

Objectifs / Durée de la formation

Durée: 3 jours, soit 21 heures

Vue.JS est un facilitateur de création d'interfaces Web performance et dynamiques, dans le sens où il met à profit des langages connus et reconnus : HTML, CSS, Javascript.

Bien que très simple d'utilisation, il permet de développer des grandes applications en intégrant le pattern de composants, tout en étant extrêmement léger.

A l'issue de notre formation «Vue.JS», vous serez en mesure de :

- Comprendre la vision et l'essence même de Vue.JS
- Créer des composants Vue et exploiter les avantages apportés par le fichier .vue
- Prototyper des composants rapidement
- Utiliser le Tooling existant
- Développer un annuaire pour accentuer votre compréhension de Vue
- Tester vos composants

Participants / Pré-requis

- Développeurs
- Connaissances HTML et CSS
- Bonne connaissance de Javascript, (idéalement une expérience avec un autre framework front tel que React ou Angular)

Moyens pédagogiques

- Formateur expert dans le domaine
- Mise à disposition d'un ordinateur, support de cours remis à chaque participant, vidéo projecteur, tableau blanc et paperboard
- Feuille de présence signée en demi-journée, évaluation des acquis tout au long de la formation, questionnaire de satisfaction, attestation de stage

Programme

1. Introduction

- Vue.js un framework progressif
- Data driven interfaces
- Ecosystème
- Binding
- Conditionnelles
- Boucles
- Événements
- Composants

2. Bases

- Instance Vue
- Cycle de vie
- Templates
- Interpolation
- attributs
- Directives
- Filtres
- Computed properties
- v-model
- Inputs Natifs
- Rendering de Listes
- Caveats
- (TP) Début de l'annuaire
- Creation d'une recherche simple
- Class and Style bindings
- Rendering conditionnel
- Gestion d'Événements
- (TP) Edition sur place
- Ajout d'édition sur place des personnes
- Composants
- Props
- Événements
- Slots
- Composants dynamiques
- v-model
- (TP) Refactoring annuaire
- Création d'un composant pour l'affichage
- Utilisation de vue build
- Utilisation du template webpack

3. Avancé

- Réactivité avancé
- Transitions
- CSS transitions
- CSS animations
- Transitions de listes
- Transitions dynamiques
- (TP) Animations
- Animations sur la recherche
- Animations sur l'édition
- Render functions
- Composants fonctionnels
- Directives Custom
- (TP) Optimisations
- Exemple pratique de l'utilité des composants fonctionnels
- Création d'une directive custom

4. Extensions

- Mixins
- Plugins
- Single File components

5. Routing avec vue-router

- Vue Router

- Routes
- Modes
- Lien
- La vue principale
- Utilisation basique dans l'annuaire

6. State global avec vuex

- Vuex
- State
- Getters
- Mutations
- Actions
- Modules
- Utilisation basique dans l'annuaire