three HOT (Higher-Order Thinking) questions in the class, which promotes critical thinking. 3. Adopt Case Studies Based Learning , which fosters students' Analytical skills, and develops design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it. 4. Introduce Topics in manifold representations. 5. Test their understanding through quizzes or presentations.	

References/ Readings:	**Main Reading** 1. Ben Hunt (2011). Convert!:(Designing Websites For Traffic and Conversions, John Wiley & Sons 2. Dave Chaffey & Fiona Ellis-Chadwick,(2019) Digital Marketing: Strategy, Implementation and Practice, Pearson Education 3. Ekaterina Walter,(2014) The Power of Visual Storytelling, McGraw-Hill Education **Additional Reading** 1. Anglona's Books. (2022). *Google Adwords 2022: A Beginner's Guide to BOOST YOUR BUSINESS Use Google Analytics, SEO Optimization, YouTube and Ads*. **2.** Marshall, P., Rhodes, M., & Todd, B. (2020). *Ultimate Guide to Google Ads.* December 10, 2020.
Course Outcomes:	**On completion of the course student will be able to** **CO1.** Understand digital landscape and build a case to leverage online channels **CO2.** Analyze online campaigns successfully and Develop and design Online Advertising campaigns, AdWords Campaign Management and Campaign Basics across search. **CO3.** Evaluate organic traffic through Search Engine Optimization and Apply advance concept of Search Engine Optimization to capture the right intent

Name of the Programme: Bachelor of Computer Applications

Course Code: CSA 222

Title of the Course: Data Analysis

Number of Credits: 4 (3T +1P)

Effective from AY: 2024-25

Prerequisite for the Course :	None	
Course Objectives :	1. To understand the fundamentals of Data Analysis. 2. To learn concepts of Data Visualization and Statistical Inference. 3. To perform Regression on a dataset. 4. To implement a comprehensive data analysis project based on a real-world scenario or dataset.	
UNIT	**Content**	**No of Hours 75 (45T+30P)**
I	**Foundations of Data Analysis** **Introduction to Data Analysis** ● Definition, importance, and applications of data analysis. ● Overview of the data analysis process. **Data Types and Sources** ● Types of data (categorical, numerical). ● Sources of data: structured vs. unstructured data. **Data Exploration and Descriptive Statistics** ● Descriptive statistics. ● Data visualization techniques. **Data Cleaning and Preprocessing** ● Handling missing data. ● Dealing with outliers. ● Data transformation. ● Feature scaling and normalization.	**15**

II	**Exploratory Data Analysis (EDA) and Statistical Inference** **Exploratory Data Analysis (EDA)** ● Univariate and bivariate analysis. ● Correlation and covariance. ● Outlier detection. **Data Visualization and Statistical Inference** ● Introduction to data visualization libraries (e.g., Matplotlib, Seaborn). ● Creating effective visualizations. ● Hypothesis testing. ● Confidence intervals. **Introduction to Data Modeling** ● Types of models (linear regression, logistic regression, decision trees, etc.). ● Model evaluation metrics.	**15**
III	**Regression Models** **Simple and Multiple Linear Regression** ● Estimating the Coefficients ● Assessing the accuracy of the Coefficient estimate ● Assessing the accuracy of the Model ● Estimating the Regression Coefficients **K-Nearest Neighbour** ● K-NN Demonstration with example ● Compare LR with k-NN ● Evaluation for regression ● Model selection and over-fitting	**15**
IV	**PRACTICAL WORK**	**30**
	List of practical :	
Week 1	Installing the software (R/Python/MS-Excel) and understanding the GUI and various menu options	**2**
Week 2	Types and sources of data	**1**
Week 3	Data Exploration and Descriptive Statistics	**2**
Week 4 & 5	**Data Cleaning and Preprocessing** 1. Introduce missing values and outliers to a dataset. 2. Implement techniques to handle missing data (e.g., imputation) and outliers (e.g., removal or transformation). 3. Normalize and scale numerical features.	**5**

Week 6 & 7	**Exploratory Data Analysis (EDA) using R/Python** ● Univariate and bivariate analysis. ● Correlation and covariance. ● Outlier detection.	**5**
Week 8 to 10	**Data Visualization (R/Python/Tableau)** 1. Explore the library for data visualization. 2. Create advanced visualizations, such as heatmaps and pair plots. 3. Apply data visualization techniques to a new dataset.	**7**
Week 11 & 12	**Regression Analysis** 1. Implement linear regression using a dataset. 2. Visualize the regression line and predictions.	**7**
Week 13 to 15	**Mini Project** 1. Formulate a data analysis project based on a real-world scenario or dataset. 2. Apply data cleaning, exploration, and modeling techniques. 3. Create a presentation or report summarizing the analysis and findings.	**6**
Pedagogy	1. At the start of course, the course delivery pattern, evaluation scheme, prerequisite will be discussed. 2. Lectures to be conducted with the aid of multi-media projector, black board, etc. 3. One internal written exam will be conducted as a part of internal theory evaluation. 4. One assignment based on the course content for each unit will be given to the student and evaluated at regular interval. 5. The course has lab component as integral part, where students have an opportunity to build an appreciation for the concepts being taught in Theory. 6. Experiments to be performed in the laboratory as suggested in the syllabus. 7. Mini Project applying all the learnt concepts.	
References	**Main Reading** 1. Jiawei Han, Micheline Kamber, 3rd Edition, (2011), Data Mining Concepts and Techniques, Morgan Kaufmann. 2. K.P. Soman, Shyam Diwakar and V. Ajay, (2016), Insight into Data mining Theory and Practice, Prentice Hall of India. 3. Pang-Ning Tan, Michael Steinbach, Vipin Kumar,, (2016), Introduction to Data Mining, Pearson Education.	
Course Outcomes	**On completion of the course, the students will be able to:** **CO1.** Demonstrate comprehension of core concepts and principles in data analysis, emphasizing foundational skills. **CO2.** Acquire proficiency in visualizing data effectively and making informed statistical inferences, showcasing an ability to interpret and communicate insights visually.	

	CO3. Demonstrate competence in selecting and applying regression techniques to analyze relationships within datasets, interpreting results, and drawing meaningful conclusions. **CO4.** Design and implement a data analysis project, showcasing the ability to apply learned concepts to solve real-world problems, effectively communicating findings and insights.s

Name of the Programme: Bachelor of Computer Applications
Course Code: CSA 223
Title of the Course: Advanced JavaScript
Number of Credits: 4 (3T+1P)
Effective from AY: 2024-25

Prerequisites for the Course	Basic Programming	
Course Objectives	1. To understand and execute JavaScript code in both browser and command-line environments. 2. To perform numerical operations, handle string manipulations, and apply Boolean logic. 3. To analyze nested objects, object methods and property deletion. 4. To Apply ES5 and beyond features of JavaScript.	
Units	**Content**	**No of Hours 75 (45T+30P)**
I	**Overview of JavaScript:** Brief history. Common use-cases (Eg: data validations, notifications etc). Runtime environments. ECMAScript standards. Overview of language features. Running JavaScript in the browser and at the command line. Debugging JavaScript in the browser. The console and REPL. **Basic syntax:** Values and literals. Primitive types. Numbers. Integer and floating point as a single type. Special floating point numbers. Rounding errors. The Math library. Strings. Immutability of strings. + and [] operators. toString. Common string utilities. Booleans. Ternary operator. Truth-y and False-y values. null and undefined. Regular expressions. Dynamic typing. Weak typing. The typeof operator. The === and !== operators. Control statements	**15**
II	**Arrays and Objects: Arrays.** Array insertion and deletion. Array length. Sparse arrays. Multidimensional arrays. Object as maps. Object creation, modification and lookup syntax. Nested objects. Object methods. The delete keyword. The for... in statement, and the hasOwnProperty method. The global window object. Object references. Aliasing. Pass-by-reference-copy semantics. **Functions:** Function declaration and invocation syntax. Anonymous functions. Functions as data. The arguments object. Variadic functions. Optional parameters. Named parameters. Function overloading. Duck typing.	**15**

III	**ES5 and beyond** Strict Mode, JSON (JavaScript Object Notation) New Array Methods: forEach(), map(),filter(), every(), some(), indexOf(), lastIndexOf() Object.create(), Function.prototype.bind(), Getters and Setters, Array.isArray(), String.trim() Arrow Functions, Let and Const, Template Literals, Destructuring Assignment, Default Parameters, Classes, Promises, Async/Await, Modules, Rest and Spread Operators, Map and Set, Proxy and Reflect.	**15**
IV	**Practical Work** Using javascript programming language, the concepts learned in the units from I to III are required to be implemented practically. The broad area of practical problems is mentioned below.	**Practical Hours (30)**
Week 1	Write simple JavaScript with HTML for arithmetic expression evaluation and message printing.	**2**
Week 2	Develop JavaScript to use decision making and looping statements	**2**
Week 3	Develop JavaScript to implement Array functionalities	**2**
Week 4	Develop Javascript to implement functions	**2**
Week 5	Develop JavaScript to implement Strings.	**2**
Week 6	Create web page using Form Elements and perform Validations	**2**
Week 7	Create web page to implement Form Events	**2**
Week 8	Develop a web page for creating sessions and persistent cookies. Observe the effects with browser cookies settings.	**2**
Week 9	Develop javascript to implement validations using regular expressions.	**2**
Week 10 to 15	Practicals based on ES5 and beyond features of JavaScript	**12**
Pedagogy:	Suggested strategies for use to accelerate the attainment of the various course outcomes. 1. Lecture method need not be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. You may use a) Video/Animation to explain various concepts. b) Collaborative, Peer, Flipped Learning etc. 2. Ask at least three HOT (Higher-Order Thinking) questions in the class, which promotes critical thinking. 3. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, develop design thinking skills such as the ability to design, evaluate,	

	generalize, & analyse information rather than simply recall it. 4. Introduce Topics in manifold representations. 5. Show the different ways to solve the same problem and encourage the students to come up with their own creative ways to solve them. 6. Discuss how every concept can be applied to the real world 7. To promote self-learning, give atleast one assignment where they can complete at least one MOOCs (certificate or equivalent) course out of lecture hour. Test their understanding through quizzes or presentations.
References/ Readings:	**Main Reading** 1. David Flanagan (2020). JavaScript: The Definitive Guide. O.Reily. 2. Minnick (2023). JavaScript All-in-One For Dummies. John Wiley & Sons Inc **Additional Reading** 1. Zachary Shute (2019). Advanced JavaScript. Packt Publishing. **2.** Laurence Lars Svekis, Maaike Van Putten, Rob Percival (2021). JavaScript from Beginner to Professional. Packt Publishing.
Course Outcomes	**On completion of the course, students will be able to:** **CO1.** Recall basic and advanced concepts and features of JavaScript. **CO2.** Understand the concepts and features of JavaScript. **CO3.** Apply JavaScript concepts to create and validate interactive web pages. **CO4.** Analyze the use and working of JavaScript to meet industry standards.

Name of the Programme: Bachelor of Computer Applications
Course Code: CSA-261
Title of the Course: Digital Media Marketing & Analytics [Exit Internship Course - 2]
Number of Credits: 4 (2T + 2P)
Effective from AY: 2024-25

Pre-requisites for the Course:	Website Designing and Programming knowledge	
Course Objectives:	4. To understand the concepts and techniques of Search Engine Optimization and Social Media Marketing. 5. To learn Web & Social Media Analytics, Inbound Marketing and emerging trends. 6. To apply the understanding of Search Engine Optimization, Social Media Marketing, web analytics and inbound marketing. 7. To analyze case studies of successful digital marketing campaigns and apply it in real-world scenario.	
Units	**Content**	**No. of Hours 90 (30T + 60P)**
I	**Search Engine Optimization** ● Introduction to SEO - How do Search Engines work?, Organic Search vs. Paid Search Results, Keyword Research ● On-page optimization - On-page SEO Elements, Technical SEO, Mobile SEO, Schema Markup ● Off-page optimization - Link Building, Social SEO, Local SEO, Backlink Audits using SEMrush ● SEO Audit, Tools, Measurement - SEO Audit, Algorithm Updates, Measurement with Google Analytics, SEO Resources, Careers in SEO **Social Media Marketing** ● Introduction to Social Media Marketing ● Creating Content for Facebook & Social Media, Tools for Content Creation ● Facebook Marketing - Facebook for Business, Facebook Insight, Facebook Pages and Post Best Practices, Facebook Ads – Campaign Objectives, Facebook Ads – Targeting Audiences, Facebook Ads – Impactful Creatives, Facebook Avatar, Apps, Live, Hashtags, Optimization and Reporting, Facebook Ad	**15**

	Policies, Facebook Messenger, Facebook Shop, Building Brand Awareness, Driving In-store Footfall, Facebook Pixel, Driving Online Sales, Generating Leads ● LinkedIn Marketing - Importance of LinkedIn presence, LinkedIn Strategy, Content Strategy, LinkedIn analysis, Targeting, Ad Campaign ● Instagram Marketing, X (Twitter) & Snapchat Marketing ● Social Media Marketing Tools, Crafting a Successful Social Media Strategy	
II	**Web and Social Media Analytics** ● Introduction to web analytic - What's analysis?, Is analysis worth the effort?, Small businesses, Medium and Large scale businesses, Analysis vs intuition ● Google Analytics -Getting Started With Google Analytics, How Google Analytics works?, Accounts, profiles, and users navigating Google Analytics, Basic metrics, Main sections of Google Analytics reports, Traffic Sources Direct, referring, and search traffic Campaigns AdWords, Adsense. ● Content Performance Analysis- Pages and Landing Pages, Event Tracking and AdSense, Site Search. ● Visitor Analysis- Unique visitors, Geographic and language information, Technical reports, Benchmarking. ● Social Media Analytics- Facebook insights, Twitter analytics, YouTube analytics, Social Ad analytics /ROI measurement. ● Actionable Insights **Inbound Marketing** ● Attracting your potential customers into the conversion funnel ● Converting your prospects into leads using emails ● Landing Page ● Conversion Optimization, Conversion Optimization Patterns for Engaging website Visitors ● Lifecycle Emails **Emerging Trends - An Introduction** ● AI and machine learning in digital marketing, Voice search optimization, Chatbots and conversational marketing, Augmented Reality (AR) and Virtual Reality (VR) marketing	**15**

III	**Practical Activities - To be carried out along in sync with the concepts mentioned in Unit I & II respectively.** 1. To learn to optimize web content for better search engine visibility, Perform keyword research using tools like Google Keyword Planner or SEMrush and optimize a webpage accordingly. 2. To understand the importance of content planning and creation, develop a content calendar for a hypothetical business, create blog posts or articles, and schedule their publication. 3. To gain hands-on experience in managing social media accounts and creating engaging content, create social media profiles for a business on platforms like Facebook, Instagram, and LinkedIn, and develop a social media content calendar. 4. To gain practical experience in launching and optimizing PPC advertising campaigns, set up a Google Ads campaign targeting specific keywords relevant to a business, create ad copies, and monitor the campaign's performance. 5. To collect and interpret data to measure the effectiveness, set up Google Analytics for a website, track key metrics such as traffic sources, user behavior, and conversions, and generate a report analyzing the data. 6. To gain practical experience in strategic planning and decision-making, develop a comprehensive digital marketing strategy for a fictional business, including setting objectives, identifying target audiences, allocating budgets, and selecting appropriate digital marketing channels. 7. To explore innovative ways to incorporate emerging trends, experiment with emerging technologies like AI-powered chatbots or virtual reality experiences and evaluate their potential applications in digital marketing.	**35**
IV	**Case Studies** Analyze case studies of successful digital marketing campaign, like 1. ICICI Bank: Building India's Most Social Bank on facebook 2. Barclays Business Banking SEO Campaign **Mini - Project** Develop a mini-project applying the insights gained from the case studies to a real-world scenario.	**25**

	Optional - *Prepare for industry-recognized certifications by taking practice exams, completing online courses, and participating in certification programs offered by platforms like Google, Facebook, or HubSpot. It will enhance the credentials and increase the employability in the digital marketing field.*	
Pedagogy:	Suggested strategies for use to accelerate the attainment of the various course outcomes. 1. A plan is to be developed by the student/s in consultation with the teacher incharge and to be approved. 2. One or methods mentioned below may be used for learning purposes. a. Intensive training / teaching b. Online or offline training (approved by the college or instructor) c. Approved MOOCS Courses d. Workshops - on-campus or off-campus e. Self-learning means & methods f. Enquiry-based learning 3. A work diary to be maintained where all the learning & work carried out to maintained and certified by the teacher incharges. 4. All deliverable & artifacts to be submitted in the college for evaluation and assessments.	
References/ Readings:	**Main Reading:** 1. Alhlou, F., Asif, S., & Fettman, E. (2016). Google Analytics Breakthrough: From Zero to Business Impact.(1st ed.). [Kindle Edition]. Wiley. 2. Deiss, R., & Henneberry, R. (2020). *Digital Marketing for Dummies.* [Paperback]. Wiley. 3. Enge, E., Spencer, S., & Stricchiola, J. (2023). *The Art of SEO.*(4th ed.). O'Reilly Media. 4. Gupta, Seema. (2022). *Digital Marketing* (3rd ed.). [Paperback]. McGraw Hill. 5. Rai, A. K. (2014). Social Media Marketing: Theories and Applications. Pearson Education India. **Additional Reading:** 1. Chaffey, D., Ellis-Chadwick, F., Johnston, K., & Smith, P. R. (2019). Digital Marketing: Strategy, Implementation, and Practice. Pearson. 2. Dover, D., & Agrawal, A. (2016). Search Engine Optimization (SEO) Secrets. Wiley. 3. Kumar, V. (2018). Analytics in Digital Marketing. Wiley. 4. Ratan, A. (2019). Digital Marketing: Concepts and Strategies. Oxford University Press.	

Course Outcomes:	On completion of the course, student will be able to **CO1.** Understand the concepts and techniques of Search Engine Optimization, Social Media Marketing, Web & Social Media Analytics, Inbound Marketing. **CO2.** Apply Search Engine Optimization, Social Media Marketing, web analytic and inbound marketing strategies. **CO3.** Analyze the performances of digital marketing campaigns. **CO4.** Create and run a small digital marketing campaign successfully.

Third Year - Semester V

Name of the Programme: Bachelor of Computer Applications

Course Code: CSA-300

Title of the Course: UI-UX Design

Number of Credits: 4 (3T + 1P)

Effective from AY: 2024-25

Pre-requisites for the Course:	None	
Course Objectives:	1. To understand user-centered design principles and practical skills in graphic design, prototyping, and usability testing. 2. To explore graphical user interfaces, affinity diagrams, personas, and scenarios. 3. To apply Acquire an understanding of various tools to enhance the design of user experiences. 4. To design wireframes and prototypes that prioritize user experience through iterative design, incorporating usability tests.	
Unit	**Content:**	**No of hours 75 (45T + 30P)**
I	**FOUNDATIONS OF UI DESIGN** ● Introduction to User Interface (UI) Design, The Relationship Between UI and UX, Roles in UI/UX, Formal/Active Elements of Interface Design, Composing the Elements of Interface Design, UI Design Process (Core stages) ● Visual and UI Principles - UI Elements and patterns-Interaction behaviors and Principles	**15**
II	**FOUNDATIONS OF UX DESIGN** ● Introduction to User Experience (UX) Design, application, and relevance in the current scenario, 5 Elements of UX - strategy, scope, structure, skeleton, surface ● Good and poor design, understanding your users, tools and methods used for UX design research, user needs and its goals, knowing about business goals ● Designing the Experience - Elements of User Experience, Visual Design Principles, Functional Layout, Interaction design, Introduction to the Interface, Navigation Design, User Testing, Developing and Releasing Your Design.	**15**

III	**UI/ UX Design and Testing** ● User Study- Interviews, writing personas: user and device personas, Creating User Stories, Creating Scenarios, Flow Diagrams, Flow Mapping, Information Architecture ● User Context, Responsive Design-Wireframing- Creating Wireflows- building a Prototype- building high-fidelity mockups, Sharing and Exporting Design, Conducting Usability tests, Other Evaluative User Research Methods in brief.	**15**
Unit IV Practical	**The practical exercises can be implemented utilizing any of the tools listed below.** ● Figma, Adobe XD, Penpot, Pencil, GIMP, Inkscape, etc.	**Practical Hours (30)**
Week 1 & 2	● Develop proficiency in iterative user-centered design for graphical user interfaces. ● Construct user interfaces for diverse applications.	**04**
Week 3 & 4	● Assess the user experience design of products or applications effectively. ● Exhibit user experience skills in the process of product development	**04**
Week 5 to 7	● Generate wireframes and prototypes as integral components of the design process. ● Implement responsive design techniques for seamless user experiences across devices. ● Employ A/B testing to evaluate and optimize different design variations.	**06**
Week 8 & 9	● Create detailed personas and scenarios to inform the UI/UX design process. ● Visualize user interactions and navigation through the development of flow diagrams and wireflows.	**04**
Week 10 & 11	● Develop an effective information architecture for a given project, focusing on content organization and structure. ● Translate wireframes into high-fidelity mockups, incorporating visual design elements.	**04**
Week 12 & 13	● Develop an interactive prototype that simulates user interactions with the finalized UI design. ● Create and implement a comprehensive user testing plan for a UI/UX design project.	**04**
Week 14 & 15	● Assess the accessibility of a given UI design to ensure it meets inclusive design standards.	**04**

Pedagogy:	**Suggested strategies for use to accelerate the attainment of the various course outcomes.** 1. The lecture method need not be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. You may use a. Video/Animation to explain various concepts. b. Collaborative, Peer, Flipped Learning, etc. 2. Ask at least three HOT (Higher-Order Thinking) questions in the class, which promotes critical thinking. 3. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, and develops design thinking skills such as the ability to design, evaluate, generalize, & analyze information rather than simply recall it. 4. Show the different ways to solve the same problem and encourage the students to come up with their own creative ways to solve them. 5. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding 6. To promote self-learning, give at least one assignment where they can complete one MOOCs (certificate or equivalent) course out of lecture hour. Test their understanding through quizzes or presentations.
References/ Readings:	**Main Reading:** **1.** Don Norman. (November 2013). *The Design of Everyday Things*. Basic Books. 2. Joel Marsh (2022). *UX for Beginners.* OReilly. 3. Wilbert O. Galitz (2007). *The Essential Guide to User Interface Design: An Introduction to GUI Design Principles and Techniques (Third Edition).* Wiley Publishing. **Additional Reading:** 1. Jesse James Garrett (2011). *The Elements of User Experience: User-Centered Design for the Web and Beyond (Second Edition).* Pearson Education. 2. Russ Unger and Carolyn Chandler (2012). *A Project Guide to UX Design: For user experience designers in the field or in the making (Second edition).* New Riders Publishing USA.
Course Outcomes:	**On completion of the course, students will be able to:** **CO1.** Remember the iterative user-centered design of graphical user interfaces and build UI for user applications. **CO2.** Understand the UX design of any product or application **CO3.** Apply UX skills in product development **CO4.** Design Wireframe and Prototype

Name of the Programme : Bachelor of Computer Applications

Course Code : CSA 301

Title of the Course : Full Stack Development

Number of Credits : 4 (3P + 1 Tutorial)

Effective from AY : 2024-25

Pre-requisites for the Course:	Basics of Web Technology & Web App Development	
Course Objectives:	1. To Learn JavaScript Fundamentals for Full-Stack Development 2. To Write Robust Backend APIs with Node.js 3. To Design Dynamic User Interfaces with React.js 4. To Integrate Data Flow between Frontend application and Backend Application	
Units & Weeks	**The broad area of practical concepts are mentioned / suggested below.**	**No of hours 105 (90P + 15 Tutorials)**
Tutorial Session Instructions	1. Tutorial lecture of 1 hour duration to be conducted each week. 2. Concepts needed for the conduct of Practical Sessions to be discussed. 3. These sessions may also be utilized for the doubt clearance	
I	**Introduction to Node.js**	**42 (36 + 06)**
Week 1	● Installation of Node.js ● Learn Node.js REPL ● Understanding Node js folder Structure ● Configuration of Package.JSON file in a new web application. ● Install Express ● Create a server using Express	**7**
Week 2	● Node Modules ● Module Dependencies ● Module Functionality	**7**
Week 3	● The Event Loop, Concurrency, Asynchronous Coding ● Callback Functions, Calling Conventions, Exception Handling ● Event Emitters, Listening for Events	**7**

Week 4	• Promises, Promise Chaining • Modules, Command Line Arguments • Working with the File System, Reading Files, Writing Files	**7**
Week 5	• Readable Streams, Writable Streams • The Standard Streams, Creating a Server, Routes • Accessing Request Headers • Create gateway using node js	**7**
Week 6	• Create cron jobs using Node js • Blocking vs Non Blocking methods • Webpack	**7**
II	**Backend APIs**	**28 (24+04)**
Week 7	• Installing Sequelize ORM for MySQL • Connecting to database • Testing the connection • Closing the connection	**7**
Week 8	• Create Models using sequelize • Sequelize Migration • Model Querying-Basics	**7**
Week 9	• Model Querying-Finders • Validation and Constraints • Raw Queries	**7**
Week 10	• Sequelize Association(1:1,1:M) • Advanced M:N Associations	**7**
III	**Front end Framework**	**28 (24+04)**
Week 11	• Installation of React js • Components (Build-in and Custom) • Props • States	**7**
Week 12	• Hooks(useState, useReducer, useContext, useRef, useEffect, useMemo, useCallback etc.)	**7**
Week 13	• Routes in React Js • Navigation	**7**
Week 14	• Redux • dispatch	**7**

IV	**Integrate between Front end and Back end Application**	**7 (6+1)**
Week 15	● Integrate Node js Application with React js	**7**
Pedagogy:	1. Course delivery pattern, evaluation scheme, prerequisite shall be discussed at the beginning. 2. Tutorials preferably to be conducted with the aid of multimedia projector, black board, LMS, mini projects etc. 3. One live project based on the course content may be given to the students to evaluate how learning of objectives was achieved. 4. The course has a separate laboratory, where students gain hands on experience of working with the various frameworks	
References/ Readings:	**Text Book** 1. Ethan Brown (2014). *Web Development with Node and Express: Leveraging the JavaScript Stack (Second edition).* O'Reilly. 2. Frank W. Zammetti (2020). *Modern Full-Stack Development.* Apress 3. Greg Lim. (July 2021). *Beginning MERN Stack Development. ISBN-10 9811815526.* Greg Lim.	
Course Outcomes	**On completion of the course, students will be able to** **CO1.** Understand JavaScript fundamentals **CO2.** Write Robust Backend APIs with Node.js **CO3.** Design Dynamic User Interfaces with React.js: **CO4.** Integrate Data Flow between Frontend and Backend applications	

Name of the Programme: Bachelor of Computer Applications

Course Code: CSA-302

Title of the Course: Cloud Computing

Number of Credits: 4 (3T + 1P)

Effective from AY: 2024-25

Pre-requisites for the Course:	The student should have basic knowledge of operating systems and computer networks.	
Course Objectives:	1. To describe the fundamentals of Cloud computing. 2. To understand the architecture and the types of Cloud systems. 3. To apply the concepts of service models and deployment models to decide suitability of migrating to cloud solutions. 4. To compare the services and applications made available by leading Cloud Service Providers	
Units	**Content**	**No of hours 75 (45T+30P)**
I	**Introduction to Cloud Computing** **Overview of Computing Paradigm** • Recent trends in Computing, Types of Computing: Parallel/Distributed computing, Grid Computing, Utility Computing, Cluster Computing, Cloud Computing. **Cloud Computing** • Introduction to Cloud Computing, Properties and Characteristics, Cloud service providers, Cloud applications, Cloud Architecture, Cloud Service Models **Deployment Models** • Types: Public Cloud, Private Cloud, Hybrid Cloud, Community Cloud; Key Drivers to adopting Cloud; Challenges and Issues • Popular Cloud Vendors (Amazon, Google, Microsoft etc.)	**15**
II	**IaaS - Infrastructure as a Service** • Introduction to Virtualization, Characteristics of Virtualized environment, Virtualization of Cloud, Types of Virtualization, Pros and Cons of Virtualization • Technology Examples- Xen, VMware, Microsoft Hyper-V **Capacity Planning** • Introduction, Defining Baseline and Metrics-Baseline Measurements, System Metrics, Load Testing, Resource Ceilings, Server and Instance types; Network Capacity, Scaling	**15**

III	**PaaS & SaaS** **Platform as a Service** • Introduction: Introduction to PaaS, Characteristics, Service Oriented Architecture (SOA), Applications, Issues and challenges. • Cloud Platform and Management: Computation, Storage, Case studies, Examples: Google App Engine, Microsoft Azure, SalesForce.com, Amazon AWS **Software as a Service** • Introduction to SaaS, Characteristics, Web Services, Web 2.0, Web OS, APIs, Service management, SaaS Implementation, Security, Case studies, Cloud Issues and Challenges: Cloud provider Lock-in, Security	**15**
IV	**List of Practicals:** The broad area of practical problems is mentioned/ suggested below:	**30**
Week 1 & 2	• Understanding Computer Network fundamentals and Designing LANs	**05**
Week 3 to 10	• Working on tools used in cloud computing online a) Storage b) Sharing of data c) Manage your calendar, to-do lists (e.g. Office365) d) A document editing tool • Leveraging any cloud service to work on document, spreadsheet, presentation, task management and collaborative tools in real time; chat with other collaborators. (e.g. Google sheet, docs & Google Meet, Google Keep)	**15**
Week 11 to 15	• Enlisting various companies in cloud business and the corresponding services provided by them and tag them under SaaS, PaaS & IaaS. • Exploring public cloud service providers' tools for exploring the usage of IaaS, PaaS and SaaS cloud services. a. AWS EC2 / Azure Compute b. AWS S3 / Azure Storage c. AWS VPC / Azure Vnets d. AWS Security / Azure Security	**10**

Pedagogy	1. The lecture method need not be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. You may use a. Video/Animation to explain various concepts. b. Collaborative, Peer, Flipped Learning, etc. 2. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding. 3. Explore the cloud platforms to solve real life problems. 4. To promote self-learning, give at least one assignment where they can complete one MOOCs (certificate or equivalent) course wherever necessary. Test their understanding through quizzes or presentations.
References/ Readings:	**Main Reading:** 1. Buyya, R., Vecchiola, C., & Selvi, T. (2013). *Mastering Cloud Computing*. TMH. 2. Halper, F., Hurwitz, R., Bloor, R., & Kaufman, M. (2010). *Cloud Computing For Dummies.* Wiley India Pvt. Ltd. **Additional Reading:** 1. Buyya, R. K., Broberg, J., & Goscinski, A. M. (2011). Cloud Computing: Principles And Paradigms. Wiley India Pvt. Ltd. ISBN-13: 978-81-265-4125-6 2. Sosinsky, B. (2011). Cloud Computing Bible. Wiley India Pvt. Ltd. ISBN-13: 978-81-265-2980-3
Course Outcomes:	**On completion of the course, students will be able to:** **CO1.** Recall the fundamentals of cloud computing. **CO2.** Understand the architecture and the types of cloud service models **CO3.** Apply the concepts of service models and deployment models for for migration to cloud. **CO4.** Analyze the services and applications made available by leading Cloud Service Providers

Name of the Programme: Bachelor of Computer Applications

Course Code: CSA-303

Title of the Course: Internet Technologies

Number of Credits: 2 (2T)

Effective from AY: 2024-25

Pre-requisites for the course:	None	
Course Objectives:	1. To understand the anatomy of the internet and the internet addressing Scheme. 2. Identify common security threats and attacks. 3. Utilize crawling and bots for efficient search engine performance.	
Units	**Content**	**No of hours 30**
I	**TCP/IP – Internet Technology and Protocol** ● Network Definition ● Network Components & Hardware ● Types of Networks: Peer to Peer, Client Server ● TCP/IP Structure **Network Communication:** ● Internet Layer Logical Addresses (IPv4): Classful and Classless Addressing, sub-netting, IPv4 vs IPv6. ● Network Address Translation (NAT), basics of ISPs ● Process-to-Process Delivery, Connectionless vs Connection Oriented and Reliable vs Unreliable; TCP and UDP ● DHCP, HTTP and HTTPS, DNS, TLDs	**15**
II	**Network Security** ● Overview of Network Security ● Importance of Firewalls in Network Security ● Common Security Threats and Attacks ● Basics of Firewalls - Definition and Purpose of Firewalls ● Aspects of security **Search Engines** ● Introduction ● Components of Search Engine ● Working of Search Engine in details	**15**

	Internet Applications ● FTP, Telnet, Email, Chat ● World Wide Web ● E-Commerce and Security Issues ● Emerging Trends	
Pedagogy:	**Suggested strategies for use to accelerate the attainment of the various course outcomes.** 1. Lecture method need not be only a traditional lecture method, but Alternative effective teaching methods could be adopted to attain the outcomes. You may use a. Video/Animation to explain various concepts. b. Collaborative, Peer, Flipped Learning etc. 2. Ask at least three HOT (Higher-Order Thinking) questions in the class, which promotes critical thinking. 3. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, develop design thinking skills such as the ability to design, evaluate, generalize, and Analyze information rather then simply recall it. 4. Introduce Topics in manifold representations. Show the different ways to solve the same problem and encourage the students to come up with their own creative ways to solve them. Discuss how every concept can be applied to the real world . 5. To promote self-learning give at least one assignment where they can complete at least one MOOCs (certificate or equivalent) course out of lecture hour. 6. Test their understanding through quizzes or presentations.	
References/ Readings:	**Main Reading:** 1. Andre S. Tanenbaum (2018). *Computer Networks 4th Edition.* Pearson Publication. 2. Greenlaw R and Hepp E (2007). *Fundamentals of Internet and www, 2nd EL.* Tata McGrawHill 3. Kurose, J. F., & Ross, K. W. (2017). *Computer Networking: A Top-Down Approach (6th ed.).* Addison-Wesley.	
Course Outcomes:	**On completion of the course, students will be able to:** **CO1** Recall the internet technologies **CO2** Understand the development of the internet ,the anatomy and growth. **CO3** Analyze the working of different protocols.	

Name of the Programme: Bachelor of Computer Applications

Course Code: CSA - 321

Title of the Course: Internship

Number of Credits: 4

Effective from AY: 2024-25

Pre-requisites for the Course:	None	
Course Objectives:	1. To carry out work-based vocational education and training to enhance substantial skill for employability at Semester-V. 2. To promote Analyze knowledge-gap, and plan & skill upgrade through training and self-development mode. 3. To develop decision-making and teamwork skills. 4. To provide sufficient hands-on learning experience related to the design, development and analysis of suitable product / process so as to enhance the technical skill sets in the chosen field.	
Units	**Content**	**No of hours**
I	1. The internship is to be carried-out by the student individually (or in a group of 5) and to be completed during the duration of semester-V in the field of Computer Applications. 2. The internship may be taken in any IT or IT enabled services Industry (in part time mode if permitted) or at the College (home institution). 3. The internship course shall include set of the following activities (but not limited to) in order to develop confidence, aptitude and skills during the course of internship: a. Orientation on the Internship process, conduct and expected course outcomes. b. Internship topic Identification: A list of topics (social/ organizational/ academic/ any other area) may be prepared by the College. c. Identification of tools & technologies needed. d. Gap Analysis of knowledge / skills needed to upgrade upon through training, workshop, and self-learning mode. e. Study journals / entrepreneurs of related & relevant area. f. Getting trained in the area of gaps identified as mentioned below i. Self-enroll in the training in online/offline mode at any institution of his/her choice. ii. Participation in the hands-on training/ workshop in the area of Application Development Tools & Technologies	**120**

such as Software Quality Assurance, IoT, Drone Technology, Machine Learning, AR / VR, Concepts & Tools, report writing, etc.

iii. Participation in the seminar related to internships and project best practices, latest tools and technologies, project/ internship topics identification, entrepreneurship, etc.

4. The College may decide till what extent to include and schedule the activities listed at point number (3) above in the academic year as per the need. More activities may be conducted according to the need.
5. The College may also decide whether the student interning in the industry (on part time) to be allowed to attend the set of activities scheduled as per point number (4) above or not. This is to be done well in advance, in consultation with the student and the institute/ organisation where student is interning.
6. At-most 60 hours of the time duration may be utilized to complete the tasks scheduled as per point number (4) above. This may be ensured by verifying the internship diary by the internship supervisor (industry supervisor).
7. The topic of the Internship (Or the training course & related project) shall be finalized by the student in consultation with the internship in-charge of the College/Programme/Industry Mentor (External Guide) of the company/institution in which the student is doing his/her internship.
8. The internship (internship project) is to be completed by the student in the 13^{th} week of the semester.
9. The industry supervisor shall certify, in the prescribed proforma, that the Internship is the work of the student completed under her/his supervision.
10. A student shall submit their Internship (or training & project) report to the College through the Industry supervisor (or training & project supervisor) at-least 15 days prior to the start date of Semester End Examination of semester V, or when intimated by the Faculty coordinator.
11. Ordinarily, no student shall be permitted to submit the Internship report after the due date specified by the College.

	12. The student is expected to present his/her work at the end of the Internship and should submit the internship report in the format as prescribed by the University. 13. Internship Report, Presentation and Viva shall be the integral component of the evaluation. 14. Students are instructed to refer the **"Computer Applications Internships and Project Guide"** prescribed by University for all necessary guidelines, instructions and formats.	
Pedagogy:	● As per the specification of Institution where student is seeking internship. ● As per the specification mentioned in the "Computer Applications Internships and Project Guide".	
References/ Readings:	1. Computer Applications Internships and Project Guide. 2. References as per the need of internship	
Course Outcomes:	**On completion of the course, students will be able to:** **CO1.** Understand the amount of complexity, effort and planning needed in solving real-world problems. **CO2.** Appreciate the need of training, gap analysis, and self- development. **CO3.** Demonstrate professional and ethical responsibility. **CO4.** Design and develop solutions of the internship problem through implementation of the skills developed during the course of study.	

Name of the Programme: Bachelor of Computer Applications

Course Code: CSA - 361

Title of the Course: Summer Internship

Number of Credits: 2

Effective from AY: 2024-25

Pre-requisites for the Course:	If students wishes to continue for Semester-V of Computer Applications Programme.	
Course Objectives:	1. To expose students as interns/trainees to the industrial environment. 2. To provide a platform to learn skills required for employability. 3. To inculcate work ethics.	
Content	1) This internship is to be carried-out by the students individually and to be completed in four weeks (30 hours per week) of duration during the summer term, i.e. *duration between end of semester IV and beginning of semester V.* 2) The internship topic shall be from the broad discipline of area of study i.e. Computer Application or allied. 3) The internship may be taken in any Firm, Industry, Organizations, , Health and allied areas, Local Governments (such as Panchayats and Municipalities), Parliament or elected representatives, media, artists, crafts persons, NGOs and other such organizations to improve their employability. 4) Online Internships are allowed. 5) If a student is unable to find the internship in any of the organization mentioned at **Sr. No. 3 and 4** above, then the student shall do the following: a) **Training (or self-learning):** Student shall enroll for any skill based vocational course of their choice, in any mode (Online/Offline), and at any institution of his/her choice. The course have to be completed in a maximum duration of 30 hours within 1.5 weeks duration. b) **Project:** A project of minimum 30 hours is to be completed in maximum duration of 3 weeks by using the skills developed in the training undertaken as per point no. (5.a) above and the skill developed during First and Second Year of the Computer Applications Programme. Guidance with respect to the project may be taken by the internship in-charge of the College.	**60 Hours**

	6) The topic of the Internship (Or the training course & related project) shall be finalized by the student in consultation with the internship in-charge of the College/Programme/Industry Mentor (External Guide) of the company/institution in which the student is doing his/her internship (Or training). 7) Upon completion of the internship program, the industry supervisor shall certify the intern, in a prescribed proforma, based on the conduct of the intern under her/his supervision. 8) A student shall submit their Internship (or training & project) report to the College through the Industry supervisor (or training & project supervisor) not later than one week after the start of fifth semester, or when intimated by the Faculty coordinator. 9) Ordinarily, no student shall be permitted to submit the Internship report after the due date specified by the College. 10) The student is expected to present his/her work at the end of the Internship and should submit the internship report in the format as prescribed by the University. 11) Internship Report, Presentation and Viva shall be the integral component of evaluation. 12) Students are instructed to refer the **"Computer Applications Internships and Project Guide"** prescribed by University for all necessary guidelines, instructions and formats in details.	
Pedagogy:	1. As per the specification of Institution/organization where student is seeking internship. 2. As per the specification mentioned in the "Computer Applications Internships and Project Guide".	
References/ Readings:	1. Computer Applications Internships and Project Guide. 2. As per the directives of the Industry/Organization.	
Course Outcomes:	**On completion of the internship program, students will be able to:** **CO1.** Understand the industrial environmental. **CO2.** Apply the concepts and skills learnt during employment and life-long learning. **CO3.** Inculcate discipline and work ethics.	

Third Year - Semester VI

Name of the Programme: Bachelor of Computer Applications
Course Code: CSA-304
Title of the Course: Cyber Security
Number of Credits: 4 (3T + 1P)
Effective from AY: 2024-25

Pre-requisites for the Course:	The student should have basic knowledge of information technology.	
Course Objectives:	1. To understand the concepts of cyber security, challenges and its awareness. 2. To comprehend the underlying principles of various cybersecurity techniques and technologies. 3. To apply cyber security measures to safeguard information and systems.	
Units	**Content**	**No of hours 75 (45T+30P)**
I	**a. Fundamentals of Cyber Security and Threat Landscape** ● Importance and challenges in Cyber Security ● Cyberspace, and Cyber threat ● Cyber warfare ● CIA Triad ● Cyber Terrorism ● Cyber Security of Critical Infrastructure **b. Cyber Attacks and Intrusion Techniques** ● Types of Hackers - Hackers and Crackers ● Cyber-Attacks and Vulnerabilities ● Malware threats ● Sniffing ● Gaining Access - Escalating Privileges ● Executing Applications ● Hiding Files ● Covering Tracks ● Worms, Trojans, Viruses, Backdoors ● Unauthorized Access ● Computer Intrusions ● White collar Crimes	**15**

	• Pornography • Software Piracy • Mail Bombs • Exploitation	
II	**a. Ethical Hacking and Information Security Practices** • Ethical Hacking Concepts and Scopes • Threats and Attack Vectors • Information Assurance • Threat Modeling • Enterprise Information Security Architecture • Vulnerability Assessment and Penetration Testing **b. Investigation** • Investigation Tools • eDiscovery • Digital Evidence Collection • Evidence Preservation • E-Mail Investigation • E-Mail Tracking • IP Tracking • E-Mail Recovery • Hands on Case Studies • Recovering Deleted Evidences • Password Cracking	**15**
III	**a. Social Engineering and Insider Threats** • Types of Social Engineering • Insider Attack • Preventing Insider Threats • Social engineering Targets and Defence Strategies • Securing data transit **b. Legal Framework and Countermeasures in Cyber Security** • IT Act • Hackers-Attack-Countermeasures • Web Application Security • Counter Cyber Security Initiatives in India	**15**

	● Cyber Security Incident Handling ● Cyber Security Assurance	
IV	**Practicals Works** The concepts learned in the units from I to III are required to be implemented practically. The broad area of practical problems is mentioned below.	**(30 Hours)**
Week 1 to week 5	● Implementation to gather information from any PCs connected to the LAN using whois, port scanners, network scanning, Angry IP scanners etc. ● Implementation of MITM-attack using wireshark or any network sniffers.	**10**
Week 6 to week 10	● Implementation of Windows security using firewall and other tools. ● Implementation to identify web vulnerabilities, using OWASP project. ● Disk Encryption Using Windows BitLocker, Disk Encryption Using Open Source Tools.	**10**
Week 11 to week 15	● Implementation to gather information from any search engine about a target entity. ● Implementation of IT Audit, malware analysis and Vulnerability assessment.	**10**
Pedagogy	1. The lecture method need not be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. You may use a) Video/Animation to explain various concepts. b) Collaborative, Peer, Flipped Learning, etc. 2. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding. 3. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, and develops design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it. 4. Show the different ways to solve the same problem and encourage the students to come up with their own creative ways to solve them. 5. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding.	

References/ Readings:	1. MariE-Helen Maras. (2nd Edition, 2014). *Computer Forensics: Cyber criminals, Laws, and Evidence.* Jones & Bartlett Learning. 2. Nihad Hassan, Rami Hijazi (2017). *Digital Privacy and Security Using Windows: A Practical Guide*. Apress. 3. Nilakshi Jain Wiley (2020). *Cyber Security and Cyber Laws.* Wiley. 4. Nina Godbole (2011). *Cyber Security.* Wiley.
Course Outcomes:	**On completion of the course, students will be able to:** **CO1**. Remember Legal Framework and Countermeasures of Cyber Security **CO2**. Understand the key concepts of cyber security, threat awareness and the fundamental principles of ethical hacking, techniques and tools. **CO3**. Apply the understanding of cyber security, threat awareness and the ethical hacking tools & techniques. **CO4**. Analyse the methods for authentication, access control, intrusion detection and prevention in Cyber Security.

Name of the Programme: Bachelor of Computer Applications

Course Code: CSA-305

Title of the Course: Mobile Application Development

Number of Credits: 4 (3P + 1 Tutorial)

Effective from AY: 2024-25

Pre-requisites for the Course:	None	
Course Objectives:	1. To understand the features and installation of Flutter 2. To get understanding of basic constructs of Dart programming. 3. To develop simple mobile applications in Flutter using dart and firebase.	
Units & Weeks	**Content**	**No of hours 105 (90P + 15 Tutorials)**
Tutorial Session Instructions	1. Tutorial lecture of 1 hour duration to be conducted each week. 2. Concepts needed for the conduct of Practical Sessions to be discussed. 3. These sessions may also be utilized for the doubt clearance	
I	**Introduction**	**07**
Week 01	Getting Started with Android – Installing the Development Environment, Configuring Android Stack, Configuring and Installing Flutter SDK, Creating a New Flutter Project and Understanding Folder Structure.	07
II	**Dart Programming**	**35**
Week 02	Introduction to Dart Programming: Using dart pad, data types, variables, Dart Programming: loops, decision making, functions	7
Week 03 & week 04	OOP concept in dart, getters and setters Exception handling and debugging	14
Week 05 & week 06	Asynchronous and synchronous operations async, await, streams, listening to streams, broadcast streams, manipulating streams	14
III	**Flutter**	**42**
Week 07 to week 09	Introduction to Flutter Widgets: Scaffold Widget. Image Widget, Container Widget, Column and Row Widgets, Icon Widget Layouts in Flutter, Card Widget, Stateful and Stateless Widgets Hot Reload and Hot Restart Styles and assets: Custom fonts, assets in flutter, media query, Null safety *Create a Restaurant Menu using Flutter Widgets*	21

	Button Widget: FloatingActionButton, RaisedButton, FlatButton, and IconButton, DropdownButton Button Widget: OutlineButton, ButtonBar, PopupMenuButton Navigation and Routing: Navigate to a New Screen and Back, Navigate with Named Routes, Send and Return Data Among Screens	
Week 10 to week 12	Motion Rich Widgets: BottomNavigatorBar Widget, DefaultTabController, TabBar, and TabBarView Widgets Motion Rich Widgets: ListTile Widget, ListView Widget, Drawer widgets Motion Rich Widgets: DataTable Widget, SelectableText Widget, Stack Widget Input and Selections: Text Field Widget, CheckboxGroup and RadioButtonGroup Widgets .DatePicker, Time Picker, Slider Widget, Switch Widget Dialogs, Alerts, and Panels: Alert Dialog Widget, Cupertino Alert Dialog Widget, Expansion Panel Widget, Snack Bar Widget *Creating a Hotel Reservation App using Widgets*	21
IV	**Firebase**	**21**
Week 13 to week 15	Firebase with flutter: Add firebase to flutter application, register app with firebase, firebase database and authentication Firebase with flutter: firebase cloud messaging, notification handling, using firebase storage with flutter Create a User Profile Interface using Firebase, Adding a Google Map on Your Flutter App Screen, Adding a Google Map Marker	21
Pedagogy:	**Suggested strategies for use to accelerate the attainment of the various course outcomes.** 1. Lecture methods need not be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. You may use a) Video/Animation to explain various concepts. b) Collaborative, Peer, Flipped Learning etc. 2. Ask at least three HOT (Higher-order Thinking) questions in the class, which promotes critical thinking. 3. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, develop design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it.	

	4. Introduce Topics in manifold representations. 5. Show the different ways to solve the same problem and encourage the students to come up with their own creative ways to solve them. 6. Discuss how various concepts can be applied to the real world - and when that's possible, it helps improve the students' understanding 7. To promote self-learning give atleast one assignment where they can complete atleast one MOOCs (certificate or equivalent) course out of lecture hour. Test their understanding through quizzes or presentations. 8. One internal practical exam will be conducted as a part of internal evaluation. 9. Practical shall be performed in the laboratory as indicated in the syllabus. 10. A Hand written Hard Copy (or digital copy) of the journal shall be maintained clearly mentioning the name of the experiment and other required information.
References / Readings:	**Main Reading** 1. Marco L. Napoli. (September 2019). *Beginning Flutter: A Hands On Guide to App Development (First Edition).* Wiley publication. 2. Nathan Metzler. (April 2022). *Dart Programming for Beginners: An Introduction to Learn Dart Programming with Tutorials and Hands-On Examples.* Kindle **Additional Reading** 1. Simone Alessandria, Brian Kayfitz. (2021). Flutter Cookbook. Packt Publishing. 2. Thomas Bailey, Alessandro Biessek**.** (2023). Flutter for Beginners (Third Edition). Packt Publishing.
Course Outcomes:	**On completion of the course, students will be able to:** **CO1.** Recall the installation process of Flutter, Dart and Firebase. **CO2.** Understand the various concepts and constructs of Mobile Application Development using Flutter, Dart and Firebase. **CO3.** Design and Develop animation & application using Flutter, Dart and Firebase. **CO4.** Debug and Analyze the programming logic.

Name of the Programme: Bachelor of Computer Applications

Course Code: CSA 306

Title of the Course: Machine Learning

Number of Credits: 4 (3T +1P)

Effective from AY: 2024-25

Pre-requisite for the Course :	**None**	
Course Objectives :	1. To learn the fundamentals of Data Analysis and the Science behind it. 2. To apply Machine Learning algorithms for performing complex data analysis. 3. To discover interesting patterns, correlations, associations and causal structures in the data found in data repositories. 4. To solve problems using fundamental concepts (Case Studies)	
UNIT	**Content**	**No of Hours (75) (45T + 30P)**
I	**Fundamentals of Analytics and Statistics** • Various Data Science Disciplines: Data Science and Business Buzzwords, Difference between Analysis and Analytics, Continuing with BI, ML and AI. • Careers in Data Science: Finding the Job - What to Expect and What to Look for. • Identification of a data science project. **Data Wrangling and Data Analysis** • Roadmap to Data Science workflow, Introduction and Implementation of Inferential and Descriptive Statistics. • Cleaning Data: Missing Values, Outliers. • Preparing Data for Modelling: Transformations, Derived Variables. Visualization Methods and Applications. • Case Studies. **Feature Selection and Dimensionality Reduction** • Why to do Feature Selection? • Feature Selection Techniques • Feature Selection vs Dimensionality Reduction	**15**

II	**Introduction to Machine Learning, Regression And Classification Models** • Overview of Machine learning • Overview of Statistical learning • Supervised Versus Unsupervised Machine Learning • Regression Versus Classification Problems • Simple Linear Regression • Multiple Linear Regression • Linear Discriminant Analysis • Logistic Regression • Naive Bayes • K-Nearest Neighbours • Artificial Neural Networks	**15**
III	**Tree Based Model, Unsupervised Learning, Association** **Basics of Decision tree** • Bagging and Boosting • Random Forest • Gradient Boosting Machines **Overview of Clustering** • K-means Clustering • K-medoid **Overview of Association Rule Mining** • Market Basket Analysis	**15**
IV	**PRACTICAL** **Tools to be used** **Programming Languages** : Python / R **Packages required** : numpy, pandas, scikit-learn **List of Practicals :**	**30**
Week 1 & week 2	• Merging several data sources into one data-set for analysis • Identifying gaps or empty cells in data and either filling or removing them and deleting irrelevant or unnecessary data • Identifying severe outliers in data and either explaining the inconsistencies or deleting them to facilitate analysis	**04**
Week 3 to week 5	Data Wrangling and Data Analysis • Feature selection and Data reduction • Covariance-based • Feature Selection using ANOVA F-Score	**06**

Week 6	Introduction to Machine Algorithms	**02**
Week 7 to Week 12	Regression And Classification Models and Tree Based Models • Experiments using Linear and Multiple Regression • Experiments using Decision Tree • Experiments using Random Forest	**12**
Week 13 to Week 15	Unsupervised Machine Learning and Association • Experiments using K-Means Clustering • Experiments using Dendrogram	**06**
Pedagogy:	1. At the start of course, the course delivery pattern, evaluation scheme, and prerequisites will be discussed. 2. Lectures to be conducted with the aid of multimedia projector, black board, etc. 3. One internal written exam will be conducted as a part of internal theory evaluation. 4. One assignment based on the course content for each unit will be given to the student and evaluated at regular intervals. 5. The course has a lab component as an integral part, where students have an opportunity to build an appreciation for the concepts being taught in Theory. 6. Experiments to be performed in the laboratory as suggested in the syllabus. **7.** Data Science Projects of basic level, if needed. **8. Data Science Methodology** ● Problem to Approach ● Requirements to collection ● Understanding to preparation ● Modelling to Evaluation ● Deployment to Feedback	
References:	1. Jiawei Han, Micheline Kamber, 3rd Edition(2011). *Data Mining Concepts and Techniques.* Morgan Kaufmann. 2. K.P. Soman, Shyam Diwakar and V. Ajay (2016). *Insight into Data mining Theory and Practice.* Prentice Hall of India. 3. Pang-Ning Tan, Michael Steinbach, Vipin Kumar (2016). *Introduction to Data Mining.* Pearson Education.	

Course Outcomes:	**At the end of the course, the students will be able to** : **CO1.** Demonstrate a solid understanding of the fundamentals of Machine Learning. **CO2.** Apply Machine Learning algorithms proficiently to perform complex data analysis tasks. **CO3.** Identify and interpret interesting patterns, correlations, associations, and causal structures within diverse datasets. **CO4.** Solve data science problems using fundamental concepts through case studies.

Name of the Programme: Bachelor of Computer Applications

Course Code: CSA - 307

Title of the Course: Project

Number of Credits: 4

Effective from AY: 2024-25

Pre-requisites for the Course:	None	
Course Objectives:	1. To provide students with knowledge of practical skills for various technological applications. 2. To enable the student to develop an application with their respective domain. 3. Ensuring the formation of research thinking of students, forming a clear idea of the main task and ways to solve them. 4. Developing the basic skills for problem-solving that arise in the course of research/development activities.	
Units	**Content**	**No of hours**
I	1. The Project is to be carried out in a group of students (as mentioned in ordinance OA38) and is to be completed during the duration of semester VI in the field Study. 2. The Project shall include a set of the following activities (but not limited to) to develop confidence, aptitude, and skills during the course of the project a) Orientation on the process, conduct, and expected course outcomes. b) Topic Identification: A list of topics (social/ organizational/ academic/ any other area) may be prepared by the students. c) Identification of tools and technologies needed. d) Conduct a literature review and understand gap analysis. e) Getting trained in the area of gaps identified. 3. The Project Guide in every college may decide to what extent to include and schedule the activities listed at point number 2 in the academic year as per the need. More activities may be conducted according to the need. This is to be done well in advance, in consultation with the Project Guide and the institute/organization where students are undergoing training. 4. The topic of the project shall be finalized by the student in consultation with the Project Guide. 5. The background work, group formation, assignment of guide, selection of project titles, problem definition formulation,	**120**

	decision on technology stack, and planning may be completed before the beginning of 6th Semester in consultation with the project guide. 6. The project is to be completed by the student by the 11th week of the semester. 7. The Project Guide shall certify, in the prescribed proforma, that the project is the work of the student completed under her/his supervision. 8. A student shall submit their project report in the format as prescribed by the University to the College at least a month before the start date of the Semester End Examination of semester VI, to be sent to the External Examiner decided by the university. 9. No student shall be permitted to submit the project report after the due date specified by the College/ University. 10. Project Report, Presentation, and Viva shall be the integral component of the evaluation jointly conducted by the Project Guide and External Examiner. 11. The final project report will be certified by the Project Guide, External examiner, and the head of the institution. 12. Students are instructed to refer to the **Computer Applications Project Manual** prescribed by the University for all necessary guidelines, instructions and formats.	
Pedagogy:	As per the specification mentioned in the **Computer Applications Project Manual**.	
References/ Readings:	Computer Applications Project Manual.	
Course Outcomes:	**On completion of the course, students will be able to:** **CO1.** Understand the amount of complexity, effort, and planning needed in solving real-world problems. **CO2.** Demonstrate the need for training, gap analysis, and self-development, professional and ethical responsibility. **CO3.** Design and develop solutions to real-world problems adhering to coding learned during the course of study. **CO4.** Evaluate using quality testing standards.	

Name of the Programme: Bachelor of Computer Applications

Course Code: CSA-323

Title of the Course: Social Media Marketing and Analytics

Number of Credits: 4 (3T+1P)

Effective from AY: 2024-25

Pre-requisites for the Course:	None	
Course Objectives:	1. To understand the concept of Social Media Marketing platform. 2. To acquire understanding of Facebook, Instagram, LinkedIn, Twitter, Pinterest Marketing 3. To understand video and mobile platform advertising and concepts of web and google analytics 4. To Measure, and Analyze Social Media Marketing Campaigns	
Units	**Content**	**No of hours 75 (45T + 30P)**
I	**Introduction to Social Media Marketing** ● Evolution and significance of social media. ● Understanding the potential benefits of social media. ● Overview of different social media platforms. **Managing Information – Aggregators** ● Introduction to information aggregators. ● Effectively managing and curating content. **Facebook & Instagram Marketing** ● Creating and managing groups and pages on Facebook. ● Tips and guides for effective posts, paid promotions, and contests. ● In-depth exploration of Facebook Ads, Ad Manager, Power Editor, and targeting strategies. ● Utilizing Facebook tabs, apps, and understanding Facebook Page Insights. **Twitter, LinkedIn, Pinterest** ● Twitter setup, usage tips, and terminology. ● LinkedIn profile review and usage guides. ● Pinterest setup and management strategies.	**15**

II	**YouTube Video and Mobile Advertising** **YouTube Channel Management** ● Setting up a YouTube channel. ● Content management and optimization. ● Practical examples and strategies for effective channel management. **Video and Mobile Advertising** ● Importance of YouTube in marketing. ● YouTube formats, tools, and targeting. ● Video campaign creation, tracking, optimization, and analytics. ● Mobile advertising: Key objectives, ad formats, networks, site, and app considerations. **Social Media Marketing Strategy** ● Introduction to Social Media Marketing Strategy ● Audience Identification and Persona Development ● Platform Selection and Planning ● Content Creation and Calendar Management ● Paid Advertising Strategies ● Monitoring and Analytics	**15**
III	**Introduction to Analytics Tools** ● Overview of Social Media Analytics ● Importance of Analytics in Social Media Marketing ● Understanding key metrics (engagement, reach, impressions) ● Defining Key Performance Indicators (KPIs) for social media ● Setting SMART (Specific, Measurable, Achievable, Relevant, and Time-bound) goals for social media campaigns **Introduction to Facebook Analytics and Instagram Insights** ● Connecting Instagram Business Account to Facebook ● Accessing Facebook Analytics and Instagram Insights ● Understanding Key Metrics on Facebook and Instagram ● Engagement Metrics (Likes, Comments, Shares) ● Reach and Impressions ● Click-Through Rates (CTR) and Conversion Metrics **Hootsuite Analytics** ● Hootsuite Analytics Overview ● Exploring Hootsuite Reports: Overview, Engagement, Trends ● Social Listening with Hootsuite	**15**

IV	**Practical** Students are expected to have a valid account of following social media platforms: Google, YouTube, Facebook, Twitter, Pinterest, LinkedIn, Hootsuite	**(30)**
Week 1 & week 2	**Comparison of Social Media Platforms:** Analyze and compare different social media platforms, outlining their unique features, target demographics, and potential for marketing **Information Aggregator Implementation:** Set up an account on an information aggregator (e.g., Feedly) and curate relevant content for a specific industry or topic.	**04**
Week 3 & Week 4	**Facebook & Instagram Marketing Campaign:** Plan and execute a marketing campaign on Facebook and Instagram, including creating engaging posts, running paid promotions, and analyzing results using insights.	**04**
Week 5 & Week 6	**Twitter, LinkedIn, Pinterest Optimization:** Optimize profiles on Twitter, LinkedIn, and Pinterest based on best practices.	**04**
Week 7 & Week 8	**Pinterest Board Creation and Optimization:** Create a Pinterest board for a specific business or topic, optimize it with relevant content, and implement strategies to enhance visibility.	**04**
Week 9 & Week 10	**YouTube Channel Creation:** Create a YouTube channel, upload a video, and optimize the channel for visibility. Discuss strategies for managing content effectively.	**04**
Week 11 & Week 12	**Mobile Advertising Campaign:** Develop and run a mobile advertising campaign, considering key objectives, ad formats, and targeting options. Evaluate the campaign's performance on both mobile sites and apps. **Social Media Marketing Strategy Development:** Develop a comprehensive social media marketing strategy, including audience identification, platform selection, content planning, and paid advertising strategies.	**04**

Week 13 & Week 14	**Social Media Analytics Application:** Use analytics tools (e.g., Facebook) to analyze key metrics for a social media campaign. Evaluate the effectiveness of the campaign and propose improvements. **Instagram Business Account Integration:** Connect an Instagram Business Account to Facebook, explore analytics, and analyze key engagement metrics.	**04**
Week 15	**Hootsuite Analytics Practice:** Explore Hootsuite Analytics features, generate reports on engagement and trends, and demonstrate social listening capabilities.	**02**
Pedagogy:	• Course delivery pattern, evaluation scheme, prerequisite shall be discussed at the beginning. • Conduct group activities to encourage collaboration and the exchange of ideas among students. • Practical Hands-On Sessions • Assign practical tasks related to creating and managing social media accounts, running campaigns, and analyzing results.	
References/ Readings:	**Main Reading:** 1. Dave Chaffey & Fiona Ellis-Chadwick, Digital Marketing: Strategy, Implementation and Practice, Pearson Education 2. Linda Coles Adams Media (2015). *Marketing with Social Media.* Adams Media. First Edition. 3. Sameer Deshpande, Nancy R. Lee. (2013). *Social Marketing in India.* Sage Response. First Edition. **Additional Reading:** 1. Dan Zarrella, (2009). *The Social Media Marketing Book.* O'Reilly. First Edition. 2. Lon Safko, The Social Media Bible: Tactics, Tools, & Strategies for Business Success, Brilliance Audio; Unabridged edition	
Course Outcomes:	**On completion of the course, students will be able to:** **CO1.** Understand social media marketing and analytics, the various channels through which it operates, and its role in marketing strategy. **CO2.** Develop effective ways of creating social media marketing strategy **CO3.** Analyze a Video Marketing Strategy and learn YouTube Advertising. **CO4.** Design Facebook Ads and Instagram Ads and understand how to effectively brand their Social Media Pages.	

Name of the Programme: Bachelor of Computer Applications

Course Code: CSA 324

Title of the Course: E- Commerce Applications

Number of Credits: 4 (3T +1P)

From AY: 2024-25

Pre-requisites For the Course:	None	
Course Objectives:	1. To understand the basic concept of e-commerce 2. To develop an understanding of Web-based Commerce 3. To understand marketing strategies for an online business 4. To equip students to assess e-commerce requirements of a business	
Units	**Content**	**No of hours 75 (45T+30P)**
I	**Introduction to Electronic Commerce and Application of E-commerce** • Meaning, Nature and scope of e-commerce, History of e-commerce, Business applications of e-commerce, E-Commerce Models: - (B2B, B2C, C2C, B2G), Advantages and Disadvantages of e-commerce, Applications of M-Commerce E-Commerce Web-sites as marketplace, Role of web site in B2C e-commerce, Web site design principles, Alternative methods of customer communication such as e-mail. • Applications of E-commerce Applications of e-commerce to Supply chain management Applications of e-commerce to Customer Relationship Management, Product and service digitization, Remote servicing	**15**
II	**Online Marketing and Business to Consumer E-Commerce Applications** • Online marketing and advertising, Push and pull approaches, Web counters, Web advertisements, Content marketing, Need of Digital Marketing for an e-commerce Business, Search Engine Optimization (SEO), Search Engine Marketing (SEM), Social Media Marketing (SMM), Web Analytics • Cataloging, Order planning and order generation, Cost estimation and pricing, Order receipt and accounting, Order selection and prioritization, Order scheduling, Order fulfilling, Order delivery, Order billing, Post sales service	**15**

III	**Business to Business E-Commerce , Electronic Payment System and Security Issues in E-Commerce** • Need and Models of B2B e-commerce, Using public and private computer networks for B2B trading; EDI and paperless trading, Characteristic features of EDI service arrangement, EDI architecture and standards, Reasons for slow acceptability of EDI , Value Added Networks • Types of payment systems, credit cards, debit cards, mobile wallets, Electronic Fund Transfer (EFT), Operational credit and legal risk of e-payment, Risk management options for e-payment systems • Risks of e-commerce, Types and sources of threats to e-commerce ; Protecting electronic commerce assets and intellectual property, Firewalls, Client server network security, Security tools, Digital identity and electronic signature; Risk management approach to e-commerce security	**15**
IV	**Practical Work.**	**30 Hours**
Week 1 & Week 2	• Case study to understand e commerce model • Practical on understanding the process of registering a business on the marketplace, listing your catalog.	**4**
Week 3 & Week 4	• Implement retargeting techniques.	**4**
Week 5 to Week 7	• Understanding implementing email advertising. • Understanding and implementing video advertisement, reels, story creation and other visual advertisement strategies.	**6**
Week 8 & Week 9	• Use different Tools for SEO (on page and off page) • Case study on different tools	**4**
Week 10 & Week 11	• Implement different types of Content marketing strategies.	**4**
Week 12 & Week 13	• Use Social media marketing platforms to market the products e.g. : facebook, LinkedIn, Instagram	**4**
Week 14 & Week 15	• Practical to use Web analytics tools e.g. Google Analytics, crazy egg • Implementing online payment for a website. • Case study on EDI model and understand various EDI message passing.	**4**

Pedagogy:	**Suggested strategies for use to accelerate the attainment of the various course outcomes.** 1. Lecture methods need not be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. 2. Lectures preferably to be conducted with the aid of multimedia projector, black board, group activities, charts, cases, etc. 3. Use of Case studies to illustrate concepts of Ecommerce 4. Introduce Topics in manifold representations. 5. Discuss how every concept is applied to the real world products 6. Assignment based on the course content may be given to the students to evaluate how learning of objectives was achieved.
References/ Readings:	**Main Reading:** 1. Agarwala, Kales N., Amity All Deeksha Agarwala (2000). *Business on the Net: An Introduction to the Whats and Hows of ECommerce.* Macmillan India Ltd. 2. Diwan, Prag and Sunil Sharma(2002). *Electronic Commerce- A Manager's Guide to EBusiness.* Vanity Books International Delhi. 3. Fitzerald (1998). *Business Data Communication Network.* McGraw Hill. **Additional Reading:** 1. Praveen Iyer (2020). *Electronic Data Interchange - edi made simple* Paperback
Course Outcomes :	**On completion of the course, students will be able to** **CO1** Recall the basics of e-commerce. **CO2** Understand the design principles of e-commerce websites and different models of e-commerce. **CO3** Apply the marketing strategies for an online business **CO4** Analyze the modern ways of doing e-commerce and threats to e-commerce

Name of the Programme: Bachelor of Computer Applications
Course Code: CSA-325
Title of the Course: Modern Frameworks
Number of Credits: 4(3T + 1P)
Effective from AY: 2023-24

Pre-requisite for the Course:	Knowledge of web designing using HTML, CSS, JavaScript, fundamentals of web application development and database queries.	
Course Objectives:	1. To understand the Fundamentals of Modern Frameworks 2. To design modern web interfaces using Tailwind CSS and VueJS 3. To explore NoSQL Database Management with MongoDB 4. To Build a simple web application using Tailwind CSS, VueJS and MongoDB	
Units	**Content**	**No of hours 75 (45T + 30P)**
I	**Fundamentals of Modern Frameworks** • Introduction to modern frameworks • Types of framework architectures - monolithic, microservices, serverless, three-tier, Model–view–controller (MVC), Client-side and Server-side features. **Microservice Architecture** • Microservice Characteristics • Understanding Microservices • Microservice Architecture • Adopting Microservices • Issues with monolithic architecture • REST Architecture principles • Microservice Transaction Management.	**10**
II	**Tailwind CSS Framework** Introduction to utility-first CSS framework • Features of Tailwind CSS • Tailwind CSS installation with CLI • @tailwind directive • CSS layout • CSS Flexbox • CSS Grid • CSS effects and filters • CSS Transitions and Animation	**20**

	• CSS Transforms • CSS Interactivity **VueJS Framework** • Introduction to Vue.js • Advantages of using Vue.js • Understanding the Vue.js ecosystem • Setting up a development environment • Virtual DOM • Data Binding • Understanding Vue instance and data • Vue directives and event handling • Conditional rendering and loops • Vue components and props • Routing with Vue Router • Creating and managing forms • Handling user input with v-model • Validating form data • Consuming APIs with Vue.js	
III	**Introduction to NoSQL Database** • NoSQL Databases • Difference between RDBMS and NoSQL • Benefits of NoSQL • JSON Introduction • JSON Structure **Introduction to MongoDB** • History of MongoDB, • Node Packaged Modules (npm), Installing MongoDB Locally, The Mongo Shell- Shell Collection Methods, MongoDB Database Commands **MongoDB query language** • CRUD (Creating, Reading & Updating Data) Mongo Shell • Query Operators • Update Operators and a Few Commands • Aggregation pipeline • Map-Reduce **MongoDB Cloud** • MongoDB Atlas (or any other platform)	**15**

	• The Developer Data Platform • Creating and Deploying an Cluster (Atlas or any other)	
IV	**Practical Work**	**30**
Week 1 & week 2	• **Setting up a Tailwind CSS Project** In this exercise, create a new web project and set up Tailwind CSS using the CLI. Utilize the @tailwind directive to integrate Tailwind into your HTML file and demonstrate basic utility-first styling principles. • **Building Responsive Layouts with Tailwind CSS** Design a responsive web page layout using Tailwind CSS, incorporating Flexbox and Grid to create a visually appealing and adaptive interface suitable for various screen sizes.	**04**
Week 3 & week 4	• **Implementing CSS Transitions and Animation with Tailwind** Enhance user experience by adding smooth transitions and animations to different elements of your webpage using Tailwind CSS. Experiment with various transition and animation classes provided by Tailwind. • **Introduction to Vue.js and Vue Instance** Set up a Vue.js project, create a Vue instance, and explore the basics of data binding. Display dynamic content on the webpage by manipulating data properties within the Vue instance.	**04**
Week 5 & week 6	• **Vue.js Directives and Event Handling** Implement Vue directives such as v-bind and v-on to handle events and dynamically update the DOM. Create interactive elements that respond to user actions through Vue.js. • **Routing with Vue Router** Integrate Vue Router into your Vue.js project to enable navigation between different views or pages. Define routes, create navigation links, and demonstrate the seamless transition between components.	**04**
Week 7 to week 9	• **Creating Vue.js Components and Props** Build modular and reusable components in Vue.js, passing data between them using props. Create a simple application with multiple components to demonstrate the power of Vue.js components. • **Form Handling and Validation in Vue.js** Develop a form in Vue.js, implement two-way data binding using v-model, and introduce form validation techniques. Ensure that user input is processed and validated effectively within the Vue.js framework.	**06**

	• **Consuming APIs with Vue.js** Fetch data from an external API using Vue.js and display it dynamically on your webpage. Explore the lifecycle hooks provided by Vue.js to manage the API request and response cycle.	
Week 10 & week 11	• **Introduction to NoSQL and JSON** Understand the basics of NoSQL databases and JSON data structure. Create a sample JSON document. • **MongoDB CRUD Operations** Install MongoDB locally, interact with the Mongo Shell, and perform CRUD operations (Create, Read, Update, Delete) on a MongoDB database. Practice inserting, querying, updating, and deleting documents. • **MongoDB Query Operators** Explore various query operators in MongoDB, such as $eq, $gt, $lt, etc. Build queries that retrieve specific data from a collection based on different criteria using these operators.	**04**
Week 12	• **Aggregation Pipeline in MongoDB** Dive into MongoDB's aggregation pipeline and construct complex queries that involve stages like $match, $group, $sort, and $project. Understand how to perform data transformations and aggregations in MongoDB. • **MongoDB Cloud Platform (Atlas or any other)** Sign up for the platform, create a new cluster, and deploy it. Configure the connection to your local MongoDB instance and explore the features provided by MongoDB cloud platform for managing databases in the cloud. Explore features of MongoDB cloud platform, such as data backups, scaling, and monitoring.	**02**
Week 13 to week 15	• **Building a Web Application** Create a simple web application integrating Tailwind CSS for styling, Vue.js for dynamic web interface, and MongoDB cloud platform for cloud data storage.	**06**
Pedagogy:	**Suggested strategies for use to accelerate the attainment of the various course outcomes.** 1. The lecture method need not be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. You may use a. Video/Animation to explain various concepts. b. Collaborative, Peer, Flipped Learning, etc. 2. Ask at least three HOT (Higher-Order Thinking) questions in the class, which promotes critical thinking. 3. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills,	

	and develops design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it. 4. Show the different ways to solve the same problem and encourage the students to come up with their own creative ways to solve them. 5. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding 6. To promote self-learning, give at least one assignment where they can complete one MOOCs (certificate or equivalent) course out of lecture hour. Test their understanding through quizzes or presentations.
References / Readings:	**Main Reading** 1. Callum Macrae (2018). *Vue.js: Up and Running.* O'Reilly Publication. 2. Kristina Chodorow (2014). *MongoDB – The Definitive Guide (2nd Edition).* O'Reilly Publication 3. Noel Rappin (2021). *Modern CSS with Tailwind: Flexible Styling without the Fuss. ISBN-13: 978-1680508185.* The Pragmatic Programmers Publication. **Additional Reading** 1. Nicholas Cloud (2019). JavaScript Frameworks for Modern Web Development. APRESS Publication. 2. Sam Newman(2021). Building Microservices: Designing Fine-grained Systems(2nd Edition). O'Reilly Publication
Course Outcomes:	**On completion of the course, students will be able to:** **CO1.** Understand modern framework fundamental concepts. **CO2.** Apply Tailwind CSS for Stylish Web Design and VueJS for creating modern web interfaces. **CO3.** Manage Data Effectively with NoSQL database MongoDB. **CO4.** Design web applications using Tailwind CSS, VueJS and MongoDB.

Fourth Year - Semester VII

Name of the Programme: Bachelor of Computer Applications
Course Code: CSA-400
Title of the Course: Statistical Tools
Number of Credits: 4 (3T +1P)
Effective from AY: 2024-25

Prerequisite for the Course :	None	
Course Objectives :	1. To understand principles of sampling in data collection 2. To learn the techniques of estimation 3. To test hypothesis in problem solving 4. To apply the regression techniques in solving real life problems (Case Studies)	
UNIT	**Content**	**No of Hours 75 (45T + 30P)**
I	**SAMPLING AND SAMPLING DISTRIBUTIONS** Principles of Sampling, Sampling methods, Sampling Distributions: mean, difference and proportions **ESTIMATION AND CONFIDENCE INTERVALS** Point Estimation, properties and drawback, Confidence Interval Estimation of population mean and proportions	**15**
II	**HYPOTHESIS TESTING** General Procedure, Errors in Hypothesis Testing, testing related to parametric test like Z test, t –test, nonparametric statistics: advantages and limitations, the Chi-Square Distribution, applications of Chi-Square Test Statistic, Mann Whitney U-Test	**15**
III	**MULTIPLE REGRESSION ANALYSIS** Assumptions, the basics, testing the accuracy of models, robust regression: bootstrapping, reporting the regression results, regression with categorical data, dummy coding **ANALYSIS OF VARIANCE** One Way and Two-Way Classification, assumptions, logic of F Ratio, post hoc procedures and violations of test assumptions - Case Study related to the above discussed topics using R	**15**
IV	**Practical Work**	**30**
Week 1	Getting Started with R environment : downloading , installing , using scripts , R workspace, installing packages in R	**2**
Week 2	Getting data into R workspace : creating variables, creating data frames , organizing data	**2**
Week 3	Manipulating Data : selecting parts of a data frame , data frames and matrices	**2**

Week 4	Exploring data with graphs in R	**2**
Week 5	Exploring the assumptions of normality in R	**2**
Week 6	Understanding Interval Estimation in R	**2**
Week 7	Parametric and Non-Parametric Tests in R	**2**
Week 8 & week 9	Testing the Regression models for accuracy	**4**
Week 10 & week 11	Comparing means Using ANOVA	**4**
Week 12 to week 15	Case Studies	**8**
Pedagogy:	**Suggested strategies to use to accelerate the attainment of the various course outcomes:** 1. Lecture methods need not be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. You may use a. Video/Animation to explain various concepts. b. Collaborative, Peer, Flipped Learning etc. 2. Ask at least three HOT (Higher-order Thinking) questions in the class, which promotes critical thinking. 3. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, develop design thinking skills such as the ability to design, evaluate, generalize, and analyse information rather than simply recall it. 4. Introduce Topics in manifold representations. 5. Show the different ways to solve the same problem and encourage the students to come up with their own creative ways to solve them. 6. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding 7. To promote self-learning give atleast one assignment (equivalent to 50% assignment weightage) where they can complete atleast one MOOCs (certificate or equivalent) course out of lecture hour. Test their understanding through quizzes or presentations. 8. One internal practical exam will be conducted as a part of internal evaluation. 9. Practical shall be performed in the laboratory as indicated in the syllabus. 10. A Hand written Hard Copy (soft copy) of the journal shall be maintained clearly mentioning the name of the experiment and other required information.	
References:	**Main Reading :** 1. Douglas C. Montgomery.(2006) *Introduction to Linear Regression Analysis.* Wiley india.3rd Edition. 2. Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani.(2013). A*n Introduction to Statistical Learning: with Applications in R.* Springer.1st Edition.	

	3. P. J. Bickel and K. A. Docksum. (2015). *Statistical Inference.* Prentice Hall. 2nd edition **Additional Reading :** 1. Andrie de Vries, Joris Meys (2006). *R Programming for Dummies, Wiley;* Second edition. *2.* Torsten Hothorn, Brian S. Everitt (2009)*.A Handbook of Statistical Analyses Using R,* Second Edition , Chapman and Hall/CRC.
Course Outcomes:	**At the end of the course, the students will be able to : –** **CO1.** Demonstrate a thorough understanding of the principles of sampling in data collection **CO2.** Explain the concept of estimation and confidence intervals **CO3.** Perform hypothesis testing **CO4.** Develop competence in utilizing regression techniques to address real-life problems through case studies.

Name of the Programme: Bachelor of Computer Applications
Course Code: CSA-401
Title of the Course: DevOps
Number of Credits: 4 (3P + 1 Tutorial)
Effective from AY: 2024-25

Pre-requisites for the Course:	Knowledge of OS, Programming, Networks and Web Development.	
Course Objectives:	1. To learn Git fundamentals and version control. 2. To install and utilise Docker for containerization and establish a Continuous Integration pipeline using Jenkins. 3. To understand the Configuration Management using Ansible, Infrastructure as Code (IaC) principles with Terraform. 4. To equip participants with the knowledge and skills to proficiently set up local Kubernetes clusters and deploy applications.	
Units & Weeks	**Content (Practical)**	**No of hours 105 (90P + 15 Tutorials)**
Tutorial Session Instructions	1. Tutorial lecture of 1 hour duration to be conducted each week. 2. Concepts needed for the conduct of Practical Sessions to be discussed. 3. These sessions may also be utilized for doubt clearance	
I	**Introduction to Version Control**	**28**
Week 01	• The session this week is to be conducted as classroom teaching not in the lab to discuss the concepts mentioned below. • From 2nd week onwards the sessions will be conducted in the lab setup in batches. **Introduction to DevOps** • DevOps Principles in detail • DevOps Engineer Skills in the market • Knowing DevOps Delivery Pipeline • Market trend of DevOps • DevOps Technical Challenges • Tools we use in DevOps **Introduction to Version Control** • Version Numbering(Major,Minor & Patch) • Semantic Versioning (SemVer) • Revision Control • Branching and Merging • Rollback • Changelog • Release Notes	**07**

Week 02	**Git Basics** • Essentials of Git in industry and in DevOps. • Install and configure Git • Set up a local repository • Perform basic Git commands (add, commit, push, pull) • Working with various commands in Git • Recording changes to the Repository • Working with Remotes Repositories	**07**
Week 03 & 04	**Git Branching and Merging** • Basic in Branching and Merging • Branch Management in GIT • Branching Workflows and its usage • Remote Branches – create and delete • Rebasing • Resolve merge conflicts	**14**
II	**Containerization and Continuous Integration**	**35**
Week 05 to 07	**Docker Basics** • Install Docker and create Docker images • Pull a pre-built Docker image from Docker Hub. Run the image and explore its contents • Write a simple Dockerfile to build a custom image. Run the image and verify that it works as expected • Run and manage Docker containers • Learn how to use Docker volumes to persist data between container restarts **Docker Compose** • Define multi-container applications using Docker Compose	**21**
Week 08	**Jenkins for Continuous Integration** • Install and configure Jenkins • Create a basic Jenkins job for continuous integration	**07**
Week 09	**Jenkins Pipeline** • Create a simple Jenkins Pipeline for a sample application • Explore scripted and declarative pipeline syntax • Use Jenkins plugins to integrate your pipeline with Git and Docker	**07**
III	**Configuration Management & Infrastructure as Code**	**28**
Week 10 to 13	**Ansible Playbooks** • Write Ansible playbooks to configure a sample environment • Use Ansible variables and loops to manage multiple servers with one playbook. • Create Ansible roles to modularize your playbooks and make them reusable. **Terraform Basics** • Write Terraform scripts to provision and manage infrastructure	**28**

IV	**Orchestration and Deployment**	**14**
Week 14 & 15	**Kubernetes Basics** • Set up a local Kubernetes cluster • Deploy and manage applications on Kubernetes	**14**
Pedagogy:	**Suggested strategies for use to accelerate the attainment of the various course outcomes.** 1. Lecture methods need not be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. You may use a. Video/Animation to explain various concepts. b. Collaborative, Peer, Flipped Learning etc. 2. Ask at least three HOT (Higher-order Thinking) questions in the class, which promotes critical thinking. 3. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, develop design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it. 4. Introduce Topics in manifold representations. 5. Show the different ways to solve the same problem and encourage the students to come up with their own creative ways to solve them. 6. Discuss how various concepts can be applied to the real world - and when that's possible, it helps improve the students' understanding 7. To promote self-learning, give atleast one assignment where they can complete at least one MOOCs (certificate or equivalent) course out of lecture hour. Test their understanding through quizzes or presentations. 8. One internal practical exam will be conducted as a part of internal evaluation. 9. Practical shall be performed in the laboratory as indicated in the syllabus. 10. A Hand written Hard Copy (or digital copy) of the journal shall be maintained clearly mentioning the name of the experiment and other required information.	
References/ Readings:	**Main Reading:** 1. Jenkins, M. K. (2019). Learning DevOps: Jenkins, Kubernetes, Terraform, Azure DevOps. Packt Publishing Limited. 2. Joakim Verona (2016). Practical DevOps. Packt Publishing Limited. **Additional Reading:** 1. Hornbeek,M.(2019).Engineering DevOps:From Chaos to Continuous Improvement... and Beyond. Book Baby. 2. Kim,G.,Humble,J.,Deoise,P.,Wills,J.(2016).The DevOps Hand Book: How to Create World-Class Agility, Reliability, and Security in Technology Organizations. IT Revolution Press.	

Course Outcomes:	On completion of the course, students will be able to: **CO1.** Understand the concepts & fundamentals of using DevOps tools **CO2.** Apply DevOps tools for application development under different phases. **CO3.** Set up local Kubernetes clusters and effectively deploy applications. **CO4.** Analyze the implementation and use of all DevOps tools for the phases of software development.

Name of the Programme: Bachelor of Computer Applications
Course Code: CSA 402
Title of the Course: Software Design Patterns
Number of Credits: 4 (3T + 1P)
Effective from AY: 2024-25

Pre-requisites for the Course:	Object-Oriented Concepts	
Course Objectives:	1. To understand patterns related to object-oriented design. 2. To apply the design patterns that are common in software applications. 3. To analyze a software development problem and evaluate alternatives. 4. To create a module structure to solve a problem.	
Units	**Content**	**No of hours 75 (45T+30P)**
I	**Introduction** • Definition- Design Pattern • Describing Design Patterns • Discussion on composition v/s inheritance • basic rules of OO design • The Catalog of Design Patterns • Organizing the Catalog • How Design Patterns Solve Design Problems • How to Select a Design Pattern • How to Use a Design Pattern	**15**
II	**Creational Patterns** • Factory Method • Singleton • Prototype • Abstract Factory **Structural Pattern** • Adapter • Decorator • Façade • Proxy	**15**
III	**Behavioral Patterns** • Chain of Responsibility • State • Strategy • Command • Observer **Architectural Patterns** • Pipe & Filter • layered • MVC/MVVM	**15**

IV	**Practical:** 1. Use of an object-oriented programming language for the concepts learnt in the units from I to III are required to be implemented practically. 2. The broad area of practical problems are mentioned below. 3. Most of the design pattern is to be covered during practical sessions as mentioned below. 4. The rest of the design patterns to be given to the students to implement as part of their assignments.	**Practical Hours (30)**
Week 1	Write a program to implement the following concepts: a. Method overriding, b. Interface c. Abstract class.	**02**
Week 2 to 5	Write programs to implement Creational Patterns.	**08**
Week 6 to 9	Write programs to implement Structural Patterns.	**08**
Week 10 to 13	Write programs to implement Behavioral Patterns. (Any 4)	**08**
Week 14 to 15	Write programs to implement Architectural Patterns - MVC & MVVM.	**04**
Pedagogy:	Suggested strategies for use to accelerate the attainment of the various course outcomes. 1. The lecture method need not be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. You may use a. Video/Animation to explain various concepts. b. Collaborative, Peer, Flipped Learning, etc. 2. Ask at least three HOT (Higher-Order Thinking) questions in the class, which promotes critical thinking. 3. Adopt Problem-Based Learning (PBL), which fosters students' Analytical skills, and develops design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it. 4. Introduce Topics in manifold representations. 5. Show the different ways to solve the same problem and encourage the students to come up with their own creative ways to solve them. 6. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding 7. To promote self-learning, give at least one assignment where they can complete one MOOCs (certificate or equivalent) course out of lecture hour. Test their understanding through quizzes or presentations.	

References/ Readings:	**Main Reading:** 1. Freeman, E., Freeman, E., Bates, B., & Sierra, K. (2004). *Head First Design Patterns*. Shroff. 2. Gamma, E. (2015). Design Patterns. Pearson Education. 3. Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (1994). Design Patterns: Elements of Reusable Object-Oriented Software. Wesley. 4. Shalloway, A. (2006). Design Patterns Explained: A New Perspective on Object-Oriented Design (Software Patterns Series). Pearson Education. **Additional Reading:** 1. Buschmann, F. (1996). *Pattern-Oriented Software Architecture - A System of Patterns V 1 (Wiley Software Patterns Series)*. Wiley. 2. Mark Grand, JAVA Enterprise Design Patterns, Wiley DreamTech, Vol
Course Outcomes:	**On completion of the course, students will be able to-** **CO1.** Recall basic concepts of design patterns and its types **CO2.** Understand design patterns, types, and where to apply them **CO3.** Apply the design patterns that are common in software applications conceptually as well as practically. **CO4.** Analyze and justify the suitability of design patterns for the given problem and conceptually as well as its implementation.

Name of the Programme: Bachelor of Computer Applications
Course Code: CSA-403
Title of the Course: Natural Language Processing
Number of Credits: 4 (3T + 1P)
Effective from AY: 2024-25

Pre-requisites for the Course:	1. Knowledge of standard concepts in artificial intelligence. 2. Basic familiarity with logic and probability. 3. Adequate experience with programming. 4. Knowledge of using Python libraries.	
Course Objectives:	1. Understand the fundamental concepts and ideas in Natural Language Processing (NLP). 2. To be familiar with natural language processing methods and tools. 3. Understanding both the algorithms available for processing linguistic information and the underlying computational properties of natural languages. 4. Apply NLP techniques to real-world problems and datasets, and gain hands-on experience in implementing and evaluating NLP models.	
Unit	**Content**	**No. of Hours 75 (45T + 30P)**
I	**Introduction to NLP** What is NLP?, NLP vs. Computational Linguistics. Levels of Linguistic Representation, Morphology, Lexical Analysis, Syntax, Semantics, Pragmatics and Discourse. Introduction to Machine Learning and Deep Learning The evaluation of NLP applications **NLP Applications** Machine Translation, Question Answering and Information Retrieval, Chatbots, and Dialogue Systems, Automatic Speech Recognition and Text-to-Speech	**15**
II	**NLP Algorithms** Regular Expressions, Text Normalization, Edit Distance, N-gram Language Models, Naive Bayes and Sentiment Classification, Logistic Regression, Vector Semantics and Embeddings, Neural Networks and Neural Language Models, Sequence Labelling for Parts of Speech and Named Entities, RNNs and Transformers and Pretrained Language Models, Fine-tuning and Masked Language Models, Prompting and Instruct Tuning.	**15**

III	**Annotating Linguistic Structure** Context-Free Grammar and Constituency Parsing, Dependency Parsing, Logical Representations of Sentence Meaning, Computational Semantics and Semantic Parsing, Relation and Event Extraction, Time and Temporal Reasoning, Word Senses and WordNet, Semantic Role Labelling and Argument Structure, Lexicons for Sentiment, Affect, and Connotation, Coreference Resolution, Discourse Coherence, Phonetics	**15**
IV	**The broad area of practical problems are mentioned below.**	**(30)**
week 1 to Week 4	• NLTK, Python 3, and the Jupyter Notebook similar IDE, Introduction to Keras, or the Natural Language Toolkit in Python for basic text processing tasks. • Perform tokenization, stemming, and lemmatization on a given text dataset. Handle common text preprocessing tasks, such as removing stop words, punctuation, and special characters. • Train a basic language model (e.g., n-gram model) and generate text based on the learned language model.	**8**
Week 5 to Week 9	• Train word embeddings using Word2Vec or GloVe on a small corpus. Utilize pre-trained word embeddings and explore semantic relationships between words. • APIs for Social Media Web Scraping, Implement a text classification task (e.g., sentiment analysis) using a machine learning algorithm (e.g., Naive Bayes, SVM) and evaluate its performance. • Build a simple named entity recognition model using a pre-trained model or a custom model on a labelled dataset.	**10**
Week 10 to Week 15	• Implement a part-of-speech tagging system using a rule-based or machine-learning approach. • Build a basic information retrieval system using techniques like TF-IDF and evaluate its effectiveness on a dataset. • Fine-tune a pre-trained BERT model on a specific NLP task, such as text classification or named entity recognition. • Create a text generation model using recurrent neural networks (RNNs) or transformers and generate coherent text based on a given prompt.	**12**

Pedagogy	Suggested strategies to use to accelerate the attainment of the various course outcomes. 1. The lecture method need not be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. You may use a. Video/Animation to explain various concepts. b. Collaborative, Peer, Flipped Learning etc. 2. Ask at least three HOT (Higher-order Thinking) questions in the class, which promotes critical thinking. 3. Adopt Problem-Based Learning (PBL), which fosters students' Analytical skills, and develops design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it. 4. Introduce Topics in manifold representations. 5. Show the different ways to solve the same problem and encourage the students to come up with creative ways to solve them. 6. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding 7. To promote self-learning give at least one assignment where they can complete at least one MOOCs (certificate or equivalent) course out of lecture hour. Test their understanding through quizzes or presentations. 8. One assignment in the form of a mini-project collecting data and using analytic tools may be given to the students.
References / Readings	1. Allen, J. (1995). *Natural language understanding.* Benjamin-Cummings Publishing Co., Inc. 2. Bird, S., Klein, E., & Loper, E. (2009). *Natural language processing with Python: analyzing text with the natural language toolkit.* O'Reilly Media. 3. Eisenstein, J. (2019). *Introduction to natural language processing.* MIT press 4. Jurafsky, Dan and Martin, James. (2008). *Speech and Language Processing, Second Edition.* Prentice Hall. 5. McEnery, T. (2019). *Corpus linguistics*. Edinburgh University Press.
Course Outcomes	**At the end of course students will be able to:** **CO1.** Define fundamental concepts in NLP, including tokenization, stemming, lemmatization, and syntactic and semantic analysis. **CO2.** Interpret and compare representing and encoding language using various techniques such as bag-of-words, TF-IDF, and word embeddings. **CO3.** Use the necessary tricks for making their models work on practical problems. **CO4.** Connect NLP techniques to real-world problems and datasets, demonstrating the ability to choose appropriate methods and evaluate model performance.

Name of the Programme: Bachelor of Computer Applications
Course Code: CSA 411
Title of the Course: Project Management
Number of Credits: 4 (3T + 1P)
Effective from AY: 2024-25

Pre-requisites:	None	
Course Objectives:	1. To remember Project management concepts 2. To understand organizing a project 3. To apply project management concepts and quality management concepts 4. To analyze the use of appropriate Project Management Tools for documentation of the project.	
Units	**Content**	**No of hours 75 (45T +30P)**
I	**Foundations of Project Management**: -The Context of Project Management, The Project Life Cycle in the context of IT **Project Management Process Stages & The Project Plan-** Initiation, Planning, Execution,Monitoring and Controlling, Closing (Project Selection and Approval, Project Charter and Detailed Plan, Project Planning Framework, Project's Scope, Budget and Schedule) **The Project Team :**The Role of the Project Manager**,** Team Selection and Acquisition**,** The Project Environment and Team Performance	**15**
II	**Defining and Managing Project Scope**:- Project Scope Management Process**,** Scope Planning**,** Project Scope Verification**,** Scope Change Control **The Work Breakdown Structure and Project Estimation:-**Developing the Work Breakdown Structure, Deliverables and Milestones**.** Project Estimation Techniques**,** Software Engineering Metrics and Approaches**.** **The Project Schedule, Budget and Risk Management:-** Developing the Project Schedule and Budget**,** Gantt and PERT Charts**,** Project Network Diagrams**,** Critical Path Analysis**,** Project Management Software Tools. Identifying IT Project Risks**,** Risk Strategies**,** Risk Monitoring and Control	**15**
III	**Project Communication, Tracking, and Reporting**:- The Project Communication Plan, Project Metrics, Reporting Performance and Progress **IT Project Quality Management:-** Quality Tools**,** Quality Systems (ISO, Six Sigma, CMMI) **Project Implementation and Evaluation**, Project Procurement, Outsourcing, Project Implementation Methods, Project Evaluation, Literature review.	**15**

IV	List of Practicals	Practical Hours (30)
Week 1 & 2	**Gantt Charts**- Project Tasks BreakDown, duration on each task, assignment to each task, task dependencies, meeting approvals and deadlines, work progression, full project schedules	**04**
Week 3 & 4	**Network Diagram**:- Drawing network to represent project, finding critical path, arrow diagrams for project analysis . (Based on Case Study)	**04**
Week 5 & 6	**Kanban Board (Agile Board) :-** Mapping of workflow, using swim lanes, creating sub teams, creating a project development and procurement board.	**04**
Week 7 & 8	**Time Sheets**: Creating a work schedule, assigning task to employees, tracking of employee work hours , reviewing and approving timesheets , sharing of time sheets and work load with stakeholders	**04**
Week 9 to 11	**Project Dashboards** for Activity Tracking (deadlines and resource availability), risk status, financials , strategic alignment (business objectives and key results), change requests, time tracking and budget, resource estimates, project deliverables and milestones (Based on Case Study)	**06**
Week 12 & 13	**Stakeholder Mapping**:- Creating a database of stakeholders, creating a grid map, determining level of involvement, connecting stakeholders.(Case Study Based)	**04**
Week 14 & 15	Project Management Documentation for a Case Study	**04**
Pedagogy:	**Suggested strategies for use to accelerate the attainment of the various course outcomes.** 1. The lecture method need not be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. You may use a. Video/Animation to explain various concepts. b. Collaborative, Peer, Flipped Learning, etc. 2. Ask at least three HOT (Higher-Order Thinking) questions in the class, which promotes critical thinking. 3. Adopt Problem-Based Learning (PBL), which fosters students' Analytical skills, and develops design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it. 4. Introduce Topics in manifold representations. 5. Show the different ways to solve the same problem and encourage the students to come up with their own creative ways to solve them. 6. Discuss how every concept can be applied to the real world project - and when that's possible, it helps improve the students' understanding 7. To promote self-learning, give at least one assignment where they can complete one MOOCs (certificate or equivalent) course out of lecture hours. 8. Test their understanding through quizzes or presentations.	
References/ Readings:	**Main Reading:** 1. Marchewka, J. (2018). *Information Technology Project Management (3rd ed.)*. Wiley. 2. Schwalbe, K. (2018). *Information Technology Project Management (6th ed.).* Course Technology.	

	Additional Reading: 1. Ashfaque Ahmed (2012). *Software Project Management: A Process-Driven Approach*. CRC Press. Taylor & Francis Group.
Course Outcomes:	**On completion of the course, students will be able to:** **CO1.** Remember Project Management Concepts. **CO2.** Understand organizing a project. **CO3.** Apply project management and quality management concepts in the context of IT. **CO4.** Analyze the use of appropriate Project Management Tools for documentation of the project.

Name of the Programme: Bachelor of Computer Applications
Course Code: CSA-412
Title of the Course: Dashboard Development
Number of Credits: 4 (3T+1P)
Effective from AY: 2024-25

Pre-requisites for the Course:	Basic knowledge of data visualization concepts	
Course Objectives:	1. To understand the concept of data visualization for dashboard design. 2. To apply the design principles for developing effective dashboards. 3. To create custom interactive dashboards for organizations. 4. To develop and deploy dashboards for web applications.	
Units	**Content**	**No. of hours 45 (45T + 30P)**
I	**Introduction to Data Visualisation and dashboard design** • Definition of data visualisation, Principles of effective data visualisation, characteristics of data visualisation, types of data visualisation techniques, tools used for data visualisation. **Dashboard design and its principles** • Introduction to dashboards, architecture of dashboards, uses of dashboards • Principles of dashboard design, methods for selecting the right dashboard • Techniques and tools for dashboard development • Dashboard Content	**15**
II	**Dashboard development process** • Dashboard development process and dashboard models • Different aspects of dashboards • Technologies used for dashboard development **Roles and responsibilities in dashboard development** • Roles and responsibilities **Variations in use of dashboards** • Typical dashboard data • Characteristics of a well designed dashboard	**15**
III	**Dashboard creation, deployment and maintenance** • Power of visual perception • Principles of visual perception • Key goals in visual design process **Select appropriate display media** • Design dashboards for usability • Create interactive dashboards	**15**

	• Deployment and maintenance **Dashboards development for web applications** • Case studies	
IV	**List of Practicals**	**30 Hours**
Week 1	Practical exercise to analyse data using any data visualization tool tableau	**2**
Week 2	Create a static dashboard	**2**
Week 3	Create a dashboard to track Key Performance Indicators	**2**
Week 4 & Week 5	Dashboard of graphs and charts	**4**
Week 6 & Week 7	Design a dashboard for web using templates	**4**
Week 8 & Week 9	Design Custom dashboard	**4**
Week 10 to Week 12	Create dynamic dashboard	**6**
Week 13 to Week 15	Develop interactive dashboard	**6**
Pedagogy:	**Suggested strategies for use to accelerate the attainment of the various course outcomes.** 1. The lecture method need not be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. You may use a. Video/Animation to explain various concepts. b. Collaborative, Peer, Flipped Learning, etc. 2. Ask at least three HOT (Higher-Order Thinking) questions in the class, which promotes critical thinking. 3. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, and develops design thinking skills such as the ability to design, evaluate, generalize, and analyze information to develop dashboards. 4. Introduce Topics in manifold representations. 5. Show the different ways to solve the same problem and encourage the students to come up with their own creative ways to solve them. 6. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding 7. To promote self-learning, give at least one assignment where they can complete one MOOCs (certificate or equivalent) course out of lecture hour. Test their understanding through quizzes or presentations.	

References/ Readings:	**Main Reading:** 1. Few, Stephen. (2006). *Information dashboard design: The effective visual communication of data*. O'Reilly Media, Inc.. 2. Staron, M. (2015). *Dashboard development guide How to build sustainable and useful dashboards to support software development and maintenance.* 3. Steve Wexler, Jeffrey Shaffer and Andy Cotgreave. (2017). The Big Book of Dashboards: Visualizing Your Data Using Real-World Business Scenarios. **Additional Reading:** 1. *Elias Dabbas. Interactive Dashboards and Data Apps 2. with Plotly and Dash.* 2. *Nathan Yau. Visualize This: The Flowing Data Guide to Design, Visualization, and Statistics.*
Course Outcomes:	**On completion of the course, students will be able to:** **CO1.** Demonstrate a comprehensive understanding of data visualization concepts for dashboard design. **CO2.** Apply design principles to develop visually effective dashboards. **CO3.** Independently create custom interactive dashboards tailored to organizational needs. **CO4.** Successfully develop and deploy dashboards for web applications.

Name of the Programme: Bachelor of Computer Applications
Course Code: CSA - 413
Title of the Course: Introduction to Quantum Computing
Number of Credits: 4 (3T + 1P)
Effective from AY: 2024-25

Pre-requisites for the Course:	Basic Knowledge of Boolean Algebra, Data Structures, Computational Complexity, and Algorithm Analysis	
Course Description	The feasibility of quantum computers remains uncertain, but if they become a reality, they will revolutionize computational methods and have profound effects on various applications, including communication and computer security. Despite the uncertainty, it is still fascinating to explore the principles of quantum computing.	
Course Objectives:	1. To introduce students to the fast-growing field of quantum computing 2. To create an understanding of the differences between quantum bits and classical bits 3. To familiarize with the basic quantum logical operations and algorithms 4. To provide an initial overview of quantum computing, emphasizing the shift in paradigm from classical computing and introducing fundamental quantum algorithms. 5. To equip students with future-proof skills, enable them to tackle complex problems, enhance critical thinking abilities, and promote interdisciplinary learning.	
Unit	**Content**	**No. of hours 75 45T + 30P**
I	**Introduction to Quantum Computing** One Quantum Bit 1. Superposition- superposition, complex numbers 2. Measurement- measurement in Z-basis, normalization, measurement on other basis, consecutive measurements 3. Bloch Sphere Mapping- global and relative phases, Bloch sphere 4. Physical qubits 5. Quantum Gates- linear maps, classical reversible gates, common one-qubit quantum gates, General one-qubit gates 6. Quantum Circuits **Linear Algebra** 1. Quantum States- Column and row vectors 2. Inner Products- Inner products, Orthonormality, Projection, Measurement, Change of basis 3. Quantum Gates- Gates as matrices, Common one-qubit gates as matrices, sequential quantum gates, Circuit identities, Unitarity, Reversibility 4. Outer Products- Outer products, Completeness relation	**15**

II	**Multiple Quantum Bits** 1. States and Measurement- Tensor product, Kronecker product, Measuring individual qubits, sequential single-qubit measurements 2. Entanglement- Product states, Entangled states 3. Quantum Gates- One-qubit quantum gates, Two-qubit quantum gates, Toffoli gate 4. No-cloning theorem 5. Quantum Adders- Classical adders, Converting classical adder to quantum gate, Quantum setup, Quantum sum, Quantum carry, Quantum ripple-carry adder, Circuit complexity, Adding in Superposition 6. Universal Quantum Gates- Definition, Components, Examples, Solovay-Kitaev theorem 7. Quantum Error Correction- Decoherence, Bit-flip code, Phase-flip code, Shor code	**15**
III	**Entanglement and Quantum Protocol** 1. Measurements- Product states, Maximally entangled states, Partially entangled states 2. Bell Inequalities- ERP paradox and local hidden variables, Bell inequalities and the CHSH inequality, Quantum processor experiment, No-signaling principle 3. Monogamy and Entanglement- Classical correlations, Quantum entanglement 4. Superdense Coding- The problem, Classical solution, Quantum solution 5. Quantum Teleportation- The problem, Classical solution, Quantum Solution 6. Quantum Key Distribution- Encryption, Classical solution: public key cryptography, Quantum solution: BB84	**15**
IV	**List of Practicals**	**Practical Hours (30)**
Week 1 to week 3	Parity- The problem, Classical solution, Quantum solution: Deutsch's Algorithm. Constant vs Balanced Functions- The problem, Classical solution, Quantum solution: Deutsch-Jozsa Algorithm	**06**
Week 4 to week 6	Secret Dot Product String- The problem, Classical solution, Quantum solution: Bernstein-Vazirani Algorithm, Recursive problem Secret XOR Mask: The problem, Classical solution, Quantum solution: Simon's Algorithm	**06**
Week 7 to week 9	Brute-Force Searching: The problem, Classical solution, Quantum solution: Grover's Algorithm Discrete Fourier Transform(DFT)- An Application, Classical solution(DFT) and Quantum solution(QFT)	**06**

Week 10 to week 15	Eigenvalue Estimation- The problem, Classical solution, Quantum solution Period of Modular Exponentiation- The problem, Classical solution, Quantum solution Factoring- The problem, Classical solution, Quantum solution(Shor's Algorithm)	**12**
References	**Main Reading** 1. Bernhardt, C. (2019). *Quantum computing for everyone*. Mit Press. 2. Hidary, J. D., & Hidary, J. D. (2019). *Quantum computing: an applied approach* (Vol. 1). Cham: Springer. 3. Nielsen, M. A., & Chuang, I. L. (2010). *Quantum computation and quantum information*. Cambridge university press. **Additional Reading:** 1. Nielsen, M. A. (2005). Cluster-state quantum computation. 2. Sutor, R. S. (2019). *Dancing with Qubits: How quantum computing works and how it can change the world*. Packt Publishing Ltd.	
Course Outcomes	**On completion of the course, students will be able to –** **CO1.** Recall the basic concepts and characteristics of classical and quantum computing systems **CO2.** Understand the characteristics of classical & quantum computing systems and quantum algorithms. **CO3.** Describe systems with qubits. **CO4.** Perform basic quantum computing operations and quantum Fourier transform.	

Fourth Year - Semester VIII

Name of the Programme: Bachelor of Computer Applications
Course Code: CSA-404
Title of the Course: Introduction to Functional Programming
Number of Credits: 4 (3T +1P)
Effective from AY: 2024-25

Pre-requisites	• Basic Programming Knowledge • Basic understanding of mathematical concepts like functions and algebra.	
Course Objectives	1. To understand the basics of lambda calculus 2. To study the concepts of functional programming 3. To learn the applications of modules, arrays and trees in functional programming 4. To apply the concepts to solve practical problems	
Unit	**Content**	**No. of Hours 75 (45T+ 30P)**
I	**Introduction to Lambda Calculus** • Motivation and Historical Context • Syntax of Lambda Calculus • Beta Reduction • Alpha and Eta reduction **Advanced Concepts in Lambda Calculus** • Currying and Partial Application • Fix Point Combinator • Combinatory Logic • Types and Typing Rules	**15**
II	**Functional Programming Concepts** • Computation as rewriting • Polymorphism • Higher-order functions: Map, Filter and Fold • Recursive functions: tail and general recursion. • Pattern Matching for function definition • Guards for conditional expressions • Measuring Efficiency • Infinite Lists • Conditional Polymorphism • Defining functions in GHCii	**15**
III	**Datatypes, Modules** • User-defined data types • Abstract data types • Recursive data types • Modules **Arrays, I/O** • Arrays • Sorting	**15**

	• Input/ Output **Search trees** • Binary search tree • Balanced binary search tree	
IV	**Practical**	**30**
Week 1 & Week2	**Haskell Programming** • Introduction to Haskell Programming • Running Haskell Program • Basic Syntax: variables, functions, expressions	4
Week3 & Week4	**Basics of Haskell Programming** • Define and use functions • Basic data types, List and tuples • Currying	4
Week5 to Week9	**Functional Programming Concepts** • Computation as rewriting • Polymorphism • Higher-order functions: Map, Filter and Fold • Recursive functions: tail and general recursion. • Pattern Matching for function definition • Guards for conditional expressions	10
Week10 to Week13	**Array and Applications** • Array • Sorting • Using Infinite lists • Conditional Polymorphism • Defining functions in GHCi	8
Week 14 & Week15	**Datatypes, Modules** • User-defined data types • Abstract data types • Recursive data types • Modules	4
Pedagogy	1. Lectures will be conducted with the aid of multimedia projector, black board, etc. 2. Implement the concepts of functional programming using IDE like Visual Studio Code	
Textbooks/ Reference Books	**Main Reading:** 1. Revised Edition. (1985) .*The Lambda Calculus, Its Syntax and Semantics (Studies in Logic and the Foundations of Mathematics, Volume 103).* North-Holland. 2. Simon Peyton Jones.(1987). *The Implementation of Functional Programming Languages.* Prentice-Hall. **Additional Reading:** 1. Hindley, J. R., & Seldin, J. P. (2008). *Lambda-calculus and combinators: an introduction*. Cambridge University Press. 2. Hutton, G. (2016). *Programming in haskell*. Cambridge University Press.	

Course Outcomes	**On completion of the course, students should be able to** **CO1.** Recall the basics of lambda calculus **CO2.** Understand the concepts of functional programming **CO3.** Apply advanced concepts of functional programming like Higher order functions, conditional polymorphism, etc **CO4.** Implement concepts of modules, arrays, sorting in functional programming

Name of the Programme: Bachelor of Computer Applications
Course Code: CSA 405
Title of the Course: Information Systems Audit
Number of Credits: 4 (3T+1P)
Effective from AY: 2024-25

Pre-requisites for the Course:	The student should have basic knowledge of computer technology.	
Course Objectives:	1. To know the importance of the Information System Audit Process. 2. To review the nature and demand of audits as well as the need for control and audit of computer-based IS. 3. To assess the risk analysis to facilitate risk-based audits. 4. To analyze the process of audit reporting and follow-ups.	
Units	**Content**	**No of hours 75 (45T + 30P)**
I	**Information System Audit:** Role of Information System (IS) in Organization, Concept of Information Audit. **IS Audit Function Knowledge:** What is Information System Management? Understanding the Organization's Business Processes, Establishing the Needs of implementing IS, Identifying Key Activities, Establish Performance Objectives, Decide the Control Strategies, Implement and Monitor the Controls, Executive Management's Responsibility and Corporate Governance, Audit Role, Relationship of Internal IS Audit to the External Auditor, Relationship of IS Audit to Other Company Audit Activities.	**15**
II	**IS Risk and Fundamental Auditing Concepts:** Business process, Business Applications, Business Risk Assessment, Computer Risks and Exposures, Effect of Risk, Audit Evidence. **Internal Controls Concepts Knowledge:** Internal Controls, Elements of Internal Control, Manual and Automated Systems, Control Procedures, Control Objectives and Risks. **Application Controls:** What is application control, What is the relationship between application controls and general controls, Why rely on application controls, How to scope a risk-based application control review, What are the steps to conduct an application controls review.	**15**

	Risk Management of the IS Function: Nature of Risk, Elements of Risk Analysis, Computer System Threats, Risk Management.	
III	**Information System Audit:** IS auditor's role in review of application controls (AI, Data warehouse, EFT, Point of Sale, DSS, ERP, etc.) Computer Assisted Audit Techniques (CAATS).User controls, Database controls and Preparation of IS audit programme. Practical aspects of validation of reports from business application software. **Audit Reporting:** Regulations pertaining to IS audit, IS audit report format.	**15**
IV	**Practical Work** The concepts learned in the units from I to III are required to be implemented practically. Use of Open Source software to be used for the practicals.	**Practical Hours (30)**
Week 1	Case Studies on the Information Audit Process. Discuss and analyze real-world cases where the information audit process played a crucial role in identifying and mitigating risks. Tools: Document management systems, audit trail tools.	**04**
Week 2	Discuss and Analyse Information System Auditing Process, Key Aspects Enterprise Governance of Information and Technology,Information Systems Acquisition, Development and Implementation, Business case and Feasibility Analysis.	**04**
Week 3	Preparation of IS Audit Program and Online Footprints Analysis,Discuss the importance and components of an IS audit program,Guide students in preparing a comprehensive IS audit program for a hypothetical scenario,Emphasize the role of the audit program in identifying and addressing security risks,Tools: Document management systems, template tools for IS audit program.	**04**
Week 4	To understand and apply Information Systems Operations, Maintenance and Support, End User Computing, Protection of Information Assets, Privacy Principles, Factor of Authentication, Virtual Private Network	**04**

Week 5	Demonstrate the use of network scanning, finding the open ports, finding the vulnerability from the network(Use of Nmap scanning tool or any other similar software tool),Log Management and Analysis, Tools**:** These tools collect, store, and analyze log data from various sources to identify suspicious or anomalous activities. Examples include Splunk, ELK Stack (Elasticsearch, Logstash, Kibana), and Graylog.	**04**
Week 6	To identify and understand website vulnerability (Use of Netcat Tool or any other similar software tool)	**04**
Week 7	To understand and apply Brute Force Techniques to check the login portal's security. To demonstrate tools for retrieving information of organization website (Use of OWASP ZAP tool or any other similar software tool)	**04**
Week 8	To demonstrate the payload and the remote process. (Use of Metasploit tool or any other similar software tool). Demonstration of a web-based information system and to check its vulnerability (Use of Burp Suite tool or any other similar software tool)	**02**
Pedagogy:	Suggested strategies to use to accelerate the attainment of the various course outcomes. 1. The lecture method need not be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. You may use: a. Video/Animation to explain various concepts b. collaborative, peer, flipped learning etc. 2. Adopt Problem-Based Learning (PBL), which fosters students' Analytical skills, and develops design thinking skills such as the ability to design, evaluate, generalize, and analyse information rather than simply recall it. 3. Show the different ways to solve the same problem and encourage the students to come up with creative ways to solve them. 4. Discuss how every concept can be applied to the real world. 5. Assignments based on the course content shall be given to the student and evaluated at regular intervals. 6. Experiments to be performed in the laboratory as suggested in the syllabus.	

References/ Readings:	**Main Reading:** 1. Cascarino, R. E. (2007). Auditor's Guide to Information Systems Auditing (Standard Edition 2007). John Wiley & Sons. 2. Christine Bellino, Steve Hunt (2007).Global Technology Audit Guide 8: Auditing Application Controls. The IIA Research Foundation. 3. Hemang Doshi, Hiral Patel. (2022).The Beginner's Guide to Information System Audit. Amazon Asia Pacific Holdings Private Limited. **Additional Reading:** 1. Weber, R. (1999). Information Systems Control and Audit. Prentice Hall. 2. Weber, R. (1988). EDP Auditing: Conceptual Foundations and Practice (2nd ed.). Tata McGraw Hill.
Course Outcomes:	**On completion of the course the student will be able to:** **CO1.** Recall the concepts of the Information Audit System. **CO2.** Understand the different types of Information System Audits **CO3.** Apply an audit strategy for Information Systems based on risk management. **CO4.** Analyze Information Systems audit tools and techniques.

Name of the Programme: Bachelor of Computer Applications
Course Code: CSA 406
Title of the Course: Internet of Things
Number of Credits: 4 (3T + 1P)
Effective from AY: 2024-25

Pre-requisites for the Course:	The student should have basic knowledge of computer architecture and networking.	
Course Objectives:	1. Understand the basic concepts of IoT and its applications 2. Manipulate sensors/actuators. 3. Implement IoT Projects.	
Units	**Content**	**No of hours 75 (45T+30P)**
I	**Introduction to IoT** • Overview of Internet of Things (IoT) • Characteristics of devices and applications in IoT ecosystem, • Building blocks of IoT • Technologies making up IoT ecosystem • IoT levels, IoT design methodology • Physical Design/Logical Design of IoT • Functional blocks of IoT and Communication Models. **Controlled Systems and Connectivity Models** • Working of Controlled Systems • Real-time systems with feedback loop (e.g., thermostat in refrigerator, AC, etc.) • Connectivity models – TCP/IP versus OSI model • Different type of models using wired and wireless methodology • Process flow of an IoT application	**15**
II	**Sensors, Actuators, and Microcontrollers** • Sensor - Measuring physical quantities in digital world (e.g., light sensor, moisture sensor, temperature sensor, etc.) • Actuator – moving or controlling system (e.g., DC motor) • Different type of actuators Controller – Role of microcontroller as gateway to interfacing sensors and actuators • Microcontroller vs Microprocessor • Types of Microcontrollers in Embedded Ecosystem **Embedded Programming Basics and Control Structures** • Embedded Programming Language- Basics • Variables and Identifiers • Built-in Data Types • Arithmetic Operators and Expressions • Constants and Literals, Assignment. • Conditional Statements, Loops	**15**

	• Decision making using Relational Operators, Logical Connectives, If-else statement • Loops: while loop, do while, for loop, Nested loops, Infinite loops, Switch statement	
III	**Embedded Programming** **Interfacing Sensors, Functions, and Practical Implementations** • Arrays – Declaring and manipulating single dimension arrays • Functions - Standard Library of C functions in Arduino IDE • Prototype of a function: Formal parameter list • Return Type • Function call • Interfacing sensors – The working of digital versus analog pins in Arduino platform • Interfacing LED, Button, Sensors-DHT, LDR, MQ135. • Display the data on Liquid Crystal Display (LCD) • Interfacing keypad • Serial communication – interfacing HC-05 (Bluetooth module) Control/handle 220v AC supply – interfacing relay module.	**15**
IV	**List of practicals** Using embedded C programming language , the concepts learned in the units from I to III are required to be implemented practically. The broad area of practical problems is mentioned/ suggested below.	**Practical Hours (30)**
Week 1 to week 5	• Design a simple IoT system using Arduino and sensors to monitor environmental conditions and communicate data to a centralized server. • Develop an IoT application that controls the temperature of a simulated environment using a thermostat and provides real-time feedback. • Interface a light sensor and a DC motor with an Arduino microcontroller to create a system that responds to changes in light intensity.	**10**
Week 6 to week 10	• Write an embedded "C" program that utilizes conditional statements and loops to control the behavior of an LED based on input from a button. • Create an Arduino program that interfaces with a DHT sensor to measure temperature and humidity. Use functions to display the data on an LCD. • Implement a system that uses arrays to store and manipulate sensor data from multiple sensors. Create functions to perform specific operations on the array.	**10**
Week 11 to week 15	• Develop a system that uses a relay module to control a 220v AC device (e.g., a light bulb) based on sensor input. Ensure safety measures are implemented. • **Mini project:** Create a small IOT based project using the concepts learnt in previous weeks	**10**

Pedagogy	1. The lecture method need not be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. You may use i. Video/Animation to explain various concepts. ii. Collaborative, Peer, Flipped Learning, etc. 2. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding. 3. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, and develops design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it. 4. Show the different ways to solve the same problem and encourage the i. students to come up with their own creative ways to solve them. 5. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding.
References/ Readings:	**Main Reading** 1. Arshdeep Bahga, Vijay Madisetti. (2014) I*nternet Of Things: A Hands-On Approach.* Publisher Arshdeep Bahga & Vijay Madisetti 2. Olivier Hersent and David Boswarthick. (2012) *Internet Of Things: Key Applications and Protocols.* John Wiley & Sons Limited 3. Raj Kamal. (2017). *Internet of Things Architecture and Design Principles.* Mc Graw Hill India **Additional Reading:** 1. F. John Dian.(2022*)Fundamentals of Internet of Things: For Students and Professionals.*Wiley-IEEE Press 2. Vinay Chowdary, Abhinav Sharma, Naveen Kumar, Vivek Kaundal(2023) *Internet of Things in Modern Computing Theory and Applications.*CRC Press
Course Outcomes:	**On completion of the course, students will be able to:** **CO1.** Remember the characteristics of IOT , functional blocks, process flow , components and its uses. **CO2.** Understand the basic concepts of IoT and how to interface sensors and actuators with the microcontroller Arduino platform. **CO3.** Develop IoT based applications using Arduino or Raspberry Pi.

Name of the Programme: Bachelor of Computer Applications
Course Code: CSA - 407
Title of the Course: Research Methodologies
Number of Credits: 4 (3T+1P)
Effective from AY: 2024-25

Pre-requisites for the Course:	The student should have basic knowledge of Statistics.	
Course Objectives:	1. To identify characteristics of scientific method, foundations of research, problem identification and problem formulation. 2. To understand the design concepts for qualitative, quantitative research, and concepts of measurements. 3. To apply concepts of research reporting/publishing. 4. To use statistical techniques/tools for data analysis.	
Units	**Content**	**No of hours 75 (45T+30P)**
I	**Foundations of Research:** Meaning, Objectives, Motivation, Utility. Concept of theory, empiricism, deductive and inductive theory; Characteristics of scientific method, Understanding the language of research – Concept, Construct, Definition, Variable. Problem Identification & Formulation, Research Question, Investigation Question – Measurement Issues. **Research Design:** Concept and Importance in Research – Features of a good research design; Exploratory Research Design – concept, types and uses; Descriptive Research Designs – concept, types and uses; Experimental Design: Concept of Independent & Dependent variables. **Qualitative and Quantitative Research:** Concept, Approach and Application: Qualitative research & Quantitative research examples and problems.	**15**

II	**Data Collection Methods:** Collection of Primary Data, Observation Method, Interview Method, Questionnaires, Schedules, Other Methods of Data Collection, Collection of Secondary Data, Case study method. **Measurement:** Concept of measurement– What is measured? Problems in measurement in research – Validity and Reliability. Levels of measurement – Nominal, Ordinal, Interval, Ratio**.** **Processing and Analysis of Data:** Processing operations, Elements/ types of analysis, Statistics in research-measures of central tendency or statistical averages, measures of dispersion, measures of asymmetry (skewness), measures of relationship, Simple regression analysis, Multiple correlation and regression, Partial correlation, Association in case of attributes.	**15**
III	**Hypothesis**: Qualities of a good Hypothesis –Null Hypothesis & Alternative Hypothesis, procedure for hypothesis testing, flow diagram, Test of hypothesis, procedure for hypothesis testing, Hypothesis for means, difference between means, comparing two related samples, proportions, difference between proportions, comparing a variance to some hypothesized population variance, power of test. **Chi-square test:** $\chi 2$ test and their applications in research studies. **Analysis of variance:** Basic principles of ANOVA, ANOVA technique, setting up of analysis of variance table, one way, ANOVA, two way ANOVA. **Research Reporting:** Scientific Writing Structure and components of Scientific Reports – types of Report – Technical Reports and Thesis – Significance – Different steps in the preparation – Layout, structure and Language of typical reports - Illustrations and tables – Bibliography, Referencing and footnotes	**15**
IV	**Practical work** The broad area of practical problems is to be taken from the following two heads:	**Practical Hours (30)**
Week 1 to week 8	**I.** Data Analysis using statistical tools: Data Preparation – Univariate analysis (frequency tables, bar charts, pie charts, percentages), Bivariate analysis – Cross tabulations and Chi-square test including testing hypothesis of association. Interpretation of Data and results.	16

Week 9 to week 15	**II.** Paper Writing – Layout of a Research Paper, Software for paper formatting like LaTeX/MS Office. ✓ Explore Journals in Computer Science, Impact factor of Journals, When and where to publish? UGC Care List, Scopus Indexed, Web of Science. ✓ Explore ethical issues related to publishing, Plagiarism and Self-Plagiarism. ✓ Explore softwares for detection of Plagiarism. ✓ Use of Encyclopedias, Research Guides, Handbook etc., Academic ✓ Databases for Computer Science Discipline. ✓ Use of tools / techniques for Research: methods to search required information effectively, Reference Management Software like Zotero/Mendeley	14
Pedagogy:	**Suggested strategies for use to accelerate the attainment of the various course outcomes.** 1. The lecture method need not be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. You may use a. Video/Animation to explain various concepts. b. Collaborative, Peer, Flipped Learning, etc. 2. Ask at least three HOT (Higher-Order Thinking) questions in the class, which promotes critical thinking. 3. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, and develops design thinking skills such as the ability to design, evaluate, generalize, & analyze information rather than simply recall it. 4. Introduce Topics in manifold representations. 5. Show the different ways to solve the same problem and encourage the students to come up with their own creative ways to solve them. 6. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding 7. To promote self-learning give at least one assignment where they can complete one MOOCs (certificate or equivalent) course out of lecture hour. Test their understanding through quizzes or presentations.	

References/ Readings:	**Main Reading:** 1. Jain, R. K. (2021). Research Methodology: Methods and Techniques (5th ed). Vayu Education of India. 2. Kothari, C. R. (2004). Research Methodology (2nd ed.). New Age International Publishers. **Additional Reading:** 1. Panneer Selvam. (2007). Research Methodology. PHI Learning Pvt. Ltd.
Course Outcomes:	**On completion of the course, students will be able to:** **CO1.** Recall the characteristics of scientific method, foundations of research, research process and design. **CO2.** Understand the design concepts for qualitative, quantitative research, and concepts of measurements. **CO3.** Apply concepts of research reporting/publishing. **CO4.** Use statistical techniques to analyze data.

Name of the Programme: Bachelor of Computer Applications
Course Code: CSA - 414
Title of the Course: Interactive Media
Number of Credits: 4(3T + 1P)
Effective from AY: 2023-24

Pre-requisites for the Course:	None	
Course Objectives:	1. To learn interactive digital media concepts 2. To understand to conceptualize, plan, and execute creative ideas using necessary tools/technology 3. To apply appropriate learning and skills to create new digital and interactive media content. 4. To create digital content for multimedia.	
Units	**Content**	**No of hours 75 (45T + 30P)**
I	**Interactive Digital Media:** Introduction, Forms of Interactive Digital Media ,Interactive Digital Media vs Other Forms of Media, Developing Interactive Digital Media, Essential Skills for the Interactive Digital Media Developer, The Impact of Interactive Digital Media, Career Opportunities in Interactive Digital Media **Fundamental Components of Interactive Digital Media:** Introduction, Analog vs. Digital Media, Bits and Bytes, File Formats, Analog to Digital, The Pros of Digital Media, Compression, Description vs. Command-Based Encoding of Media, Color on the Screen	**15**
II	**Media Content:** Introduction, Graphics, Pixel-based Images, Vector-based Images, 2D Animation, 3D Graphics and Animation, Audio, Video & Text in Interactive Digital Media. **Aesthetics in Interactive Digital Media:** Introduction, Typography, Color Theory, Design Principles, and Layout and Visualization.	**15**
III	**Authoring Interactive Digital Media:** Introduction, Multimedia Authoring, Making Video Games: Casual and Console, Building Apps, Building Interactive Media for Performance and Public Spaces, Building Websites **Usability:** Introduction, Importance of good usability, Guidelines for Good Usability, Usability and Play Testing	**15**
IV	**List of Practicals**	**Practical Hours (30)**

Week 1 & week 2	● **Interactive multimedia presentations and Story Boarding** :- create engaging presentations using interactive power point features ● Use of Text Content, intended graphics , Audio/voice over/ Music or sound effects, Animation, Video, user interface Design(Tile bars, navigation buttons, Position of text and graphics)	**4**
Week 3 to Week 5	● **Social media interaction design:-** Plan and execute a social media campaign with interactive content. ● Optimization of the campaign for better interaction after taking user feedback.	**6**
Week 6 & week7	● **Editing** : audio & Video Editing, Colour Correction , visual effects and exporting	**4**
Week 8 to Week 10	● **Video production** :- planning / capturing videos , pre production and post production and rendering	**6**
Week 11 & Week 12	● **Designing Interactive Interface**: for web based application .	**4**
Week 13 to Week15	● **Design Interactive interfaces** for mobile based applications	**6**
Pedagogy:	Suggested strategies for use to accelerate the attainment of the various course outcomes. 1. The lecture method need not be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. You may use a. Video/Animation to explain various concepts. b. Collaborative, Peer, Flipped Learning, etc. 2. Ask at least three HOT (Higher-Order Thinking) questions in the class, which promotes critical thinking. 3. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, and develops design thinking skills such as the ability to design, evaluate, generalize, & analyze information rather than simply recall it. 4. Introduce Topics in manifold representations. 5. Show the different ways to solve the same problem and encourage the students to come up with their own creative ways to solve them. 6. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding 7. To promote self-learning, give at least one assignment where they can complete one MOOCs (certificate or equivalent) course out of lecture hour. Test their understanding through quizzes or presentations.	

References/ Readings:	**Main Reading:** 1. Julia V. Griffey.(2020). *Introduction to Interactive Digital Media: Concept and Practice.* A Focal Press book. 2. Roy Rada,(2012) *A. Michailidis. Interactive Media.* Springer New York. 3. Thakur, D. (2005). *Interactive Multimedia: Concepts and Practices.* I.K. International Publishing House Pvt. Ltd **Additional Reading:** 1. Yue-Ling Wong.(2012). *Digital Media Primer.* Pearson Education.
Course Outcomes:	**On completion of the course, students will be able to:** **CO1.** Understand audio, video and animations that demonstrate both technical knowledge and design principles. **CO2.** Apply knowledge of software applications and tools/emerging technologies to create digital media **CO3.** Analyze and Implement critical thinking skills by solving challenges related to multimedia creation. **CO4.** Design user friendly interactive interfaces.

Name of the Programme: Bachelor of Computer Applications
Course Code: CSA-415
Title of the Course: Game Design
Number of Credits: 04 (3T+1P)
Effective from AY: 2024-25

Pre-requisites for the Course:	None	
Course Objectives:	1. To understand the different types of games and navigations 2. To become creative and competent to work with 2d characters and vector graphics. 3. To create storyboards, paper prototype of the game and design the document. 4. To understand the different UI Patterns.	
Units	**Content**	**No of hours 75 (45T + 30P)**
I	**Introduction to Game Design** ● Game development Different types of game and use cases - FPS, RPG, Racing, Fighting, Casual, Money, Spinner, Casino, Massively Multiplayer Online (MMO). ● Game Simulations. ● Adventure - Real Time Strategy (RTS) - Puzzle, Action - Stealth Shooter, Combat. ● Revert Settings - Launching Your First Project - Importing a Project - Switching Between Projects - Customizing the UI – Navigation - Manipulating Objects - Position Game Objects - Place Light Probes.	**15**
II	**Working With 2D** ● 2D characters ● Characters from Different Countries and Styles ● Asian characters vs. Western characters ● Making sprites ● Working with vector graphics. **2D Game Design Pipeline** ● The market - The audience - The platforms where to publish the game - The competitor - Define the story - Create timelines - Storyboards - Level Design - Game play mechanics - Costs of the game - Making and maintenance- Create a game design document.	**15**

III	**2D Environment and 2D Background** ● **2D environment** - Form and Shape, Anatomy and Proportions, Perspective, Breaking Down Color, Lighting and Shading. ● **2D background** - Form and Shape - Anatomy and Proportions Perspective - Breaking Down Color - Lighting and Shading - 2D Character Design – Primitives – Textures - creating face – expressions – anatomy - body parts - cartoon making. **Different UI Patterns** ● Introduction - UI and UX - What Does a Good UI Do? - Case study – Games - With Poor UIs bad and good cases - Success rates and compilation - Oblivion- case study - Far Cry 3 - case study - Mortal Kombat X- Case Study - Fight of the legends - case study. 2D Platformer – Build with assets.	**15**
IV	**List of suggested Practicals**	**Practical Hours (30)**
Week 1 to 3	• Create a design for a puzzle game.	**6**
Week 4 to 6	• Creating Storyboard for a racing game.	**6**
Week 7 to 9	• Create a prototype of a tic tac game.	**6**
Week 10 to 12	• Create a 2D toy character with suitable animation effects.	**6**
Week 13 to 15	• Create a test plan for testing a board game.	**6**
Pedagogy:	**Suggested strategies for use to accelerate the attainment of the various course outcomes.** 1. The lecture method need not be only a traditional lecture method, but alternative, effective teaching methods could be adopted to attain the outcomes. You may use a. Video/Animation to explain various concepts. b. Collaborative, Peer, Flipped Learning, etc. 2. Ask at least three HOT (Higher-Order Thinking) questions in the class, which promotes critical thinking. 3. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, and develops design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it. 4. Introduce Topics in manifold representations. 5. Show the different ways to solve the same problem and encourage the	

	students to come up with their own creative ways to solve them. 6. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding 7. To promote self-learning, give at least one assignment (equivalent to 50% assignment weightage) where they can complete one MOOCs (certificate or equivalent) course out of lecture hour. Test their understanding through quizzes or presentations.
References/ Readings:	**Main Readings:** 1. Adams. (2015). *Fundamentals of Game Design.* Third edition, New Riders Publication. 2. Alan Thorn. (2007). *Introduction to Game Programming with C++.* BPB Publications, First Edition. 3. Chris Solarski. (2012). *Drawing Basics and Video Game Art: Classic to Cutting-Edge Art Techniques for Winning Video Game Design.* First Edition, Watson – Guptill Publication. **Additional Readings:** 1. Crawford, C. *The Art of Computer Game Design*. Berkeley, California: Osborne/McGraw-Hill. 2. Gibson, J. *Introduction to Game Design, Prototyping, and Development: From Concept to Playable Game—with Unity® and C#*. 3. Rouse III, R. *Game Design: Theory & Practice* (2nd ed.). Illustrations by S. Ogden. Foreword by N. Falstein.
Course Outcomes:	**On completion of the course, students will be able to:** **CO1.** Understanding gaming concepts and different gaming components. **CO2.** Demonstrate the flow of 2D game designing. **CO3.** Applying 2D environment and background for designing Characters. **CO4.** Preparing different case studies on UI patterns.

Name of the Programme: Bachelor of Computer Applications
Course Code: CSA 416
Title of the Course: Educational Technology
Number of Credits:4 (3T+1P)
Effective from AY: 2024-25

Prerequisite for the Course :	The student should have basic knowledge of the use of computer technology.	
Course Objectives :	1. To understand the basic concepts of Educational Technology, Systems Approach to planning lessons and instructional materials 2. To understand the concept, functions and elements of Communication, types of instructional media and materials 3. To apply the knowledge to develop skill in preparing and using different instructional materials.	
UNIT	**Content**	**No of hours 75 (45T +30P)**
I	**Introduction to Educational Technology** **a. Understanding Educational Technology:** Meaning and definition of Educational Technology; Objectives of Educational Technology; Types of Educational Technology - Teaching Technology, Behavioural Technology, Instructional Technology; Components of Educational Technology - Hardware approach and Software approach. **b. System approach in Educational Technology:** Definition of System Approach, Components of Educational Instructional System, Stages of System Approach in Teaching, Importance of system approach, Role of teacher in system approach. **c. Classroom Applications of Educational Technologies**: Survey of educational hardware and software. Technology in Education: Meaning, Evolution and Development Traditional Educational Technology/Materials. ICT in Education: Computer, Internet, Multimedia/Hypermedia.	**15**

II	**Communication and Learning Experiences** **a. Communication Process:** Definition, Characteristics and Importance of Communication; Communication Cycle; Principles of Communication; Classroom communication -Verbal and Non-Verbal communication; Factors affecting Classroom Communication; Barriers to effective Classroom Communication and methods to overcome these barriers; Flanders Interaction Analysis Categories System (FIACS). **b. Edgar Dale's Cone of Experiences:** Direct and Purposeful (Games & Experiments), Contrived Experiences (Three Dimensional, Mock up, Diorama), Dramatised Experiences (Pageant, Socio-Drama), Demonstration Boards (chalkboard, peg board, flipped classroom and MOOCs), Field Trips, Exhibits, Still Pictures (drawings, graphs, cartoon etc.)	**15**
III	**Trends in Educational Technology:** **a. Teaching Aids:** Principles of selecting Teaching Aids; Types of Teaching Aids - Non-Projected (Chalkboards Electronic Marker Boards, Flip charts, Dioramas, flipped classroom), Projected (Slide Projector, Film Strips, Epidiascope, Microfilm), Audio-Visual Materials (Motion picture, Videos); Online tools (LMS, MOOCs, Content Creation and Collaboration, Assessment and Feedback, Virtual Classroom, Web Conferencing, Interactive Whiteboard, Coding, Note Taking, Adaptive Learning Platforms). **b. Flipped Classroom:** Introduction, Types of Flipped learning: Standard Inverted Classroom, Group based Flipped Classroom, Debate-Focused Flipped Classroom, Discussion Focused Flipped Classroom, Micro-Flipped Classroom, Faux Flipped Classroom, Virtual Flipped Classroom, and Flipped Teacher Approach. Advantages and disadvantages of flipped Classroom**.** **c. Technology and Student Assessment:** Difference between Evaluation and Assessment, Types of Evaluation: Product, Process, Formative and Summative Assessment, Objective-based evaluation, Rubrics, Checklist, Blogs, Polls, Discussions, Quiz.	**15**

IV	**List of Practicals** The concepts learned in the units from I to III are required to be implemented practically.	**Practical Hours (30)**
Week 1 & week 2	Based on concepts and techniques learnt in Unit I (Multimedia/Hypermedia/Presentations)	04
Week 3 & week 4	Based on concepts and techniques learnt in Unit II (Verbal and Non-Verbal communication, Effective Communication)	04
Week 5 & week 6	Based on concepts and techniques learnt in Unit II (Games & Experiments, peg board).	04
Week 7 & week 8	Based on concepts and techniques learnt in Unit II(Digital Exhibits, Still Pictures)	04
Week 9 & week 10	Based on concepts and techniques learnt in Unit III (LMS, MOOCs)	04
Week 11 & week 12	Based on concepts and techniques learnt in Unit III (MOOCs, Rubrics, Checklist)	04
Week 13 & week 14	Based on concepts and techniques learnt in Unit III (Flipped Classroom,Blogs, Polls, Discussions, Quiz)	04
Week 15	Based on concepts and techniques learnt in Unit III (Quiz)	02

Pedagogy:

Suggested strategies for use to accelerate the attainment of the various course outcomes.

1. The lecture method need not be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. You may use
    a. Video/Animation to explain various concepts.
    b. Collaborative, Peer, Flipped Learning, etc.
2. Ask at least three HOT (Higher-Order Thinking) questions in the class, which promotes critical thinking.
3. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, and develops design thinking skills such as the ability to design, evaluate, generalize, & analyze information rather than simply recall it.
4. Introduce Topics in manifold representations.
5. Show the different ways to solve the same problem and encourage the students to come up with their own creative ways to solve them.
6. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding
7. To promote self-learning, give at least one assignment where they can complete one MOOCs (certificate or equivalent) course out of lecture hour. Test their understanding through quizzes or presentations.

References:	**Main Reading:** 1. Kanvaria, V. K. (2014). A comprehension on educational technology and ICT for education. New Delhi: GBO. 2. S.K. Mangal, Uma Mangal (2009). Essentials of Educational Technology. PHI Learning Private Limited, New Delhi. ISBN : 978-81-203-3727-7. **Additional Reading:** 3. Chetna Jathol, Sonal Chabra (2005). Educational Technology. Vikas publishing house. 4. Dr. Vikram Sharma, Dr. Amandeep Chaulia. Educational Technology & ICT. Iterative International Publishers (IIP). ISBN:9789393364180 5. Kanvaria, V. K. (2014). *A comprehension on educational technology and ICT for education.* New Delhi: GBO. 6. M. D. Roblyer, Aaron H. Doering, (February 25, 2012) *Student Value Edition 6th Edition. Integrating Educational Technology into Teaching.* Pearson. 7. Mishra, S. & Sharma, R.C. (eds) (2005). *Interactive Multimedia in Education and Training.* London: Idea Group Inc (IGI). 8. Roblyer, M.D. (2007). *Integrating Educational Technology into Teaching, (Edn 4).* Delhi: Pearson Education India. 9. Shelly Cashman Gunter,(2006), 2nd Edition. *Teachers Discovering Computers, Integrating Technology in the Classroom.*
Course Outcomes:	**On completion of the course, students will be able to:** **CO1.** Recall the concepts of Educational Technology, and its systems approach to planning lessons and instructional materials **CO2.** Understand the foundations of communication and the different types of instructional media and materials **CO3.** Apply the knowledge to develop skill in preparing and using different instructional materials.

Name of the Programme: Bachelor of Computer Applications
Course Code: CSA- 417
Title of the Course: Blockchain Technology
Number of Credits: 4 (3T + 1P)
Effective from AY: 2024-25

Pre-requisites for the Course:	The student should have basic knowledge of Information technology and Python Programming .	
Course Objectives:	1. To understand blockchain technology and its applications. 2. To demonstrate the implementation of blockchain solutions. 3. To apply insights of blockchain across applications.	
Units	**Content**	**No of hours 75 (45T + 30P)**
I	**Introduction to Blockchain Technology** • Overview of blockchain concepts, Decentralized systems, Consensus algorithms • Blockchain types: Public, Private, and Consortium • Cryptography fundamentals for blockchain **Blockchain Development Platforms and Tools** • Introduction to blockchain development frameworks (e.g. Ethereum, Hyperledger and Polygon) • Setting up the blockchain development environment	**10**
II	**Solidity Programming** • Introduction to Solidity, Solidity Syntax and Structure • Data Types and Variables, Control Structures, Functions and Modifiers, Mappings and Arrays, Error Handling • Smart Contract Deployment (Eg. Polygon Network) **Blockchain Security and Privacy** • Blockchain security challenges and attacks. • Cryptographic techniques for securing blockchain transactions. • Privacy and anonymity considerations in blockchain systems. • Types of Smart contract attacks.	**15**

III	**Decentralized Applications (DApps)** • Smart contract development and testing on DApp • Interacting with smart contracts using web interfaces and APIs • Building and deploying decentralized applications (DApps) **Blockchain Applications and Industry Use Cases** • Blockchain applications in finance, supply chain, healthcare, and other domains. • Regulatory and legal considerations for blockchain adoption. • Evaluating the potential impact of blockchain on various industries	**20**
IV	**List of Practicals** The concepts learned in the units from I to III are required to be implemented practically. The broad area of practical problems is mentioned/ suggested below.	**Practical Hours (30 Hours)**
Week 1 to week 5	• Set up a basic private blockchain using a platform like Hyperledger Fabric. Explore the consensus algorithms and configure a decentralized system. • Install and configure development environments for Ethereum OR Polygon. • Develop a simple smart contract in Solidity and deploy it on the Polygon network.	**10**
Week 6 to week 10	• Create a sample smart contract using Solidity, incorporating data types, control structures, and functions. • Implement cryptographic techniques in a smart contract to enhance security. • Create a smart contract practicing Self Destruction contract. • Develop a basic decentralized application (DApp) that interacts with a smart contract. Use web interfaces and APIs to showcase the functionality of the DApp.	**10**
Week 11 to week 15	• Explore real-world blockchain applications by developing a prototype for a specific industry (e.g., finance, supply chain). • Exploring Reentrancy attack on smart contract.	**10**

Pedagogy	1. The lecture method need not be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. You may use i. Video/Animation to explain various concepts. ii. Collaborative, Peer, Flipped Learning, etc. 2. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding. 3. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, and develops design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it. 4. Show the different ways to solve the same problem and encourage the students to come up with their own creative ways to solve them. 5. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding.
References/ Readings:	**Main Reading** 1. Bashir, I. (2020). *Mastering Blockchain: Unlocking the Power of Cryptocurrencies, Smart Contracts, and Decentralized Applications.* 2. Chris Dannen(2017). Introducing Ethereum and Solidity: Foundations of Cryptocurrency and Blockchain Programming for Beginners 3. Drescher, D. (2017). Blockchain Basics: A Non-Technical Introduction in 25 Steps. 4. Modi Ritesh(2022).Solidity Programming Essentials. **Additional Reading:** 1. Elad Elrom(2019).The Blockchain Developer 2. Jitendra Chittoda(2019).Mastering Blockchain Programming with Solidity.
Course Outcomes:	**On completion of the course, students will be able to:** **CO1.** Recall the underlying concepts and principles of blockchain technology. **CO2.** Understand the usage of blockchain applications using appropriate frameworks and tools. **CO3.** Apply smart contracts and decentralized applications (DApps) in blockchain development. **CO4.** Analyse the potential use cases and implications of blockchain technology.