Documentation Francis Verreault

## Introduction

Notre équipe de quatre programmeurs a reçu le mandat de mettre à jour un site web de gestion de patrimoine. Cette tâche nous demandait tout d’abord de bien comprendre le projet dans son ensemble puis de trouver les fonctionnalités que le client voulait améliorer ou supprimer. Pour ma part j’ai reçues la tâche de modifier grandement l’affichage des objets. De plus je devais ajouter la possibilité de faire des contenants principaux. J’expliquerai premièrement l’ajout des contenants principaux et deuxièmement l’indexation des objets qui a été plus laborieux.

## Ajout contenant principal [bug #15]

L’ajout d’un contenant principal a impliqué beaucoup plus de changements que je le croyais. En effet l’ensemble du système de stockage des objets fonctionnait de façon à ce qu’un seul contenant principal existe. La manière de trouver les objets d’un propriétaire consistait à trouver son unique contenant avec le type contenant principal et de prendre ses enfants. La recherche et l’affichage des objets impliquaient des méthodes PHP, du JavaScript, du html-css et du SQL. Toutes ces méthodes fonctionnaient en cherchant l’unique contenant principal. Je tenterai de plus expliquer mes changements que de montrer du code puisqu’il y a eu des changements à une dizaine d’endroits. La solution aux contenants principaux est venue de Marc qui a suggéré d’enlever l’affichage de l’unique contenant principal déjà existant. Ainsi nous pouvions avoir plusieurs contenants principaux soit les premiers objets du contenant initial. La façon d’identifier les contenants principaux devenait très simple ; regarder si son parent est le contenant initial. Ceci a donc impliqué l’ajout de quelques méthodes PHP effectuant de simples querys à la bd. Voici donc de quoi à l’air la nouvelle fonctionnalité d’ajout de contenant racine:



L’ajout de ce bouton permet donc d’ajouter des contenants racines, s’il n’est pas sélectionné le nouvel objet sera ajouté dans l’objet présentement sélectionné sauf si il n’y aucun contenant principal le nouvel objet sera un contenant principal. De plus il nous a obligé à changer les données de quelques clients qui désiraient conserver leur ancien contenant principal.

## Indentation arborescence [bug #16]

Cette partie était de loin la plus compliquée puisque le code existant n’était pas commenté et il avait été optimisé par un programme qui enlève les sauts de lignes donc l’indentation était à refaire. Cela n’aurait pas été très difficile mais ce code était en jquery une extension de js que je connaissais très peu. J’ai beaucoup travaillé avec Marc sur cette fonctionnalité et le principal problème était de changer les fonctions PHP qui allaient chercher les objets du user. Dans l’ancienne version les objets parents n’étaient affichés alors que maintenant ils le sont lorsqu’on va plus loin dans l’arborescence des objets. Voici un exemple de la nouvelle arborescence.



En fin de compte la fonction qui était déjà présente fût complètement remplacée. Nous n’avons gardé que les styles css qu’elle appliquait au tableau et à ses éléments.

## Augmenter le nombre d’objet visible maximum [bogue # 17]

Ce problème c’est réglé par lui-même en changeant complètement la méthode PHP qui obtenait les objets à afficher dans l’arborescence. L’ancien code ne permettait que de récupérer 8 objets. Le nouveau code récupère absolument tous les objets et ils sont ensuite affichés par le JavaScript.

## Conclusion

En conclusion nous avons appris que travailler sur un projet déjà codé demande beaucoup de préparation. Lorsque je parle de préparation je veux dire apprendre les technologies du programme et sa fonctionnalité générale. Sans cette préparation il aurait été extrêmement difficile et pénible de se retrouver dans les milliers de lignes de codes pour chercher la cause d’un bug. L’implémentation d’une version locale était une partie cruciale de la préparation. Cette partie était absolument nécessaire puisque le projet se trouvait sur un serveur distant et que nous devions pouvoir tester rapidement les modifications. Cette importation fût difficile dû au fait de notre faible expérience en matière de serveur. De plus le projet Patrimoine possédait une base de donnée qui heureusement c’est facilement installé dans MySql. En bref travailler sur le code d’autres personnes a été difficile et d’autant plus lorsque le code n’était pas commenté. Côté coéquipier je n’ai pour ma part que du positif à dire sur mes camarades. La division des tâches au début du projet c’est très bien déroulée et tout le monde a accompli ce qui était attendu de leur part. De plus Mike a été un très bon leader pour Marc, Kevin et moi, il était celui qui avait la meilleure vue d’ensemble sur le projet et qui savait si le travail avançait suffisamment selon les buts que nous nous étions fixés.