

COURSE TITLE	WEB 3.0 DEVELOPMENT USING BLOCKCHAIN
COURSE CODE	05MF0206
COURSE CREDITS	4

Objective:

- 1 Students will be able to understand the core concepts of Blockchain technology, including decentralized ledgers, consensus mechanisms, and cryptographic principles, to lay a solid foundation for further exploration.
- 2 Students will be able to learn how to build and deploy a basic Blockchain network using Python, including creating blocks, implementing consensus algorithms, and securing the network with cryptographic techniques
- 3 Students will be able to acquire the skills to design, write, and deploy smart contracts on the Ethereum Blockchain using Solidity, including handling contract lifecycle, implementing functions, and interacting with contracts.
- 4 Students will be able to learn the principles and techniques of developing decentralized applications (DApps), including front-end development, integration with smart contracts, and deployment on Blockchain platforms.
- 5 Students will possess comprehensive knowledge in Blockchain fundamentals, including its underlying principles, components, and types, enabling precise analysis and evaluation of Blockchain-based solutions.

Course Outcomes: After completion of this course, student will be able to:

- 1 Students will demonstrate a thorough understanding of Blockchain fundamentals, including its underlying principles, components, and types, enabling them to analyze and evaluate Blockchain-based solutions effectively.
- 2 Students will be able to design, develop, and deploy a functional Blockchain network using Python, demonstrating competency in block creation, consensus mechanisms, and network security.
- 3 Students will develop the ability to write, deploy, and interact with smart contracts on the Ethereum Blockchain using Solidity, enabling them to automate and execute decentralized transactions securely.
- 4 Students will showcase their proficiency in designing and developing decentralized applications (DApps), including front-end interfaces, integration with smart contracts, and deployment, empowering them to contribute to the growing ecosystem of decentralized technologies.
- 5 Students will be able to design, develop, and deploy a functional Blockchain network using Python, demonstrating competency in block creation, consensus mechanisms, and network security. Additionally, students will develop the ability to write, deploy, and interact with smart contracts on the Ethereum Blockchain using Solidity, enabling them to automate and execute decentralized transactions securely. These skills empower students to actively contribute to the growing ecosystem of decentralized technologies by designing and developing decentralized applications (DApps), including front-end interfaces and integration with smart contracts, ultimately showcasing their proficiency in this emerging field.

Pre-requisite of course:Basic Knowledge of Frontend and Python

Teaching and Examination Scheme

Theory Hours	Tutorial Hours	Practical Hours	ESE	IA	CSE	Viva	Term Work
3	0	2	50	30	20	25	25

Contents : Unit	Topics	Contact Hours
1	Fundamentals of Blockchain Introduction to Blockchain Technology, Historical Overview and Evolution of Blockchain, Key Concepts: o Decentralized Ledger. o Consensus Mechanisms. o Cryptographic Hash Functions., Types of Blockchain: o Public. o Private. o Consortium., Blockchain Components: o Blocks. o Transactions. o Nodes. o Wallets., Distributed Ledger Technology (DLT) and its relation to Blockchain., Use Cases and Applications of Blockchain Across Various Industries., Challenges and Limitations of Blockchain Technology.	11
2	Blockchain Implementation Using Python Setting Up a Local Blockchain Network Using Python Libraries., Understanding Block Structure and Creating Blocks. Implementing Proof of Work (PoW) Consensus Algorithm., Building a Basic Blockchain Network with Peer-to-Peer Communication., Validating Transactions and Maintaining Consensus Among Nodes., Securing the Blockchain Network with Cryptographic Techniques., Exploring Tools and Frameworks for Blockchain Development in Python., Testing and Debugging Blockchain Applications.	11
3	Smart Contract Development Using Solidity Introduction to Smart Contracts: o Their significance in Blockchain applications., Overview of Ethereum and its Virtual Machine (EVM), Introduction to Solidity Programming Language., Writing and Deploying Smart Contracts: o On the Ethereum Blockchain., Understanding Contract Lifecycle: o Creation. o Execution. o Termination., Implementing Functions, Variables, and Modifiers in Solidity Contracts., Handling Exceptions and Error Handling in Smart Contracts.	11
4	DApp Application Development Introduction to Decentralized Applications (DApps). Design Principles and Architecture of DApps. Interacting with Smart Contracts: o Using Ethereum wallets and web interfaces. Front-End Development for DApps: o Using web3.js or similar libraries. Interacting with Smart Contracts from the Client-Side Application.	12
Total Hours		45

Suggested List of Experiments:

Contents : Unit	Topics	Contact Hours
1	Unit 1 - Explain the concept of decentralization in Blockchain technology and discuss its significance. - Compare and contrast public, private, and consortium Blockchains, providing examples of each. - Describe the process of adding a new block to the Blockchain and explain how consensus is achieved among network participants. - Demonstrate the use of cryptographic hash functions in securing transactions on the Blockchain. - Investigate real-world applications of Blockchain technology across industries and discuss their potential impact. - Analyze the limitations and challenges associated with Blockchain adoption and scalability. - Design a conceptual Blockchain network architecture for a supply chain management system, outlining its components and interactions. - Debate the ethical considerations and regulatory challenges surrounding Blockchain technology adoption and implementation. - Implement a basic Blockchain network in Python, including classes for blocks, transactions, and the Blockchain itself. - Develop a proof-of-work (PoW) consensus algorithm to validate new blocks added to the Blockchain. - Create a peer-to-peer communication system for nodes in the Blockchain network using sockets or a similar mechanism. - Implement data serialization and deserialization methods to ensure data integrity and consistency within the Blockchain. - Explore different methods for storing Blockchain data, including in- memory storage, flat files, and databases.	15
2	Unit 2 - Write a simple smart contract in Solidity to implement a basic token transfer function. - Deploy the smart contract to a local Ethereum test network using tools like Ganache or Remix. - Implement modifiers in a Solidity smart contract to enforce access control and permission-based functions. - Write unit tests for the smart contract using tools like Truffle or Solidity tests, ensuring its functionality and integrity. - Develop a decentralized voting application using Solidity smart contracts, enabling users to cast votes securely and transparently. - Explore the concept of gas and its significance in executing transactions and deploying smart contracts on the Ethereum Blockchain. - Create a decentralized marketplace smart contract that facilitates peer-to- peer trading of digital assets, ensuring fairness and transparency., - Build a simple decentralized application (DApp) front end using HTML, CSS, and JavaScript, connecting it to a smart contract deployed on the Ethereum Blockchain. - Implement user authentication and authorization mechanisms in a DApp using Ethereum wallet integrations like MetaMask. - Integrate the Web3.js library into a DApp front end to interact with smart contracts deployed on the Ethereum Blockchain.	15
Total Hours		30

Textbook :

- 1 Mastering Blockchain: Unlocking the Power of Cryptocurrencies, Smart Contracts, and Decentralized Applications, Imran Bashir, Packt Publishing, 2018

References:

- 1 Python Blockchain Programming, Python Blockchain Programming, Harish Kumar Garg, Packt Publishing, 2017
- 2 Building Ethereum Dapps: Decentralized Applications on the Ethereum Blockchain, Building Ethereum Dapps: Decentralized Applications on the Ethereum Blockchain, Roberto Infante, O'Reilly Media, 2019
- 3 Solidity Programming Essentials: A Beginner's Guide to Build Smart Contracts for Ethereum and Blockchain, Solidity Programming Essentials: A Beginner's Guide to Build Smart Contracts for Ethereum and Blockchain, Ritesh Modi, Packt Publishing, 2018
- 4 Ethereum Smart Contract Development: Build blockchain-based decentralized applications using Solidity, Ethereum Smart Contract Development: Build blockchain-based decentralized applications using Solidity, Mayukh Mukhopadhyay, Packt Publishing, 2020

Suggested Theory Distribution:

The suggested theory distribution as per Bloom’s taxonomy is as follows. This distribution serves as guidelines for teachers and students to achieve effective teaching-learning process

Distribution of Theory for course delivery					
Remember / Knowledge	Understand	Apply	Analyze	Evaluate	Higher order Thinking / Creative
10.00	20.00	25.00	25.00	10.00	10.00

Instructional Method:

- 1 Board Work
- 2 PPT
- 3 Demo

Supplementary Resources:

- 1 <https://web3js.readthedocs.io/en/v1.10.0/>
- 2 <https://archive.trufflesuite.com/>