

Festival Search Engine

Steven Liatti et Vincent Tournier

Cours de systèmes distribués - Prof. Nabil Abdennadher - Hepia ITI 3^{ème} année

13 décembre 2017

Plan

- 1 Introduction
 - Buts du projet
 - API utilisées

- 2 Serveur
 - Route events
 - Route infos
 - Route tracks
 - MongoDB
 - Divers

- 3 Client
- 4 Conclusion

Introduction

Buts du projet

Créer un moteur de recherche d'événements musicaux, permettant à l'utilisateur de :

- Afficher des événements sur une carte interactive
- Afficher des informations d'un événement en particulier
- Afficher des informations à propos des artistes
- Jouer (en arrière plan) un extrait d'un son d'un artiste de l'événement

API utilisées

Serveur

Route events

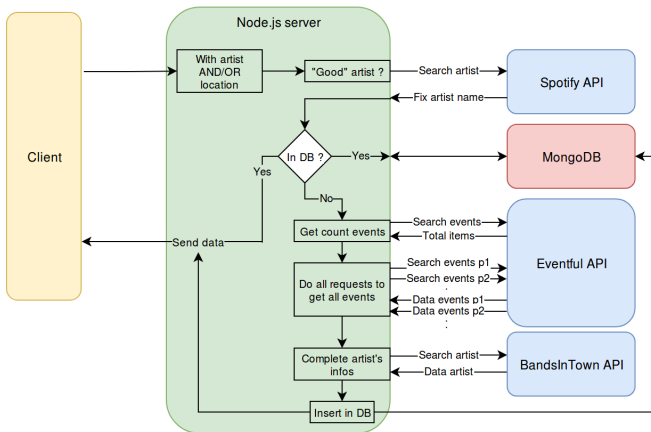


Figure – Route events

Serveur

Route infos

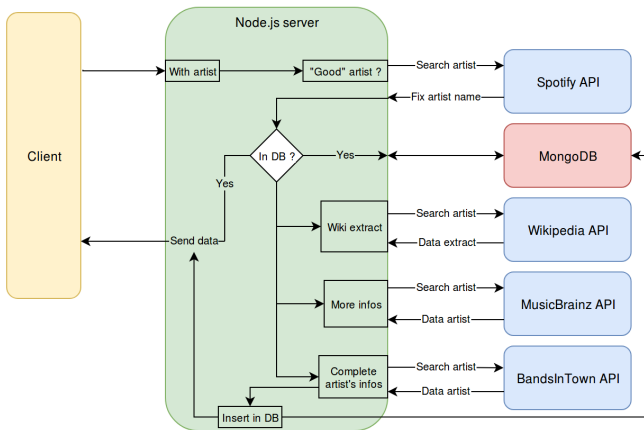


Figure – Route infos

Serveur

Route tracks

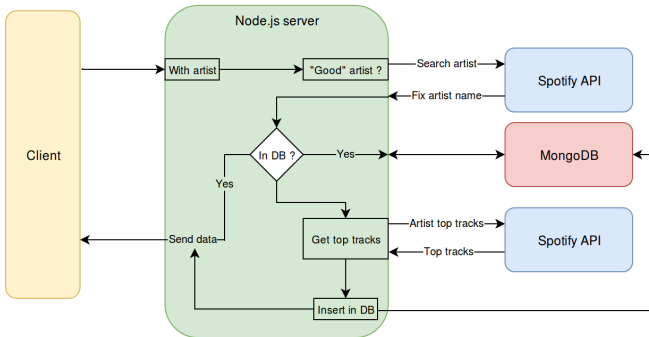


Figure – Route tracks

Serveur

MongoDB

Utilisation de MongoDB comme base de données :

- Simplicité/compatibilité avec Javascript et Node.js (JSON)
- Données "temporaires" avec la feature MongoDB Time To Live
- Découverte d'un SGBD alternatif (NoSQL)

Serveur

Divers

Serveur Node.js avec structure basée sur les promesses (Promise) :

```
1 // doSomething() and doSomethingElse() return a Promise
2 doSomething()
3 .then(result => { return doSomethingElse(result); })
4 .then(finalResult => { console.log("OK :" + finalResult); })
5 .catch(error => { console.log(error); });
```

Utilisation d'AWS (avec scripts shell pour le déploiement) :

- Pour héberger le serveur Node.js, la base de données MongoDB, la documentation de l'API et le serveur Apache pour servir les fichiers client (HTML)
- Mettre en pratique le cours de Cloud

Client



texte important

Conclusion

Steven :

- Approfondissement de Node.js et découverte des Promise Javascript et de MongoDB
- Découverte de bonnes (Spotify, Wikipédia) et "moins bonnes" (MusicBrainz, Eventful) API

Vincent :

-

Beaucoup de plaisir à travailler sur ce projet