D. Y. PATIL DEEMED TO BE UNIVERSITY
SCHOOL OF ENGINEERING AND MANAGEMENT
Teaching and Evaluation Scheme from Year 2023-24 (as per NEP-2020)
B. Tech. Computer Science Engineering (SEMESTER- V)

Sr. No.	Course Code	Course Type	Course Name	Teaching Scheme				Theory			Practical		Total Marks
				Credits	Contact Hrs			ISE	MSE	ESE	INT	OE/ PoE	
					L	P	T						
1	23CSEU5P01	PCC	Database Engineering	**3**	3	-	-	20	30	50	-	-	100
2	23CSEU5P02	PCC	Database Engineering Lab	**1**	-	2	-	-	-	-	25	25	50
3	23CSEU5P03	PCC	Information Security	**3**	3	-	-	20	30	50	-	-	100
4	23CSEU5P04	PCC	Smart Phone Application Development	**3**	2	2	-	-	-	-	25	25	50
5	23CSEU5P05	PCC	Software Engineering	**2**	2	-	-	20	30	50	-	-	100
6	23CSEU5M06	MDM-III	Test Driven Development	**3**	3	-	-	20	30	50	-	-	100
7	23CSEU5M07	MDM-III	Test Driven Development Lab	**1**	-	2	-	-	-	-	25	-	25
8	23CSEU5E08	PEC-I	Internet of Things	**3**	**3**	**-**	**-**	**20**	**30**	**50**	**-**	**-**	**100**
9	23CSEU5E09		Computer Graphics and Multimedia Techniques										
10	23CSEU5E10		Principles of AI/ML										
11	23CSEU5E11	PEC-I	Internet of Things Lab	**1**	**-**	**2**	**-**	**-**	**-**	**-**	**25**	**-**	**25**
12	23CSEU5E12		Computer Graphics and Multimedia Techniques Lab										
13	23CSEU5E13		Principles of AI/ML Lab										
14	23CSEU5O14	OEC-III	Cloud Computing	**2**	2	-	-	-	-	50	-	-	50
15	23CSEU5D15	AC	Liberal Learning	**-**	2*	-	-	-	-	-	50*	-	
16	23CSEU5D16	AC	Finishing School Training - V	**-**	2*	-	-	-	-	-	50*	-	
			Total	**22**	**18**	**8**	**0**						**700**

HONORS

Sr. No.	Course Code	Course Type	Course Name	Teaching Scheme				Theory			Practical		Total Marks
				Credits	L	P	T	ISE	MSE	ESE	INT	OE/ PoE	
1	23CSEU5Z01	Honors	Introduction to AI/ML	3	3	-	-	20	30	50	-	-	100
2	23CSEU5Z02	Honors	Introduction to AI/ML Lab	1	-	2	-	-	-	-	25	-	25
3	23CSEU5Z03	Honors	Data Security Systems	3	3	-	-	20	30	50	-	-	100
4	23CSEU5Z04	Honors	Data Security Systems Lab	1	-	2	-	-	-	-	25	-	25

Note:
$ - Open & Distance Learning
* - Values are not included in total marks
Min. Marks for Passing: 40% of total marks of individual course

D Y PATIL EDUCATION SOCIETY Deemed to be University Kasba Bawada, Kolhapur School of Engineering & Management KOLHAPUR

D. Y. PATIL DEEMED TO BE UNIVERSITY
SCHOOL OF ENGINEERING AND MANAGEMENT
Teaching and Evaluation Scheme from Year 2023-24 (as per NEP-2020)
B. Tech. Computer Science Engineering (SEMESTER- VI)

Sr. No.	Course Code	Course Type	Course Name	Teaching Scheme				Theory			Practical		Total Marks
				Credits	Contact Hrs			ISE	MSE	ESE	INT	OE/ PoE	
					L	P	T						
1	23CSEU6P01	PCC	System Programming	**3**	3	-	-	20	30	50	-	-	100
2	23CSEU6P02	PCC	Cloud Computing	**3**	3	-	-	20	30	50	-	-	100
3	23CSEU6P03	PCC	Cloud Computing Lab	**1**	-	2	-	-	-	-	25	25	50
4	23CSEU6P04	PCC	Web Technology-II	**3**	2	2	-	-	-	-	50	50	100
6	23CSEU6M05	MDM-IV	Web Testing	**2**	2	-	-	-	-	50	-	-	50
7	23CSEU6E06	PEC-II	Programming Paradigms	**3**	3	-	-	20	30	50	-	-	100
8	23CSEU6E07		Ethical Hacking										
9	23CSEU6E08		Image Processing										
10	23CSEU6E09	PEC-II	Programming Paradigms Lab	**1**	-	2	-	-	-	-	25	-	25
11	23CSEU6E10		Ethical Hacking Lab										
12	23CSEU6E11		Image Processing Lab										
13	23CSEU6E12	PEC-III	FOSS Tools	**3**	3	-	-	20	30	50	-	-	100
14	23CSEU6E13		Blockchain Technology										
15	23CSEU6E14		Augmented Reality/ Virtual Reality										
16	23CSEU6E15	PEC-III	FOSS Tools Lab	**1**	-	2	-	-	-	-	25	-	25
17	23CSEU6E16		Blockchain Technology Lab										
18	23CSEU6E17		Augmented Reality/ Virtual Reality Lab										
19	23CSEU6N18	VSEC	Project Management Tools	**2**	1	2	-	-	-	-	25	25	50
20	23CSEU6D19	AC	Liberal Learning	**-**	2*	-	-	-	-	-	50*	-	
21	23CSEU6D20	AC	Finishing School Training - VI	**-**	2*	-	-	-	-	-	50*	-	
			Total	**22**	**17**	**10**							**700**

HONORS

Sr. No.	Course Code	Course Type	Course Name	Teaching Scheme				Theory			Practical		Total Marks
				Credits	L	P	T	ISE	MSE	ESE	INT	OE/ PoE	
1	23CSEU6Z01	Honors	Artificial Neural Networks	3	3	-	-	20	30	50	-	-	100
2	23CSEU6Z02	Honors	Artificial Neural Networks Lab	1	-	2	-	-	-	-	25	-	25
3	23CSEU6Z03	Honors	Ethical Hacking	3	3	-	-	20	30	50	-	-	100
4	23CSEU6Z04	Honors	Ethical Hacking Lab	1	-	2	-	-	-	-	25	-	25

Note:
$ - Open & Distance Learning
* - Values are not included in total marks
Min. Marks for Passing: 40% of total marks of individual course

Semester V
Course Contents

Course Code:	23CSEU5P01
Course Name:	Database Engineering

L	T	P	Credit
3	0	0	3

Course Prerequsites:

Set Theory, Operating System, Data Structures, Basic Software Engineering Concept (SDLC)

Course Description:

The Database Engineering course provides a comprehensive understanding of database systems and their role in the design, development, and management of information systems. It introduces students to database theory, architecture, design methodologies, query languages, and data modeling techniques.

Course Outcomes: After the completion of the course the student will be able to -

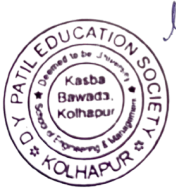
CO1	Understand fundamentals of Database Management Systems
CO2	Analyze the problem & construct good database design
CO3	Apply SQL queries to design & manage the database
CO4	Understand Transactions Model and the Recovery Schemes in Database Management Systems

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	1	2		1	3	1	1			1		2	2	1
CO2	2	2	2	2	2	3	2	3	2	3	3	2	3	3
CO3	2	2	2	2	2	3		1	1	1	1	1	3	3
CO4	1	3	2	3	3	3	1	1		1	2	1	3	3

Assessment Scheme:

SN	Assessment	Weightage	Remark
1	In Semester Evaluation 1 (ISE1)	10%	Assignment, Test, Quiz, Seminar, Presentation, etc.
2	Mid Semester Examination (MSE)	30%	50% of course contents
3	In Semester Evaluation 2 (ISE2)	10%	Assignment, Test, Quiz, Seminar, Presentation, etc.
4	End Semester Examination (ESE)	50%	100% course contents



Course Contents:

Unit 1	**INTRODUCTION TO DATABASES**	**6 Hours**

Database System Applications, Purpose of Database Systems, View of Data, Database Languages, Specialty Databases, Database Users & Administrators, Structure of Relational Databases, Database Schema, Keys, Relational Query Languages, Relational Operations.

Unit 2	**E-R MODEL AND DATABASE DESIGN**	**8 Hours**

E-R Model: The Entity-Relationship Model, Constraints, Entity-Relationship Diagrams, Reduction to Relational Schemas
Normalization: Data Redundancies & Update Anomalies, Functional Dependencies, The Process of Normalization, First Normal Form, Second Normal Form, Third Normal Form, Boyce-Codd Normal Form.

Unit 3	**STRUCTURED QUERY LANGUAGE (SQL**	**7 Hours**

Overview of the SQL Query Language, SQL Data Definition, Basic Structure of SQL Queries, Additional Basic Operations, Set Operations, Aggregate Functions, Nested sub Queries, Modification of Databases.

Unit 4	**DATA STORAGE & INDEXING**	**7 Hours**

File Organization, Organization of records in File, Data Dictionary Storage, Database Buffer, Basic Concepts indexing & hashing, Ordered Indices, B+ Tree Index files, Multiple-Key Access, Static Hashing.

Unit 5	**TRANSACTION MANAGEMENT**	**7 Hours**

Transaction Concept, A Simple Transaction Model, Transaction Atomicity and Durability, Transaction Isolation, Serializability, Lock-Based Protocols, Deadlock Handling, Timestamp-Based Protocols, Validation-Based Protocols

Unit 6	**RECOVERY SYSTEM**	**6 Hours**

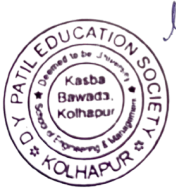
Failure Classification, Storage, Recovery and Atomicity, Recovery Algorithm, Failure with Loss of Nonvolatile Storage, Remote Backup Systems

Text Books:

1. Database System Concepts, A. Silberschatz, H.F. Korth, S. Sudarshan, 6th Edition, Mc Graw Hill Education.
2. Database Systems - A practical approach to Design, Implementation and Management Thomos Connolly, Carolyn Begg, 3rd Edition, Pearson Education

Reference Books:

1.Database Systems – Design, Implementation and Management, Rob & Coronel 5th Edition, Thomson Course Technology
2.Fundamentals of Database Systems, Ramez Elmasri, Shamkant B. Navathe, 4th Edition,Pearson Education



Course Code:	23CSEU5P02
Course Name:	Database Engineering Lab

L	T	P	Credit
0	0	2	1

Course Prerequsites:

Set Theory, Fundamental of Software Engineering (SDLC)

Course Description:

The Database Engineering course provides a comprehensive understanding of database systems and their role in the design, development, and management of information systems. It introduces students to database theory, architecture, design methodologies, query languages, and data modeling techniques.

Course Outcomes: After the completion of the course the student will be able to -

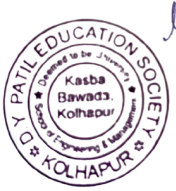
CO1	Understand fundamentals of database management systems
CO2	Analyze & construct good database design
CO3	Apply SQL queries to design & manage the database

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	1	1	1		3	1	1			1	1	1		
CO2	1	3	2	3	2	3	1	3	1	1	3	2		
CO3	2	3	2	2	3	3	1	3	1	1	3	2		
CO4														

Assessment Scheme:

SN	Assessment	Weightage	Remark
1	Internal	50%	Assignments, Seminar, Presenatation etc.
2	POE	50%	POE



Course Contents:	
Assessment No. 1 : Draw an E-R Diagram of any organization	
Assessment No. 2 : Reduce above mentioned E-R Diagram into Relational Model	
Assessment No. 3 : Normalize any database from first normal form to Boyce-Codd Normal Form (BCNF)	
Assessment No. 4 :Use DDL Queries to create, alter (add, modify, rename, drop) & drop Tables	
Assessment No. 5 : Use DML Queries to insert, delete, update & display records of the tables	
Assessment No. 6 : Create table with integrity constraints like primary key, check, not null and unique	
Assessment No. 7 : Create table with referential integrity constraints with foreign key, on delete cascade and on delete set null	
Assessment No. 8 : Display the results of set operations like union, intersections & set difference	
Assessment No. 9 : Display the results of Join Operations like cross join, self join, inner join, natural join, left outer join, right outer join and full outer join	
Assessment No. 10 : Display the records using Aggregate functions like min, max, avg, sum & count. Also use group by, having clauses	
Assessment No. 11 : Display the results using String operations	
Assessment No. 12 : Create & Update views for any created table	
Assessment No. 13 : Study of B+ tree indexing	
Assessment No. 14 : Implement static hashing (Simulation)	
Text Book:	
Williams Stallings – Cryptography and Network Security Principles and Practices (Unit 1 to 5) Pearson Education (LPE), 7th Edition	

Course Code:	23CSEU5P03
Course Name:	Information Security

L	T	P	Credit
3			3

Course Prerequsites:

Computer Network, Data Communication, Engg. Mathematics

Course Description:

This course gives you practical survey of both the principles and practice of cryptography and network security. In the first part of course, the basic issues to be addressed by a network security capability are explored by providing a tutorial and survey of cryptography and network security technology. The later part of course deals with the practice of network security.

Course Outcomes: After the completion of the course the student will be able to -

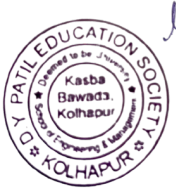
CO1	Explain the use of Cryptographic algorithms to ensure data protection and integrity.
CO2	Apply the knowledge of cryptographic techniques to solve the problems on security.
CO3	Illustrate the different Network and Internet security protocols in TCP/IP stack.
CO4	Analyze the security facilities designed to provide system security.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	2					1						2	2	2
CO2	2	3	3		2	2		2				2	2	2
CO3	1				2	2						2	2	2
CO4		2	2		3	3		2				2	2	2

Assessment Scheme:

SN	Assessment	Weightage	Remark
1	In Semester Evaluation 1 (ISE1)	10%	Assignment, Test, Quiz, Seminar, Presentation, etc.
2	Mid Semester Examination (MSE)	30%	50% of course contents
3	In Semester Evaluation 2 (ISE2)	10%	Assignment, Test, Quiz, Seminar, Presentation, etc.
4	End Semester Examination (ESE)	50%	100% course contents



Course Contents:

Unit 1	**Introduction to Information Security**	**5 Hours**

Overview:
Computer Security Concepts, The OSI Security Architecture, Security Attacks, Security Services, Security Mechanisms, A Model for Network Security
Classical Encryption Techniques:
Symmetric Cipher Model,Substitution Techniques, Transposition Techniques, Rotor machines, Steganography.
Case Study 1.1: Perform Encryption and Decryption using crypt tool.

Unit 2	**Symmetric and Asymmetric Key Cryptography**	**8 Hours**

Block Ciphers and the Data Encryption Standard:
Block Cipher Structure, Data Encryption Standard (DES), A DES Example, Strength of DES, Block Cipher Design Principles, AES Structure, Multiple Encryption and Triple-DES
Public Key Cryptography:
Principles of Public-Key Cryptosystems, RSA Algorithm, Other Public key Cryptosystems - Diffie-Hellman Key Exchange, ElGamal Cryptographic system

Unit 3	**Cryptographic Authentication Functions**	**8 Hours**

Cryptographic Hash Functions:
Applications of Cryptographic Hash Functions, Two Simple Hash Functions, Requirements and Security, Hash Functions Based on Cipher Block Chaining, Secure Hash Algorithm (SHA)
Message Authentication Codes:
Message Authentication Requirements, Message Authentication Functions, Requirements for MAC and Security of MACs, MACs Based on Hash Functions: MAC, MACs Based on Block Ciphers: DAA and CMAC
Digital Signatures:
Digital Signatures, ElGamal Digital Signature Scheme, Schnorr Digital Signature Scheme, Digital Signature Standard (DSS)
Case Study 3.1: Working of Digital signature software tool Sign server

Unit 4	**Key Management and User Authentication**	**8 Hours**

Key management:
Symmetric Key Distribution Using Symmetric Encryption,Symmetric Key Distribution Using Asymmetric Encryption, Distribution of Public Keys, X.509 Certificates, Public Key Infrastructure
User Authentication Protocol:
Remote User-Authentication Principles, Remote User-Authentication UsingSymmetric Encryption, Kerberos, Remote User Authentication Using Asymmetric Encryption.

Unit 5	**Internet security Protocols**	**6 Hours**

Transport-Level Security:
Web Security Issues, Secure Sockets Layer (SSL), Transport Layer Security (TLS), HTTPS, SSH
Electronic Mail Security:
Pretty Good Privacy (PGP), S/MIME, SET
IP Security:
IP Security Overview, IP Security Policy, Encapsulating Security Payload
Case Study 5.1: Perform surveillance through packet sniffer tool like Wireshark &TCP Dump.

Unit 6	**Firewall and Intrusion detection system**	**8 Hours**

Firewalls: Introduction, Types of firewall, Firewall configuration, VPN, Types of VPN **IDS:** Overview of IDS, IDS Components, Approaches of IDS **SIEM:** Introduction to SIEM,SIEM Scenario and process flow, SIEM architecture, SIEM features Case study 6.1: Run Online Scanners like Virus Total, Jotti and No VirusThanks

Reference Books:
Textbooks: 1. Williams Stallings – Cryptography and Network Security Principles and Practices (Unit 1 to 5) Pearson Education (LPE), 7th Edition 2. Network Security, Firewalls, and VPNs, 3rd Edition by J. Michael Stewart, Denise Kinsey (Unit 6)
References: 1. Cryptography & Network Security B.A. Forouzan McGrawHill 2. Cryptography and network security – Atul Kahate (TMGH) 3. Handbook of Applied Cryptography - Menezes, an Oorschot, and S.A. Vanstone

Course Code:	23CSEU5P04
Course Name:	Smart Phone Application Development

L	T	P	Credit
2	0	2	3

Course Prerequsites:

Basic programming knowledge

Course Description:

This course provides a comprehensive introduction to Kotlin for Android development, covering language basics, and advanced features. Students will explores user interface design with layouts and views, navigation using intents, dialogs, menus, and various Android storage options, including SQLite databases and content providers.

Course Outcomes: After the completion of the course the student will be able to -

CO1	Learn the basics of Kotlin and how to use it for Android apps
CO2	Understand how the Android system works and the main building blocks of apps
CO3	Get hands-on experience in designing how apps look and how users move around in them

CO-PO Mapping:

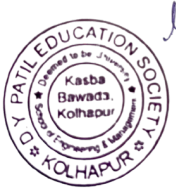
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	1								1	1		3	1	
CO2	1	1	2		3			2	1	2			3	2
CO3	1	1	2		3			2	1	1		3	3	2
CO4					3				1				2	

Assessment Scheme:

SN	Assessment	Weightage	Remark
1	Internal Assessment	50%	Assignment, Test, Quiz, Seminar, Presentation, etc.
2	POE	50%	Practical/Oral Examination

Course Contents:		
Unit 1	**Introduction to Kotlin Programming Language**	**6 Hours**
"Kotlin Basics: Kotlin Advantages, How Kotlin Program Work, Creating, Compiling and running Kotlin program, Input and Output, Kotlin Variables, Kotlin Data Types, operators, Type checks and casts, arrays, Control Flow statements, functions, Null safety. Object-Oriented Programming (OOP) in Kotlin: Object and Classes, constructors, overloading, Abstract Class, Interface, Sealed Class, Generic Class, Enum Class, Inner and anonymous Inner class. Non-Blocking Programming Techniques in Kotlin: Threading, Callbacks, Futures, promises, and others, Reactive Extensions, Coroutines. Advanced Kotlin Concepts: Kotlin collections, exception handling, Packages and Imports. Introduction to Kotlin Multiplatform"		
Unit 2	**Introduction to Android Application**	**5 Hours**
Background: Evolution of Mobile Operating Systems, History of Android versions, Android OS architecture: Android OS Stack, Linux kernel, Native Libraries/DVM, Application Framework, Applications. Android Application Components: Activity, Fragments, Intents, BroadcastReceivers, Content Providers, Services Activities: Activity lifecycle, Activity Back Stack Fragments: Definition and purpose of Fragments, Fragment Lifecycle and its relationship with Activity lifecycle.		
Unit 3	**Creating Android Applications**	**5 Hours**
Introduction to Android SDK Components of android SDK (sdk tools) Compilation, Building and running of android application Creating a Android Project: Project Directory Structure, Logging in Android (Logcat), Managing Exception with Logcat, Android Manifest File, Android Resources and Resource Directories, Permissions."		
Unit 4	**Layouts**	**5 Hours**
"View Hierarchy: Android View and View Group classes and XML Elements, Layouts: Linear Layouts, Relative Layout, Table Layout, Frame Layout, Constraint Layouts, and their important properties. Padding and Margins with Layouts. Basic Views their properties and Events: TextView, Buttons & types, ImageView, EditText, CheckBox, ImageView, VideoView, GridView, RatingBar etc. Advanced Views: ListView, RecyclerView, Card View Intents: Implicit & Explicit Intents, Component of Intent, examples of some standard intents (Telephony, SMS etc), Android Intent Messaging via Intent Objects, Using Intents with Activities, pending Intents, Broadcast Receivers"		
Unit 5	**Dialogs and Menu**	**5 Hours**
Dialogs: Components of Dialog, Alert Dialog, Seek Bar, Date Picker Dialog, Time Picker Dialog, Custom Dialogs Menus: Menu Inflators, Context Menu, Options Menu, Handling menu click events.		
Unit 6	**Android Storage**	**5 Hours**
Android Storage Options: Shared Preferences, Internal Storage, External Storage SQLite Databases: SQLite Database in your application, Installing SQLite plugin, DbHelper, The Database Schema and Its Creation, Four Major Operations of SQLite. SQLite databases and Content Providers: Creating an SQLite Database, querying an SQLite DB Table		

Text Books:
1. Android Programming with Kotlin for Beginners" by John Horton 2. Programming Android with Kotlin by Pierre-Olivier Laurence, Amanda Hinchman-Dominguez, Mike Dunn, G. Blake Meike, O'Reilly Media, Inc. 3. Learn Kotlin for Android Development by Peter Späth , APress; 1st ed. edition



Course Code:	23CSEU5P05		L	T	P	Credit
Course Name:	Software Engineering		2			2

Course Prerequsites:

Problem Solving Using C

Course Description:

This course gives you fundamentals of software development in the current IT industry. The fundamentals are divided into different parts. The first part deals with different software models followed for development of software. The subsequent parts deals with requirement specification, software design with UML, coding and testing respectively. You will get complete insight of software development process which will help you a lot in your career in IT industry

Course Outcomes:	After the completion of the course the student will be able to -
CO1	Summarize the basic processes of software development and various SDLC models.
CO2	Analyze software requirements analysis and formulate design solution for a software.
CO3	Apply new software design techniques and technologies to bring out innovative solutions for the socital problems evolving into their continuous professional development.
CO4	Use knowledge of software testing approaches for verification and validation.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	1										2			
CO2		1	2		1				3	3	1		1	2
CO3		1	1		2			1	3	3	2	3	3	2
CO4	1	1	1		2			1	3	2		3	3	1

Assessment Scheme:

SN	Assessment	Weightage	Remark
1	End Semester Examination (ESE) (50 Marks)	100%	100% course contents

Course Contents:		
Unit 1	**Software and Software Process**	**5 Hours**
The Problem Domain, SE Challenges, SE Approaches, Software Process, Desired Characteristics of a Software Process, Software Development Process Models- Waterfall Model, Prototype Methodology, Agile Software Development Methodology, Rapid Application Development (RAD),Dynamic Systems Development Model Methodology, Spiral Model, Extreme Programming Methodology, Feature Driven Development.		
Unit 2	**Software Requirement Analysis and Specifications**	**4 Hours**
Software Requirements, Problem Analysis, Requirements Specification, Functional Specifications with use cases, Validation, Metrics		
Unit 3	**Software Design Approaches**	**6 Hours**
Design Principles, Module-Level Concepts, Design Notation and Specification, Structured Design Methodology, OO Analysis and OO Design, OO Concepts, Design Concepts		
Unit 4	**UML Structural Modeling**	**4 Hours**
Classes,Relationship,Common Mechanics, Diagrams and Class Diagrams,Advanced Classes, Advanced Relationships, Interfaces, Types, and Roles, Packages, Instances and Object Diagram		
Unit 5	**UML Behavioral and Architectural Modeling**	**5 Hours**
Behavioral: Interactions, Use Cases, Use Case Diagrams, Interaction Diagrams, Activity Diagrams Architectural: Components, Deployment, Collaborations, Patterns and Frameworks, Component Diagrams, Deployment Diagrams		
Unit 6	**Coding and Testing**	**4 Hours**
Programming Principles and Guidelines, Coding Process, Refactoring, Verification, Metrics, Testing Fundamentals, Black-Box Testing, White-Box Testing.		

Text Books:
1. An Integrated approach to Software Engineering' –Pankaj Jalote, 3rd Edition, Narosa Publication. (1,2,3,6) 2. UML User Guide- Grady Booch, James Rumbaugh, Publisher: Addison Wesley (4,5)
Reference Books:
1. Software Engineering- A Practitioner's Approach – Roger S. Pressman (TMH) , ISBN-13: 978- 0071267823 ISBN-10: 0071267824 2. Software Engineering- Ian Sommerville – Pearson, 10th Edition, ISBN-13: 9780137503148 3. Software Engineering, Kogent Learning Solutions Inc., Dreamtech Press India Pvt. Ltd, ISBN: 9789350042663, 9789350042663

Course Code:	23CSEU5E08
Course Name:	Internet of Things

L	T	P	Credit
3	0	0	3

Course Prerequsites:

"Computer Networks and Internet fundamentals

Programming skills in C/C++ or Python

Basic understanding of sensors and electronic components"

Course Description:

This course introduces the fundamentals of the Internet of Things (IoT) and its integration with cloud computing. It covers IoT architecture, sensors, microcontrollers (Arduino, ESP32, Raspberry Pi), communication protocols, and cloud platforms such as AWS. Students will learn to interface devices, collect data, and use cloud services for storage and analytics. The course also explores real-world IoT applications in various domains and addresses challenges related to security, privacy, and ethics.

Course Outcomes: After the completion of the course the student will be able to -

CO1	Understand the fundamental concepts, architecture, and enabling technologies of the Internet of Things (IoT).
CO2	Demonstrate the ability to interface sensors and actuators with microcontrollers and implement basic IoT
CO3	Analyze the use of wireless communication protocols and cloud services in designing scalable IoT solutions.
CO4	Apply data handling and analytics techniques to IoT applications and examine real-world use cases and etl

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	2	2				2							3	1
CO2	3	1	2	1	1	1					2		3	2
CO3	2	3		2	2	2							3	3
CO4														

Assessment Scheme:

SN	Assessment	Weightage	Remark
1	In Semester Evaluation 1 (ISE1)	10%	Assignment, Test, Quiz, Seminar, Presentation, etc.
2	Mid Semester Examination (MSE)	30%	50% of course contents
3	In Semester Evaluation 2 (ISE2)	10%	Assignment, Test, Quiz, Seminar, Presentation, etc.
4	End Semester Examination (ESE)	50%	100% course contents

D Y PATIL EDUCATION SOCIETY KOLHAPUR Kasba Bawada, Kolhapur

Course Contents:

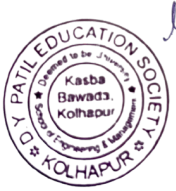
Unit 1	**Fundamentals of IoT**	**6 Hours**
Introduction, Definitions & Characteristics of IoT, IoT Architectures, Physical & Logical Design of IoT, Enabling Technologies in IoT, History of IoT, About Things in IoT, The Identifiers in IoT, About the Internet in IoT, IoT frameworks, IoT and M2M		
Unit 2	**IoT Physical Devices and Endpoints**	**8 Hours**
Microcontrollers, Introduction to Arduino board, various boards of Arduino. Arduino Uno : Arduino Uno Pin Layout, Arduino IDE, Arduino programming, ESP32 : ESP32 pin layout, advantages of ESP32 board, Interfacing sensors with microcontroller Raspberry-Pi : Introduction to Raspberry-Pi, installation of raspberry-pi, raspberry pi configuration,Introduction to Python, Interfacing sensors with raspberry pi.		
Unit 3	**Sensors and Protocol**	**7 Hours**
Sensors : Light sensor, temperature sensor with thermistor, voltage sensor, ADC and DAC, Gas sensors, Temperature and Humidity Sensor , Motion Detection Sensors, Wireless Bluetooth Sensors,Level Sensors, Embedded Sensors, Distance Measurement with ultrasound sensor, Biometric, Load, Flow, and pressure sensor Accuators, Actuators, Actuator Types, Actuator Characteristics		
Unit 4	**IoT Physical Servers and Cloud Offerings**	**7 Hours**
Introduction to Cloud Storage models and communication APIs Web Server – Web server for IoT, Cloud for IoT, AWS services for IoT		
Unit 5	**Data Handling& Analytics**	**7 Hours**
Introduction, Bigdata, Types of data, Characteristics of Big data, Data handling Technologies, Flow of data, Data acquisition, Data Storage, Introduction to Hadoop. Introduction to data Analytics, Types of Data analytics, Local Analytics, Cloud analytics and applications		
Unit 6	**Applications of IoT**	**6 Hours**
Home Automation, Smart Cities, Energy, Retail Management, Logistics, Agriculture, Health and Lifestyle, Industrial IoT, Legal challenges, IoT design Ethics, IoT in Environmental Protection, Security and challenges in IoT		

Text Books:

1. Peter Waher, 'Learning Internet of Things', Packt Publishing, 2015 3. Editors Ovidiu Vermesan
2. Peter Friess,'Internet of Things – From Research and Innovation to Market Deployment', River Publishers, 2014

Reference Books:

1. Daniel Minoli, ― "Building the Internet of Things with IPv6 and MIPv6: The Evolving World of M2M Communications", ISBN: 978-1-118-47347-4, Willy Publications
2. Pethuru Raj and Anupama C. Raman, "The Internet of Things: Enabling Technologies, Platforms, and Use Cases", CRC Press
2. Internet of Things (A Hands-on Approach)" by Arshdeep Bahga and Vijay Madisetti



Course Code:	23CSEU5E09
Course Name:	Computer Graphics and Multimedia Techniques

L	T	P	Credit
3	0	0	3

Course Prerequsites:

1. Basic programming knowledge, Linear Algebra

Course Description:

This course provides a comprehensive introduction to the principles and applications of computer graphics and multimedia technologies. Students will learn about graphics systems, algorithms for rendering 2D and 3D graphics, animation techniques, and multimedia development.

Course Outcomes: After the completion of the course the student will be able to -

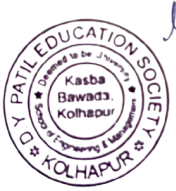
CO1	Understand fundamental computer graphics concepts and algorithms
CO2	Develop multimedia applications integrating various media types
CO3	Create basic animations and interactive graphics
CO4	Apply graphics and multimedia techniques to solve real-world problems

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	2	2				1							1	1
CO2	1	2	3	1	3	2				1	2		3	3
CO3	1	2	3	1	3	2				1	2		3	3
CO4	1	3	3	1	3	3				1	2		3	3

Assessment Scheme:

SN	Assessment	Weightage	Remark
1	In Semester Evaluation 1 (ISE1)	10%	Assignment, Test, Quiz, Seminar, Presentation, etc.
2	Mid Semester Examination (MSE)	30%	50% of course contents
3	In Semester Evaluation 2 (ISE2)	10%	Assignment, Test, Quiz, Seminar, Presentation, etc.
4	End Semester Examination (ESE)	50%	100% course contents



Course Contents:

Unit 1	**An Introduction Graphics System**	**4 Hours**
Computer Graphics and Its Types, Application of computer graphics, Graphics Systems : Video Display Devices, Raster Scan Systems, Random Scan Systems, Graphics Monitors and Work Stations, Input Devices, Hard Copy Devices, Graphics Software.		
Unit 2	**Geometric Transformations**	**10 Hours**
Basic Transformations, Matrix Representation and Homogeneous Coordinates, Composite Transformations, Reflection and Shearing. Two-Dimension Viewing : The viewing Pipeline, Window to view port coordinate transformation, Clipping Operations, Point Clipping, Line Clipping, Polygon Clipping, Text Clipping, Exterior Clipping Three-Dimensional Concepts : Three Dimensional Display Methods, 3D Transformations, Parallel Proection and Perspective Projection		
Unit 3	**Raster Scan Graphics**	**7 Hours**
Line Drawing Algorithms, Circle Generating Algorithms, Scan-Line Polygon Fill Algorithm, Boundary-Fill Algorithm, Flood Fill Algorithm		
Unit 4	**ILLUMINATION AND COLOR MODELS**	**7 Hours**
Light sources, Basic illumination models, Displaying light intensities, Halftone patterns and Dithering Techniques, Polygon Rendering methods, Ray tracing methods		
Unit 5	**MULTIMEDIA SYSTEM DESIGN & MULTIMEDIA FILE HANDLING**	**7 Hours**
Multimedia basics – Multimedia applications – Multimedia system architecture – Evolving technologies for multimedia – Defining objects for multimedia systems – Multimedia data interface standards – Multimedia databases. Compression and decompression – Data and file format standards – Multimedia I/O technologies – Digital voice and audio – Video image and animation – Full motion video – Storage and retrieval technologies.		
Unit 6	**Animation**	**6 Hours**
Basics of Animation: Definition , Traditional Animation Techniques, Frame based Animation Techniques, Tweeking , Morphing, Computer Animation Tools : Hardware , Software ,Applications for Computer Animation		

Text Books:

1. Donald Hearn & M. Pauline Baker, “Computer Graphics with OpenGL”, Third Edition, 2004, Pearson Education, Inc. New Delhi.
2. Ze-NianLi and Mark S. Drew, “Fundamentals of Multimedia”, First Edition, 2004, PHI Learning Pvt. Ltd., New Delhi.
3. DP Mukherjee, Fundamentals of Computer Graphics and Multimedia, PHI

Reference Books:

1. Andleigh, P. K and KiranThakrar, ―Multimedia Systems and Designl, PHI, 2003

Course Code:	23CSEU5E10
Course Name:	Principles of AI/ML

L	T	P	Credit
3	0	0	3

Course Prerequsites:

Probability and Statistics
Programming skills in C/C++ or Python
Data Structures and Algorithms

Course Description:

This course offers a comprehensive introduction to Artificial Intelligence, covering fundamental concepts such as knowledge representation, search strategies, logic-based reasoning, and learning algorithms. It explores classical and heuristic search methods, propositional and predicate logic, and key machine learning techniques including decision trees, neural networks, and reinforcement learning. The course also introduces probabilistic reasoning using Bayesian networks, equipping students with the theoretical and practical tools to design intelligent systems and solve complex real-world problems.

Course Outcomes: After the completion of the course the student will be able to -

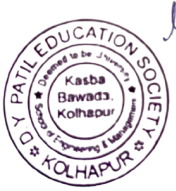
CO1	Explain the main approaches to Artificial intelligence, Machine learning along with the limitations and challenges associated with AI and ML.
CO2	Apply AI concepts to solve real-world problems
CO3	Analyse and interpret data using AI techniques.
CO4	

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	2	2				2							3	1
CO2	3	1	2	1	1	1					2		3	2
CO3	2	3		2	2	2							3	3
CO4														

Assessment Scheme:

SN	Assessment	Weightage	Remark
1	In Semester Evaluation 1 (ISE1)	10%	Assignment, Test, Quiz, Seminar, Presentation, etc.
2	Mid Semester Examination (MSE)	30%	50% of course contents
3	In Semester Evaluation 2 (ISE2)	10%	Assignment, Test, Quiz, Seminar, Presentation, etc.
4	End Semester Examination (ESE)	50%	100% course contents



Course Contents:

Unit 1	Introduction to Artificial Intelligence	6 Hours

Introduction, what is AI, Strong Methods and weak Methods. Uses and Limitations,
Knowledge Representation: Need for good representation, semantic nets, Frames, , Search Spaces, Semantics Tress, Search Trees, Combinatorial Explosion, Problem reduction, Goal Trees, Combinatorial Explosion.

Unit 2	Search Methodologies	8 Hours

Introduction, Problem solving as search, Data driven or goal driven search, Generate and test, Properties of search methods, Depth First Iterative Deepening, Using Heuristics for Search, Hill Climbing, Best-First Search, Identifying Optimal Paths, Constraint Satisfaction search, Forward Checking, Local Search and Meta heuristics, Simulated Annealing. Genetic Algorithms for search, Real time A*, Bidirectional search, Nondeterministic search, non-chronological backtracking.

Unit 3	Prepositional and Predicate Logic	7 Hours

Introduction, what is Logic, Why Logic is used in Artificial Intelligence, Logical Operators, translating between English and Logic Notation, The deduction Theorem, Soundness, Completeness, Decidability, Monotonicity, Abduction and Inductive reasoning, Modal logics and possible worlds, Dealing with change.

Unit 4	Introduction to Machine Learning	7 Hours

Introduction, Training Rote Learning, Learning Concepts, General-to Specific Ordering, Version Spaces, Candidate Elimination, Inductive Bias, Decision-Tree Induction, The Problem of Overfitting, The Nearest Neighbor Algorithm, Backpropagation algorithms, Reinforcement Learning

Unit 5	Neural Networks:	7 Hours

Introduction, Neurons, Perceptrons, Multilayer Neural Networks, Recurrent Networks,
Unsupervised Learning Networks, Evolving Neural Networks.

Unit 6	Probabilistic Reasoning and Bayesian Belief Networks	6 Hours

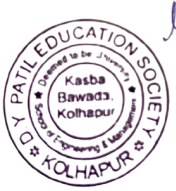
Introduction, Probabilistic Reasoning, Joint Probability Distributions, Bayes' Theorem, Simple Bayesian Concept Learning, Bayesian Belief Networks, The Noisy-V Function, Bayes' Optimal Classifier, The Naïve Bayes Classifier.

Text Books:

1. Ben Coppin, Artificial Intelligence Illuminated, Jones and Bartlett, 2004.
2. Tom M. Mitchell, "Machine Learning", Mcgraw-Hill Education (Indian Edition), 2013.

Reference Books:

1. Elaine Rich Kevin Knight, Shivashankar B Nair: Artificial Intelligence, Tata McGraw Hill 3rd edition 2013.
2. Stuart Russel, Peter Norvig: Artiificial Intelligence A Modern Approach, Pearson 3rd edition 2013.
3. Ethem Alpaydin, "Introduction to Machine Learning", 2nd Edition, PHI Learning Pvt. Ltd., 2013.
4. T Hastie, R. Tibshirani, J.H.Fiedman, "The Elements of statistical learning", Springer, 1st Edition 2001.



Course Code:	23CSEU5E11
Course Name:	Internet of Things Lab

L	T	P	Credit
0	0	2	1

Course Prerequsites:

"Computer Networks and Internet fundamentals

Programming skills in C/C++ or Python

Basic understanding of sensors and electronic components"

Course Description:

This course introduces the fundamentals of the Internet of Things (IoT) and its integration with cloud computing. It covers IoT architecture, sensors, microcontrollers (Arduino, ESP32, Raspberry Pi), communication protocols, and cloud platforms such as AWS. Students will learn to interface devices, collect data, and use cloud services for storage and analytics. The course also explores real-world IoT applications in various domains and addresses challenges related to security, privacy, and ethics.

Course Outcomes: After the completion of the course the student will be able to -

CO1	Configure and program microcontrollers like Arduino, ESP32, and Raspberry Pi by interfacing various sensors and actuators.
CO2	Connect IoT devices to cloud platforms and visualize sensor data.
CO3	Implement IoT communication using protocols like MQTT and HTTP.
CO4	Develop a mini-project applying IoT and cloud integration in a real-world use case.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	2	2				2							3	1
CO2	3	1	2	1	1	1					2		3	2
CO3	2	3		2	2	2							3	3
CO4														

Assessment Scheme:

SN	Assessment	Weightage	Remark
1	Internal	100%	Assignment, Test, Quiz, Seminar, Presentation, etc.

List of Experiments

1Study and setup of Arduino/ESP32 boards and basic programming using IDE
2Interfacing LED and Push Button with Arduino/ESP32
3Interfacing Temperature and Humidity sensor (e.g., DHT11) and displaying data
4Interfacing Ultrasonic sensor for distance measurement
5Interfacing Gas sensor or Light sensor with ESP32/Arduino and displaying output
6Interfacing with Actuators – DC Motor/Relay using Arduino/ESP32
7Introduction to Raspberry Pi: OS Installation, Python Programming Basics
8Sensor interfacing with Raspberry Pi and sending data over local network
9Implementing communication between two IoT devices using MQTT protocol
10Sending sensor data to cloud using ThingSpeak or Blynk platform
11Using AWS IoT Core for device connection and real-time data visualization
12Mini-project: Design and develop a complete IoT solution using microcontroller, sensors, cloud connectivity, and a simple dashboard for data monitoring

Text Books:

1. Peter Waher, 'Learning Internet of Things', Packt Publishing, 2015 3. Editors Ovidiu Vermesan
2. Peter Friess,'Internet of Things – From Research and Innovation to Market Deployment', River Publishers, 2014
1. Hakima Chaouchi, — "The Internet of Things Connecting Objects to the Web" ISBN : 978-1- 84821-140-7, Wiley Publications
2. Olivier Hersent, David Boswarthick, and Omar Elloumi, — "The Internet of Things: Key Applications and Protocols", WileyPublications
3. Vijay Madisetti and ArshdeepBahga, — "Internet of Things (A Hands-on-Approach)",

Reference Books:

1. Daniel Minoli, — "Building the Internet of Things with IPv6 and MIPv6: The Evolving World of M2M Communications", ISBN: 978-1-118-47347-4, Willy Publications
2. Pethuru Raj and Anupama C. Raman, "The Internet of Things: Enabling Technologies, Platforms, and Use Cases", CRC Press
2. Internet of Things (A Hands-on Approach)" by Arshdeep Bahga and Vijay Madisetti
3. Introduction to IoT, Sudip Misra, Anandarup Mukherjee, Arjit Roy, CAMBRIDGE UNIVERSITY, PRESS.
4. Internet of Things Hanads on Approach, Ashdreep Bhaga, Vijay Midishetti, Universities Press

Course Code:	23CSEU5E12
Course Name:	Computer Graphics and Multimedia Techniques Lab

L	T	P	Credit
0	0	2	1

Course Prerequsites:

1. Basic programming knowledge, Linear Algebra

Course Description:

This laboratory course provides hands-on experience in implementing computer graphics algorithms and developing multimedia applications. Students will work with graphics programming, implement fundamental algorithms, create animations, and develop interactive multimedia systems.

Course Outcomes: After the completion of the course the student will be able to -

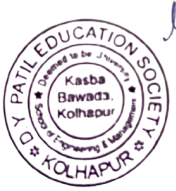
CO1	Implement fundamental computer graphics concepts and algorithms
CO2	Develop multimedia applications integrating various media types
CO3	Create basic animations and interactive graphics
CO4	Apply graphics and multimedia techniques to solve real-world problems

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	1			1								2	
CO2	1	2	2	1	3	2							3	3
CO3	1	2	2		3								3	3
CO4	2	3	3	3	3	3							3	3

Assessment Scheme:

SN	Assessment	Weightage	Remark
1	Internal	100%	Laboratory assignments, internal POE etc.



Course Contents:		
Experiment 1:	**Graphics System Setup and Basic Drawing**	**2 Hours**
Experiment 2:	**Implementation of 2D Transformations**	**2 Hours**
Experiment 3:	**Implemenation of Line drawing algorithm**	**2 Hours**
Experiment 4:	**Implementation of Circle drawing algorithm**	**2 Hours**
Experiment 5:	**Implementation of Polygon filling algorithms**	**2 Hours**
Experiment 6:	**Implementation of basic ray tracing algorithm.**	**2 Hours**
Experiment 7:	**Create a simple multimedia file converter that can convert between different image an**	**4 Hours**
Experiment 8:	**Create a simple database system to store and retrieve multimedia file information with basic search functionality.**	**4 Hours**
Experiment 9:	**Create a basic 2D animation tool that can generate simple animations using frame-bas**	**4 Hours**

Text Books:

1. David F. Rogers, J. Alan Adams, "Mathematical elements for Computer Graphics", MGH International
2. David F. Rogers, "Procedural elements for Computer Graphics", MGH International
3. Donald Hearn & M. Pauline Baker, “Computer Graphics with OpenGL”, Third Edition, 2004, Pearson Education, Inc. New Delhi.
4. Ze-NianLi and Mark S. Drew, “Fundamentals of Multimedia”, First Edition, 2004, PHI Learning Pvt. Ltd., New Delhi.

Reference Books:

1. Andleigh, P. K and KiranThakrar, ―Multimedia Systems and Designl, PHI, 2003
2. Principles of Computer Graphics Theory and Practice Using OpenGL and Maya, Shalini Govil-Pai, (Springer) .
3. Computer Graphics (second Edition) - Zhigang Xiang & Roy Plastock (Schaum's Outline Series, TMGH).
4. Computer Graphics Using OpenGL F.S. Hill Jr. Stephen M. Kelley, (Pearson Education).

Course Code:	23CSEU5E13
Course Name:	Principles of AI/ML Laboratory

L	T	P	Credit
0	0	2	1

Course Prerequsites:

Basics of Mathematics, Statistics etc.

Course Description:

This course offers a comprehensive introduction to Artificial Intelligence, covering fundamental concepts such as knowledge representation, search strategies, logic-based reasoning, and learning algorithms. It explores classical and heuristic search methods, propositional and predicate logic, and key machine learning techniques including decision trees, neural networks, and reinforcement learning. The course also introduces probabilistic reasoning using Bayesian networks, equipping students with the theoretical and practical tools to design intelligent systems and solve complex real-world problems.

Course Outcomes: After the completion of the course the student will be able to -

CO1	Implement and evaluate various AI search algorithms and knowledge representation techniques for problem-solving applications.
CO2	Design and develop logic-based reasoning systems using propositional and predicate logic.
CO3	Build and analyze machine learning models for real-world datasets.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	2	2											3	3
CO2	3	2	2	2	3	3							3	3
CO3	3	3	3	3	3	3							3	3
CO4														

Assessment Scheme:

SN	Assessment	Weightage	Remark
1	In Semester Evaluation 1 (ISE1)	10%	Assignment, Test, Quiz, Seminar, Presentation, etc.
2	Mid Semester Examination (MSE)	30%	50% of course contents
3	In Semester Evaluation 2 (ISE2)	10%	Assignment, Test, Quiz, Seminar, Presentation, etc.
4	End Semester Examination (ESE)	50%	100% course contents

Course Contents:		
Experiment 1:	**Study of basic AI concepts and implement knowledge representation using semantic networks and frames.**	**2 Hours**
Experiment 2:	**Implement and analyze uninformed search algorithms for problem-solving.**	**2 Hours**
Experiment 3:	**Implement informed search algorithms using heuristics.**	**2 Hours**
Experiment 4:	**Implement logical reasoning systems using propositional and predicate logic.**	**2 Hours**
Experiment 5:	**Implement decision tree learning algorithm and analyze its performance.**	**2 Hours**
Experiment 6:	**Implement perceptron and multi-layer neural networks for classification tasks.**	**4 Hours**
Experiment 7:	**Implement Naive Bayes classifier and understand probabilistic reasoning in AI.**	**4 Hours**
Experiment 8:	**Implement K-means clustering algorithms**	**2 Hours**
Experiment 9:	**Implement Q-learning algorithm for reinforcement learning problems.**	**2 Hours**

Text Books:

1. Ben Coppin, Artificial Intelligence Illuminated, Jones and Bartlett, 2004.
2. Tom M. Mitchell, "Machine Learning", Mcgraw-Hill Education (Indian Edition), 2013.

Reference Books:

1. Elaine Rich Kevin Knight, Shivashankar B Nair: Artificial Intelligence, Tata McGraw Hill 3rd edition 2013.
2. Stuart Russel, Peter Norvig: Artiificial Intelligence A Modern Approach, Pearson 3rd edition 2013.
3. Ethem Alpaydin, "Introduction to Machine Learning", 2nd Edition, PHI Learning Pvt. Ltd., 2013.
4. T Hastie, R. Tibshirani, J.H.Fiedman, "The Elements of statistical learning", Springer, 1st Edition 2001.

Course Code:	23CSEU5O14	L	T	P	Credit
Course Name:	Cloud Computing	2			2

Course Prerequsites:

Basic knowledge of computer networks, operating systems, and distributed systems.

Course Description:

This course introduces the fundamental concepts of cloud computing, including its architecture, service models, and deployment methods. It covers essential technologies such as virtualization, cloud infrastructure, and major cloud platforms like AWS, Azure, and Google Cloud. The course also explores cloud security, applications in various domains, and emerging trends enabling students to understand and apply cloud technologies in real-world scenarios.

Course Outcomes:	After the completion of the course the student will be able to -
CO1	Understand fundamental concepts, architecture, and service models of cloud computing.
CO2	Analyze virtualization technologies and cloud infrastructure components.
CO3	Demonstrate the use of cloud platforms and assess security and application aspects in cloud environments.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	1	2	2				1		1	2	1	1
CO2	3	3	1	3	3		1		1	2	2	2	3	2
CO3	3	2	1	3	3		2		1	2	2	2	3	3

Assessment Scheme:

SN	Assessment	Weightage	Remark
1	End Semester Evaluation [50 Marks]	100%	100% Course Contents

Course Contents:

Unit 1	**Introduction to Cloud Computing**	**7 Hours**
Definition, characteristics and benefits of cloud computing, history of cloud computing, types of clouds (public, private, hybrid), service models: IaaS, PaaS, SaaS, challenges and risks in cloud computing, cloud computing vs grid computing vs cluster computing.		
Unit 2	**Virtualization and Cloud Infrastructure**	**7 Hours**
Virtualization: concept, types, hypervisors (Type 1 and Type 2), virtual machines, server, storage, and network virtualization; Cloud infrastructure: architecture, data centers, scalability, elasticity, load balancing, containerization (Docker, Kubernetes).		
Unit 3	**Cloud Services and Platforms**	**7 Hours**
Cloud service providers: AWS, Microsoft Azure, Google Cloud; compute services (EC2, Azure VM), storage services (S3, Azure Blob), database services, cloud deployment models; APIs in cloud computing, monitoring and management tools.		
Unit 4	**Security, Applications and Future Trends**	**7 Hours**
Security in cloud: data security, privacy, identity and access management, encryption; compliance and legal issues; cloud applications in healthcare, education, IoT; cloud-native development; edge and fog computing, future trends in cloud.		

Text Books:

1. Mastering Cloud Computing - Buyya R, Vecchiola C, Selvi S T, McGraw Hill Education (India), 2013
2. Cloud Computing - Principles and Paradigms - Buyya R, Broberg J, Goscinski A, Wiley,2011
3. Docker Cookbook - Sébastien Goasguen, O'reilly Nov. 2015 First Edition

Reference Books:

1. Cloud Computing Concepts, Technology & Architecture - Thomas Erl, Zaigham Mahmood, and Ricardo Puttini
2. Cloud Computing Bible - Barrie Sosinsky ,Wiley Publishing Inc. 2011
3. Cloud Native DevOps with Kubernetes – John Arundel and Justin Domingus
4. A to z on Docker: A complete Hands-On Guide to Docker Container – Swapnil Jain

Course Code:	23CSEU5M06
Course Name:	Test Driven Development

L	T	P	Credit
3	0	0	3

Course Prerequsites:

Basic programming knowledge (any language)
Logical and analytical thinking
Familiarity with software development process

Course Description:

This course provides an in-depth introduction to Test-Driven Development (TDD) using Java and the JUnit framework. Students will learn how to develop robust, scalable, and maintainable software by writing tests before code. The course also covers core Java programming, object-oriented principles, strings, collections, and unit testing practices. By the end of the course, students will be able to apply TDD in real-world software projects with an emphasis on code quality and reliability.

Course Outcomes: After the completion of the course the student will be able to -

CO1	Explain the principles and process of Test-Driven Development along with its benefits and limitations.
CO2	Develop Java programs using core syntax, object-oriented concepts, and collections framework.
CO3	Design and implement unit tests using JUnit for ensuring code quality and correctness.
CO4	Integrate test-driven practices in software development, including writing tests first and refactoring code.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1														
CO2														
CO3														
CO4														

Assessment Scheme:

SN	Assessment	Weightage	Remark
1	In Semester Evaluation 1 (ISE1)	10%	Assignment, Test, Quiz, Seminar, Presentation, etc.
2	Mid Semester Examination (MSE)	30%	50% of course contents
3	In Semester Evaluation 2 (ISE2)	10%	Assignment, Test, Quiz, Seminar, Presentation, etc.
4	End Semester Examination (ESE)	50%	100% course contents

D Y PATIL EDUCATION SOCIETY Deemed to be University Kasba Bawada, Kolhapur KOLHAPUR

Course Contents:

Unit 1	**Introduction to Test driven development**	**6 Hours**
Understanding the core principles of TDD, The benefits of TDD, Common misconceptions and limitations of TDD, Writing Effective Tests, Refactoring and Code Quality		
Unit 2	**Introduction to Java Programming**	**8 Hours**
Introduction to Java: History, features, and the Java Virtual Machine (JVM). Basic Syntax: Data types, variables, operators, control flow statements (if/else, loops)		
Unit 3	**Object-Oriented Programming (OOP)**	**7 Hours**
Working String and there functions, Regular expressions (regex), Working with Lists, Sets, Maps, and their implementations (ArrayList, HashMap, etc.).		
Unit 4	**String and Collections Framework**	**7 Hours**
Working String and there functions, Regular expressions (regex), Working with Lists, Sets, Maps, and their implementations (ArrayList, HashMap, etc.).		
Unit 5	**Unit Testing - JUnit Testing**	**7 Hours**
Importance of unit testing, Characteristics of good unit tests, Writing clear and concise test cases, What is Junit, Benefits of JUnit Testing in Java, Write test cases using Junit		
Unit 6	**JUnit Framework**	**6 Hours**
Benefits of JUnit Testing Framework, Features and Extensions, Set Up JUnit Testing, Annotations, JUnit Assertions,		

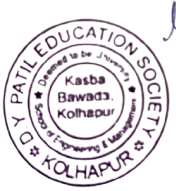
Text Books:

1. "Test-Driven Development: By Example" Author: Kent Beck Publisher: Addison-Wesley
2."Head First Java" (2nd Edition or later) Authors: Kathy Sierra and Bert Bates Publisher: O'Reilly Media
3. "Practical Guide to Testing in Java with JUnit" Author: Boni Garcia Publisher: Manning
4. "Clean Code: A Handbook of Agile Software Craftsmanship" Author: Robert C. Martin Publisher: Prentice Hall

Reference Books:

"JUnit in Action" (2nd Edition) Authors: Petar Tahchiev, Vincent Massol, Gary Gregory, and Felipe Leme

Oracle Java Documentation https://docs.oracle.com/javase/



Semester VI
Course Contents

Course Code:	23CSEU6P01		L	T	P	Credit
Course Name:	System Programing		3			3

Course Prerequsites:

Data Structures, Programming concepts.

Course Description:

This course explores the principles, algorithms, and data structures involved in the design and construction of compilers. Units include Language processors, lexical analysis, context-free grammars, push-down parsers, LR and LALR parsers, other parsing techniques, symbol tables and introduction to intermediate code generation.

Course Outcomes:	After the completion of the course the student will be able to -
CO1	Explain basics of Languages and Language processors for the System programming.
CO2	Analyze phases and steps for Software Program Execution in detail from Analysis to Execution.
CO3	Explain different phases of compiler in detail.
CO4	Design and develop modules for different phases of Compiler.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	1				2						2			
CO2	1			1	1						1		2	1
CO3	1				2						1	1	3	1
CO4	1			1	2						1	1	3	1

Assessment Scheme:

SN	Assessment	Weightage	Remark
1	In Semester Evaluation 1 (ISE1)	10%	Assignment, Test, Quiz, Seminar, Presentation, etc.
2	Mid Semester Examination (MSE)	30%	50% of course contents
3	In Semester Evaluation 2 (ISE2)	10%	Assignment, Test, Quiz, Seminar, Presentation, etc.
4	End Semester Examination (ESE)	50%	100% course contents

Course Contents:

Unit 1	**Language Processes & Assembler**	**8 Hours**
Introduction, language processing activities, Fundamentals of language processing, Fundamentals of language Specification Assemblers: Elements of assembly language programming, a simple assembly scheme, pass structure of assemblers, design of a two pass assembler.		
Unit 2	**Macros and Macro Processors**	**4 Hours**
Macro definition and call, Macro expansion, Nested macro calls, Advanced macro facilities, Design of macro pre-processor.		
Unit 3	**Phases in Compilers-Lexical Analysis & Syntax Analysis**	**10 Hours**
Role of a Lexical analyzer, input buffering, specification and recognition of tokens, Role of Parser, Writing grammars for context free environments, Top-down parsing- Recursive descent and predictive parsers (LL), Bottom-Up parsing		
Unit 4	**Syntax Directed Translation and Intermediate Code Generation**	**4 Hours**
Syntax directed definitions, construction of syntax tree, S-attributed definitions, L-attributed definitions.		
Unit 5	**Code Optimization**	**8 Hours**
Sources of optimization, Peephole optimization and basic blocks, loops in flow graphs, Data flow analysis and equations, code improving transformation and aliases		
Unit 6	**Code Generation**	**6 Hours**
Issues in design of a code generator and target machine, Run time storage management, Basic blocks and flow graphs, Next use information and simple code generator		

Text Books:

1. Systems Programming and Operating Systems- D.M. Dhamdhere, Second revised Edition, 2005, Tata McGraw- Hill Publishing Company limited, New Delhi.
2. Compilers - Principles, Techniques and Tools- A.V. Aho, R. Shethi and J.D. Ullman, Pearson Education.

Reference Books:

1. System Programming - J. J. Donovan (Mc-Graw Hill).
2. Compilers - Principles, Techniques and Tools- A.V. Aho, R. Shethiand J.D. Ullman, Addison Wesley Publishing Company.

Course Code:	23CSEU6P02	L	T	P	Credit
Course Name:	Cloud Computing	3	0	0	3

Course Prerequsites:

Basic knowledge of computer networks, operating systems, and distributed systems.

Course Description:

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 Outcomes:	After the completion of the course the student will be able to -
CO1	Explain the cloud computing architecture, types and models
CO2	Classify the virtualization techniques
CO3	Compare different architectures and platforms of cloud computing.
CO4	Summarize security threats and security measure for cloud computing

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	1	1	2				1	2	2	1	3	3
CO2	3	3	2	3	3				2	2	2	2	3	3
CO3	3	3	3	2	3	2	1	1	2	2	2	2	3	3
CO4	3	2	2	3	2	2	1		2	2	3	2	3	3

Assessment Scheme:

SN	Assessment	Weightage	Remark
1	In Semester Evaluation1 [10 Marks]	10%	Assignment, Test, Quiz, Seminar, Presentation, etc.
2	Mid Semester Examination [30 Marl	30%	50% Course Contents
3	In Semester Evaluation2 [10 Marks]	10%	Assignment, Test, Quiz, Seminar, Presentation, etc.
4	End Semester Examination [50 Marks	50%	100% Course Contents

D Y PATIL EDUCATION SOCIETY Deemed to be University Kasba Bawada, Kolhapur School of Engineering & Management KOLHAPUR

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**	**7 Hours**
Characteristics, Virtualization Techniques, Virtualization and Cloud Computing, Pros and Cons of Virtualization		
Unit 3	**Cloud Computing Architecture**	**7 Hours**
Cloud Reference Model, Types of Clouds – Public, Private, Hybrid and Community cloud, Types of Services – IaaS, PaaS, SaaS, Economics of Clouds, Open Challenges, Public Clouds: Amazon Web Services (AWS), Google Cloud Platform (GCP), and 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.		
Unit 5	**Advanced Concepts – Docker, Container and Kubernetes**	**7 Hours**
Introduction to CaaS, Why containers? Difference between Virtualization and Containers. Introduction to Containers, Docker and its architecture (Jain), Understanding Docker Container, Networking. Kuberentes – Introduction, Architecture. (cookbook) Case Study (Any case study available on the Internet such as - IBM, AWS, Google Qwiklabs using Kubernetes, docker container).		
Unit 6	**Cloud Security & Management**	**7 Hours**
Fundamental cloud security – Basic terms and concepts, Threat agents, cloud security threats, case study example. Cloud Management Mechanisms - SLA management and case study. Cloud Security Mechanisms – PKI, IAM and SSO with case studies.		

Text Books:

1. Mastering Cloud Computing - Buyya R, Vecchiola C, Selvi S T, McGraw Hill Education (India), 2013
2. Cloud Computing - Principles and Paradigms - Buyya R, Broberg J, Goscinski A, Wiley,2011
3. Cloud Computing Concepts, Technology & Architecture - Thomas Erl, Zaigham Mahmood, and Ricardo Puttini
4. A to z on Docker: A complete Hands-On Guide to Docker Container – Swapnil Jain
5. Docker Cookbook - Sébastien Goasguen, O'reilly Nov. 2015 First Edition

Reference Books:

1. Cloud Computing Bible - Barrie Sosinsky ,Wiley Publishing Inc. 2011
2. Cloud Native DevOps with Kubernetes – John Arundel and Justin Domingus

Course Code:	23CSEU6P03	L	T	P	Credit
Course Name:	Cloud Computing Lab			2	1

Course Prerequsites:

Basic understanding of computer networks, operating systems, and programming skills in Python or Java.

Course Description:

This laboratory course offers hands-on experience with fundamental cloud computing services and tools. Students will learn to work with virtual machines, cloud storage, databases, web hosting, and basic security features on platforms like AWS, Azure, or Google Cloud. The course aims to develop practical skills in deploying, managing, and monitoring cloud-based applications and resources.

Course Outcomes:	After the completion of the course the student will be able to -
CO1	Use public cloud environment
CO2	Build virtual machines using virtualization techniques
CO3	Make use of containers for software deployment

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3		2		3	1					2	3	3	2
CO2	2		2		3	1					2	3	3	2
CO3	2		2		3	1					2	3	3	2

Assessment Scheme:

SN	Assessment	Weightage	Remark
1	In Semester Evaluation [25 Marks]	50%	Experiment, Practical Performance and Oral Exam etc.
2	POE [25 Marks]	50%	100% course contents

List of Experiments:	
1	Use Google Collab book for writing program
2	Use google APIs to access google cloud services
3	Create Virtual Machine using emulator - emue and virtual library
4	Create Virtual Machines using KVM library - paravirtualized machine
5	Create bare-metal virtual machine
6	Create container using lxc
7	Create a container using docker - docker desktop , docker CLI
8	Networking of Docker Containers
9	Building Docker Image
10	Check the usage reports or activity logs of your cloud resources.

Text Books:

1. Mastering Cloud Computing - Buyya R, Vecchiola C, Selvi S T, McGraw Hill Education (India), 2013
2. Cloud Computing - Principles and Paradigms - Buyya R, Broberg J, Goscinski A, Wiley,2011
3. Cloud Computing Concepts, Technology & Architecture - Thomas Erl, Zaigham Mahmood, and Ricardo Puttini
4. A to z on Docker: A complete Hands-On Guide to Docker Container – Swapnil Jain
5. Docker Cookbook - Sébastien Goasguen, O'reilly Nov. 2015 First Edition

Reference Books:

1. Cloud Computing Bible - Barrie Sosinsky ,Wiley Publishing Inc. 2011
2. Cloud Native DevOps with Kubernetes – John Arundel and Justin Domingus

Course Code:	23CSEU6E13
Course Name:	Web Technology-II

L	T	P	Credit
2		2	3

Course Prerequsites:

Basic Programming Knowledge, Basic knowledge of HTML, CSS, and JavaScript, Introduction to Java, Basic knowledge of relational databases

Course Description:

This course provides comprehensive training in building full-stack web applications using React for the frontend and Spring Boot for the backend. Students will learn to design responsive user interfaces, develop RESTful APIs, and integrate both ends to create modern web applications. Emphasis is placed on component-based development, routing, state management, secure API development, and deployment.

Course Outcomes: After the completion of the course the student will be able to -

CO1	Design and implement dynamic user interfaces using React and its component-based architecture.
CO2	Develop secure and scalable backend services using Spring Boot and RESTful APIs.
CO3	Integrate frontend and backend technologies to build full-stack web applications.
CO4	Deploy and test full-stack applications with effective state management and secure API communication.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	1										1			
CO2	1	2			2					1	2		1	
CO3	1	1	2	1	3			2				2	2	2
CO4	1	2		1	1							1	2	

Assessment Scheme:

SN	Assessment	Weightage	Remark
1	In Semester Evaluation	50%	Assignment, Test, Quiz, Seminar, Presentation, etc.
2	POE	50%	100% course contents

Course Contents:

Unit 1	**Introduction to React**	**3 Hours**
Introduction to SPA and React.js JSX and Virtual DOM Functional Components and Props State and Lifecycle Methods Handling Events in React		
Unit 2	**Advanced React Features**	**5 Hours**
Conditional Rendering and Lists Forms and Input Handling Lifting State Up React Hooks: useState, useEffect Context API and Custom Hooks		
Unit 3	**React Routing and State Management**	**6 Hours**
React Router: Navigation, Route Parameters, Nested Routing Global State Management: useReducer, Context API Introduction to Redux (optional) API Calls using Axios / Fetch Error Handling and Loading States		
Unit 4	**Introduction to Spring Boot**	**5 Hours**
Overview of Spring Framework and Spring Boot Spring Boot Architecture and Dependencies (Maven/Gradle) Building REST APIs with Spring Boot		
Unit 5	**Data Persistence and Security**	**5 Hours**
Spring Data JPA and Hibernate CRUD Operations using Repositories Connecting to MySQL/PostgreSQL Spring Boot Security Basics (JWT/OAuth2 overview) Role-Based Access Control (RBAC)		
Unit 6	**Full Stack Integration and Deployment**	**4 Hours**
Connecting React Frontend with Spring Boot Backend Handling CORS and API Authentication Environment Configuration and .env files Deployment of application Project: Full-stack CRUD application with secure login		

Text Books:

1. Full Stack Development with Spring Boot 3 and React byJuha Hinkula Packt Publishing; 4th edition
2. Learning Spring Boot 3.0: Simplify the development of production-grade applications using Java and Spring by Greg L. Turnquist Dave Syer Mark Heckler Josh Long

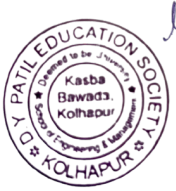
Reference Books:

Experiment List

1Setup React Development Environment (Node.js, npm, VS Code) and create a basic React app

D Y PATIL EDUCATION SOCIETY KOLHAPUR Kasba Bawada, Kolhapur

2Create React components using JSX, Props, and State
3Build forms in React and handle form events and validations
4Implement routing in React using React Router
5Use React Hooks (useState, useEffect) for state and side effects
6Setup Spring Boot project using Spring Initializr and build a basic REST API
7Develop CRUD operations using Spring Boot and MySQL/PostgreSQL
8Implement exception handling and validation in Spring Boot APIs
9Connect React frontend with Spring Boot backend using Axios
10Implement user login and role-based authentication (Spring Security + JWT)
11Manage environment variables and integrate .env in frontend/backend
12Final mini-project: Develop and deploy a full-stack web app (e.g., Task Manager, E-Commerce Admin, Event Manager)



Course Code:	23CSEU6E06
Course Name:	Programming Paradigms

L	T	P	Credit
3			3

Course Prerequsites:

Basic knowledge of data structures, object-oriented programming, and discrete mathematics is required.

Course Description:

This is one of the core course of Computer Science & Engineering Programme. In this course you will become familiar with the core concepts of OS - how OS work, how a processes & threads are created, inter-process communication & synchronisation, the various scheduling algorithms, memory management & memory allocation strategies, etc.

Course Outcomes: After the completion of the course the student will be able to -

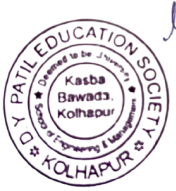
CO1	Explain the principles, strengths, and limitations of various programming paradigms.
CO2	Apply imperative, object-oriented, functional, and logic programming techniques to solve computational problems.
CO3	Analyze and compare programming paradigms based on problem requirements and performance considerations.
CO4	Design and implement multi-paradigm software solutions integrating concepts from different programming models.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	1	2									1	2	1	
CO2	2	2			2					1	2			
CO3	1	1	2	1	3							1		2
CO4	2	2		1	1								1	2

Assessment Scheme:

SN	Assessment	Weightage	Remark
1	In Semester Evaluation 1 (ISE1)	10%	Assignment, Test, Quiz, Seminar, Presentation, etc.
2	Mid Semester Examination (MSE)	30%	50% of course contents
3	In Semester Evaluation 2 (ISE2)	10%	Assignment, Test, Quiz, Seminar, Presentation, etc.
4	End Semester Examination (ESE)	50%	100% course contents



Course Contents:

Unit 1	**Introduction**	**6 Hours**
Definition and classification of programming paradigms, Historical evolution of programming languages Comparison of paradigms: strengths and limitations, Problem-solving approaches across paradigms, Language design principles and trade-offs		
Unit 2	**Imperative Programming**	**8 Hours**
Imperative programming model and state changes, Variables, assignment, and sequence control Structured programming principles, Procedures, functions, and parameter passing mechanisms, Memory management and pointer arithmetic, Control structures and iteration patterns Data Representation: The role of types, Basic types, Arrays, Records, Unions & Variant Records,Sets, Pointers.		
Unit 3	**Object-Oriented Programming**	**6 Hours**
Advanced OOP concepts: polymorphism, inheritance hierarchies, Abstract classes and interfaces Design patterns: Singleton, Factory, Observer, Strategy SOLID principles in software design Exception handling and resource management Generic programming and templates		
Unit 4	**Functional Programming**	**7 Hours**
Functional programming philosophy and pure functions, Higher-order functions and function composition, Recursion patterns and tail recursion optimization, Immutability and persistent data structures, Lambda expressions and closures, Currying, partial application, and function combinators, Monads and functional error handling, Lazy evaluation and infinite data structures		
Unit 5	**Logic Programming**	**6 Hours**
Logic programming paradigm and declarative thinking, Predicate logic and Horn clauses, Unification and backtracking mechanisms, Cut operator and control in Prolog, List processing and recursive data structures, Constraint logic programming basics		
Unit 6	**Concurrent and Parallel Programming**	**8 Hours**
Concurrency vs parallelism concepts, Thread creation and synchronization primitives, Race conditions, deadlocks, and their prevention, Message passing and actor model, Parallel algorithms and data parallelism, Asynchronous programming patterns		

Text Books:

1."Programming Language Pragmatics" by Michael L. Scott
2. "Concepts of Programming Languages" by Robert W. Sebesta

Reference Books:

1. "Modern Programming Languages: A Practical Introduction" by Adam Brooks Webber

Course Code:	23CSEU6E07
Course Name:	Ethical Hacking

L	T	P	Credit
3	0	0	3

Course Prerequsites:

Set Theory, Operating System, Data Structures, Basic Software Engineering Concept (SDLC)

Course Description:

Ethical hacking course is designed to help learners to develop a deeper understanding of threats to information system. We hope learners will develop a lifelong passion and appreciation for ethical hacking, which we are certain will help in future endeavours. Students will benefit from this learning experience. Almost all aspects of security are covered in this course.

Course Outcomes: After the completion of the course the student will be able to -

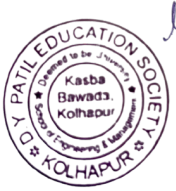
CO1	Define the basic concepts of components of Information and systems security.
CO2	Explain Footprinting, Reconnaissance, Network Scanning, Vulnerability Assessment, System Hacking, Malware Threats
CO3	Describe Sniffing and Social Engineering tools and techniques
CO4	Explain Sesion Hijacking, Firewall and IDS, Honeypot, Web Server and web applications security issues with SQL injection

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3												2	1
CO2	2				2			1					1	3
CO3	2				2			1					3	3
CO4	2				2			1					2	3

Assessment Scheme:

SN	Assessment	Weightage	Remark
1	In Semester Evaluation 1 (ISE1)	10%	Assignment, Test, Quiz, Seminar, Presentation, etc.
2	Mid Semester Examination (MSE)	30%	50% of course contents
3	In Semester Evaluation 2 (ISE2)	10%	Assignment, Test, Quiz, Seminar, Presentation, etc.
4	End Semester Examination (ESE)	50%	100% course contents



Course Contents:

Unit 1	**Introduction to Ethical Hacking**	**5 Hours**
Essential Terminology, Elements of Information Security, The Security, Functionality, and Usability Triangle, Top Information Security Attack Vectors, Information Security Threat Categories, Types of Attacks on a System, Information Warfare, Hacking Concept and Scope, Vulnerability Assessment, Penetration Testing		
Unit 2	**Footprinting and Reconnaissance**	**6 Hours**
FootprintingConcept,Footprinting Methodology, Overview of Network Scanning, Scanning Methodology, Vulnerability Assessment Concept, System Hacking, Mulware Threats		
Unit 3	**Sniffing and Social Engineering**	**8 Hours**
Sniffing Concepts, MAC attacks,DHCP attacks,ARP Poisoning, Spoofing Attack,DNS Poisoning,Sniffing Tools, Social Engineering Concepts and Techniques,Impersonation on Social Networking Site, Identity Theft		
Unit 4	**Session Hijacking and Firewall and Web Server**	**7 Hours**
IDS and Firewall Concepts and System,Evading IDS,Firewall, Web Server Concepts and attacks, Attack Methodology,Countermeasures,Patch Management		
Unit 5	**Web Application Hacking and SQL Injection**	**7 Hours**
Web App concepts and attack methodology, Countermeasures ,SQL Injection methodology, SQL Injection Techniques		
Unit 6	**Hacking Wireless Network and Mobile Platform**	**7 Hours**
Wireless Concept,Wireless Encryption, Wireless Threats,Hacking Methodology, Bluetooth Hacking, Wireless Security Tool, Mobile Platform Attack Vector, Hacking Android, iOS ,Blackberry, Understanding IoT Attack		

Text Books:

1. CEH V10: EC-Council Certified Ethical Hacker Complete Training Guide by IPSpecialist

Reference Books:

1.CEH v10 Certified Ethical Hacker Study Guide,Ric Messier, CEH, GCIH, GSEC, CISSP,SYBEX Publication

D Y PATIL EDUCATION SOCIETY Kasba Bawada, Kolhapur KOLHAPUR

Course Code:	23CSEU6E08	L	T	P	Credit
Course Name:	Image Processing	3	0	0	3

Course Prerequsites:

Basic knowledge of signals and systems, linear algebra, probability, and programming in Python

Course Description:

This course introduces the principles and techniques of digital image processing, covering image acquisition, enhancement, restoration, segmentation, compression, and color processing, with practical applications using tools like Python

Course Outcomes:	After the completion of the course the student will be able to -
CO1	Understand the basics of digital images and image acquisition.
CO2	Apply spatial and frequency domain techniques for image enhancement.
CO3	Perform image restoration and color image transformations.
CO4	Implement image segmentation, morphological operations, and compression methods.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	2	2	2				1	2	1	1	3	3
CO2	3	3	2	3	2				1	2	2	1	3	3
CO3	3	2	2	2	2				1	2	1	1	3	3
CO4	3	2	2	2	2				1	2	2	1	3	3

Assessment Scheme:

SN	Assessment	Weightage	Remark
1	In Semester Evaluation1 [10 Marks]	10%	Assignment, Test, Quiz, Seminar, Presentation, etc.
2	Mid Semester Examination [30 Mar	30%	50% Course Contents
3	In Semester Evaluation2 [10 Marks]	10%	Assignment, Test, Quiz, Seminar, Presentation, etc.
4	nd Semester Examination [50 Marks	50%	100% Course Contents

D Y PATIL EDUCATION SOCIETY KOLHAPUR Kasba Bawada, Kolhapur

Course Contents:

Unit 1	**Introduction to Digital Image Processing**	**5 Hours**
Fundamentals of image processing and computer vision, image sensing, acquisition and representation, sampling and quantization, basic relationships between pixels, imaging modalities: grayscale, color, binary		
Unit 2	**Image Enhancement in Spatial Domain**	**7 Hours**
Intensity transformations: contrast stretching, thresholding, histogram processing: equalization, matching, smoothing and sharpening filters (linear and non-linear), Laplacian and gradient-based enhancement		
Unit 3	**Image Enhancement in Frequency Domain**	**6 Hours**
Fourier Transform and DFT for images, frequency domain filtering: low pass, high pass, band pass, homomorphic filtering, Fast Fourier Transform (FFT)		
Unit 4	**Image Restoration and Color Processing**	**7 Hours**
Degradation model, noise models, restoration filters: inverse, Wiener, median, color image processing: RGB, HSV, HSI models, pseudocolor and full-color processing		
Unit 5	**Image Segmentation and Morphological Processing**	**8 Hours**
Edge detection: Sobel, Prewitt, Canny, thresholding, region growing, watershed segmentation, morphological operations: erosion, dilation, opening, closing, boundary detection and object representation		
Unit 6	**Image Compression and Applications**	**7 Hours**
Fundamentals of image compression, lossless vs lossy compression, Huffman coding, Run-Length encoding, JPEG compression, applications: medical imaging, satellite image processing, OCR		

Text Books:

1. Rafael C. Gonzalez & Richard E. Woods, Digital Image Processing, Pearson Education, 4th Edition
2. S. Jayaraman, S. Esakkirajan, T. Veerakumar, Digital Image Processing, McGraw Hill Education

Reference Books:

1. Anil K. Jain, Fundamentals of Digital Image Processing, Prentice-Hall
2. William K. Pratt, Digital Image Processing, Wiley
3. Bhabatosh Chanda & Dwijesh Dutta Majumder, Digital Image Processing and Analysis, PHI 4. Kenneth R. Castleman, Digital Image Processing, Pearson

Course Code:	23CSEU6E09
Course Name:	Programming Paradigms Lab

L	T	P	Credit
		2	1

Course Prerequsites:

Fundamentals of Programming, Data Structures and Algorithms, Object-Oriented Programming, Operating Systems

Course Description:

This lab course focuses on hands-on implementation of various programming paradigms, including imperative, object-oriented, functional, and declarative. Students will design concurrent and distributed systems, apply design patterns, and integrate paradigms to build robust software solutions.

Course Outcomes: After the completion of the course the student will be able to -

CO1	Apply different programming paradigms (imperative, object-oriented, functional, declarative) to solve problems.
CO2	Implement and analyze concurrent, parallel, and distributed programming using threads, actors, and synchronization.
CO3	Design and develop integrated software solutions combining multiple paradigms and design patterns.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	2		3								1	
CO2	3	3	2	2	3								2	
CO3	3	2	3		3			2					2	

Assessment Scheme:

SN	Assessment	Weightage	Remark
1	In Semester Evaluation(Internal)	100%	Assignment, Test, Quiz, Seminar, Presentation, etc.

Experiment List

1. Implement a scientific calculator that supports basic arithmetic, trigonometric functions, and memory operation
2. Implement a dynamic data structure library in C containing:
   - Dynamic array with automatic resizing
   - Linked list with insertion, deletion, and search operations
   - Hash table with collision handling
   - Binary search tree with balancing
3. Apply SOLID principles and implement design patterns in a real-world scenario.
4. Master functional programming concepts through practical implementation.
5. Build a file processing pipeline.
6. Develop proficiency in declarative logic programming
7. Implement thread-safe data structures and understand synchronization mechanisms.
8. Implement distributed systems using the actor model paradigm.
9. Design and implement parallel algorithms for computational problems.
10. Integrate multiple programming paradigms in a comprehensive software system.

Text Books:

1."Programming Language Pragmatics" by Michael L. Scott
2. "Concepts of Programming Languages" by Robert W. Sebesta

Reference Books:

1. "Modern Programming Languages: A Practical Introduction" by Adam Brooks Webber

Course Code:	23CSEU6E11	L	T	P	Credit
Course Name:	Image Processing Laboratory			2	1

Course Prerequsites:

Basic knowledge of signals and systems, linear algebra, and programming in Python.

Course Description:

This laboratory course provides hands-on experience with fundamental and advanced image processing techniques. Students will implement methods for image acquisition, enhancement in spatial and frequency domains, restoration, segmentation, morphological processing, and compression. The course emphasizes practical applications such as medical imaging, satellite image analysis, and optical character recognition (OCR).

Course Outcomes:	After the completion of the course the student will be able to -
CO1	Apply fundamental image processing techniques including acquisition, enhancement, and transformation.
CO2	Implement spatial and frequency domain methods for image enhancement and restoration.
CO3	Perform image segmentation, morphological processing, and basic compression techniques for practical applications.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	2	3	2	2			1	1	3	1	3	3
CO2	3	3	2	3	3	2			1	1	3	1	3	3
CO3	2	2	2	3	3	3			1	1	3	1	3	3

Assessment Scheme:

SN	Assessment	Weightage	Remark
1	In Semester Evaluation [25 Marks]	100%	Experiment, Practical Performance and Oral Exam etc.

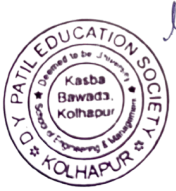
List of Experiments:	
1	Perform basic image operations like grayscale conversion, resizing, and rotation.
2	Demonstrate image sampling and quantization with different resolutions
3	Apply histogram equalization to enhance image contrast.
4	Use mean and median filters for image smoothing.
5	Apply Laplacian and gradient filters for image sharpening.
6	Perform Fourier Transform and apply low-pass filtering in frequency domain.
7	Restore a noisy image using Wiener and inverse filtering.
8	Convert an image from RGB to HSV and apply pseudocolor mapping.
9	Detect edges using Sobel, Prewitt, and Canny operators.
10	Compress an image using Run-Length Encoding and Huffman Coding.

Text Books:

1. Rafael C. Gonzalez and Richard E. Woods, Digital Image Processing, Pearson Education.
2. Anil K. Jain, Fundamentals of Digital Image Processing, PHI Learning.
3. S. Jayaraman, S. Esakkirajan, and T. Veerakumar, Digital Image Processing, McGraw Hill Education.

Reference Books:

1. William K. Pratt, Digital Image Processing, Wiley
2. B. Chanda and D. Dutta Majumder, Digital Image Processing and Analysis, PHI
3. Milan Sonka, Vaclav Hlavac, and Roger Boyle, Image Processing, Analysis, and Machine Vision, Cengage
4. Kenneth R. Castleman, Digital Image Processing, Pearson



Course Code:	23CSEU6E12
Course Name:	FOSS Tools

L	T	P	Credit
3	0	0	3

Course Prerequsites:

Basic Programming Knowledge and Software Engineering Fundamentals

Course Description:

This course introduces students to Free and Open Source Software (FOSS) concepts and essential tools used in software development. Students will learn FOSS philosophy, version control, document preparation, web development stack, containerization, and study real-world FOSS projects.

Course Outcomes: After the completion of the course the student will be able to -

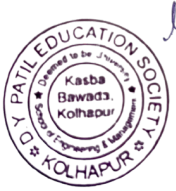
CO1	Understand FOSS philosophy and different software licensing models.
CO2	Use various FOOS tools like Git, docker, LaTeX.
CO3	Illustrate use of LAMP stack for web applications.
CO4	Summarize successful FOSS projects and their development models.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	1	2	1	1	2	3		2					3	
CO2	1	1	2	1	3				2	2	1	3	3	2
CO3	1	1	2	1	3							3	3	2
CO4	1	3	3	3	3									

Assessment Scheme:

SN	Assessment	Weightage	Remark
1	In Semester Evaluation 1 (ISE1)	10%	Assignment, Test, Quiz, Seminar, Presentation, etc.
2	Mid Semester Examination (MSE)	30%	50% of course contents
3	In Semester Evaluation 2 (ISE2)	10%	Assignment, Test, Quiz, Seminar, Presentation, etc.
4	End Semester Examination (ESE)	50%	100% course contents



Course Contents:

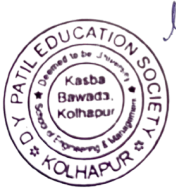
Unit 1	**FOSS Philosophy and Licensing**	**6 Hours**
Introduction to Free Software and Open Source Software. History of FOSS: GNU Project, Linux, Open Source Initiative. FOSS Philosophy: Four Freedoms, Benefits of Open Source. Software Licenses: Copyright basics, GPL, MIT, BSD, Apache License. License selection and compatibility. Legal aspects of FOSS.		
Unit 2	**Version Control with Git**	**8 Hours**
Introduction to Version Control Systems. Git basics: Repository, Working Directory, Staging Area. Basic Git commands: init, add, commit, status, log. Branching and Merging: Creating branches, switching branches, merging. Remote repositories: GitHub, clone, push, pull. Collaborative development: Pull requests, code reviews.		
Unit 3	**LaTeX Document Preparation**	**8 Hours**
Introduction to LaTeX: Advantages over word processors. Document structure: Document classes, packages, basic formatting. Text formatting: Sections, lists, emphasis. Mathematical equations and symbols. Including figures and tables. Bibliography and citations. Creating presentations with Beamer.		
Unit 4	**LAMP Stack**	**8 Hours**
Introduction to LAMP: Linux, Apache, MySQL, PHP. Apache Web Server: Installation, configuration, virtual hosts. MySQL Database: Basic SQL commands, database creation, user management. PHP Programming: Syntax, variables, functions, connecting to MySQL. Building a simple web application using LAMP.		
Unit 5	**Docker Containerization**	**8 Hours**
Introduction to Containers: Containers vs Virtual Machines. Docker basics: Images, containers, Docker Hub. Docker commands: run, build, pull, push. Creating Dockerfiles: Instructions and best practices. Docker Compose: Managing multi-container applications. Container deployment and management.		
Unit 6	**Case Studies and FOSS Projects**	**4 Hours**
Success stories: Linux, Apache, MySQL, Firefox. FOSS development models: Community vs corporate projects. Business models: Red Hat, Ubuntu, MongoDB. Community aspects: Governance, contributions, documentation. Current trends in FOSS. Getting involved in FOSS projects.		

Text Books:

Unit 1: "Understanding Open Source and Free Software Licensing" - Andrew M. St. Laurent (O'Reilly Media)
Unit 2: "Pro Git" - Scott Chacon and Ben Straub (Apress) - Available online at git-scm.com
Unit 3: "LaTeX: A Document Preparation System" - Leslie Lamport (Addison-Wesley)
Unit 4: "Learning PHP, MySQL & JavaScript" - Robin Nixon (O'Reilly Media)
Unit 5: "Docker: Up & Running" - Karl Matthias and Sean P. Kane (O'Reilly Media)
Unit 6: "The Cathedral and the Bazaar" - Eric S. Raymond (O'Reilly Media)

Reference Books:

"Free Software, Free Society" - Richard M. Stallman (GNU Press)
"Version Control with Git" - Jon Loeliger and Matthew McCullough (O'Reilly Media)
"The LaTeX Companion" - Frank Mittelbach, Michel Goossens (Addison-Wesley)
"Web Development with Apache and PHP" - James Lee and Brent Ware (Manning Publications)
"Docker Deep Dive" - Nigel Poulton (Independent Publishing)



Course Code:	23CSEU6E13
Course Name:	Blockchain Technology

L	T	P	Credit
3			3

Course Prerequsites:

Basics of Programming, networks, and cryptography

Course Description:

This course introduces the fundamentals of blockchain technology and smart contracts. It covers the architecture, cryptographic principles, consensus mechanisms, and key platforms such as Bitcoin and Ethereum. Students will gain hands-on experience in writing and deploying smart contracts using Solidity. The course also explores real-world applications of blockchain across industries like finance, supply chain, and healthcare.

Course Outcomes: After the completion of the course the student will be able to -

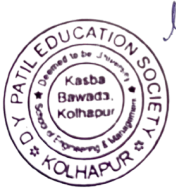
CO1	Write, deploy, and test smart contracts using Solidity on Ethereum
CO2	Set up and configure blockchain development environments and tools.
CO3	Develop mini DApps integrating smart contracts for real-world use cases.
CO4	Simulate blockchain transactions and interactions using decentralized tools.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	1										1			
CO2	1	2			2					1	2		1	
CO3	1	1	2	1	3			2				2	2	2
CO4	1	2		1	1							1	2	

Assessment Scheme:

SN	Assessment	Weightage	Remark
1	In Semester Evaluation 1 (ISE1)	10%	Assignment, Test, Quiz, Seminar, Presentation, etc.
2	Mid Semester Examination (MSE)	30%	50% of course contents
3	In Semester Evaluation 2 (ISE2)	10%	Assignment, Test, Quiz, Seminar, Presentation, etc.
4	End Semester Examination (ESE)	50%	100% course contents



Course Contents:

Unit 1	**INTRODUCTION TO BLOCKCHAIN**	**6 Hours**
Distributed DBMS – Limitations of Distributed DBMS, Introduction to Block chain -History, Evolution of Blockchain, Definition, Need of Blockchain, Distributed Vs Centralized Vs Decentralized, Public Ledgers : Blockchain as Public Ledgers, Distributed Ledger, Blockchain Categories – Public, Private, Consortium, Blockchain Network and Nodes, Peer-to-Peer Network, Mining Mechanism, Generic elements of Blockchain, Features of Blockchain, and Types of Blockchain, Benefits and Challenges of Blockchain Usages		
Unit 2	**BLOCKCHAIN ARCHITECTURE**	**7 Hours**
Operation of Bitcoin Blockchain, Blockchain Design Principles, Components of blockchain, Layered Architecture of Blockchain Ecosystem, Blockchain Architecture – Block, Hash, Distributed P2P, Merkle Tree, Structure of Blockchain- Types of Networks : Distributed Network , P2P Network, Consensus mechanism: Proof of Work (PoW), Proof of Stake (PoS), Byzantine Fault Tolerance (BFT), Proof of Authority (PoA) and Proof of Elapsed Time (PoET)etc. How Blockchain Works? Blockchain Demo - How Mining Works? (The NONCE and Cryptographic Puzzle) Immutable Ledger, Hard and Soft Forks, double spending		
Unit 3	**CRYPTO CURRENCY**	**6 Hours**
Bitcoin: Bitcoin and its History, Why use bitcoins? Where and how to buy bitcoins, Bitcoin transactions, How bitcoin transactions work, Bitcoin scripts and wallets. Ethereum: Ethereum Virtual Machine (EVM) – Wallets for Ethereum, Ethereum and Smart Contract, Solidity - Smart Contracts, Ether, Gas DApps , Decentralized Autonomous Organizations (DAO) Compare Bitcoin and Ether		
Unit 4	**SMART CONTRACT AND SOLIDITY FUNDAMENTALS**	**7 Hours**
Smart contracts, features of smart contract, types of Smart contract, advantages and challenges of smartcontract, Solidity: Introduction to solidity, Basic syntax, Data types, Operators, control flow, functions A programming structure in solidity		
Unit 5	**Solidity Advanced**	**7 Hours**
Constructors, inheritance, abstract contracts, interfaces, events, mapping, error handling, libraries		
Unit 6	**DIFFERENT BLOCKCHAIN FRAMEWORKS AND USE CASES**	**6 Hours**
Study of Blockchain Frameworks: Hyperledger, IOTA, Corda, Multichain, Quorum etc. Different use cases of blockchain other than cryptocurrencies		

Text Books:

1. Beginning Blockchain : A Beginner's Guide to Building Blockchain Solutions By Bikramaditya Singhal, Gautam Dhameja, Priyansu Sekhar Panda, Apress Media.
2. Imran Bashir, "Mastering BlockChain: Distributed Ledger Technology, Decentralization and Smart Contracts Explained", Packt Publishing, first edition – 2012
3. Satoshi Nakamoto, Bitcoin: A Peer-to-Peer Electronic Cash System
4. Mastering Ethereum: Building Smart Contracts and DAPPS, by Andreas Antonopoulos, Dr. Gavid Wood, Oreilly Publication
5. Anshul Kaushik, "BlockChain and Crypto Currencies", Khanna Publishing House, Delhi

Reference Books:

1. Learn Ethereum: Build your own decentralized applications with Ethereum and smart contracts by by Xun (Brian) Wu , Zhihong Zou , Dongying Song

Course Code:	23CSEU6E14
Course Name:	Augmented Reality/Virtual Reality

L	T	P	Credit
3	0	0	3

Course Prerequsites:

Computer Graphics and Mathematics

Course Description:

This course provides comprehensive coverage of Augmented Reality (AR) and Virtual Reality (VR) technologies. Students will learn to develop AR/VR applications using industry-standard tools. The course imparts theoretical knowledge, preparing students for careers in the rapidly growing AR/VR industry across gaming, healthcare, education, and enterprise applications.

Course Outcomes: After the completion of the course the student will be able to -

CO1	Demonstrate comprehensive understanding of AR/VR technologies and their applications
CO2	Apply computer vision techniques for tracking and recognition in AR systems
CO3	Create intuitive user interfaces for immersive 3D environments
CO4	Analyze current trends and future directions in AR/VR technology

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	2	1	2	1	3	2	1					1	3	2
CO2	2	1	1	1	3	1						1	3	3
CO3	1	1	2	1	2	2						1	3	3
CO4	1	2		2	3	3	2					1	2	3

Assessment Scheme:

SN	Assessment	Weightage	Remark
1	In Semester Evaluation 1 (ISE1)	10%	Assignment, Test, Quiz, Seminar, Presentation, etc.
2	Mid Semester Examination (MSE)	30%	50% of course contents
3	In Semester Evaluation 2 (ISE2)	10%	Assignment, Test, Quiz, Seminar, Presentation, etc.
4	End Semester Examination (ESE)	50%	100% course contents

Course Contents:

Unit 1	**Introduction to AR/VR**	**6 Hours**
Definitions and Terminology: Virtual Reality (VR) fundamentals, Augmented Reality (AR) fundamentals, Mixed Reality (MR) and Extended Reality (XR), Difference between AR, VR, and MR Historical Development: Evolution of AR/VR technologies, Key milestones and breakthrough moments Types of Reality Systems: Immersive VR systems, Non-immersive VR systems, Marker-based and markerless AR, Location-based AR		
Unit 2	**Hardware and Software Architecture**	**8 Hours**
VR Hardware Components: Head-mounted displays (HMDs), Motion tracking systems, Input devices and controllers, Audio systems and haptic feedback, Graphics processing requirements AR Hardware Components: Cameras and sensors, Display technologies (optical see-through vs video see-through), Mobile AR platforms, Smart glasses and wearable devices Software Architecture: VR/AR development platforms and engines, Unity 3D for AR/VR development, Unreal Engine fundamentals, WebXR and browser-based solutions, Mobile AR frameworks (ARCore, ARKit) Performance Considerations: Frame rate and latency requirements, Optimization techniques, Memory management, Battery life considerations for mobile AR		
Unit 3	**3D Graphics and Rendering for AR/VR**	**8 Hours**
3D Graphics Fundamentals: 3D coordinate systems and transformations, Perspective and orthographic projections, Viewing transformations for VR, Stereoscopic rendering Lighting and Shading: Real-time lighting models, Shadow mapping techniques, Physically-based rendering (PBR), Environmental lighting for AR Texture Mapping and Materials: UV mapping and texture coordinates, Normal mapping and bump mapping, Material properties for realistic rendering, Procedural textures Optimization Techniques: Level of detail (LOD) systems, Occlusion culling, Frustum culling, Batching and instancing		
Unit 4	**Computer Vision for AR Applications**	**8 Hours**
Image Processing Fundamentals: Digital image representation, Image filtering and enhancement, Edge detection algorithms, Feature extraction techniques Marker-Based Tracking: Fiducial markers and QR codes, Marker detection algorithms, Pose estimation from markers, Marker design principles Markerless Tracking: Natural feature tracking, SLAM (Simultaneous Localization and Mapping), Plane detection and estimation, Object recognition and tracking Camera Calibration: Intrinsic and extrinsic parameters, Calibration procedures, Distortion correction, Multiple camera systems		
Unit 5	**User Interface and Interaction Design**	**8 Hours**
VR Interaction Paradigms: 3D user interface design principles, Gaze-based interaction, Hand tracking and gesture recognition, Voice commands and spatial audio, AR Interface Design, Spatial UI design principles, Occlusion and depth perception, Touch and gesture interfaces, Context-aware interfaces Human Factors and Ergonomics: Motion sickness and cybersickness, Visual comfort and eye strain , Accessibility considerations, User experience evaluation methods, Interaction Techniques: Selection and manipulation in 3D space, Navigation and locomotion in VR, Menu systems and information display, Collaborative and multi-user interfaces		
Unit 6	**Applications and Emerging Trends**	**4 Hours**

Industry Applications: Medical visualization and surgical training, Architectural visualization and design, Industrial training and maintenance, ducational simulations and learning environments
Emerging Trends: Social VR and metaverse concepts, AI integration in AR/VR, Cloud-based AR/VR solutions, Haptic feedback and sensory integration

Text Books:

1. "Virtual Reality Technology" by Grigore C. Burdea and Philippe Coiffet
2. "Augmented Reality: Principles and Practice" by Dieter Schmalstieg and Tobias Hollerer

Reference Books:

1. "Learning Virtual Reality" by Tony Parisi
2. "Augmented Reality for Developers" by Jonathan Linowes and Krystian Babilinski
3. "Unity in Action" by Joe Hocking
4. "Real-Time Rendering" by Tomas Akenine-Möller, Eric Haines, and Naty Hoffman

Course Code:	23CSEU6E16
Course Name:	Blockchain Technology Lab

L	T	P	Credit
		2	1

Course Prerequsites:

Basics of Programming, networks, and cryptography

Course Description:

This course introduces the fundamentals of blockchain technology and smart contracts. It covers the architecture, cryptographic principles, consensus mechanisms, and key platforms such as Bitcoin and Ethereum. Students will gain hands-on experience in writing and deploying smart contracts using Solidity. The course also explores real-world applications of blockchain across industries like finance, supply chain, and healthcare.

Course Outcomes: After the completion of the course the student will be able to -

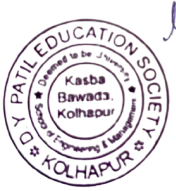
CO1	Describe the fundamentals, architecture, and types of blockchain systems.
CO2	Analyze cryptographic techniques and consensus mechanisms used in blockchain.
CO3	Develop smart contracts using Solidity and deploy them on Ethereum-like platforms.
CO4	Evaluate blockchain platforms and applications for real-world problem-solving in various domains.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	1										1			
CO2	1	2			2					1	2		1	
CO3	1	1	2	1	3			2				2	2	2
CO4	1	2		1	1							1	2	

Assessment Scheme:

SN	Assessment	Weightage	Remark
1	In Semester Evaluation 1 (ISE1)	100%	Assignment, Test, Quiz, Seminar, Presentation, etc.



Experiment List	
1	Introduction to Blockchain simulators and tools (Ganache, MetaMask, Remix IDE)
2	Setting up Ethereum blockchain environment using Ganache and connecting with MetaMask
3	Creating and deploying a basic smart contract using Solidity in Remix
4	Writing a smart contract for a voting system
5	Implementing a smart contract for a crowdfunding platform
6	Demonstrating a cryptocurrency transfer between accounts using smart contract
7	Managing ownership and access control in smart contracts
8	Testing smart contracts with Truffle framework (optional advanced)
9	Mini project: Develop a DApp with front-end integration

Text Books:

1. Beginning Blockchain : A Beginner's Guide to Building Blockchain Solutions By Bikramaditya Singhal, Gautam Dhameja, Priyansu Sekhar Panda, Apress Media.
2. Imran Bashir, "Mastering BlockChain: Distributed Ledger Technology, Decentralization and Smart Contracts Explained", Packt Publishing, first edition – 2012
3. Satoshi Nakamoto, Bitcoin: A Peer-to-Peer Electronic Cash System
4. Mastering Ethereum: Building Smart Contracts and DAPPS, by Andreas Antonopoulos, Dr. Gavid Wood, Oreilly Publication
5. Anshul Kaushik, "BlockChain and Crypto Currencies", Khanna Publishing House, Delhi

Reference Books:

1. Learn Ethereum: Build your own decentralized applications with Ethereum and smart contracts by by Xun (Brian) Wu , Zhihong Zou , Dongying Song

Course Code:	23CSEU6E14
Course Name:	Augmented Reality/Virtual Reality Laboratory

L	T	P	Credit
0	0	2	1

Course Prerequsites:

Computer Graphics and Mathematics

Course Description:

This laboratory course provides hands-on experience with Augmented Reality (AR) and Virtual Reality (VR) technologies. Students will develop practical skills in creating AR/VR applications using industry-standard tools and frameworks.

Course Outcomes: After the completion of the course the student will be able to -

CO1	Implement and demonstrate AR/VR applications using various development platforms
CO2	Develop computer vision-based tracking systems for marker-based and markerless AR applications
CO3	Design and implement intuitive 3D user interfaces with interaction techniques for immersive environments
CO4	Create and evaluate AR/VR applications for real-world scenarios

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3		3		3	2							3	3
CO2	3			1	3								3	3
CO3	1		3		3	2							3	3
CO4	1	3		3	3	3							3	3

Assessment Scheme:

SN	Assessment	Weightage	Remark
1	Internal	100%	Practical performance, Internal POE

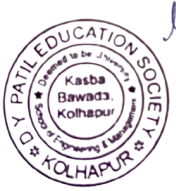
Course Contents:		
Experiment 1:	**Introduction to Unity 3D for AR/VR Development**	**2 Hours**
Experiment 2:	**Marker-Based AR Application Development**	**2 Hours**
Experiment 3:	**Markerless AR with Plane Detection**	**2 Hours**
Experiment 4:	**3D Graphics Rendering and Optimization**	**2 Hours**
Experiment 5:	**VR Interaction Systems**	**2 Hours**
Experiment 6:	**Computer Vision for AR Tracking**	**2 Hours**
Experiment 7:	**AR/VR User Interface Design**	**2 Hours**
Experiment 8:	**Medical AR Visualization**	**4 Hours**
Experiment 9:	**Educational VR Application**	**4 Hours**

Text Books:

1. "Virtual Reality Technology" by Grigore C. Burdea and Philippe Coiffet
2. "Augmented Reality: Principles and Practice" by Dieter Schmalstieg and Tobias Hollerer

Reference Books:

1. "Learning Virtual Reality" by Tony Parisi
2. "Augmented Reality for Developers" by Jonathan Linowes and Krystian Babilinski
3. "Unity in Action" by Joe Hocking
4. "Real-Time Rendering" by Tomas Akenine-Möller, Eric Haines, and Naty Hoffman



Course Code:	23CSEU6N18		L	T	P	Credit
Course Name:	Project Management Tools		1		2	2

Course Prerequsites:

1. Software Engineering

Course Description:

This course aims to provide students with a practical and theoretical foundation in project management principles and the use of modern tools to plan, execute, monitor, and closure of software projects. Emphasis is placed on tools like MS Project, Trello, JIRA, and others for managing real-world project scenarios.

Course Outcomes:	After the completion of the course the student will be able to -
CO1	Explain the fundamentals of software project management and project life cycle
CO2	Apply project planning and scheduling techniques using modern tools
CO3	Demonstrate tracking, monitoring, and risk management in software projects
CO4	Collaborate effectively using online tools to manage a mini-project

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	1	2	2								1			1
CO2	1	2	2	2	3				3	1		1	1	
CO3	1	1	2	2	3				3	2		1	3	1
CO4	1	1	2	1	3					2		3	3	2

Assessment Scheme:

SN	Assessment	Weightage	Remark
1	In Semester Examination (ISE)	50%	100% course contents
2	External Oral Examination (OE/POE)	50%	100% course contents

D Y PATIL EDUCATION SOCIETY Deemed to be University Kasba Bawada, Kolhapur School of Engineering & Management KOLHAPUR

Course Contents:		
Unit 1	**Introduction to Project Management**	**3 Hours**
Definition, characteristics, and importance of projects, Project Life Cycle & stakeholders, Role and responsibilities of a project manager, Project initiation: Charter, scope, and objectives, Traditional vs. Agile project management methodologies		
Unit 2	**Project Planning and Scheduling**	**4 Hours**
Work Breakdown Structure (WBS), Time estimation techniques (PERT/CPM overview), Task dependencies and milestones, Gantt Charts, Critical Path Method (CPM), Hands-on overview of MS Project / OpenProject for schedule generation		
Unit 3	**Monitoring, Risk, and Cost Management**	**4 Hours**
Resource allocation and leveling, Cost estimation and budgeting, Project tracking: Earned Value Analysis (EVA), Risk identification and mitigation strategies, Change control and performance metrics		
Unit 4	**Agile Tools and Project Closure**	**3 Hours**
Agile principles: Scrum, Kanban, sprints, Tools: Trello, JIRA – Backlogs, sprint planning, burndown charts, Project closure: Reports, audits, lessons learned, Final documentation and project handoff.		
Text Books:		
1. “Information Technology Project Management”, Kathy Schwalbe, Cengage Learning, 7/e, 2013. 2. "Technology Ventures From Idea to Enterprise",Thomas H. Byers, Richard C. Dorf, Andrew J., Nelson		
Reference Books:		
1. "Software Project Management", M. Cottrell and B. Hughes, McGraw-Hill, 5/e, 2009. 2. "Project Management Software Tools: A Guide to Choosing the Right Tools" by Michael S. Dobson 3. "The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses" by Eric		

Course Code:	23CSEU6M05
Course Name:	Web testing

L	T	P	Credit
2	0	0	2

Course Prerequsites:

Basic understanding of software testing concepts (manual testing, SDLC)

Familiarity with Java programming

Fundamental knowledge of HTML, CSS, and browsers

Course Description:

This course introduces students to the principles and practices of automation testing in modern software development. It covers essential concepts including the benefits, limitations, and frameworks for automation testing. Students will gain practical experience using Selenium WebDriver, learn to write and manage automated test scripts using TestNG, and implement testing frameworks such as Page Object Model and Data-Driven testing. The course also integrates topics like continuous integration, reporting, and cross-browser testing to prepare students for real-world testing environments.

Course Outcomes: After the completion of the course the student will be able to -

CO1	Explain the principles, benefits, and types of automation testing and frameworks used in industry.
CO2	Develop and execute automated test scripts using Selenium WebDriver and TestNG framework.
CO3	Design and implement modular, reusable test automation frameworks with CI/CD integration and reporting tools.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1														
CO2														
CO3														
CO4														

Assessment Scheme:

SN	Assessment	Weightage	Remark
4	End Semester Examination (ESE)	100%	100% course contents

Course Contents:

Unit 1	**Introduction to Automation Testing**	**6 Hours**
What is automation testing?, Benefits and limitations of automation testing, Types of automation testing (e.g., functional, regression), Manual vs. Automation testing, Test automation frameworks (e.g., Data-Driven, Keyword-Driven, Page Object Model)		
Unit 2	**Selenium Fundamentals**	**8 Hours**
What is Selenium and its components (IDE, WebDriver, Grid), Selenium WebDriver architecture, Locating elements on a webpage (e.g., XPath, CSS selectors), Handling different browser types and versions, Working with various web elements (buttons, text boxes, dropdowns, etc.), Handling pop-ups, alerts, and frames		
Unit 3	**TestNG Framework**	**7 Hours**
Introduction to TestNG and its annotations, Writing test cases and test suites, TestNG listeners and reporting, Parallel test execution, and Data-driven testing with TestNG.		
Unit 4	**Framework Design**	**7 Hours**
Page Object Model (POM) design pattern, Keyword-driven framework, Data-driven framework, and Developing reusable test components, dependency management and build automation, continuous integration and continuous deployment (CI/CD), Logging and reporting using log4j or other tools, , Cross-browser testing		

Text Books:

"Selenium WebDriver Practical Guide" Author: Unmesh Gundecha Publisher: Packt Publishing

"TestNG Beginner's Guide" Author: Varun Menon Publisher: Packt Publishing

"Learning Selenium Testing Tools with Python" (if using Java, refer for architecture concepts) Author: Unmesh Gundecha

Reference Books:

"Effective Software Testing: A Developer's Guide" Author: Mauricio Aniche Publisher: Manning Publications

"Continuous Integration: Improving Software Quality and Reducing Risk" Authors: Paul M. Duvall, Steve Matyas, and Andrew Glover Publisher: Addison-Wesley