

Web Development Workshop

Overview

These workshops covers frontend and backend development using React, Spring Boot with SQL, and Node.js with NoSQL. Each section is conducted by a different instructor.

1. React (Frontend)

Instructor: LEGUEFCHE Alaa Eddine

Key Elements :

HTML, CSS, JS REACT : hooks, routing, state mangement, form handling Figma Shadcn + Tailwind Libraries

Objectives :

- Understand frontend development.
- Learn how to build static webpages using HTML, CSS, and JavaScript.
- Learn how to use React for static then dynamic webpages using best practices
- Learn to design using Figma and then implement React (Use Shadcn Library).

Frontend Development Training Schedule

Session 1: Web Development Basics & Building a Static Webpage (HTML, CSS, JS)

- Introduction to **Frontend, Backend, and Full-Stack Development**
- How browsers render **HTML, CSS, and JavaScript**
- Understanding **static vs. dynamic webpages**
- How APIs (REST) provide dynamic data to web applications
- Create a basic webpage and link external **CSS & JavaScript files**
- Build a **static To-Do list** with simple styling
- Implement an **onclick event** to trigger an alert when a To-Do item is clicked
- Introduction to **DevTools** for debugging

Session 2: Introduction to React

- Why use **frontend frameworks**? Why **React**?
- Setting up a React project using **Vite** & present key react concepts
- Convert the previously built **static webpage into a React app**
- Use **Redux (RTK Query)** to fetch **dynamic To-Do data** from an API (dummy JSON) manage loading states and errors

Session 3: React CRUD To-Do App (After Wassim Oubaziz's Session & Mohamed Talbi's Session)

- Gather requirements and **design UI in Figma** using Shadcn UI

- Convert Figma designs into a **functional React app**
 - Implement **CRUD operations** (Create, Read, Update, Delete)
 - Introduction to **authentication in React** and protected routes
 - Build **high-quality forms with validation and error handling**
 - Redirect users to the **To-Do app after successful login**
-