Updated 06/06/2025

Sign up

Vue.js 3 training

3 days (21 hours)

Presentation

Discover our "Vue.js - Web Application Development" training course: 3 intensive days to master dynamic Web application development and create your own Single Page Applications (SPA) with Vue.js. This modern program using the new TypeScript language (an evolution of JS) will give you an in-depth understanding of the framework's key principles and best practices.

Thanks to an active, hands-on approach, you'll quickly progress from the initial setup with Node.js, npm and Vite, to advanced skills including indispensable tools such as Pinia for global state management and Vue Router for routing. You'll also learn how to take full advantage of Vue 3's Composition API, structure your projects into modular, reusable components, and integrate robust, accessible forms with FormKit and VeeValidate. To guarantee optimal project quality, you'll carry out automated tests (unit tests with Vitest, end-to-end tests with Cypress), while mastering performance optimization and monitoring via Vue DevTools and Sentry. Finally, you'll deploy your application seamlessly using modern solutions such as Netlify or Vercel, while integrating the latest Vue 3.4 and 3.5 innovations.

As with all our training courses, this one will introduce you to the latest version of Vue (at the time of writing: Vuejs 3.5).

Objectives

- Master the developer environment and the TypeScript language
- Understanding the key principles and best practices of Vue.js
- Learn about the essential tools for developing dynamic web applications with Vue.js
- Developing a SPA with Vue.js
- Performing tests

Target audience

- Front-end developers
- Tech Lead or Lead Dev
- Web project manager
- Technical architects
- Design engineer

Prerequisites

- Knowledge of HTML and CSS
- Basic knowledge of JavaScript
- Test My Knowledge

Technical requirements

- Have Node.js installed

Vue.js training program

Day 1: Vue.js fundamentals

Modern development environment

- Node.js & npm: Installation and dependency management
- Quick: Rapid development server and zero configuration
- VS Code & Extensions: Tools for improving productivity
- Project structure View 3: File organization and conventions
- Practical workshop: Initialize a Vue 3 project with Vite and configure the development environment.

Important concepts before you start

- The different web applications
- HTTP, Rest and JSON communication
- The Component paradigm

TypeScript and JS

- JS vs TypeScript
- Introduction to TypeScript and its use in Vuejs
- Basic types and variables
- Interfaces and classes
- Optional functions and parameters

Introduction to Vue.js

- Vue 3 Composition API: Understanding the new composition approach
- Render function
- Responsiveness: Data and virtual DOM linking mechanisms
- Basic syntax: v-if, v-for, v-model directives
- Templates : Using {{ }} moustache expressions
- Practical workshop: Create a simple component displaying dynamic data.

Vue.js templating language

- Interpolation
- Attribute binding
- Event binding
- Computed properties
- Using modifiers

Components and communication

- Single File Components (SFC): Structure and benefits
- Props & Events : Communication between parent and child components
- Slots: Dynamic content insertion
- Component lifecycle : Hooks onMounted, onUpdated
- Hands-on workshop: Developing a reusable film card with personalized props and events.

Day 2: SPA applications and status management

Routing with Vue Router

- Vue-Router introduction and usage
- Route configuration : Define paths and associated components
- Navigation: Using <RouterLink> and <RouterView>.
- Dynamic routes: Parameters and redirections
- Nested routes
- Lazy Loading: Delayed component loading
- Practical workshop: Implementing navigation between the film list and film details.

Status management with Pinia (Data Store)

- Storage principle
- Introduction to Pinia: Vuex replacement for Vue 3
- How to type your Pinia blind? How to manage a global SPA status?
- Creating blinds: Defining state, getters and actions
- Integration with components : Using blinds in components
- State persistence: save state in localStorage
- Practical workshop: Setting up a store to manage movie lists and favorites.

Forms and validation

- FormKit : Robust form creation with built-in validation
- VeeValidate: Form field validation
- Error management: Displaying and handling user errors
- Accessibility: Best practices for accessible forms
- Hands-on workshop: Create a film addition form with validation of required fields.

Day 3: Testing, performance and deployment

Automated testing

- How do you test your application so that nothing is left to chance?
- Introduction to Vue-Test-Utils and Vitest
- Unit testing with Vitest
- Installation and configuration
- Simulating events and interactions
- Mocks and stubs
- End-to-end testing with Cypress
- Good testing practices
- Automation via a CI/CD pipeline
- Practical workshop: implementing automated tests on key functions and interactions.

Performance optimization

- Performance analysis: Using Vite Bundle Analyzer and Vue DevTools
- Optimization: lazy loading techniques, list virtualization
- Monitoring: Sentry integration for error tracking
- Best practices: Tips for maintaining a high-performance application
- Practical workshop: Optimize component loading and analyze application performance.

Best deployment practices

- Production Build: Generate an optimized version of the application
- Deployment : Publish the application on Netlify or Vercel
- CI/CD: Implement a continuous integration and deployment pipeline
- Documentation: Using Storybook to document components
- Practical workshop: Deploy the complete application and configure a basic CI/CD pipeline.

Key new features in Vue.js

- Latest news Vue 3.4 and 3.5
- Optimized responsiveness
- Quick template parser
- defineModel() stabilized
- useId() generation IDs
- useTemplateRef() simplified
- Lazy hydration
- onWatcherCleanup() cleaning

- Shorthand v-bind
- Practical workshop: Integrating defineModel() and useId() in a dynamic form to simplify field management.

Companies concerned

This course is aimed at both individuals and companies, large or small, wishing to train their teams in a new advanced computer technology, or to acquire specific business knowledge or modern methods.

Positioning on entry to training

Positioning at the start of training complies with Qualiopi quality criteria. As soon as registration is finalized, the learner receives a self-assessment questionnaire which enables us to assess his or her estimated level of proficiency in different types of technology, as well as his or her expectations and personal objectives for the training to come, within the limits imposed by the selected format. This questionnaire also enables us to anticipate any connection or security difficulties within the company (intra-company or virtual classroom) which could be problematic for the follow-up and smooth running of the training session.

Teaching methods

Practical course: 60% Practical, 40% Theory. Training material distributed in digital format to all participants.

Organization

The course alternates theoretical input from the trainer, supported by examples, with brainstorming sessions and group work.

Validation

At the end of the session, a multiple-choice questionnaire verifies the correct acquisition of skills.

Sanction

A certificate will be issued to each trainee who completes the course.