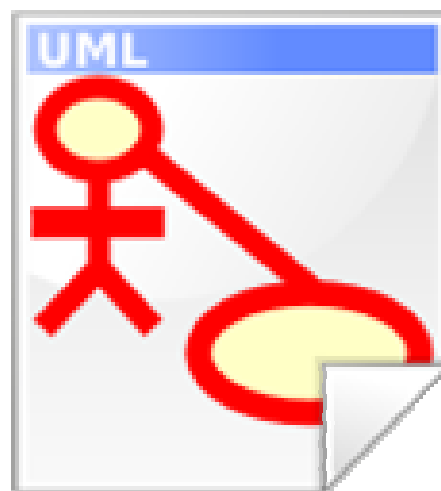




BILAN DU PROJET

Version 1.0

UMBRELLO



	Bilan du projet		Version 1.0
			22/03/2007

VERSION	DATE	DESCRIPTION EVOLUTIONS	DES	AUTEURS
1.0	22/03/2007	Création		Equipe de projet M1



SOMMAIRE

SOMMAIRE.....	3
1. INTRODUCTION.....	4
1.1. Objectif.....	4
1.2. Portée.....	4
1.3. Références	4
2. BILAN DU PROJET	5
2.1. Introduction	5
2.2. Bilan sur la communauté KDE	5
2.3. Bilan sur l'annuler / refaire	5
2.4. Bilan sur l'UML 2.0.....	6
2.4.1. <i>Diagramme de séquence</i>	6
2.4.2. <i>Diagramme d'activités</i>	7
2.5. Bilan sur la gestion de projet.....	8
2.5.1. <i>Cycle itératif</i>	8
2.5.2. <i>Estimation du temps de travail</i>	8
2.5.3. <i>Wiki</i>	8
3. CONCLUSION	9



1. INTRODUCTION

1.1. Objectif

Ce document a pour but de faire le point sur l'ensemble du projet Umbrello. Il a pour objectif de montrer les différents points abordés dans ce BE et d'apporter une conclusion sur les acquis au sein de l'équipe.

1.2. Portée

Ce document est destiné aux membres de l'équipe Umbrello ainsi qu'aux superviseurs du projet.

1.3. Références

- Plan de développement
- Bilans des itérations

	Bilan du projet		Version 1.0
			22/03/2007

2. BILAN DU PROJET

2.1. Introduction

Cela fait 5 mois que nous travaillons sur Umbrello, nous sommes maintenant dans la phase de transition, le moment est venu de présenter un bilan de notre travail.

Nous tirerons d'abord un premier bilan de nos rapports avec la communauté puis nous parlerons de notre travail.... Enfin, un bilan objectif sera énoncé.

2.2. Bilan sur la communauté KDE

La communauté KDE a été très accueillante et enthousiaste sur le fait que des étudiants travaillent sur un de leur projet. En effet, l'université Paul Sabatier -Toulouse 3- est la première université en France qui permettait à ses élèves de travailler sur un projet KDE. Des réactions du monde entier ont été perçues (espagnole, française, américaine...).

Ces passionnés qui travaillent sur KDE nous ont promis de nous aider si un problème survenait, et ils n'ont pas failli à leurs promesses: chacune de nos questions, qu'elles soient sur IRC ou sur les listes de diffusions ont eu des réponses satisfaisantes. De plus, plusieurs fois durant le déroulement du projet, nous avons eu des retours sur notre travail par différentes personnes de la communauté, surtout de la part de notre mentor Jonathan RIDDELL, cela nous a permis de modifier notre orientation de travail à certains moments.

2.3. Bilan sur l'annuler / refaire

Suite à l'analyse des performances réalisées lors de la phase d'élaboration, nous avons identifié le problème majeur à la perte de performances d'Umbrello: il s'agissait d'une mauvaise gestion de l'annuler/refaire. Nous avons donc décidé de remplacer l'existant par une architecture beaucoup plus optimisée.

Notre choix s'est porté sur la gestion par KundoStack, ceci est une pile fournie par KDE qui sert à la gestion des annuler/refaire. Une fois celle-ci implémentée, il a fallu modifier l'architecture pour répondre aux exigences de la nouvelle pile: ce travail fut très long et demanda beaucoup de réflexion sur la nouvelle architecture à adopter. Il reste encore quelques bugs à corriger afin que l'annuler/refaire soit complètement effectif.

	Bilan du projet		Version 1.0
			22/03/2007

Nous utilisons un modèle de conception, appelé le “pattern Command”, qui permet de modéliser des actions comme un objet de conception. La succession de ces objets formes des opérations qui sont stockées dans un tableau appelé pile de Undo/Redo, l’ordre d’exécution de celles ci provoque des “Redo” ou “Undo” suivant que l’on se dirige vers la fin ou le début de ce tableau. C’est cette technique qui permet d’obtenir des performances supérieures, on appelle cette architecture “Action based Undo/Redo Command”, a contrario du “State based Undo/Redo Command” qui était implanté.

2.4. Bilan sur l’UML 2.0

Durant le développement, toutes les fonctionnalités permettant d’obtenir le statut compatible UML 2.0 n’ont pu être implantées. Cependant les diagrammes de séquences et d’activités possèdent ce statut, en effet nous pouvons créer ces diagrammes avec le nouveau formalisme comprenant les nouveaux composants UML 2.0 :

2.4.1. Diagramme de séquence

- Les fragments combinés.
 - Opérateur “Alternative”
 - Opérateur “Option”
 - Opérateur “Break”
 - Opérateur “Parallel”
 - Opérateurs “Reference”
 - Opérateur “Negative”
 - Opérateur “Critical”
 - Opérateurs “Ignore” et “Consider”
 - Opérateur “Assertion”
 - Opérateur “Loop”
- Messages Perdus / Trouvés.

2.4.2. Diagramme d'activités

- Pré/Post conditions.
- Symbole « accept time event ».
- Flot de contrôle.
- Flot d'objets.
- Région interruptible.
- Région d'expansion.
- Control node : flow final.
- Transformation.
- Objects nodes :
 - Invocation d'activité.
 - Activity Parameter Nodes.
 - Central buffer nodes.
 - Data stores nodes.
 - Input / Output / Constant Pins.

Les composants implantés actuellement sont des composants graphiques, il n'y a pas de modèle UML métier associé. Cependant les contrôles sémantiques (associations inter-objets, ...) sont effectifs au niveau du tracé.

	Bilan du projet		Version 1.0
			22/03/2007

2.5. Bilan sur la gestion de projet

2.5.1. Cycle itératif

Ce projet nous a permis de mettre en pratique un cycle de développement itératif. Cette méthode de travail s'est avérée bénéfique pour avancer et terminer le projet, car à chaque itération, les retours des enseignants ont permis de nous réorienter dans les itérations suivantes.

2.5.2. Estimation du temps de travail

Au début du projet, on avait estimé notre charge de travail à **650** heures à partir de la phase de lancement jusqu'à celle de transition, mais à la fin on est arrivé à **793** heures. Ce gros écart peut s'expliquer par la mauvaise estimation selon les compétences de chacun et surtout par le temps qu'on a mis pour installer l'environnement de développement. De plus, une grosse partie du dépassement concerne la phase de construction où on découvrait la bibliothèque Qt.

2.5.3. Wiki

Nous avons utilisé pendant ce projet, un Wiki qui s'est avéré bien utile. Il nous a permis de centraliser les informations et les documents décrivant notre avancement sur le projet.



3. CONCLUSION

Travailler sur Umbrello nous a apporté diverses choses.

Dans un premier temps, il y a le plan communautaire. En effet, travailler dans un bureau d'étude se situant dans le monde du libre et plus précisément KDE est une chose complètement innovante pour nous. Personne dans notre groupe n'avait travaillé sur un projet d'une portée internationale.

Il est vraiment enrichissant de partager nos avancements avec des personnes d'une autre culture, qui ont souvent des points de vues différents. Le fait de travailler avec des personnes du monde entier nous a apporté la notion de « communication à grande échelle »: nous ne pouvions pas rester dans l'optique d'un projet « local », il fallait communiquer avec les personnes extérieures aux projets, nous nous sommes donc inscrits dans les listes de diffusions KDE, nous nous sommes connectés sur les channels IRC dédiés au développement (#kde4-devel, #umbrello...).

Le second point important est l'apprentissage de l'environnement KDE et de son fonctionnement. Un de nos premiers travaux a effectivement été de comprendre partiellement l'architecture de KDE (KDElibs, KDEpimlibs...). Grâce à cela, nous avons pu réaliser dans de bonnes conditions notre travail sur Umbrello. Il va s'en dire que nous avons découvert le Qt et approfondie notre connaissance du C++ en même temps, ainsi que l'utilisation avancée d'un gestionnaire de version, Subversion en l'occurrence..

Le troisième point est l'étude des performances, ce travail fut très instructif, ce n'est pas une chose que nous avons étudié dans nos cours. Pour réaliser ce travail, nous avons dû utiliser un logiciel de KDE : Valgrind, qui nous a fournit une liste de statistiques et des graphes permettant de trouver la source des problèmes de performances d'Umbrello. Le problème se trouvant principalement dans le système de l'annuler/refaire, nous avons donc appliqué des patterns pour corriger cela.

Enfin, ce projet nous a permis de créer un véritable esprit de groupe au sein de l'équipe Umbrello, le travail en binôme (application de la méthode XP) nous a grandement aidé dans la gestion de projet et dans l'intégration des L3. Mais un problème a été soulevé, en effet, lorsque les L3 sont arrivés dans notre projet, le C++ ne leur étant pas encore été enseigné, il y a eu quelques ralentissements dans le travail. Quand à elle, la planification de nos tâches s'est effectuée avec plus ou moins de mal, mais nous avons réussi à bien avancer.

En conclusion, nous dirons que l'expérience a été concluante sur tous les points : humain et technique. Nous n'avons qu'une seule chose à espérer: c'est que l'expérience se renouvellera l'année prochaine !