

## Objectifs :

---

- § Découvrir ReactJS et son fonctionnement
- § Écrire des composants avec les nouvelles syntaxes JSX et ES6
- § Comprendre et intégrer l'architecture Flux
- § Mettre en place un environnement de développement pour ReactJS

## Programme :

---

### Introduction à ReactJs

Node.js et npm JavaScript  
ES6 Webpacks  
Les avantages de React  
Création d'une application  
ReactJs  
IDE, outils et plugins

### ReactJs et le DOM

Événements en React  
JSX, différence entre JSX et HTML/JavaScript  
Méthodes principales de l'API  
DOM virtuel

### Les composants React

Hiérarchie des composants  
Construire un composant de vue  
Composants à état  
Stratégie de composition des composants et bonnes pratiques  
Cycle de vie des composants  
Render, Binding  
Props VS state  
Contrôle des composants de formulaire  
Validateurs  
Créer son StoryBook

### Interactions avancées en ReactJs

Animations  
Drag and Drop  
Mise en place d'un thème avec Styled Component

### Routage

Implémentation du routage  
Gérer les routes  
React Router  
Récupérer les paramètres de l'URL

### Architecture avec Redux

Pourquoi Redux ?  
Separation of Concern  
Implémentation de Redux

---

Créer ses actions et reducers  
Introduction à React-Redux  
Connecter ses composants

## Performances

Réconciliation process works  
Bonnes pratiques concernant la performance

## Isomorphisme

Principes et bénéfices du développement isomorphe  
Node.js pour ReactJS  
Internationalisation (i18n)  
Communiquer avec une API rest  
Configuration et organisation des routes

## Tester les applications ReactJS

Préparer les composants pour les tests  
Les utilitaires de test pour Enzyme, Jest...  
Simulation d'événements

## Introduction à React Native

Hétérogénéité des OS mobiles  
Natif vs hybride vs cross-platform  
Principales solutions cross-platform