

Projet d'optimisation linéaire

Activité d'Apprentissage I-MARO-035

Membres du groupe :
DOM Eduardo , PALGEN Arnaud

Année Académique : 2017 - 2018
BAC 2 en Sciences Informatiques

Faculté des Sciences, Université de Mons

6 novembre 2017

Résumé

Ce *rapport* est rendu dans le cadre de l'AA I-MARO-035 "Optimisation linéaire", dispensé par le Prof. *Nicolas Gillis* en année académique 2017-2018. Le but de ce rapport est de présenter la réalisation de notre projet.

1 Introduction

1.1 Objectifs

Le travail s'inscrit dans le cadre du cours de génie logiciel.

Il consiste à planifier, modéliser et implémenter un Laser Maze selon les consignes présentes sur Moodle.

1.2 Exigences fonctionnelles

Il nous est demandé de développer un moteur de jeu qui permet au joueur de réaliser les différents niveaux inclus (au moins 10) dont chacun doit être dans un fichier de description au format XML. Chaque étudiant doit implémenter individuellement une extension ainsi que des niveaux correspondant à celle-ci. Ces extensions doivent toutes être intégrées au jeu et doivent pouvoir être activées et désactivées simultanément. Les exigences détaillées sont présentes sur Moodle.

1.3 Exigences non-fonctionnelles

La fonctionnalité, la complétude et la qualité des livrables seront évaluées par les enseignants après l'échéance imposée pour chaque étape clé.

1.4 Contraintes de temps

Premièrement, il nous est demandé de respecter les dates d'échéance des étapes clés (Table 8). En plus, nous devons concilier ce projet avec le reste des cours et des éventuels projets (Optimisation linéaire, Base de données, Réseaux, ...). Donc, nous disposons essentiellement de notre temps libre et des vacances pour le développement du projet.

1.5 Contraintes de budget

Aucun achat n'est requis pour la réalisation de ce projet.

2 Ressources

2.1 Les ressources humaines (personnel)

Nom	Rôle	durée	Responsabilité	Pourcentage du temps (Par rapport au temps total)
AMEZIAN Aziz	Développeur	8 mois	Planification, Modélisation, Implémentaion	33 %
DHEUR Victor	Développeur	8 mois	Planification, Modélisation, Implémentaion	33 %
DOM Eduardo	Développeur	8 mois	Planification, Modélisation, Implémentaion	33 %
MENS Tom	Professeur	8 mois	Supervision	*
DUBRULLE Jérémy	Assistant	8 mois	Supervision	*
DEVILLEZ Gauvain	Assistant	8 mois	Supervision	*

TABLE 1 – Ressources humaines.

* Il nous est impossible de connaitre le pourcentage de temps pour les professeurs et assistants.

2.2 Les ressources logicielles

Nom	Type	Rôle	Version	License
ProjectLibre	Outil de planification	Planification	1.7.0	CPAL
Visual Paradigm	Environnement de développement UML	Modélisation UML	14.2	Community Edition
Yakindu	Outil de modélisation de diagrammes d'états	Modélisation de diagrammes d'états	2.9.3	Eclipse Public License
Java	Langage de programmation	Implémentation	1.8	GNU GPL
libGDX	Framework de développement de jeux	Implémentation	1.9.6	Apache License 2.0
IntelliJ IDEA	IDE	Implémentation	2017.2.1	Community Edition
Eclipse	IDE	Implémentation	4.7	Eclipse Public License
Git	Gestionnaire de versions	Gestion de versions	2.14.2	GNU GPLv2

TABLE 2 – Différentes ressources logicielles utilisées

2.3 Les ressources matérielles

Nom	Processeur	RAM	GPU	OS	Disque dur
Dom Eduardo	Intel Core i5 7300HQ	6Go	NVIDIA GeForce GTX 1050	Windows 10	HDD 1To
Dheur Victor	Intel Core i5 7200U	8Go	NVIDIA GeForce 940MX	Windows 10	SSD 512Go
Amezian Aziz	Intel Core i7 7500U	8Go	NVIDIA GeForce 920MX	Windows 10	SSD 120Go

TABLE 3 – Spécifications des machines des membres du groupe

3 Analyse des risques

3.1 Identification des risques

Nous avons exprimé l'importance d'un risque en fonction de la probabilité et la severité de celui-ci.

	Très basse	Basse	Modérée	Haute	Très haute
Non significative	1	1	2	3	4
Faible	1	2	3	4	5
Tolérable	2	3	4	5	5
Sérieuse	3	4	5	5	5
Catastrophique	4	5	5	5	5

TABLE 4 – Importance des risques.

Risque	Catégorie	Probabilité	Sévérité	Importance
Mauvaise interprétation des consignes	Personnel	Modérée	Catastrophique (Le projet ne respecterait plus les critères)	5
Mauvaise gestion du temps	Personnel	Modérée	Catastrophique (Impossibilité de respecter les deadlines)	5
Présence de bugs importants	Logiciel	Basse	Catastrophique (Certaines fonctionnalités pourraient ne pas satisfaire les consignes)	5
Surcharge de travail due à d'autres cours	Personnel	Haute	Sérieuse (Cela limite le temps dédié au projet)	5
Indisponibilité d'un des membres du groupe	Personnel	Basse	Sérieuse (Cela augmenterait la charge de travail des autres membres)	4
Mauvaise communication entre les membres du groupe	Personnel	Basse	Sérieuse (Cela entraînerait des incohérences au niveau du code, et également une mauvaise coordination dans la gestion du temps.)	4
Panne d'une de nos machines	Materiel	Basse	Tolérable (Possible perte du projet)	3

TABLE 5 – Identification des risques génériques, triés par importance.

Risque	Catégorie	Probabilité	Sévérité	Importance
Non-détection de niveaux impossibles par le level checker	Logiciel	Modérée	Catastrophique (Un tel dysfonctionnement est inacceptable vu le rôle du level checker)	5
Création de niveaux incohérents par le level editor	Logiciel	Modérée	Catastrophique (Un tel dysfonctionnement est inacceptable vu le rôle du level editor)	5
Connaissance insuffisante de LibGDX	Personnel	Modérée	Tolérable (Mauvaise utilisation des outils)	4
Conflits entre les différentes extensions	Personnel	Basse	Sérieuse (C'est gênant mais l'utilisation de chaque extension de façon isolée réglerait le problème)	4
Lenteur du level checker sur certains niveaux	Logiciel	Modérée	Tolérable (Cela pourrait être gênant mais ça reste dans les limites de l'acceptable)	4

TABLE 6 – Identification des risques spécifiques, triés par importance.

3.2 Gestion des risques

Risque	Prévention	Vérification	Solution
Panne d'une de nos machines	Utilisation d'un logiciel de gestion de versions	/	Recuperation des données grâce au logiciel de gestion des versions
Indisponibilité d'un des membres du groupe	/	Communication entre les membres du groupe	Nouvelle planification du projet
Mauvaise interprétation des consignes	Poser des questions aux professeurs et assistants	Questions aux professeurs/assistants	Recommencer les parties du projet correspondantes
Mauvaise gestion du temps	Bonne planification	Se référer au rapport de planification	Nouvelle planification du projet (Meilleure gestion du temps)
Présence de bugs importants	Tests manuels et Tests unitaires	Tests manuels et Tests unitaires	Débogage
Surcharge de travail due à d'autres cours	Bonne gestion du temps	Bonne communication entre les membres du groupe	S'y prendre à l'avance
Connaissance insuffisante de LibGDX	Lecture préalable de la documentation de LibGDX	Niveau d'aisance lors de l'utilisation de LibGDX	Lecture de la documentation, tutoriaux, forums d'aide
Mauvaise communication entre les membres du groupe	Mise au point régulière de l'avancement du projet	Cela se reflète sur la qualité du projet	Meilleure communication
Conflits entre les différentes extensions	Lecture préalable des consignes	Lors de l'implémentation des extensions ou de l'exécution du programme	Implémenter des extensions n'ayant pas de conflits
Lenteur du level checker sur certains niveaux	Test de quelques niveaux avec le level checker	Test exhaustif des niveaux avec le level checker	Une meilleure implémentation du level checker
Non-détection de niveaux impossibles par le level checker	Tests unitaires et manuels	Créer un niveau impossible et utiliser le level checker	Une meilleure implémentation du level checker
Création de niveaux incohérents par le level editor	Tests unitaire et manuels	Essayer de créer des niveaux incohérents	Une meilleure implémentation du level editor

TABLE 7 – Gestion des risques

4 Répartition du travail

4.1 Work Breakdown Structure

ID	Tâche	Durée	Responsable	% travail
T1	Planification	24 jours	Dom Eduardo, Amezian Aziz, Dheur Victor	20 %
T2	Modélisation	40 jours	Dom Eduardo, Amezian Aziz, Dheur Victor	20 %
T3	Jeu de base : logique	40 jours	Dom Eduardo, Amezian Aziz, Dheur Victor	30 %
T4	Jeu de base : graphisme	40 jours	Dom Eduardo, Amezian Aziz, Dheur Victor	30 %
T5	Test du jeu de base et débogage	4 jours	Dom Eduardo, Amezian Aziz, Dheur Victor	20 %
T6	Editeur : logique	30 jours	Amezian Aziz	30 %
T7	Editeur : graphisme	15 jours	Amezian Aziz	30 %
T8	Extension couleurs et intensité : logique	25 jours	Dom Eduardo	30 %
T9	Extension couleurs et intensité : graphisme	20 jours	Dom Eduardo	30 %
T10	Level Checker : logique	35 jours	Dheur Victor	30 %
T11	Level Checker : graphisme	10 jours	Dheur Victor	30 %
T12	Intégration des extensions	3 jours	Dom Eduardo, Amezian Aziz, Dheur Victor	60 %
T13	Tests finaux et débogage	8 jours	Dom Eduardo, Amezian Aziz, Dheur Victor	40 %

TABLE 8 – Tableau des tâches.

4.2 Etapes clés

Date	Etape clé	Livrables
22/10/2017	Planification	Rapport de planification
03/12/2017	Modélisation	Rapport de modélisation et maquette de l'interface utilisateur
30/03/2018	Implémentation	Code source, code compilé, tests unitaires, JavaDoc, vidéo d'utilisation, rapport d'implémentation

TABLE 9 – Tableau d'étapes clés.