



ASSOCIATION LIBERTIC



ENTREPRISE LES  
POLYPODES

STAGE DE FIN D'ÉTUDE

---

## Rapport de Stage

---

*Auteur :*  
Thibaud COURTOISON

*Responsables :*  
M. Ronan GUILLOUX  
M. Nicolas HERNANDEZ



UNIVERSITÉ DE NANTES



IUT DE NANTES

Du 13 avril 2015 au 19 juin 2015

## Remerciements

Je tiens à remercier L<sup>A</sup>T<sub>E</sub>X et les tutoriels sur internet.

Bla bla bla bla bla.

## Resumé

# Sommaire

## Table des matières

<b>Remerciements</b>	<b>1</b>
<b>Resumé</b>	<b>2</b>
<b>Sommaire</b>	<b>3</b>
<b>Introduction</b>	<b>4</b>
<b>1 Présentation de l'entreprise</b>	<b>5</b>
<b>2 Analyse et conception</b>	<b>6</b>
2.1 Environnement de travail . . . . .	6
2.2 Mise à niveau préliminaire . . . . .	6
2.3 Analyse des RFC . . . . .	6
2.4 Analyse des versions précédentes . . . . .	7
2.5 Analyse cahier des charges . . . . .	8
2.5.1 La partie serveur . . . . .	8
2.5.2 La partie application . . . . .	9
2.6 Conception UML du projet . . . . .	9
<b>Conclusion</b>	<b>10</b>
<b>Références</b>	<b>11</b>

## Introduction

Ce document est un exemple de rapport écrit en L<sup>A</sup>T<sub>E</sub>X.

Écrit par Thibaud Courtoison grâce à <http://blog.hikoweb.net/>.

# 1 Présentation de l'entreprise

## 2 Analyse et conception

Comme tous projets de développement, il faut commencer par une phase d'analyse et de conception. Le projet Open Data Event a comme particularité d'avoir déjà eu plusieurs versions avant que je ne commence à travailler dessus. C'est pourquoi la partie d'analyse des versions précédentes était très importante.

### 2.1 Environnement de travail

J'ai effectué mon stage dans les bureaux de l'entreprise **Les Polypodes**, situés au 7<sup>ème</sup> de l'immeuble *Sigma 2000*. Avec une vue panoramique sur l'île de Nantes, il fallait cependant ne pas se laisser divertir par les différents utilisateurs du terrain multi-sport en bas de l'immeuble.

Côté technique, il a été mis à ma disposition un **iMac 21"** avec un second écran de la même taille. J'ai donc pu découvrir l'environnement de développement offert par **OS X**. Globalement, la connaissance de l'environnement **Linux** m'aura aidé tout au long de l'utilisation d'OS X.

Après m'avoir créé un compte administrateur sur l'ordinateur fourni, j'ai pu installer tout les logiciels nécessaire à la réalisation du projet. Pour des raisons personnels, j'ai choisi d'utiliser les outils suivants :

- **iTerm** (Version 2.0.0)
- **Sublime Text 2** (Version 2.0.2)
- **Google Chrome** (Version 43)
- **Wireshark** (Version 1.12.5)

Pour l'utilisation de serveurs SabreDAV, Baïkal, ElasticSearch ou autre (XXX citer la section correspondante XXX), j'ai installé une machine virtuelle grâce à **Vagrant**.

**LiberTIC** étant une association militant pour les licences libres, il était évident que la majorité de mon travail (qu'il s'agisse de développement ou de compte-rendus) soit mis lui aussi sous licence libre. Ainsi, il est possible de retrouver mon travail sur **GitHub** à l'adresse suivante : <https://github.com/LiberTIC/ODEV2>.

En plus de mon environnement local, j'ai pu accéder à un serveur de pré-production hébergé par OVH.

### 2.2 Mise à niveau préliminaire

Dès le début de mon stage, il a été défini que le projet allait se construire sur des technologies que je ne connaissais pas et/ou ne maîtrisais pas. C'est pourquoi il a été convenu que la première semaine de mon stage devais me servir pour effectuer une mise à niveau pour différentes technologies.

Premièrement, j'ai suivi un récapitulatif des commandes **Git** [1] puis lu un article sur les bonnes pratiques de l'utilisation de Git et de ses branches [2].

Ensuite, j'ai lu plusieurs articles sur les bonnes pratiques du développement **PHP** [3] [4].

Enfin, j'ai lu et appliqué la totalité du **Symfony Book** (<http://symfony.com/doc/current/book/index.html>)

### 2.3 Analyse des RFC

Avant de commencer le projet, il fallait déjà que je comprenne toutes les technologies abordés. **CalDAV** étant la base du projet, j'ai du approfondir mes connaissances sur le sujet.

Pour re-situer **CalDAV**, il s'agit d'une extension de WebDAV, elle-même extension du protocole HTTP. Globalement, elle définit et précise la façon d'envoyer des données au format **iCalendar** sur le réseau Internet.

**iCalendar**, quand à lui, est un format de fichier (.ical ; .ics ; .icalendar ). Il permet de stocker des événements (tels que "Cinéma à 16h" ou "Réunion service compta' tous les lundi à 9h"). Ce format est ensuite utilisé par des logiciels (tels que iCal, Google Calendar, etc..) pour en faire un affichage plus adapté aux utilisateurs.

Pour plus d'informations sur CalDAV et iCalendar, j'ai établi un "tutoriel" pour expliquer l'utilisation de chacun : [https://github.com/LiberTIC/ODEV2/blob/master/doc/Thibaud\\_Printemps2015/Tutoriel\\_Comprendre\\_CalDAV.md](https://github.com/LiberTIC/ODEV2/blob/master/doc/Thibaud_Printemps2015/Tutoriel_Comprendre_CalDAV.md)

Pour effectuer ce travail d'analyse, j'ai du lire la presque-totalité des RFC suivantes :

- **RFC 4918** : <https://tools.ietf.org/html/rfc4918>
- **RFC 4791** : <https://tools.ietf.org/html/rfc4791>
- **RFC 2324** : <https://tools.ietf.org/html/rfc2324>

## 2.4 Analyse des versions précédentes

Avant que je ne commence à travailler dessus, ce projet a traversé de nombreuses phases d'analyses, de conceptions et de développement.

### La première réunion

L'élément déclencheur de ce projet est la réunion du 11 juin 2014 à Stéréolux avec l'association LiberTIC et quelques acteurs de l'événementiel à Nantes. Il est apparu qu'il y avait une nécessité d'améliorer la diffusion des données.

C'est ainsi qu'est né le projet ODE (Open Data Event). Il s'agit d'un agrégateur d'événement permettant de relier les fournisseurs d'événements (les organisateurs de festivals, concerts, conférence, etc...) avec les réutilisateurs d'événements (les utilisateurs de l'API, les journaux, etc...).

Cette réunion a permis de produire un cahier des charges pour le projet (disponible à cette adresse : [https://github.com/LiberTIC/ODEV2/blob/master/doc/Documents/120622\\_ODE\\_cahierDesCharges\\_MakinaCorpus.pdf](https://github.com/LiberTIC/ODEV2/blob/master/doc/Documents/120622_ODE_cahierDesCharges_MakinaCorpus.pdf) )

### La première version

Makina Corpus ( <http://makina-corpus.com/> ), entreprise de développement de logiciels libres, a répondu au cahier des charges en proposant de réaliser le projet. Plusieurs mois plus tard, la première version était finie et fut présentée à des responsables de l'association LiberTIC.

Cependant, durant la réunion, il est apparu de nombreuses différences dans la direction du projet menée par Makina Corpus avec la direction voulue par l'association. A la fin de la réunion, il a été décidé de ne pas continuer la relation entre l'association et l'entreprise.

### La reprise par Les Polypodes

Suite à une réunion de l'association LiberTIC abordant l'échec de la première version, Ronan Guilloux de l'entreprise Les Polypodes, présent à cette réunion, fit la proposition suivante : Les Polypodes serait prêt à accueillir des stagiaires pour effectuer le projet. Les stagiaires seraient



accueilli par l'entreprise et l'association s'engagerait à faire un suivi plus approfondit du projet pour éviter la même fin que la version précédente.

## Version 2 - Les premiers stagiaires

En janvier 2015, deux stagiaires de BTS chez Les Polypodes ont commencé à travailler sur ODE version 2.

Leur travail a été de produire un prototype d'une application sous Symfony2 permettant une gestion des événements avec un serveur CalDAV. Ils ont fait un comparatif des serveurs CalDAV et fait une analyse des sémantiques disponibles pour les événements.

Sachant que la durée de leur stage n'était que de six semaines, ils n'ont pas pu faire évoluer le prototype ( disponible à cette adresse : <https://github.com/polypodes/CalDAVClientPrototype> )

## Version 2 - Mon stage

Depuis le 13 avril 2015, je travaille donc sur la version 2 du projet. Le but est de pouvoir réaliser le projet dans le temps qui m'est imparti ( 10 semaines ).

## 2.5 Analyse cahier des charges

Le 30 avril, j'ai participé à une réunion avec des représentants de l'association LiberTIC pour définir leurs besoins et leurs attentes quand à la réalisation de ce projet avec moi.

Durant cette réunion, je leur ai expliqué ce que j'avais effectué comme travail jusque là, je leur ai ensuite présenté plusieurs diagramme de conception UML, puis, nous avons discuté des points importants sur lesquels je devrai me focaliser durant mon stage.

### 2.5.1 La partie serveur

J'ai longtemps réfléchi sur la pertinence d'utiliser CalDAV pour le projet. En effet, il s'agit d'une extension du protocole HTTP datant de plus de 15 ans qui n'est pas très utilisé.

Cependant, sachant que le partage des données doit être encouragé, il faut s'assurer que n'importe qui puisse inscrire des données dans ODE, que ce soit un(e) commercial(e), un(e) responsable technique ou un(e) secrétaire. Or, l'avantage de CalDAV est que de nombreux clients calendrier (dont Google Calendar et Apple Calendar) permet l'ajout d'un compte compatible CalDAV.

Ainsi, en utilisant CalDAV, nous pourrions disposer d'interfaces utilisateurs déjà optimisé pour la création et la gestion d'événements.

Le deuxième point que j'ai abordé durant la réunion du 30 avril fut la façon dont les événements serait stockés. J'ai d'abord pensé qu'une base de données SQL serait la façon la plus simple et efficace de stocker des événements. Cependant, en réfléchissant à la structure d'un événement (dont je parlerais plus tard), je me suis rendu compte que la structure ne convenait pas à un stockage linéaire comme SQL.

Après m'être renseigné sur les différentes alternatives à SQL, j'ai opté pour ElasticSearch (<https://www.elastic.co/>). Il s'agit d'un moteur d'indexation permettant un stockage au format JSON, idéal pour la structure d'un événement. De plus, grâce à ElasticSearch, il sera possible d'implémenter facilement une recherche d'événements par mot-clés.

### **2.5.2 La partie application**

## **2.6 Conception UML du projet**

## Conclusion

Pour conclure, avec  $\text{\LaTeX}$  on obtient un rendu impeccable mais il faut s'investir pour le prendre en main.

## Références

- [1] Gitimmersion. *Best GIT tuto ever -50 steps*. URL : <http://gitimmersion.com/>. Consulté le 13 avril 2015.
- [2] Vincent Driessen. *A successful Git branching model*. Janvier 2010. URL : <http://nvie.com/posts/a-successful-git-branching-model/>. Consulté le 13 avril 2015.
- [3] Brian Fenton. *Best Practices for Modern PHP Development*. URL : <https://www.airpair.com/php/posts/best-practices-for-modern-php-development>. Consulté le 14 avril 2015.
- [4] Hugo Hamon. *Votre code est STUPID ? Rendez le SOLID*. Décembre 2013. URL : <http://afsy.fr/avent/2013/02-principes-stupid-solid-poo>. Consulté le 14 avril 2015.