

Projet de Génie Logiciel

Rapport de Planification

Année Académique 2014-2015

Groupe numéro: **7**

Membres du groupe: **DELGRANGE FLORENT, LECOQ ALEXIS,
LEMPEREUR MARTIN**

BAC 2 INFO

Faculté des Sciences, Université de Mons

`martin.lempereur@hotmail.be {florent.delgrange — alexis.lecoq}@student.umons.ac.be`

October 14, 2014

Ce rapport de planification est rendu dans le cadre du cours “Projet de Génie Logiciel” (dispensé par le Prof. *Tom Mens* en année académique 2014-2015). Le but de ce rapport est d’expliquer quels sont les objectifs du projet, quelles sont les différentes tâches à réaliser et planifier ces différentes tâches.

1 Introduction

1.1 Objectifs

Le travail s'inscrit dans le cadre du projet de Génie logiciel.

Le travail consiste à concevoir un jeu de rôle dont l'univers est celui de l'Université.

Nous concevrons donc une application java en décomposant la réalisation en plusieurs étapes clés.

1.2 Exigences fonctionnelles

Nous devons implémenter au moins une quête principale et secondaire de l'année d'étude BAC2 et un framework permettant au joueur de réaliser les différentes quêtes implémentées.

1.3 Exigences non-fonctionnelles

L'application doit être exécutable sur les systèmes exploitation Linux, Mac et Windows.

Le framework pourra supporter l'ajout de quêtes, de caractéristiques et même de classes réalisées par d'autres personnes ayant respecté l'interface commune.

1.4 Contraintes de temps

Les dates importantes sont :

Mardi 14 octobre 2014 : date de remise pour la phase 1 : Rapport de planification et cahier des charges.

Dimanche 30 novembre 2014 : date de remise pour la phase 2 (rapport de modélisation et maquette de l'interface utilisateur) .

Fin de la semaine du 10 ou début de la semaine du 17 décembre 2014 : réunions d'inspection de modélisation.

Jeudi 3 avril 2014 : date de remise de la phase 3 : Implémentation.

Les dates pour la remise de la phase 4 et de la défense orale et des tests d'acceptation sont encore à déterminer.

Remarque : du lundi 22 décembre au vendredi 2 janvier, le blocus commence, il sera donc extrêmement difficile d'être productif pendant cette période.

1.5 Contraintes de budget

Aucun achat n'est prévu. De plus, tous les logiciels indispensables à la réalisation du projet sont soit gratuits, soit disponibles avec une clé académique sur moodle.

2 Ressources

2.1 Les ressources humaines (personnel)

nom	rôle	durée
Florent Delgrange	développeur	environ 8 mois
Martin Lempereur	développeur	environ 8 mois
Alexis Lecoq	développeur	environ 8 mois
Tom Mens	Titulaire du projet	
Mathieu Goeminne	Enseignant	
Romuald Deshayes	Assistant	

Table 1. Ressources humaines.

2.2 Les ressources logicielles

Les logiciels utilisés dans le cadre du projet sont :

Gantt Project (gratuit), utilisé pour réaliser les diagrammes de Gantt et de PERT. <http://www.ganttproject.biz/>

Visual Paradigm (licence académique), utilisé pour réaliser les diagrammes lors de la phase 2 (modélisation ; diagramme de cas d'utilisation, de classe, d'état, de séquence,...)

<http://www.visual-paradigm.com/>

IntelliJ (licence académique), environnement de développement. <https://www.jetbrains.com/idea/>

Eclipse (gratuit), environnement de développement. <https://www.eclipse.org/>

Git (gratuit), logiciel de gestion de versions. <http://git-scm.com/>

Maven (gratuit), gestion et automatisation de production des projets logiciels Java.

<http://maven.apache.org/>

Vim (gratuit), éditeur de texte. <http://www.vim.org/>.

Kile (gratuit), éditeur de texte pour les documents écrits en Latex. <http://kile.sourceforge.net/>

JUnit (gratuit), bien qu'il soit intégré par défaut dans Eclipse et IntelliJ, il est utile de spécifier que nous utilisons ce framework pour réaliser nos tests unitaires. <http://junit.org/>

LibGDX (gratuit), framework utilisé pour concevoir l'interface graphique. <http://libgdx.badlogicgames.com/>

Tiled Map Editor (gratuit), outil ayant pour objectif de créer des cartes où le joueur pourra se déplacer (interface graphique). <http://www.mapeditor.org/>

2.3 Les ressources matérielles

Nom	Systeme d'exploitation	Memoire	Processeur	Carte graphique	Disque dur
Lempereur Martin	Linux	2 Go	Intel Core 2 Quad CPU Q6600 @ 2.40GHz 4		300 Go
Delgrange Florent Pc 1	Windows 8.1 pro	8 Go 1600 MHz DDR3	Intel Core i5 3470 CPU @ 3.20 GHZ	His Radeon R9 280X	1To
Delgrange Florent Pc 2	Mac OS X 10.9.5	4 Go 1067 MHz DDR3	2.53 GHz Intel Core 2 Duo	NVIDIA GeForce 9400M 256 Mo	
Lecoq Alexis	Smoby	VTech	2Ko		

Table 2. Ressources materielles

3 Analyse des risques

3.1 Identification des risques

Quels sont les risques potentiels du projet? Classifiez les risques par catégorie (par exemple les risques liés au personnel, au produit, au projet; d'autres classifications sont permises). Distinguez les risques génériques (au moins 4), applicables à tout projet (informatique), des risques spécifiques (au moins 4) pour votre propre projet. Donnez une description textuelle de chaque risque.

Prioritisez les risques selon leur importance, en fonction de leur probabilité et sévérité. Vous devez justifier la probabilité et sévérité de chaque risque. Si vous le souhaitez, vous pouvez utiliser une table pour résumer les risques retenus, comme par exemple la Table 3.

Risque	Probabilité	Sévérité
risque 1	basse	non significative
risque 2	haute	tolérable
risque 3	modérée	sérieuse
risque 4	basse	catastrophique

Table 3. Analyse des risques.

3.2 Gestion des risques

Pour chaque risque que vous avez retenu dans la section 3.1 (en fonction de son importance), expliquez comment vous comptez

1. éviter que le risque se produira
2. vérifier si le risque s'est produit
3. résoudre ou réduire l'ampleur du risque quand il s'est produit

4 Répartition du travail

4.1 Work Breakdown Structure

Mettez la répartition des tâches et sous-tâches, ou l'horaire du projet ici. Indiquez également qui fait quelle tâche. Pour ce faire, vous pouvez inclure une figure ici (extrait d'un outil de planification), ou vous pouvez mettre les informations dans un tableau de tâches, comme par exemple Table 4.

ID	Tâche	Dates	Responsable	% travail	Autre remarques
T1	établir le cahier de charges		... l'étudiant ...	100%	
T2	description tâche 2				
T3	description tâche 3				

Table 4. Tableau des tâches.

4.2 Etapes clés

Quels sont les étapes clés (milestones)? Quelles sont les livrables (si présent) pour chaque étape clé? Donnez une description du contenu de chaque livrable. Vous pouvez utiliser un tableau d'étapes clés pour représenter une partie de l'information requise, comme par exemple Table 5.

Date	étape clé	Livrables
31 octobre	réunion d'inspection de la planification de projet	rapport de planification

Table 5. Tableau d'étapes clés.

5 Ordonnancement

5.1 Diagramme GANTT

Vous devriez inclure et discuter ici un ou plusieurs diagrammes de GANTT, comme ceux montrés dans les Figures 1 et 2. À vous de choisir comment générer le(s) diagramme(s). Précisez le ou les outils utilisés pour la création de chaque diagramme.

Fig. 1. Exemple d'un diagramme GANTT.

Fig. 2. Un autre exemple d'un diagramme GANTT.

N'oubliez pas d'indiquer, sur le diagramme, les étapes clés et les dépendances entre les tâches. Si l'outil que vous avez utilisé ne permet pas de faire cela, vous pouvez mettre cette information dans une table séparée, comme par exemple Table 6.

Tâche	Prédecesseurs
id tâche 1	
id tâche 2	
id tâche 3	

Table 6. Dépendances entre les tâches.

5.2 Diagramme PERT

Vous devriez inclure et discuter ici un ou plusieurs diagrammes PERT, comme ceux montrés dans les Figure 3 et Figure 4. À vous de choisir comment générer ce diagramme. Précisez le ou les outils utilisés pour la création de chaque diagramme.

Si l'outil que vous avez utilisé pour générer le diagramme PERT le permet, vous devriez indiquer dans le diagramme, pour chaque tâche, la durée, l'effort (en personne/mois), le temps "Earliest Start (ES)", le temps "Latest Start (LS)", le temps lâche (*Slack Time (ST)*), et le *Free Float (FF)*. Le diagramme doit également montrer où se trouve le(s) chemin(s) critique(s).

Si l'outil que vous avez utilisé ne le permet pas vous devriez mettre ces informations dans une table séparée, comme par exemple la Table 7.

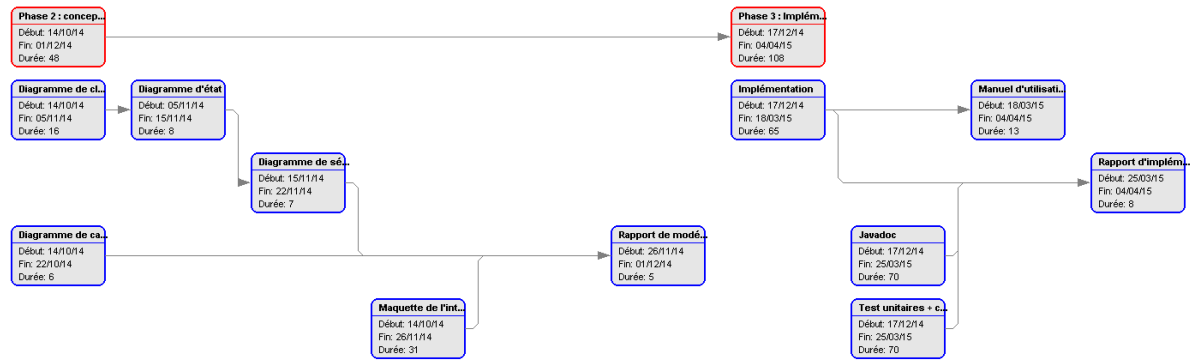


Fig. 3. Exemple d'un diagramme PERT.

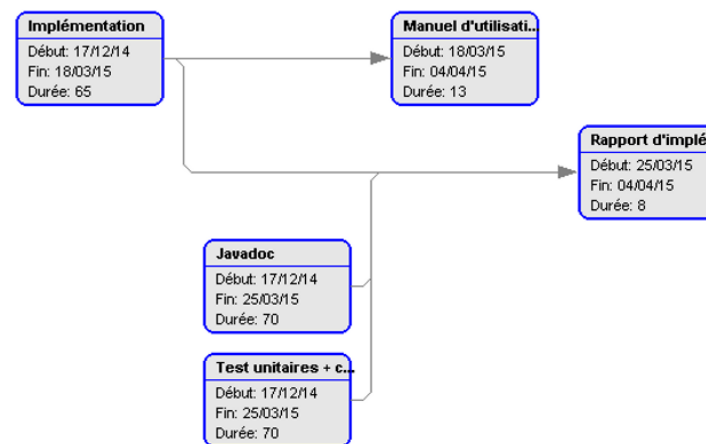


Fig. 4. Un autre exemple d'un diagramme PERT.

5.3 Analyse de l'ordonnancement

Analysez les diagrammes en calculant le chemin critique, les temps lâches (marge totale) et la marge libre (free float), tout en vérifiant que vous avez fait un bon choix. (Si

Tâche	Durée (jours)	Effort	Earliest Start	Latest Start	Slack Time	Free Float
Diagramme de classes	16	Eleve	14/10/14	14/10/14	0	0
Diagramme de cas d'utilisation	6	Moyen	14/10/14	24/11/14	41	41
Diagramme d'états	8	Moyen	05/11/14	05/11/14	0	0
Diagramme de séquences	7	Moyen	15/11/14	15/11/14	0	0
Maquette de l'interface	31	Eleve	14/10/14	14/10/14	0	0
Rapport de modélisation	5	Moyen	26/11/14	26/11/14	0	0
Implementation	65	Eleve	17/12/14	17/12/14	0	0
Javadoc	70	Faible	17/12/14	17/12/14	0	0
Tests unitaires	70	Eleve	17/12/14	17/12/14	0	0
Manuel d'utilisation	13	Faible	18/03/15	18/03/15	0	0
Rapport d'implémentation	8	Moyen	25/03/15	25/03/15	0	0

Table 7. Informations temporelles importantes sur les tâches.

non, vous devriez modifier votre ordonnancement afin de l'optimiser...) Justifiez votre décision.

5.4 Surveillance

Expliquez comment vous comptez surveiller les retards éventuels du projet quand il est en cours de route. Quelle procédure allez vous suivre pour détecter et éviter ces retards? Qu'allez vous faire quand un retard se manifeste?

Expliquez également comment vous comptez surveiller les risques.