



Technology + Knowledge = Innovation

 Via Copenhagen, 10 Roma

 www.vivasoft.it

 info@vivasoft.it

Vivasoft

Consulting & Training



Vivasoft is a leading company in the technology sector, specializing in offering innovative solutions based on Microsoft. With years of experience in the market, we are proud to be certified Microsoft partners, committed to supporting the digital transformation of businesses. We provide a comprehensive range of Microsoft products and highly qualified training courses, designed to help companies optimize their processes, enhance productivity, and acquire advanced skills in the world of technology.



Module 1: Introduction to React

What is React and why use it?

Installing and setting up a React development environment with create-react-app

Structure of a React app: components, JSX, and the virtual DOM

Writing the first React component

Rendering components on a web page

Basic JSX (JavaScript XML) concepts

Using ReactDOM.render

Module 2: React Fundamentals - Components and Props

What is a React component and how does it work?

Creating functional and class-based components

Differences between functional and class-based components

Passing data between components using props

Conditional rendering of components

Handling events in React

Module 3: State and Component Lifecycle

Introduction to state in React

Using state in functional components with the useState hook

Component lifecycle and using hooks like useEffect and componentDidMount

Managing side effects with useEffect

Hands-on exercises on state management and handling side effects



Module 4: Event Handling and Form in React

Handling events in React (click, input, submit, etc.)
Passing data between components using **props** and **state**
Managing forms in React: creating forms and handling user input
Form validation and error handling
Submitting form data to an API
Building a form management application

Module 5: Working with Lists and Keys

Dynamically rendering lists of elements in React
Using **map()** to iterate over arrays
The importance of **keys** in lists
Adding, modifying, and removing elements from a list
Creating a **component to manage a dynamic list**

Module 6: Styling in React

Inline styling with JavaScript objects
Using **CSS files** for components
Styling with libraries like **styled-components** and **emotion**
Introduction to **CSS Modules** in React
Customizing components with **Tailwind CSS**



Module 7: Global State Management - Context API and Redux

Introduction to **React Context API**

Creating a global context to share state between multiple components

Introduction to **Redux** for global state management

Actions, reducers, and store in Redux

Integrating Redux with React and handling side effects with **redux-thunk**

Hands-on exercises with Context API and Redux

Module 8: Routing in React with React Router

What is **React Router** and how does it work?

Creating routes with `<Route>` and `<Link>`

Programmatic navigation with `useHistory` and `useNavigate`

Handling **dynamic route parameters**

Authentication and **protected routes**

Rendering components based on routes (**Lazy loading**)

Module 9: Consuming APIs in React

Introduction to `fetch()` and `axios` for making HTTP requests

Managing **loading states** and handling errors

Working with **JSON data** and rendering results

Authentication with **JWT tokens** and session management

Building a small application that consumes an external API (e.g., weather API or product API)



Module 10: Testing in React

Introduction to testing with Jest and React Testing Library
Writing tests for React components
Simulating user events and interactions
Testing component rendering and internal logic
Mocking and testing components that interact with external APIs
Writing tests for state management and side effects

Module 11: Performance and Optimization

Optimization principles in React: conditional rendering and memoization
Using React.memo() and useMemo() to prevent unnecessary re-renders
Optimizing list rendering with virtualization
Lazy loading components with React.lazy() and Suspense
Analyzing performance using React DevTools

Module 12: Final Project

Planning and designing a complete React application
Developing the project using components, routing, API integration, and state management
Testing and debugging
Performance optimization
Presenting and discussing the final project