



# REACT JS

Veille Technologie

## RESUME

React Js, est une technologie Javascript créée en 2013 Par les équipes de Facebook Mais il ne s'agit pas d'un Framework à proprement parler. En fait, il s'agit plus d'une librairie Javascript open source qui permet de construire des interfaces utilisateur dynamique.

Anas EL KHIAT  
IPSSI SQY

METTRE EN ŒUVRE DES OUTILS ET STRATEGIES DE VEILLE  
INFORMATIONNELLE

# Sommaire :

<a href="#">Outils de veille.....</a>	<a href="#">2</a>
<a href="#">ReactJS.....</a>	<a href="#">3</a>
<a href="#">Installation et configuration.....</a>	<a href="#">3</a>
<a href="#">Composants.....</a>	<a href="#">3</a>
<a href="#">Gestion des événements.....</a>	<a href="#">3</a>
<a href="#">Gestion de l'état.....</a>	<a href="#">4</a>
<a href="#">Routage.....</a>	<a href="#">4</a>
<a href="#">Communication avec les services Web.....</a>	<a href="#">4</a>
<a href="#">Conclusion.....</a>	<a href="#">4</a>

## Outils de veille :

### Google alertes

# Alertes

Recevez des alertes lorsque du contenu susceptible de vous intéresser est publié sur le Web

Fréquence

Une fois par jour maximum

Sources

Automatique

Langue

français

Région

Toutes les régions

Nombre de résultats

Seulement les meilleurs résultats

Envoyer à

anaselkhiat78@gmail.com

Mettre à jour l'alerte

Masquer les options

google.fr/alerts

Google

## Alertes

Recevez des alertes lorsque du contenu susceptible de vous intéresser est publié sur le Web

Mes alertes (1)

React js

Ma présence sur le Web

Anas

anaselkhiat78@gmail...

## **ReactJS :**

ReactJS est un Framework JavaScript open-source créé par Facebook. Il est utilisé pour construire des interfaces utilisateur interactives et est devenu très populaire ces dernières années. ReactJS utilise une approche de programmation déclarative pour faciliter la création de composants d'interface utilisateur réutilisables. Il est également hautement modulaire, ce qui le rend facile à intégrer à d'autres bibliothèques ou Framework. La documentation officielle de ReactJS est une ressource précieuse pour les débutants et les développeurs expérimentés.

## **Installation et configuration de l'environnement de développement:**

Avant de pouvoir commencer à travailler avec ReactJS, vous devez configurer votre environnement de développement. Pour cela, vous devez installer Node.js et npm sur votre ordinateur. Node.js est une plateforme JavaScript qui permet d'exécuter du code JavaScript côté serveur. npm est un gestionnaire de paquets JavaScript qui permet d'installer et de gérer les bibliothèques et les frameworks JavaScript. Une fois que vous avez installé Node.js et npm, vous pouvez installer l'outil de ligne de commande Create React App. Create React App est un outil qui facilite la création d'un nouveau projet ReactJS en générant une structure de fichiers de base pour vous. Vous pouvez trouver des informations plus détaillées sur l'installation et la configuration de l'environnement de développement dans la documentation officielle de ReactJS.

## **Création de composants React :**

Les composants sont la pierre angulaire de toute application ReactJS. Les composants sont des blocs de construction réutilisables qui constituent l'interface utilisateur de votre application. Dans ReactJS, vous pouvez créer des composants de deux façons : en utilisant des fonctions ou des classes. Les fonctions de composant sont des fonctions JavaScript qui retournent des éléments JSX, tandis que les classes de composant sont des classes JavaScript qui étendent la classe Component de ReactJS. La documentation officielle de ReactJS propose des exemples de code pour vous aider à vous familiariser davantage avec la création de composants.

## **Gestion des événements:**

Les événements sont un élément important de toute application interactive. Dans ReactJS, la gestion des événements se fait de manière un peu différente de ce à quoi vous pourriez être habitué. Dans ReactJS, vous utilisez la syntaxe JSX pour définir les gestionnaires d'événements. Les gestionnaires d'événements sont des fonctions qui sont exécutées lorsque l'utilisateur interagit avec l'interface utilisateur.

## **Gestion de l'état avec ReactJS :**

ReactJS propose une approche unique pour gérer l'état d'une application. Plutôt que de manipuler directement l'état, vous pouvez utiliser le système de gestion de l'état de ReactJS, appelé "state". Le state est un objet JavaScript qui contient des données qui peuvent être modifiées au fil du temps. Lorsque vous modifiez l'état d'un composant React, React met à jour automatiquement l'interface utilisateur pour refléter les nouvelles données.

## **Routage avec ReactJS :**

Le routage est un concept important dans le développement web. Le routage permet à une application web d'avoir des URL différentes pour différentes parties de l'interface utilisateur. Dans ReactJS, vous pouvez utiliser une bibliothèque de routage tierce pour gérer le routage de votre application. L'une des bibliothèques de routage les plus populaires pour ReactJS est React Router.

## **Communication avec les services Web :**

Dans la plupart des applications web modernes, vous devez communiquer avec des services web pour récupérer ou soumettre des données. Dans ReactJS, vous pouvez utiliser des bibliothèques tierces pour communiquer avec des services web. L'une des bibliothèques les plus populaires pour la communication avec des services web est axios.

## **Conclusion :**

ReactJS est un framework puissant et flexible pour la création d'interfaces utilisateur interactives. Dans ce document, nous avons présenté les concepts de base de ReactJS, depuis l'installation et la configuration de l'environnement de développement jusqu'à la communication avec des services web. Nous espérons que ce document vous a aidé à comprendre les concepts de base de ReactJS et à commencer à créer des applications ReactJS. Si vous voulez en savoir plus sur ReactJS, nous vous recommandons de consulter la documentation officielle de ReactJS et de participer à la communauté ReactJS en ligne.