

ELAIN Kevin
ELIE Benoît

JUSTIFICATION DES CHOIX TECHNIQUES

SOMMAIRE

1) Introduction.....	2
2) GIT	2
2.1) En lui même :	2
2.2) Face aux autres :	2
3) Dropbox	2
3.1) En lui même :	2
3.2) Face aux autres	2
4) Html5/CSS3/media-query et responsive design	3
4.1) En eux mêmes :	3
4.2) Face aux autres :	3
5) AJAX	3
5.1) En lui même :	3
5.2) Face aux autres :	3
6) MySQL	3
6.1) En lui même :	3
6.2) Face aux autres :	3
7) BootStrap.....	4
7.1) En lui même :	4
7.2) Face aux autres :	4
8) NoSql – version personnel.....	4
8.1) En lui même :	4
8.2) Face aux autres :	4

1) INTRODUCTION

Les choix techniques pris lors de ce projet vont être détaillés dans ce document.

2) GIT

2.1) EN LUI MÊME :

Une gestion des versions simple, par auteur et par ligne, le tout daté et avec les fichiers concernés.

La possibilité d'un travail en branche et d'une navigation entre les versions.

2.2) FACE AUX AUTRES :

GIT a l'avantage d'être multiplate-forme, de disposer d'un mode en ligne de commande simple à prendre en main et qui ne change pas entre les plate-formes.

Il se caractérise aussi vis-à-vis de SVN par la possibilité d'un travail en local, sans se référer à un serveur principal.

3) DROPBOX

3.1) EN LUI MÊME :

Synchronisation rapide, possibilité de restaurer des versions antérieures, bon espace de stockage disponible.

3.2) FACE AUX AUTRES

Le principe de dossier synchronisé est simple à prendre en main et automatique.

L'espace disponible (2Go) est aussi adapté à un travail sur des sources logicielles.

Il est lui aussi multiplate-forme et dispose même d'une interface web.

3.3) ABANDONNÉ

Des problèmes de synchronisation avec des dépôts GIT qui s'étoffent peuvent arriver, normalement avec des conflits entre les différentes versions des fichiers internes de Git, le choix de changer de centralisation a donc été pris.

4) **GITHUB**

4.1) **LUI MÊME :**

Permet une centralisation des versions sur un serveur, la consultation des données générés et le travail collaboratif.

Il se base du la gestion de version local réalisé par le logiciel Git.

4.2) **POURQUOI LUI ?**

GitHub présente l'avantage d'être une des plus grosse plate forme de centralisation de code.

Les dépôts open source sont gratuit à mettre en place et la gestion commence à être éprouvé.

5) **HTML5/CSS3/MEDIA-QUERY ET RESPONSIVE DESIGN**

5.1) **EN EUX MÊMES :**

Ils permettent une gestion simple des différentes tailles d'écrans que peut rencontrer une application web.

Le fait que l'application s'adresse à des employés permet de s'assurer que les machines pourrons êtres mise à jour pour le support des technologies requises.

5.2) **FACE AUX AUTRES :**

Le responsive design présente l'avantage de pouvoir s'adapter à la fois à un poste fixe doté d'un affichage supérieur à 17", comme à celui d'un smartphone ou d'une tablette.

L'application n'est donc qu'à développé une seul fois, avec une compatibilité maximale.

Ces technologies permettent de tirer profits de toute la surface d'affichage disponible, ainsi que de mettre en place des améliorations d'ergonomie.

6) **AJAX**

6.1) **EN LUI MÊME :**

Il permet d'éviter des rechargements intempestif de page entière et permet donc à l'utilisateur une navigation plus fluide par la modification en temps réelle de la page qu'il consulte.

6.2) **FACE AUX AUTRES :**

Il reste assez bas niveau et ne demande pas le chargement de bibliothèque externe (jQuery).

La manipulation du DOM étant bien développé en JavaScript, l'AJAX permet une dynamisation rapide des contenus sans complexifier réellement le code.

7) MySQL

7.1) EN LUI MÊME :

SGBDR dans le trio de tête, il permet une gestion de base via les moteurs les plus courants.

Distribué sous licence GPL il est à la fois utilisé par le grand public et les professionnels.

7.2) FACE AUX AUTRES :

Il est derrière Oracle, mais présente l'avantage d'être plus simple à administrer et mettre en place (au détriment d'une petite perte de fonctionnalité).

Il est aussi moins lourd en terme de ressources consommées pour le serveur.

8) BOOTSTRAP

8.1) EN LUI MÊME :

C'est un kit de CSS, de JQuery et de modèle HTML. Il permet une mise en place rapide d'un site aux normes responsive design.

Il se compose sous la forme de plug-in rapide à appeler dans le code.

8.2) FACE AUX AUTRES :

Très rapide à prendre en main, personnalisable rapidement par des surcouches.

Son aspect « responsive-friendly » l'a rendu rapidement incontournable sur le web mobile. Il permet notamment de gérer des menus et des block.

Peu de concurrence, si ce n'est la version « from scratch » ou des frameworks bien plus lourds et moins souples.

9) NoSQL – VERSION PERSONNEL

9.1) EN LUI MÊME :

C'est une petite classe de gestion de données en fichier/dossier. Le but est de rendre la manipulation rapide, simple et avec le moins de contrainte possible.

Des méthodes de base sont mises à disposition et permettent dans notre cas de gérer les jetons d'identification générés pour les utilisateurs qui veulent être « toujours connecté », sans avoir à retoucher à la base de données.

9.2) FACE AUX AUTRES :

Une solution SQL standard est envisageable mais pour une gestion de fichier qui change souvent et ne présente pas d'importance critique en cas de disparition.

Les grandes solution NoSql sont quand à elle un peu complexe à mettre en place (car plus optimisée), ce qui justifie le choix d'une solution personnel.