Rapport de projet

DUT Informatique - Projet Tortuga

2011-2012

Introduction

Dans le cadre de notre DUT Informatique, nous avons participer en groupe à la réalisation d'un projet tuteuré, où nous avons développé une application reprenant le jeu de société **Tortuga**.

Après une présentation complète du sujet et du jeu, nous étudierons dans ce rapport le cahiers des charges, nous verrons comment a été réalisé le programme, nous comparerons le planning prévu et le planning effecif, et nous terminerons par une revue des problèmes rencontrés, ainsi qu'un bilan.

Le projet a été réalisé par des braves types :
- Sacha du Bourg Palette ;
- Pierre d'Argenta ;
- Ondine d'Azuria ;
- ...

Table des matières

Chapitre 1

Présentation du sujet

Le but de **Tortuga**, le jeu de société que nous adaptons, est très simple à comprendre. Deux joueurs adverses contrôlent une équipe de 8 tortues se faisant face, sur un plateau de forme hexagonale. L'objectif à atteindre est la case située à la pointe de l'hexagone juste en face de la position initiale. **Tortuga** est un jeu de stratégie, où deux adversaires s'affrontent, et ne laisse logiquement aucune place au hasard.

La progression des tortues sur le plateau est semblable au jeu de dames : chacun leur tour, les joueurs déplacent une de leur tortue (obligatoirement vers l'avant) en la faisant simplement avancer d'une case, ou en sautant par dessus une autre tortue. A contrario du jeu de dames, il existe une notion de « capture de tortue » dans **Tortuga**, renforçant considérablement l'aspect stratégique : lors d'un déplacement de tortue, si celle-ci saute par dessus une tortue adverse, la tortue sautée est alors retournée et devient neutre. Ces tortues neutres sont alors « désactivées » et inutilisables jusqu'à ce qu'un joueur décide de sauter à nouveau cette tortue neutre. Lorsque que cela se produit, le joueur réactive alors la tortue neutre, en la retournant. Il a alors le choix : il peut ajouter la tortue à son équipe, ou la faire rejoindre le camp adverse, si cela lui parait être une meilleure stratégie !

Notre sujet de projet tuteuré est donc l'adaptation du jeu de société Tortuga. Le programme a été réalisé avec le langage de programmation Java, et l'interface graphique utilise la bibliothèque graphique Swing.

Chapitre 2

Cahiers des charges

2.1 Fonctionnalités et différents modes de jeu

Tortuga est un jeu dit « de stratégie combinatoire abstrait » (comme le jeu de dames ou des échecs), faisant s'affronter deux joueurs qui agissent à tour de rôle. Les modes de jeux apparaissent donc clairement :

- un mode où un joueur humain affronte un autre joueur humain ;
- un mode où un joueur humain affronte une intelligence artificielle (IA).

Cependant, une variante du jeu existe, ajoutant une notion « d'éclosion ». Au début de la partie, aucunes tortues ne se trouvent sur le plateau. Lors du premier tour, chaque joueur doit placer une ou deux tortues (au choix) dans « son camp ». Au second tour, les joueurs ont alors le choix :

- faire éclore une ou plusieurs nouvelles tortues ;
- déplacer une de ses tortues.

Toutefois, les joueurs ne peuvent en tout faire éclore que 8 tortues au maximum (le nombre de tortues restantes est dans ce que l'on appelle le « stock »), et la différence du nombre de tortues dans le stock entre les deux joueurs ne peut être supérieur à deux (par exemple, si le joueur A a 6 tortues en stock, et le joueur B n'en possède plus que 4, le joueur B doit attendre que le joueur A fasse éclore de nouvelles tortues pour qu'il puisse lui aussi le faire).

Pour résumer, l'application comporte deux modes de jeux (le mode normal et le mode avec variante) et deux types de parties (contre un autre joueur humain ou contre une intelligence artificielle).

2.2 Contraintes

L'adaptation de **Tortuga** doit être réalisée avec le langage de programmation Java. L'interface graphique doit être fichtrement bien développée grâce à la bibliothèque graphique Swing.

L'aspect de l'application doit être la plus fidèle possible au jeu original : le plateau doit avoir la même forme (hexagonale) et les pions doivent être des tortues.

Enfin, le jeu doit bien évidemment reprendre et respecter les règles originales !

Chapitre 3

Réalisation

3.1 Charte graphique

Comme dit précédemment, l'application doit reprendre l'apparence du jeu de société **Tortuga**. Le plateau hexagonale est donc au centre de l'affichage, les tortues (en bas du plateau pour un joueur, en haut pour l'autre), qui dans le jeu original sont de la même couleur, ont été faites dans l'application avec des teintes bien différentes pour les différencier : des tortues rouges contre des tortues vertes.

Pour faciliter la jouabilité, les tortues que l'on peut déplacer sont entourées par un cercle jaune lorsque c'est au tour du joueur d'agir, et les déplacements qui sont possibles sont affichés par une tortue en transparence. Enfin, pour plus de sobriété et pour mettre en évidence les tortues, le fond de l'interface est en bleu foncé, et les cases du plateau sont bleues claires.

3.2 Architecture

Le projet a été divisé en différentes classes, parceque c'est cool

3.2.1 *public class Tortuga*

main de l'application

3.2.2 *public class Tortue*

On gère dans cette classe la tortue, en chargeant son image blabla (meme principe pour public class TortueRouge extends Tortue et public class TortueVerte extends Tortue)

3.2.3 *public class Plateau*

On gère le plateau (du coté graphique) : il est dessiné, on récupère les clics de souris pour les déplacements de pions, ...

3.2.4 *public class MainWindow*

Utilisée pour afficher l'interface graphique

3.2.5 *public class IA*

Calcul la liste des configurations possibles à partir d'une configuration (et après un saut par dessus une autre tortue, evaluation, algo alpha-beta

3.2.6 *public class Configuration*

On gère le plateau (du coté « programme ») : on gère les tableaux qui représente les pions. Gestion des placements de tortues possibles : méthode pour voir si les tortues peuvent encore bougées, une qui donne tous les mouvements possibles... C'est aussi ici que l'on change la valeur du tableau lors des déplacements

Chapitre 4

Dépassement du sujet : application smartphone

Parceque c'était beaucoup trop simple comme sujet de projet tut, on s'est dit que « Hey tiens, si on développait ça sur smartphone ! »
Et hop !

4.1 Application Android

4.2 Application Windows Phone 7 ? (gl PJ !)

Chapitre 5

Planning

Le projet a démarré lors de la fin de notre fin de semestre 2 (Mai 2011). Nous avons dès lors élu notre chef de projet : Pierre-Jean HUCKEL, qui a pris les décisions et réparties nos différentes tâches. Le développement du projet a donc commencé à notre rentrée en semestre 3, en septembre. Le projet fut divisé en 4 grandes étapes :

1. Programmation de l'interface graphique.
2. Programmation de la gestion du plateau
3. Définition d'un barème permettant d'étudier la qualité de la situation : barème de l'intelligence artificielle.
4. Programmation de l'intelligence artificielle

5.1 Planning prévu

5.2 Planing effectif

5.3 Répartitions des tâches

Tâche à réaliser	**Personne(s) ayant travaillées dessus**
La plupart du projet	PerJean
Etre un sale type	Thomas

Chapitre 6

J'ai pas de titre pour ça, *help !*

6.1 Trucs utilisés

Projet réalisé sous Eclipse, logiciel de gestion de versions Github, rapport en LaTeX, planning avec GanttProject, ...
Parceque l'open source, ça coule de source ! :D

6.2 Problèmes rencontrés

6.3 Bilan