Anekant Education Society's

Tuljaram Chaturchand College, Baramati

(Empowered Autonomous)

Four Year B. Sc. Degree Program in Computer Science

(Faculty of Science & Technology)

CBCS Syllabus

T. Y. B. Sc. (Computer Science) Semester – V

For Department of Computer Science

Tuljaram Chaturchand College, Baramati

Choice Based Credit System Syllabus (2023 Pattern)

(As Per NEP 2020)

To be implemented from Academic Year 2025-2026

Title of the Programme: T. Y. B. Sc. (Computer Science)

Preamble

AES's Tuljaram Chaturchand College has made the decision to change the syllabus of various faculties from June, 2023 by incorporating the guidelines and provisions outlined in the National Education Policy (NEP), 2020. The NEP envisions making education more holistic and effective and to lay emphasis on the integration of general (academic) education, vocational education and experiential learning. The NEP introduces holistic and multidisciplinary education that would help to develop intellectual, scientific, social, physical, emotional, ethical and moral capacities of the students. The NEP 2020 envisages flexible curricular structures and learning based outcome approach for the development of the students. By establishing a nationally accepted and internationally comparable credit structure and courses framework, the NEP 2020 aims to promote educational excellence, facilitate seamless academic mobility, and enhance the global competitiveness of Indian students. It fosters a system where educational achievements can be recognized and valued not only within the country but also in the international arena, expanding opportunities and opening doors for students to pursue their aspirations on a global scale.

In response to the rapid advancements in science and technology and the evolving approaches in various domains of Computer Science and related subjects, the Board of Studies in Computer Science at Tuljaram Chaturchand College, Baramati - Pune, has developed the Credit, Course Structure of T.Y.B.Sc. (Computer Science) Sem- V, VI and curriculum for the Fifth semester of T.Y.B.Sc.(Computer Science), which goes beyond traditional academic boundaries. The syllabus is aligned with the NEP 2020 guidelines to ensure that students receive an education that prepares them for the challenges and opportunities of the 21st century. This syllabus has been designed under the framework of the Choice Based Credit System (CBCS), taking into consideration the guidelines set forth by the National Education Policy (NEP) 2020, LOCF (UGC), NCrF, NHEQF, Prof. R.D. Kulkarni's Report, Government of Maharashtra's General Resolution dated 20th April and 16th May 2023, and the Circular issued by SPPU, Pune on 31st May 2023.

A degree in Computer Science subject equips students with the knowledge and skills necessary for a diverse range of fulfilling career paths. Career in Computer Science is considered one of the most high-paying jobs and is full of opportunities; particularly when India's prowess in information technology industry is recognized across the globe. The pool of talented computer professionals working in IT companies of the USA, Canada and other

countries shows that IT can take a person to higher levels. Numerous IT companies from India employ huge number of computer professionals in their Indian and overseas offices. Students who are interested in **programming,** software development, and have good analytical and reasoning skills may pursue this course. Job opportunities are available for Graduates and Post Graduates in Government as well as Private sector. Graduates may take up the following job posts- Software Engineer, Software Tester, Data Analyst, Project Manager, Network Administrator, Database Administrator and Application Developer.

Overall, revising the Computer Science syllabus in accordance with the NEP 2020 ensures that students receive an education that is relevant, comprehensive, and prepares them to navigate the dynamic and interconnected world of today. It equips them with the knowledge, skills, and competencies needed to contribute meaningfully to society and pursue their academic and professional goals in a rapidly changing global landscape.

Programme Outcomes (POs)
for
B.Sc. (Computer Science)

PO1. Comprehensive Knowledge and Understanding: Graduates will possess a profound understanding of their field of study, including foundational theories, principles, methodologies, and key concepts, within a broader multidisciplinary context.

PO2. Practical, Professional, and Procedural Knowledge: Graduates will acquire practical skills and expertise essential for professional tasks within their field. This includes knowledge of industry standards, best practices, regulations, and ethical considerations, with the ability to apply this knowledge effectively in real-world scenarios.

PO3. Entrepreneurial Mindset and Knowledge: Graduates will cultivate an entrepreneurial mindset, identifying opportunities, fostering innovation, and understanding business principles, market dynamics, and risk management strategies.

PO4. Specialized Skills and Competencies: Graduates will demonstrate proficiency in technical skills, analytical abilities, problem-solving, effective communication, and leadership, relevant to their field of study. They will also adapt and innovate in response to changing circumstances.

PO5. Capacity for Application, Problem-Solving, and Analytical Reasoning: Graduates will possess the capacity to apply learned concepts in practical settings, solve complex problems, and analyze data effectively. This requires critical thinking, creativity, adaptability, and a readiness to learn and take calculated risks.

PO6. Communication Skills and Collaboration: Graduates will effectively communicate complex information, both orally and in writing, using appropriate media and language. They will also collaborate effectively in diverse teams, demonstrating leadership qualities and facilitating cooperative efforts toward common goals.

PO7. Research-related Skills: Graduates will demonstrate observational and inquiry skills, formulate research questions, and utilize appropriate methodologies for data collection and analysis. They will also adhere to research ethics and effectively report research findings.

PO8. Learning How to Learn Skills: Graduates will acquire new knowledge and skills through self-directed learning, adapt to changing demands, and set and achieve goals independently.

PO9. Digital and Technological Skills: Graduates will demonstrate proficiency in using ICT, accessing information sources, and analyzing data using appropriate software.

PO10. Multicultural Competence, Inclusive Spirit, and Empathy: Graduates will engage effectively in multicultural settings, respecting diverse perspectives, leading diverse teams, and demonstrating empathy and understanding of others' perspectives and emotions.

PO11. Value Inculcation and Environmental Awareness: Graduates will embrace ethical and moral values, practice responsible citizenship, recognize and address ethical issues, and take appropriate actions to promote sustainability and environmental conservation.

PO12. Autonomy, Responsibility, and Accountability: Graduates will apply knowledge and skills independently, manage projects effectively, and demonstrate responsibility and accountability in work and learning contexts.

PO13. Community Engagement and Service: Graduates will actively participate in community-engaged services and activities, promoting societal well-being.

Programme Specific Outcomes (PSOs)
for
B.Sc. (Computer Science)

PSO1: Apply fundamental principles and methods of Computer Science to a wide range of applications.

PSO2: Design, correctly implement and document solutions to significant computational problems.

PSO3: Impart an understanding of the basics of our discipline.

PSO4: Prepare for continued professional development.

PSO5: Understand the impact of the IT analyst solutions in societal and environmental contexts, and demonstrate the knowledge and need for sustainable development.

PSO6: Develop proficiency in the practice of computing.

PSO7: Develop the capacity to study and research independently that will help to develop skills for transition to employment in hardware/software companies.

Anekant Education Society's

Tuljaram Chaturchand College, Baramati

(Empowered Autonomous)

Board of Studies (BOS) in Computer Science

(Academic Year 2025-26 to 2027-28)

Sr.No.	Name of Member	Designation
1.	**Dr. Choudhari Upendra Durgadas**	**Chairperson**
2.	**Dr. Kardile Vilas Vasantrao**	Member
3.	**Mr. Mankar Abhijeet Dnyaneshwar**	Member
4.	**Ms. Kulkarni Prajakta Pankaj**	Member
5.	**Ms. Bhagat Asmita Amol**	Member
6.	**Mr. Shah Rahul Adesh**	Member
7.	**Mr. Dixit Purushottam Suresh**	Member
8.	**Ms. Londhe Kalyani Waman**	Member
9.	**Ms. Swami Poornima Chandrashekhar**	Member
10.	**Ms. Theurkar Komal Manoj**	Member
11.	**Mr. Chemte Swapnil Pandurang**	Member
12.	**Ms. Attar Naziya Shahabuddin**	Member
13.	**Ms. Gharge Jyostna Pratik**	Member
14.	**Ms. Shivarkar Vaishnavi Kishor**	Member
15.	**Dr. Manisha Bharambe**	Vice-Chancellor Nominee Subject Expert from SPPU, Pune
16.	**Dr. Bhoite Sudhakar D.**	Subject Expert from Outside the Parent University
17.	**Dr. Patki Ulhas S.**	Subject Expert from Outside the Parent University
18.	**Mr. Yadav Preetam**	Representative from industry/ corporate sector/allied areas
19.	**Mr. Bhaskar Ranaware**	Member of the College Alumni
20.	**Ms. Sakshi Vargar**	UG Student Representative
21.	**Mr. Adesh Jagtap**	PG Student Representative

Course Structure for F.Y.B.Sc. (Computer Science) (2023 Pattern)

Sem.	Course Type	Course Code	Title of Course	TH/PR	Credits
I	Major Mandatory	COS-101-MJM	Basic Programming using C	Theory	2
	Major Mandatory	COS-102-MJM	DBMS-I	Theory	2
	Major Mandatory	COS-103-MJM	Computer Science Practical – I	Practical	2
	Open Elective (OE)	COS-116-OE	Internet Awareness	Theory	2
	Open Elective (OE)	COS-117-OE	Introduction to MS-Office	Practical	2
	Skill Enhancement Course (SEC) (Any one)	COS-126-SEC(ST) COS-126-SEC(MT) COS-126-SEC(EL)	Introduction to Statistical Software Mathematics for Computer Science Electronics	Practical	2
	Vocational Skill Course (VSC)	COS-121-VSC	Problem Solving Skills & DBMS Using PostgreSQL	Theory	2
	Ability Enhancement Course (AEC)	ENG-131-AEC	Functional English - I	Theory	2
	Value Education Course (VEC)	COS-135-VEC	Environmental Science	Theory	2
	Indian Knowledge System (IKS)	COS-137-IKS	Evolution of Computers	Theory	2
	Co-curricular Course (CC)	---------	To be selected from the Basket	------	2
Total Credits I:					**22**
II	Major Mandatory	COS-151-MJM	Advanced Programming Using C	Theory	2
	Major Mandatory	COS-152-MJM	DBMS-II	Theory	2
	Major Mandatory	COS-153-MJM	Computer Science Practical – II	Practical	2
	Minor (Any one)	COS-161-MN(ST) COS-161-MN(MT) COS-161-MN(EL)	Exploratory Data Analysis-I Discrete Mathematics Fundamentals of Electronics	Theory	2
	Open Elective (OE)	COS-166-OE	Advanced MS-Excel	Practical	2
	Open Elective (OE)	COS-167-OE	E-Banking	Theory	2
	Vocational Skill Course (VSC)	COS-171-VSC	Database Applications using PL/pgSQL	Theory	2
	Skill Enhancement Course (SEC)	COS-176-SEC	Basic Graphics Design using C	Practical	2
	Ability Enhancement Course (AEC)	ENG-181-AEC	Functional English – II	Theory	2
	Value Education Course (VEC)	COS-185-VEC	Digital and Technological Solutions	Theory	2
	Co-curricular Course (CC)	------	To be selected from the Basket	------	2
Total Credits II:					**22**
Cumulative Credits Semester I and II:					**44**

Credit & Course Structure for S. Y. B. Sc. (Computer Science) (2023 Pattern)

Sem	Course Type	Course Code	Title of Course	TH/PR	Credits
Sem-III	**Major Mandatory**	COS-201-MJM	Basic Data Structures	Theory	2
	Major Mandatory	COS-202-MJM	Introduction to Web Technology	Theory	2
	Major Mandatory	COS-203-MJM	Software Engineering Principles and Techniques	Theory	2
	Major Mandatory	COS-204-MJM	Lab Course I – Based on COS-201- MJM, COS-202-MJM	Practical	2
	Minor (Any one) (For B.Sc. (CS)) Statistics, Mathematics, Electronics	COS-211-MN(A), COS-211-MN (B), COS-211-MN(C)		Theory	2
	Minor (Any one) (For B.Sc. (CS)) Statistics, Mathematics, Electronics	COS-212-MN (A), COS-212-N(B), COS-212-MN(C)		Practical	2
	Minor *(For Others)*	COS-211-MN(D)	HTML5 using CSS	Theory	2
	Minor *(For Others)*	COS-212-MN(D)	Lab Course based on COS-211- MN (D)	Practical	2
	Open Elective (OE)	COS-216-OE	Fundamental Concepts in Computer Science	Theory	2
	Vocational Skill Course (VSC)	COS-221-VSC	Programming in C++	Theory	2
	Ability Enhancement Course (AEC)	MAR-231-AEC,	HIN-231-AEC, SAN-231-AEC,	Theory	2
	Co-curricular Course (CC)	YOG/PES/CUL/ NSS/NCC-239-CC	To be selected form the Basket	Theory	2
	Field Project (CEP)	COS-235-FP	Field Project	Practical	2
	Generic IKS Course (IKS)	GEN-245-IKS	Generic IKS	Theory	2
			Total Credits of SEM – III :		**24**

Sem	Course Type	Course Code	Title of course	TH/PR	Credits
	Major Mandatory	COS-251-MJM	Advanced Data Structure	Theory	2
	Major Mandatory	COS-252-MJM	Advanced Web Technology	Theory	2
	Major Mandatory	COS-253-MJM	Java Programming	Theory	2
	Major Mandatory	COS-254-MJM	Lab Course I – Based on COS-251-MJM, COS-252-MJM	Practical	2
	Minor (any one)	COS-261 MN(A), COS-261- MN(B), COS-261-MN(C)	Statistics, Mathematics, Electronics	Theory	2
	Minor (any one)	COS-262- MN(A) COS-262- MN(B) COS-262-MN(C)	Statistics, Mathematics, Electronics	Practical	2

Sem - IV	Minor (For Other)	COS-261-MN(D)	JAVA Script & Bootstrap	Theory	2
	Minor (For Other)	COS-262-MN(D)	Lab Course based on COS-261-MN(D)	Practical	2
	Open Elective (OE)	COS-216-OE	Basic Tools of Digital Marketing	Practical	2
	Skill Enhancement Course (VSC)	COS-276-SEC	Lab Course on COS-253-MJM	Practical	2
	Ability Enhancement Course (AEC)	MAR-231- AEC, HIN-231-AEC , SAN-231-AEC	Marathi, Hindi, Sanskrit	Theory	2
	Co-curricular Course (CC)	YOG/PES/CUL/ NSS/ NCC-239-CC	To be selected form the basket	Theory	2
	Community Engagement Project	COS-285-CEP	Community Engagement Project	Practical	2
Total Credits of SEM – IV					**22**

Credit & Course Structure for T. Y. B. Sc. (Computer Science) (2023 Pattern)
As per NEP-2020

Sem	Course Type	Course Code	Course Title	TH/PR	Credits
V	Major Mandatory	COS-301-MJM	Operating Systems	Theory	02
	Major Mandatory	COS-302-MJM	Theoretical Computer Science	Theory	02
	Major Mandatory	COS-303-MJM	Foundation of Computer Networking	Theory	02
	Major Mandatory	COS-304-MJM	Object Oriented Software Engineering	Theory	02
	Major Mandatory	COS-305-MJM	Lab Course based on COS-306-MJE(A) and COS-306-MJE(B)	Practical	02
	Major Elective (MJE)	COS-306-MJE(A)	PHP Essentials	Theory (Any two)	02
	Major Elective (MJE)	COS-306-MJE(B)	Advanced Java		02
	Major Elective (MJE)	COS-306-MJE(C)	Blockchain Technology		02
	Minor (Any one) (For B.Sc.(CS)) Statistics, Mathematics, Electronics	COS-311-MN(A)	Predictive Analytics	Theory	02
		COS-311-MN(B)	Linear Algebra		
		COS-311-MN(C)	8051 Programming and Applications		
		COS-312-MN(A)	Practical based on Predictive Analytics	Practical	02
		COS-312-MN(B)	Linear Algebra Practical using GeoGebra Software		
		COS-312-MN(C)	8051 Practical Lab		
	Minor (For Others)	COS-311-MN(D)	Web Design using WordPress	Theory	02
		COS-312-MN(D)	Lab Course based on COS-311-MN	Practical	02
	Vocational Skill Course (VSC)	COS-321-VSC	Lab Course based on COS-301-MJM and COS-303-MJM	Practical	02
	Field Project (FP)	COS-335-FP	Field Project	Practical	02
				Total Credits Semester – V	**22**
VI	Major Mandatory	COS-351-MJM	Advanced Operating Systems	Theory	02
	Major Mandatory	COS-352-MJM	Compiler Construction	Theory	02
	Major Mandatory	COS-353-MJM	Higher Layers of Networking and Network Security	Theory	02
	Major Mandatory	COS-354-MJM	Software Metrics and Project Management	Theory	02
	Major Mandatory	COS-355-MJM	Lab Course based on COS-356-MJE(A) and COS-356-MJE(B)	Practical	02
	Major Elective (MJE)	COS-356-MJE(A)	PHP Beyond Basics	Theory (Any two)	02
		COS-356-MJE(B)	Java Web Technologies		02
		COS-356-MJE(C)	Internet of Things		02
	Minor (Any one) (For B.Sc.(CS)) Statistics, Mathematics, Electronics	COS-361-MN(A)	Machine Learning	Theory	02
		COS-361-MN (B)	Computational Geometry		
		COS-361-MN(C)	Arduino and its programming		
		COS-362-MN(A)	Practical based on Machine Learning	Practical	02
		COS-362-MN (B)	Computational Geometry practical using Python programming		
		COS-362-MN(C)	Arduino Lab Experiments		
	Minor (For Others)	COS-361-MN(D)	Business Analysis using Advanced Excel	Theory	02
		COS-362-MN(D)	Lab Course based on COS-361-MN(D)	Practical	02
	On Job Training (OJT)	COS-385-OJT	On Job Training	Practical	04
				Total Credits Semester – VI	**22**
				Grand Total Semester- V + Semester- VI	**44**

T. Y. B. Sc. (Computer Science)

(2023 Pattern)

SEM- V Syllabus

Implementing from June – 2025

(A.Y. 2025-26)

SYLLABUS (CBCS as per NEP 2023) FOR T.Y.B. Sc. (Computer Science)
(w. e. from AY 2025-26)

Name of the Programme	**:** B.Sc. Computer Science
Program Code	**:** USCOS
Class	**:** T.Y. B.Sc. (Comp. Sci.)
Semester	**:** V
Course Type	**:** Major (TH)
Course Name	**:** Operating System
Course Code	**:** COS-301-MJM
No. of Lectures	**:** 30
No. of Credits	**:** 2

Prerequisites:

• Basic Knowledge of Computer

Course Objectives:

- To understand Complexity of Operating system as a software. .
- To understand design issues related to process management and various related algorithms
- To understand design issues related to memory management and various related algorithms

Course Outcome:

CO1: Learn the working of various operating system modules.
CO2: Able to understand the concept of process scheduling.
CO3: Learn system calls and their role in interacting with the operating system.
CO4: Study the role of the operating system in managing memory.
CO5: Learn the Multithreading Concepts
CO6: Understand how process synchronization is done
CO7: Learn concept of deadlock management.

UNIT	Chapter Name with Topics	No. of Lectures
UNIT- I	**Operating System as System Software** 1.1 What Operating Systems Do – User View, System View, Defining OS 1.2 Computer System Architecture – Single processor system, Multiprocessor systems, Clustered Systems 1.3 Operating System Operations – Dual mode operation, Timer 1.4 Process Management 1.5 Memory Management 1.6 Operating System Services 1.7 User Operating-System Interface – Command interpreter, GUI 1.8 System Calls and Types:-Process control, File management, Device management, Information maintenance, Communication, Protection	**06**
UNIT- II	**Process Management and Scheduling** 2.1 Process Concept – The process, Process states, Process control block.	**10**

	2.2 Process Scheduling – Scheduling queues, Schedulers, context switch 2.3 Operations on Process – Process creation with program using fork(), Process termination 2.4 Inter-process Communication – Shared memory system, Message passing systems. 2.5 Basic Concept – CPU-I/O burst cycle, CPU scheduler, Preemptive scheduling, Dispatcher 2.6 Scheduling Criteria 2.7 Scheduling Algorithms – FCFS, SJF, Priority scheduling, Round-robin scheduling, Multiple queue scheduling, Multilevel feedback queue scheduling. 2.8 Multithreaded Programming 2.9 Multithreading Models 6.6 Thread Scheduling	
UNIT-III	**Multithreaded Programming** 3.1 Overview 3.2 Multithreading Model 3.3 Thread Libraries P-Tread, Java Thread 3.4 Thread Life Cycle	**04**
UNIT-IV	**Process Synchronization and Deadlock** 4.1 Background 4.2 Critical Section Problem 4.3 Semaphores: Usage, Implementation 4.4 Classic Problems of Synchronization – The bounded buffer problem, The reader writer problem, The dining philosopher problem 4.5 System model 4.6 Deadlock Characterization – Necessary conditions, Resource allocation graph 4.7 Deadlock Prevention 4.8 Deadlock Avoidance - Safe state, Resource allocation graph algorithm, Banker's Algorithm 4.9 Deadlock Detection 4.10 Recovery from Deadlock – Process termination, Resource Preemption	**10**

Mapping of CO with PO Mapping of CO with PO

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PO13
CO1	3	2	2	2	2	1	1	2	3	1	1	2	1
CO2	3	3	2	3	3	2	1	2	3	1	1	2	1
CO3	3	3	2	3	3	2	1	2	3	1	1	2	1
CO4	3	3	2	3	3	1	1	2	3	1	1	2	1
CO5	2	3	3	3	3	2	1	2	3	1	1	2	1
CO6	3	3	3	3	3	2	1	2	3	1	1	2	1
CO7	3	3	2	3	3	2	1	2	3	1	1	2	1

Justification:

PO1 with All COs

CO1: Understanding OS modules directly contributes to strong foundational knowledge.

CO2: Process scheduling forms a core concept in OS knowledge.

CO3: System calls are crucial for understanding OS-user interaction.

CO4: Memory management is a vital theoretical concept.
CO5: Multithreading is significant but more applied in nature.
CO6: Synchronization is essential for OS core understanding.
CO7: Deadlock management is a key concept in OS theory.

PO2 with All COs:

CO1	Understanding modules contributes moderately to practical OS use.
CO2	Process scheduling is practically implemented and evaluated.
CO3	System calls are widely used in OS programming.
CO4	Hands-on memory management is key in labs and development.
CO5	Multithreading is heavily used in practical scenarios.
CO6	Synchronization techniques are important for concurrency control.
CO7	Practical strategies for avoiding deadlocks are crucial in system design.

PO3 with All COs

CO1	Encourages system-level thinking useful in tech entrepreneurship.
CO2	Process optimization can inspire performance-driven innovation.
CO3	System-level APIs can lead to low-level system innovation.
CO4	Efficient memory strategies can be a business differentiator.
CO5	Multithreading enables scalable, high-performance applications.
CO6	Safe synchronization is critical in commercial software.
CO7	Deadlock resolution plays a role in designing reliable systems.

PO4 with All COs

CO1	Introduces specialized OS components.
CO2	Develops algorithmic scheduling skills.
CO3	Enhances ability to interact with system-level APIs.
CO4	Builds competency in memory management techniques.
CO5	Develops skills in handling concurrent programming.
CO6	Strengthens ability in synchronization methods.
CO7	Builds deadlock handling skills vital for system safety.

PO5 with All COs

CO1	Moderate application of module concepts in solving OS problems.
CO2	Analytical reasoning in selecting appropriate scheduling algorithms.
CO3	Requires applying system calls in various scenarios.
CO4	Involves solving memory-related problems.
CO5	Needs problem-solving in thread coordination.
CO6	Analytical handling of synchronization issues.
CO7	Deadlock detection and resolution demands strong reasoning.

PO with All COs

CO1	Limited relevance to communication.
CO2	Collaborative problem-solving in labs.
CO3	Encourages communication through coding and debugging.
CO4	Less focused on group collaboration.
CO5	Possible collaboration in multithreaded projects.
CO6	Requires communication in team-based synchronization tasks.

CO7 Discussion and debugging deadlocks involve teamwork.

PO7 with All COs
CO1 Offers theoretical base but limited research.
CO2 Not typically research-oriented.
CO3 Some scope in systems research.
CO4 Limited direct research applications.
CO5 Multithreading has potential research aspects.
CO6 Can be expanded into research (e.g., new algorithms).
CO7 May lead to research in deadlock prevention techniques.

PO8 with All COs
CO1 Encourages independent exploration of OS design.
CO2 Promotes self-study of algorithms.
CO3 System-level coding improves learning independence.
CO4 Requires exploring various memory strategies.
CO5 Self-learning of thread libraries and tools.
CO6 Motivates learners to understand synchronization methods.
CO7 Encourages analytical learning in deadlock scenarios.

PO9 with All COs
CO1 Builds strong digital systems understanding.
CO2 Involves modern scheduling simulators/tools.
CO3 Teaches low-level system interaction.
CO4 Core to understanding memory in tech systems.
CO5 Essential for modern parallel computing.
CO6 Integral in building safe tech environments.
CO7 Enhances skills in tech system robustness.

PO10 with All COs
CO1–CO7 These topics are technical and don't directly engage with diversity, inclusion, or empathy themes.

PO11 with All COs
CO1–CO7 OS concepts do not directly relate to environmental or ethical values. Some indirect value inculcation via responsibility in programming.

PO12 with all COs
CO1 Encourages self-learning and exploration.
CO2 Needs self-discipline in understanding algorithms.
CO3 Promotes accountable coding and debugging.
CO4 Mistakes in memory management demand accountability.
CO5 Threading errors demand responsible handling.
CO6 Requires responsible synchronization practices.
CO7 Demands careful handling of deadlocks in systems.

PO13 With all COs
CO1–CO7 These technical topics have no direct engagement with community service.

SYLLABUS (CBCS as per NEP 2020) FOR T.Y.B. Sc. (Computer Science)
(w. e. from AY 2025-26)

Name of the Programme : B.Sc. Computer Science
Program Code : USCOS
Class : T.Y. B.Sc. (Comp. Sci.)
Semester : V
Course Type : Major Mandatory (TH)
Course Name : **Theoretical Computer Science**
Course Code : COS-302-MJM
No. of Lectures : 30
No. of Credits : 2

Prerequisite:

• Sets, Operations on sets, Finite & infinite sets Formal Language
/• Relation, Equivalence Relation, (reflexive, transitive and symmetric closures)

Course Objectives: Students successfully completing this course will be able:

- Understanding Foundations of Computation
- To have an understanding of finite state and pushdown automata.
- To have a knowledge of regular languages and context free languages.
- To know the relation between regular language, context free language and corresponding recognizers.
- To study the Turing machine and classes of problems.
- Problem Solving and Critical Thinking
- Applications of Theoretical Computer Science

Course Outcome:

CO1: Knowledge of automata, formal language theory and computability
CO2: Demonstrate advanced knowledge of formal computation and its relationship to languages.
CO3: Distinguish different computing languages and classify their respective types.
CO4: Recognize and comprehend formal reasoning about languages.
CO5: Show a competent understanding of the basic concepts of complexity theory.
CO6: The students will be able to design.
CO7: To know basic models of information processing.

Units	Topic Contents	No. of Lectures
Unit -I	**Finite Automata and Regular Languages** • Deterministic finite Automaton – Definition, DFA as language recognizer, DFA as a pattern recognizer. • Nondeterministic finite automaton – Definition and Examples. • NFA TO DFA • NFA with ε- transitions Definition and Examples. • NFA with ε-Transitions to DFA & Examples • Finite automaton with output – Mealy and Moore machine, Definition and Examples. • Minimization of DFA, Algorithm & Problem using Table Method.	**10**

	• Regular language-Definition and Examples. • Conversion of RE To FA-Examples. • Pumping lemma for regular languages and applications. • Closure properties of regular Languages	
Unit –II	**Context Free Grammar and Languages** • Grammar - Definition and Examples. • Derivation - Definition and Examples. • Chomsky Hierarchy. • CFG: Definition & Examples. LMD, RMD, ,Parse Tree • Simplification of CFG : • Removing Useless Symbols, • Removing unit productions • Removing ϵ productions & Nullable symbols • Normal Forms : • Chomsky Normal Form (CNF) Method & Problem • Greibach Normal form (GNF) Method & Problem • Regular Grammar : Definition and types • Closure Properties of CFL's	**08**
Unit- III	**Push Down Automaton** • Definition of PDA and examples • Construction of PDA using empty stack and final State method : Examples using stack method • Definition DPDA & NPDA, their correlation and Examples of NPDA • CFG (in GNF) to PDA : Method and examples	**06**
Unit – IV	**Turing Machine** • The Turing Machine Model and Definition of TM • Design of Turing Machines • Problems on language recognizers. • Language accepted by TM • Turing Machine Limitations	**06**

References:-

1. Introduction to Automata theory, Languages and computation By John E. Hopcroft and Jeffrey Ullman – Narosa Publishing House.
2. Introduction to Automata theory, Languages and computation By John Hopcroft, Rajeev Motwani and Jeffrey Ullman –Third edition Pearson Education
3. Introduction to Computer Theory Daniel I. A. Cohen – 2nd edition – John Wiley & Sons
4. Theory of Computer Science (Automata, Language & Computation) K. L. P. Mishra & N. Chandrasekaran, PHI Second Edition
5. Introduction to Languages and The Theory of Computation John C. Martin TMH, Second Edition

Mapping of this course with Programme Outcomes & it's justification

Course Outcomes	Programme Outcomes (POs)												
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PO13
CO1	3	2	1	2	3	1	2	2	2	1	1	2	1
CO2	3	3	1	2	3	1	3	2	2	1	1	2	1
CO3	3	2	2	2	3	1	2	2	2	1	1	2	1
CO4	3	2	1	2	3	1	2	2	2	1	1	2	1
CO5	3	2	2	3	3	1	3	2	2	1	1	2	1
CO6	2	3	2	3	3	2	2	2	2	1	1	3	2
CO7	3	3	2	2	3	2	2	2	2	1	1	2	

Weight: 1 - Partially related 2 - Moderately Related 3 - Strongly related

PO1 with all COs
CO1–CO7: All COs focus on deep theoretical understanding of automata, languages, computation, and complexity—making this PO highly relevant.

PO2 with all COs
CO2, CO6, CO7: Designing and analysing computation models contributes to professional skillsets, especially when applying theory in software or hardware systems.

PO3 with all COs
CO3, CO5, CO6, CO7: Identifying language types and complexity enables innovation in tools like compilers, interpreters, or low-level processors—skills beneficial in entrepreneurial ventures.

PO4 with all COs
CO2, CO3, CO5, CO6: These COs focus on high-level abstractions and designing models, which are crucial for roles in formal methods, language processing, or compiler design.

PO5 with all COs
CO1–CO7: Each CO enhances critical thinking in algorithmic reasoning, decidability, and computational complexity—all fostering strong problem-solving skills.

PO6 with all COs
CO6, CO7: Design tasks and group discussions on computational models can foster collaboration and enhance articulation of complex theoretical ideas.

PO7 with all COs

CO2, CO5, CO7: Advanced knowledge of computability and complexity theory lays the foundation for academic or industrial research in algorithms and formal systems.

PO8 with all COs

CO1–CO7: Theory of Computation requires mastering abstract thinking and evolving paradigms, helping students develop independent learning strategies.

PO9 with all COs

CO1–CO7: Though theoretical, this course supports digital foundations by informing the development of compilers, programming languages, and logic systems.

PO10 with all COs

CO3, CO4: Understanding diverse computation models and abstract languages encourages appreciation of varied problem-solving perspectives.

PO11 with all COs

All COs: Indirect connection—developing responsible and ethical thinking when designing systems, though this PO is less emphasized in theory-heavy courses.

PO12: with all COs

CO6, CO7: Designing models and understanding processes cultivates ownership, self-discipline, and academic rigor.

PO13 with all COs

CO6, CO7: Theories may indirectly support development of educational tools or community-based tech solutions focused on logic, computation, and decision-making support.

SYLLABUS (CBCS) FOR T.Y.B.Sc. (Computer Science) (SEM-V)
(w.e.f. A.Y.- 2025-2026)

Name of the Programme	**:** B.Sc. Computer Science
Program Code	**:** USCOS
Class	**:** T.Y. B.Sc. (Comp. Sci.)
Semester	**:** V
Course Type	**:** Major (TH)
Course Name	**:** Foundation of Computer Network
Course Code	**:** COS-303-MJM
No. of Lectures	**:** 30
No. of Credits	**:** 2

Pre-requisites: Basics knowledge of computer

Objectives: This course will prepare students in Basic networking concepts.

1. Understand different types of networks, various topologies and application of networks.
2. Understand types of addresses, data communication.
3. Understand the concept of networking models, protocols, functionality of each layer.
4. Learn basic networking hardware and tools.
5. Understand wired and wireless networks, its types, functionality of layer.

Learning Outcomes:

CO1: Understanding Networking Concepts - Define and explain fundamental networking concepts, including protocols, data communication and network architectures.
CO2: Network Models - Understand and apply knowledge of OSI, TCP/IP Models.
CO3: Network Protocols – Describe and analyze various networking protocols and their Functionalities.
C04: Network Design and implementation - Design and implement LAN, WAN & MAN.
CO5: Networking devices and Technologies - Evaluate and select appropriate networking devices and technologies for specific scenarios.
CO6: Wired & Wireless Networking: Understand principles, understand design and implementation of wired & wireless communication.
CO7: Internet Technologies: Understand the functioning of the internet and related technologies.

Units No.	Title & Contents	No. of Lectures
I	**Introduction to Computer Network** 1.1 Computer Networks- Goals, applications 1.2 Network Hardware's – Broadcast and point to point. 1.3 Topology – Star, Bus, Mesh, Ring etc. 1.4 Network Types: LAN, MAN, WAN, Wireless Network, internetwork 1.5 Data Communication – Definition, Components, data representation, Data flow. 1.6 Protocols and Standards: Defacto, Dejure standard 1.7 Network Software- Protocol Hierarchies, Design issues of the layer, Connection and connectionless services,	06
II	**Network Models & Transmission Media** 2.1 Reference Model – OSI Reference Model, TCP/IP Reference Model,	06

	Comparison of OSI & TCP/IP Model, 2.2 Addressing – Physical, Logical and Port addresses 2.3 Guided Media – Twisted pair cable, Coaxial Cable, Fiber optic cable 2.4 Unguided Media – Radio Waves, Micro wave Transmission, Infrared, Light wave Transmission	
III	**Lower layers: Physical and Data link layers** 3.1 Communication at the physical layer, Data and signals. Transmission Impairment, Data rate limits, Performance Transmission Modes. 3.2 Switching – Circuit, Message and Packet Switching. 3.3 Design issues of Data Link Layer, Services – Framing, Error control, Flow Control, Congestion Control, Link layer addressing. 3.4 Data link Protocols – simplex, stop and wait and stop and wait Automatic Repeat Request (ARQ). 3.5 Sliding Window Protocols – One-bit sliding window protocol, Pipeline technique, Go back N and Selective Repeat Automatic Repeat Request with comparison. 3.6 DLL Protocols – HDLC, PPP 3.7 Medium Access Sublayer: Random Access Protocols, Controlled Access, Channelization. 3.8 Physical and Data link layer devices – Repeater, Hubs, Bridge	10
IV	**The Network Layer** 4.1 Design Issues, Store-and-forward packet switching, 4.2 Services Provided to the Transport Layer, Implementation of Connectionless and Connection Oriented Service, 4.3 Comparison of Virtual Circuit and Datagram subnets 4.4 Logical Addressing IPV4 Addresses – Address Space, Notations, ClassfulAddressing, Subnetting, Supernetting, Classless Addressing, Network Address Translation (NAT), (Problems should be covered on Addressing) 4.5 IPV4 Protocol Datagram Format 4.6 Routing Properties of routing algorithm, Adaptive and Non- Adaptive Routing Algorithms 4.7 Congestion Control – Definition, Factors of Congestion, Difference between congestion control and flow control, General Principles and Congestion Prevention Policies 4.8 Network Layer Devices –Routers	08

Reference Books:

1) Computer Networks by Andrew Tanenbaum, Pearson Education.[4th Edition]

2) Data Communication and Networking by Behrouz Forouzan,

TATA McGraw Hill.[4th/5thEd.]

3) Networking All In One Dummies Wiley Publication.[5th Edition]

CO-PO Mapping Table:

COs \ POs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PO13
CO1	3	2	1	1	2	1	1	2	2	1	1	1	1
CO2	3	2	1	1	2	1	1	2	3	1	1	1	1
CO3	3	3	1	1	3	1	2	2	3	1	1	1	1
CO4	2	3	2	3	3	2	1	2	3	1	1	2	2
CO5	2	3	1	3	3	2	2	2	3	1	1	2	2
CO6	2	3	1	2	3	2	2	2	3	1	1	2	2
CO7	2	2	2	2	3	1	2	2	3	1	1	1	2

Justification for Mapping"

CO1: Understanding Networking Concepts
- **PO1 (3):** Strong foundational knowledge.
- **PO2 (2):** Procedural understanding of network architecture.
- **PO5 (2):** Basic problem-solving with networking basics.
- **PO9 (2):** Digital literacy is inherent in understanding networks.

Other POs have low or general relevance.

CO2: Network Models
- **PO1 (3):** Requires a comprehensive grasp of conceptual models.
- **PO2 (2):** Application-based procedural knowledge.
- **PO5 (2):** Analytical reasoning for model usage.
- **PO9 (3):** Crucial for understanding digital protocols.

CO3: Network Protocols
- **PO1 (3):** In-depth theoretical understanding required.
- **PO2 (3):** Strong practical and procedural applications.
- **PO5 (3):** Requires analysis and evaluation of protocol behavior.
- **PO9 (3):** Protocols are the backbone of digital communication.

CO4: Network Design and Implementation
- **PO2 (3):** Emphasizes professional/practical application.
- **PO4 (3):** Involves specialized technical skill.
- **PO5 (3):** High-level problem-solving and reasoning.
- **PO12 & PO13 (2):** Autonomy and responsibility in project work; community engagement possible in real-world implementation.

CO5: Networking Devices and Technologies
- **PO2 (3):** Selection of devices is procedural and practical.
- **PO4 (3):** Needs specialized technical knowledge.
- **PO5 (3):** Requires judgment and problem-solving.
- **PO9 (3):** Heavy emphasis on digital tools and tech.
- **PO12 & PO13 (2):** Applicable in practical network setups, especially in collaborative or community projects.

CO6: Wired & Wireless Networking

PO2 (3): Practical/professional knowledge of both mediums.
PO4 (2): Competency in hardware and protocols.
PO5 (3): Analytical and design-based problem-solving.
PO9 (3): Technologies are deeply digital.
PO6, PO12, PO13 (2): Collaboration and real-world application aspects.

CO7: Internet Technologies

PO1 (2): Fundamental understanding needed.
PO2 (2): Functional and procedural knowledge.
PO3 (2): Innovation and entrepreneurship in internet services.
PO5 (3): Evaluative and design skills are key.
PO9 (3): Focus on digital technologies.
PO13 (2): Relevant for service and outreach initiatives.

SYLLABUS (CBCS) FOR T.Y.B.Sc. (Computer Science) (SEM-V)
(w.e.f. A.Y.- 2025-2026)

Name of the Programme	**:** B.Sc. Computer Science
Program Code	**:** USCOS
Class	**:** T.Y. B.Sc. (Comp. Sci.)
Semester	**:** V
Course Type	**:** Major (TH)
Course Name	**:** Object Oriented Software Engineering
Course Code	**:** COS-304-MJM
No. of Lectures	**:** 30
No. of Credits	**:** 2

Prerequisites: Knowledge of Classical Software Engineering

Objectives

- Understanding Object Orientation in Software engineering concepts and importance.
- Understand the Unified Modeling Language concepts, importance and its component.
- Understand Structural, Behavioral, Dynamic modeling techniques and diagrams.
- Apply Object Oriented concepts and its techniques for software development.
- Implement the Unified Modeling Language concepts, importance and its component.
- Apply Structural, Behavioral, Dynamic modeling techniques and diagrams.
- Implement the concept of Object Oriented software development process model.

OUTCOMES:

CO1: Develop models using the UML notation.

CO2: Apply an iterative, agile process.

CO3: Analyze requirements with use cases.

CO4: Create domain models

CO5: Relate analysis and design artifacts.

CO6: Design object solutions with patterns and architectural layers.

CO7: Apply concepts to a semester-long software engineering project.

Title and Contents		No. of Lectures
Unit 1	**Object Oriented Concepts and Principles** .1. Introduction, Object, Classes and Instance, Polymorphism, Inheritance 1. 2 Object Oriented System Development- Introduction, Function / Data Methods (With Visibility), Object Oriented Analysis, Object Oriented Construction .2. Identifying the Elements of an Object Model Aggregations, .3. Identifying Classes and Objects, Identity, Dynamic binding, Persistence, Meta classes 1.5 Specifying the Attributes (With Visibility) 1.6 Defining Operations 1.7 Finalizing the Object Definition	**04**
Unit 2	**Introduction to UML and Object Oriented Methodology** 2.1 Concept of UML 2.2 Advantages of UML 2.3 Object oriented Methods (The Booch Method, The Coad & Yourdon Method, Jacobson Method and Raumbaugh Method)	**02**
Unit 3	**Structural Modeling** 3.1 Classes 3.2 Relationship (Interface, Types and Roles, Packages , Common Mechanism) 3.4 Class Diagram (Minimum three examples should be covered) 3.5 Object Diagram (Minimum three examples should be covered) **Behavioral Modeling** 3.6 Interactions 3.7 Use Cases and Use Case Diagram with stereo types (Minimum three examples should be covered) 3.8 Interaction Diagram (Minimum two examples should be covered) 3.9 Sequence Diagram (Minimum two examples should be covered) 3.10 Activity Diagram (Minimum two examples should be covered) 3.11 State Chart Diagram (Minimum two examples should be covered)	**14**
Unit 4	**Object Oriented Analysis, Architectural modeling, , Testing** 4.1 Iterative Development and the Rational Unified Process 4.2 Inception 4.3 Understanding Requirements 4.4 Use Case Model from Inception to Elaboration 4.5 Elaboration 4.6 Component 4.7 Components Diagram (Minimum two examples should be covered) 4.8 Deployment Diagram (Minimum two examples should be covered) 4.9 Collaboration Diagram (Minimum two examples should be covered) Object Oriented Testing Strategies 4.10 Test Case Design for Object Oriented Software 4.11 Inter Class Test Case Design(Use of any freeware designing tool)	**10**

References

- Ivar Jacobson, "Object Oriented Software Engineering", Pearson Education INC
- Craig Larman, "Applying UML and Patterns" Pearson Education INC

- Bennett, Simon, "Object Oriented Analysis and Design" McGraw Hill
- Ali Bahrami, "Object Oriented System Development", McGraw Hill International Edition, 2008
- Brahma Dathan, Sarnath Ramnath, "Object-Oriented Analysis, Design and Implementation", Universities Press, 2010
- Bernd Bruegge, Allen H. Dutoit, Object Oriented Software Engineering using UML,Patterns and Java, Pearson 2004
- Craig Larman, Applying UML and Patterns – An Introduction to Object-Oriented Analysis and Design and Iterative Development" , 3rd Edition, Pearson Education, 2005
- Grady Booch, James Rumbaugh, Ivar Jacobson, "The Unified Modeling Language User Guide", Addison Wesley Long man, 1999
- Martin Fowler, "UML Distilled A Brief Guide to Standard Object Modeling Language", 3rd Edition, Addison Wesley, 2003
- Russ Miles, Kim Hamilton, "Learning UML 2.0", O'Reilly, 2008

Mapping of this course with Programme Outcomes

Course Outcomes	**Programme Outcomes (POs)**												
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PO13
CO1	3	3	2	3	3	2	2	2	3	2	1	2	1
CO2	3	3	3	3	3	3	2	3	3	2	1	3	2
CO3	3	3	2	3	3	2	2	3	3	2	1	3	2
CO4	3	3	2	3	3	2	2	2	3	2	1	2	1
CO5	3	3	2	3	3	2	2	2	3	2	1	2	1
CO6	3	3	2	3	3	3	2	2	3	2	1	3	1
CO7	3	3	3	3	3	3	2	3	3	2	2	3	3

Weight: 1 - Partially related 2 - Moderately Related 3 - Strongly related

Justification:

PO1With All COs
CO1: Provides a foundational understanding of UML modeling and its role in software development.
CO2: Enhances theoretical knowledge of agile methodologies.
CO3: Builds conceptual understanding of requirement analysis through use cases.
CO4: Deepens knowledge of object-oriented principles through domain modeling.
CO5: Connects theoretical knowledge with design documentation.
CO6: Introduces architectural layers and design patterns as key knowledge areas.
CO7: Applies comprehensive concepts in a full-cycle project, reinforcing understanding.

PO2 with all COs
CO1: Builds hands-on skills in creating UML diagrams.
CO2: Applies agile practices in a procedural manner.
CO3: Translates user requirements into professional artifacts.
CO4: Develops domain models essential for structured development.
CO5: Links analytical and design activities into a procedural flow.
CO6: Applies architectural standards and reusable design practices.
CO7: Implements a complete development cycle, integrating procedural knowledge.

PO3 with all COs
CO1: Supports early-stage conceptualization for product planning.
CO2: Fosters adaptive thinking through iterative development.
CO3: Encourages customer-oriented analysis, key in product design.
CO4: Enhances ability to define innovative domain structures.
CO5: Connects ideas to real design strategies, aiding innovation.
CO6: Enables design of scalable solutions with entrepreneurial vision.
CO7: Facilitates innovation and implementation in project settings.

PO4 with all COs
CO1: Enhances modeling skills using standard notations.
CO2: Builds competency in agile methodologies.
CO3: Strengthens requirement elicitation and analysis abilities.
CO4: Develops specific skills in object-oriented analysis.
CO5: Focuses on linking analysis with specialized design skills.
CO6: Applies design patterns and architecture knowledge.
CO7: Integrates all competencies in a practical setting.

PO5 with all COs
CO1: Applies visual modeling to real-world software problems.
CO2: Encourages iterative problem-solving through agile.
CO3: Analyzes problems via use case scenarios.
CO4: Uses domain analysis for structured reasoning.
CO5: Applies analytical thinking in transforming requirements to design.
CO6: Solves design-level problems using established patterns.
CO7: Applies and tests analytical reasoning in a semester-long project.

PO6 with all COs
CO1: Improves communication of ideas using UML.
CO2: Encourages collaborative development through agile processes.
CO3: Requires clear articulation of requirements.
CO4: Supports group understanding of domain structure.
CO5: Enhances documentation and verbal communication of design.
CO6: Encourages collaborative architectural decisions.
CO7: Demands teamwork and communication throughout project phases.

PO7 with all COs
CO1: Involves exploration of best practices in modeling.
CO2: Requires adapting and researching iterative development methods.
CO3: Includes exploration of diverse user needs and analysis methods.
CO4: Encourages investigation into domain structures and patterns.
CO5: Supports research into effective design practices.
CO6: Requires understanding advanced architectural frameworks.
CO7: Promotes research-based solutions within project work.

PO8 with all COs
CO1: Develops self-learning skills in using UML tools.
CO2: Promotes continuous improvement via iterations.
CO3: Encourages questioning and redefining requirements.

CO4: Fosters independent exploration of modeling techniques.
CO5: Engages learners in refining design strategies.
CO6: Builds learning through use of design templates.
CO7: Cultivates lifelong learning habits in end-to-end projects.

PO9 with all COs
CO1: Uses modeling software tools effectively.
CO2: Utilizes agile tools and task trackers.
CO3: Applies tools for requirement documentation.
CO4: Integrates CASE tools for modeling.
CO5: Involves digital tools for design documentation.
CO6: Applies architectural tools and technologies.
CO7: Combines multiple digital tools in project development.

PO10 with all COs
CO1: Encourages inclusivity in design thinking.
CO2: Agile teamwork fosters multicultural collaboration.
CO3: Promotes empathy by understanding diverse user requirements.
CO4: Designs systems adaptable for diverse user domains.
CO5: Includes ethical and inclusive considerations in design.
CO6: Designs solutions for broad user demographics.
CO7: Encourages inclusive team practices and stakeholder empathy.

PO11 with all COs
CO1: Encourages clarity and integrity in modeling.
CO2: Agile methodology reinforces ethical team dynamics.
CO3: Requires analysis of responsible user scenarios.
CO4: Promotes design for sustainability and clarity.
CO5: Encourages value-based system design.
CO6: Designs solutions with social/environmental consideration.
CO7: Projects can target socially relevant themes.

PO12 with all COs
CO1: Students take ownership of their models.
CO2: Builds responsibility in agile task assignment.
CO3: Encourages accountability in requirement gathering.
CO4: Students are responsible for domain accuracy.
CO5: Requires independent work in transforming analysis to design.
CO6: Demands careful design decisions.
CO7: Projects require planning, delegation, and accountability.

PO13 with all COs
CO1: Modeling can focus on community-based systems.
CO2: Agile approach suits projects with social impact.
CO3: Use case analysis can reflect public needs.
CO4: Domain modeling can represent community structures.
CO5: Design can include public service components.
CO6: Patterns and architecture can support civic infrastructure.
CO7: Projects may aim to solve real community problems.

SYLLABUS (CBCS as per NEP 2023) FOR T.Y.B. Sc. (Computer Science)
(w. e. from AY 2025-26)

Name of the Programme	**:** B.Sc. Computer Science
Program Code	**:** USCOS
Class	**:** T.Y. B.Sc. (Comp. Sci.)
Semester	**:** V
Course Type	**:** Vocational Skill Course (PR)
Course Name	**:** Lab Course based on COS-MJM-301 and COS-MJM-303
Course Code	**:**COS-321-VSC
No. of Practical's	**:** 15
No. of Credits	**:** 2

Prerequisites:

• Basic Knowledge of Computer

Course Objectives:

- To understand Complexity of Operating system as a software. .
- To understand design issues related to process management and various related algorithms
- To understand design issues related to memory management and various related algorithms

Course Outcome:

CO1: Working of Shell and system call
CO2: Implementation of process scheduling
CO3: Implementation of deadlock avoidance algorithm
CO4: Study the role of the operating system in managing memory.
CO5: Learn the Multithreading Concepts
CO6: Learn networking essentials
CO7: Learn network administration and configuration

Assignment No.	Name of Assignment	No. of Practical Required
1.	UNIX Shell Programming	02
2.	CPU Scheduling	04
3.	Bankers Algorithm	02
4.	Case Study	02
5.	Topology Connectivity	02
6.	Transmission Media Assignment	01
7.	Network Configuration	01
8.	Network devices	01

Mapping of PO and CO with Justification

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PO13
CO1	3	3	2	3	3	2	2	2	3	1	1	2	1
CO2	3	3	2	3	3	2	1	2	3	1	1	2	1
CO3	3	3	2	3	3	2	1	2	3	1	1	2	1
CO4	3	3	2	3	3	1	1	2	3	1	1	2	1
CO5	2	3	3	3	3	2	1	2	3	1	1	2	1
CO6	2	3	3	3	3	3	2	2	3	2	2	3	2
CO7	2	3	3	3	3	3	2	2	3	2	2	3	2

Justification:

PO1 with All COs

CO1: Builds strong conceptual foundation about shell and system calls, which are core components of OS.

CO2: In-depth knowledge of how CPU scheduling works improves understanding of OS internals.

CO3: Studying deadlock avoidance deepens theoretical knowledge and resource allocation strategies.

CO4: Covers essential concepts of memory hierarchy and management.

CO5: Offers insight into concurrency models and OS-level multithreading.

CO6: Introduces networking concepts essential for modern OS design.

CO7: Helps understand administrative tasks and real-world network configuration as part of OS functionality.

PO2 with All COs

CO1: Practical understanding of how shell and system calls operate within an OS.

CO2: Hands-on implementation of scheduling algorithms demonstrates real-time process control.

CO3: Deadlock handling simulations develop procedural problem-solving skills.

CO4: Demonstrates memory allocation methods in actual systems.

CO5: Applying multithreading principles enhances procedural concurrency skills.

CO6: Involves practical labs on network essentials with OS interaction.

CO7: Involves configuring and troubleshooting networks—a professional skill in admin roles.

PO3 with All COs

CO1: Encourages creative system-level thinking in shell programming.

CO2: Process scheduling knowledge can inspire optimization tools or products.

CO3: Encourages thinking about innovative solutions for system deadlocks.

CO4: Sparks entrepreneurial thinking for building efficient memory solutions.

CO5: Multithreading concepts support ideas in developing performance-oriented applications.

CO6: Enables understanding of network essentials for tech-based startups.

CO7: Useful for entrepreneurial efforts in system administration or network solutions.

PO4 with All COs

CO1: Skill in writing and debugging system-level programs.
CO2: Specialization in designing and analyzing scheduling systems.
CO3: Expertise in handling resource allocation and system stability.
CO4: Deepens knowledge of memory layout, paging, segmentation.
CO5: Prepares for advanced roles in multi-core and parallel programming.
CO6: Develops essential networking skills critical for specialized OS tasks.
CO7: Prepares students for system/network admin roles with configuration expertise.

PO5 with All COs

CO1: Requires analytical thinking to understand shell behavior and system services.
CO2: Problem-solving in implementing and analyzing scheduling algorithms.
CO3: Encourages developing strategies to avoid deadlock scenarios.
CO4: Requires reasoning about memory utilization and management techniques.
CO5: Involves solving concurrency problems using multithreading.
CO6: Requires analytical setup and troubleshooting of networks.
CO7: Involves applying knowledge to solve configuration and network issues.

PO6 with All COs

CO1: Basic system programming discussions help improve technical communication.
CO2: Collaborative coding in scheduling projects.
CO3: Peer reviews on deadlock logic help develop clarity in explanation.
CO4: Less emphasis here, but discussions improve teamwork.
CO5: Working in teams to simulate thread synchronization.
CO6: Strong teamwork and communication needed during networking labs.
CO7: Real-world admin tasks often require group coordination.

PO7 with All COs

CO1: Introduces basic investigation of how systems operate internally.
CO2: Can inspire scheduling optimization research.
CO3: Encourages studying resource-allocation models.
CO4: Triggers investigation into memory management research topics.
CO5: Can lead to exploring performance benchmarks for threading models.
CO6: Opens inquiry into protocol design and network performance.
CO7: Research-driven understanding of secure and optimized configurations.

PO8 with All COs

CO1–CO7: Each CO contributes by requiring students to independently explore and learn system internals, configurations, and concepts through hands-on labs, manuals, and simulation tools.

PO9 with All COs

CO1–CO7: All outcomes heavily support this PO. They involve command-line tools, system programming, OS configuration, threading, and networking—core digital competencies.

PO10 with All COs

CO1–CO7: Indirectly supported when students collaborate across cultures during group tasks or peer reviews. Networking and OS administration might also include diverse user considerations.

PO11 with All COs

CO1–CO7: While technical, responsible resource usage (like CPU, memory, network bandwidth) can help foster eco-conscious software design.

PO12 with All COs

CO1–CO7: All tasks promote independent coding, debugging, and problem-solving. Configuration and administration tasks develop a sense of responsibility and accountability.

PO13 with All COs

CO1–CO7: Especially CO6 and CO7 foster skills that can be applied in community tech support, digital literacy training, or volunteer IT setup in local networks.

SYLLABUS (CBCS as per NEP 1.0 2023 Pattern) For T. Y. B. Sc. (Computer Science) Sem-V

(w. e. from June, 2025)

Name of the Programme : B.Sc. Computer Science
Program Code : USCOS
Class : T. Y. B. Sc. (Computer Science)
Semester : V
Course Type : Major Elective (TH)
Course Name : PHP Essentials
Course Code : COS-306-MJE(A)
No. of Lectures : 30
No. of Credits : 02

Prerequisites: • HTML, CSS

Objectives:

- To learn basics of PHP.
- To design dynamic, interactive web pages.
- To learn string and arrays in PHP.
- To learn about sanitizing user inputs.
- To learn object-oriented programming.
- To learn database connectivity with PHP.
- To understand how to structure code efficiently.

Outcome:

CO1.Grasp the fundamentals of PHP syntax and programming constructs.
CO2. Develop dynamic and interactive web pages using PHP.
CO3. Understand the various string related operations in PHP.
CO4.Understand basic security practices, such as sanitizing user inputs and preventing SQL injection.
CO5. Gain basic knowledge of PHP's object-oriented features like classes, objects, and inheritance.
CO6. Gain an understanding of how PHP interacts with web servers and databases.
CO7. Design and structure code efficiently, promoting readability and reusability.

Unit	Contents	No. of Lectures Required
1.	**Introduction to PHP** 1.1 Lexical structure 1.2 Language basics 1.3 Defining and calling a function 1.4 Default parameters 1.5 Variable parameters, Missing parameters 1.6 Variable function, Anonymous function	**04**
2.	**String and Arrays** 2.2 Types of strings in PHP 2.2 Comparing strings 2.3Manipulating and searching strings 2.4Regular Expressions 2.5Indexed Vs Associative arrays 2.6Storing data in arrays 2.7Multidimensional arrays 2.8Extracting multiple values	**08**

	2.9Sorting 2.10Action on entire arrays	
3.	**Introduction to Object Oriented Programming** 3.1Classes and Objects 3.2Inheritance 3.3Interfaces 3.4Encapsulation 3.5Traits 3.6Autoloading classes 3.7Exception handling 3.8Predefined exceptions 3.9Namespaces in OOP in PHP 3.10 Predefined PHP classes and interfaces	**12**
4	**Databases (PHP-PostgreSQL)** **4.1**Introduction to PDO 4.2Predefined constants 4.3Supported databases 4.4The PDO class 4.5PDO class methods 4.5Security using PDO 4.6PDOStatement class 4.7 Create, Read, Update and Delete (CRUD) operations	**06**

References:

1 Kevin Tatroe, Peter MacIntyre (2020), Programming PHP: Creating Dynamic Web Pages(4th ed.). O'Reilly.

a. **Web References**:

2 https://www.php.net/manual/en/manual.php
3 https://www.php-fig.org/
4 https://phptherightway.com
5 https://w3schools.com

Mapping of this course with Programme Outcomes

Course Outcomes	Programme Outcomes(POs)												
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PO13
CO1	3	2	1	2	2	1	1	2	2	1	1	1	1
CO2	3	3	2	3	3	2	2	3	3	2	2	2	2
CO3	3	2	1	2	2	1	1	2	2	1	1	1	1
CO4	2	3	2	3	3	1	2	3	3	2	2	3	2
CO5	3	3	2	3	3	2	2	2	3	1	1	2	2
CO6	3	3	2	3	3	2	2	3	3	2	2	2	2
CO7	2	3	2	3	3	2	2	3	3	2	2	3	2

Weight: 1 - Partially related 2 - Moderately Related 3 - Strongly related

Justification of mapping

Mapping of PO1 to all COs

CO1:PO1- Strongly relates as understanding syntax is fundamental knowledge.

CO2:PO1- Applies fundamental knowledge in real-world use.

CO3:PO1- Strongly tied to core programming understanding.
CO4:PO1- Requires solid theoretical understanding of vulnerabilities.
CO5:PO1- Deepens understanding of advanced programming concepts.
CO6:PO1- Strongly enhances understanding of web backend logic.
CO7:PO1- Relates to better understanding of coding principles.

Mapping of PO2 to all COs
CO1:PO2- Moderately related since basic syntax supports practical knowledge.
CO2:PO2- Strongly involves procedural and practical knowledge.
CO3:PO2- Moderately relates to practical usage of strings.
CO4:PO2- Strongly tied to professional and procedural application.
CO5:PO2- Strongly relates to procedural and professional design.
CO6:PO2- Core procedural skill for web programming.
CO7:PO2- Strongly tied to professional coding practices.

Mapping of PO3 to all COs
CO1:PO3- Partially helps in initiating simple entrepreneurial projects.
CO2:PO3- Enables creating business websites/applications.
CO3:PO3- Partial use in formatting content for business tools.
CO4:PO3- Important for secure business application development.
CO5:PO3- Useful in scalable entrepreneurial projects.
CO6:PO3- Essential for database-backed business apps.
CO7:PO3- Supports maintainable, scalable business solutions.

Mapping of PO4 to all COs
CO1:PO4- Provides foundational programming skills.
CO2:PO4- Strongly relates to developing specialized coding skills.
CO3:PO4- Enhances problem-solving through string manipulations.
CO4:PO4- Vital skill in secure coding.
CO5:PO4- Builds competency in software development.
CO6:PO4- Specialized web development competency.
CO7:PO4- Key skill for software competency.

Mapping of PO5 to all COs
CO1:PO5- Involves logical thinking and problem-solving basics.
CO2:PO5- Involves problem-solving for dynamic page generation.
CO3:PO5- Involves logic in handling and parsing string data.
CO4:PO5- Involves analytical reasoning for security loopholes.
CO5:PO5- Promotes structured thinking and reuse.
CO6:PO5- Involves logic and application in DB handling.
CO7:PO5- Strongly linked to problem-solving via clean code.

Mapping of PO6 to all COs
CO1:PO6- Indirectly aids in communicating logic through code.
CO2:PO6- Supports team collaboration in developing web projects.
CO3:PO6- Aids communication in terms of user-facing content.
CO4:PO6- Indirect role in team discussions on security practices.
CO5:PO6- Enhances teamwork through modular code design.
CO6:PO6- Collaboration in web and DB integration projects.
CO7:PO6- Improves teamwork and code sharing.

Mapping of PO7 to all COs
CO1:PO7- Minor support to research via foundational knowledge.
CO2:PO7- Helps design research tools like data collection sites.
CO3:PO7- Minor role in data formatting for research tools.
CO4:PO7- Research relevance in secure data collection.
CO5:PO7- Supports research tool modularization.
CO6:PO7- Useful for research involving backend data storage.
CO7:PO7- Structured code helps in research simulations.

Mapping of PO8 to all COs
CO1:PO8- Learning PHP syntax supports independent learning.
CO2:PO8- Strongly aids in learning by doing approach.
CO3:PO8- Helps build learning patterns through examples.
CO4:PO8- Encourages continuous learning due to evolving threats.
CO5:PO8- Promotes continued learning through abstraction.
CO6:PO8- Promotes hands-on learning.
CO7:PO8- Encourages reflective learning and improvement.

Mapping of PO9 to all COs
CO1:PO9- Enhances basic digital literacy through scripting.
CO2:PO9- Enhances use of digital and tech tools.
CO3:PO9- Relates to technical processing and output.
CO4:PO9- Strongly linked to secure tech implementation.
CO5:PO9- Critical for modern object-oriented development.
CO6:PO9- Deepens digital and tech capabilities.
CO7:PO9- Deep tech skill in design patterns and clarity.

Mapping of PO10 to all COs
CO1:PO10- Minimal impact; may help in inclusive tech practices.
CO2:PO10- Can involve inclusive web development practices.
CO3:PO10- Limited role, may aid in content personalization.
CO4:PO10- Ensures inclusive and ethical handling of user data.
CO5:PO10- Minimal direct impact.

CO6:PO10- Inclusion via accessible, data-driven design.
CO7:PO10- Supports inclusive coding through documentation.

Mapping of PO11 to all COs
CO1:PO11- Can support digital solutions for environmental awareness.
CO2:PO11- Web solutions can target environmental causes.
CO3:PO11- Could support formatting eco-focused messages.
CO4:PO11- Promotes values in ethical programming.
CO5:PO11- Object-oriented methods can apply in environmental software.
CO6:PO11- Useful in environmental data tracking.
CO7:PO11- Can be used in reusable environmental tools.

Mapping of PO12 to all COs
CO1:PO12- Slightly related to individual coding responsibility.
CO2:PO12- Requires accountability in code development.
CO3:PO12- Encourages responsibility in data handling.
CO4:PO12- Reinforces accountability in code responsibility.
CO5:PO12- Encourages ownership in design decisions.
CO6:PO12- Requires accountability in DB interactions.
CO7:PO12- High level of responsibility in code clarity.

Mapping of PO13 to all COs
CO1:PO13- Can indirectly support web solutions for community needs.
CO2:PO13- Directly applicable to service-oriented web solutions.
CO3:PO13- Can support basic string processing in web forms.
CO4:PO13- Security is critical in public service platforms.
CO5:PO13- Supports scalable and reusable service applications.
CO6:PO13- Applicable in public service platforms.
CO7:PO13- Reusable code useful for community apps.

SYLLABUS (CBCS as per NEP 2020) FOR T.Y.B. Sc. (Computer Science)
(w. e. from AY 2025-26)

Name of the Programme : B.Sc. Computer Science
Program Code : USCOS
Class : T.Y. B.Sc. (Comp. Sci.)
Semester : V
Course Type : Major Elective (TH)
Course Name : Advanced Java
Course Code : COS-306-MJE(B)
No. of Lectures : 30
No. of Credits : 2

Prerequisites:

• Strong Knowledge of Core Java

Course Objectives:

- To understand Collection
- To understand Threading
- To understand how to design GUI
- To understand database design

Course Outcome:

CO1: Understand and Apply Collection Framework
CO2: Develop Multithreaded Applications
CO3: Design Graphical User Interfaces using Swing
CO4: Implement Event-Driven Programming
CO5: Develop Database-Driven Applications using JDBC
CO6: Analyze and Optimize Data Handling in Java
CO7: Integrate Multithreading, Event Handling, and Database Connectivity for Real-World Solutions

UNIT No.	Chapter Name with Topics	No. of Lectures Required
UNIT- I	**Collection** - Introduction to the Collection framework - List – Array List, Linked List and Vector, Stack, Queue - Set – Hash Set, Tree Set, and Linked Hash Set - Map – Hash Map, Linked Hash Map, Hash table and Tree Map - Interfaces such as Comparator, Iterator, List Iterator, Enumeration	**06**
UNIT- II	**Multithreading** - What are threads? - Life cycle of Thread - Running and starting thread using class and interface - Thread priorities - Running multiple threads - Synchronization and interthread communication	**08**
UNIT-III	**User Interface Components with Swing** - What is Swing? - The MVC Architecture and Swing - Layout Manager and Layouts, The JComponent class - Different components from Swing	**08**

	- Dialogs (Message, confirmation, input), JFileChooser, JColorChooser - Event Handling: Event sources, Listeners - Mouse and Keyboard Event Handling - Adapters and Anonymous inner class	
UNIT-IV	**Database Programming** - The design of JDBC, JDBC configuration - Types of Drivers - Executing SQL statements, query execution - Scrollable and updatable result sets - Metadata – DatabaseMetadata, ResultSetMetadata - Transactions – commit(), rollback(), SavePoint	**08**

Mapping of POs with All COs

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PO13
CO1	3	3	2	3	3	2	2	3	3	1	1	2	1
CO2	3	3	3	3	3	2	2	3	3	1	1	3	2
CO3	3	3	3	3	3	2	2	2	3	2	1	3	2
CO4	3	3	3	3	3	2	2	2	3	2	1	3	2
CO5	3	3	3	3	3	2	2	3	3	2	1	3	2
CO6	3	3	2	3	3	2	3	3	3	1	1	3	2
CO7	3	3	3	3	3	3	3	3	3	2	1	3	3

Justification:

PO1 with all COs

CO1–CO7: Each CO contributes to a solid understanding of Java's core areas like collections, multithreading, GUI design, JDBC, and event handling. Together, they provide a well-rounded conceptual foundation in Java application development.

PO2 with all COs

CO1–CO7: Students gain hands-on experience with frameworks, concurrency, event-driven interfaces, and database programming. Each CO fosters real-world coding practices, debugging, and structured implementation techniques.

PO3 with all COs

CO1–CO7: Integrating multithreading, event handling, and databases (CO7), designing UIs (CO3), and applying data optimization (CO6) are essential for innovative software product development, nurturing entrepreneurial thinking.

PO4 with all COs

CO1–CO7: Learners acquire skills in advanced Java techniques—threads, collections, GUIs, JDBC—enabling them to solve specialized and domain-specific problems effectively.

PO5 with all COs

CO1–CO7: Each CO involves designing or building something functional. From structuring data (CO1) to solving concurrency challenges (CO2, CO7), analytical reasoning is applied throughout.

PO6 with all COs CO1–CO7: Group projects or collaborative debugging enhance technical communication. CO7 especially demands integration of multiple systems, promoting teamwork and effective communication.

PO7 with all COs
CO1–CO7: CO6 and CO7 push learners to explore performance analysis and integration strategies—skills aligned with research and experimentation. Others support understanding of current technologies and best practices.

PO8 with all COs
CO1–CO7: As learners explore diverse topics—from GUIs to multithreading and JDBC—they develop the ability to self-learn, troubleshoot, and adapt to unfamiliar technologies.

PO9 with all COs
CO1–CO7: Every CO utilizes contemporary software development tools and practices, improving proficiency in using IDEs, APIs, libraries, and frameworks—core to digital fluency.

PO10 with all COs
CO1–CO7: While technical in nature, tasks such as UI/UX design (CO3, CO4) can be designed with inclusivity and accessibility in mind. Collaborative projects further promote empathy and mutual respect.

PO11 with all COs
CO1–CO7: Although not directly linked, CO7 may allow for integration of values-based application development (e.g., educational or environmental data systems), fostering ethical coding practices.

PO12 with all COs
CO1–CO7: Learners are expected to build, test, and debug independently, cultivating responsibility in writing clean, maintainable code and managing their learning process.

PO13 with all COs
CO1–CO7: CO7 especially enables students to create practical tools or systems that may serve the community. All COs offer foundational skills to design helpful, service-oriented applications.

SYLLABUS (CBCS as per NEP 2023) FOR B. Sc. (Computer Science) Sem-V
(w. e. f. A.Y 2025-26)

Name of the Program	: B.Sc. Computer Science
Program Code	: USCOS
Class	: T.Y. B.Sc. (Computer Science)
Semester	: V
Course Type	: Major Elective (TH)
Course Name	: Introduction to Blockchain
Course Code	: COS-306- MJE (C)
No. of Lectures	: 30
No. of Credits	: 02

Course Objectives:

- Understand the basic concepts of blockchain technology, including its definition, purpose, and core principles.
- Learn the history and evolution of blockchain, from Bitcoin to modern blockchain platforms.
- Explore key components of blockchain, such as blocks, transactions, nodes, and ledgers.
- Understand cryptographic techniques used in blockchain, including hashing and digital signatures.
- Learn about different types of blockchains (public, private, and consortium) and their use cases.
- Understand the role of smart contracts and their applications in automating transactions.
- Understand emerging trends in blockchain.

Course Outcomes:

CO 1 - Explain the fundamentals of blockchain technology and how it works.
CO 2 - Describe the structure of a blockchain, including blocks, transactions, and the distributed ledger.
CO 3 - Develop and deploy smart contracts using Solidity on Ethereum or other blockchain platforms.
CO 4 - Evaluate the suitability of blockchain solutions for various industry applications, such as supply chain, healthcare, and finance.
CO 5 - Identify and mitigate security risks associated with blockchain networks and smart contracts.
CO 6 - Compare blockchain scalability solutions and implement strategies to enhance performance.
CO 7 - Demonstrate knowledge of emerging blockchain trends such as decentralized finance (DeFi), NFTs, and the role of blockchain in Web3.

Units	Contents	No of Lectures
Unit 1	**Introduction to Block Chain Technology** • Basic ideas behind blockchain • how it is changing the landscape of digitalization • introduction to cryptographic concepts • Hashing • Public key cryptosystems • Private Vs. public blockchain and use cases • Hash Puzzles	**6**
Unit 2	**..1. Block Chain Fundamentals and Components** • Basic architecture of Block Chain • Different terminologies associated Characteristics of Block chain • Types of networks • Introducing Smart contract concept in Block chain • Core components of Block Chain • Types of Block chain • Block chain Protocol • Permission & Permission less Block chains • Applications of Block Chain Technology • Few case studies on Block Chain Technology	**10**
Unit 3	**Digital Ledger** • Short History of Money and Trust • Bitcoin Mechanics • Other Applications of Blockchains • Introduction to Hyperledger • Hyperledger Fabric and its architecture • Hyperledger Composer	**7**
Unit 4	**Emerging Trends in Block Chain** • Cloud-based blockchain • Multichain • Geth • Stellar • Ripple • R3 Corda • Blockchain API • Blockchain Sandboxes	**7**

Online Links:

https://www.w3schools.com/training/aws/introduction-to-blockchain.php

Reference Books:

- Artemis Caro, "Blockchain: The Beginners Guide to Understanding the Technology Behind Bitcoin & Cryptocurrency".
- Scott Marks, "Blockchain for Beginners: Guide to Understanding the Foundation and Basics of the Revolutionary Blockchain Technology"
- Mark Watney, "Blockchain for Beginners".
- Alwyn Bishop, "Blockchain Technology Explained".

Course Outcomes	Programme Outcomes (POs)												
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PO13
CO1	3	2	2	2	2	3	2	2	2	2	2	2	1
CO2	3	2	2	2	2	3	2	2	2	2	2	2	1
CO3	3	3	3	3	3	2	3	3	3	3	3	3	2
CO4	3	3	3	3	3	3	3	3	3	3	3	3	3
CO5	3	3	3	3	3	2	3	3	3	3	3	3	2
CO6	2	3	3	3	3	2	3	3	3	3	3	3	2
CO7	2	2	3	3	3	3	3	3	3	3	3	2	2

Weight: 1 - Partially related 2 - Moderately Related 3 - Strongly related

Course Objectives (CO) and Program Outcomes (PO) Mapping:

Justification of PO1 to ALL COs :

CO 1 – PO 1 : Provides foundational knowledge of blockchain, including its principles, consensus mechanisms, and cryptographic security. Essential for understanding the broader applications of blockchain.
CO 2– PO 1 : Builds on fundamental knowledge by explaining how blockchain operates at a technical level, including data storage and validation.
CO 3 – PO 1 : Introduces practical application of blockchain concepts through coding and implementation, enhancing technical competency.
CO 4 – PO 1 : Expands understanding by applying blockchain knowledge to real-world multidisciplinary contexts, demonstrating its impact across industries.
CO 5 – PO 1 : Strengthens understanding of blockchain security challenges and methodologies to prevent vulnerabilities, a critical aspect of blockchain expertise.
CO 6 – PO 1 : Demonstrates knowledge of advanced blockchain topics, including performance optimization and scalability, which are crucial for real-world applications.
CO 7– PO 1 : Broadens knowledge beyond fundamental principles to include the latest industry innovations, ensuring graduates remain relevant in a rapidly evolving field.

Justification of PO2 to ALL COs :

CO 1 – PO 2 : Provides foundational knowledge necessary for understanding blockchain's role in professional settings but does not directly involve hands-on application.
CO 2 – PO 2 : Essential for grasping blockchain operations, which is crucial for industry applications, but focuses more on theory than direct professional practice.
CO 3 – PO 2 : Involves hands-on technical skills in blockchain development, directly applicable to industry standards and professional blockchain development.
CO 4 – PO 2 : Involves applying blockchain concepts to real-world industry scenarios, making it highly relevant for professional knowledge and decision-making.
CO 5 – PO 2 : Directly applicable to industry best practices in blockchain security, addressing critical professional concerns in cyber security and compliance.
CO 6 – PO 2 : Involves understanding and implementing scalability solutions, which is a key concern for enterprises deploying blockchain at scale.
CO 7 – PO 2 : Important for staying up-to-date with industry trends, but less directly linked to professional procedures and regulatory frameworks.

Justification of PO3 to ALL COs :

CO 1 – PO 3 : Provides the foundational knowledge necessary to explore business opportunities using blockchain technology but does not directly focus on entrepreneurship.
CO 2 – PO 3 : Helps entrepreneurs understand how blockchain operates, which is crucial for designing innovative blockchain-based solutions. However, it is more technical than business-oriented.
CO 3 – PO 3 : Directly enables entrepreneurs to create blockchain-based applications, automate transactions, and develop business models like decentralized finance (DeFi) and NFTs.
CO 4 – PO 3 : Essential for entrepreneurs to identify market opportunities, apply blockchain solutions to industries, and develop innovative business strategies.
CO 5 – PO 3 : Important for business sustainability, ensuring secure blockchain-based startups, reducing financial risk, and maintaining trust in blockchain solutions.
CO 6 – PO 3 : Helps entrepreneurs understand how to scale blockchain applications, a critical factor for successful business adoption and market growth.
CO 7 – PO 3 : Essential for identifying new business opportunities, fostering innovation, and leveraging blockchain trends to create competitive advantages.

Justification of PO4 to ALL COs :

CO 1 – PO 4 : Provides foundational knowledge required for developing specialized skills in blockchain but is more theoretical in nature.
CO 2 – PO 4 : Enhances analytical abilities by breaking down blockchain architecture, but it is still more theoretical than skill-based.
CO 3 – PO 4 : Strongly related, as it builds technical proficiency in blockchain development, problem-solving, and practical coding skills.
CO 4 – PO 4 : Enhances analytical and problem-solving skills by requiring students to assess real-world challenges and develop blockchain-based solutions.
CO 5 – PO 4 : Directly develops technical and analytical skills by addressing blockchain vulnerabilities, problem-solving security issues, and ensuring data integrity.
CO 6 – PO 4 : Develops technical problem-solving abilities and requires innovation in optimizing blockchain performance.
CO 7 – PO 4: Encourages adaptability and innovation by staying updated with industry trends and emerging blockchain applications.

Justification of PO5 to ALL COs :

CO 1 – PO 5 : Establishes foundational knowledge necessary for problem-solving in blockchain but does not involve direct application.

CO 2 – PO 5 : Helps develop analytical reasoning by understanding how blockchain works, but it is more theoretical than problem-solving-oriented.

CO 3 – PO 5 : Strongly related as it requires practical application, problem-solving, and analytical thinking to write, test, and optimize smart contracts.

CO 4 – PO 5 : Develops critical thinking and problem-solving skills by assessing blockchain's impact and feasibility in different industries.

CO 5 – PO 5 : Requires analytical reasoning to assess vulnerabilities, solve security issues, and apply risk management strategies in blockchain applications.

CO 6 – PO 5 : Strongly related as it requires problem-solving and adaptability to optimize blockchain systems for performance and efficiency.

CO 7 – PO 5 : Encourages critical thinking, adaptability, and creativity by exploring new blockchain trends and applications.

Justification of PO6 to ALL COs :

CO 1 – PO 6 : Strongly related as it requires clear communication of blockchain concepts to technical and non-technical audiences, fostering collaboration in teams.

CO 2 – PO 6 : Requires students to communicate technical blockchain architecture effectively, which is essential for collaboration in blockchain development teams.

CO 3 – PO 6 : Moderately related, as effective collaboration is required when developing and debugging smart contracts within a team. However, it is more technical than communication-focused.

CO 4 – PO 6 : Strongly related, as students must analyze and present findings on blockchain applications, requiring communication with stakeholders from different industries.

CO 5 – PO 6 : Moderately related, as students need to communicate security risks and collaborate with cybersecurity teams, but the focus is primarily technical

CO 6 – PO 6 : Moderately related, as students may need to discuss and present scalability **solutions** within a team but mostly focus on technical implementations.

CO 7 – PO 6 : Strongly related, as students must communicate emerging trends and their impact to different audiences, fostering discussion and collaboration.

Justification of PO7 to ALL COs :

CO 1 – PO 7 : Provides foundational knowledge essential for formulating research questions in blockchain technology. However, it is more theoretical, with less focus on direct research methodologies.

CO 2 – PO 7 : Supports research inquiry by enabling students to understand blockchain architecture, which is crucial for identifying key research questions and exploring new methodologies.

CO 3 – PO 7 : Strongly related, as students will need research skills to explore smart contract development, evaluate existing tools, and experiment with different frameworks. Research skills will also be needed to analyze code effectiveness and scalability.

CO 4 – PO 7 : Directly related, as it involves researching real-world problems and identifying how blockchain can be applied effectively in different industries. This requires data collection, analysis, and inquiry into industry-specific needs.

CO 5 – PO 7 : Involves researching vulnerabilities, developing hypotheses, and testing mitigation strategies. Requires understanding existing research, analyzing risks, and applying methodologies to safeguard blockchain systems.

CO 6 – PO 7 : Directly involves research and experimentation with scalability solutions, requiring data analysis, performance evaluation, and the ability to compare and contrast methodologies.

CO 7 – PO 7 : Requires research on emerging trends in blockchain, identifying gaps in current solutions and exploring new methodologies for integration and application of Web3 concepts.

Justification of PO8 to ALL COs :

CO 1 – PO 8 : While essential for building foundational knowledge, this outcome is more theoretical and sets the stage for self-directed learning but doesn't emphasize independent learning or goal-setting.

CO 2 – PO 8 : Provides essential background knowledge but requires ongoing independent exploration of blockchain structures, potentially involving self-study of more complex topics.

CO 3 – PO 8 : Highly related to self-directed learning, as it requires students to independently develop coding skills (e.g., Solidity), learn to troubleshoot, and adapt to emerging development tools**.**

CO 4 – PO 8 : Encourages independent learning and critical thinking to research real-world blockchain applications, adapt knowledge to specific industries, and set independent goals for solutions.

CO 5 – PO 8 : Involves self-directed learning to keep up with security challenges and new strategies for mitigating risks in blockchain. It requires continuous independent problem-solving.

CO 6 – PO 8 : Encourages independent research, experimenting with scalability solutions, and adapting to performance challenges, which are core to learning how to learn in a dynamic field.

CO 7 – PO 8 : Requires students to continually learn about new trends and adapt to rapid technological changes, fostering independent research and goal-setting in an evolving field.

Justification of PO9 to ALL COs :

CO 1 – PO 9 : Provides foundational knowledge that is necessary for using digital tools in blockchain technology but does not directly involve the use of specialized software or digital platforms.

CO 2 – PO 9 : Supports understanding of blockchain architecture, which is crucial for working with blockchain platforms and analyzing data. However, it is more conceptual and does not require direct use of software tools.

CO 3 – PO 9 : Strongly related, as this requires proficiency in digital tools, programming languages (e.g., Solidity), and blockchain platforms (Ethereum), which are essential technological skills for blockchain development.

CO 4 – PO 9 : Involves analyzing industry-specific data and utilizing digital tools to assess the potential of blockchain solutions in various sectors. It may involve using tools to model solutions, conduct analysis, and research.

CO 5 – PO 9 : Strongly related, as digital tools for security testing (e.g., penetration testing software, vulnerability scanners) are required to identify and mitigate risks within blockchain systems and smart contracts.

CO 6 – PO 9 : Involves using digital tools for performance analysis and scalability testing. Students will need to compare different blockchain platforms and scalability solutions using software tools for benchmarking and optimization.

CO 7 – PO 9 : Requires keeping up-to-date with digital tools and platforms used in emerging blockchain applications like DeFi platforms**,** NFT marketplaces, and Web3 technologies, which requires familiarity with various blockchain tools.

Justification of PO10 to ALL COs :

CO 1 – PO 10 : This outcome is primarily technical, but an understanding of blockchain's global application in diverse sectors (e.g., cross-border payments, international supply chains) encourages students to appreciate the cultural and international contexts in which blockchain is used.

CO 2 – PO 10 : Similar to CO1, this is technical in nature. However, blockchain's decentralized nature and its potential to bridge global gaps in access and equality make it relevant for fostering empathy and inclusiveness when designing and implementing blockchain solutions in various cultural settings.

CO 3 – PO 10 : Strongly related to inclusive leadership, as blockchain solutions (especially smart contracts**)** can impact global communities. Developers need to consider multicultural implications of contract terms and accessibility across diverse regions and legal frameworks.

CO 4 – PO 10 : This outcome is strongly related to empathy and multicultural competence because blockchain applications often address global challenges (e.g., supply chain transparency, equitable healthcare). Understanding the needs and perspectives of diverse cultures and communities is essential for effective application.

CO 5 – PO 10 : Security risks in blockchain, such as fraud, privacy violations, and ethical concerns**,** can have far-reaching consequences across cultures and regions. A global understanding of these risks ensures empathy for vulnerable populations impacted by security failures.

CO 6 – PO 10 : Scalability solutions are often implemented to serve diverse regions and address varying infrastructural needs. This outcome is related to inclusive thinking because it requires considering how to scale blockchain solutions in a way that works for underrepresented or marginalized groups globally.

CO 7 – PO 10 : Emerging trends like DeFi and NFTs have global significance, especially in terms of democratizing access to finance and creative industries. This outcome is strongly related to cultural competence as it involves understanding how global communities interact with these technologies and the ethical considerations in their adoption.

Justification of PO11 to ALL COs :

CO 1 – PO 11 : This outcome provides foundational knowledge that is necessary for understanding the ethical implications of blockchain. It enables students to explore the decentralized nature of blockchain, which can challenge traditional power structures and potentially promote more ethical financial and social systems. However, it does not directly address environmental awareness or sustainability.

CO 2 – PO 11 : Understanding blockchain structure can lead to ethical considerations such as the transparency of transactions, the immutability of data, and the potential for fairer systems**.** However, the direct link to environmental awareness is not very strong in this outcome.

CO 3 – PO 11 : Smart contracts have the potential to reduce human error**,** increase transparency**,** and reduce fraud, which aligns with ethical principles. Furthermore, understanding how to deploy contracts on Ethereum or similar platforms can also involve an ethical responsibility related to the energy consumption and environmental impact of blockchain networks, particularly Proof of Work (PoW**)** systems.

CO 4 – PO 11 : This CO is directly related to value inculcation and ethical practice, as blockchain applications can help ensure fair trade**,** supply chain transparency, and data integrity. Additionally, blockchain in supply chains can significantly improve environmental sustainability by tracking and reducing waste and carbon footprints in industries like food production and manufacturing.

CO 5 – PO 11 : The identification and mitigation of security risks are directly tied to ethical responsibility and safety in the blockchain space. Protecting sensitive data and ensuring that blockchain applications are secure and non-exploitative is an important ethical responsibility. Furthermore, security in blockchain applications contributes to sustainable and responsible digital ecosystems**.**

CO 6 – PO 11 : Scalability solutions are closely tied to resource management and energy efficiency**.** In the context of blockchain**,** sustainability can be enhanced by adopting energy-efficient consensus mechanisms (e.g., Proof of Stake, instead of Proof of Work), which can have a positive environmental impact**.** This outcome aligns with environmental awareness through the evaluation of blockchain's energy footprint**.**

CO 7 – PO 11 : Emerging trends like DeFi**,** NFTs, and Web3 are redefining ethical paradigms, particularly regarding financial inclusion**,** democratization of access, and digital ownership**.** DeFi in particular has the potential to create more inclusive financial systems, which can align with ethical citizenship. However**,** environmental considerations (e.g., energy consumption in NFTs**)** are important in this context, especially given the debate around the sustainability of blockchain networks.

Justification of PO12 to ALL COs :

CO 1 – PO 12 : This CO involves understanding and explaining fundamental concepts independently. The graduate will need to communicate their understanding of blockchain technology, showcasing accountability in ensuring the accuracy of their explanations.

CO 2 – PO 12 : Describing blockchain's structure requires the independent application of knowledge and the responsibility to clearly articulate how each element works. Graduates need to ensure their explanations are precise and accurate, reflecting responsibility in learning.

CO 3 – PO 12 : Developing and deploying smart contracts independently is a highly relevant outcome for PO 12. It demonstrates both technical proficiency and personal responsibility, as the graduate must manage the entire lifecycle of smart contracts, from development to deployment, while considering their implications.

CO 4 – PO 12 : Evaluating the suitability of blockchain solutions requires independent research, critical thinking, and decision-making. Graduates need to take responsibility for their evaluation, considering how these solutions will impact real-world industries.

CO 5 – PO 12 : Identifying and mitigating security risks is a key responsibility for any blockchain professional. Graduates must independently assess vulnerabilities and take accountability for implementing effective security measures, ensuring safe blockchain usage.

CO 6 – PO 12 : Comparing scalability solutions and implementing strategies requires a high level of independent problem-solving and accountability. Graduates are responsible for ensuring that scalability improvements align with best practices while managing the complexity of performance optimization.

CO 7 – PO 12 : Demonstrating knowledge of emerging trends involves independently keeping up with new developments and taking responsibility for staying informed. Graduates should also communicate their understanding of how these trends fit into the broader blockchain ecosystem, reflecting accountability for their continuous learning.

Justification of PO13 to ALL COs :

CO 1 – PO 13 : Explaining the fundamentals of blockchain technology can contribute to raising awareness within communities about the impact of blockchain. However, it does not directly involve community engagement or societal service. It's more focused on theoretical knowledge, which indirectly may benefit broader society by informing individuals or organizations about the technology.

CO 2 – PO 13 : Describing blockchain's structure can be helpful in community settings, especially when engaging in educational outreach to explain how blockchain could impact various sectors. However, like CO 1, it is primarily a knowledge-based skill rather than direct involvement in community engagement or societal service.

CO 3 – PO 13 : Developing and deploying smart contracts has the potential to promote societal well-being, especially in sectors such as healthcare, finance, and government, where automation, transparency, and security are needed. If blockchain solutions are designed with social impact in mind (e.g., enabling better access to services or improving transparency in public governance), this would align more strongly with community engagement.

CO 4 – PO 13 : Evaluating blockchain solutions for applications like healthcare, supply chain, and finance directly impacts societal well-being. Blockchain can help increase transparency, reduce fraud, and improve access to services, which contributes to the betterment of society. Graduates could apply this evaluation in community-driven projects that address issues like poverty, access to healthcare, or supply chain inefficiencies.

CO 5 – PO 13 : Identifying and mitigating security risks is important for ensuring that blockchain solutions are safe for public use. A focus on security can help protect vulnerable communities from scams or misuse of blockchain applications. It also indirectly promotes societal well-being by ensuring that blockchain-based services, which may be used for financial transactions or healthcare, are secure.

CO 6 – PO 13 : Improving blockchain scalability has the potential to expand blockchain's applications in public service systems, financial inclusion, and other community-serving platforms. For instance, increasing blockchain performance can help scale decentralized finance (DeFi) applications that provide financial services to underserved populations. However, this is more technical in nature and doesn't directly engage with community service.

CO 7 – PO 13 : Demonstrating knowledge of trends like DeFi and NFTs can directly benefit community engagement efforts. For example, understanding DeFi could lead to solutions for financial inclusion, while NFTs could be used for community-driven projects in art or charity. Understanding how blockchain fits into the emerging Web3 paradigm could also offer opportunities for decentralized solutions that benefit society.

SYLLABUS For T. Y. B. Sc. (Computer Science) Sem-V
(CBCS as per NEP 2020) (2023 Pattern)(w. e. from June, 2025)

Name of the Programme	**:** For T.Y. UG Sem-V
Program Code	: USCOS
Class	**:** T.Y.U.G.
Semester	**:** V
Course Type	**:** Minor for TY UG (TH)
Course Name	**:** Web Design Using WordPress
Course Code	**:** COS-311-MN (D)
No. of Lectures	**: 30**
No. of Credits	**: 02**

Prerequisites:

- Understand the self – hosted CMS tool.
- Learn to manage contents and setup a blog on WordPress.

Course Objectives:

- To understand the basics of WordPress and its functionalities.
- To learn to install and set up a WordPress website.
- To develop proficiency in creating and managing different types of content in WordPress, such as pages, posts, and media.
- To explore and customize WordPress themes to create visually appealing websites.
- To gain knowledge of various WordPress plugins and their functionalities.
- To learn to optimize and secure WordPress websites for better performance and protection against cyber threats.
- To understand search engine optimization (SEO) techniques and apply them to WordPress websites.

Course Outcomes:

CO1. Ability to set up and configure a WordPress website from scratch.

CO2. Proficiency in creating and managing different types of content in WordPress.

CO3. Understanding of WordPress themes and ability to customize them according to specific requirements.

CO4. Knowledge of various WordPress plugins and ability to utilize them to enhance website functionality.

CO5. Ability to optimize and secure WordPress websites for better performance and protection against cyber threats.

CO6. Understanding of SEO techniques and ability to apply them to WordPress websites to improve search engine visibility.

CO7. Proficiency in using WordPress analytics tools to monitor and analyze website traffic

Unit	Title and Contents	No. of Lectures
Unit 1	**Introduction to WordPress** .1. What is WordPress? Overview and history .2. Understanding CMS (Content Management System) and WordPress's role .3. Features ,advantages, and disadvantages .4. The WordPress ecosystem: themes, plugins, community, and resources .5. System Requirements for Wordpress .6. Download & Installing WordPress	7
Unit 2	**Setting Up a WordPress Website** **2.1** Choosing a domain and web hosting for WordPress **2.2** WordPress - General settings **2.3** WordPress - Writing Setting **2.4** WordPress -Reading Setting **2.5** WordPress - Media Setting **2.6** WordPress - Permalink **2.7** Basic WordPress setup and configuration **2.8** Overview of the WordPress Dashboard	8
Unit 3	**Working with WordPress Categories ,Post ,Media & Pages** **3.1** WordPress- Add ,Edit ,Delete ,Arrange Category **3.2** WordPress-Add, Edit , Delete , Preview ,Publish Post **3.3** WordPress Media-Library, Add Media ,Insert Media **3.4**WordPress Pages : Add Pages ,Title Wysiwyg Editor, Text Insertion , Publish Pages , Edit Pages , Delete Pages. **3.5** WordPress Tags ,Links	6
Unit 4	**WordPress Themes and Design &Plugins** **4.1** Understanding WordPress themes and how they work **4.2** Exploring free vs. premium themes **4.3** Installing and activating themes **4.4** Customizing themes through the WordPress Customizer **4.5** Theme options and settings overview **4.6** Page builders and their uses **4.7** What are plugins? Introduction to WordPress plugins **4.8** Searching, installing, and activating plugins **4.9** Recommended plugins for performance, security, functionality **4.10** Managing plugins: updates, activation, and deactivation	9

References:

- WordPress for Beginners 2019: A Visual Step-by-Step Guide to Mastering WordPress, Dr. Andy Williams.
- "E-Commerce Platform Architecture: Concepts and Applications" by João Gama and Manuel C. Oliveira
- "Shopify for Dummies" by Paul Mladjenovic
- "Building E-Commerce Sites with WooCommerce: A Complete Guide" by Mark R. Linton
- WordPress All-In-One For Dummies, Lisa Sabin-Wilson 3
https://www.tutorialspoint.com.
- https://www.javatpoint.com

Mapping of this course with Programme Outcomes

Course Outcomes	Programme Outcomes (POs)												
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PO13**
CO1	3	3	1	3	3	2	1	3	3	1	1	3	1
CO2	3	3	1	3	3	2	1	3	3	1	1	3	1
CO3	3	3	1	3	3	2	1	3	3	1	1	3	1
CO4	3	3	1	3	3	2	1	3	3	1	1	3	1
CO5	3	3	1	3	3	2	1	3	3	1	1	3	1
CO6	3	3	1	3	3	3	1	3	3	1	1	3	1
CO7	3	3	1	3	3	2	3	3	3	**1**	**1**	3	**1**

Weight: 1 - Partially related 2 - Moderately Related 3 - Strongly related

Justification of Mapping of PO1 with All CO'S

CO1: PO1: Constructing HTML documents with proper structures and utilizing HTML tags effectively directly demonstrates a profound understanding of foundational theories, principles, and methodologies in web development, aligning strongly with the comprehensive knowledge and understanding of the field of study.

CO2: PO1: Implementing specific HTML5 elements and various layout components requires a deep understanding of foundational concepts and key principles in web development, contributing significantly to the comprehensive knowledge and understanding of the field.

CO3: PO1: Applying the basics of CSS and understanding its role in styling HTML elements enhances the comprehension of foundational theories and methodologies in web design, closely linked with the broader multidisciplinary context of the field of study.

CO4: PO1: Applying CSS rules effectively to control the appearance of HTML elements demonstrates a profound understanding of CSS principles and methodologies, reinforcing the comprehensive knowledge and understanding of the field.

CO5: PO1: Designing web pages using HTML5 and CSS showcases a deep understanding of foundational theories, methodologies, and key concepts in web development, contributing significantly to the broader multidisciplinary context of the field of study.

CO6: PO1: Designing dynamic, interactive, and elegant websites requires a profound understanding of advanced concepts and principles in web development, aligning strongly with the comprehensive knowledge and understanding of the field.

CO7: PO1: Analyzing and exploring web pages to identify their elements and attributes demonstrates a deep understanding of foundational theories and methodologies in web development, reinforcing the comprehensive knowledge and understanding of the field.

Mapping of PO2 with All CO'S

CO1: PO2: Constructing HTML documents with proper structures and utilizing HTML tags effectively aligns closely with practical skills and expertise essential for professional tasks in web development, demonstrating knowledge of industry standards and best practices.

CO2: PO2: Implementing specific HTML5 elements demonstrates practical skills essential for professional tasks in web development, incorporating knowledge of industry standards and best practices to create functional and effective web pages.

CO3: PO2: Applying the basics of CSS to enhance web page aesthetics reflects practical knowledge and expertise in web design, considering industry standards and best practices to create visually appealing websites.

CO4: PO2: Applying CSS rules effectively to control the appearance of HTML elements showcases practical skills in web development, incorporating industry standards and best practices to achieve desired styling outcomes.

CO5: PO2: Designing web pages using HTML5 and CSS demonstrates practical expertise in web development, applying industry standards and best practices to create professional and functional websites.

CO6: PO2: Designing dynamic, interactive, and elegant websites requires practical skills and expertise in web development, incorporating industry standards and best practices to create engaging user experiences.

CO7: PO2: Analyzing and exploring web pages to identify their elements and attributes showcases practical knowledge essential for professional tasks in web development,

incorporating industry standards and best practices to understand and manipulate web content effectively.

Mapping of PO3 with All CO'S

CO1: PO3: Constructing HTML documents demonstrates technical proficiency, which is partially related to fostering innovation and understanding business principles but is not directly linked to entrepreneurial mindset or risk management strategies.

CO2: PO3: Implementing specific HTML5 elements showcases technical skills but is only partially related to identifying opportunities and fostering innovation in entrepreneurial contexts.

CO3: PO3: Applying CSS basics to enhance web page aesthetics is partially related to understanding business principles and market dynamics, but it doesn't directly contribute to cultivating an entrepreneurial mindset.

CO4: PO3: Applying CSS rules to control the appearance of HTML elements is partially related to understanding business principles and market dynamics but doesn't directly contribute to fostering innovation or risk management strategies.

CO5: PO3: Designing web pages using HTML5 and CSS demonstrates technical skills but is only partially related to cultivating an entrepreneurial mindset.

CO6: PO3: Designing dynamic, interactive, and elegant websites may indirectly contribute to fostering innovation and understanding market dynamics but is only partially related to cultivating an entrepreneurial mindset.

CO7: PO3: Analyzing and exploring web pages is partially related to identifying opportunities and understanding market dynamics but is not directly linked to fostering innovation or risk management strategies.

Mapping of PO4 with All CO'S

CO1: PO4: Constructing HTML documents and utilizing HTML tags effectively demonstrates technical proficiency and problem-solving skills, directly related to specialized skills and competencies in web development.

CO2: PO4: Implementing specific HTML5 elements requires technical skills and problem-solving abilities, closely related to specialized competencies in web development.

CO3: PO4: Applying CSS basics to enhance web page aesthetics demonstrates technical proficiency and analytical abilities, directly relevant to specialized skills and competencies in web design.

CO4: PO4: Applying CSS rules effectively to control the appearance of HTML elements showcases technical skills and problem-solving abilities, directly related to specialized competencies in web development.

CO5: PO4: Designing web pages using HTML5 and CSS requires technical proficiency, problem-solving skills, and effective communication, all of which are specialized competencies in web development.

CO6: PO4: Designing dynamic, interactive, and elegant websites showcases technical skills, analytical abilities, and innovation, directly relevant to specialized competencies in web development.

CO7: PO4: Analyzing and exploring web pages demonstrates analytical abilities, problem-solving skills, and adaptability, all of which are specialized competencies in web development.

Mapping of PO5 with All CO'S

CO1: PO5: Constructing HTML documents and utilizing HTML tags effectively requires problem-solving and analytical reasoning, directly linked to the capacity for application and problem-solving.

CO2: PO5: Implementing specific HTML5 elements involves problem-solving and analytical reasoning to select appropriate elements and layouts, directly contributing to the capacity for application and problem-solving.

CO3: PO5: Applying CSS basics to enhance web page aesthetics requires analytical reasoning and adaptability, directly relevant to problem-solving and analytical reasoning in practical web development scenarios.

CO4: PO5: Applying CSS rules effectively to control the appearance of HTML elements involves problem-solving and analytical reasoning to achieve desired outcomes, directly related to the capacity for application and problem-solving.

CO5: PO5: Designing web pages using HTML5 and CSS necessitates problem-solving, analytical reasoning, and creativity, directly contributing to the capacity for application and problem-solving in web development.

CO6: PO5: Designing dynamic, interactive, and elegant websites requires problem-solving, creativity, and adaptability, directly linked to the capacity for application, problem-solving, and analytical reasoning.

CO7: PO5: Analyzing and exploring a web page demonstrates problem-solving, analytical reasoning, and adaptability, directly relevant to the capacity for application, problem-solving, and analytical reasoning in web development contexts.

Mapping of PO6 with All CO'S

CO1: PO6: Constructing HTML documents and utilizing HTML tags effectively may involve communication when collaborating with team members to ensure consistency and clarity in code, moderately related to effective communication in diverse teams.

CO2: PO6: Implementing specific HTML5 elements requires collaboration and communication within teams to discuss layout components and functionalities, moderately related to effective communication and collaboration.

CO3: PO6: Applying CSS basics to enhance web page aesthetics may involve communicating design preferences or requirements, moderately related to effective communication in conveying visual concepts.

CO4: PO6: Applying CSS rules effectively may involve communication with team members to ensure consistency in styling across web pages, moderately related to effective communication and collaboration.

CO5: PO6: Designing web pages using HTML5 and CSS may involve collaboration and communication to discuss design choices and functionalities, moderately related to effective communication and collaboration.

CO6: PO6: Designing dynamic, interactive, and elegant websites often requires collaboration and communication to align on design goals and functionalities, strongly related to effective communication and collaboration.

CO7: PO6: Analyzing and exploring a web page may involve communication within teams to discuss findings and implications for design changes, moderately related to effective communication and collaboration.

Mapping of PO7 with All CO'S

CO1: PO7: Constructing HTML documents and utilizing HTML tags effectively may involve some observational skills when analyzing existing web pages for inspiration or research purposes, partially related to research-related skills.

CO2: PO7: Implementing specific HTML5 elements requires some observational skills when analyzing different websites for functionality and layout ideas, partially related to research-related skills.

CO3: PO7: Applying CSS basics to enhance web page aesthetics involves observational skills when evaluating design choices, partially related to research-related skills.

CO4: PO7: Applying CSS rules effectively may involve some observational skills when examining the impact of styling changes on web page appearance, partially related to research-related skills.

CO5: PO7: Designing web pages using HTML5 and CSS may involve observational skills when researching design trends or analyzing competitor websites, partially related to research-related skills.

CO6: PO7: Designing dynamic, interactive, and elegant websites involves observational skills when researching user preferences and behaviors, partially related to research-related skills.

CO7: PO7: Analyzing and exploring a web page demonstrates strong observational and inquiry skills, directly related to research-related skills in data collection and analysis.

Mapping of PO8 with All CO'S

CO1: PO8: Constructing HTML documents and utilizing HTML tags effectively requires self-directed learning and goal setting to acquire and apply new knowledge independently.

CO2: PO8: Implementing specific HTML5 elements involves self-directed learning and adaptability to learn new techniques and functionalities independently.

CO3: PO8: Applying the basics of CSS and understanding its role in styling HTML elements necessitates self-directed learning and goal setting to master CSS techniques independently.

CO4: PO8: Applying CSS rules effectively requires self-directed learning and adaptability to keep up with evolving CSS standards and techniques independently.

CO5: PO8: Designing web pages using HTML5 and CSS involves self-directed learning and goal setting to integrate HTML and CSS effectively to achieve design goals independently.

CO6: PO8: Designing dynamic, interactive, and elegant websites requires continuous self-directed learning and adaptability to incorporate new technologies and design trends independently.

CO7: PO8: Analyzing and exploring a web page demonstrates self-directed learning and adaptability to acquire and apply analytical skills independently.

Mapping of PO9 with All CO'S

CO1: PO9: Constructing HTML documents and utilizing HTML tags effectively demonstrate proficiency in using ICT and accessing information sources to create web content.

CO2: PO9: Implementing specific HTML5 elements involves proficiency in using ICT and accessing information sources to select and integrate appropriate elements into web pages.

CO3: PO9: Applying the basics of CSS to enhance web page aesthetics requires proficiency in using ICT and accessing information sources to learn and apply styling techniques.

CO4: PO9: Applying CSS rules effectively involves proficiency in using ICT to manipulate and control the appearance of HTML elements.

CO5: PO9: Designing web pages using HTML5 and CSS demonstrates proficiency in using ICT to create functional and aesthetically pleasing websites.

CO6: PO9: Designing dynamic, interactive, and elegant websites showcases proficiency in using ICT to integrate advanced features and technologies into web design.

CO7: PO9: Analyzing and exploring a web page demonstrates proficiency in using ICT to navigate and understand web content and identify its elements and attributes.

Mapping of PO10 with All CO'S

CO1: PO10: Constructing HTML documents and utilizing HTML tags effectively doesn't directly involve engagement in multicultural settings or demonstrating empathy and understanding of diverse perspectives.

CO2: PO10: Implementing specific HTML5 elements may involve collaboration with diverse teams, but it doesn't inherently demonstrate multicultural competence or empathy.

CO3: PO10: Applying CSS to enhance web page aesthetics doesn't directly relate to engagement in multicultural settings or demonstrating empathy.

CO4: PO10: Applying CSS rules effectively also doesn't directly involve engagement in multicultural settings or demonstrating empathy.

CO5: PO10: Designing web pages using HTML5 and CSS may involve considering diverse user perspectives, but it doesn't inherently demonstrate engagement in multicultural settings or empathy.

CO6: PO10: Designing dynamic, interactive, and elegant websites may involve collaboration with diverse teams, but it doesn't directly demonstrate multicultural competence or empathy.
CO7: PO10: Analyzing and exploring a web page may involve considering diverse user perspectives, but it doesn't inherently demonstrate engagement in multicultural settings or empathy.

Mapping of PO11 with All CO'S

CO1: PO11: Constructing HTML documents and utilizing HTML tags effectively doesn't inherently involve embracing ethical and moral values, practicing responsible citizenship, or addressing environmental issues.

CO2: PO11: Implementing specific HTML5 elements may involve considering accessibility standards, but it doesn't directly relate to promoting sustainability or environmental conservation.

CO3: PO11: Applying CSS to enhance web page aesthetics doesn't directly relate to embracing ethical values or promoting environmental awareness.

CO4: PO11: Applying CSS rules effectively also doesn't directly involve embracing ethical values or promoting environmental awareness.

CO5: PO11: Designing web pages using HTML5 and CSS may involve considering the environmental impact of design choices, but it doesn't inherently promote sustainability or environmental conservation.

CO6: PO11: Designing dynamic, interactive, and elegant websites may involve considering user experience, but it doesn't directly relate to embracing ethical values or promoting environmental awareness.

CO7: PO11: Analyzing and exploring a web page may involve considering user needs, but it doesn't inherently involve embracing ethical values or promoting environmental awareness.

Mapping of PO12 with All CO'S

CO1: PO12: Constructing HTML documents and utilizing HTML tags effectively demonstrates autonomy and responsibility in applying knowledge and skills independently in web development projects.

CO2: PO12: Implementing specific HTML5 elements requires autonomy and responsibility in managing projects effectively and applying skills independently to achieve project goals.

CO3: PO12: Applying CSS to enhance web page aesthetics demonstrates autonomy and responsibility in independently managing design aspects of web development projects.

CO4: PO12: Applying CSS rules effectively showcases autonomy and responsibility in managing the appearance of HTML elements independently within web development projects.

CO5: PO12: Designing web pages using HTML5 and CSS requires autonomy and responsibility in independently managing the entire web development process.

CO6: PO12: Designing dynamic, interactive, and elegant websites demonstrates autonomy and responsibility in managing complex web development projects independently.

CO7: PO12: Analyzing and exploring a web page showcases autonomy and responsibility in independently evaluating and understanding web content within work and learning contexts.

Mapping of PO13 with All CO'S

CO1: PO13: Constructing HTML documents and utilizing HTML tags effectively doesn't directly involve community engagement or promoting societal well-being.

CO2: PO13: Implementing specific HTML5 elements may involve designing web pages for community organizations or initiatives, but it doesn't inherently involve active participation in community-engaged services and activities.

CO3: PO13: Applying CSS to enhance web page aesthetics doesn't directly relate to community engagement or promoting societal well-being.

CO4: PO13: Applying CSS rules effectively also doesn't directly involve community engagement or promoting societal well-being.

CO5: PO13: Designing web pages using HTML5 and CSS may involve creating websites for community projects, but it doesn't inherently involve active participation in community-engaged services and activities.

CO6: PO13: Designing dynamic, interactive, and elegant websites may involve creating platforms for community initiatives, but it doesn't directly demonstrate active participation in community-engaged services and activities.

CO7: PO13: Analyzing and exploring a web page may involve evaluating community-related content, but it doesn't inherently involve active participation in community-engaged services and activities.

SYLLABUS (CBCS as per NEP 2023) FOR T.Y.B.Sc.(Computer Science)
Sem-V (w. e. from June, 2025)

Name of the Programme :B.Sc.Computer Science
Program Code :USCOS
Class :T. Y. U.G.
Semester : V
Course Type :Minor(PR)
Course Name :Lab Course based on COS-311-MN(D)
Course Code :COS-312-MN(D)
No. of Lectures :60
No. of Credits :02

Prerequisites:

Basic knowledge of computers and its concepts.

Course Objectives:

- To learn to install and setup WordPress website:.
- To Create and Customize WordPress Themes:.
- To develop proficiency increating and managing different types of content in WordPress, such as pages, posts, and media.
- ToexploreandcustomizeWordPressthemestocreatevisuallyappealing websites.
- TogainknowledgeofvariousWordPresspluginsandtheirfunctionalities.
- To Modify theme or page elements to make site responsive and mobile-friendly.
- To understand search engine optimization (SEO) techniques and apply them to WordPress

Course Outcomes:

CO1. Ability to set up and configure a WordPress website from scratch.

CO2. Proficiency in creating and managing various types of content in WordPress.

CO3.Understanding of WordPress themes and ability to customize them according to specific requirements.

CO4.Knowledge of various WordPress plugins and ability to utilize them to enhance website functionality.

CO5.Ability to optimize and secure WordPress websites for better performance and protection against cyber threats.

CO6. Understanding of SEO techniques and ability to apply them to WordPress websites to improve search engine visibility.

CO7.Proficiency in using WordPress analytics tools to monitor and analyze website traffic

Sr.No.	AssignmentName	No.ofPractical's
1.	WordPress -Installation	1
2.	WordPress Dashboard	1
3.	WordPress General Settings	1
4.	WordPress Writing Settings	1
5.	WordPress Discussion Settings	1
6.	WordPress Categories	1
7.	Publishing Posts	2
8.	Working with WordPress Media	1
9.	Working with WordPress Pages	1
10.	WordPress Tags	1
11.	WordPress Themes and Design	1
12.	Working with Plug-in	1
13.	Case Study	2

Mapping of PO's With CO's

Course Outcomes	ProgrammeOutcomes(POs)												
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PO13
CO1	3	3	1	3	3	2	1	3	3	1	1	3	1
CO2	3	3	1	3	3	2	1	3	3	1	1	3	1
CO3	3	3	1	3	3	2	1	3	3	1	1	3	1
CO4	3	3	1	3	3	2	1	3	3	1	1	3	1
CO5	3	3	1	3	3	2	1	3	3	1	1	3	1
CO6	3	3	1	3	3	3	1	3	3	1	1	3	1
CO7	3	3	1	3	3	2	3	3	3	**1**	**1**	3	**1**

Weight : 1-Partially related 2- Moderately Related 3-Strongly related

Justification of Mapping of PO1 with All CO'S

CO1 With PO1: Setting up and configuring WordPress requires knowledge of the platform and related web technologies but doesn't fully emphasize the broader, multidisciplinary context that PO1 demands.

CO2 With PO1: Creating and managing content in WordPress requires an understanding of the platform's methodology, though it does not necessarily require an in-depth multidisciplinary understanding.

CO3 With PO1: Customizing WordPress themes deeply engages with the principles of web design, user interface, and development. It connects well to PO1 as it requires understanding foundational concepts and methodologies.

CO4With PO1: Knowledge and use of plugins is crucial for extending WordPress functionality and requires understanding not only the platform itself but also broader web technologies and their interaction with WordPress.

CO5 With PO1: While optimization and security involve technical knowledge, they don't fully address the broader multidisciplinary context of PO1, even though they require an understanding of foundational principles.

CO6With PO1:SEO involves understanding algorithms and marketing strategies, which are somewhat related to web development but don't cover a broader multidisciplinary view.

CO7With PO1: Using analytics tools requires knowledge of data analysis techniques and how they fit into broader web development strategies, but doesn't deeply engage with the foundational theories and principles of web development.

Mapping of PO2 with All CO'S

CO1 With PO2 : Setting up and configuring a WordPress website from scratch is a fundamental professional skill. It involves following industry standards for creating websites and requires practical knowledge of web technologies.

CO2 With PO2: Content management in WordPress is a core professional skill. Creating, organizing, and managing content follows industry best practices, making this outcome strongly relevant to PO2.

CO3 With PO2: Customizing WordPress themes involves using professional design and development standards, adhering to best practices to ensure a functional, usable, and visually appealing website. This directly aligns with PO2.

CO4 With PO2: Knowledge of WordPress plugins and their implementation in real-world scenarios requires a professional understanding of how plugins fit into the overall website functionality. It adheres to standards for extending WordPress's capabilities.

CO5 With PO2: Optimizing and securing a WordPress website involves applying practical skills in performance tuning, security measures, and adhering to best practices and ethical standards related to cybersecurity.

CO6 With PO2: SEO techniques are an essential professional skill in web development, requiring knowledge of industry best practices, search engine algorithms, and ethical considerations in digital marketing.

CO7 With PO2: Proficiency in analytics tools involves practical skills in monitoring, interpreting, and applying website data to optimize performance. It requires knowledge of industry standards for analyzing web traffic and making data-driven decisions.

Mapping of PO3 with All CO'S

CO1 With PO3 :Setting up and configuring a WordPress website from scratch is a fundamental professinal skill. It involves following industry standards for creating websites and requires practical knowledge of web technologies.

CO2 With PO3: Content management in WordPress is a core professional skill. Creating, organizing, and managing content follows industry best practices, making this outcome strongly relevant to PO2.

CO3 With PO3: Customizing WordPress themes involves using professional design and development standards, adhering to best practices to ensure a functional, usable, and visually appealing website. This directly aligns with PO2.

CO4 With PO3: Knowledge of WordPress plugins and their implementation in real-world scenarios requires a professional understanding of how plugins fit into the overall website functionality. It adheres to standards for extending WordPress's capabilities.

CO5 With PO3: Optimizing and securing a WordPress website involves applying practical skills in performance tuning, security measures, and adhering to best practices and ethical standards related to cybersecurity.

CO6 With PO3: SEO techniques are an essential professional skill in web development, requiring knowledge of industry best practices, search engine algorithms, and ethical considerations in digital marketing.

CO7 With PO3: Proficiency in analytics tools involves practical skills in monitoring, interpreting, and applying website data to optimize performance. It requires knowledge of industry standards for analyzing web traffic and making data-driven decisions.

Mapping of PO4 with All CO'S

CO1 With PO4 :Setting up and configuring a WordPress website from scratch is a fundamental professinal skill. It involves following industry standards for creating websites and requires practical knowledge of web technologies.

CO2 With PO4: Content management in WordPress is a core professional skill. Creating, organizing, and managing content follows industry best practices, making this outcome strongly relevant to PO2.

CO3 With PO4: Customizing WordPress themes involves using professional design and development standards, adhering to best practices to ensure a functional, usable, and visually appealing website. This directly aligns with PO2.

CO4 With PO4: Knowledge of WordPress plugins and their implementation in real-world scenarios requires a professional understanding of how plugins fit into the overall website functionality. It adheres to standards for extending WordPress's capabilities.

CO5 With PO4: Optimizing and securing a WordPress website involves applying practical skills in performance tuning, security measures, and adhering to best practices and ethical standards related to cybersecurity.

CO6 With PO4: SEO techniques are an essential professional skill in web development, requiring knowledge of industry best practices, search engine algorithms, and ethical considerations in digital marketing.

CO7 With PO4: Proficiency in analytics tools involves practical skills in monitoring, interpreting, and applying website data to optimize performance. It requires knowledge of industry standards for analyzing web traffic and making data-driven decisions.

SYLLABUS (CBCS as per NEP 1.0 2023 Pattern) For T. Y. B. Sc. (Computer Science) Sem-V

(w. e. from June, 2025)

Name of the Programme : B.Sc. Computer Science
Program Code **:** USCOS
Class **:** T. Y. B. Sc. (Computer Science)
Semester **:** V
Course Type **:** Major (PR)
Course Name **:** Lab Course based on COS-306-MJE(A) and COS-306-MJE(B)
Course Code **:** COS-305-MJM
No. of Practical **:** 15
No. of Credits **:** 02

Prerequisites:

• HTML, CSS, Core Java.

Objectives:

- To design dynamic, interactive web pages in PHP.
- To learn the server-side scripting language in PHP.
- To learn database connectivity with PHP
- To learn collections, multithreading in Java.
- To learn AWT in Java
- To learn database connectivity with Java.

Outcome:

CO1. Implement input sanitization and validation techniques while creating programs in PHP.
CO2. Implement regular expressions for strings in PHP.
CO3. Implement various functions related to arrays and sorting functions in PHP.
CO4. Execute SQL operations securely using PDO to interact with databases in PHP.
CO5. Develop database-driven applications using JDBC.
CO6. Analyze and optimize data handling in Java.
CO7. Integrate multithreading, event handling, database connectivity for real-world solutions in Java.

Unit	Contents
1.	**Assignments on COS-306-MJE(A)** • Assignment on function and string • Assignment on arrays • Assignment on object oriented programming • Assignment on database
2.	**Assignments on COS-306-MJE(B)** • Assignment on collection • Assignment on multithreading • Assignment on AWT • Assignment on database

Mapping of this course with Programme Outcomes

Course Outcomes	Programme Outcomes (POs)												
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PO13
CO1	2	3	1	3	3	2	2	2	3	1	2	2	1
CO2	2	3	1	3	3	2	2	2	3	1	2	2	1
CO3	2	3	1	3	3	2	2	2	3	1	1	2	1
CO4	2	3	2	3	3	2	2	2	3	1	2	3	1
CO5	2	3	2	3	3	2	2	2	3	1	2	3	2
CO6	2	3	2	3	3	2	2	2	3	1	2	3	2
CO7	3	3	3	3	3	3	3	3	3	2	2	3	3

Weight: 1 - Partially related 2 - Moderately Related 3 - Strongly related

Justification of mapping

Mapping of PO1 to all COs

CO1:PO1 - Understanding sanitization and validation builds foundational secure coding concepts.
CO2:PO1 - Regex use deepens understanding of string handling.
CO3:PO1 - Array and sorting functions require a sound understanding of basic algorithms.
CO4:PO1 - Secure SQL operations need core knowledge of database interaction.
CO5:PO1 - JDBC operations build understanding of backend data flow.
CO6:PO1 - Data handling needs solid programming logic understanding.
CO7:PO1 - Integrating threads, events, and databases requires deep conceptual knowledge.

Mapping of PO2 to all COs

CO1:PO2 - Involves correct use of validation procedures.
CO2:PO2 - Regex implementation is procedural and practice-oriented.
CO3:PO2 - Array handling involves structured logic and syntax.
CO4:PO2 - PDO is used in a secure procedural workflow.
CO5:PO2 - JDBC requires procedural steps to create functional DB applications.
CO6:PO2 - Optimization relies on practical Java techniques.
CO7:PO2 - Real-world integration demands comprehensive procedural skills.

Mapping of PO3 to all COs

CO1:PO3 - Clean code practices are valued in product design.
CO2:PO3 - Regex helps streamline data input in applications.
CO3:PO3 - Algorithmic thinking supports efficient product logic.
CO4:PO3 - Secure database interaction is essential for scalable applications.
CO5:PO3 - JDBC enables building enterprise-level applications.
CO6:PO3 - Data optimization enhances product efficiency.
CO7:PO3 - Integration of systems aligns with entrepreneurial problem-solving.

Mapping of PO4 to all COs

CO1:PO4 - Input validation requires backend security skills.
CO2:PO4 - Regex use demands precision and fluency.
CO3:PO4 - Sorting and arrays are core specialized skills.
CO4:PO4 - PDO usage reflects advanced backend interaction.
CO5:PO4 - JDBC demands knowledge of Java-DB connectivity.
CO6:PO4 - Data optimization is a critical, specialized task.
CO7:PO4 - Combining threads, events, and databases is an advanced competency.

Mapping of PO5 to all COs

CO1:PO5 - Requires identifying and addressing input security gaps.
CO2:PO5 - Regex improves efficiency and precision in string analysis.
CO3:PO5 - Sorting and structuring data builds analytical capabilities.
CO4:PO5 - Applying SQL securely needs clear problem resolution.
CO5:PO5 - JDBC-driven apps demand logical flow and analysis.
CO6:PO5 - Optimization is inherently analytical.
CO7:PO5 - Integrating multiple systems demands problem-solving across layers.

Mapping of PO6 to all COs

CO1:PO6 - Requires clear code and documentation.
CO2:PO6 - Regex patterns must be clearly communicated for team use.
CO3:PO6 - Collaborating on sorting logic involves clear communication.
CO4:PO6 - Secure database interaction needs accurate messaging and handling.
CO5:PO6 - Building DB apps involves code sharing and teamwork.
CO6:PO6 - Optimization often includes peer collaboration.
CO7:PO6 - Multicomponent integration requires collaborative planning and execution.

Mapping of PO7 to all COs

CO1:PO7 - Researching best validation techniques strengthens code.
CO2:PO7 - Understanding regex requires exploring patterns and practices.
CO3:PO7 - Sorting techniques can be improved via algorithm research.
CO4:PO7 - Secure SQL requires understanding best practices and threats.
CO5:PO7 - JDBC integration involves backend architecture exploration.
CO6:PO7 - Optimization methods often come from research and testing.
CO7:PO7 - Multithreading and integration demand extensive exploration and adaptation.

Mapping of PO8 to all COs

CO1:PO8 - Validating input encourages self-learning to handle new types.
CO2:PO8 - Regex updates require continuous learning.
CO3:PO8 - Sorting improvements come from ongoing learning.
CO4:PO8 - Secure SQL operations are updated frequently.
CO5:PO8 - JDBC development involves continuous learning.
CO6:PO8 - Optimization techniques evolve over time.
CO7:PO8 - Real-world integrations require lifelong learning mindset.

Mapping of PO9 to all COs

CO1:PO9 - Sanitization is essential for secure web development.
CO2:PO9 - Regex is a core digital manipulation skill.
CO3:PO9 - Sorting algorithms are fundamental computing tools.
CO4:PO9 - PDO bridges server and database securely.
CO5:PO9 - JDBC is key to database-centric application development.
CO6:PO9 - Java optimization is a tech-intensive skill.
CO7:PO9 - Real-world Java solutions rely heavily on integrated tech.

Mapping of PO10 to all COs

CO1–CO6:PO10 - Secure and efficient systems promote access for diverse users.
CO7:PO10 - Building inclusive systems and solving real-world problems supports diversity.

Mapping of PO11 to all COs

CO1:PO11 - Secure code shows ethical and responsible development.
CO2:PO11 - Prevents misuse via robust input handling.
CO3:PO11 - Efficient code supports resource-conscious systems.
CO4:PO11 - Promotes ethical handling of sensitive data.
CO5:PO11 - Supports system integrity and ethical use of data.
CO6:PO11 - Optimizing reduces resource consumption.
CO7:PO11 - Real-world systems can embed sustainability and ethics.

Mapping of PO12 to all COs

CO1:PO12 - Requires disciplined secure coding practices.
CO2:PO12 - Demands personal responsibility in regex use.
CO3:PO12 - Encourages accountability in logic development.
CO4:PO12 - Handling DBs securely involves accountability.
CO5:PO12 - JDBC applications reflect design responsibility.
CO6:PO12 - Data handling must be accurate and accountable.
CO7:PO12 - Real-world systems require autonomous and responsible implementation.

Mapping of PO13 to all COs

CO1:PO13 - Clean and secure input supports broader app use.
CO2:PO13 - Regex helps create accessible input systems.
CO3:PO13 - Structured data helps build community-facing tools.
CO4:PO13 - Secure database usage protects public info.
CO5:PO13 - JDBC-based apps can serve enterprise and social needs.
CO6:PO13 - Optimized systems benefit end users and organizations.
CO7:PO13 - Integrated real-world systems support scalable community applications.