



UNIVERSITÀ  
DEGLI STUDI DI MILANO-BICOCCA

## COURSE SYLLABUS

### Design of Web Applications

2627-3-E3101Q148

---

#### Aims

The course provides the fundamental knowledge required for the analysis, design, and development of modern Web applications, with particular emphasis on the interaction between client-side applications and Web services. By the end of the course, students will understand the architectural principles of the Web, the core technologies underlying front-end development, and the mechanisms used to communicate with distributed Web services. They will acquire the skills needed to design and develop interactive Web interfaces, integrate client applications with Web APIs, and use modern tools for development, testing, and debugging.

The acquired competencies will enable students to contribute to the development of distributed Web applications, fostering a solid understanding of both design and implementation aspects and promoting independent judgment in making architectural and implementation decisions.

#### Contents

The course introduces the principles of designing Web applications based on client-server and service-oriented architectures. It covers the fundamental technologies of the Web, the interaction models between browsers and servers, and the techniques used to develop dynamic user interfaces.

A significant part of the course is devoted to laboratory activities, during which students will develop Web applications using HTML5, CSS3, and JavaScript/React, integrating Web services through HTTP APIs and JSON data exchange formats. The course also introduces Node.js and the Express framework, as well as development environments and tools for debugging, testing, and project management.

#### Detailed program

1. Web Application Architecture: evolution of the Web; client-server architecture; HTTP protocol; REST API

principles; Single Page Applications (SPAs); Web APIs.

2. Core Web Technologies: HTML5; CSS3; modern JavaScript (ES6+); JavaScript modules; JSON data format.
3. Client-Side Programming: Document Object Model (DOM); event handling; asynchronous programming with Promises and async/await; Fetch API; communication with RESTful services.
4. Web Application Development with React: component-based architecture; JSX; declarative rendering; props; component composition; state management; Hooks; forms and event handling; side effects; introduction to the Context API.
5. React Application Design: client-side routing; integration with RESTful services; error handling; authentication concepts; application organization and architecture.
6. Development and Deployment Tools: Node.js; Express; Vite; debugging tools; testing; client-side security; build and deployment.

## Prerequisites

Basic knowledge of programming in Java or C++, computer networks, and distributed systems is recommended.

## Teaching form

The course is delivered primarily through in-person classes.

The teaching activities include different instructional approaches:

Lectures (Direct Teaching – DE): classroom lectures introducing the principles, models, and technologies for the design and development of modern Web applications. The lectures cover the main programming languages, frameworks, development tools, and methodologies used in Web application development.

Interactive Teaching (DI): hands-on laboratory sessions and practical exercises focused on the design, development, and testing of Web applications. Through the incremental development of one or more Web applications, students apply the concepts presented during the lectures, addressing the main phases of the software development process, including requirements analysis, design, implementation, Web API integration, testing, and debugging.

Interactive Teaching (DI): learning support activities delivered through the e-learning platform, which provides lecture materials, code examples, exercises, discussion forums, and additional learning resources, fostering collaborative learning and independent study.

Part of the teaching material is also made available in audiovisual format through the e-learning platform, in accordance with the University guidelines.

*Language of instruction: Italian*

## Textbook and teaching resource

There is no single required textbook for the course.

The teaching material includes lecture notes, lecture slides, code examples, technical documentation, tutorials, and additional learning resources, which will be made available or referenced through the course e-learning platform.

The official documentation of the programming languages, technologies, frameworks, and development tools used during the course is considered an integral part of the study material and will be appropriately referenced throughout the lectures.

## **Semester**

Second semester

## **Assessment method**

Student learning is assessed through a written examination, which may be complemented by an oral examination at the instructor's discretion. During the semester, mid-term assessments may also be offered. Successfully completing these assessments may replace all or part of the final written examination, according to the procedures communicated at the beginning of the course.

The written examination is designed to assess the achievement of the course learning objectives and includes theoretical questions, Web application design and development exercises, and problem-solving tasks covering the topics addressed during the course.

Laboratory activities involve the development of Web applications and practical programming assignments, allowing students to consolidate the knowledge and skills acquired during the course. Laboratory assignments may contribute up to 2 bonus points toward the final grade.

An optional individual or group project is also available, allowing students to further explore selected course topics. The project may contribute up to 4 bonus points toward the final grade.

Assessment is based on the correctness of the proposed solutions, the ability to apply the acquired knowledge to the design and development of Web applications, the quality of the produced code, the appropriate use of the technologies and development tools presented during the course, as well as the clarity of the presentation and the ability to justify design decisions.

The final grade is expressed on a 30-point scale and is determined by the written examination (or the mid-term assessments), possibly complemented by the oral examination and any bonus points obtained through laboratory activities and the optional project.

## **Office hours**

prof. De Paoli: Wednesday from 10:00 to 12:00 or by appointment by writing to [flavio.depaoli@unimib.it](mailto:flavio.depaoli@unimib.it)

Questions and discussions on teaching topics can be posed using the forums in e-learning.

**Sustainable Development Goals**

---