

Caracterização da Unidade Curricular / Characterization of the Curricular Unit

Designação da Unidade Curricular (UC) / Title of Curricular Unit (CU): Bases de Dados / Databases

Área científica da UC / CU Scientific Area: Informática / Computer Science

Semestre / Semester: 3º

Número de créditos ECTS / Number of ECTS credits: 6

Carga horária por tipologia de horas / Workload by type of hours: TP: 22,5; PL: 45; OT: 9; O: 13,5

Carga letiva semanal / Weekly letive charge: 4,5h

Objetivos de aprendizagem (conhecimentos, aptidões e competências a desenvolver pelos estudantes):

Pretende-se que, no final do semestre, os alunos sejam capazes de:

- Perceber e explicar os conceitos fundamentais da teoria das Bases de Dados (BD) e a sua aplicação prática;
- Definição de um modelo relacional de dados, de acordo com os princípios gerais das bases de dados, usando diferentes tipos de notações e linguagens (E-A, UML);
- Usar álgebra relacional para formular perguntas a bases de dados;
- Desenhar uma base de dados normalizada;
- Interrogar uma base de dados usando SQL, e implementar as operações de definição e manipulação de dados;
- Efectuar e gerir transacções e mecanismos de recuperação que garantam a integridade dos dados;
- Obter conhecimentos de implementação de uma BD em SQL Server e Oracle;
- Implementar pequenos desenvolvimentos de programas em PL/SQL;
- Projectar e implementar a estrutura de dados de uma BD num Sistema de Gestão de Bases de Dados (SGBD) relacional;
- Conhecer as diversas tipologias de bases de dados relacionais e NoSQL;
- Usar as ferramentas disponibilizadas por um SGBD comercial.

Intended learning outcomes (knowledge, skills and competences to be developed by the students):

It is intended that, by the end of the semester, students will be able to:

- Understand and explain the fundamental concepts of Database (DB) theory and their practical application;
- Define a relational data model, according to the general principles of databases, using different types of notations and languages (E-A, UML);
- Using relational algebra to formulate queries to databases;
- Design a normalized database;
- Interrogate a database using SQL, and implement data definition and manipulation operations;

- Perform and manage transactions and recovery mechanisms that ensure data integrity;
- Get knowledge of implementing a DB in SQL Server and Oracle;
- Implement small program developments in PL/SQL;
- Design and implement the data structure of a DB in a Relational Database Management System (DBMS);
- Know the various types of relational and NoSQL databases;
- Use the tools provided by a commercial DBMS.

Conteúdos programáticos:

1. Fundamentos de Bases de Dados

- 1.1. Independência lógica e física dos dados
- 1.2. Arquitetura de um SGBD
- 1.3. Utilização empresarial de Base de Dados

2. Modelo Relacional

- 2.1. Conceito de relação e esquema relacional
- 2.2. Associações entre entidades e hierarquias ISA
- 2.3. Regras de transformação do modelo E-A para o modelo Relacional.

3. Modelização de dados usando a UML

- 3.1. Diagramas UML estáticos e dinâmicos
- 3.2. Mapeamento de classes, associações e agregações.

4. Álgebra relacional

- 4.1. Teoria dos conjuntos
- 4.2. Interrogações em álgebra relacional básicas e complexas
- 4.3. Passagem de álgebra relacional para linguagem SQL
- 4.4. Otimização de interrogações em álgebra relacional

5. Normalização

- 5.1. Dependências funcionais (DF) e sua decomposição
- 5.2. Formas Normais.
- 5.3. Refinamento de um esquema relacional com base nas DF.

6. Linguagem SQL (Structured Query Language)

- 6.1. Interrogações em SQL

- 6.2. Utilização de gatilhos
- 6.3. Implementação de SGBD com recurso a SQL Server e Oracle

7. Linguagem PL/SQL (Procedural Language/Structured Query Language)

- 7.1. Variáveis e instruções em PL/SQL
- 7.2. Cursores implícitos e explícitos
- 7.3. Considerações de design e desempenho

8. Transações, Controlo da Concorrência e Segurança

- 8.1. Conceito de transação e concorrência
- 8.2. Algoritmos de recuperação.
- 8.3. Considerações sobre segurança da BD e SQL

9. Índices e Desempenho

- 9.1. Classificação de índices
- 9.2. Otimização de queries
- 9.3. Tuning

10. NoSQL Databases

- 10.1. ACID vs. BASE
- 10.2. Tipologia de NoSQL (key-value, column, graph, document)
- 10.3. Cenários de utilização

Syllabus:

1. Database Fundamentals

- 1.1. Logical and physical independence of data.
- 1.2. DBMS architecture.
- 1.3. Business use of databases.

2. Relational Model

- 2.1. Relational concept and relational schema.
- 2.2. Associations between entities and ISA hierarchies
- 2.3. Transformation rules from E-A model to Relational model.

3. Data modeling using UML

- 3.1. Static and dynamic UML diagrams
- 3.2. Class mapping, associations and aggregations.

Sem Validade
Administrativa

4. Relational Algebra

- 4.1. Set theory
- 4.2. Basic and complex relational algebra queries
- 4.3. Switching from relational algebra to SQL language
- 4.4. Optimization of relational algebra queries

5. Normalization

- 5.1. Functional dependencies (FD) and their decomposition
- 5.2. Normal forms
- 5.3. Refinement of a relational schema based on DFs.

6. SQL Language (Structured Query Language)

- 6.1. Queries in SQL
- 6.2. Use of triggers
- 6.3. Implementation of DBMS using SQL Server and Oracle

7. PL/SQL Language (Procedural Language/Structured Query Language)

- 7.1. Variables and statements in PL/SQL
- 7.2. Implicit and explicit cursors
- 7.3. Design and performance considerations

8. Transactions, Concurrency Control and Security

- 8.1. Transaction and concurrency concept
- 8.2. Recovery algorithms
- 8.3. DB and SQL security considerations

9. Indexes and Performance

- 9.1. Index Classification
- 9.2. Query optimization
- 9.3. Tuning

10. NoSQL Databases

- 10.1. ACID vs.
- 10.2. NoSQL typology (key-value, column, graph, document)
- 10.3. Usage scenarios