Introduction to Blockchain Technology (BCT)

E4SEM1 Elective **Subject Code: CS3312**

Course Objectives

By the end of the course, students will be able to

1. Understand how blockchain systems (mainly Bitcoin and Ethereum) work,
2. To securely interact with them,
3. Design, build, and deploy smart contracts and distributed applications,
4. Integrate ideas from blockchain technology into their own projects.

Syllabus:

UNIT I: Introduction to blockchain- History, definition, overview of centralized, decentralized and distributed architectures. Simplified architecture, Structure of a block, Introduction to hash functions - SHA256, Merkle tree and merkle root, introduction to bitcoin- Definition, Transaction life cycle, Bitcoin mining, Introduction to public key cryptography- Signature and Authentication,

UNIT II: Bitcoin – Creation of bitcoins, Transactions, Address generation, Use of public and private keys in bitcoin. Introduction to FORTH language, Bitcoin Script- Understanding of operators and execution of script using stack, Transaction validation using bitcoin script. Bitcoin peer-to-peer network- Joining procedure, Relaying transactions, Relaying blocks.

UNIT III: Consensus in Bitcoin- Introduction, Proof of work, Proof of stake, Proof of elapsed time, Proof of burn, Monopoly in bitcoin, Attacks- Double spending attack, Sybil attack, Denial of service attack, Bitcoin mining, Difficulty of mining, Permissioned blockchain – Definition, smart contracts- Distributed state machine replication (crowd funding example) Consensus algorithms : RAFT consensus, Network faults, Byzantine general problem, Practical byzantine fault tolerance systems.

Unit IV: Blockchains for enterprises (Hyperledger fabric)– Introduction, Actors and components in blockchain, System architecture, Transaction flow, ordering services, Channels, Single channel and multi channel networks, Hyperledger fabric network setup, Usecases- Blockchain in financial services, Identity management and other sectors.

Unit V: Introduction to Docker, Docker compose, Node.js, Git client, Creating a network using Hyperledger fabric- Executing first network(github), Ethereum- Introduction, Network types, gas, Tools for ethereum application development, Introduction to solidity programming, challenges of blockchain technology-scalability, Interoperability, standardizations, Energy intensive, Regulations.

Course Outcomes

1. Explain design principles of Bitcoin and Ethereum.
2. Explain Nakamoto consensus.
3. Explain the Simplified Payment Verification protocol.
4. List and describe differences between proof-of-work and proof-of-stake consensus.
5. Interact with a blockchain system by sending and reading transactions.
6. Design, build, and deploy a distributed application.

7. Evaluate security, privacy, and efficiency of a given blockchain system.