



---

# DATABASES AND WEB PROGRAMMING M - Z

12 CFU - 1° and 2° Semester

## Teaching Staff

**CONCETTO SPAMPINATO** - Module DATABASES - ING-INF/05 - 6 CFU

**Email:** cspampin@dieei.unict.it

**Office:** DIEEI, Plesso 13, stanza 8, Cittadella Universitaria

**Phone:** 095/7382057

**Office Hours:** Su prenotazione via email

**CONCETTO SPAMPINATO** - Module Web Programming - ING-INF/05 - 6 CFU

**CARMELO PINO** - Module Web Programming

**Email:** cpino@dieei.unict.it

**Office:** Openlab, Via S.Sofia 102

**Phone:** 095-7387902

**Office Hours:** da concordare per e-mail

---

## LEARNING OBJECTIVES

### ▪ DATABASES

The course covers the fundamental concepts of management and design of database systems. Topics include data models (ER and relational); query languages (SQL); implementation techniques of database management systems (index structures and nd query processing); noSQL databases.

The learning objectives of the course, expressed according to the Dublin descriptors, are:

### Knowledge and understanding

- To understand the main concepts of management and design of database systems
- To understand concepts and tools for querying databases of different nature (therefore with structured and unstructured data) and at different scales
- To understand the role and the impact that data and their correct modeling may have on software applications using them

### Applying knowledge and understanding

- To be able to design, starting from the requirements, a relational (and non) relational database, from a conceptual, logical and physical point of view.
- To be able to understand and use the SQL language to manage data and perform efficient queries.
- To be able to create and query unstructured databases (NoSQL) by manipulating JSON

documents

### **Making judgements**

- Analyzing functional and non-functional requirements
- Understanding the implications of design choices related to response times and alternative implementations
- Understanding the implications of design choices related to change management and maintenance of alternative software stacks
- Understanding the performance issues associated to alternative DBMS

### **Communication skills**

- Preparing database specification documents, addressing the specificity of conceptual and logical modelling
- Explaining the motivations behind specific design choices

### **Learning skills**

- Learning how to use the technologies for the development of databases and their scalability
- Learning how to compare alternative projects schemes from different perspectives
- Learning to recognize design principles underlying alternative modelling development strategies

### **▪ Web Programming**

The module deals with the methodological tools for the design of modern web-based information systems. In particular the module covers concepts for designing and developing the back-end (server-side programming) and front-end (client-side programming) of a web application. Particular emphasis will be given to the development of "responsive" front-ends. The module covers also design and implementation techniques based on the Model-View-Controller (MVC) architectural pattern.

The module will finally presents elements on programming techniques for the implementation of interactive GUIs (e.g., React).

The learning objectives of the course, expressed according to the Dublin descriptors, are:

### **Knowledge and understanding**

- Understanding the main design concepts of web information systems
- Understanding concepts and tools for the development of efficient back-end components and function front-end interfaces, that satisfy project requirements
- Understanding concepts and tools that support separation of responsibilities, roles and code reuse in the development of a web application

### **Applying knowledge and understanding**

- Designing and implementing a multi-platform and multi-device web application, on both the client side and the server side
- Implementing web applications that handle data interaction (DBMS) and the integration with other web systems (REST API)
- Designing and implementing the user experience of a web application

## **Making judgements**

- Analyzing functional and non-functional requirements
- Understanding the implications of design choices related to alternative software stacks, both server-side and client-side
- Understanding the implications of design choices related to change management and maintenance of alternative software stacks
- Understanding the performance issues associated to alternative software stacks

## **Communication skills**

- Preparing web application specification documents, addressing the specificity of web applications
- Explaining the motivations behind specific design choices

## **Learning skills**

- Learning how to use the technologies for the development of complex web applications
- Learning how to compare alternative projects schemes from different perspectives
- Learning to recognize design principles shared by apparently different architectural styles, data transmission mechanisms and development toolkits

## **COURSE STRUCTURE**

### **▪ DATABASES**

Lectures, hands-on exercises, laboratory and seminars.

Should teaching be carried out in mixed mode or remotely, it may be necessary to introduce changes with respect to previous statements, in line with the programme planned and outlined in the syllabus.

### **▪ Web Programming**

Lectures, hands-on exercises, laboratory and seminars.

Should teaching be carried out in mixed mode or remotely, it may be necessary to introduce changes with respect to previous statements, in line with the programme planned and outlined in the syllabus.

---

## **DETAILED COURSE CONTENT**

### **▪ DATABASES**

#### **Models and Languages for Database Management**

- Fundamentals of Database Management Systems (DBMS)
- Relational databases:
  - Relational Model: basic concepts, integrity\* constraints and keys\*.
  - SQL language: data definition\*, data modification\*, queries\*, views\*, transactions\*.
  - Advanced SQL: triggers\*, procedures\* and assertions

- An example of relational DMBS: MySQL.
- Database NoSQL:
  - JSON\*
  - ACID vs BASE\*
  - "Scheme-less"\* models
  - NoSQL DB types\*: key-value, document-based, column-based and graph-based
  - CAP Theorem\*
  - Overview of NoSQL frameworks: MongoDB, Redis, Cassandra, HBase, Neo4j
- MongoDB\*: Selectors, Search, Sort, Pipeline, Grouping

### **Design of databases**

- Conceptual design: E/R model\*, conceptual design strategies\*, quality verification.
- Logical design: restructuring of the E / R schemes,\* translation into the relational model\*.
- Database normalization: normal forms\*

\* Topics to know and understand to pass the exam.

### **▪ Web Programming**

#### **Foundations of web programming and server-side development**

- HTML, CSS and Document Object Model (DOM)
- PHP language
- HTML and PHP: forms, sessions, cookies
- DBMS interaction with PHP

#### **Web information Systems design and client-side application development**

- JavaScript: forms, functions, object model (window, document, form objects), events
- Page manipulation: CSS selectors, position selectors and operations with objects (DOM manipulation)
- AJAX: asynchronous requests, text/HTML/JSON responses
- The MVC (Model-View-Controller) pattern: examples of software design within an MVC-based framework

#### **Advanced Web Programming Techniques**

- Design and implementazione of interactive GUI (e.g., React)

## **TEXTBOOK INFORMATION**

### **▪ DATABASES**

1. Basi di dati - 4/ed, Modelli e linguaggi di interrogazione. Di: Paolo Atzeni, Stefano Ceri, Piero Fraternali, Stefano Paraboschi e Riccardo Torlone, Casa editrice: Mc-Graw-Hill
2. Teaching materials provided by the instructor

### **▪ Web Programming**

