CSE 311 – Database Management Systems

Credits:	3 SCH
Course type:	Required, Computer Science, Lecture
Course Prerequisites:	CSE 115 (Computing Concept), CSE 135 (Introduction to Computer Programming), CSE 225 (Data Structures), CSE 173(Discrete Mathematics)
Course Schedule/Timing:	Lecture – 3 Hours/Week, Lab works 3 Hours/Week
Instructor:	Dr. Kamruddin Nur kamruddin.nur@northsouth.edu
Consultation Hours:	SAT 09.40 AM – 12.50 PM
Course Assessment:	Lab Assignment: 4 Lab works: 7 Project work: 1 Exam: Quizzes – 4, Midterm - 1, Final - 1
Grading policy:	Attendance- 10%, Lab works and Assignments – 15%, Project:10% Quiz – 15%, Midterm-20%, Final – 30%

Course Content:

- Data Management Fundamentals:
  - Flat file system and drawbacks of the flat file system, introduction of modern database systems, users of database system
  - Benefits and characteristics of database approach, Integrated data management
- Data Modeling:
  - Creating conceptual schema and elaborate entity-relationship diagrams
  - Introducing concept of hierarchy and inheritance in ER model that leads to Extended E-R models
  - Converting an E-R, and EE-R schema into a relational schema
- Relational Database Systems:
  - Creating a relational database
  - Formulating SQL queries
  - Creating indices to improve performance
- Relational Algebra:
  - Query formulation
  - Simple query optimization by introduction of query tree and use of heuristics to optimize the performance of query
- Database Refinement and Normalization
  - Formal and informal guidelines for a good database design
  - Introduction to functional dependency and normalization
  - Different normal forms and schema refinement to avoid redundancy.

- Transaction Processing
    - Introduction to transaction processing and read write problems, deadlocks and recovery from deadlock.
- Web-Based Database Access:
    - Introduce different components and interfaces of a Web-based database system
    - Server-side scripting in PHP

Topic	Lecture Hours
Introduction, users of database, characteristics of database approach	2
Entity Relationship Model (E-R concept)	3
Extended E-R model to support class hierarchy and inheritance	3
Mapping ER and EER Models into Relational Model	4.5
Data Definition Language in SQL and Declarative Constraints	2
SQL and complex queries in SQL	4.5
Relational Algebra	4.5
Views	2
Query Processing and Optimization	2
Schema Refinement and Functional Dependencies	3
Normal Forms	3
Indexing (Single and Multilevel)	2

Learning Resources:

- Ramez Elmasri, Shamakant B. Navathe, Fundamental of Database Systems, , 6th Edition, Pearson Publishing (required).
- Raghu Ramakrishnan and Johannes Gehrke, *Database Management Systems*, McGraw Hill, ISBN 0-07-246563-8 (optional)
- Hugh E. Williams and David Lane, *Web Database Applications with PHP & MySQL,* O'Reilly (optional)
- MySQL DBMS, Apache 2 Web-server, PHP 5, CASE tools for data modeling, and Web-application generator

Course Learning Outcomes: At the completion of the course, students will be able to:

1. **Know** both traditional and structured paradigms to store data, and **Describe** the difference between a relational database and a flat file
2. **Model** real world data requirements by using conceptual model through entity, relationship (E-R diagram), able to Model more complex hierarchy and inheritance through specialization/generalization (Extended ER diagram)
3. **Learn** different alternatives to design a relational schema from those ER and EER diagram
4. **Generate** a relational database from a relational schema
5. **Create** multiple indices in a relational database, and **explain** when and why such indices are appropriate
6. **Learn** SQL DDL (Data Definition Language) for data definition, SQL DML (Data Manipulation Language) statements for data retrieving
7. **Formulate** simple queries in relational algebra by using projection, selection, product, and join operations as well some complex operations like division
8. **Describe** the components and interfaces of a Web-based database system

9. **Design** and **implement** a Web-based relational database system, using one or more scripting languages (e.g., PHP) and an open-source database development system (e.g., MySQL)