Curriculum of Semester-II

Executive M.Tech in Computer Science and Engineering

(With Specialization in Cloud Computing & Blockchain)

Semesters	Subject Code	Courses Name	Category	L	T	P	Credits
Semester-2	CS35211CS	Data Engineering and Analytics	DC	2	0	2	**3**
	CS35212CS	Cloud Computing and Distributed Systems	DC	2	0	2	**3**
	CS35213CS	Blockchain Technologies	DC	2	0	2	**3**
	CS3522XCS	Elective-2	DC	2	0	2	**3**
			Total Credits				**12**

Tentative List of Elective Subjects (may be updated as and when required)

Elective-2
Cryptography and Network Security (CS35221CS)
Grid Computing (CS35222CS)

DATA ENGINEERING AND ANALYTICS

Subject Code: CS35211CS	**Subject Type:** DC	**Evaluation Scheme:** MSE (30), TA (20), ESE (50)	**L-T-P:** 2-0-2 (3)

Pre-Requisites: Database Management Systems, SQL Programming, Data Structures, Basic Programming Knowledge (Python/Java), and Fundamentals of Distributed Systems

Course Objective: To understand the evolution of database systems from relational databases to modern distributed data platforms. To study advanced database architectures designed for Big Data and large-scale distributed systems. To explore different NoSQL database models including document, column-family, key-value, and graph databases.

Course Outcomes: At the end of the course, the student will be able to

CO	Course Outcomes
CO1	Analyze modern database architectures and explain the evolution from traditional RDBMS to NoSQL and distributed databases.
CO2	Design data models using document, column-family, key-value, and graph database systems.
CO3	Evaluate distributed database concepts such as sharding, replication, CAP theorem, and consistency models.
CO4	Implement and analyze modern database technologies including MongoDB, Cassandra, HBase, and Neo4j.
CO5	Assess emerging database systems such as NewSQL, cloud-native databases, and data-intensive architectures.

Syllabus:

Unit 1: Evolution of Database Systems and Big Data Foundations: Data Management Concepts, Structured Query Language (SQL), Data Types and Constraints in MySQL, SQL for Data Definition (DDL), SQL for Data Manipulation (DML), SQL for Data Query (DQL), Data Updation and Deletion, Transactions and ACID Properties. Early Database Systems, Hierarchical and Network Models, Relational Database Systems, First Database Revolution, Second Database Revolution, Third Database Revolution, Limitations of Relational Databases. Big Data Revolution, Characteristics of Big Data (Volume, Velocity, Variety, Veracity, Value), Introduction to Hadoop Ecosystem, Hadoop Distributed File System (HDFS), MapReduce Overview, Hive, Pig, Return of SQL in Big Data Systems.

Unit 2: Distributed Databases and NoSQL Foundations: Distributed Database Architecture, Distributed System Characteristics, Data Partitioning, Horizontal and Vertical Partitioning, Sharding Techniques, Replication Strategies. Consistency Models (Strong Consistency, Eventual Consistency, Causal Consistency), ACID vs BASE Properties, CAP Theorem, Distributed Transactions, Two-Phase Commit (2PC). NoSQL Databases Overview, Types of NoSQL Databases (Key-Value, Document, Column-Family, Graph Databases), NoSQL Data Models

Unit 3: Document and Key-Value Databases: Document Data Model, Documents and Collections, JSON Data Representation, Embedded Data Models, Normalized Data Models. MongoDB Architecture, MongoDB Data Manipulation Language, CRUD Operations, indexing in MongoDB, MongoDB Replication using Replica Sets, MongoDB Sharding. Key-Value Databases, Redis Architecture, Key-Value Database Use Cases.

Unit 4: Column-Family Databases: Column-Family Data Model, Wide Column Databases, Databases and Tables, Columns, Types and Keys. Apache Cassandra Architecture, Cassandra Query Language (CQL), Keyspaces and Column Families, Data Partitioning in Cassandra, Replication Strategies, Cluster Management, Fault Tolerance. Hadoop HBase Architecture, HBase Data Model, Comparison of Cassandra and HBase, Large-scale Distributed Storage Systems.

Unit 5: Graph Databases and Emerging Database Technologies: Graph Data Model, Nodes, Edges, and Properties, Graph Database Use Cases. Neo4j Architecture, Neo4j Deployment (Standalone and Cluster), Cypher Query Language, Querying Graph Data, Applications in Social Networks, Recommendation Systems, Fraud Detection Systems, Knowledge Graphs. NewSQL Databases, Google Spanner Architecture (Overview), Distributed SQL Databases, Cloud Databases, Multi-model Databases, Database-as-a-Service (DBaaS), Data Lakes and Data Warehouses, Modern Data-Intensive Architectures.

Learning Resources:

Text Books:

1. Sadalage, P. & Fowler, M. NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence. Pearson Education, 2012.
2. Redmond, E. & Wilson, J. Seven Databases in Seven Weeks: A Guide to Modern Databases and the NoSQL Movement. Pragmatic Programmers, 2012.
3. Dan Sullivan. NoSQL for Mere Mortals. Addison-Wesley Professional, 2015.
4. Guy Harrison. Next-Generation Databases. Apress, 2016.
5. Martin Kleppmann. Designing Data-Intensive Applications. O'Reilly Media, 2017.

CLOUD COMPUTING AND DISTRIBUTED SYSTEMS

Subject Code: CS35212CS	**Subject Type:** DC	**Evaluation Scheme:** MSE (30), TA (20), ESE (50)	**L-T-P:** 2-0-2 (3)

Pre-Requisites: Operating Systems, Computer Networks, Data Structures and Algorithms, Database Systems, Programming (Python/Java), Basic Knowledge of Virtualization and Parallel Computing

Course Objective: To provide a comprehensive understanding of the principles, architectures, and technologies underlying cloud computing and distributed systems. To enable students to design, analyze, and deploy scalable, fault-tolerant, and secure distributed applications in modern cloud environments.

Course Outcomes: At the end of the course, the student will be able to

CO	Course Outcomes
CO-1	Analyze the fundamental principles and theoretical models of distributed systems and cloud computing
CO-2	Design scalable and fault-tolerant distributed architectures using modern cloud paradigms
CO-3	Implement distributed computing frameworks and cloud-native applications
CO-4	Evaluate performance, security, and reliability of cloud-based systems in real-world scenarios

Syllabus:

Unit-1: Foundations of Distributed Systems

Models of distributed systems: synchronous vs asynchronous systems, system architectures (client-server, peer-to-peer, microservices), Communication paradigms: RPC, RMI, message passing, publish-subscribe systems, Time and ordering: logical clocks, vector clocks, causality. Consistency models: strong consistency, eventual consistency, CAP theorem, Distributed coordination: consensus (Paxos, Raft), leader election, distributed mutual exclusion.

Unit-2: Cloud Computing Architecture and Virtualization

Cloud computing fundamentals: service models (IaaS, PaaS, SaaS), deployment models (public, private, hybrid), Virtualization technologies: hypervisors, containers, container orchestration (Docker, Kubernetes), Resource management and scheduling in cloud environment, Data center design and energy-efficient computing, Serverless computing and Function-as-a-Service (FaaS).

Unit-3: Distributed Data Management and Big Data Systems

Distributed file systems: GFS, HDFS, Distributed databases: NoSQL systems (key-value, column-family, document stores), Data partitioning, replication, and sharding strategies, Consistency, availability, and partition tolerance trade-offs, Big data processing frameworks: MapReduce, Apache Spark, Stream processing systems and real-time analytics.

Unit-4: Scalability, Fault Tolerance, and Security in Cloud Systems

Scalability techniques: load balancing, auto-scaling, elastic resource provisioning, Fault tolerance: replication, checkpointing, failure detection, recovery mechanisms, Monitoring and observability in distributed systems, Cloud security: identity and access management (IAM), data security, encryption, zero-trust architecture, Privacy, compliance, and governance in cloud systems, Emerging trends: edge computing, fog computing, blockchain integration with cloud.

Learning Resources:

Text Books:

1. George Coulouris, Jean Dollimore, Tim Kindberg, Distributed Systems: Concepts and Design, Pearson.
2. Kai Hwang, Geoffrey Fox, Jack Dongarra, Distributed and Cloud Computing, Morgan Kaufmann.

Reference Books:

1. Martin Kleppmann, Designing Data-Intensive Applications, O'Reilly.
2. Brendan Burns, Kubernetes: Up and Running, O'Reilly.
3. Tanenbaum & Van Steen, Distributed Systems, Pearson.
4. Rajkumar Buyya et al., Cloud Computing: Principles and Paradigms, Wiley.

BLOCKCHAIN TECHNOLOGIES

Subject Code: CS35213CS	**Subject Type:** DC	**Evaluation Scheme:** MSE (30), TA (20), ESE (50)	**L-T-P:** 2-0-2 (3)

Pre-Requisites: Cryptography, Basic knowledge of computer security, Networking

Course Objective: To demonstrate a familiarity with the concepts related to blockchain technology. To apply the knowledge of cryptography and distributed systems to design decentralized applications. To design and build smart contracts and distributed applications (DApps) for different applications.

Course Outcomes: At the end of the course, the student will be able to

CO	Course Outcomes
CO-1	Students will be able to explain the core concepts and components of blockchain technology.
CO-2	Students will be able to design and implement basic blockchain architectures and understand the security and consensus mechanisms required for their development.
CO-3	Students will understand smart contracts, their technical capabilities, practical applications, limitations and security constraints.
CO-4	Students will be able to analyse and explore the Real-World applications of blockchain technology.

Syllabus:

Unit-1: Introduction to Blockchain

Introduction to Blockchain and Digital Currency, Cryptographic primitives, Zero-Knowledge Hash Functions, Evolution, Blockchain as public ledger, Structure of a Block, Transactions, Merkel Trees, Peer-to-Peer Networks, Blockchain Architecture, Types of Blockchain, Characteristics, Benefits and Challenges, Double Spend Problem, Byzantine Generals problem.

Unit-2: Consensus Algorithm and Smart Contract

Consensus Algorithms: Proof of Work (PoW), Hashcash PoW, Bitcoin PoW, Attacks on PoW and monopoly problem, Proof of Stake (PoS), Delegated Proof of Stake (DPoS), Byzantine Fault Tolerance (BFT), Proof of Burn, Proof of Elapsed Time, Proof of History (PoH), Leader based Consensus Mechanism: Paxos, Raft. Smart Contract, Applications of Smart Contracts, Mining, Hardness of Mining.

Unit-3: Ethereum and Hyperledger

Ethereum and Smart Contracts, The Turing Completeness of Smart Contract Languages and verification challenges, Using smart contracts to enforce legal contracts, comparing Bitcoin scripting vs. Ethereum Smart Contracts, Ethereum Virtual Machine (EVM), Wallets for Ethereum, Solidity, Hyperledger fabric, the plug and play platform and mechanisms in permissioned blockchain, Hyperledger Fabric components, Hyperledger Composer, Corda, Chaincode, Design a Distributed Application (DAPP).

Unit-4: Security and Real-World Applications

Pseudo-anonymity vs. anonymity, Zcash and Zk-SNARKS for anonymity preservation, attacks on Blockchains – such as Sybil attacks, selfish mining, 51% attacks, advent of Algorand, and Sharding based consensus algorithms, Blockchain Applications: Cryptocurrencies, Healthcare, Financial system, Supply chain management, E-governance, Real-Estate, Judiciary, Micropayments, Insurance, Sidechains, Agriculture, Cross-chain interoperability, DeFi & NFTs.

Learning Resources:

Text Books:

1. S. Shukla, M. Dhawan, S. Sharma, S. Venkatesan, 'Blockchain Technology: Cryptocurrency and Applications', Oxford University Press, 2019.
2. Josh Thompson, 'Blockchain: The Blockchain for Beginners, Guide to Blockchain Technology and Blockchain Programming', Create Space Independent Publishing Platform, 2017.
3. Blockchain Basics: A Non-Technical Introduction in 25 Steps by Daniel Drescher, Apress.
4. The Basics of Bitcoins and Blockchains by Antony Lewis, O'Reilly Media.

Elective-2

CRYPTOGRAPHY AND NETWORK SECURITY

Subject Code: CS35221CS	**Subject Type:** DE (Elective-2)	**Evaluation Scheme:** MSE (30), TA (20), ESE (50)	**L-T-P:** 3-0-0 (3)

Pre-Requisites: Mathematics, Computer Networks

Course Objective: To understand securing computer network protocols based on cryptography techniques. To understand various protocols for network security to protect against the threats in the networks. To get an insight into the working of different existing cryptographic algorithms.

Course Outcomes: At the end of the course, the student will be able to

CO	Course Outcomes
CO1	Students will be able to understand classical and modern symmetric cryptographic techniques.
CO2	Students will be able to apply number theory and algebraic concepts in cryptography.
CO3	Students will be able to analyse and implement public-key cryptographic algorithms.
CO4	Students will be able to evaluate advanced cryptographic systems and security protocols.

Syllabus:

Unit-I: Introduction to Cryptography with Public-Key Cryptosystems

Introduction to Cryptography, Classical Cryptosystem, Cryptanalysis on Substitution Cipher (Frequency Analysis), Play Fair Cipher, Block Cipher, Data Encryption Standard (DES), Triple DES and Modes of Operation, Stream Cipher, Pseudorandom Sequence, LFSR Based Stream Cipher.

Unit-II: Elementary Number Theory

Abstract Algebra: Group, Ring, Field, Number Theory: Modular Arithmetic: Addition, Subtraction, Multiplication, Exponentiations, Prime Numbers, Preliminaries and Basic Group Theory: Cyclic Group, Greatest Common Divisor (GCD): Properties of GCD, Euclidean and Extended Euclidean Algorithm, Multiplicative Inverse, Linear Congruence, Solving Systems of Linear Congruence using Chinese Remainder Theorem, Fermat's Little Theorem: Application in Primality Testing and Carmichael Numbers, Euler's theorem, Quadratic Residue, Polynomial Arithmetic, Advanced Encryption Standard (AES).

Unit-III: Introduction to Public-Key Cryptosystems and Primality Testing

Introduction to Public Key Cryptosystem, Discrete Logarithm Problem (DLP), Diffie-Hellman Key Exchange, Knapsack Cryptosystem, RSA Cryptosystem, Primarily Testing, ElGamal Cryptosystem, Elliptic Curve over the Reals, Elliptic curve Modulo a Prime, Generalised ElGamal Public Key Cryptosystem, Rabin Cryptosystem, Legendre and Jacobi Symbols, Message Authentication, Digital Signature.

Unit-IV: Key Management, Hashing, Cryptanalysis, and Network Security Protocols

Key Management, Key Exchange, Hash Function, Universal hashing, Cryptographic Hash Function, Secure Hash Algorithm, Digital Signature Standard, Key Exchange Protocols, Cryptanalysis: Memory Trade off Attack, Differential Cryptanalysis, Linear Cryptanalysis, Cryptanalysis and Stream Cipher, Shamir Secret Sharing, Identity Based Encryption (IBE), Attribute Based Encryption, Functional Encryption, Implementation Attacks, The Secure Sockets Layer (SSL), Pretty Good Privacy (PGP).

Learning Resources:

Text Books:

1. Introduction to Modern Cryptography by Jonathan Katz and Yehuda Lindell.
2. Cryptography: Theory and Practice by Douglas Stinson.
3. Cryptography: An Introduction by Nigel Smart.
4. A Graduate Course in Applied Cryptography by Dan Boneh and Victor Shoup.
5. Handbook of Applied Cryptography by Alfred Menezes, Paul van Oorschot, and Scott Vanstone.

GRID COMPUTING

Subject Code: CS35222CS	**Subject Type:** DE (Elective-2)	**Evaluation Scheme:** MSE (30), TA (20), ESE (50)	**L-T-P:** 3-0-0 (3)

Pre-Requisites: Distributed systems, C/C++/Java/Python, Computer network

Course Objective: To understand how to create a Virtual Organization (VO), a temporary or permanent alliance of individuals and institutions that share resources to achieve a common goal.

Course Outcomes: At the end of the course, the student will be able to

CO	Course Outcomes
CO-1	Understand the fundamental principles of grid computing and explain the layered architecture of the Grid Protocol.
CO-2	Design and apply security measures using GSI (Grid Security Infrastructure) and digital certificates to ensure secure resource sharing.
CO-3	Compare different grid middleware (e.g., Globus, gLite) and select the appropriate tools for specific large-scale computational problems.
CO-4	Build or deploy a functional grid service that solves a complex scientific or business problem.

Syllabus:

Unit-1: Foundation of Grid Computing

Introduction, Anatomy and Physiology of the Grid, Grid vs. Cluster Computing, Grid vs. Cloud Computing, The Grid Protocol Architecture (Fabric, Connectivity, Resource, Collective, and Application layers), Open Grid Services Architecture (OGSA) and Open Grid Services Infrastructure (OGSI).

Unit-2: Grid Monitoring & Resource Management

Overview of monitoring systems (R-GMA, GridICE, MDS), Ganglia, Hawkeye, and Network Weather Service (NWS), Local Schedulers (Condor, PBS, LSF, SGE), Principles of Gridway and Gridbus Broker, Scheduling with Quality of Service (QoS).

Unit-3: Grid Security & Virtualization

Grid Security Infrastructure (GSI), PKI, X.509 Certificates, and Grid Security Primer, Trust models, CAS (Community Authorization Service), and Identity Management, Role of Virtual Machines (VMs) and Containers in Grid environments, Resource isolation, and abstraction.

Unit-4: Data Grids & Middleware Tools

GridFTP for high-speed transfer, Replica Management, and Metadata catalogs, Comprehensive study of the Globus Toolkit (architecture and components like GRAM and MDS), User interfaces for job submission and monitoring.

Learning Resources:

Text Books:

1. Joshy Joseph, Craig Fellenstein, Grid Computing, IBM Press, 2003.
2. Barry Wilkinson, Grid Computing: Techniques and Applications, Chapman & Hall/CRC Press, 2009.