Kolhapur Institute of Technology's
College of Engineering
(Autonomous), Kolhapur

Department of Computer Science & Engineering

Curriculum and Syllabus for
B. Tech. Computer Science & Engineering
Scheme 2024-25 (As Per NEP, 2020)

R.S. Sutar

Dr. Lingraj A. Hadimani
Head,
Department of Comp. Sci. & Engg.
KIT's College of Engg. (Autonomous)
Kolhapur

Dr. Ranjeeta Sutar
Academic Coordinator, CSE

Dr. L. A. Hadimani
Head, CSE

About the Department

The department was established in the year 1999 with an initial intake of 60. However, in 2018, due to demand from the industry andengineeringaspirants the intakewas increased to 180. The department has been accredited by the National Board of Accreditation (NBA) in 2008, 2018 and recently in 2023. The institution is accredited with A+ grade by National Assessment and Accreditation Council (NAAC) with CGPA 3.33 in 2023.

We have a highly committed and competitive team of faculty fraternity, who experiments and adapts new innovative pedagogy methods for making all the technical concepts clear as well to keep our students abreast with the latest technologies, skills and developments emerging in this field. We have one Doctorate Faculty member and 8 Faculty members are pursuing Ph.D. from various universities. This practice has proved our students very challenging and successful in various domains of competitions at university, state and national level.

Department Vision

To be preferred choice of stakeholders by building core theoretical and practical skills in the students and abreast them to be globally competent with the latest technologies, skills and developments emerging in the field of Computer Science and Engineering.

Department Mission

M1:	To empower students with essential technical comprehension and skills.
M2:	To create awareness of societal and ethical needs in the field of Computer Science and Engineering.
M3:	To build competency among the students with modern tools & technologies and collaborative research.

Programme Educational Objectives (PEO)	
PEO1	Graduates will possess the fundamental knowledge of computer science and engineering needed to address challenges in the real world.
PEO2	Graduates will excel at using their theoretical knowledge in practice, which will enable them to reach professional success and corporate standards.
PEO3	Graduates will acquire humanity skills, professionalism, ethics, entrepreneurial skills and multidisciplinary approach for wider societal and environmental context.
PEO4	Graduates will get a platform for research and development that will assist them in recognizing and addressing the challenges of the IT sector.

Programme Outcomes (PO)	
PO1:	Engineering knowledge: Apply knowledge of mathematics, natural science, computing engineering fundamentals, in depth technical competence in computer science and engineering discipline and an engineering specialization to develop the solution of complex engineering problems.
PO2:	Problem analysis: Identify, formulate, review various computer science research literatures and analyze complex engineering problems reaching substantiated conclusions with consideration for sustainable development.
PO3:	Design/Development of solutions: Design creative solutions for complex computer science and engineering problems and design/develop systems/components/processes to meet identified needs with consideration for the public health and safety, whole-life cost, net zero carbon, culture, society and environment as required.
PO4:	Conduct investigations of complex problems: Conduct investigations of complex computer science and engineering problems using research-based knowledge including design of experiments, modelling, analysis & interpretation of data to provide valid conclusions.
PO5:	Engineering Tool Usage: Create, select and apply appropriate techniques, resources and modern engineering & IT tools, including prediction and modelling recognizing their limitations to solve complex computer science and engineering problems.
PO6:	The Engineer and The World: Analyze and evaluate societal and environmental aspects while solving complex computer science and engineering problems for its impact on sustainability with reference to economy, health, safety, legal framework, culture and environment.
PO7:	Ethics: Apply ethical principles and commit to professional ethics, human values, diversity and inclusion; adhere to national & international laws.
PO8:	Individual and Collaborative Team work: Function effectively as an individual, and as a member or leader in diverse/multi-disciplinary teams.
PO9:	Communication: Communicate effectively and inclusively within the engineering community and society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations considering cultural, language, and learning differences.
PO10:	Project Management and Finance: Apply knowledge and understanding of engineering management principles and economic decision-making and apply these to one's own work, as a member and leader in a team, and to manage projects and in multidisciplinary environments.
PO11:	Life-Long Learning: Recognize the need for and have the preparation and ability for i) independent and life-long learning ii) adaptability to new and emerging technologies and iii) critical thinking in the broadest context of technological change in computer science and engineering discipline.

PROGRAMME SPECIFIC OUTCOMES (PSO)	
PSO1:	Learn and adapt the latest software technologies in the field of computer science and engineering.
PSO2:	Identify and analyze real life problems and provide innovative software solutions.

As per NEP Guidelines Proposed
Scheme of Credit Distribution

Year		FY		SY		TY		B. Tech.			
Sr No		I	II	III	IV	V	VI	VII	VIII	Actual	NEP Guidelines
1	Basic Science course	8	8							16	14-18
2	Engineering Science course	7	6							13	12-16
3	Programme Core Courses	3		15	16	10	10	11		61	44-56
4	Programme Elective course					3	4	3	6	16	20
5	Multi Minor			2	3	3	3	3		14	14
6	Open elective					3	3	2		8	8
7	Vocational and Skill Enhancement course	1	3		1		1			6	8
8	Ability enhancement course		3			1				4	4
9	Entrepreneurship /Economics/ Management courses (Mgt/Economics/Mkt/Finance)			2		2				4	4
10	Indian knowledge system	2								2	2
11	Value Education course			2	2					4	4
12	Research Methodology (Project)							4		4	4
13	Comm. Engg Project/Field Project (PBL/Seminar/Mini Project)					1	1			2	2
14	Project								4	4	4
15	Internship/OJT (PBL/Seminar/Mini Project/Virtual Internship/Physical)			1	1				6	8	12
16	Co-curricular courses		1		1		1		1	4	4
		20-22	20-22	20-22	20-22	20-22	20-22	20-22	20-22	174	
		21	21	22	24	23	23	23	17	174	

Semester VII												
Sr. No	**Category**	**Course Code**	**Course Name**	**L**	**T**	**P**	**Hrs / Week**	**Credits**	**Evaluation Scheme (Component)**			
1	PC	UCSPC0701	Big Data Analytics	3	-	-	3	3	ISE1	10		40
									MSE	30		
									ISE2	10		
									ESE	50	20	
2	PC	UCSPC0702	Cloud Computing	3	-	-	3	3	ISE1	10		40
									MSE	30		
									ISE2	10		
									ESE	50	20	
3	PC	UCSPC0703	Software Testing & Quality Assurance	2	-	-	2	2	ISE1	10		40
									MSE	30		
									ISE2	10		
									ESE	50	20	
4	PEC	UCSPE071*	Program Elective-III (PE-III)	3	-	-	3	3	ISE1	10		40
									MSE	30		
									ISE2	10		
									ESE	50	20	
5	OE	UCSOE072*	Open Elective-III (OE-III)	2	-	-	2	2	ISE1	10		40
									MSE	30		
									ISE2	10		
									ESE	50	20	
7	PC	UCSPC0731	Cloud Computing Laboratory	-	-	2	2	1	ISE	25	10	
									ESE (POE)	25	10	
8	PC	UCSPC0732	Devops Laboratory	-	-	4	4	2	ISE	25	10	
									ESE (OE)	25	10	
9	RM	UCSIL0771	Project-I	-	-	8	8	4	ISE	50	20	
									ESE (OE)	50	20	
10	MM	UCSMM074*	Multi-Disciplinary Minor (MDM-V)	3	-	-	3	3	ESE	100	40	
						Total:	**30**	**23**	**Total Marks: 800** **Total Credit: 23**			

PROGRAM ELECTIVE-III (PE-III)		
Sr. No.	**Course Code**	**Course Name**
1	UCSPE0711	Deep Learning
2	UCSPE0712	IoT Security and Privacy
3	UCSPE0713	Natural Language Processing
4	UCSPE0714	Digital Image Processing

Title of the Course: Big Data Analytics **Course Code:** UCSPC0701	**L**	**T**	**P**	**Credits**
	3	-	-	3

Course Pre-Requisite: Basic knowledge of Programming, Database Systems, and Statistics.

Course Description:

This course introduces foundational and advanced concepts of Big Data Analytics. Students will learn distributed computing frameworks, data processing techniques, storage architectures, and real-world big data applications.

Course Learning Objectives:

1. To understand Big Data concepts, challenges, and ecosystem
2. To learn distributed frameworks such as Hadoop and Spark
3. To apply big data processing techniques to real-world datasets
4. To design scalable big data solutions using analytics tools

Course Outcomes:

CO	**After Completion of the course, the student should be able to**	**Bloom's Cognitive**	
		Level	**Descriptor**
CO1	Describe the core concepts and components of the Big Data ecosystem	2	Understand
CO2	Explain Hive and Pig Architecture	2	Understand
CO3	Apply data processing techniques using Hadoop/Spark.	3	Apply
CO4	Analyze big data analytics results to derive actionable insights for real-world decision-making	4	Analyze

CO-PO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO01	PSO02
CO1	3	2			1						3	1	
CO2	3	1			1						3	2	1
CO3	3	2	2	2	3						3	3	1
CO4	3	3	2	2	2						3	3	1

Assessment:

Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one End Semester Examination (ESE) having 20%, 30% and 50% weights respectively.

Assessment	Marks
In Semester Evaluation 1 (ISE1)	10
Mid Semester Examination (MSE)	30
In Semester Evaluation 2 (ISE2)	10
End Semester Examination (ESE)	50

ISE-1 and **ISE-2** are based on assignment/declared test/quiz/seminar/Group Discussions etc.

MSE: Assessment is based on 50% of course content (Normally first three modules)

ESE: Assessment is based on 100% course content with 60-70% weightage for course content (normally last three modules) covered after MSE.

Course Contents:

Unit 1: Introduction to Big Data	**8 Hours**
Definition of Big Data, Characteristics (Volume, Velocity, Variety, Veracity, Value), Big Data challenges, Big Data lifecycle, Big Data ecosystem, introduction to distributed systems.	
Unit 2: Hadoop Fundamentals	**8 Hours**
HDFS architecture, NameNode, DataNode, YARN, Hadoop MapReduce, Input/Output formats, Hadoop workflows, limitations of Hadoop.	
Unit 3: Apache Spark for Big Data Processing	**06 Hours**
RDDs, DataFrames, Spark SQL, Transformations & Actions, Lazy Evaluation, DAG, Spark MLib introduction, Performance considerations.	
Unit 4: Introduction to Hive and Pig	**8 Hours**
Introduction to Hive: What is Hive, Hive Architecture, Hive data types, Hive file formats, Hive	

Query Language (HQL), RC File implementation, User Defined Function (UDF). **Introduction to Pig:** What is Pig, Anatomy of Pig, Pig on Hadoop, Pig Philosophy, Use case for Pig, Pig Latin Overview, Data types in Pig, Running Pig, Execution Modes of Pig, HDFS Commands, Relational Operators, Eval Function, Complex Data Types, Piggy Bank, User Defined Function, Pig Vs Hive.	
Unit 5: Big Data Analytics Techniques	**9 Hours**
Data preprocessing at scale, Feature engineering, Stream processing (Kafka, Spark Streaming), Real-time analytics, Machine learning for big data.	
Unit 6: Big Data Applications	**6 Hours**
Case studies in finance, healthcare, retail, transport, IoT analytics, fraud detection, recommendation systems, sentiment analysis.	
Textbooks:	
1. Anand Rajaraman, Jure Leskovec, and Jeffrey Ullman, 'Mining of Massive Datasets', Cambridge University Press. **Link:** https://mrce.in/ebooks/Mining%20of%20Massive%20Datasets%203rd%20Ed.pdf **2.** Tom White, 'Hadoop: The Definitive Guide', O'Reilly. **Link:**https://piazzaresources.s3.amazonaws.com/ist3pwd6k8p5t/iu5gqbsh8re6mj/OReilly.Hadoop.The.Definitive.Guide.4th.Edition.2015.pdf	
Reference Books:	
1. Martin Kleppmann, 'Designing Data-Intensive Applications', O'Reilly. 2. Holden Karau and Matei Zaharia, 'Learning Spark', O'Reilly. 3. Nathan Marz, 'Big Data: Principles and Best Practices of Scalable Real-Time Data Systems'.	

Title of the Course: Cloud Computing **Course Code:** UCSPC0702	L	T	P	Credits
	3	-	-	3

Course Pre-Requisite: Basics of Computer Network, Operating Systems

Course Description:

The Cloud Computing course will focus on the evolution of cloud environment, its architecture, types, prominent cloud platform examples, virtualization techniques and migration, docker-container & Kubernetes, security and management.

Course Learning Objectives:

1. Introduce the fundamentals of cloud computing
2. Provide conceptual understanding of cloud computing architecture
3. Develop an understanding of virtualization concepts, techniques and migration
4. Develop practical understanding of containerization technologies
5. Create awareness of cloud security and management issues

Course Outcomes:

CO	After Completion of the course, the student should be able to	Bloom's Cognitive	
		Level	Descriptor
CO 1	Explain the cloud computing architecture, types and models	1	Remember
CO 2	Classify the virtualization techniques	2	Understand
CO 3	Make use of containers for software deployment	3	Apply
CO 4	Analyse security threats and security measure for cloud computing	4	Analyze

CO-PO Mapping:

CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO1 0	PO1 1	PSO0 1	PSO0 2
CO 1	2		2		2	2					1	2	1
CO 2	2		2		3						1	3	2
CO 3	2		3		3						1	3	2
CO 4	2		2		2	2		2				2	2

Assessment:

Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one End Semester Examination (ESE) having 20%, 30% and 50% weights respectively.

Assessment	Marks
ISE-1	10
MSE	30
ISE-2	10
ESE	50

ISE-1 and **ISE-2** are based on assignment/declared test/quiz/seminar/Group Discussions etc.

MSE: Assessment is based on 50% of course content (Normally first three modules)

ESE: Assessment is based on 100% course content with 30% weightage on first 3 Units & 70% weightage for course content covered after MSE.

Course Contents:

Unit 1: Introduction	**7 Hours**

Definition, Historical Developments, Computing Platforms and Technologies. Building cloud computing environments, Principles of Parallel and Distributed Computing: Parallel versus Distributed Computing, Elements of Parallel Computing, Elements of Distributed Computing, and Technologies for Distributed Computing.

Unit 2: Virtualization	**8 Hours**

Characteristics, Virtualization Techniques - emulated, para virtualization, full virtualization, (KVM, VBox, Citrix), Virtualization and Cloud Computing, Pros and Cons of Virtualization	
Unit 3: Cloud Computing Architecture	**8 Hours**
Cloud Reference Model, Types of Clouds – Public, Private, Hybrid and Community cloud, Types of Services – IaaS, PaaS, SaaS, Economics of Clouds, Open Challenges, case studey of any one Public Clouds: Amazon Web Services (AWS), Google Cloud Platform (GCP), Microsoft Azure	
Unit 4: Migration into cloud and Virtual machine Provisioning	**7 Hours**
Broad Approaches to Migrating into the Cloud, The Seven-Step Model of Migration into a Cloud, Virtual Machines Provisioning and Manageability, Virtual Machine Migration Services, VM Provisioning and Migration in Action, Provisioning in the Cloud Context. KVM -Migration	
Unit 5: Advanced Concepts – Docker, Container and Kubernetes:	**8 Hours**
Introduction to CaaS, why containers? Difference between Virtualization and Containers. Introduction to Containers, Docker and its architecture, Understanding Docker Container, Networking. Creating docker containers with yaml, Kuberentes – Introduction, Architecture	
Unit 6: Cloud Security & Management:	**7 Hours**
Fundamental cloud security – Basic terms and concepts, Threat agents, cloud security threats, case study example. Cloud Security Mechanisms – PKI, IAM and SSO with case studies.	
Textbooks:	
1. Mastering Cloud Computing - Buyya R, Vecchiola C, Selvi S T, McGraw Hill Education (India), 2013 (Unit 1,2,3) 2. Cloud Computing - Principles and Paradigms - Buyya R, Broberg J, Goscinski A, Wiley,2011 (Unit 4) 3. Cloud Computing Concepts, Technology & Architecture - Thomas Erl, Zaigham Mahmood, and Ricardo Puttini, (Unit 6) 4. A to z on Docker: A complete Hands-On Guide to Docker Container – Swapnil Jain (Unit 5. Docker Cookbook - Sébastien Goasguen, O'reilly Nov. 2015 First Edition (Unit 5)	
Reference Books:	

References:

1. Cloud Computing Bible - Barrie Sosinsky ,Wiley Publishing Inc. 2011(Unit,6)

2. Cloud Native DevOps with Kubernetes – John Arundel and Justin Domingus (Unit 5)

Title of the Course: Software Testing & Quality Assurance	L	T	P	Credits
Course Code: UCSPC0703	2	-	-	2

Course Pre-Requisite: Software Project Management, Agile Project Management

Course Description:

This course provides comprehensive coverage of software testing methodologies, quality assurance principles, and modern testing practices essential for developing reliable software systems. Students will master systematic test design techniques, defect management, quality metrics, and test automation concepts while gaining practical knowledge of testing in Agile and DevOps environments. The curriculum balances theoretical foundations with industry-relevant practices, preparing students for professional roles in software quality assurance and testing.

Course Learning Objectives:

1. Apply fundamental testing principles and quality concepts to evaluate software systems in different SDLC phases.
2. Design effective test cases using black-box and white-box testing techniques for given software specifications.
3. Select appropriate test levels and testing types to develop comprehensive test strategies for software projects.
4. Apply defect management processes, calculate quality metrics, and recommend suitable test automation strategies.

Course Outcomes:

CO	After Completion of the course, the student should be able to	Bloom's Cognitive	
		Level	Descriptor
CO1	Illustrate software quality concepts, testing principles, and testing's role in SDLC.	2	Understand
CO2	Develop design strategies for functional and non-functional testing.	3	Apply
CO3	Apply systematic test design techniques to create comprehensive test cases for software validation.	3	Apply
CO4	Analyze test automation frameworks and automation strategies	4	Analyze

	with respect to the test automation pyramid		
CO5	Dissect defect lifecycle management, calculate quality metrics, and interpret them for process improvement.	4	Analyze

CO-PO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO01	PSO02
CO1	2	1	1	1	2						1	2	2
CO2	2	2	3	2	2						1	2	2
CO3	2	1	1	1	2						1	2	2
CO4	2	1	1	1	2						1	2	2
CO5	2	1	1	1	1						1	2	2

Assessment:

Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one End Semester Examination (ESE) having 20%, 30% and 50% weights respectively.

Assessment	Marks
ISE-1	10
MSE	30
ISE-2	10
ESE	50

ISE-1 and **ISE-2** are based on assignment/declared test/quiz/seminar/Group Discussions etc.

MSE: Assessment is based on 50% of course content (Normally first three modules)

ESE: Assessment is based on 100% course content with 70% weightage for course content (normally last three modules) covered after MSE.

Course Contents:

Unit 1: Foundations of Software Testing & Quality	**05 Hours**

Software Quality Concepts: Definition of software quality, Quality characteristics (ISO 25010) - Cost of quality, Impact of poor-quality **Testing Fundamentals:** Definition and objectives of software testing, Testing vs Debugging, Error, Defect, and Failure concepts **Testing in SDLC:** Software Development Life Cycle overview, Four test levels: Unit Testing, Integration Testing, System Testing, Acceptance Testing, V-Model, Basic test process: Planning, Design, Execution, Reporting,

Closure	
Unit 2: Test Design Techniques	**05 Hours**
Black-Box Testing Techniques: Equivalence Partitioning, Boundary Value Analysis, Decision Table Testing **White-Box Testing Techniques:** Statement and Branch Coverage, Cyclomatic Complexity	
Unit 3: Test Levels and Types	**05 Hours**
Test Levels: Unit Testing, Integration Testing - Purpose and need, Integration strategies (Big-Bang, Top-Down (stubs), Bottom-Up (drivers)), System and Acceptance Testing **Types of Testing:** Functional vs Non-Functional Testing, Performance Testing - Load Testing, Stress Testing and Endurance Testing, Security Testing - Injection attacks, Broken authentication, Sensitive data exposure	
Unit 4: Test Automation Concepts	**05 Hours**
Introduction to Test Automation: Automation Fundamentals: Manual vs Automated testing, Benefits of automation, **Automation ROI:** Return on Investment concept, Cost of automation. **Automation Frameworks:** Framework definition, Types of frameworks: Linear (Record-Playback), Data-Driven, Keyword-Driven **Test Automation Pyramid:** Pyramid concept, Three layers: Unit Tests, Integration/API Tests, UI Tests **Automation Best Practices:** Test independence, Test naming conventions, Test data management	
Unit 5: Defect Management and Metrics	**05 Hours**
Defect Management: Defect Basics - Defect definition, Defect vs Failure, Defect lifecycle, Defect type classifications **Defect Reporting:** Defect report components **Defect Metrics** Purpose, Defect Density, Defect Detection Efficiency, Defect Leakage **Test Metrics:** Coverage Metrics, Test Effectiveness Metrics.	
Unit 6: Quality Assurance and Modern Practices	**05 Hours**
Quality Assurance Fundamentals: Quality Assurance, Quality Control **Quality Standards:** Importance of standards, ISO 25010 overview, IEEE 829 (Test documentation**) Agile Development and Testing:** Testing in Agile: Continuous testing, Whole team approach, Collaboration Agile Testing Quadrants, **DevOps and Continuous Testing:** Continuous Integration (CI), Continuous Delivery (CD), Continuous Testing concept, Testing in CI/CD pipelines	
Textbooks:	

1. Graham, D., Van Veenendaal, E., Evans, I., & Black, R. (2023). Foundations of software testing: ISTQB certification (4th ed.). Cengage Learning.
2. Desikan, S., & Ramesh, G. (2017). Software testing: Principles and practices (2nd ed.). Pearson Education.

Reference Books:

1. Patton, R. (2005). Software testing (2nd ed.). Sams Publishing.
2. Myers, G. J., Sandler, C., & Badgett, T. (2011). The art of software testing (3rd ed.). Wiley.
3. International Software Testing Qualifications Board. (2024). ISTQB certified tester foundation level syllabus. https://www.istqb.org

Title of the Course: Deep Learning	**L**	**T**	**P**	**Credits**
Course Code: UCSPE0711	3	-	-	3

Course Pre-Requisite: Machine Learning, Linear Algebra, Probability and Information Theory, Numerical Computation

Course Description:

The purpose of this course is to provide advanced knowledge of Machine learning with the students. It aims to enable the students to understand the design of various Deep Learning models and applications.

Course Learning Objectives:

1. To introduce the idea of artificial neural networks and their architecture
2. Illustrate techniques used for training artificial neural networks
3. To enable design of an artificial neural network for classification
4. Design and deployment of deep learning models for machine learning problems

Course Outcomes:

CO	After Completion of the course, the student should be able to	Bloom's Cognitive	
		Level	Descriptor
CO1	Discuss the mathematics and fundamental concepts behind functioning of artificial neural networks	2	Understand
CO2	Describe and learn different Neural Network Architectures.	2	Understand
CO3	Apply optimization strategies on neural networks for large scale applications and loss functions.	3	Apply
CO4	Make use of different models in Neural Networks(CNN,ANN,RNN) and their applications	3	Apply
CO5	Examine the case studies and applications of deep learning techniques	4	Examine

CO-PO Mapping:

CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO1 0	PO1 1	PSO0 1	PSO0 2
CO1	3			2							2	1	1
CO2	3	1		2	2						2	2	2
CO3	3	2	2	2	3						2	3	2
CO4	3	3	2	3	3						2	2	3
CO5	1	3	2	3	2	1					2	3	3

Assessment:

Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one End Semester Examination (ESE) having 20%, 30% and 50% weights respectively.

Assessment	**Marks**
In Semester Evaluation 1 (ISE1)	10
Mid Semester Examination (MSE)	30
In Semester Evaluation 2 (ISE2)	10
End Semester Examination (ESE)	50

ISE-1 and **ISE-2** are based on assignment/declared test/quiz/seminar/Group Discussions etc.

MSE: Assessment is based on 50% of course content (Normally first three modules)

ESE: Assessment is based on 100% course content with 60-70% weightage for course content (normally last three modules) covered after MSE.

Course Contents:

Unit 1: Introduction to Artificial Neural Network	**08 Hours**

Explore the history, evolution, and fundamental concepts of deep learning, Artificial Neuron, Input Layers, Hidden Layers, Output Layers, Weights assigned to neurons, Bias and variance, Learning Rate, Hyperparameter, Gradient Descent, underfitting, overfitting, Training dataset & Training error, Test dataset & Test error, validation dataset, Cross Validation.

Unit 2: Artificial Neural Network architecture:	**08 Hours**

Neural Network Architecture, Perceptron, Network Architecture with One Hidden Layer, Network Architecture with One Hidden Layer and Multiple Outputs, Activation functions, Back propagation(Loss Functions-MSE & MAE, Chain rule of differentiation, Vanishing Gradient problem)

Unit 3: Regularization for Deep Learning:	06 Hours
Generalization Error, Regularization, Curse of Dimensionality, Optimization for Training Deep Models: Gradient Descent, Stochastic Gradient Descent (SGD), Mini Batch SGD, SGD with Momentum, Adagrad, RMSPROP, Adam Optimizers etc.	
Unit 4: Convolution Neural Network:	**09 Hours**
Motivation and Applications, Architecture of Convolutions, Filters, pooling layers, Padding CNN Architectures (Alex Net, VGG, NiN, GoogLeNet, ResNet, DensNet), Application in Image segmentation, Automated Object Detection models.	
Unit 5: Deep Sequence Models:	**08 Hours**
Sequence Modeling Problems, Motivation and Applications, Traditional Models: Recurrent Neural Networks, Back-propagation through time; Modern Recurrent Neural Networks: Gated Recurrent Units, Long Short Term Memory (LSTM), Bidirectional LSTM.	
Unit 6: Applications of Deep learning:	**06 Hours**
Deep learning applications are making an impact across many different industries. Case studies like Fraud detection, financial services, Natural language processing, Facial recognition, Self-driving vehicles, Recommender systems, Health care	

Textbooks:

1. Ian Goodfellow, Yoshua Bengio and Aaron Courville, Deep Learning, MIT Press, 2016. https://www.deeplearningbook.org/lecture_slides.html
2. Simon Haykin, Neural networks and Learning Machines, Third Edition, Pearson, 2016
3. Nikhil Buduma, "Fundamentals of Deep Learning: Designing Next-Generation Machine Intelligence Algorithm", O'Reilly, 2017

Reference Books:

1. Machine Learning: An Algorithmic Perspective, Second Edition , Author Stephen Marsland Publisher Chapman and Hall/CRC
2. Nikhil Ketkar, "Deep Learning with Python: A Hands-on Introduction", Apress, 2017..
3. Aurélie Geron, "Hands-On Machine Learning with Scikit- Learn and TensorFlow", O'Reilly, 2017.

Title of the Course: IoT Security and Privacy	**L**	**T**	**P**	**Credits**
Course Code: UCSPE0712	3	-	-	3

Course Pre-Requisite: IoT fundamentals, Information Security, Cyber security

Course Description:

This course on IoT Security and Privacy is designed to provide comprehensive knowledge and practical skills for securing interconnected devices and cyber-physical systems. It covers various aspects, from theoretical foundations to real-world application of security and privacy measures of IoT

Course Learning Objectives:

1. To know the state-of-the-art methodologies in Cyber Physical systems.
2. To impart knowledge on Model threats and countermeasures.
3. To explore the Privacy Preservation and Trust Models in Internet of Things (IoT)
4. To apply the concept of Internet of Things Security in the real-world scenarios
5. To explore cloud security for the IoT.

Course Outcomes:

CO	**After Completion of the course, the student should be able to**	**Bloom's Cognitive**	
		Level	**Descriptor**
CO1	Identify common IoT threats, vulnerabilities, and attacks.	1	Remember
CO2	Explain privacy preservation and trust models in IoT systems.	2	Understand
CO3	Apply security measures and cloud security practices to IoT systems.	3	Apply
CO4	Analyze IoT protocols and security frameworks for potential attacks.	4	Analyze
CO5	Evaluate solutions to improve trust, privacy, and resilience in IoT networks.	5	Evaluate

CO-PO Mapping:

CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PSO 01	PSO 02
CO1	1	1	1			2					1		
CO2	1	1	1			1					1		
CO3	1	1	1			1					1	2	3
CO4	1	2	1								2	2	
CO5	1	1	1								1		

Assessment:

Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one End Semester Examination (ESE) having 20%, 30% and 50% weights respectively.

Assessment	**Marks**
In Semester Evaluation 1 (ISE1)	10
Mid Semester Examination (MSE)	30
In Semester Evaluation 2 (ISE2)	10
End Semester Examination (ESE)	50

ISE-1 and **ISE-2** are based on assignment/declared test/quiz/seminar/Group Discussions etc.

MSE: Assessment is based on 50% of course content (Normally first three modules)

ESE: Assessment is based on 100% course content with 60-70% weightage for course content (normally last three modules) covered after MSE.

Course Contents:

Unit 1: IOT Threats And Attacks	**08 Hours**
IoT and cyber-physical systems, IoT security (vulnerabilities, attacks, and countermeasures, security engineering for IoT development, IoT security lifecycle. Network Robustness of Internet of Things-Sybil Attack Detection in Vehicular Networks- Malware Propagation and Control in Internet of Things- Solution-Based Analysis of Attack Vectors on Smart Home Systems.	
Unit 2: Privacy Preservation For IOT	**06 Hours**
Privacy Preservation Data Dissemination- Privacy Preservation Data Dissemination- Social Features for Location Privacy Enhancement in Internet of Vehicles- Lightweight and Robust Schemes for Privacy Protection in Key Personal IoT Applications: Mobile WBSN and Participatory Sensing	
Unit 3: Trust Models For IOT	**09 Hours**
Authentication in IoT- Computational Security for the IoT- Privacy-Preserving Time Series Data Aggregation- Secure Path Generation Scheme for Real-Time Green Internet of Things- Security Protocols for IoT Access Networks- Framework for Privacy and Trust in IoT- Policy-	

Based Approach for Informed Consent in Internet of Things.	
Unit 4: Internet Of Things Security	**09 Hours**
Security and Impact of the Internet of Things (IoT) on Mobile Networks- Networking Function Security-IoT Networking Protocols, Secure IoT Lower Layers, Secure IoT Higher Layers, Secure Communication Links in IoTs, Back-end Security -Secure Resource Management, Secure IoT Databases, Security Products-Existing Test bed on Security and Privacy of IoTs, Commercialized Products.	
Unit 5: Social Awareness	**08 Hours**
SocIoTal Security Framework,Device-Centric Enablers for Privacy and Trust,Problems Defining Informed Consent in the Internet of Things,Security Threats against IoT Embedded Devices and Systems,IoT Security Impacts against Mobile Networks, Scalability of Large Deployments of Cellular IoT Systems	
Unit 6: Cloud Security for The IOT And IOT Attacks	**06 Hours**
Cloud services and the IoT – Exploring cloud service provider IoT offerings – Cloud IoT security controls – IoT cloud security architecture – IoT incident response – Threats both to safety and security. MIRAI Botnet Attack - Stuxnet Attack –Tesla Crypto jacking Attack -The TREND net Web cam Attack -The Jeep SUV Attack -The Owlet Wi-Fi Baby Heart Monitor Vulnerabilities -St. Jude_ Hackable Cardiac Devices.	
Textbooks:	
1. Hu, Fei. Security and privacy in Internet of things (IoTs): Models, Algorithms, and Implementations, 1st edition,CRC Press, 2016. 2. Russell, Brian, and Drew Van Duren. "Practical Internet of Things Security", 1st edition, Packt Publishing Ltd, 2016.	
Reference Books:	
1. Zaigham Mahmood, “Security, Privacy and Trust in the IoT Environment”, Springer, 2019. (ISBN: 9783030180744) 2. DaCosta, Francis, and Byron Henderson.," Rethinking the Internet of Things: a scalable approach to connecting everything", 1st edition, Springer Nature, 2013. 3. Whitehouse O. "Security of things: An implementers’ guide to cyber-security for internet of things devices and beyond" 1st edition, NCC Group, 2014	

Title of the Course: Natural Language Processing Course Code: UCSPE0713	L	T	P	Credits
	3	-	-	3

Course Pre-Requisite: Theory of Computation, Machine Learning

Course Description:

This course is about how to build systems that analyze unstructured natural language texts and extract useful information from them. The course will cover the technical methodology to apply these frameworks in an informed way, and to make current research accessible.

Course Learning Objectives:

1. To introduce the fundamental concepts and techniques of natural language processing (NLP).
2. To acquire the knowledge of Morphological, Syntactic & Semantic NLP tasks.
3. To use appropriate tools & techniques to solve real world applications in the NLP domain.
4. To describe Machine translation & information retrieval.

Course Outcomes:

CO	After Completion of the course, the student should be able to	Bloom's Cognitive	
		Level	Descriptor
CO1	Explain the basic concepts and basic algorithms of Natural language processing.	2	Understand
CO2	Summarize data from text automatically by utilizing natural language processing (NLP) methods and techniques.	2	Understand
CO3	Apply machine learning algorithms to natural language processing.	3	Apply
CO4	Examine current methods for statistical approaches to machine translation.	4	Analyze
CO5	Analyze the syntax, semantics, and pragmatics of statements written using a natural language.	4	Analyze

CO-PO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO01	PSO02
CO1	3	2	2	2	2							2	3
CO2	3	2	2	2	3							3	2
CO3	3	3	2	2	3							3	3
CO4	3	2	2	2	3							3	2
CO5	2	2	2	2	2							2	2

Assessment:

Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one End Semester Examination (ESE) having 20%, 30% and 50% weights respectively.

Assessment	Marks
ISE-1	10
MSE	30
ISE-2	10
ESE	50

ISE-1 and **ISE-2** are based on assignment/declared test/quiz/seminar/Group Discussions etc.

MSE: Assessment is based on 50% of course content (Normally first three modules)

ESE: Assessment is based on 100% course content with 60-70% weightage for course content (normally last three modules) covered after MSE.

Course Contents:

Unit 1: Introduction to NLP Various stages of NLP –The Ambiguity of Language: Why NLP Is Difficult? Parts of Speech: Nouns and Pronouns, Words: Determiners and adjectives, verbs, Phrase Structure. Statistics Essential Information Theory : Entropy, perplexity, The relation to language, Cross entropy	**5 Hours**
Unit 2: Text Pre-Processing and Morphology Character Encoding, Word Segmentation, Sentence Segmentation, Introduction to Corpora, Corpora Analysis. Inflectional and Derivation Morphology, Morphological analysis and generation using Finite State Automata and Finite State transducer	**8 Hours**

Unit 3: Language Modeling Words: Collocations- Frequency Mean and Variance – Hypothesis testing: The t test, Hypothesis testing of differences, Pearson's chi- square test, Likelihood ratios. Statistical Inference: n -gram Models over Sparse Data: Bins: Forming Equivalence Classes - N gram model - Statistical Estimators- Combining Estimators	**8 Hours**
Unit 4: Word Sense Disambiguation and POS tagging Methodological Preliminaries, Supervised Disambiguation: Bayesian classification, An information theoretic approach, Dictionary- Based Disambiguation: Disambiguation based on sense, Thesaurus based disambiguation, Disambiguation based on translations in a second-language corpus, POS tagging and its application.	**8 Hours**
Unit 5: Syntax and Semantics Shallow Parsing and Chunking, Shallow Parsing with Conditional Random Fields (CRF), Lexical Semantics, WordNet, Thematic Roles, Semantic Role Labelling with CRFs. Statistical Alignment and Machine Translation, Text alignment, Word alignment, Information extraction, Text mining, Information Retrieval, NL interfaces	**7 Hours**
Unit 6: NLP Tools & Techniques Treebank, Universal Dependency Treebank, Natural Language Toolkit, spacy, Textblob, Gensim etc. Sentiment Analysis, Question answering, Recent Trends in NLP	**6 Hours**
Textbooks:	
1. Christopher D. Manning and Hinrich Schutze, " Foundations of Natural Language Processing" , 6 th Edition, The MIT Press Cambridge, Massachusetts London, England, 2003 2. Daniel Jurafsky and James H. Martin "Speech and Language Processing", 3rd edition,	

Prentice Hall, 2009.
Reference Books:
1. Nitin Indurkhya, Fred J. Damerau “Handbook of Natural Language Processing”, Second Edition, CRC Press, 2010. 2. James Allen “Natural Language Understanding”, Pearson Publication 8th Edition. 2012. 3. Chris Manning and HinrichSchütze, “Foundations of Statistical Natural Language Processing”, 2nd edition, MITPress Cambridge, MA, 2003. 4. Hobson lane, Cole Howard, Hannes Hapke, “Natural language processing in action” MANNING Publications, 2019

Title of the Course: Digital Image Processing	**L**	**T**	**P**	**Credits**
Course Code: UCSPE0714	3	-	-	3

Course Pre-Requisite: Linear Algebra, Calculus

Course Description: This course aims to introduce fundamental concepts of Digital Image processing. It will start with representation of images, data structures and eventually go towards standard image processing tasks such as various image enhancement, restoration, image compression, etc. It will also include some advanced topics such as Image segmentation techniques

Course Learning Objectives:

1. Define the image processing fundamentals
2. Explain Image Enhancement techniques in special and frequency domains.
3. Illustrate Image degradation, restoration and Compression Techniques.
4. Interpret Image segmentation techniques.

Course Outcomes:

CO	**After Completion of the course, the student should be able to**	**Bloom's Cognitive**	
		Level	**Descriptor**
CO1	Explain the image processing fundamentals	2	Understand
CO2	Summarize different image pre-processing and filtering techniques to enhance the image.	2	Understand
CO3	Illustrate Image compression and Image Restoration Techniques.	2	Understand
CO4	Choose appropriate Image Segmentation Techniques for Line, Edge, Boundary detection	3	Apply
CO5	Apply image processing techniques for solving problems in real world.	3	Apply

CO-PO Mapping:

CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PSO 01	PSO 02
CO1	1	1		1							3		2
CO2	1	3	2	3							2	1	3
CO3	2	3	2	3	3						3	3	3
CO4	2	2	2	2	2						2	2	1
CO5	1	2	2	2							2		3

Assessment:

Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one End Semester Examination (ESE) having 20%, 30% and 50% weights respectively.

Assessment	Marks
ISE-1	10
MSE	30
ISE-2	10
ESE	50

ISE-1 and **ISE-2** are based on assignment/declared test/quiz/seminar/Group Discussions etc.

MSE: Assessment is based on 50% of course content (Normally first three modules)

ESE: Assessment is based on 100% course content with 60-70% weightage for course content (normally last three modules) covered after MSE.

Course Contents:

Unit 1: Introduction	**7 Hours**

What is Image processing? Examples. Fundamental Steps in Digital Image Processing, Components of an Image Processing System, Sampling and Quantization, Representing Digital Images (Data structure), Some Basic Relationships Between Pixels- Neighbors and Connectivity of pixels in image

Unit 2: Image Enhancement in the Spatial Domain	**8 Hours**

Some basic Gray level Transformations, Histogram Processing, Enhancement using Arithmetic/Logic Operations, Spatial Filtering, Smoothing Spatial Filters, Sharpening Spatial Filters, Combining Spatial Enhancement methods

Unit 3: Image Enhancement in the Frequency Domain	**7 Hours**

Fourier Transform and the Frequency Domain, Smoothing Frequency-Domain Filters, Sharpening Frequency Domain Filters, Homomorphism Filtering, Implementation.	
Unit 4: Image Degradation and Image Restoration	**7 Hours**
Image Degradation/Restoration Process, Linear, Position-Invariant Degradations, Inverse Filtering, Minimum, Mean Square Error (Wiener) Filtering, Constrained Least Squares Filtering.	
Unit 5: Image Compression	**8 Hours**
Image Compression: Image Compression Models, Error-Free Compression, Lossy Compression, Image Compression Standards.	
Unit 6: Image Segmentation	**8 Hours**
Introduction, Detection of isolated points, line detection, Edge detection, Segmentation using Threshold. Detection of Discontinuities, Edge Linking and Boundary Detection, Region growing, split and merge technique	
Textbooks:	
1. Rafael Gonzalez, Richard E. Woods; "Digital Image Processing 'Addison Wesley Pubs (Second Edition), 2007. 2. Milan Sonka, Vaclav Hlavac, Roger Boyle Image Processing. Analysis, and Machine Vision (Second Edition, 2003).	
Reference Books:	
1. Fundamentals of Digital Image Processing- Anil K. Jain, 2nd Edition, Prentice Hall of India. 2. S. Sridhar, Digital Image Processing, Oxford University Press, 2nd Ed, 2016.).	

Course Code:	UCSPC0731	L	T	P	Credit
Course Name:	Cloud Computing Laboratory			2	1

Course Prerequisites:

Data Structures, Programming Language

Course Description:

In this course, students will learn necessary tools and techniques required for deploying projects on cloud and handling various services on the cloud.

Course Learning Objective

Enable students to gain practical knowledge of cloud computing concepts and services. By the end of the course, students will be able to create and manage cloud resources, deploy applications, and apply basic security practices. The course also develops skills in monitoring, troubleshooting, and documenting cloud-based solutions.

Course Outcomes:

COs	After completion of the course, students shall be able to -	Bloom's Cognitive Level	Bloom's Cognitive Descriptor
CO1	Make use of the public cloud environment.	3	Apply
CO2	Build virtual machines using virtualization techniques	5	Build
CO3	Create custom images and containers for software deployment	5	Create

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	2	2	3	2	2	1		2	2	3	3
CO2	3	3	3	1	3			1			2	3	3
CO3	3	3	3	1	3			1			2	3	3

Assessment Scheme:

SN	Assessment	Marks	Remark
1	In Semester Evaluation 1 (ISE)	25	Assignment, Test, Quiz, Seminar, Presentation.
2	End Semester Examination (ESE)	25	Practical Performance S Viva

Course Contents:

Experiment No. 1	2 Hours
Use Google Collab book for writing program	

Experiment No. 2	**2 Hours**
Use google APIs to access google cloud services	
Experiment No. 3	**2 Hours**
Create Virtual Machine using emulator - emue and virtual library	
Experiment No. 4	**2 Hours**
Create Virtual Machines using KVM library - paravirtualized machine	
Experiment No. 5	**2 Hours**
Create bare-metal virtual machine	
Experiment No. 6	**2 Hours**
Create container using lxc	
Experiment No. 7	**2 Hours**
Create a container using docker - docker desktop , docker CLI	
Experiment No. 8	**2 Hours**
Networking of Docker Containers	
Experiment No. 9	**2 Hours**
Building Docker Image	
Experiment No. 10	**4 Hours**
Dockerfile creation and container deployment	
References https://docs.aws.amazon.com/ https://docs.cloud.google.com/docs https://docs.docker.com/reference/ https://kubernetes.io/docs/home/	

Title of the Course: Devops Laboratory	**L**	**T**	**P**	**Credits**
Course Code: UCSPC0732	-	-	2	1

Course Pre-Requisite: Cloud Basics, Linux fundamentals, Scripting skills.

Course Description:

This intensive, hands-on lab course provides practical experience in implementing a full Continuous Integration and Continuous Delivery (CI/CD) pipeline using industry-standard DevOps tools. Students will move beyond theory to design, build, deploy, and monitor a containerized application within a simulated enterprise environment.

The lab is designed to mirror real-world scenarios, requiring students to integrate various tools to achieve seamless automation from code commit to production deployment.

Course Learning Objectives:

1. Implement CI/CD Workflows: Design and configure automated pipelines using Jenkins (or similar CI server).
2. Manage Infrastructure as Code (IaC): Automate environment provisioning using tools like Terraform or Ansible.
3. Containerize Applications: Build, manage, and push Docker images to a container registry.
4. Orchestrate with Kubernetes: Deploy, manage, and scale containerized applications using Kubernetes (K8s).
5. Apply DevOps Principles: Understand how to collaborate effectively across development and operations teams using GitOps principles.

Course Outcomes:

COs	**After Completion of the course, the student should be able to**	**Bloom's Cognitive**	
		Level	**Descriptor**
CO1	Develop Continuous Integration (CI).	3	Apply
CO2	Create Continuous Delivery/Deployment (CD).	3	Apply
CO3	Apply Version Control Effectively.	3	Apply
CO4	Build Infrastructure as Code (IaC) using automated cloud	3	Apply

	resources.		
CO5	Execute the deployment of a specific application image to a local Docker engine	3	Apply

CO-PO Mapping:

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO01**	**PSO02**
CO1	3		3		3				2	3		3	3
CO2	2		2		3					2			
CO3	3		3		3				1	3	1	3	3
CO4	3		3		3					3			
CO5	2	3	2	3	3	2	3			2	2		

Assessment:

Two components - In Semester Evaluation (ISE) and End Semester Examination (ESE).

Assessment	**Marks**
ISE	25
ESE	25

ISE: Assessment is based on experiments/assignment etc.

ESE: Assessment is based on Oral Examination.

Course Contents:

Experiment No. 1- Sample Application development	**2 Hours**

1. Aim and Objectives: Write code for a simple application for any purpose.

Experiment No. 2 - Explore Version Control Systems	**2 Hours**

1. Aim and Objectives: Explore Git Version control system.

2. Experimentation:
   - Practice Source code management on GitHub. Experiment with the source code written in exercise

• Practice core Git Commands for Configuration and status.	
Experiment No. 3 - Jenkins installation and setup	**2 Hours**
1. Aim and Objectives: Jenkins installation and setup, explore the environment. 2. Experimentation: Install Java Development Kit (JDK), Download Jenkins, Run the Installer/Deploy WAR, Initial Setup (Web Interface)	
Experiment No. 4 – CICD using Jenkins	**4 Hours**
1. Aim and Objectives: Demonstrate continuous integration and development using Jenkins. 2. Experimentation: 1 Code Commit 2. Trigger 3. Build	
Experiment No. 5 - Docker commands for content management.	**2 Hours**
1. Aim and Objectives: Explore Docker commands for content management. 2. Experimentation: managing a content management system (CMS) with Docker involves commands for image management, container lifecycle, data persistence, and orchestration	
Experiment No. 6 - Containerized application	**4 Hours**
1. Aim and Objectives: Develop a simple containerized application using Docker 2. Experimentation: Developing a simple containerized application involves three main steps: writing the application code, creating a Docker file to define its environment, and using Docker commands to build and run the container.	
Experiment No. 7 - Integrate Kubernetes and Docker.	**4 Hours**
1. Aim and Objectives: Integrate Kubernetes and Docker. 2. Experimentation: Integrating Kubernetes (K8s) and Docker involve using Docker as the container runtime for the containerized applications that Kubernetes manages and orchestrates.Docker primarily builds the container images and runs containers on a single host, while Kubernetes manages clusters of these containers across multiple hosts for scalability, high availability, and automated management.	
Experiment No. 8 – Managing deployment	**4 Hours**
1. Aim and Objectives: Automate the process of running containerized application developed in exercise 7 using Kubernetes. 2. Experimentation: To automate running the containerized application developed in the previous step, you will use Kubernetes (K8s). Kubernetes uses declarative YAML files to define the desired state of your application and automatically manages its deployment, scaling, and self-healing	
Experiment No. 9 – Build and Test	**4 Hours**
1. Aim and Objectives: Install and Explore Selenium for automated testing. 2. Experimentation: Installing and exploring Selenium involves setting up your environment, installing the necessary libraries, downloading a browser driver, and writing a simple	

automation script.	
Experiment No. 10 - Run automated tests	**2 Hours**
1. Aim and Objectives: Write a simple program in Java Script/ python and perform testing using Selenium. 2. Experimentation: Develop test cases for the above containerized application using selenium.	
Experiment No. 11 - Release and deploy	**2 Hours**
1. Aim and Objectives: Finalize the tested artifact and prepare it for deployment, often storing it in a package repository and automate the deployment of the application artifact from the repository to a target environment (staging, production).	
Textbooks:	
1. The DevOps Handbook: How to Create World-Class Agility, Reliability, & Security in Technology Organizations, Second Edition (Full Colour Edition). 2. Real-World DevOps Practices, Wiley (17 October 2024); Wiley India Pvt Ltd. 3. Hands-on DevOps with Linux, BPB Publications (24 March 2021)	
Reference Books:	
1. The DevOps Handbook: How to Create World-Class Agility, Reliability & Security in Technology Organizations, written by Gene Kim, Jez Humble, Patrick Debois, and John Willis, IT Revolution Press in 2016. 2. Effective DevOps: Building a Culture of Collaboration, Affinity, and Tooling at Scale by Jennifer Davis and Ryn Daniels, published by O'Reilly Media, Inc.2016 3. he DevOps Adoption Playbook: A Guide to Adopting DevOps in a Multi-Speed IT Enterprise by Sanjeev Sharma , published by Wiley, 2017	

Title of the Course: Project-I	L	T	P	Credits
Course Code: UCSIL0771	-	-	8	4

Course Pre-Requisite: Data Structures, Software Project Management, Database Management Systems, Programming Language

Course Description:

Project Phase–I focus on identifying a real-world engineering problem, conducting literature review, performing feasibility study, designing the system architecture, and developing a preliminary prototype. Students work in teams under the guidance of a faculty mentor.

Course Learning Objectives:

1. Identify real-world technical problems through need analysis and feasibility study.
2. Conduct structured literature reviews to classify existing work and identify gaps.
3. Design suitable system architecture and methodology based on project requirements.
4. Develop an initial prototype demonstrating core functionalities of the proposed system.

Course Outcomes:

COs	After Completion of the course, the student should be able to	Bloom's Cognitive	
		Level	Descriptor
CO1	Identify and understand real-world technical problems based on need analysis and feasibility study.	2	Understand
CO2	Apply structured literature review methods to classify existing research and identify gaps.	3	Apply
CO3	Analyze requirements and design suitable system architecture, models, and methodologies.	4	Analyze
CO4	Develop an initial prototype demonstrating core components of the project.	5	Develop

CO-PO Mapping:

CO	PO1	PO2	PO3	PO 4	PO 5	PO6	PO7	PO8	PO9	PO 10	PO11	PSO 01	PSO 02
CO1	2	3	1	3	2	2	3	1	2	3	3	2	1
CO2	2	2	1	2	2	1	3	3	2	2	3	2	1
CO3	2	2	2	2	2	2	3	3	2	3	2	3	3
CO4	2	1	3	1	1	3	3	3	2	2	2	3	2

Assessment:

One component of In Semester Evaluation (ISE) and One End Semester Examination (ESE).

Assessment	Marks
ISE	50
ESE (POE)	50

ISE (Internal Sessional Evaluation)

ISE will assess students based on continuous project progress, including problem identification, literature review, system design, prototype development, and regular reviews conducted by the project guide.

ESE – POE (End Semester Examination – Practical/Oral Examination)

ESE-POE will evaluate the final Phase-I project work through a formal presentation, oral examination, and demonstration of the prototype before an evaluation panel.

Course Contents:

Guidelines for Students – Phase–I

A. Selecting a Project

- Choose a topic in any CSE domain (AI/ML, IoT, Web, Cloud, Security, Data Science, etc.).
- Ensure the problem is relevant, feasible, and technically sound.
- Projects must involve design, development, experimentation, or system integration.
- **B.** Forming Teams
- Teams of 2–4 students are recommended.
- Every team must choose a faculty project guide (mentor).

C. Problem Identification

- Understand the context of the problem (industry, societal needs, research gap).
- Conduct requirement gathering or stakeholder analysis where applicable.
- Prepare a Problem Statement clearly describing the issue.

D. Feasibility Study

Perform a feasibility check in terms of:

- Technical feasibility

- Resource and data availability
- Time feasibility for two semesters
- Risk assessment and constraints

E. Literature Review Guidelines

- Review 10–20 recent research papers (preferably from 2018 onward).
- Identify existing solutions and their limitations.
- Highlight research gaps leading to your proposed solution.
- Summarize findings in a structured format.

F. Designing the Proposed Solution

- Develop system architecture, models, or flow diagrams.
- Choose the tools, technologies, programming languages, frameworks, and datasets.
- Define the methodology (algorithmic approach, models, workflow).

G. Prototype Development

- Implement 20–30% of the core system.
- Show feasibility of the approach.
- Maintain version control using Git/GitHub.

H. Documentation Requirements

Students must submit a Phase–I Report containing:

- Title, abstract
- Problem definition
- Literature review
- Proposed system & architecture
- Methodology
- Prototype results
- References (IEEE/APA)
- Plagiarism certificate

I. Presentation Guidelines

- Each team must deliver a **Phase–I presentation** and prototype demonstration.
- Explain problem significance, approach, and initial results clearly.

Title of the Course: Financial Analytics **Course Code:** UCSMM0741	**L**	**T**	**P**	**Credits**
	3	-	-	3

Course Pre-Requisite: Business Statistics, Multivariate Data Analysis, Basic Financial Awareness

Course Description:

This course introduces students to the concepts, methods, and tools of financial analytics with a strong emphasis on data-driven financial decision-making. The course focuses on analysing financial data, modelling financial problems, assessing risk, valuing assets, and supporting strategic financial decisions using analytical techniques.

Course Learning Objectives:

1. To understand the role of analytics in modern financial decision-making.
2. To apply statistical and optimization techniques to financial problems.
3. To analyze financial data for risk, return, and performance evaluation.
4. To develop analytical models for valuation, forecasting, and portfolio decisions.
5. To interpret and communicate financial insights effectively.

Course Outcomes:

CO	**After Completion of the course, the student should be able to**	**Bloom's Cognitive**	
		Level	**Descriptor**
CO1	Explain fundamental concepts of financial markets, instruments, and financial data used in analytics.	2	Understand
CO2	Apply financial statement analysis and basic analytical measures to evaluate financial performance.	3	Apply
CO3	Analyse risk and return characteristics of financial assets using statistical and probabilistic models.	4	Analyze
CO4	Dissect portfolio and valuation decisions using optimization and forecasting techniques.	4	Analyze

CO-PO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO01	PSO02
CO1	2	1	1	1	1					1	1	2	2
CO2	2	2	2	2	1					1	1	2	2
CO3	2	2	2	2	2					1	1	2	2
CO4	2	3	3	3	2					1	1	2	2

Assessment:

Assessment	Marks
ESE	100

ESE: Assessment is based on 100% course contents

Course Contents:

Unit 1: Introduction to Financial Analytics	**08 Hours**

Overview of financial analytics and its importance, Types of financial data: market data, accounting data, macroeconomic data, Financial markets and instruments: equity, debt, derivatives, Role of analytics in corporate finance, investments, and risk management, Review of prerequisite concepts: statistics and optimization in finance

Unit 2: Financial Statement and Performance Analytics	**08 Hours**

Financial statements: income statement, balance sheet, cash flow statement, Financial ratio analysis: liquidity, profitability, leverage, efficiency, Trend and comparative analysis, Multivariate analysis for financial performance evaluation

Unit 3: Risk and Return Analytics	**07 Hours**

Concept of risk and return in finance, Descriptive and inferential statistics for financial returns, Probability distributions in finance (normal and empirical distributions), Risk measures: variance, standard deviation, beta, Value at Risk (VaR), Correlation and diversification benefits

Unit 4: Portfolio Analytics and Optimization	**08 Hours**

Portfolio theory and diversification, Markowitz mean–variance portfolio model, Efficient frontier and portfolio selection, Application of optimization techniques to portfolio allocation, Introduction to constraint-based portfolio optimization

Unit 5: Valuation and Forecasting Analytics	07 Hours
Time value of money and discounted cash flow (DCF), Equity valuation models: dividend discount model, basic cash flow models, Regression-based financial forecasting, Time series concepts for financial data, Applications in capital budgeting and investment decisions	
Unit 6: Advanced Applications and Case Studies	**07 Hours**
Financial fraud detection using analytics, Credit risk and scoring models, Financial analytics in banking and fintech, Integration of social and alternative data in finance, Case studies in financial analytics and decision-making	
Textbooks:	
1. Simon Benninga, Financial Modeling, 4th Edition, MIT Press, 2014 2. David Ruppert, Statistics and Data Analysis for Financial Engineering, 2nd Edition, Springer, 2015 3. John C. Hull, Risk Management and Financial Institutions, 5th Edition, Pearson Education, 2018	
Reference Books:	
1. Ruey S. Tsay, Analysis of Financial Time Series, 3rd Edition, Wiley, 2010 2. Frank J. Fabozzi, Investment Management, 2nd Edition, Pearson Education, 2009 3. Marcos López de Prado, Advances in Financial Machine Learning, 1st Edition, Wiley, 2018	

Title of the Course: IoT with Arduino, ESP, and Raspberry Pi **Course Code:** UCSMM0742	**L**	**T**	**P**	**Credits**
	3	-	-	3

Course Pre-Requisite: Digital Logic, Microprocessor, IoT

Course Description:

The course provides a comprehensive theoretical foundation in the Internet of Things (IoT) with a focus on embedded hardware ecosystems such as Arduino, ESP8266/ESP32, and Raspberry Pi. Students will study IoT system architecture, device-level design concepts, communication models, and networking protocols.

Course Learning Objectives:

1. Understand basic IoT concepts, architecture, and communication models, and their applications.
2. Learn the theoretical architecture and working principles of Arduino and its role in IoT.
3. Study the architecture and networking features of ESP8266/ESP32 with focus on WiFi and IoT communication.
4. Understand the architecture and operating principles of Raspberry Pi, including ARM processors.

Course Outcomes:

CO	**After Completion of the course, the student should be able to**	**Bloom's Cognitive**	
		Level	**Descriptor**
CO1	Classify IoT architecture, models, and applications.	2	Understand
CO2	Explain internal architectures and concepts of Arduino, ESP, and Raspberry Pi	2	Understand
CO3	Examine IoT communication technologies and protocols	4	Analyze
CO4	Analyze real-world IoT systems through theory-based design and case studies	4	Analyze

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	2			2	2	1	2				3	2	2
CO2	2	2	1		2		1				2	2	2
CO3	1	2	1			1			3	1	3	2	2
CO4	2	1	1		2	2	3		3	3	2	3	2

Assessment:

SN	Assessment	Marks	Remark
1	End Semester Examination (ESE)	100	100% course contents

Course Contents:

Unit1	Fundamentals of IoT	7 Hours

Definition, Evolution, and Scope of IoT, IoT Architecture: Perception, Network, Middleware & Application Layers, Characteristics of IoT Systems, Overview of Embedded Networks in IoT, IoT Communication Models (D2D, D2C, Gateway-based Communication), IoT Challenges: Privacy, Security, Scalability, Interoperability.

Unit 2	Arduino Concepts and Architecture	8 Hours

Introduction to Arduino Platform, Hardware Architecture of Arduino (UNO, Mega, Nano), Microcontroller Basics: ATmega328P Architecture Overview, Power Supply, I/O Ports, Clocks, Role of Arduino in IoT System Design, Serial Communication Concepts: UART, I2C, SPI (Theory Framework), Typical Use Cases of Arduino in IoT Systems.

Unit 3	ESP8266 and ESP32: Architecture & IoT Connectivity	8 Hours

Overview of ESP8266/ESP32: Features and Internal Architecture, Wi-Fi SoC Fundamentals, TCP/IP Stack Basics, Wi-Fi Standards (802.11 b/g/n) and IoT Applications, Comparison: Arduino vs ESP Modules, Cloud Connectivity Concepts with ESP: Data Transmission, Cloud Endpoints, Use Cases: Low-Cost IoT Devices, Home Automation, Sensors Networks .

Unit 4	Raspberry Pi: Architecture & OS Concepts	7 Hours

Introduction to Raspberry Pi and Its Generations, Broadcom SoC Architecture, ARM CPU Basics, Understanding GPIO, Raspberry Pi Operating System Overview, Applications of Raspberry Pi in Edge Computing.

Unit 5	IoT Networking and Communication Protocols	7 Hours

Wireless Technologies for IoT: Wi-Fi, Bluetooth, BLE, ZigBee, LoRaWAN ,IoT Protocol Stack Overview, Application Layer Protocols: MQTT, CoAP, HTTP/HTTPS, Transport Layer: TCP vs UDP for IoT,IPv6 and 6LoWPAN Concepts.

Unit 6	IoT System Design, Architecture & Case Studies	7 Hours

End-to-End IoT Architecture, Data Acquisition Models: Sampling, Filtering ,IoT Cloud Platforms Overview: AWS IoT, Azure IoT Hub, Google IoT Core, Interoperability and Standardization in IoT,Reliability, Fault Tolerance, and Quality of Service in IoT Systems, Case Studies: Smart Home Automation, Smart Agriculture System, Environment Monitoring, Health Monitoring System, Industrial IoT.

Text Books:

The Internet of Things: Enabling Technologies, Platforms, and Use Cases — Pethuru Raj & Anupama C. Raman

Reference Books:

1. Internet of Things with Raspberry Pi and Arduino — Rajesh Singh, Anita Gehlot, Lovi Raj Gupta, Bhupendra Singh & Mahendra Swain
2. Internet of Things (A Hands-On Approach) — Arshdeep Bahga & Vijay Madisetti

Title of the Course: Algorithms in Computational Biology	**L**	**T**	**P**	**Credits**
Course Code: UCSMM0743	3	-	-	3

Course Pre-Requisite: Data Structures and Algorithms, Basic understanding of Biology, Programming Skills

Course Description:

This course introduces fundamental algorithms and computational techniques used in molecular biology and bioinformatics. Students will learn to apply algorithmic thinking to solve problems in genomics, proteomics, and personalized medicine. The course emphasizes both theoretical understanding and practical tool usage relevant to the Smart Healthcare domain.

Course Learning Objectives:

1. To understand the computational challenges in analysing biological data.
2. To learn dynamic programming approaches for sequence alignment.
3. To comprehend multiple sequence alignment and evolutionary tree construction.
4. To study pattern matching algorithms for motif discovery in biological sequences.
5. To explore genome assembly techniques and their applications.
6. To apply computational biology methods to healthcare and precision medicine.

Course Outcomes:

CO	**After Completion of the course, the student should be able to**	**Bloom's Cognitive**	
		Level	**Descriptor**
CO1	Explain dynamic programming algorithms for sequence alignment and comparison problems.	2	Understand
CO2	Apply multiple sequence alignment and phylogenetic methods for biological sequence analysis.	3	Apply
CO3	Apply pattern matching and genome assembly algorithms for biological data analysis.	3	Apply
CO4	Analyze computational biology techniques for healthcare applications and precision medicine.	4	Analyze

CO-PO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO01	PSO02
CO1	3	3	2	2	2						2	2	2
CO2	3	3	2	2	2						2	2	2
CO3	3	3	3	2	2						2	2	3
CO4	2	2	2	2	2	1	1				2	1	3

Assessment:

Assessment	Marks
ESE	100

ESE: Assessment is based on 100% course contents

Course Contents:

Unit 1: Introduction to Computational Biology	06 Hours

Introduction: Computational biology and Bioinformatics, Basics of DNA, RNA, proteins, Central dogma of molecular biology, biological sequences and representation, Introduction to databases (NCBI, UniProt), File formats: FASTA and GenBank, Applications in healthcare: precision medicine, disease diagnosis

Unit 2: Pairwise Sequence Alignment	09 Hours

Biological motivation for sequence alignment, Dot matrix method, Dynamic programming fundamentals, Global alignment: Needleman-Wunsch algorithm, Scoring schemes: match, mismatch, gap penalties, Local alignment: Smith-Waterman algorithm, Heuristic methods: BLAST algorithm Time and space complexity analysis

Unit 3: Multiple Sequence Alignment	09 Hours

Multiple sequence alignment problem, Computational complexity and challenges, Progressive alignment methods, ClustalW algorithm, Guide trees and pairwise alignment, Consensus sequences and conserved regions, Applications in protein family analysis

Unit 4: Phylogenetics and Pattern Matching	07 Hours

Evolutionary trees and phylogenetic analysis, Tree representation and interpretation, Distance-based methods: UPGMA algorithm, Overview of Neighbor-Joining and Maximum Parsimony, Pattern

matching in biological sequences, Exact string-matching algorithms, Motifs and regulatory elements Position Weight Matrix (PWM)	
Unit 5: Genome Assembly	**08 Hours**
DNA sequencing technologies overview, Genome assembly problem formulation, Fragment assembly approaches, Greedy algorithms for shortest superstring, Overlap graphs, De Bruijn graph representation, Challenges: repeats and sequencing errors, Gene finding and Open Reading Frame (ORF) detection	
Unit 6: Applications in Healthcare	**06 Hours**
RNA secondary structure basics, Protein structure prediction overview, Drug target identification using sequence analysis, Personalized medicine and pharmacogenomics, Gene expression analysis fundamentals, CRISPR target prediction	
Textbooks:	
1. Jones, N.C. and Pevzner, P.A., "An Introduction to Bioinformatics Algorithms", MIT Press, 2004 2. Durbin, R., Eddy, S., Krogh, A., and Mitchison, G., "Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids", Cambridge University Press, 1998 3. Setubal, J.C. and Meidanis, J., "Introduction to Computational Molecular Biology", Brooks/Cole Publishing Company, 1997	
Reference Books:	
1. Waterman, M.S., "Introduction to Computational Biology: Maps, Sequences and Genomes", Chapman & Hall/CRC Press, 1995 2. Mount, D.W., "Bioinformatics: Sequence and Genome Analysis", Cold Spring Harbor Laboratory Press, 2nd Edition, 2004 3. Pevzner, P.A. and Shamir, R., "Bioinformatics for Biologists", Cambridge University Press, 2011	

Semester VIII												
Sr. No.	**Category**	**Course Code**	**Course Name**	**L**	**T**	**P**	**Hrs/ Week**	**Credits**	**Evaluation Scheme (Component)**			
1	PEC	UCSPE081*	Program Elective - IV (PE-IV)	3	-	-	3	3	ISE1	10		40
									MSE	30		
									ISE2	10		
									ESE	50	20	
2	PEC	UCSPE081*	Program Elective - V (PE-V)	3	-	-	3	3	ISE1	10		40
									MSE	30		
									ISE2	10		
									ESE	50	20	
3	RP	UCSIL0871	Project-II	-	-	8	8	4	ISE I	50	20	
									ESE (OE)	50	20	
4	OJT	UCSIL0872	Internship	-	-	12	12	6	ISE1	75	30	
									ISE2	75	30	
5	CC	UCSCC0831	Co-Curricular Activities-IV	-	-	2	2	1	ISE	50	20	
							Total	**28**	**17**	**Total Marks: 500 Total Credit: 17**		

Program Elective IV (PE-IV)		
Sr. No.	**Course Code**	**Course Name**
1	UCSPE0811	Computer Vision
2	UCSPE0812	Cyber Security & Forensics
3	UCSPE0813	Robotic Process Automation

Program Elective V (PE-V)		
Sr. No.	**Course Code**	**Course Name**
1	UCSPE0814	Introduction to Blockchain
2	UCSPE0815	High Performance Computing
3	UCSPE0816	Quantum Computing

Title of the Course: Computer Vision	**L**	**T**	**P**	**Credits**
Course Code: UCSPE0811	3	-	-	3

Course Pre-Requisite: Programming including algorithms and data structures, Python: Students should either have prior experience with Python or be able to quickly learn a new language.

Course Description: Computer Vision focuses on development of algorithms and techniques to analyze and interpret the visible world around us. This requires understanding of the fundamental concepts related to multi-dimensional signal processing, feature extraction, pattern analysis, visual geometric modeling, stochastic optimization etc. Knowledge of these concepts is necessary in this field, to explore and contribute to research and further developments in the field of computer vision. Applications range from Biometrics, Medical diagnosis, document processing, mining of visual content, to surveillance, advanced rendering etc.

Course Learning Objectives:

1. To impart knowledge of advanced techniques in computer vision.
2. To acquaint students with the concepts of colour image processing, texture analysis, object recognition, video processing, 3D imaging etc. by applying the algorithms to build applications.
3. To allow students to compare various algorithms and select the one most appropriate for a particular application.

Course Outcomes:

CO	**After Completion of the course, the student should be able to**	**Bloom's Cognitive**	
		Level	**Descriptor**
CO1	Explain the Computer vision fundamentals	2	Understand
CO2	Apply the concepts of color image processing, fundamentals of texture analysis, object recognition methods, video processing concepts, 3D imaging.	3	Apply
CO3	Choose appropriate algorithms and evaluate results.	3	Apply
CO4	Analyse problems and algorithms to build solutions to the real-world computer vision problems	4	Analyse

CO-PO Mapping:

CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PSO 01	PSO 02
CO1	1	1		1							3		2
CO2	1	3	2	3							2	1	3
CO3	2	3	2	3	3						3	3	3
CO4	2	2	2	2	2						2	2	1

Assessment:

Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one End Semester Examination (ESE) having 20%, 30% and 50% weights respectively.

Assessment	Marks
ISE-1	10
MSE	30
ISE-2	10
ESE	50

ISE-1 and **ISE-2** are based on assignment/declared test/quiz/seminar/Group Discussions etc.

MSE: Assessment is based on 50% of course content (Normally first three modules)

ESE: Assessment is based on 100% course content with 60-70% weightage for course content (normally last three modules) covered after MSE.

Course Contents:

Unit 1: Introduction	**7 Hours**

Overview, computer imaging systems, lenses, Image formation and sensing, Image analysis, pre-processing and Binary image analysis.

Unit 2: Texture Analysis	**7 Hours**

Definition, Types of texture, Texels, Texture analysis – concept and categories, Approaches to texture analysis, Statistics, Texture descriptors - statistical - Auto-correlation, co-occurrence matrices and features, edge density and direction, local binary partition, Law's texture energy measures, Wavelets and texture analysis.

Unit 3: Object Recognition & Restoration	**8 Hours**

Object Recognition: Object Detection Vs recognition, Patterns and Pattern Classes, Knowledge Representation, Statistical Pattern Recognition, Neural Nets, Syntactic Pattern Recognition, Optimization Techniques in Recognition.

Restoration: Image Restoration Model, Noise Models, Restoration using spatial filtering.	
Unit 4: The Geometry of Multiple Views	**8 Hours**
Two Views, Stereopsis: Reconstruction, Human Stereposis, Binocular Fusion, Using More Cameras, Segmentation by Clustering: What Is Segmentation? Human Vision: Grouping and Gestalt, Applications: Shot Boundary Detection and Background Subtraction, Image Segmentation by Clustering Pixels.	
Unit 5: Moving Object Detection and Tracking	**8 Hours**
Introduction, Background Modeling, Connected Component Labeling, Shadow Detection, Single Object Tracking, Discrete Kalman Filtering, Particle-filter based tracking, Mean-shift tracking, Segmentation tracking via graph cuts	
Unit 6: 3 D Vision	**7 Hours**
Introduction to 3D imaging and its applications. Study of any Research Paper(s) based on the current trends in 3D imaging or any case study	
Textbooks:	
1. Computer Vision: Algorithms and Applications by Richard Szeliski. Available for free online. 2. Computer Vision: A Modern Approach (Second Edition) by David Forsyth and Jean Ponce. Available for free online. 3. Elements of Statistical Learning by Trevor Hastie, Robert Tibshirani, and Jerome Friedman. Available for free online	
Reference Books:	
1. Richard Hartley and Andrew Zisserman, “Multiple View Geometry in Computer Vision”, Second Edition, Cambridge University Press, 2004.	

Title of the Course: Cyber Security & Digital Forensics Course Code: UCSPE0812	L	T	P	Credits
	3	-	-	3

Course Pre-Requisites: Fundamentals of Operating Systems, Computer Networks, Information Security

Course Description:

This course introduces the fundamentals of cyber security and digital forensics, covering cyber threats, defence mechanisms, and investigative techniques. Students learn ethical hacking, secure communication practices, and methods for acquiring and analysing digital evidence. The course provides essential skills for system protection, incident response, and cybercrime investigation

Course Learning Objectives:

1. Understand the fundamental concepts of cyber security, including threats, vulnerabilities, attacks, and security controls.
2. Demonstrate the ability to identify, analyze, and mitigate common cyber-attacks using ethical hacking and vulnerability assessment techniques.
3. Apply digital forensic processes for acquiring, preserving, analyzing, and documenting digital evidence in compliance with legal and ethical guidelines.
4. Develop skills to perform basic incident response, log analysis, and forensic reporting using industry-standard tools and methodologies.

Course Outcomes:

CO	After Completion of the course, the student should be able to	Bloom's Cognitive	
		Level	Descriptor
CO1	Define the fundamental concepts of cyber security,	1	Remember
CO2	Classify the cybercrimes and analyze major cyber threats	**2**	Understand
CO3	Explain security policies, standards, cyber terrorism and cyber forensics concepts	2	Understand
CO4	Apply digital forensic investigations for lawful evidence acquisition, preservation, analysis, and interpretation.	3	Apply
CO5	Identify and analyze security weaknesses in networks and systems using the vulnerability assessment and ethical hacking activities	4	Analyze

CO-PO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO01	PSO02
CO1	1				3		3				3	2	2
CO2	1	2	1	1	3	2	3				2	3	2
CO3	1	3		1	3	2	3				3	1	2
CO4	1	2	1	2	3		3				3	3	2
CO5	1	2	1	1	3		3				3	3	2

Assessment:

Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one End Semester Examination (ESE) having 20%, 30% and 50% weights respectively.

Assessment	**Marks**
ISE-1	10
MSE	30
ISE-2	10
ESE	50

ISE-1 and **ISE-2** are based on assignment/declared test/quiz/seminar/Group Discussions etc.

MSE: Assessment is based on 50% of course content (Normally first three modules)

ESE: Assessment is based on 100% course content with 60-70% weightage for course content (normally last three modules) covered after MSE.

Course Contents:

Unit 1: Introduction to Cyber Security	**7 Hours**

Defining Cyberspace, Architecture of cyberspace, Concept of cyber security, Cyber security increasing threat landscape, Cyber security terminologies- Cyberspace, attack, attack vector, attack surface, threat, risk, vulnerability, exploit, exploitation, hacker., Non-state actors, Cyber terrorism, Protection of end user machine, Critical IT and National Critical Infrastructure, Cyber warfare.

Unit 2: Cyber Crime & Information security	**8 Hours**

Classification of cybercrimes, Common cybercrimes- cybercrime targeting computers and mobiles, cybercrime against women and children, financial frauds, social engineering attacks, malware and ransomware attacks, zero day and zero click attacks, Online scams and frauds- email scams, Phishing,

Vishing, Smishing, Online job fraud, Online sextortion, Debit/ credit card fraud, Online payment fraud, Cyber bullying, website defacement, Cybersquatting, Pharming, Cyber espionage, Crypto jacking, Dark net- illegal trades, drug trafficking, human trafficking., Social Media Scams & Frauds-impersonation, identity theft, job scams, misinformation, cyber stalking.	
Unit 3: Tools and methods used in Cybercrime	**8 Hours**
Proxy servers and Anonymizers, Phishing, Password cracking, keyloggers and spywares, Virus And worms, Trojan horse, Backdoors, DoS and DDoS attacks, Steganography, SQL injection, Buffer overflow, Attacks on Wireless networks	
Unit 4: Cyber Forensics	**7 Hours**
Computer Forensics Definition and Cardinal Rules Types of Computer Forensics Technology – Types of Computer Forensics Systems – Vendor and Computer Forensics Services. Computer forensics evidence and capture: Data Recovery – Evidence Collection and Data Seizure-Duplication and Preservation of Digital Evidence-Computer Image Verification and Authentication.	
Unit 5: Forensic Tools and Processing of Electronic Evidence	**8 Hours**
Introduction to Forensic Tools, Usage of Slack space, tools for Disk Imaging, Data Recovery, Vulnerability Assessment Tools, Encase and FTK tools, Anti Forensics and probable counters, retrieving information, process of computer forensics and digital investigations, processing of digital evidence, digital images, damaged SIM and data recovery, multimedia evidence, retrieving deleted data: desktops, laptops and mobiles, retrieving data from slack space, renamed file, ghosting, compressed files.	
Unit 6: **Cybercrimes & cyber security: Legal, social , ethical and Psychological perspectives**	**7 Hours**
Introduction, Cybercrime & legal landscape around the world, The Indian IT Act, Challenges to Indian law & cybercrime scenario in India, Digital Signatures & Indian IT Act, Cybercrime and Punishment, Cyber law Technology & students Intellectual Property in the cyberspace, Ethical dimensions in cybercrimes, Mindset and skills of hackers and other cybercriminals	
Textbooks:	
1. Cyber Security Understanding Cyber Crimes, Computer Forensics and Legal Perspectives by Sumit Belapure and Nina Godbole, Wiley India Pvt. Ltd. (Unit 1 , 2, 3, 6) 2. John R. Vacca, "Computer Forensics: Computer Crime Scene Investigation", Cengage Learning, 2nd Edition, 2005 (Unit 4) 3. C. Altheide & H. Carvey Digital Forensics with Open Source Tools, Syngress, 2011. ISBN:	

9781597495868 (Unit 5)

4. Cyber law-The Indian Perspective, Pawan Duggal (2009)

Reference Books:

1. Computer Security Fundamentals - Chuck Easttom, Pearson, third edition.
2. Jason Luttgens, Matthew Pepe, Kevin Mandia, Incident Response & Computer Forensics, McGraw-Hill Osborne Media, 3rd edition , 2014.
3. Keith J. Jones, Richard Bejtlich, Curtis W. Rose, Real Digital Forensics: Computer Security and Incident Response, Paperback – Import, 2005.
4. 7 Years of Indian Cyber Laws, Rohas Nagpal (2008)

Title of the Course: Robotic Process Automation Course Code: UCSPE0813	L	T	P	Credits
	3	-	-	3

Course Pre-Requisites: Knowledge of basic computer skills, Knowledge of Programming and Scripting and Knowledge of database

Course Description:

This course provides a comprehensive understanding of the fundamental concepts, tools, and techniques required to automate repetitive and rule-based tasks using software robots. Students will gain hands-on experience with popular RPA tools and learn how to design, implement, and manage RPA solutions in various industries.

Course Learning Objectives:

1. Understand the fundamentals of RPA, its benefits, and the types of robots used in automation.
2. Explain the components of RPA tools, control flow activities, and the RPA development life cycle.
3. Illustrate techniques for data manipulation and basic UI automation using selectors..
4. Apply data extraction and automation techniques to desktop, web, and email processes.
5. Implement error and exception handling techniques, and use Orchestrator for automation management.
6. Apply advanced RPA features and best practices to real-time use cases and industry-specific scenarios.

Course Outcomes:

CO	After Completion of the course, the student should be able to	Bloom's Cognitive	
		Level	Descriptor
CO1	Define the basic concepts of Robotic Process Automation.	1	Remember
CO2	Illustrate RPA with use of different types of variables and control flow activities.	2	Understand
CO3	Explain different data manipulation and automation techniques.	2	Understand
CO4	"Demonstrate the application of user-event handlers and various exception strategies to maintain system stability within a specific use case."	3	Apply

CO-PO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO 01	PSO 02
CO1	2	1		1	2							1	
CO2	1		2		2				1			2	1
CO3	1	2	1	1	1				1		1	1	2
CO4	2	2	3	3	3	2					3	1	2

Assessment:

Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one End Semester Examination (ESE) having 20%, 30% and 50% weights respectively.

Assessment	**Marks**
ISE-1	10
MSE	30
ISE-2	10
ESE	50

ISE-1 and **ISE-2** are based on assignment/declared test/quiz/seminar/Group Discussions etc.

MSE: Assessment is based on 50% of course content (Normally first three modules)

ESE: Assessment is based on 100% course content with 60-70% weightage for course content (normally last three modules) covered after MSE.

Course Contents:

Unit 1: Introduction to RPA	**6 Hours**

Robotic Process Automation: History of Automation, Automation and its benefits, Introduction to RPA, Automation vs RPA, RPA Programming Constructs.
Process Understanding: Process and Flowchart, Robots in RPA, Introduction to Robots, Types of Robots, Benefits and Implementation of RPA.

Unit 2: Introduction to RPA Tools	**8 Hours**

RPA Tool Basics: RPA Development Life Cycle, Working of RPA, Challenges in RPA, Variables and its Types, Arguments, Variables vs Arguments, Namespaces, and Importing New Namespace.
Control Flow Activity: Sequences, Control Flow and its types, Decision control (IF, Switch, IF vs Switch), Loops (Do While, While, For Each), Other Control Flow Activities (Delay, Break, Assign, Continue, Parallel).

Unit 3: Data Manipulation and UI Automation	**7 Hours**

Data Manipulation: Importance of Data Manipulation, String Manipulations, Data Table Manipulations, Data Collection, Types, and Manipulations. UI Automation & Selectors: UI interactions, Input actions, Input methods, Containers, Recording & its types, Selectors (Full, Partial, Containers, Dynamic Selectors).	
Unit 4: Automation Concepts and Techniques	**8 Hours**
Data Extraction: Desktop and Web Recording, Extraction techniques (Screen Scraping, Data Scraping, PDF Extraction). Automation Techniques: Workbook and Excel Automation (Read/Write). Email Automation: Incoming Email Automation, Sending Email Automation	
Unit 5: Error and Exception Handling	**8 Hours**
Error and Exception Handling: Errors, Error handling approaches, Try-Catch, Retry Scope, Exception Handling, Types of Exceptions, Global Exception Handler, Best Practices for Error Handling. Orchestrator: Overview, Orchestrator Functionalities, Orchestrator UI Automations, Management, Monitoring.	
Unit 6: **Advanced RPA and Real-time Use Cases**	**7 Hours**
Advanced RPA Features: Handling Structured & Unstructured Data, API Integration in RPA, Database Connectivity. Best Practices & Optimization: RPA Project Implementation Steps, Performance Optimization, Security in RPA. Use Cases & Case Studies: Real-world RPA Implementation Scenarios, Industry-wise RPA Use Cases	
Textbooks:	
1. Tom Taulli, The Robotic Process Automation Handbook: A Guide to Implementing RPA Systems,2020, ISBN-13 (electronic): 978-1-4842-5729-6, Publisher: A press 2. Alok Mani Tripathi, Learning Robotic Process Automation, Publisher: Packet Publishing Release Date: March 2018 ISBN: 9781788470940	
Reference Books:	
1. Frank Casale, Rebecca Dilla, Heidi Jaynes, Lauren Livingston, “Introduction to Robotic Process Automation: a Primer”, Institute of Robotic Process Automation. 2. Richard Murdoch, Robotic Process Automation: Guide to Building Software Robots, Automate Repetitive Tasks & Become an RPA Consultant. 3. Srikanth Merianda, Robotic Process Automation Tools, Process Automation and their benefits: Understanding RPA and Intelligent Automation. 4.https://www.uipath.com/rpa/robotic-process-automation	

Title of the Course: Introduction to Blockchain Course Code: UCSPE0814	L	T	P	Credits
	3	-	-	3

Course Pre-Requisite: Computer Network, Information Security, Programming Language

Course Description:

This course will cover the technical aspects of cryptocurrencies, blockchain technologies including distributed consensus, distributed ledger, smart contracts, types of blockchain and real-world case studies along with scalability and privacy challenges. Students will learn how these systems work and how to engineer secure software that interacts with a blockchain system like Ethereum and Hyperledger kind of blockchain.

Course Learning Objectives:

1. Understand the Fundamentals of decentralization, distributed ledger technology and blockchain
2. Understand security aspects of blockchain network
3. Compare the different types of blockchain like bitcoin, Ethereum, Hyperledger and others
4. Discuss real world case studies in the market and understand challenges

Course Outcomes:

CO	After Completion of the course, the student should be able to	Bloom's Cognitive	
		Level	Descriptor
CO1	Explain core concepts of distributed ledger technology	2	Understand
CO2	Compare Bitcoin and Ethereum based cryptocurrency protocol stack	2	Understand
CO3	Illustrate the enterprise blockchain architecture of Hyperledger	2	Understand
CO4	Analyze blockchain use cases and its overall impact	4	Analyze

CO-PO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO01	PSO02
CO1	2	1		1	3						1	3	1
CO2	1	1		1	2					1	2	1	3
CO3	1	2	2	1	3					1	3	3	3
CO4		2		2						1	1		1

Assessment:

Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one End Semester Examination (ESE) having 20%, 30% and 50% weights respectively.

Assessment	Marks
ISE-1	10
MSE	30
ISE-2	10
ESE	50

ISE-1 and **ISE-2** are based on assignment/declared test/quiz/seminar/Group Discussions etc.

MSE: Assessment is based on 50% of course content (Normally first three modules)

ESE: Assessment is based on 100% course content with 60-70% weightage for course content (normally last three modules) covered after MSE.

Course Contents:

Unit 1: Introduction to Blockchain	**7 Hours**

Traditional databases and distributed ledger technology, Distributed systems, History of blockchain, Introduction to blockchain, Types of blockchain, Consensus, CAP theorem and blockchain, Benefits and limitations of blockchain.

Unit 2: Decentralisation and Security	**7 Hours**

Decentralization using blockchain, Methods of decentralization, Routes to decentralization, Decentralized organizations. Cryptography and Technical Foundations: Cryptographic primitives, Asymmetric cryptography, Public and private keys.

Unit 3: Bitcoin and Alternative Coins	**6 Hours**

Bitcoin, Digital Keys and address, transactions, blockchain, mining, Bitcoin Network wallets, Bitcoin payments, Bitcoins limitations and alternative coins Litecoin, Primecoin and zcash.	
Unit 4: Ethereum and Smart Contracts	**8 Hours**
Ethereum network, Ethereum components, Transaction, EVM, Programming Languages, Gas, Fee schedule and Supporting protocols, Smart contracts	
Unit 5: Hyperledger and Alternative Blockchain	**8 Hours**
Hyperledger, Projects under Hyperledger, Hyperledger as a protocol, The reference architecture Requirements and design goals of Hyperledger Fabric, Fabric, Sawtooth Lake, Corda, Alternative Blockchains	
Unit 6: Blockchain Outside of Currencies	**6 Hours**
Internet of Things, Government: Border control, Voting, Citizen identification (ID cards), Miscellaneous: Health, Finance, Insurance, Post-trade settlement, financial crime prevention, Media, Scalability and Other Challenges: Scalability, Privacy	
Textbooks:	
1. Mastering Blockchain - Distributed ledgers, decentralization and smart contracts explained, Author- Imran Bashir, Packt Publishing Ltd, Second Edition, ISBN 978-1- 78712-544-5, 2017	
Reference Books:	
1. Bitcoin and Cryptocurrency Technologies, Author- Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller, Steven Goldfeder, Princeton University, 2016 2. Blockchain Basics: A Non-Technical Introduction in 25 Steps, Author- Daniel Drescher, Apress, First Edition, 2017 3. Mastering Bitcoin: Unlocking Digital Cryptocurrencies, Andreas M. Antonopoulos, O'Reilly Media, First Edition, 2014	

Title of the Course: High Performance Computing Course Code: UCSPE0815	L	T	P	Credits
	3	-	-	3

Course Pre-Requisites: Programming skills, Data structures, Computer Algorithms

Course Description:

This course covers the design of advanced modern computing systems. In particular parallel computers and their architectures. It also helps users to choose different parallel programming models for different applications. In this course students are exposed to parallel programming tools such as open MP, MPI and CUDA through which simple parallel programs can be written.

Course Learning Objectives:

1. Introduce the current trends in computer architecture and programming models.
2. Understand Parallel Hardware and Parallel Software.
3. Learn Distributed-Memory Programming with MPI.
4. Learn Shared-Memory Programming with Pthreads.
5. Learn Shared-Memory Programming with Open MP and Program parallel architectures.

Course Outcomes:

CO	After Completion of the course, the student should be able to	Bloom's Cognitive	
		Level	Descriptor
CO 1	Explain different parallel architecture models and terminologies of high performance computing.	2	Understand
CO 2	Choose design methodologies and parallel algorithms for optimization of real world problems.	3	Apply
CO 3	Make use of GPU programming	3	Apply
CO 4	Analyze the Architectures using MPI, threads and Open MP.	4	Analyze

CO-PO Mapping:

CO	PO 1	PO2	PO 3	PO 4	PO5	PO 6	PO 7	PO 8	PO 9	PO 10	PO1 1	PSO0 1	PSO0 2
CO1	3												
CO2		2	2	2									2
CO3	3	3	2	2	3	2	1					3	1
CO4	3	3	2	2	3	2	1					3	1

Assessment:

Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one End Semester Examination (ESE) having 20%, 30% and 50% weights respectively.

Assessment	Marks
ISE-1	10
MSE	30
ISE-2	10
ESE	50

ISE-1 and **ISE-2** are based on assignment/declared test/quiz/seminar/Group Discussions etc.

MSE: Assessment is based on 50% of course content (Normally first three modules)

ESE: Assessment is based on 100% course content with 60-70% weightage for course content (normally last three modules) covered after MSE.

Course Contents:

Unit 1:Introduction to Parallel Systems	**8 Hours**

Introduction to Parallel hardware and software, need for high performance systems and Parallel Programming, SISD, SIMD, MISD, MIMD models, Performance issues.

Unit 2: Process, Threads & Scheduling	**8 Hours**

Processors, PThreads, Thread Creation, Passing arguments to Thread function, Simple matrix multiplication using Pthreads, critical sections, mutexes, semaphores, barriers and conditional variables, locks, thread safety, simple programming assignments

Unit 3: Tools and methods used in Cybercrime	**7 Hours**

Open MP Programming: introduction, reduction clause, parallel-for-loop scheduling, atomic directive, critical sections and locks, private directive, Programming assignments, n body solvers using open MP	
Unit 4: Introduction to MPI programming	**8 Hours**
Computer Forensics Definition and Cardinal Rules Types of Computer Forensics Technology – Types of Computer Forensics Systems – Vendor and Computer Forensics Services. Computer forensics evidence and capture: Data Recovery – Evidence Collection and Data Seizure-Duplication and Preservation of Digital Evidence-Computer Image Verification and Authentication.	
Unit 5: Introduction to GPU computing	**7 Hours**
Introduction to GPU computing, Graphics pipelines, GPGPU, Data Parallelism and CUDA C Programming, CUDA Threads Organization, Simple Matrix multiplication using CUDA, CUDA memories.	
Unit 6: **Bench Marking and Tools**	**7 Hours**
Bench Marking and Tools for High Performance Computing Environments, Numerical Linear Algebra Routines BLAS for Parallel Systems evaluation.	
Textbooks:	
1. An Introduction to Parallel Programming, Peter S Pacheco, Elsevier,2011 2. Programming Massively Parallel Processors, Kirk & Hwu, Elsevier,2012	
Reference Books:	
1. CUDA by example: An introduction to General Purpose GPU Programming, Jason, Sanders, Edward Kandrit, Perason,2011 2. CUDA Programming, Shame Cook, Elsevier 3. High Performance Heterogeneous Computing, Jack Dongarra, Alexey & Lastovetsky, Wiley 4. Parallel computing theory and practice, Michel J. Quinn, TMH	

Title of the Course: Quantum Computing	L	T	P	Credits
Course Code: UCSPE0816	3	-	-	3

Course Pre-Requisite: Linear Algebra, Probability Theory, Algorithms & Data Structures, Basics of Classical Computing & Digital Logic and Programming language

Course Description:

Quantum Computing is an advanced elective course designed to introduce students to the fundamental principles, mathematical foundations, algorithms, and emerging applications of quantum computation. The course develops a strong conceptual understanding of quantum mechanics for computation, qubits and quantum gates, quantum circuits, and quantum algorithms.

Course Learning Objectives:

1. Introduce the limitations of classical computing and the motivation for quantum computing models.
2. Explain the mathematical and physical principles underlying quantum computation.
3. Describe quantum gates, circuits, and standard quantum algorithms.
4. . Discuss quantum hardware technologies, noise sources, and error correction techniques.
5. Demonstrate the use of quantum programming frameworks for implementing basic quantum applications.

Course Outcomes:

CO	After Completion of the course, the student should be able to	Bloom's Cognitive	
		Level	Descriptor
CO1	Recall fundamental quantum mechanical principles.	1	Remember
CO2	Explain the working principles of standard quantum algorithms.	2	Understand
CO3	Interpret qubit technologies, noise effects, error correction, and NISQ limitations.	2	Understand
CO4	Infer the operation of quantum gates, reversible computation, and quantum circuit models.	2	Understand
CO5	Construct quantum circuits and explore real-world quantum computing applications.	3	Apply

CO-PO Mapping:

CO	PO 1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO01	PSO02
CO1	3	2									2	2	1
CO2	3	2	2								2	2	
CO3	3	3	2								2	1	
CO4	3	2									2	1	
CO5	3	2	2		3						3	3	2

Assessment:

Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one End Semester Examination (ESE) having 20%, 30% and 50% weights respectively.

Assessment	Marks
ISE-1	10
MSE	30
ISE-2	10
ESE	50

ISE-1 and **ISE-2** are based on assignment/declared test/quiz/seminar/Group Discussions etc.

MSE: Assessment is based on 50% of course content (Normally first three modules)

ESE: Assessment is based on 100% course content with 60-70% weightage for course content (normally last three modules) covered after MSE.

Course Contents:

Unit 1: Foundations of Quantum Computing	**7 Hours**

Classical vs Quantum: Computation Limits of classical computing, Moore's law slowdown need for quantum models. Quantum Mechanical Principles: Quantum states, Dirac notation, superposition Linear algebra refresh: vectors, matrices, inner product, tensor product. Measurement and probability amplitudes, entanglement and non-locality, no-cloning theorem. Quantum Bits (Qubits), Single qubit states, bloch sphere representation, multi-qubit systems and tensor products

Unit 2: Quantum Gates & Circuits	**7 Hours**

Basic Quantum Gates: Multi-Qubit Gates, Quantum Circuits, Reversible Computing

Unit 3: Quantum Algorithms	8 Hours
Quantum Parallelism, Deutsch and Deutsch–Jozsa Algorithm, Bernstein–Vazirani Algorithm, Simon's Algorithm, Grover's Search Algorithm,Shor's Algorithm.	
Unit 4: Quantum Hardware, Noise & Error Correction	**8 Hours**
Physical Realizations of Qubits: Superconducting qubits, Trapped-ion qubits, Photonic qubits, Semiconductor/Spin qubits, Quantum Noise & Decoherence: T1, T2 times, Sources of noise, Quantum channels, Quantum Error Correction (QEC): Repetition codes, Shor code Fault tolerance basics, NISQ Era (Noisy Intermediate-Scale Quantum): Characteristics of NISQ devices.	
Unit 5: Quantum Computing Frameworks and Tools	**9 Hours**
Quantum Programming Frameworks: IBM Qiskit, Google Cirq, Microsoft Q# and Azure Quantum, Rigetti Forest, Writing Quantum Programs, building circuits on simulators, Running jobs on cloud quantum machines, Quantum Machine Learning Introduction, Quantum data encoding, Variational quantum classifiers, Hybrid quantum-classical models.	
Unit 6: Quantum Computing Applications	**6 Hours**
Cryptography, Optimization problems, Quantum chemistry (VQE), Material science, Finance.	
Textbooks:	
1. 1.Quantum Computation and Quantum Information -Michael Nielsen & Isaac Chuang, Second Edition, Cambridge University Press 2. Quantum Computing: A Gentle Introduction-Eleanor Rieffel & Wolfgang Polak, First Edition, The MIT Press	
Reference Books:	
1. An Introduction to Quantum Computing- Phillip Kaye, Raymond Laflamme, and Michele Mosca, First Edition, Oxford University Press 2. Quantum Computing for Everyone -Chris Bernhardt,The MIT Press 3. Quantum Computing: An Applied Approach-Jack D. Hidary, Second Edition,Springer Cham 4. IBM Qiskit Textbook-Qiskit Community & IBM Research	

Title of the Course: Project–II Course Code: UCSIL0871	L	T	P	Credits
	-	-	8	4

Course Pre-Requisite: Programming Languages, Databases, Design tools

Course Description:

Project Phase–II focuses on full implementation, system integration, testing, performance evaluation, optimization, documentation, and final viva. Students complete and demonstrate the final working system.

Course Learning Objectives:

1. Implement the proposed system by integrating components as per the designed architecture.
2. Test and validate the system using appropriate tools and performance metrics.
3. Analyse experimental results to evaluate system performance against objectives.
4. Optimize the system based on evaluation results and document the outcomes.

Course Outcomes:

COs	After Completion of the course, the student should be able to	Bloom's Cognitive	
		Level	Descriptor
CO1	Apply engineering tools and methodologies to implement the complete project system.	3	Apply
CO2	Analyse and validate system performance using appropriate test strategies and metrics.	4	Analyse
CO3	Evaluate and optimize the system to improve accuracy, efficiency, or usability.	5	Evaluate
CO4	Create professional documentation and effectively present the completed project in viva.	6	Evaluate

CO-PO Mapping:

CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PSO01	PSO02
CO1	2	2	1	2	2	1	3	3	2	2	3	2	1
CO2	2	2	2	2	2	2	3	3	2	3	2	3	3
CO3	2	1	3	1	1	3	3	3	2	2	2	3	2
CO4	2	1	3	1	1	3	3	3	2	2	2	3	2

Assessment:

One component of In Semester Evaluation (ISE) and One End Semester Examination (ESE).

Assessment	Marks
ISE	50
ESE (OE)	50

ISE (Internal Sessional Evaluation)

ISE will assess students based on continuous project progress, including use cases, test cases, project development based on object defined in synopsis, and regular reviews conducted by the project guide.

ESE – POE (End Semester Examination – Practical/Oral Examination)

ESE-POE will evaluate the final Phase-II project work through a formal presentation, oral examination, and demonstration of the complete project before an evaluation panel.

Course Contents:

Guidelines for Students – Phase–II

A. Complete Implementation

- Complete the full coding of all modules.
- Integrate frontend, backend, database, APIs, models, or IoT hardware as required.
- Confirm compatibility and system stability.

B. Testing & Validation

Students must:

- Prepare test cases and test plans
- Perform unit, integration, and system testing
- Conduct performance evaluation using metrics relevant to the project

C. Optimization

- Tune algorithms or models
- Improve runtime efficiency
- Fix performance bottlenecks
- Refactor code for readability and maintainability

D. Deployment

Where applicable:

- Deploy system on cloud platforms (AWS/Azure/GCP/Firebase)
- Deploy web/mobile applications
- Provide installation guidelines
- Ensure security considerations (authentication, encryption, etc.)

E. Documentation Requirements

Final report must include:

- Abstract
- Introduction
- System design
- Implementation details
- Results & discussion
- Performance metrics
- Screenshots / experimental logs
- Conclusion & future work
- References (IEEE)
- Plagiarism certificate

F. Presentation & Viva

- Prepare a professional PPT summarizing the project.
- Demonstrate the fully working model.
- Explain architecture, algorithms, results, and problem-solving approach.
- Answer technical questions confidently.

G. Submission Requirements

Students must submit:

1. Final report (hard + soft copy)
2. Source code (GitHub link or zipped folder)
3. Dataset (if applicable)
4. User manual
5. PPT presentation
6. Project diary/logbook

Title of the Course: Internship Course Code: UCSIL0872	L	T	P	Credits
	-	-	12	6

Course Pre-Requisite: Software Engineering, Project Management, Mini Project, Knowledge of Project Based Learning (PBL) concepts

Course Description:

In this course, the students will apply the knowledge of SDLC, project management & Project Based Learning to implement the projects in IT industries which will create awareness amongst the students of the recent trends of Computer Science Engineering in industries.

Course Learning Objectives:

1. Apply theoretical knowledge in real-world industrial settings .
2. Gain practical exposure to current industry tools and practices.
3. Develop technical, communication, and professional skills.
4. Understand professional work ethics and teamwork.
5. Document and present internship work effectively.

Course Outcomes:

COs	After Completion of the course, the student should be able to	Bloom's Cognitive	
		Level	Descriptor
CO1	Apply theoretical knowledge in real-world work settings.	L3	Apply
CO2	Explore and assess career options before graduation.	L3	Apply
CO3	Develop communication, interpersonal and other critical skills for job readiness..	L6	Evaluate
CO4	Develop professional work habits and attitudes for job success.	L6	Evaluate

CO-PO Mapping:

CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO1 0	PO1 1	PSO0 1	PSO02
CO1	3	3	3	2						2		1	2
CO2	1		1		1	1	1	2	1	3	2	2	1
CO3	1		1			1	1	3	2	1		1	
CO4			1		1	3	3	3	3	1	2		

Assessment:

One component of In Semester Evaluation (ISE) and One End Semester Examination (ESE).

Assessment	Marks
ISE 1	75
ISE 2	75

In Semester Evaluation 1 (ISE1) (Performance of project work in Industry)

ISE 1 will be assessed by the project guide, focusing on the student's overall performance, learning progress, and professional conduct during the industrial internship.

In Semester Evaluation 2 (ISE2) (Presentation & Demonstration of project work in Industry)

ISE 2 will be assessed by the project guide, focusing on the student's understanding, presentation, and demonstration of the internship work, along with report quality.

Course Contents:

Guidelines for Internship

As per the approved academic structure, students will be allowed to take internships during the 8th semester of B.Tech program. Below are the guidelines/rules and regulations for the students willing to opt for the internship -

1. It is a Mandatory course and the student has to complete an Industrial Internship during the 8th Semester. The internship duration can be between 16 weeks to 20 weeks.
2. The internship should be carried out in any industry with relevant applications of Computer Science and Engineering, preferably under the guidance of a mentor from the same industry.
3. Students who get the internship will be exempted from attendance of lectures and practicals of courses during 8th semester. These students must have to complete all the ISE activities of 8th Semester using LMS and ERP (VMEdulife). It will be mandatory to all students (including internship students) to appear for the Mid Semester Exam (MSE) and End Semester Exams (ESE). Failing to do so, you will not be considered for the award of B. Tech degree. There will not be any extension/exemption from MSE and ESE exams. The rest of the rules and regulations related to academics and exams are all applied as it is.
4. The student will maintain a log of work done on a daily basis and important ideas or practices that he / she has learnt during the internship. The log-book may also be duly signed by the student and the mentor from the industry.

5. The student has to submit a interim report and final detailed report based on the internship immediately after the completion of the internship
6. Students who are not getting any internship have to compulsorily attend all the lectures and practicals of the 8th Semester. They will be governed by the regular academic policies which include - mandatory attendance criteria, failing to meet the attendance criteria students will be detained.
7. Policies which include - mandatory attendance criteria, failing to meet the attendance criteria students will be detained.
8. The department holds the final authority to accept or reject the internship offered to students. The department will check the credibility of the organization offering the internship to students. If the department finds the internship is unworthy, then students will not be allowed to join the organization. Such students are bound to complete the regular academics (including 8th semester lectures and practical).

Students who are not getting any internship have to compulsorily attend all the lectures and practicals of the 8th Semester. They will be governed by the regular academics.

Title of the Course: Co -Curricular Activities-IV Course No.: UCSCC0831	L	T	P	Credits
	---	---	02	01

Course Pre-Requisite: This course is open to all final year engineering students interested in enhancing their personal and professional development through co-curricular activities.

Course Description:

Co-Curricular activities are an integral part of curriculum which provides educational activities to the students and thereby help in broadening their experiences. Co-Curricular activities can be defined as the activities that enhance and enrich the regular curriculum during the normal college hours. All Co-Curricular activities are organized with specific purpose which may according to the nature and form of activities. This course introduces students to a variety of co-curricular activities aimed at enhancing their professional and personal development within the field of engineering and technology. Through practical projects, competitions, workshops, and community engagement, students will develop teamwork, leadership, communication, and technical skills essential for success in their careers.

Course Learning Objectives:

1. To encourage students to showcase their intellectual and independent thinking skills.
2. To imbibe a sense of confidence and managerial capabilities among students.
3. To promote the ability to work in a team, organize and analyse available resources.
4. To build responsiveness among students about the social and cultural responsibilities.

Course Outcomes:

COs	After Completion of the course, the student should be able to	Bloom's Cognitive	
		Level	Descriptor
CO1	Demonstrate the ability to critically analyse information and apply independent judgment in decision-making within the context of the activity	L2	Understand
CO2	Apply principles of management and organizational skills to plan, coordinate, and execute tasks related to the co-curricular activity.	L3	Apply

CO3	Reflect on their roles and responsibilities as members of a diverse community, fostering empathy and inclusivity.	L4	Analyze
CO4	Collaborate effectively with peers to achieve common goals and objectives within the co-curricular activity.	L6	Design

CO-PO Mapping:

CO	**PO 1**	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO1 0**	**PO1 1**	**PSO1**	**PSO2**
CO1	2	3	2	3		1	3				2	2	2
CO2			2						2	2	3		3
CO3						3		3	2				2
CO4									3	2			3

Assessment:

Assessment	**Marks**
ISE	50

ISE (Internal Sessional Evaluation)

Assessment is based on the student's participation in various Co-Curricular Activities and Guidelines given in "Rules for Assigning Activity Points: Activity – Event Grade Point Scheme" Policy Document.

Course Guidelines:

1. Students are entitled to gain academic knowledge in this fast-paced environment, but it is also necessary for them to develop their personalities in both internal and external situations.
2. Co-curricular activities help students grow and develop their personalities. These activities

contribute to a student's total personality development.

3. Not every student is intellectually inclined. Similarly, not all pupils are interested in co-curricular activities. Therefore, there is a need to provide a solid balance of co-curricular and extra-curricular activities in order to achieve the course learning objectives.
4. It primarily refers to intellectual, physical, emotional, and social growth that can be attained by a careful mix of academic, co-curricular, and extra-curricular activities.
5. So, keeping the course learning objectives the "Rules for Assigning Activity Points: Activity – Event Grade Point Scheme" Policy Document is proposed.
6. Student participation is assessed and reflected in the final activity performance report in order to get most students involved in extra-curricular activities (Group A) and co-curricular activities (Group B) as shown in Table 1 in the Policy Document.
7. All undergraduate students must choose at least ONE activity/event from each group i.e. (Group A and B).
8. Students shall choose one activity/ event from Group A and One from Group B that take place on-campus or off-campus.
9. Freedom shall be given to the students to take part in more than one activity under the group.
10. Students are expected to actively participate in activities, participate in contests, and earn grade points.
11. One student in each group must earn up to 50 grades in one semester so that they can achieve up to 100 grades in one year.
12. Grades for each semester are awarded based on the points achieved by the student, as shown in Table 2 in the Policy Document.

Course Structure: ***(Refer Rules for Assigning Activity Points: Activity – Event Grade Point Scheme)***

Sr. No.	Initiatives	Criteria, Activities and Assignments
1	Introduction to Co-Curricular Activities	Orientation, Induction, Course Overview
2	National Initiatives Participation	Participation, Achievement Levels and Assigned Activity Points in NCC, NSS, Unnat Bharat/ Unnat Maharashtra Abhiyan, Ek

		Bharat Shreshtha Bharat (EBSB)
3	Sports and Games Participation	Participation, Achievement Levels and Assigned Activity Points in Sports and Games
4	Cultural Activities Participation	Participation, Achievement Levels and Assigned Activity Points in Music, Performing Arts, Literary Arts
5	Professional Self Initiatives	Participation, Achievement Levels and Assigned Activity Points in 1. Technical Events/Quiz/Paper Contest/Project Contest / Model Making etc. 2. MOOC/ NPTEL/ SWAYAM/ Coursera etc. 3. Competitions/ Events Conducted by Professional Societies (ISTE, IEI, CSI, IEEE, IETE, SAE, ISRO-IIRS, SWE, ISHRAE, ASM, ISNT etc.) 4. Attending Full time Conference/ Seminars/ Exhibitions/ Workshop/ STTP Conducted at IITs/ NITs/ Reputed Institutes/ Universities 5. Attending Full time Conference/ Seminars/ Exhibitions/ Workshop/ STTP Conducted at KITCoEK 6. Paper Presentation in National/ International Conference of High Repute 7. Poster Presentation in National/ International Conference of High Repute 8. Paper Publication in National/ International Journal of High Repute 9. Industrial Training/ Internship (at least for 04 Weeks) 10. Participation in Institute Level Student Clubs 11. Professional Certification (AWS,AZURE,Google Cloud,CISCO,CISC,ITSeC,ECH etc)
6	Entrepreneurship and Innovation	Participation, Achievement Levels and Assigned Activity Points in 1. Prototype Developed and Tested

		2. Awards for Products Developed 3. Innovative Technologies Developed and Used by Industries 4. Got Funding from Government/ Industry for Innovative Ideas 5. Patent-Filed/ Published/ Approved/ Licensed 6. Social Innovations
7	Leadership & Management of Clubs/ Activities	Participation, Achievement Levels and Assigned Activity Points in 1. Elected Student Representative of Student Council (University Representative, General Secretary, Cultural, Sports, NSS Secretary, Ladies Representative, Academic Toppers, Invitee Members) 2. Office Bearer of Professional Society Chapter (ISTE, IEI, CSI, IEEE, IETE, SAE, ISRO-IIRS, SWE, ISHRAE, ASM, ISNT etc.) 3. Office Bearer of Institute Level Student Club (Developer Student Club, Gaganvedhi, Walk With World, Team Mavericks, Cultural Club, Aura, Amateur Write Club, Rotaract Club of KIT Sunshine, Women Development and Gender Equality Cell, Shourya, Lead India etc.) 4. Office Bearer of Departmental Student Association 5. Office Bearer of ECell, Digital Content Lab etc. 6. Student Ambassador for Mayura AICTE IDEA Lab/ NIDHI iTBI etc. 7. Editorial Board Member of Annual Magazine 8. Editorial Board Member of E-Newsletter 9. Member of Governance Committee/ Statutory Committee
8	Culminating Event and Reflection	Final Presentations, Course Reflection, Documentation, Assessment and Evaluation
Participation Levels:		

1. Level: I College Level Events
2. Level: II District/ Central/ Zonal Level Events
3. Level: III State Level Events
4. Level: IV National Level Events
5. Level: V International Level Events

Approval Documents:

1. Certificate
2. Letter from Authorities
3. Appreciation recognition letter
4. Documentary evidence
5. Legal Proof

Grading Scheme:

Grade Range	Grade	Academic Performance
90-100	O	Outstanding
71 - 90	A+	Excellent
68-71	A	Very Good
65-68	B+	Good
60-65	B	Average
55-60	C	Below Average
50-55	D	Marginal
<50	F1	Fail due to Poor Performance