

Formation Vue.js

À l'issue de cette formation sur le framework Vue.js, les participants auront acquis les connaissances et compétences nécessaires pour développer des sites Web et applications avec Vue 3. Le programme de formation répond à l'ensemble des dernières recommandations données par l'équipe de Vue.js, permettant aux développeurs et développeuses d'aborder des projets complexes sereinement : utilisation de Vite, Composition API, TypeScript, Pinia...

Durée

3 jours

Objectifs pédagogiques

- Maîtriser les concepts fondamentaux du développement avec le framework Vue : templates, composants, styles, liaison de données...
- Créer et manipuler des composants
- Utiliser Vite pour développer des applications plus rapidement
- Maîtriser Vue Router pour gérer la navigation
- Gérer l'état des applications avec Pinia
- Tester les applications (tests unitaires, de composants et e2e)
- Mettre en œuvre les meilleures pratiques et outils de développement avec Vue 3

Public

Développeurs, architectes

Prérequis

Bonnes connaissances en HTML, CSS et JavaScript

Programme de formation

Introduction à la formation Vue

Objectifs et approche pédagogiques de cette formation Vue.js
Présentation du framework : philosophie, grands principes et évolution au fil des versions
Quelles différences avec Angular et React ?
ES2015+ et les modules
Programmation réactive
Les bénéfices de l'utilisation de TypeScript avec Vue, concepts fondamentaux
Web components
Outillage : utiliser Vite au lieu de Webpack, Visual Studio Code, Volar...
Installation de Vue : Vue CLI, la librairie create-vue

Prise en main et templates

L'architecture de Vue
Single-File Components
Introduction aux templates
v-bind et v-on
De Option API à Composition API
Fonctions et propriétés

Composants et réactivité

Créer et manipuler un premier composant
Props et événements
Cycle de vie des composants Vue
Directives : v-html, v-model, v-if...
Les fonctions reactive(), ref() et computed()

La syntaxe script setup pour définir un composant
watch() et watchEffect()
Timing pour la mise à jour du DOM

Styles et classes

Styles scopés
Préprocesseurs CSS
Liaison de styles et classes (objets, tableaux...)

Aller plus loin avec les composants

Utiliser des directives sur les composants
Cascade d'attributs
Les slots
Provide et Inject
Composants asynchrones

Composables, directives personnalisées et plugins

Qu'est-ce qu'un composable ? Intérêt pour le développement avec Vue
Utiliser un composable
Créer des directives personnalisées
Plugins

Routage et navigation avec Vue Router

Introduction à Vue Router
Paramètres URL
Construction de la pagination
Routes imbriquées et routes protégées
Redirections et alias
Gestion des erreurs

Gestion des états avec Pinia

De Vuex à Pinia
Configuration
Définir un store
Gérer les états
Gérer les getters
Les actions

Tests

Tests unitaires
Présentation de Vitest
Les fonctions describ(), test(), it()...
Tests asynchrones
Introduction à test-utils
Tests de composants
Tests end-to-end avec Cypress

Mise en production

Le déploiement d'applications avec Vue
Meilleures pratiques

Moyens et méthodes pédagogiques

- La formation alterne entre présentations des concepts théoriques et mises en application à travers d'ateliers et exercices pratiques.
- Les participants bénéficient des retours d'expérience terrains du formateur ou de la formatrice
- Un support de cours numérique est fourni aux stagiaires

Modalités d'évaluation

- **En amont de la session de formation**, un questionnaire d'auto-positionnement est remis aux participants, afin qu'ils situent leurs connaissances et compétences déjà acquises par rapport au thème de la formation.

- **En cours de formation**, l'évaluation se fait sous forme d'ateliers, exercices et travaux pratiques de validation, de retour d'observation et/ou de partage d'expérience.
- **En fin de session**, le formateur évalue les compétences et connaissances acquises par les apprenants grâce à un questionnaire reprenant les mêmes éléments que l'auto-positionnement, permettant ainsi une analyse détaillée de leur progression.