Micro Frontends with Vue

Audience Course Micro Frontends with Vue

This course is designed for Vue.js developers, UI engineers, and technical leads who want to explore how micro frontend patterns can be applied to modern web applications.

Prerequisites Course Micro Frontends with Vue

Participants should have solid knowledge of JavaScript and Vue development. Understanding of modular design, web components and tooling like Vue CLI or Vite is recommended.

Realization Training Micro Frontends with Vue

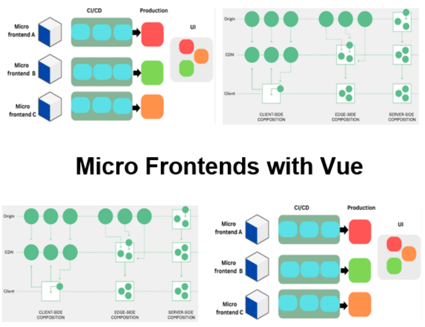
The course combines conceptual explanations with guided exercises and coding workshops. Participants work on cases that simulate challenges encountered in real-world projects.

Micro Frontends with Vue Certificate

Upon completion, participants receive a certificate of participation in Micro Frontends with Vue.

Duration: 3 days **Price: € 2299**

Open Schedule



Content Micro Frontends with Vue

The course Micro Frontends with Vue from SpiralTrain shows you how to build flexible and scalable frontend architectures using Vue.js. You will discover how to split large applications into smaller micro frontends, connect them effectively, and leverage Vue's ecosystem for smooth integration.

Introduction

The course Micro Frontends with Vue begins with an introduction to the concept of micro frontends and their differences from monolithic applications. Use cases, and core principles are explained with examples from industry practice.

Vue Recap

Next, participants review essential Vue concepts including components, directives, reactivity, props and events, routing with Vue Router, and the Composition API. This recap ensures a solid basis for building micro frontends.

Architecture

This module explores architecture patterns and design approaches for Vue micro frontends. Topics include component composition, event bus communication, routing strategies, shell applications, lazy loading, and error boundaries.

Module Federation

Participants learn how to use Webpack Module Federation with Vue. Covered are host and remote apps, dynamic imports, shared libraries, version handling, runtime integration, and best practices for setup and error handling.

Integration

Integration is discussed with focus on UI composition, shared navigation, cross-app routing, and service integration. Special attention is given to authentication, authorization, and performance monitoring in real-world scenarios.

State Management

This part addresses state management in Vue. Subjects include local state, props and events, Vuex with modules, and newer alternatives like Pinia. Cross-app state sharing and debugging tools are also reviewed.

Deployment

Deployment strategies cover CI/CD pipelines, independent releases, and environment configurations. The module also includes containerization with Docker, deployment on Kubernetes, cloud hosting options, and rollback planning.

Testing

Participants learn about testing micro frontends with Vue. Unit testing with Vue Test Utils and Jest, integration testing, and end-to-end testing with Cypress are explained, alongside automation and accessibility checks.

Advanced Topics

The course concludes with advanced subjects such as security, authentication and authorization patterns, performance optimization, error handling, and caching strategies. Future trends are discussed, and a hands-on project wraps up the training.

Modules Micro Frontends with Vue

Module 1 : Introduction	**Module 2 : Vue Recap**	**Module 3 : Architecture**
Micro Frontends Overview Monolith vs Micro Frontends Advantages and Limitations Use Cases Core Principles Team Autonomy Deployment Flexibility Integration Styles Scaling Applications Industry Examples	Vue Fundamentals Vue Components Directives Basics Reactivity System Props and Events Vue Router Intro Vue CLI Data Binding Lifecycle Hooks Composition API	Architecture Patterns Component Composition Shared vs Isolated State Event Bus Pattern Routing Strategies Shell Application Cross-App Communication Lazy Loading Modules Error Boundaries Resilience Patterns
Module 4 : Module Federation	**Module 5 : Integration**	**Module 6 : State Management**
Webpack Federation Setup Host and Remote Apps Dynamic Imports Shared Libraries Exposing Components Version Handling Runtime Integration Error Handling Configuration Files Best Practices	UI Composition Shared Navigation Authentication Integration Authorization Handling Micro Frontend Shell Cross-App Routing Service Integration Performance Monitoring Testing Integration Real-World Scenarios	State Basics in Vue Local Component State Props and Events Vuex Overview Vuex Store Setup Modules in Vuex Cross-App State Sharing Async State Handling Pinia Introduction Debugging Tools
Module 7 : Deployment	**Module 8 : Testing**	**Module 9 : Advanced Topics**
Deployment Strategies Independent Deployments CI/CD Pipelines Environment Configurations Docker with Vue Kubernetes Deployments Cloud Hosting Options Version Control Rollback Strategies Monitoring Solutions	Unit Testing Vue Vue Test Utils Jest Framework Integration Testing End-to-End Testing Cypress Framework Contract Testing Test Automation Performance Testing Accessibility Testing	Security Concerns Authentication Patterns Authorization Patterns Performance Optimization Error Handling Accessibility Concerns Caching Strategies Future Trends Case Studies Hands-On Project