

### 1. Qu'est-ce que React JS ?

React (aussi appelé React.js ou ReactJS) est une bibliothèque JavaScript libre développée par Facebook depuis 2013. Le but principal de cette bibliothèque est de faciliter la création d'application web monopage, via la création de composants dépendant d'un état et générant une page (ou portion) HTML à chaque changement d'état.

React est une bibliothèque qui ne gère que l'interface de l'application, considéré comme la vue dans le modèle MVC. Elle peut ainsi être utilisée avec une autre bibliothèque ou un Framework MVC comme AngularJS. La bibliothèque se démarque de ses concurrents par sa flexibilité et ses performances, en travaillant avec un DOM virtuel et en ne mettant à jour le rendu dans le navigateur qu'en cas de nécessité. La bibliothèque est utilisée par Netflix (côté serveur uniquement depuis le 25 octobre 2017 pour gagner 50 % de performance), Yahoo, Airbnb, Sony, Atlassian ainsi que par les équipes de Facebook, pratiquant l'auto-équipement sur le réseau social éponyme, Instagram ou encore WhatsApp. À la fin de 2015, WordPress.com annonce Gutenberg, une interface pour les éditeurs de sites WordPress, développée en JavaScript avec Node.js et React.

### 2. Quand utiliser React JS ?

Pour savoir quand utiliser React il y a différents cas de figure, comme dit ci-dessus avec les applications web « monopage » comme son nom l'indice à n'utiliser qu'une seule page simple pour son site web

qui est le cas le plus courant pour l'utilisation de ce Framework. Mais il y en a encore plein dont quatre que je pourrais citer ici :

- **Découper votre site/application en composants** : En premier React peut nous apporter une certaine clarté et une utilité en découpant en plusieurs composants votre site ou application, ce qui permettra peut-être de partager une partie de votre application ou site plus simplement avec d'autre personne ou vous-même.
- **Partager de données entre eux** : Avec React on a la possibilité de partager différentes données entre plusieurs composants comme des animations, interaction, suppressions, modifications etc....
- **Application web conséquente** : En utilisant ce Framework qui est React JS vous pourrez l'utiliser pour créer des applications très volumineuses comme des outils en ligne, réseau social ou encore des services.
- **Exemple de sites utilisant React** : Airbnb, Facebook (les créateurs), Dropbox, Reddit, Netflix, CloudFlare, Yahoo, Sony, Instagram, WhatsApp ou encore Atlassian.

### 3. Avantages et inconvénients

Voici quelque avantage que peut avoir React JS quand on l'utilise :

- Le contenu est référençable

C'est la fonctionnalité qui fait la différence par rapport aux autres Framework. Grâce à l'utilisation d'un serveur Node, le code va pouvoir être généré côté client ET côté serveur (un peu comme du PHP donc... tout en gardant les avantages du Javascript !) à la différence des autres Framework JS traditionnels (Backbone.js, AngularJS, Ember.js, etc.) qui de manière native exécutent le code seulement côté client (dans le navigateur). Jusqu'à présent il était

obligatoire de faire passer un bot (service gratuit ou payant) pour qu'il crée des fichiers HTML référençables.

- ReactJS est très rapide

ReactJS crée son propre DOM virtuel où sont rattachés vos composants. Cette approche vous donne énormément de flexibilité et des performances exceptionnelles, car ReactJS calcule quel changement dans le DOM a besoin d'être fait, et change juste LA PARTIE qui a besoin d'être mise à jour. De cette façon, ReactJS évite des opérations coûteuses dans le DOM.

- Les composants sont le futur du développement web

ReactJS a pris le concept de Shadow DOM et du Framework PolymerJS et l'a poussé à un niveau supérieur. React.js n'utilise pas Shadow DOM – à la place il vous donne l'habilité de créer vos propre composant que vous pourrez réutiliser plus tard, combiner, et/ou inclure dans le cœur de votre contenu. Cette fonctionnalité à elle seule est un gage de productivité par la facilité à définir et manipuler vos propres composants.

- La communauté

ReactJS a séduit rapidement la communauté des développeurs JS. Et c'est sans parler de React Native qui de son côté permet de publier des applications natives iPhone (et Android prochainement) en écrivant du JavaScript. Cela permet d'économiser des temps de développement considérables.

- L'intelligibilité

ReactJS produit du code « propre » (simple à lire), sa lecture permet de déterminer immédiatement quelles sont les fonctionnalités de votre application. Ce qui est essentiel pour la maintenance et l'expansion de votre projet dans le temps. Mais il y a toujours des inconvénients pour aller avec des avantages et en voici quelques un :

- Un rythme de développement élevé

L'environnement change constamment et les développeurs doivent régulièrement réapprendre les nouvelles façons de faire. Tout évolue et certains développeurs ne sont pas à l'aise pour suivre un tel rythme. Cette tendance diminue avec le temps, au fur et à mesure que ReactJS prend en maturité

- Une documentation trop légère

Le problème avec la documentation provient des montées de versions constantes de nouveaux outils. Les nouvelles bibliothèques telles que Redux et Reflux promettent d'accélérer le travail d'une bibliothèque ou d'améliorer l'ensemble de l'écosystème React. À la fin, les développeurs ont du mal à intégrer ces outils à ReactJS. Certains membres de la communauté pensent que les technologies React évoluent et s'accélèrent si rapidement qu'on n'a pas le temps de rédiger des instructions appropriées. Pour résoudre ce problème, les développeurs écrivent leur propre documentation sur des outils spécifiques utilisés par eux dans les projets en cours.

- HTML dans mon JavaScript!' - JSX comme barrière

ReactJS utilise JSX. C'est une extension de syntaxe, qui permet de mélanger HTML avec JavaScript. JSX a ses propres avantages (par exemple, la protection de code contre les injections), mais certains membres de la communauté de développement considèrent JSX comme un sérieux inconvénient. Les développeurs et les concepteurs

se plaignent de la complexité de JSX et de la courbe d'apprentissage qui en découle.

- Des problèmes de référencement supplémentaires

On craint que Google et les autres moteurs de recherche ne puissent indexer ou mal indexer des pages Web dynamiques avec un rendu côté client. Ces inquiétudes n'ont pas été pleinement prouvées et il existe des outils de démystification. En 2014, Google a lui-même confirmé que ses robots d'exploration étaient capables de lire du contenu dynamique. Donc, on ne va pas dire que votre application ReactJS ne sera pas indexée par Google. Toutefois, vous devez encore effectuer des tests pour vous assurer que votre application constitue un partenaire de Google, car certains utilisateurs ont signalé des problèmes. Les spécialistes du référencement recommandent de lancer vos applications React via Fetch en tant qu'outil Google pour mieux comprendre comment les robots les éprouvent.

### III. Méthodologie de la veille technologique

---

Afin de mettre en place ma veille technologique, il a fallu dans un premier déterminé les informations prioritaires à suivre. Dans mon cas, j'ai décidé de suivre plus particulièrement le Framework React JS, néanmoins je n'ai pas oublié de rester informer sur les autres actualités concernant le domaine de l'informatique. J'ai donc utilisé une large palette d'outil afin de rester au mieux renseigné sur l'actualité.

## 1. Site internet

Voici une liste des sites internet que j'ai pu utiliser pour alimenter ma veille :

- <https://fr.reactjs.org/docs/getting-started.html> :

Documentation officielle du Framework.

- <https://openclassrooms.com/fr/> : Site regroupant différents tutoriel et cours en ligne que j'ai utilisé pour React JS.
- [https://developer.mozilla.org/fr/docs/Learn/Tools\\_and\\_testing/Client\\_side\\_JavaScript\\_frameworks/React\\_getting\\_started](https://developer.mozilla.org/fr/docs/Learn/Tools_and_testing/Client_side_JavaScript_frameworks/React_getting_started) : Tutoriel développé par Mozilla pour apprendre React JS.

## 2. Outil

Pour pouvoir accéder aux dernières actualités le plus rapidement possible, j'ai utilisé :

- Flux RSS : <https://feedly.com/i/my>
- Google alert : <https://www.google.fr/alerts>