

## FORMATION : REACT JS

- **Formation 100% pratique**
- **DOCUMENTATION STRUCTURÉE POUR CHAQUE PARTICIPANT**

### Objectifs:

- Concevoir une application Web Front end moyennant la technologie Javascript et le Framework React JS.
- Développer une application React JS en suivant l'architecture Redux.
- Consommer une API REST au niveau d'une application Front end React JS.
- Maîtriser le concept de synchronisation (synchrone/asynchrone). et le volet Front End d'une architecture Full Stack, et donc l'appliquer pour monter ce tiers dans vos prochains projets.

### Public cible :

- Développeurs, chefs de projet, consultants en informatique, enseignants.

### Planning :

- **Chapitre 1 : Introduction à React JS**
  - Ecosystème des frameworks JavaScript.
  - Les variantes des Frameworks Cross plateformes Front (Angular, Vue).
  - Philosophie de React (Framework VS Bibliothèque).
  - Environnement de développement.
  - Build managé avec la cli create-react-app.
- **Chapitre 2 : Premiers développements avec ReactJS**
  - Le composant.
  - Principe et conséquences du Virtual Dom.
  - Le langage JSX.
  - La fonction render().
  - 'prop' et 'state'.
- **Chapitre 3 : Architecture et découpage du projet**
  - Multi composants : Composition de multiple composants.
  - Partage du state.
  - L'API Component.
  - Cycle de vie d'un composant : Autres aspects majeurs de l'API React.
  - Gestion des événements click, mouseover...

## • Chapitre 4 : Interaction de l'utilisateur avec les composants

- Gestion d'Ajax.
- Gestion des formulaires.
- Composants purs et 'Container Components'.
- Manipulation du DOM, DOM virtuel.
- Le routage : Les différents Routers/Utilisation avec ou sans CORS.

## • Chapitre 5 : Application avec ReactJS et Redux

- Limites de l'approche MVC.
- Présentation de Redux.
- Le Store et les Actions.
- Création de Reducteurs pour les Actions.
- Utilisation avec React : La fonction connect().

### Travaux pratiques :

- Création d'un projet de développement React JS à l'aide de Visual Studio Code.
- Création d'un exemple mon composant.
- Création d'un exemple suivant le modèle MVC.
- Intégration de la notion de state et props.
- Un exemple avec le routage le passage des paramètres.
- Consommation d'une API Rest.
- [Optionnel] Consommation d'une API sécurisée avec JWT.
- Création d'un projet conforme à l'architecture Redux.
- Intégration du Routage avec Redux.