

College of Software Engineering

Undergraduate Course Syllabus

Course ID	311038040	Course Name	Database Systems				
Course Attribute	<input checked="" type="checkbox"/> Compulsory <input type="checkbox"/> Selective	Course Language	<input checked="" type="checkbox"/> English <input type="checkbox"/> Chinese	Credit Hour	4	Period	80
Semester	<input type="checkbox"/> First Fall <input checked="" type="checkbox"/> Third Fall	<input type="checkbox"/> First Spring <input type="checkbox"/> Third Spring	<input type="checkbox"/> Second Fall <input type="checkbox"/> Fourth Fall	<input type="checkbox"/> Second Spring <input type="checkbox"/> Fourth Spring			
Instructors	Ruan Shuhua, Liang Gang, Lu Zhengtian						
Description	<p>This course introduces the crucial topics for database systems: Database Planning, Design, Implementation, and Administration. The course concentrates on centralized, relational database systems used for business data processing. The course covers the fundamental concepts and techniques of database systems: Database System Architectural Principles, Data Modeling, Fact-Finding Techniques, Theory and Methodology of Database Design, Query Processing, Storage Strategies, Query Optimization, and Transaction Management. The course then turns to on-going research in database systems, focusing on techniques that have recently been transferred from research to wide-spread commercial implementation. Finally, there is an introduction of new trends in database systems, focusing on applications for supporting decision-making in light of fully distributed and ubiquitously connected world consisting of all range data store and management.</p> <p>Students successfully completing this database fundamentals course will be equipped to handle small to medium size database projects. They will be able to design a database from scratch, design user interface and transaction, write queries against it, and build applications that use the database.</p>						
Prerequisites	<ul style="list-style-type: none"> ◆ Programming Fundamentals ◆ Introduction to Object-Oriented Programming ◆ Discrete Mathematics ◆ Data Structures and Algorithm/Course Design ◆ Computer Architecture ◆ Operating Systems 						
Textbook	Thomas M. Connolly, et al. <i>Database Systems: A practical approach to Design, Implementation, and Management</i> . 4th Edition ISBN: 0321210255 Addison-Wesley, 2004.						
Resource	<ol style="list-style-type: none"> 1. http://cs.scu.edu.cn/~ruanshuhua/index_7.htm 2. http://www.microsoft.com/learning/sql/2008/thankyou/default.aspx 3. http://www.ibm.com/developerworks/cn/db2/v9 4. http://www.oracle.com/database/index.html 5. http://www.postgresql.org 6. http://dev.mysql.com 7. http://java.sun.com 8. http://archive.apache.org/dist/jakarta/tomcat-5 9. http://www.eclipse.org/downloads/packages 						
Grading	<ul style="list-style-type: none"> ◆ Assignments and Class Participation: 10% ◆ Term Project: 20% ◆ Midterm: 20% ◆ Final Exam: 50% 						

	<p>1 Background (5 Periods)</p> <p>1.1 Introduction to Databases (3 Periods)</p> <p>1.1.1 History and Motivation for Information Systems</p> <p>1.1.2 Characteristics and Limitations of File-Based Systems *</p> <p>1.1.3 Motivation for Database Approach *</p> <p>1.1.4 Some Common Database Applications</p> <p>1.1.5 Meaning of the Term Database, Database Management Systems (DBMS), and Database Systems *</p> <p>1.1.6 Typical Functions of a DBMS</p> <p>1.1.7 Major Components of the DBMS Environment</p> <p>1.1.8 History of the Development of DBMSs</p> <p>1.1.9 Advantages and Disadvantages of DBMSs *</p> <p>1.1.10 Phone Number Programing on any one Language and on any one DBMS (2 Practice Periods) #1</p> <p>1.2 Database Environment (2 Periods)</p> <p>1.2.1 Personnel Involved in the Database Environment *</p> <p>1.2.2 The Concepts of Data Abstraction *</p> <p>1.2.3 The Three-Level ANSI-SPARC Database Architecture *</p> <p>1.2.4 Contents of External, Conceptual, and Internal Levels **</p> <p>1.2.5 The Concepts External, Conceptual, and Internal Schemas **</p> <p>1.2.6 Purpose of External/Conceptual and Conceptual/Internal Mappings **</p> <p>1.2.7 Meaning of Logical and Physical Data Independence *</p> <p>1.2.8 Database Languages DDL, DML and Distinction between Them *</p> <p>1.2.9 A Classification of Data Models *</p> <p>1.2.10 Purpose/Importance of Conceptual Modeling **</p> <p>1.2.11 Meaning of Client–Server Application Architecture and Advantages of This Type of Architecture for a DBMS</p>
<p>Topics</p>	<p>2 The Relational Model and Languages (20 Periods)</p> <p>2.1 The Relational Model (5 Periods)</p> <p>2.1.1 Terminology of Relational Model **</p> <p>2.1.2 How Tables Are Used to Represent Data</p> <p>2.1.3 Connection Between Mathematical Relations and Relations in the Relational Model</p> <p>2.1.4 Properties of Database Relations *</p> <p>2.1.5 How to Identify Candidate, Primary, and Foreign Keys **</p> <p>2.1.6 Meaning of Null *</p> <p>2.1.7 Meaning of Entity Integrity and Referential Integrity **</p> <p>2.1.8 Meaning of Enterprise Constraints *</p> <p>2.1.9 Purpose and Advantages of Views *</p> <p>2.1.10 Setting Integrity Constraints on any one DBMS (2 Practice Periods) #2</p> <p>2.2 Relational Algebra and Relational Calculus (5 Periods)</p> <p>2.2.1 Meaning of the Term Relational Completeness *</p> <p>2.2.2 How to Form Queries in Relational Algebra **</p> <p>2.2.3 How to Form Queries in Tuple Relational Calculus **</p> <p>2.2.4 How to Form Queries in Domain Relational Calculus **</p> <p>2.2.5 Categories of Relational DML</p> <p>2.3 SQL: Data Manipulation (5 Periods)</p> <p>2.3.1 Purpose and Importance of SQL *</p> <p>2.3.2 How to Retrieve Data From Database Using SELECT and **</p> <p>2.3.2.1 Use Compound WHERE Conditions</p> <p>2.3.2.2 Sort Query Results Using ORDER BY</p>

Topics	<ul style="list-style-type: none"> 2.3.2.3 Use Aggregate Functions (COUNT, SUM, AVG, MIN, MAX) 2.3.2.4 Group Data Using GROUP BY and HAVING 2.3.2.5 Use Subqueries 2.3.2.6 Join Tables Together 2.3.2.7 Perform Set Operations (UNION, INTERSECT, EXCEPT) 2.3.3 How to Update Database Using INSERT, UPDATE, and DELETE * 2.3.4 Using SQL Data Manipulation on any one DBMS (2 Practice Periods) #3 2.4 SQL: Data Definition (5 Periods) <ul style="list-style-type: none"> 2.4.1 Data Types Supported by SQL Standard * 2.4.2 Purpose of Integrity Enhancement Feature of SQL ** 2.4.3 How to Define Integrity Constraints Using SQL ** <ul style="list-style-type: none"> 2.4.3.1 Required Data 2.4.3.2 Domain Constraints 2.4.3.3 Entity Integrity Constraints 2.4.3.4 Referential Integrity Constraints 2.4.3.5 Enterprise Constraints 2.4.4 How to Use the Integrity Enhancement Feature in the CREATE and ALTER TABLE Statements * 2.4.5 How to Create and Delete Views Using SQL * 2.4.6 How the DBMS Performs Operations on Views * 2.4.7 Under What Conditions Views Are Updatable ** 2.4.8 Advantages and Disadvantages of Views 2.4.9 How the ISO Transaction Model Works * 2.4.10 How to Use the GRANT and REVOKE Statements as a Level of Security * 2.4.11 Using SQL Data Definition on any one DBMS (2 Practice Periods) #4
Topics	<ul style="list-style-type: none"> 3 Database Analysis and Design Techniques (20 Periods) <ul style="list-style-type: none"> 3.1 Database Planning, Design, and Administration (2 Periods) <ul style="list-style-type: none"> 3.1.1 Main Components of an Information System 3.1.2 Main Stages of Database Application Lifecycle * 3.1.3 Main Phases of Database Design: Conceptual, Logical, and Physical Design ** 3.1.4 How to Evaluate and Select a DBMS * 3.1.5 Distinction Between Data Administration and Database Administration * 3.1.6 Purpose and Tasks Associated with Data Administration and Database Administration 3.2 Fact-Finding Techniques (3 Periods) <ul style="list-style-type: none"> 3.2.1 Most Commonly Used Fact-Finding Techniques * 3.2.2 How to Use Each Fact-Finding Technique and the Advantages and Disadvantages of Each 3.2.3 Significance of Applying Fact-Finding Techniques to Early Stages of Database Application Lifecycle * 3.2.4 How to Use Fact-Finding Techniques in the Database Application Lifecycle 3.2.5 Types of Facts Collected in Each Stage of Database Application Lifecycle * 3.2.6 Types of Documentation Produced in Each Stage of Database Application Lifecycle * 3.2.7 Introduction of DreamHome 3.2.8 Discussing and confirming the Database Project Requirements in the Group (2 Practice Periods) #5 3.3 Entity-Relationship Modeling (6 Periods) <ul style="list-style-type: none"> 3.3.1 How to Use Entity-Relationship (ER) Modeling in Database Design 3.3.2 Basic Concepts Associated with ER Model ** 3.3.3 Structural Constraints in ER Model ** 3.3.4 Diagrammatic Technique for Displaying ER Model Using Unified Modeling Language

<p>Topics</p>	<p>(UML) *</p> <p>3.3.5 How to Identify and Resolve Problems with ER Models Called Connection Traps</p> <p>3.3.6 How to Build an ER Model From a Requirements Specification **</p> <p>3.3.7 Building an ER Model by Using Eclipse with UML (2 Practice Periods) #6</p> <p>3.4 Enhanced Entity-Relationship Modeling (4 Periods)</p> <p>3.4.1 Limitations of Basic Concepts of the ER Model and Requirements to Represent More Complex Applications Using Additional Data Modeling Concepts</p> <p>3.4.2 Most Useful Additional Data Modeling Concepts of Enhanced ER (EER) Model Called:</p> <p>3.4.3 Specialization/Generalization **</p> <p>3.4.4 Aggregation *</p> <p>3.4.5 Composition *</p> <p>3.4.6 A Diagrammatic Technique for Displaying Specialization/Generalization, Aggregation, and Composition in an EER Diagram Using UML</p> <p>3.4.7 Building an EER Model by Using Eclipse with UML (2 Practice Periods) #7</p> <p>3.5 Normalization (5 Periods)</p> <p>3.5.1 Problems Associated with Redundant Data *</p> <p>3.5.2 Identification of Various Types of Update Anomalies Such as Insertion, Deletion, and Modification Anomalies *</p> <p>3.5.3 How to Recognize Appropriateness or Quality of the Design of Relations *</p> <p>3.5.4 Purpose of Normalization *</p> <p>3.5.5 How to Use Functional Dependencies to Group Attributes into Relations being in a Known Normal Form **</p> <p>3.5.6 How to Undertake Process of Normalization **</p> <p>3.5.7 How to Identify Most Commonly Used Normal Forms, Namely 1NF, 2NF, 3NF, and Boyce–Codd Normal Form (BCNF) *</p> <p>3.5.8 How to Identify Fourth (4NF) and Fifth (5NF) Normal Forms</p> <p>3.5.9 An Actual Application Example for Normalization (2 Practice Periods) #8</p>
<p>Topics</p>	<p>4 Methodology (15 Periods)</p> <p>4.1 Conceptual Database Design (7 Periods)</p> <p>4.1.1 Purpose of Conceptual Database Design</p> <p>4.1.2 How to Decompose the Scope of the Design into Specific Users' Views of the Enterprise *</p> <p>4.1.3 How to Use ER Modeling to Build a Local Conceptual Data Model Based on Information Given in a View of the Enterprise **</p> <p>4.1.4 How to Validate Resultant Conceptual Model to Ensure it is a True and Accurate Representation of a View of the Enterprise *</p> <p>4.1.5 How to Document Process of Conceptual Database Design *</p> <p>4.1.6 End-Users Play an Integral Role Throughout Process of Conceptual Database Design</p> <p>4.1.7 Building ER or EER Model for the Database Project (4 Practice Periods) #9</p> <p>4.2 Logical Database Design (5 Periods)</p> <p>4.2.1 Purpose of Logical Database Design</p> <p>4.2.2 How to Remove Features From a Local Conceptual Model that are not Compatible with the Relational Model *</p> <p>4.2.3 How to Derive a Set of Relations From a Local Logical Data Model *</p> <p>4.2.4 How to Validate These Relations Using the Technique of Normalization *</p> <p>4.2.5 How to Validate a Logical Data Model to Ensure It Supports Required User Transactions *</p> <p>4.2.6 How to Merge Local Logical Data Models Based on Specific Views into a Global Logical Data Model of the Enterprise **</p> <p>4.2.7 How to Ensure that the Resultant Global Model is a True and Accurate Representation of</p>

- Enterprise *
- 4.2.8 Building Global Logical Model for the Database Project (2 Practice Periods) #10
- 4.3 Physical Database Design (3 Periods)
 - 4.3.1 Purpose of Physical Database Design
 - 4.3.2 How to Map the Logical Database Design to a Physical Database Design
 - 4.3.3 How to Design Base Relations for Target DBMS *
 - 4.3.4 How to Design Enterprise Constraints for Target DBMS *
 - 4.3.5 How to Estimate the Size of the Database *
 - 4.3.6 How to Select Appropriate File Organizations Based on Analysis of Transactions *
 - 4.3.7 When to Use Secondary Indexes to Improve Performance *
 - 4.3.8 How to Design Security Mechanisms to Satisfy User Requirements *
 - 4.3.9 Building the Project Database on the Selected DBMS (2 Practice Periods) #11
- 5 Selected Database Issues (5 Periods)**
 - 5.1 Transaction Management (5 Periods)
 - 5.1.1 Function and Importance of Transactions
 - 5.1.2 Properties of Transactions **
 - 5.1.3 Concurrency Control **
 - 5.1.3.1 Meaning of Serializability
 - 5.1.3.2 How Locking Can Ensure Serializability
 - 5.1.3.3 Deadlock and How It Can Be Resolved
 - 5.1.3.4 How Timestamping Can Ensure Serializability
 - 5.1.3.5 Optimistic Concurrency Control
 - 5.1.3.6 Granularity of Locking
 - 5.1.4 Recovery Control *
 - 5.1.4.1 Some Causes of Database Failure
 - 5.1.4.2 Purpose of Transaction Log File
 - 5.1.4.3 Purpose of Checkpointing
 - 5.1.4.4 How to Recover Following Database Failure
 - 5.1.5 Alternative Models for Long Duration Transactions
 - 5.1.6 The Database Project Application Design, Such as Transaction Design, User Interface Design (2 Practice Periods) #12
- 6 Current Trends (5 Periods)**
 - 6.1 Distributed DBMSs - Concepts and Design (5 Periods)
 - 6.1.1 What is a Distributed DBMS
 - 6.1.2 Advantages and Disadvantages of Distributed Databases
 - 6.1.3 Functions and Architecture for a DDBMS *
 - 6.1.4 Distributed Database Design *
 - 6.1.5 Levels of Transparency
 - 6.1.6 Comparison Criteria for DDBMSs *
 - 6.1.7 Building the Database System (2 Practice Periods) #13
- 7 Emerging Trends (10 Periods)**
 - 7.1 Web Technology and DBMSs (5 Periods)
 - 7.1.1 Basics of Internet, Web, HTTP, HTML, URLs
 - 7.1.2 Multi-Tier Client-Server Architecture *
 - 7.1.3 Advantages and Disadvantages of Web as a Database Platform *
 - 7.1.4 Approaches for Integrating Databases into Web
 - 7.1.4.1 Scripting Languages
 - 7.1.4.2 Common Gateway Interface (CGI)
 - 7.1.4.3 HTTP Cookies

	<p>7.1.4.4 Extending the Web Server 7.1.4.5 Java and JDBC, SQLJ, Servlets, and JSP 7.1.4.6 Microsoft Web Solution Platform: ASP and ADO 7.1.5 Building the Database System (2 Practice Periods) #14 7.2 The Concepts of Data Warehousing and Data marts (5 Periods) 7.2.1 How Data Warehousing Evolved * 7.2.2 Main Concepts and Benefits Associated with Data Warehousing 7.2.3 How Online Transaction Processing (OLTP) Systems Differ From Data Warehousing * 7.2.4 Problems Associated with Data Warehousing 7.2.5 Architecture and Main Components of a Data Warehousing * 7.2.6 Important Information Flows or Processes of a Data Warehouse ** 7.2.7 Main Tools and Technologies Associated with Data Warehousing 7.2.8 Issues Associated with the Integration of a Data Warehousing and the Importance of Managing Meta-Data * 7.2.9 Concept of a Data Mart and the Main Reasons for Implementing a Data Mart * 7.2.10 Advantages and Disadvantages of a Data Mart 7.2.11 Main Issues Associated with the Development and Management of Data Marts 7.2.12 Testing the Database System (2 Practice Periods) #15</p> <p>Notice: ♦ * for emphases ♦ ** for emphases and difficulty ♦ #x for the xth Practice Periods (Total 32 Practice Periods)</p>
<p>Tools & Environment</p>	<ul style="list-style-type: none"> ♦ Microsoft Windows Server 200X ♦ J2SDK 1.5 + Tomcat 5.0 + Eclipse 3.3 with UML ♦ Microsoft SQL SERVER 200X
<p>Projects</p>	<p>A Web-Based Database System Students will be able to plan, design, create, and maintain a Web-accessible database for a real estate company to keep track of their business, such as the parts, suppliers, and purchase orders, lease renewals, and so on. Included will be the tasks to write Web front-ends for users of the system, such as for the purchasing department to seek quotes, for the suppliers to place bids for work-orders, for the managers to monitor inventory, for client to maintenance personnel, for staff to manage property, and others.</p> <p>Phase 1 (5%) Goal: Background Procedure: Students select a database project, capture necessary facts to build the required database application by using fact-finding techniques, analyse information about the part of organization to be supported by the database application, and use this information to identify users' requirements for the new database system. Students must learn about the terminology, problems, opportunities, constraints, requirements, and priorities of the organization and the users of the new database system. Deliverables: The collection information and the report of requirements collection and analysis. Due on the fourth week in class. [Late submission policy – you lose 5% (of the maximum points) per day]</p> <p>Phase 2 (5%) Goal: Related Theories and Techniques Procedure: Students discuss related theories in the new database system, select a development platform and evaluate it, select an appropriate DBMS to support the new database application and evaluate it, and after singled out, are familiar with them as soon as possible. Deliverables: The collection information and the report of related theories and techniques. Due on the eighth week in class. [Late submission policy – you lose 5% (of the maximum</p>

<p>Projects</p>	<p>points) per day]</p> <p>Phase 3 (5%)</p> <p>Goal: The Database System Design</p> <p>Procedure: Students build data model by using the information in users' requirements specification, refine conceptual data model, map it to a logical data model, select storage structures and access methods used to achieve efficient access to data on a specific DBMS. At the same time, students define the scope and boundaries of the new database system and the major user views, define the new system and database role (such as Manager or Supervisor), and give the design of user interface and application programs that use and process the database.</p> <p>Deliverables: The report of the database system design.</p> <p>Due on the twelfth week in class. [Late submission policy – you lose 5% (of the maximum points) per day]</p> <p>Phase 4 (5%)</p> <p>Goal: The Database System Implementation and Evaluation</p> <p>Procedure: Students implement the database system design in phase 3, build working model of a database application and evaluate if the database and application programs appear to be working according to requirements, such as to identify features of a system that work well, or are inadequate, to suggest improvements or even new features, to clarify the users' requirements, to evaluate feasibility of a particular system design.</p> <p>Deliverables: The source code, the implementation state and result of the new database system, and the report of the database system implementation and evaluation.</p> <p>Due on the sixteenth week in class. [Late submission policy – you lose 5% (of the maximum points) per day]</p>
------------------------	--

Version: 2008-07-15

Author: Ruan Shuhua **Date:** 2008/07/15

Auditor: Mei Hong **Date:** 2008/07/15

Director: _____

Date: 2008/07/15