MODULE DESCRIPTION FORM

Module Information					
Module Title	**Database Systems Design and Development**			**Module Delivery**	
Module Type	**Core**			☒ **Lecture**	
Module Code	**IT2203**			☒ **Practical**	
ECTS Credits	**6**				
SWL (hr/sem)	**150**				
Module Level	UG2	**Semester of Delivery**	2		
Administering Department	Information Technology	**College**	College of Science		
Module Leader	Hussein Zaki Jassim Al-Mankoushi	**e-mail**	hussein@uowa.edu.iq		
Module Leader's Acad. Title	Asst. Lect	**Module Leader's Qualification**	M.Sc.		
Module Tutor	Hussein Zaki Jassim Al-Mankoushi	**e-mail**	hussein@uowa.edu.iq		
Peer Reviewer Name	Asst. Prof Haider Mohammed	**e-mail**	hayder.alghanami@uowa.edu.iq		
Scientific Committee Approval Date	2025-01-20	**Version Number**	1.0		

Relation with other Modules			
Prerequisite module	Principles of database systems	**Semester**	1
Co-requisites module	Principles of database systems	**Semester**	1

Department Head Approval

Dean of the College Approval

Module Aims, Learning Outcomes and Indicative Contents	
Module Aims	1. Provide a solid understanding of database concepts, principles, and best practices. 2. Familiarize students with the design, implementation, and management of databases. 3. Cover topics such as data modeling, normalization, and query optimization. 4. Develop practical skills in using database management systems and query languages. 5. Cultivate critical thinking and problem-solving abilities in the context of database design and administration. 6. Prepare students to apply their knowledge in real-world scenarios. 7. Equip students to contribute to effective database solutions in the IT industry.
Module Learning Outcomes	1. Understand the fundamental concepts and principles of databases, including data models, schemas, and normalization. 2. Demonstrate proficiency in designing, implementing, and managing databases using a database management system (DBMS). 3. Apply data modeling techniques to develop logical and physical database designs that meet specified requirements. 4. Construct and execute complex SQL queries to retrieve, update, and manipulate data stored in a database. 5. Evaluate and optimize query performance through the use of indexing, query tuning, and other optimization techniques. 6. Implement and enforce data integrity constraints, including entity relationships, referential integrity, and data validation rules. 7. Employ appropriate security measures to protect data and ensure database confidentiality, integrity, and availability. 8. Utilize backup and recovery procedures to safeguard data and restore databases in the event of failures or disasters.
Indicative Contents	Indicative content includes the following. 1. Advanced Database Design: Entity-Relationship Modeling: Extensions and enhancements to ER modeling, such as subtypes, supertypes, and specialization/generalization. Object-Oriented Data Modeling: Concepts of object-oriented databases and their modeling techniques, including inheritance, encapsulation, and polymorphism.

	UML Diagrams: Utilizing Unified Modeling Language (UML) to model databases, including class diagrams, object diagrams, and sequence diagrams. 2. Transaction Management and Concurrency Control: ACID Properties: Understanding the properties of atomicity, consistency, isolation, and durability in database transactions. Concurrency Control: Techniques for managing concurrent access to the database, including locking, timestamp-based protocols, and optimistic concurrency control. Recovery and Undo/Redo Logging: Mechanisms for ensuring database consistency in the face of failures, including log-based recovery and transaction rollback/commit. 3. Query Optimization and Execution: Query Processing: The stages involved in processing a database query, including parsing, optimization, and execution. Query Optimization: Techniques for selecting the most efficient query execution plan, such as cost-based optimization, join ordering, and index selection. Query Execution: Strategies for executing queries, including algorithms for sorting, joining, and aggregating data. 4. Data Storage and Indexing: File Structures: Storage structures for database files, such as heap files, sorted files, and hashed files. Indexing Techniques: Different indexing structures for efficient data retrieval, including B-trees, hash indexes, and bitmap indexes. Multi-Dimensional Data Structures: Introduction to data structures like R-trees and quad-trees for indexing spatial and multidimensional data. 5. Database Security and Authorization: Database Security: Concepts of access control, authentication, and authorization in database systems. Security Models: Different security models, such as discretionary access control (DAC), mandatory access control (MAC), and role-based access control (RBAC). Encryption and Auditing: Techniques for encrypting data and auditing database activities for security and compliance purposes.

Learning and Teaching Strategies	
Strategies	The learning and teaching strategies for studying the database subject in an IT department involve a balanced approach of theoretical understanding and practical application. Lectures, interactive discussions, and case studies provide the necessary theoretical foundation. Practical exercises, group work, and projects enable hands-on experience with database management systems. Workshops, demos, and industry examples offer real-world insights. Online resources, assessments, and feedback aid in reinforcing learning. Virtual labs and continuous learning emphasize practical skills development and staying updated with industry trends. These strategies ensure a comprehensive understanding of databases and their relevance in the IT field.

Student Workload (SWL)

Structured SWL (h/sem)	60	**Structured SWL (h/w)**	4
Unstructured SWL (h/sem)	87	**Unstructured SWL (h/w)**	6
Total SWL (h/sem)	147 + 3 final = 150		

Module Evaluation

		Time/Number	Weight (Marks)	Week Due	Relevant Learning Outcome
Formative assessment	**Quizzes**	5	10% (8)	2,3,4,5,6,7	All Outcome
	Onsite Assignments	5	10% (5)	All Weeks	All Outcome
	Lab	5	10% (15)	All Weeks	All Outcome
	Projects	1	10% (7)	All Weeks	All Outcome
	HW	5	10% (5)	All Weeks	All Outcome
	Report	1	10% (10)	13	
Summative assessment	**Midterm Exam**	2hr	10% (10)	7	
	Final Exam	3hr	50% (50)	16	
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus)

	Material Covered
Week 1	Introduction to databases: concepts, importance, and applications Relational database management systems (RDBMS)
Week 2	Overview: Introduction to Structured Query Language (SQL)
Week 3	Database design principles and data models

Week 4	Entity-Relationship (ER) modeling and ER diagrams
Week 5	Database constraints: primary key, foreign key
Week 6	Database constraints unique, and check constraints
Week 7	Database administration and security: user management, permissions, and access control
Week 8	Backup and recovery strategies for databases
Week 9	Indexing and query optimization techniques
Week 10	Transaction management and concurrency control in databases
Week 11	Relational model and relational calculus
Week 12	Relational model and relational algebra
Week 13	Transaction management and concurrency control in databases
Week 14	Transaction management and concurrency control in databases
Week 15	Database performance monitoring.
Week 16	**Preparatory week before the final Exam**

Delivery Plan (Weekly Lab. Syllabus)

	Material Covered
Week 1	Set up a database environment using a preferred database management system
Week 2	Create tables in the database based on the schema design
Week 3	Populate the tables with sample data to simulate real-world scenarios. Include a sufficient amount of data to perform meaningful queries.
Week 4	SELECT Queries: Write and execute basic SELECT queries to retrieve data from single tables.
Week 5	Use various clauses like WHERE, ORDER BY, and LIMIT to filter, sort, and limit the results.
Week 6	Practice different types of join operations
Week 7	NNER JOIN, LEFT JOIN, RIGHT JOIN, FULL JOIN) to combine data from multiple tables
Week 8	Create subqueries within SELECT statements to perform more complex queries
Week 9	Utilize aggregate functions (e.g., COUNT, SUM, AVG, MIN, MAX)
Week 10	Group data based on certain criteria using the GROUP BY clause. Apply the HAVING clause to filter the grouped data based on conditions.
Week 11	Practice functions like CONCAT, SUBSTRING, and LIKE
Week 12	Practice functions like CONCAT, SUBSTRING, and LIKE
Week 13	Practice writing queries with multiple levels of nested subqueries.

Week 14	Practice writing queries with multiple levels of nested subqueries.
Week 15	Implementation of an integrated database management project for each student

Learning and Teaching Resources

	Text	Available in the Library?
Required Texts	Elmasri, Ramez, and Shamkant Navathe. Fundamentals of database systems. AddisonWesley Publishing Company, 2018.	Yes
Recommended Texts	Database design, application and development.	No
Websites	http://www.sqlcourse.com/	

Grading Scheme

Group	Grade	Mark	Marks (%)	Definition
Success Group (50 - 100)	**A** - Excellent	Excellent	90 - 100	Outstanding Performance
	B - Very Good	Very Good	80 - 89	Above average with some errors
	C - Good	Good	70 - 79	Sound work with notable errors
	D - Satisfactory	Fair / Average	60 - 69	Fair but with major shortcomings
	E - Sufficient	Pass / Acceptable	50 - 59	Work meets minimum criteria
Fail Group (0 – 49)	**FX** – Fail	Fail (Pending)	(45-49)	More work required but credit awarded
	F – Fail	Fail	(0-44)	Considerable amount of work required

Note: Marks Decimal places above or below 0.5 will be rounded to the higher or lower full mark (for example a mark of 54.5 will be rounded to 55, whereas a mark of 54.4 will be rounded to 54. The University has a policy NOT to condone "near-pass fails" so the only adjustment to marks awarded by the original marker(s) will be the automatic rounding outlined above.