**Course Plan**

1. **Course Information**

|  |  |  |  |
| --- | --- | --- | --- |
| **Course Code** | 22AIE303 | **Title** | Database Management Systems |
| **Academic Year** | 2024 – 2025 | **Semester** | V |
| **Program** | 2022 BTech AIE | **L** – **T** – **P** – **C** | 2-1-3-4 |

1. **Faculty Information:**

|  |  |
| --- | --- |
| **Sl. No** | **Faculty Name** |
| 1 | Raji Ramachandran |

1. **Course Overview**

This course aims to understand the concepts of database design, database languages, database-system implementation and maintenance. The course will provide knowledge of the design and development of databases for AI applications using SQL and python. The course will provide an understanding of various database systems including modern database systems apt for AI and ML applications

1. **Course Outcomes**

|  |  |
| --- | --- |
| **CO#** | **Outcome** |
| CO1 | Formulate relational algebraic expressions, SQL and PL/SQL statements to query relational databases. |
| CO2 | Build ER models for real-world databases |
| CO3 | Design a normalized database management system for real-world databases |
| CO4 | Apply the principles of transaction processing and concurrency control. |
| CO5 | Use high-level right database for AI and ML application. |

1. **CO-PO affinity matrix**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| PO/PSO | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 | PSO1 | PSO2 |
| CO |
| CO1 | 3 |  |  |  | 3 |  |  |  | 2 | 2 |  |  | 1 |  |
| CO2 |  | 1 | 3 |  | 2 |  |  |  | 2 | 2 |  |  | 1 |  |
| CO3 | 1 |  | 2 |  |  |  |  |  | 2 | 2 |  |  | 1 |  |
| CO4 | 1 |  |  |  |  |  |  |  |  |  |  |  | 1 |  |
| CO5 | 1 |  |  |  |  |  |  |  | 2 | 2 |  | 1 |  | 2 |

1 – Week affinity, 2 – Moderate affinity, 3 – Strong affinity , - No affinity

1. **CO-PO/PSO Affinity Justification.**

|  |  |  |  |
| --- | --- | --- | --- |
| **CO-PO Justification** | | | |
| **CO** | **PO/PSO** | **Affinity** | **Justification** |
| **1** | PO1 | 3 | Formulating relational algebraic expressions or SQL queries requires mathematical operations such as union, intersection, Cartesian products, and projections to manipulate and query databases. |
| PO5 | 3 | Implementation of SQL and PL/SQL uses modern tools/ software which are widely used in the industry for managing and querying relational databases. The ability to formulate these queries equips students with the technical skills needed to use modern database management systems. |
| PO9 | 2 | For continuous evaluation, one of the component is a team project which will help the students to function effectively as an individual and as a member in team. |
| PO10 | 2 | The component team project includes both documentation and presentation which will enable the students to communicate effectively. |
| PSO1 | 1 | Relational databases are a critical part of many computing systems, including interdisciplinary applications. By learning how to formulate relational algebraic expressions, SQL, and PL/SQL statements, students gain the foundational skills required to efficiently handle and manipulate large datasets, which is often a key component of designing innovative computing solutions. |
| **2** | PO2 | 1 | Designing a relational database using an ER model begins with identifying and understanding the real-world problem or system that requires a database solution. This step mirrors the first aspect of problem analysis, where students need to clearly define the scope and requirements of a problem before solving it. |
| PO3 | 3 | Relational database design, especially using an ER (Entity-Relationship) model, is essential for solving complex real-world problems. Engineering tasks often involve managing and organizing vast amounts of data. Designing an efficient and scalable database system is crucial to solving these problems effectively, ensuring that information is stored, retrieved, and processed in an optimal manner. |
| PO5 | 2 | Designing a relational database using an ER (Entity-Relationship) model typically requires the use of modern database design tools and software, such as ERD (Entity-Relationship Diagram) modeling tools. These tools facilitate the creation, visualization of database, aligning with the program outcome of applying modern IT tools to solve complex engineering problems. |
| PO9 | 2 | For continuous evaluation, one of the component is a team project which includes design of database using ER model for a real life application which will help the students to function effectively as an individual and as a member in team. |
| PO10 | 2 | The team project enable the students to ccommunicate effectively with in the team and also to make proper documentation and effective presentations. |
| PSO1 | 1 | Designing relational databases using the ER model requires understanding of how to apply standard practices and tools to solve complex problems. Designing relational databases involves creating efficient and scalable data models that integrate interdisciplinary knowledge and leverage established technologies. |
| **3** | PO1 | 1 | Database design using normalization concepts involves understanding and application of functional dependency theory that relies on mathematical principles and logical reasoning. By mastering normalization, students demonstrate their ability to apply these mathematical and logical concepts to solve complex problems. |
| PO3 | 2 | Normalization is a critical technique in designing database schemas that can effectively handle complex data relationships and ensure data integrity. By applying normalization concepts, students address the intricate problems associated with data redundancy and anomalies, thus contributing to designing robust solutions for complex engineering problems. |
| PO9 | 2 | For continuous evaluation, one of the component is a team project in which the students’ needs to apply normalization technique for a real life problem which will help the students to function effectively as an individual and as a member in team. |
| P10 | 2 | The team project enables the students to design a normalized relational database schema for a real life application which will help the students to ccommunicate effectively and also to make proper documentation and effective presentations. |
| PSO1 | 1 | Normalization is a fundamental process in database design that helps in organizing data efficiently by reducing redundancy. By applying normalization, students can create database schemas that are not only functional but also optimized for performance and scalability. This innovative approach ensures that the database can handle complex queries and large volumes of data efficiently. |
| **4** | PO1 | 1 | Transaction processing and concurrency control rely on mathematical concepts such as algorithms, and logic. For instance, ensuring atomicity and consistency in transactions often involves understanding mathematical principles. |
| PSO1 | 1 | Designing and implementing computing solutions related to database application often involves handling multiple operations that must be executed reliably and efficiently. Transaction processing and concurrency control are crucial for ensuring that these operations are managed correctly, even in the presence of simultaneous transactions. Applying these principles ensure that solutions are not only innovative but also robust and optimal. |
| **5** | PO1 | 1 | Identifying a suitable high-level database requires a deep understanding of the specific needs of the given application and the different features of high level databases. This aligns with the program outcome of applying fundamental knowledge in engineering to solve complex problems |
| PO9 | 2 | Students were given group case study in which they need to compare different high level databases based on certain key factors . Working in a group enable students to function effectively as an individual, and as a member in a team. |
| P10 | 2 | The case study given helps the students to ccommunicate effectively in a group and to make proper documentation and presentation . |
| P12 | 1 | As new high level databases with varying features are emerging day by day, the students needs to be a life long learner to identify which database is suitable for a particular real life application. |
| PSO2 | 2 | Emerging computing paradigms often introduce new database technologies, architectures, and methodologies. Understanding these paradigms enables students to stay tuned with the latest developments in database systems, which is crucial for selecting the most suitable high-level database for modern applications. |

1. **Syllabus**

**Unit 1**

Introduction: Overview of DBMS fundamentals – Overview of Relational Databases and Keys. Relational Data Model: Structure of relational databases – Database schema – Formal Relational Query Languages – Overview of Relational Algebra and Relational Operations. Database Design: Overview of the design process - The E-R Models – Constraints - Removing Redundant Attributes in Entity Sets - E-R Diagrams - Reduction to Relational Schemas - Entity Relationship Design Issues - Extended E-R Features – Alternative E-R Notations – Overview of Unified Modelling Language (UML).

**Unit 2**

Relational Database Design: Features of Good Relational Designs - Atomic Domains and 1NF - Decomposition using Functional Dependencies: 2NF, 3NF, BCNF and Higher Normal Forms. Functional Dependency Theory - Algorithm for Decomposition – Decomposition using multi-valued dependency: 4NF and 4NF decomposition. Database design process and its issues. SQL: review of SQL – Intermediate SQL – Advanced SQL.

**Unit 3**

Transactions: Transaction concept – A simple transaction model - Storage structure - Transaction atomicity and durability - Transaction isolation – Serializability – Recoverable schedules, Casecadeless schedules. Concurrency control: Lock-based protocols – Locks, granting of locks, the two-phase locking protocol, implementation of locking, Graph-based protocols. Deadlock handling: Deadlock prevention, Deadlock detection and recovery.

Case Study: Different types of high-level databases – MongoDB, Hadoop/Hbase, Redis, IBM Cloudant, Dynamo DB, Cassandra and Couch DB etc. Tips for choosing the right database for the given problem.

**Textbooks/References**

*Silberschatz A, Korth HF, Sudharshan S. Database System Concepts. Sixth Edition, TMH publishing company limited; 2011.*

*Garcia-Molina H, Ullman JD, Widom J. Database System; The complete book. Second Edition, Pearson Education India, 2011*

*Elmasri R, Navathe SB. Fundamentals of Database Systems. Fifth Edition, Addison Wesley*

**9. Evaluation Policy**

1. **Direct Assessment Tools**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Component |  | | | Internal (70) | | | | External (30) |
| Midterm (20%) | CA theory (20) | | |  | CA lab (30) | |  |
| Quiz (10%) | Case study (5%) | | Tutorial Evaluation  (5%) | Project 20% marks | Lab evaluation (10%) | End semester Exam (30%) |
| Details | 20% | 1 quiz | Viva | | 1 quiz | Project Report (5%) + Presentation + Viva (15%) | viva |  |
| CO | CO1  CO2  CO3 | CO 2,3,4 | CO 5 | |  | CO 3,4 | CO1 | CO1, CO2, CO3, CO4 |

1. **Indirect Assessment Tools**

Course Exit Survey

1. **Mode of conduct of evaluations**

|  |  |
| --- | --- |
| **Evaluation** | **Mode of conduct** |
| Periodicals and End semester | Periodicals and End semester exams will be conducted in the examination hall. |
| Quiz | Online quizzes conducted through LMS during lab sessions. |
| Assignment | Lab activity in which students need to write and execute SQL queries for different schema given. Students need to complete and submit the given assignments within a deadline. |
| Case study | Students will be given a topic/ question in advance and the students need to submit it within a deadline and will conduct viva to evaluate. |
| Project | Students will be divided into different groups and each group will be assigned a project topic for degigning a relational database schema. It is evaluated based on presentation followed by viva and a project report submission . |

1. **Rubrics**

**Lab Assessments**

|  |  |  |  |
| --- | --- | --- | --- |
| **Criteria** | **Performance Levels** | | |
| **Good** | **Average** | **Poor** |
| **Design** | The SQL query written solves the problem correctly. | The SQL query written solves the problem partialy. | Attempted but the SQL query written is incomplete or wrong |
| **Implementation** | Clean and correct implementation of SQL query. | The SQL query is written with with less efficiency. | Incomplete query, syntax/ logical mistake query. |
| **Output** | Correct output for all test cases | Some of the test cases failed | Wrong output for most of the test cases. |

**Assignment**

|  |  |  |  |
| --- | --- | --- | --- |
| **Criteria** | **Performance Levels** | | |
| **Good** | **Average** | **Poor** |
| **Timely Submission** | Assignment submitted on or before the deadline. | Late Submission by a week. | No submission with no prior communication or justification. |
| **Completion** | All questions are completed. | Partially Completed. | Answers to questions are incomplete, inaccurate, or missing. |

**10. Course Outcome Attainment Levels**

60 (out of 100) is set as threshold for all course outcomes. The target levels are given below.

|  |  |
| --- | --- |
| **Attainment Level** | **Target %** |
| High | >= 70 |
| Medium | >= 50 |
| Low | < 50 |

**11. Weekly Lecture Plan**

|  |  |  |  |
| --- | --- | --- | --- |
| **Week** | **Topics** | **Sub Topics to be covered** | **COs mapped** |
| 1 | Introduction: Overview of DBMS fundamentals.  Relational Databases. | * Database System Applications * Purpose of Database Systems * View of Data- Data Abstraction * Instances and Schemas * Data Models * Database Languages   Data Manipulation Language Data Definition Language | CO2 |
| 2 | Introduction to  Relational Databases – Structure of relational databases – Database  schema | * Overview of Relational Databases, * Database Design & Transactions Database Architecture * Structure of Relational Databases * Database Schema, Key | CO2 |
| 3 | Relational Data Languages  Relational Algebra | * Formal Relational Query Languages * Fundamental relational algebra operations – * Additional relational algebra operations. | CO1 |
| 4 | Database Design: E-R Model, | * E-R Diagrams   Notations, Constraints, Cardinality  Participation  Constraints,  Weak and strong entity sets   * E-R diagram for University   Enterprise | CO2 |
| 4 | Database Design:  Reduction of ER model  to relational model, ER design issues. | * Reduction to Relational Schemas Representation of Strong   Entity Sets with Simple Attributes.  Representation of Strong Entity Sets with Complex Attributes.  Representation of Weak Entity Sets.   * Mapping of Relaitionships   . | CO2 |
| 5 | Extended ER features, Alternate ER notations | Extended E-R Features   * Specialization * Generalization * Attribute Inheritance * Constraints on Generalization * Aggregation * Reduction to Relational Schema   Representation of Generalization | CO2 |
| 6 | Relational Database Design:: | * Relational Database Design * Features of Good Relational Designs Design Alternative :   Larger Schemas  Design Alternative : Smaller Schemas   * Atomic Domains and 1NF | CO3 |
| 7-8 | Decomposition using Functional  Dependencies: 2NF,  3NF,BCNF and Higher Normal Forms. | * Decomposition using Functional Dependencies   Keys and Functional Dependencies BCNF  BCNF and Dependency Preservation  3NF   * Higher Normal Forms | CO3 |
| 9 | Decomposition using Functional  Dependencies: Functional-  Dependency theory | Functional Dependency Theory Closure of a Set of Functional Dependencies  Closure of Attribute Sets Canonical Cover  Lossless Decomposition Dependency Preservation  Algorithm for decomposition  BCNF Decomposition 3NF decomposition | CO3 |
| 10-11 | Transactions: | * Transaction Concept * A Simple Transaction Model | CO4 |

12. Suggestions received from previous cycle and changes incorporated.