



To discuss this course and customizations:  
Call: 434-509-5680 or Email: [sales@cloudcontraptions.com](mailto:sales@cloudcontraptions.com)

## Vue for React Programmers

### Class Duration

3 days

### Target Audience

All students must have React, JavaScript, and HTML programming experience. Experience with CSS is helpful, but not required.

### Description

This Vue for React Programmers course provides an in-depth look at modern Vue 3 by systematically translating what students already know from React. It maps JSX to single-file component templates, React hooks to the Composition API, props/state/context to Vue props, reactivity, and provide/inject, Redux and Zustand to Pinia, and React Router to Vue Router 5. Throughout the course, every Vue concept is introduced alongside its React equivalent so students can transfer their existing mental models instead of starting from scratch. The course covers reactivity with ref, computed, and watch, component communication, forms, lifecycle, state management, routing, asynchronous data, and testing with Vitest and Vue Testing Library — and closes with practical guidance for teams migrating React applications or adding Vue to a React shop. This course is perfect for professionals with a background in React who are looking to expand their skill set and learn a new framework.

### Objectives

- Understand the fundamentals of Vue 3 and how its reactivity model compares to React's render-and-diff model.
- Set up a Vue development environment with create-vue and Vite and understand how it differs from React tooling.
- Translate JSX patterns to Vue single-file component templates and directives.
- Map React hooks (useState, useMemo, useEffect, useContext) to the Composition API (ref, computed, watch/watchEffect, provide/inject).
- Translate React props, state, and context patterns to Vue props, reactivity, and provide/inject.



To discuss this course and customizations:  
Call: 434-509-5680 or Email: [sales@cloudcontraptions.com](mailto:sales@cloudcontraptions.com)

- Replace Redux and Zustand patterns with Pinia stores.
- Map React Router concepts to Vue Router 5, including file-based routing.
- Map the wider React ecosystem (Next.js, TanStack Query, React Testing Library) to its Vue equivalents (Nuxt 4, composables and data-fetching libraries, Vue Testing Library).
- Plan an incremental migration from React to Vue.

## Training Materials

All students receive comprehensive courseware covering all topics in the course. Courseware is distributed via GitHub in the form of documentation and extensive code samples. Students practice the topics covered through challenging hands-on lab exercises.

## Software Requirements

Students will need a free, personal GitHub account to access the courseware. Student will need permission to install Node.js and Visual Studio Code on their computers. Also, students will need permission to install NPM Packages and Visual Studio Extensions. If students are unable to configure a local environment, a cloud-based environment can be provided.

## Training Topics

### Introduction

- What is Vue?
- What problem does Vue solve?
- Vue vs. React: rendering and reactivity models
- The Vue SFC compiler
- Moving from React to Vue

### Development Environment

- Requirements
- create-vue vs. Vite React template
- .vue files vs. JSX files
- Vue extension for Visual Studio Code
- Run/Debug a Vue app in Visual Studio Code
- Vue support in WebStorm
- Vue DevTools vs. React DevTools



To discuss this course and customizations:  
Call: 434-509-5680 or Email: [sales@cloudcontraptions.com](mailto:sales@cloudcontraptions.com)

### Templates vs. JSX

- Single-file component structure
- Text interpolation vs. JSX expressions
- `v-bind` vs. JSX attribute expressions
- `v-if/v-for` vs. conditional rendering and `map()`
- `class` and `style` bindings vs. `className` and `style` objects
- Scoped CSS vs. CSS-in-JS and CSS modules
- When Vue's JSX support is worth using

### Reactivity vs. Hooks

- `ref()` compared to `useState`
- `computed()` compared to `useMemo`
- `watch()/watchEffect()` compared to `useEffect`
- Mutation-based reactivity vs. immutable updates
- No dependency arrays, no stale closures, no `useCallback`
- Reactivity gotchas: destructuring and `toRefs`

### Component Basics

- Vue components vs. React function components
- Props with `defineProps()` compared to React props
- Events with `defineEmits()` compared to callback props
- `defineModel()` compared to `value/onChange` pairs
- TypeScript in components, compared to React

### Component Composition

- Nested components
- Slots compared to children and render props
- Scoped slots compared to render props with arguments
- `Provide/inject` compared to React context
- Composables compared to custom hooks

### Event Handling

- `v-on (@)` vs. JSX event handlers
- Event modifiers (`.prevent`, `.stop`) vs. manual calls
- Component events vs. callback props

### Data Binding and Forms

- Two-way binding with `v-model` vs. controlled components



To discuss this course and customizations:  
Call: 434-509-5680 or Email: sales@cloudcontraptions.com

- v-model modifiers
- Form validation and submission
- Compared to React form patterns and libraries

### Lifecycle and Effects

- onMounted, onUpdated, onUnmounted
- Watcher cleanup functions
- nextTick() for awaiting DOM updates
- Template refs compared to useRef
- Compared to React useEffect and Strict Mode double-invocation

### State Management with Pinia

- Pinia compared to Redux Toolkit
- Pinia compared to Zustand
- Setup stores: state, getters, actions
- storeToRefs() and selector patterns
- Migrating Redux/Zustand mental models to Pinia

### Routing

- Vue Router 5 compared to React Router
- Routes, dynamic params, and nested routes
- <RouterLink>/<RouterView> vs. <Link>/<Outlet>
- useRouter/useRoute vs. useNavigate/useParams
- Navigation guards vs. loaders and route guards
- File-based routing in Vue Router 5

### Asynchronous Data

- Data fetching with composables
- Loading and error states
- Async components and <Suspense> (experimental) vs. React Suspense
- TanStack Query for Vue and other data-fetching options

### Ecosystem Mapping and Migration

- Next.js → Nuxt 4
- React Testing Library → Vue Testing Library with Vitest 4
- Component libraries and the Vue ecosystem
- Incremental migration strategies: app-by-app, route-by-route
- Hiring, training, and team considerations



To discuss this course and customizations:  
Call: 434-509-5680 or Email: [sales@cloudcontraptions.com](mailto:sales@cloudcontraptions.com)

Conclusion