COURSE DESCRIPTION

Department and Course Number	CS422	**Course Coordinator**	Huiping Guo
Course Title	Principles of Database Systems	**Total Credits**	4

Current Catalog Description:

Normal forms, database system architecture, query optimization, file structures, transaction management, data warehouses, object-oriented databases, databases for e-commerce.

Textbook:

Garcia-Molina, Hector and Ullman, Jeffrey D. and Widom, Jennifer. *Database Systems: The Complete Book*, Prentice Hall, 2001.

References:

- Ramakrishnan, Raghu and Gehrke, Johannes. *Database Management Systems*, *3rd edition*, Mc.Graw Hill, 2002.
- Date, C. J. *An Introduction to Database Systems*, Mc.Graw Hill, 2005.
- Elmasri, Ramez and Navathe, Shamkant B. *Fundamentals of Database Systems*, Addison Wesley, 1994.
- Abiteboul, Serge and Hull, Richard and Vianu, Victor. *Foundations of Databases*, Addison Wesley, 1994.

Course Goals:

At the end of the course, students are able to

- Understand Entity-Relationship (ER), relational, and object-oriented data models.
- Be proficient in query languages including relational algebra, relational calculus, and SQL.
- Design and implement complex databases schemas using ER diagrams, normalization, integrity constraints, and advanced database system features such as stored procedures and triggers.
- Develop database applications using database client APIs such as embedded SQL, ODBC, and JDBC.
- Improve database performance using hardware, software, and query turning techniques.

These course goals contribute to the success of **Student Learning Outcomes 3, 5, and 6**.

Prerequisites by Topic:

- Basics of database systems and SQL
- Proficiency in at least one programming language such as C/C++ or Java
- Data structures and algorithms

Major Topics Covered in the Course:

- Introduction to database system architecture
- Entity-Relationship model and ER diagram
- Relational data model
- Object-oriented data model and Object-Relational Mapping (ORM)
- Functional dependencies and normalization
- Relational algebra
- Relational calculus
- Structured Query Language (SQL)
- Transactions, stored procedures, and triggers
- Database client API and application development
- Query tuning techniques
- Introduction to data warehousing, data mining, and XML data management.

Laboratory Projects (specify number of weeks on each):

Each week students either work on a homework assignment or complete a 2-hour in-class lab on selected topics:

- Week 1: Exercises on ER diagram.
- Week 2: Develop a database schema using ER diagram and translate it into relational model.
- Week 3: Normalize the database schema to 3NF, BCNF, or 4NF.
- Week 4: Construct and evaluate queries in relational algebra.
- Week 5: Construct and evaluate queries in datalog.
- Week 6: Construct and evaluate queries in SQL.
- Week 7: Complete the database schema with constraints, stored procedures, and triggers.

- Week 8-9: Develop a database application using embedded SQL, JDBC, or hibernate.
- Week 10: Exercises on SQL query tuning.

Estimate Curriculum Category Content (Quarter Hours)

Area	Core	Advanced	Area	Core	Advanced
Algorithms		.25	Data Structures		.25
Software Design		1.5	Prog. Languages		1.5
Comp. Arch.		0.5			

Oral and Written Communications:

Written documentation of software built in labs and homework assignments.

Social and Ethical Issues:

No significant component.

Theoretical Content:

- Entity-Relationship model and diagram (1 week)
- Relational model (1 week)
- Functional dependency and normalization (1 week)
- Relational algebra (1 week)
- Relational calculus (1 week)
- Introduction to algorithms for data warehousing, data mining, and XML data (1 week)

Problem Analysis:

Students are required to analyze the requirements of typical database applications, identify the data entities that are involved in the applications and their relationships to each other, and model these entities and relationships in relational database schemas.

Solution Design:

Students are guided through a complete database design process, which typically includes: initial data modeling in ER diagram, translation from ER diagram to relations, normalization, schema implementation, query design, client-side application development, and performance tuning.