

# Rendu 2 - Développement web avancé

Groupe 4 - Clément Miesse, Cyril Wastchenko, Ambroise Mostin

## Semaine 4 : 2 - 8 Mars → Lead Clément

### Choix Technologiques

#### React

Comparatif Angular / React →

	Angular	ReactJs
Utilisation du DOM	DOM du navigateur	Virtuel DOM
Data Binding	Bidirectionnel	Unidirectionnel
Apprentissage du Framework	TypeScript (Difficile à apprendre)	JSX (Apprentissage simplifié)
Support des plateformes mobiles	Hybride	Natif
Structure d'application	Totalement MVC ( <i>Model View Controller</i> )	MVC ( <i>Dominante View</i> )

<https://www.groupe-scala.com/reactjs-point-sur-ce-framework-javascript/>

#### Avantages de React :

- grande communauté, active depuis 2013

<https://da-14.com/blog/its-high-time-reactjs-ten-reasons-give-it-try>

- Facilite l'écriture des composants (avec utilisation de JSX)
- Booste la productivité et facilite la maintenance (exemple : Walmart avec ses différents magasins)
- Un rendu plus rapide (utilisation d'une virtual DOM)
- Un code stable (les modifications sont rétroactives - une modification affecte une composant à la fois)
- Accès à des outils de débogage grâce à : React Developer Tools

#### Désavantages de React :

- L'apprentissage et le passage à un framework MVC
- L'utilisation de JSX est assez difficile à prendre en main
- La librairie énorme de React, on se noie dans la doc
- React change constamment (comme ceci est un projet de 4-6 mois, pas trop désavantageux)

## Pourquoi pas Vue ou Angular?

Vue est plus rapide à mettre en place mais rajouter des composants et s'y retrouver dans le code (déstructuré) n'est pas aussi facile que React.

Angular est plus long à mettre en place et consomme donc plus de temps avant d'être opérationnel, en plus d'exiger de la rigueur dans la codage, ce que nous n'avons pas encore.

→ React a une communauté très active, permettant de trouver des informations un peu partout sur le web, ce que Angular et Vue ont moins.

## Node.js (Express.js)

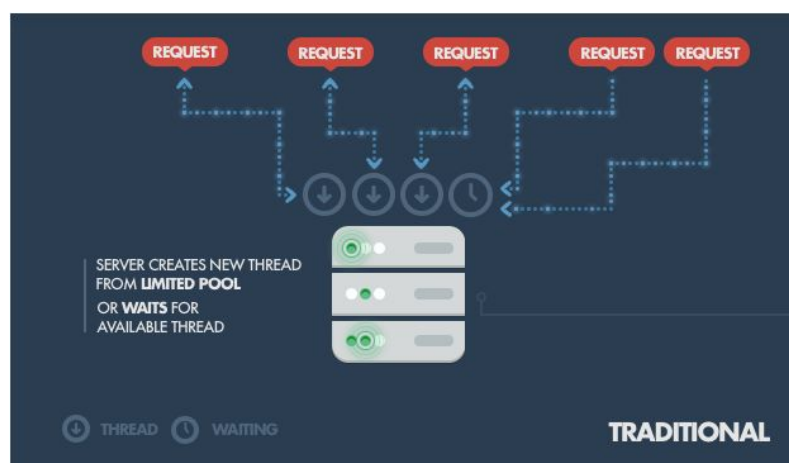
<https://www.mindinventory.com/blog/pros-and-cons-of-node-js-web-app-development/>

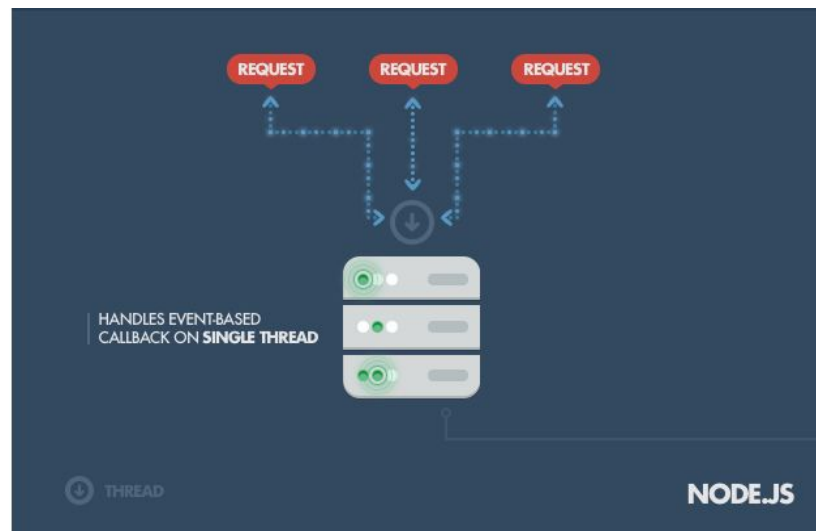
### Avantages de node.js :

- Facilement scalable (commencer petit et augmenter au fur et à mesure)
- Coder en Javascript, comme React, moins de temps nécessaire pour l'apprentissage (Full stack)
- Performances élevées grâce au moteur V8 de google et aux opérations non bloquantes - Monothread
- Grande communauté active
- Gestion de requêtes simultanément → plusieurs appels à des itinéraires = plusieurs requêtes simultanées

### Désavantages de node.js :

- API peu stable (suite aux changements de version)
- Pas de support solide au niveau des librairies
- Le développement asynchrone est plus difficile à prendre en main mais est nécessaire au plus l'app devient grande
- Ne supporte pas le multi-threading





## MySQL

Which database is right for your business?		
	MySQL	MongoDB
Use case	Legacy applications or applications that require multi-row transactions (i.e. accounting systems)	Real-time analytics, content management, internet of things, mobile apps
Data structure	Structured data with clear schema	No schema definition required
Risk	Risk of SQL injection attacks	Less risk of attack due to design
Analysis	A great choice if you have structured data and need a traditional relational database.	A great choice if you have unstructured and/or structured data with the potential for rapid growth.

Il est à noter que le client devrait nous fournir : soit la DB, soit les informations nécessaires aux différents champs de la DB.

## La structuration des fichiers

<https://www.youtube.com/watch?v=HPIjjFGYSJ4>

<https://www.youtube.com/watch?v=v0t42xBIYIs>

[https://www.youtube.com/watch?v=Law7wfdg\\_Is](https://www.youtube.com/watch?v=Law7wfdg_Is)

J'ai dû effectuer la commande SQL ci-dessous pour autoriser la connexion

```
ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY 'password'
```

## Pages web

<http://127.0.0.1:3000/>

- Page d'accueil affichant une carte de Louvain-la-Neuve et reprenant l'ensemble des points d'intérêts accessibles par l'application ainsi que de préparer un itinéraire de promenade

<http://127.0.0.1:3000/login>

- Page d'authentification, permettant au client de se connecter à sa zone de gestion de la base de données

<http://127.0.0.1:3000/admin>

- Page de gestion, accessible par le client-même, lui permettant de modifier les informations disponibles sur la base de données
- Entrées sur la BDD :
  - Nom Français
  - nom Latin
  - Localisation
  - Catégorie
  - Période de visibilité
  - accessible/non-accessible (Boolean)

<http://127.0.0.1:3000/pointInteret>

- Page reprenant tous les points d'intérêt

<http://127.0.0.1:3000/pointInteret/{id}>

- Page d'information sur un point d'intérêt (Contenu vidéo, image, texte, GIF, ...)

<http://127.0.0.1:3000/categorie>

- Page reprenant tous les points d'intérêts d'une catégorie spécifique

<http://127.0.0.1:3000/itineraire>

- Page permettant à l'utilisateur de visualiser l'itinéraire qu'il a créé sur la page d'accueil

## Infos VPS :

<https://docs.ovh.com/fr/vps/debuter-avec-vps/>

<https://docs.ovh.com/fr/dedicated/ssh-introduction/>

<https://docs.ovh.com/fr/vps/conseils-securisation-vps/>

## Géolocalisation

- <https://tympanus.net/codrops/2012/10/11/real-time-geolocation-service-with-node-js/>
- <https://pusher.com/tutorials/live-map-react>

## Admin

- <https://www.youtube.com/channel/UC3znQDQfQKaR5h5XKGkbAxA/featured>
- <https://www.youtube.com/watch?v=7Ujn7g1JF1Q> (Page admin)
- [https://www.youtube.com/watch?v=1lOKKK\\_yrlc](https://www.youtube.com/watch?v=1lOKKK_yrlc) (Page admin suite)

## QR Code

[Télécharger le lecteur/générateur de qr code](#)



(ce qr code renvoi sur 127.0.0.1:3000/admin)

