

Planification

mercredi 17 décembre 2008

Projet ***Monofin***

4^{ème} année Informatique
INSA Rennes

*YOANN CHAUDET
PAUL GARCIA
QUENTIN GAUTIER
NICOLAS LE SQUER
NICOLAS MUSSET
XAVIER VILLOING*

Encadreurs

*PATRICE LEGUESDRON
LAURENT MONIER
FULGENCE RAZAFIMAHERY*

Sommaire

I.	Introduction.....	3
II.	Méthodologie de planification	4
A.	Définition de la semaine-type	4
B.	Semaines chômées.....	4
C.	Planification générale.....	4
III.	Analyse	5
IV.	Conception	6
A.	L'architecture générale	6
B.	Architecture détaillée.....	7
1.	Interface graphique	7
2.	Éditeur de dessin	7
3.	Interface COMSOL	8
V.	Construction / Développement.....	9
A.	Interface graphique	9
B.	Éditeur de dessin	10
C.	Interface COMSOL	10
D.	Tests d'intégration.....	11
VI.	Livraison.....	12
VII.	Conclusion	13

I. Introduction

Le projet Monofin sur lequel nous travaillons depuis quelques mois, consiste en la réalisation d'une interface graphique faisant le lien entre les utilisateurs et un logiciel de simulation physique. Monofin devra notamment permettre : la modélisation d'une géométrie tridimensionnelle de monopalmes, le paramétrage d'une simulation physique ainsi que la récupération des résultats de la simulation.

Ce dossier a pour objectif premier de présenter le travail de planification initiale qui a été réalisé jusqu'à maintenant. Dans un premier temps, nous présenterons le diagramme de Gantt (voir en annexe) réalisé à l'aide de Ms Project ; puis la planification de chaque grande partie sera justifiée et expliquée.

II. Méthodologie de planification

Nous avons choisi de baser notre planification sur les prévisions au plus tard, en partant du principe que la date du 29 mai 2009 est une date butoir incontournable. Les autres dates clés du projet ont été définies comme dates jalons afin de se fixer des repères et des échéances à court terme.

A. Définition de la semaine-type

En période normale, c'est-à-dire pendant les cours à l'INSA, nous avons estimé que chaque membre du projet était en mesure de travailler en moyenne une heure par jour ainsi que deux heures le samedi et le dimanche (soit 9 heures par semaine et par membre).

Ceci a été réévalué pour la suite du projet (après janvier 2009) car la charge de travail sera plus importante. On passera ainsi à 17 heures par semaine et par membre.

B. Semaines chômées

Certaines semaines, comme les semaines de partiels ou les vacances ne sont pas travaillées. Cependant, il est apparu nécessaire de planifier du travail pour une des deux semaines de Pâques pour respecter les délais.

C. Planification générale

Il est plus facile de planifier le projet de manière générale avant d'affecter le travail aux différents membres. Ainsi, une ressource globale de capacité 600% (car six membres de projet), que nous avons sobrement nommé « Équipe », représente le travail global à effectuer sur les différentes tâches.

III. Analyse

La phase d'analyse du projet est celle qui s'est déroulée entre septembre et aujourd'hui, c'est-à-dire mi décembre.

La phase d'analyse a donnée lieu à la rédaction d'un premier rapport de spécifications générales terminé au 24 octobre 2008. Les spécifications générales ont permis de rassembler dans un dossier toutes les recherches effectuées depuis le début du projet, de présenter d'une manière générale les besoins des utilisateurs ainsi que les objectifs souhaités.

Dans un second temps nous avons rédigé un rapport plus conséquent de spécifications fonctionnelles, terminé au 5 décembre 2008. Ce rapport fait office de cahier des charges, il décrit précisément les fonctionnalités qui seront offertes par notre application aux utilisateurs. La phase d'analyse menant à la rédaction de ce second rapport est également passée par la réalisation d'une ébauche de prototype. Le prototype nous a permis de réfléchir sur l'agencement des interfaces graphiques mais également sur les fonctionnalités que nous devons proposer.

Cette première phase d'analyse s'est déroulée avant la phase de planification à proprement parler, ce n'est pas pour autant qu'elle ne figure pas dans le diagramme de Gantt. Nous avons passé environ 390 heures sur cette phase d'analyse, toutes tâches confondues (excepté le décompte des heures liées à la rédaction des rapports qui lui ne figure pas dans la planification).

IV. Conception

La conception du logiciel doit permettre de décrire toutes les parties de son architecture, ainsi que la manière dont elles vont s'articuler.

A. L'architecture générale

L'architecture générale devra fournir des informations concernant les moyens de communication des différents aspects du logiciel. Ces aspects sont l'interface de simulation, le système de dessin, et le langage de script dialoguant avec COMSOL. Enfin il faudra détailler le fonctionnement de chaque partie afin qu'elle soit vue de l'extérieur comme une boîte noire.

L'architecture générale devra être composée d'un grand diagramme UML qui représentera les sous-parties sous une forme simplifiée. En effet, il n'est pas nécessaire à ce stade d'avoir une idée précise de l'architecture détaillée. Cependant il est important de faire figurer les éléments d'une partie qui sont utiles pour les autres.

Les relations entre les parties devront également être expliquées. Par exemple, il faudra préciser le déroulement du dialogue entre l'interface de configuration de la simulation et COMSOL, via le langage de script de COMSOL. De nombreuses autres interactions comme la récupération et l'affichage des résultats ou le système de sauvegarde devront également apparaître.

L'usage d'un certain nombre de modèles de conception devra également être justifié. En effet, il sera probablement utile d'utiliser des messagers ou des classes intermédiaires pour faciliter la communication des éléments entre eux. De plus, les formats des fichiers de sauvegarde devront être détaillés à ce moment. Il faudra savoir quelles seront les informations exactes qui s'y trouveront, comment les charger dans chaque élément du logiciel, et comment les modifier, en évitant les conflits.

Pour répondre à ces questions, qui restent malgré tout assez générales, il est nécessaire qu'une équipe, relativement nombreuse, se penche dessus (au moins quatre personnes). Cela est dû au fait que la conception plus profonde des différents éléments dépendra de ce qui aura été décidé à ce niveau. Ainsi, plus il y aura de personnes s'occupant de la conception générale et plus la conception détaillée sera aisée. De cette manière, de nombreux conflits pourront être évités.

Enfin, le travail demandé à cette étape n'étant pas très important en termes de charge, il a été estimé qu'il nous faudrait soixante heures pour réaliser la tâche. Il a également été tenu compte du fait que d'autres personnes pourraient se joindre à cette étape de conception selon l'avancement des autres tâches parallèles.

B. Architecture détaillée

En ce qui concerne l'architecture détaillée, le plus important est de bien concevoir l'interface de graphique qui sera mise à la disposition de l'utilisateur. C'est en effet elle qui permettra de modéliser la palme, de saisir ses propriétés physiques et de lancer les simulations.

1. Interface graphique

Comme il a été vu dans les spécifications fonctionnelles, cette partie est composée du point de vue de l'utilisateur de plusieurs étapes qu'il doit franchir avant de lancer la simulation. Là encore, il faudra réaliser un grand nombre de diagrammes UML et les confronter afin de choisir la méthode de conception la plus efficace. Cependant, la première étape concernant le système de dessin sera laissée de côté à ce stade car sa conception sera vue par la suite.

Il faudra prévoir l'implémentation de chaque fonctionnalité vue dans les études précédentes, notamment en ce qui concerne les moyens d'enregistrement, et les différentes bibliothèques mises à la disposition de l'utilisateur (matériaux, dessins, formules).

Comme il s'agit d'une partie plus spécifique, et qui a déjà été abordée de façon assez précise lors des spécifications, une équipe plus réduite y travaillera. De plus, cela permettra à l'équipe de conception générale de continuer à travailler en cas de retard. Cependant, elle représente un volume d'heures plus important (environ cent) car il faudra être extrêmement précis sur les différents objets utiles et sur l'organisation des tâches. Il pourra être judicieux de réaliser des scénarios de fonctionnement, ainsi que des diagrammes de fonctionnement afin de détecter à l'avance d'éventuels problèmes de conception.

2. Éditeur de dessin

L'éditeur de dessin, quant à lui représentera la partie la plus complexe de la conception. Il sera indispensable de préciser à ce stade les nombreux objets manipulés par l'utilisateur, leurs interactions et les répercussions graphiques.

Il fera l'objet d'une conception détachée de celle de l'interface de configuration. En effet, la création du modèle de la palme n'a aucune influence sur les autres étapes de la configuration. De plus, les très nombreux objets et interactions qui figureront à ce niveau alourdiraient inutilement les autres aspects de la conception s'ils n'étaient pas traités séparément.

Il faudra concevoir les attributs de chaque élément : point d'intersection, point de contrôle (faisant à référence au terme « point-tangente » dans le dossier n°2), courbe, segment etc. Il faudra aussi prévoir l'influence qu'un objet peut avoir sur les autres. Par exemple, déplacer un point de contrôle déplacera les deux tangentes correspondantes, et modifiera la courbe. De plus il est nécessaire de prévoir le comportement de ces objets vis à vis des actions de l'utilisateur (clics gauches, droits etc.). Comme il est prévu de pouvoir annuler certaines opérations, il faudra penser à créer une mémoire des actions, ainsi que les différents accès qui y seront demandés. Enfin, il faudra gérer l'affichage des éléments et leur modification en temps réel. Cela implique que la vue soit actualisée régulièrement.

La réalisation de cette tâche a été estimée à environ deux-cent soixante heures. Ce n'est pas négligeable et la tâche est capitale. De plus, une équipe importante devra s'y atteler car il est nécessaire de décomposer les différents aspects présents à ce stade, vu leur complexité. C'est pourquoi au moins quatre personnes devront s'en charger.

3. Interface COMSOL

La dernière partie de conception traitera de l'interface avec COMSOL. Le format du script à générer devra être étudié, afin de spécifier les éléments composant cette partie.

Comme cet aspect du logiciel est complètement transparent du point de vue de l'utilisateur, une plus grande liberté de conception lui sera accordée. Cependant, il devra répondre à un certain nombre d'impératifs. En effet, Il faudra aller chercher dans le fichier de sauvegarde les informations concernant la simulation. Il s'agit d'une part de la géométrie de la palme afin d'en avoir le maillage dans COMSOL, et d'autre part des données physiques de la simulation.

De plus, le langage de COMSOL Script présente un certain nombre de contraintes qu'il faudra gérer. Cela implique une forme de génération particulière pour le script. Néanmoins, le système de génération sera toujours le même ce qui simplifiera grandement le travail de conception.

C'est pourquoi une équipe plus légère pourra s'en occuper et ce, en parallèle de la conception de l'interface de dessin. Il a été prévu que deux personnes réalisent ce travail dont la durée est estimée à environ cent vingt heures.

V. Construction / Développement

Cette étape va comporter deux phases principales : Une première phase de développement pur, et une phase de test. Cette partie possède logiquement le volume horaire le plus élevé de tout le projet car c'est réellement ici que le logiciel va prendre vie.

Le développement se subdivise en deux modules : le module concernant l'interface graphique et le module concernant COMSOL. Ces deux modules possédant une interaction minimale (ils peuvent fonctionner l'un sans l'autre), ils seront développés parallèlement.

Nous avons choisi de faire apparaître les tests d'intégration dans la planification mais pas les tests unitaires. En effet, nous estimons que les tests unitaires doivent être réalisés en même temps que le développement. Nous adopterons dans la mesure du possible une méthode de développement pilotée par les tests : TDD (*Test Driven Development*). Les tests d'intégration consisteront à tester le fonctionnement des modules en bonne intelligence.

A. Interface graphique

Cette partie durera environ deux cents heures et impliquera au moins deux personnes. Il s'agira de construire entièrement l'interface graphique du logiciel à partir de la conception effectuée précédemment. C'est une partie qui demandera un travail d'apprentissage de la bibliothèque Qt et de codage en langage C++. La sous-partie « Éditeur de dessin » page 10 n'est pas comptée dans le développement de l'interface graphique, puisqu'elle représente un développement à part entière, plus compliqué que celui de l'interface en général.

Le développement de l'interface graphique sera en grande partie basé sur le prototype réalisé lors des phases d'analyse. En effet, le style graphique sera bien défini, et les fonctionnalités toutes présentes au début de cette partie. Le principe sera donc de reprendre ce prototype, le retoucher de façon propre et définitive grâce à l'outil Qt Designer, qui permet de créer des fenêtres graphiques sans coder.

Une fois les fenêtres réalisées sous Qt Designer, le restant du travail sera de coder en C++ toutes les actions prévues dans les fonctionnalités en y incluant l'algorithme de reconnaissance de contour choisi. Pour ce travail, il faudra donc d'abord étudier très précisément la syntaxe de la librairie Qt. Cela dit, le fonctionnement général de cette librairie aura déjà été vu pour la partie conception, ce qui raccourcira le temps d'apprentissage.

L'interface sera prévue pour accueillir l'éditeur de dessin lorsque le développement de celui-ci sera terminé, ainsi que la partie fabrication de scripts pour COMSOL.

Cette tâche n'est pas très longue et ne nécessite que deux personnes, et ce parce que l'on se basera sur le prototype réalisé en parallèle du dossier n°2. Cela permettra de gagner du temps en termes d'organisation graphique des différentes fonctionnalités, en plus des liens entre les fenêtres qui seront en partie réalisés. De plus, la phase de conception permettra de coder facilement les différentes fonctionnalités en se basant sur les diagrammes réalisés. Pour finir, les fenêtres des étapes 2 à 5 (cf. dossier n°2) étant relativement simples, elles ne devraient pas poser beaucoup de problèmes à développer.

Tout au long de cette phase, comme pour les autres parties du développement, les tests unitaires seront effectués au fur et à mesure du codage, ainsi que la documentation.

B. Éditeur de dessin

Cette partie concerne le développement des modules relatifs à la modélisation de la forme en trois dimensions de la monopalme. Elle comprend notamment le développement du module de dessin, l'implémentation de l'algorithme de reconnaissance de forme et l'algorithme de vectorisation de la forme.

Le module de dessin contiendra la structure de données permettant de représenter la forme de la monopalme ainsi que les outils nécessaires à la construction et la modification de la forme. La structure de donnée devra être suffisamment simple pour pouvoir être travaillée aisément et devra pouvoir s'utiliser facilement dans le script COMSOL. De plus, l'ensemble de cette partie devra s'intégrer parfaitement à l'interface graphique de l'application.

Les algorithmes seront testés au préalable avant de les intégrer à l'application. L'algorithme de vectorisation travaillant directement sur la structure de donnée représentant la forme de la monopalme, il sera nécessaire de choisir cette structure avant de travailler sur l'algorithme.

Nous avons estimé que quatre personnes seront nécessaires à la réalisation de cette tâche. Par soucis d'homogénéisation, cette tâche devra se dérouler à la suite du développement de l'interface graphique. En effet, il s'est avéré préférable d'avoir une version finie de l'interface pour pouvoir intégrer correctement l'éditeur de dessin.

Enfin, la répartition individuelle des tâches se décidera certainement lors de la conception. Cependant selon une première estimation, le développement du module de dessin nécessitera probablement plus de ressources.

C. Interface COMSOL

La partie COMSOL, qui compte pour un total de trois cents heures, comporte deux sous-modules interdépendants, un sous-module dédié au script et à sa construction et un sous-module dédié à la récupération des résultats. Cette partie a été prévue pour deux personnes qui devront travailler la documentation COMSOL permettant de concevoir des scripts correspondants aux besoins et permettant une gestion fine de COMSOL. Les scripts serviront à lancer les calculs souhaités et à récupérer facilement les résultats.

Le sous-module script sera le plus complexe à concevoir car d'une part il nécessitera de bien connaître COMSOL, son mode de fonctionnement et son langage de script, et d'autre part il permettra de fabriquer dynamiquement ces scripts (suivant la monopalme et les paramètres). Nous aurons donc besoin d'un premier temps d'apprentissage avant de passer au développement proprement dit.

Le sous-module traitement des résultats sera développé par les mêmes personnes qui ont conçu le sous-module scripts pour bénéficier de leurs compétences acquises sur le logiciel COMSOL.

Le développement de ces deux sous-modules va nécessiter énormément de test d'une part pour prendre en main COMSOL, d'autre part pour assurer un fonctionnement sans problème quelques soient les conditions tant au niveau de la génération du script que pour s'assurer la récupération des bons résultats.

D. Tests d'intégration

Cette partie va consister en l'assemblage des différents modules développés et les tests visant à savoir si l'intégration s'est bien passée. Elle n'est pas très longue (quarante heures) car si le reste du développement s'est bien passé et que la conception a été correcte dès le début, il ne devrait pas y avoir beaucoup de problèmes. Une équipe de quatre personnes devrait suffire, afin d'éviter l'utilisation de trop de ressources sur une portion de code peu volumineuse et difficilement partageable.

VI. Livraison

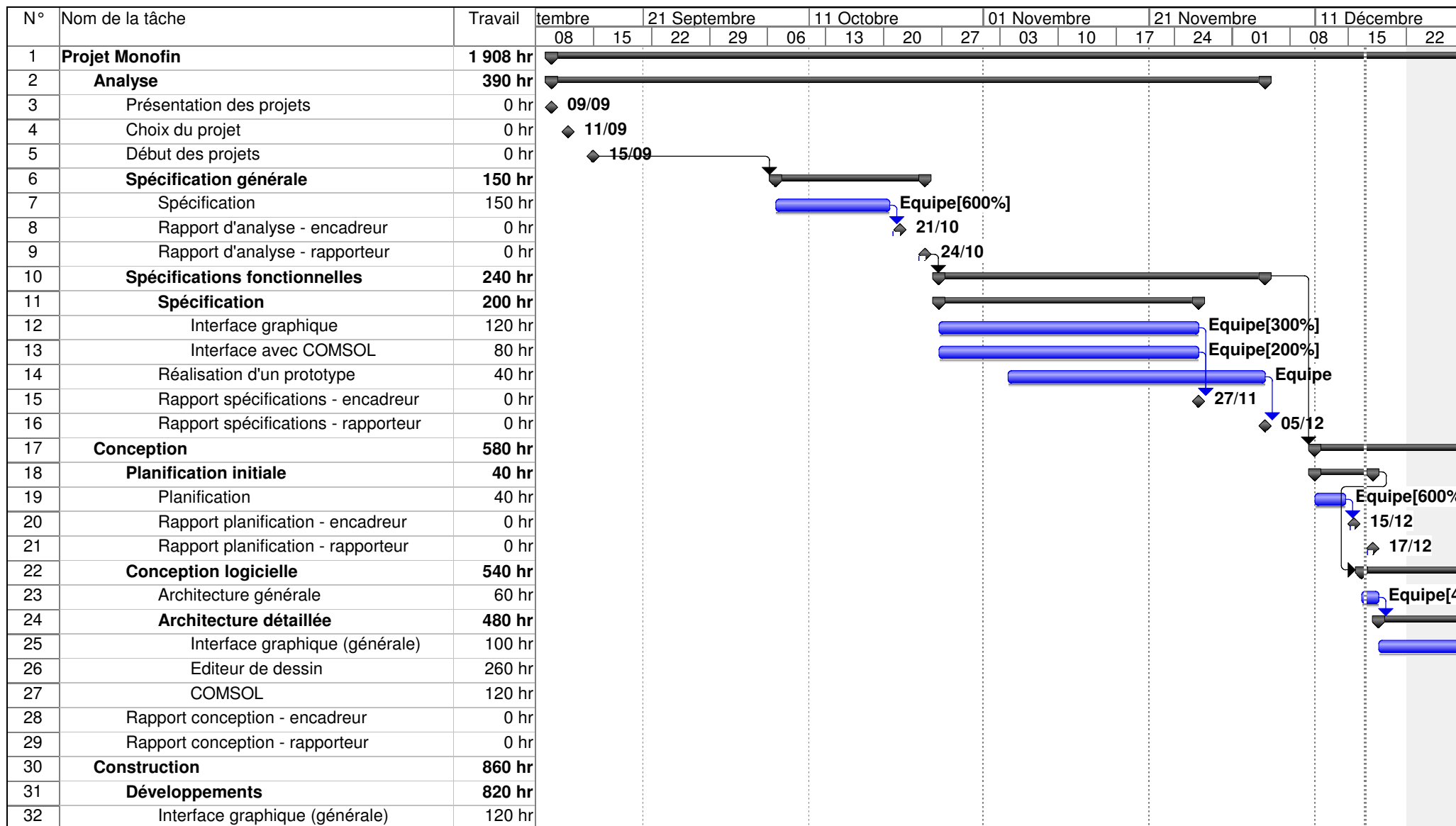
La livraison est une petite étape qui sera effectuée à la fin du projet et ne devrait durer qu'une dizaine d'heures réparties sur une ou deux personnes (pour le moment, deux personnes sont prévues). Le but sera d'une part, de simplement regrouper les rapports, les *slides* des soutenances, les fichiers source commentés du projet, la documentation du logiciel, les exécutables et le nécessaire pour les installer (fichiers DLL, etc.), et d'autre part les jeux de test effectués pour tester le logiciel, et ce, sur un CD-Rom structuré.

Il faudra enfin mettre en place une démonstration du logiciel en préparant des scripts et des tests valides, afin de présenter le résultat du travail aux encadreurs et au rapporteur.

Cette partie prendra peu de temps, puisqu'en principe, le logiciel sera terminé et fonctionnera, les parties les plus difficiles ayant éventuellement été abandonnées pour s'assurer d'un fonctionnement stable. Il suffira donc d'utiliser le logiciel final pour le mettre en forme et le rendre facilement utilisable et présentable.

VII. Conclusion

Cette première planification nous a permis d'expliciter les tâches qui restent à effectuer et surtout de nous rendre compte de la charge de travail à venir. Pour ce faire, nous avons mis en pratique les conseils de l'intervenant de la société Teamlog quand à l'utilisation de l'outil Microsoft Project. Nous avons suivi ses conseils et planifié le projet en partant de la date butoir que représente la fin de l'année : 29 mai 2009. Sous ses conseils, nous avons également créé une entité « Équipe » pouvant travailler à 600% (6 personnes dans l'équipe * 100% de disponibilité). Chaque tâche est attribuée à cette entité, de telle sorte qu'aujourd'hui la planification n'est pas individuelle mais expose uniquement le nombre de personnes requises à sa réalisation. Nous souhaitons que les membres de notre équipe soient le plus polyvalent possible et cette méthode de planification semble très bien remplir son objectif.



Projet : planification_initiale_definitive
Date : Mer 17/12/08

Tâche

Avancement

Jalon

Récapitulative



Tâche reportée

Jalon reporté

Avancement reporté

Fractionnement



Tâches externes

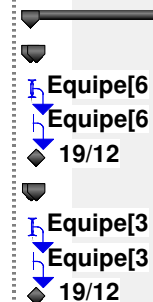
Récapitulatif du projet

Regrouper par en-tête

Échéance



N°	Nom de la tâche	Travail	tembre		21 Septembre		11 Octobre			01 Novembre		21 Novembre			11 Décembre			
			08	15	22	29	06	13	20	27	03	10	17	24	01	08	15	22
33	Editeur de dessin	400 hr																
34	COMSOL	300 hr																
35	Scripts	200 hr																
36	Traitement des résultats	100 hr																
37	Tests	40 hr																
38	Test d'intégration	40 hr																
39	Page HTML - encadreur	0 hr																
40	Page HTML - rapporteur	0 hr																
41	Livraison	10 hr																
42	Démonstrations	0 hr																
43	CD-ROM	10 hr																
44	Page Finale - encadreur	0 hr																
45	Page Finale - encadreur	0 hr																
46	Soutenances	68 hr																
47	Spécifications	26 hr																
48	Préparation	20 hr																
49	Répétition	6 hr																
50	Soutenance orale	0 hr																
51	Planification	12 hr																
52	Préparation	9 hr																
53	Répétition	3 hr																
54	Soutenance orale	0 hr																
55	Finale	30 hr																
56	Préparation	24 hr																
57	Répétition	6 hr																
58	Soutenance orale	0 hr																



Projet : planification_initiale_definitive Date : Mer 17/12/08	Tâche		Tâche reportée		Tâches externes	
	Avancement		Jalon reporté		Récapitulatif du projet	
	Jalon		Avancement reporté		Regrouper par en-tête	
	Récapitulative		Fractionnement		Échéance	

