

Université de Cergy-Pontoise

RAPPORT

pour le projet Génie Logiciel
Licence d'Informatique deuxième année

sur le sujet

Mettez le Sujet de Votre Projet Ici

rédigé par

Les auteurs ici



Mai 2015

Table des matières

1	Introduction	1
2	Spécification	2
2.1	Première sous-section	2
2.2	Deuxième sous-section	2
3	Réalisation	2
3.1	Le model - présentation des classes	2
3.1.1	La ville	2
3.1.2	La population	2
4	Manuel Utilisateur	3
5	Déroulement du projet	3
6	Conclusion	3

Table des figures

1	Classe abstraite Character	3
---	--------------------------------------	---

Liste des tableaux

1	Documents à remettre	2
2	Taille de la population	2

Remerciements

Les auteurs du projet voudraient remercier...

1 Introduction

Le projet Le projet consiste à la création d'un jeu vidéo simulant une vie urbaine, dans laquelle plusieurs individus vivent leur vie au sein d'une ville. L'utilisateur pourra influencer sur le comportement des individus et les paramètres de la ville.

Fonctionnalités Fonctionnalités du programme : Le joueur aura plusieurs actions possibles afin d'influencer sur l'évolution de la ville. Tout d'abord il pourra agir sur le temps en l'accélération ou en le mettant en pause. Ensuite il pourra accéder à des informations sur les personnages comme : leurs informations de bases, leur historique ou leurs objectifs directs (exemple : ce personnage se rend à la piscine). Ces informations pourront être changées par l'utilisateur et il pourra ainsi renommer un personnage, le faire déménager ou le faire rentrer chez lui par exemple. De même pour les bâtiments, l'utilisateur pourra accéder à ses informations et les modifier. Il pourra donc par exemple modifier les horaires d'ouverture d'un lieu ou faire varier son nombre d'utilisateur maximum. Le joueur pourra donc modifier à sa guise ses informations et voir ce que ces modifications apportent de bon ou de mauvais sur la population de la ville.

Nos motivations Nous avons choisi ce projet car il représente une opportunité pour chacun de nous d'explorer des domaines/notions qui nous intéressent, et dans lesquelles nous voulons nous perfectionner.

2 Spécification

Chapeau Nous avons présenté l'objectif du projet dans la section 1. Dans cette section, nous présentons la spécification de notre logiciel réalisé. Ceci correspond principalement au cahier des charges.

2.1 Première sous-section

Premier paragraphe On commence à expliquer...

Juste un simple paragraphe.

2.2 Deuxième sous-section

Document	Coefficient	Commentaire
Cahier des charges	37.5%	Premier document
Rapport	62.5%	Rapport final du projet

TABLE 1 – Documents à remettre

Comme ce qui est illustré dans le tableau 2, ...

3 Réalisation

3.1 Le model - présentation des classes

3.1.1 La ville

3.1.2 La population

La population est le second ensemble qui constitue notre projet. Nous l'avons voulu le plus adaptatif possible. Cependant la création de la population depend de la ville car, pour le mode normal, chaque maison ainsi que chaque lieu de travail ne peut contenir qu'un seul personnage (par soucis de réalisme et de bonne répartition de la population sur la carte). Pour le mode autonome, au lancement du jeu, chaque maison ne peut avoir qu'un seul personnage. Au cours de l'evolution de la population plusieurs personnages pourront habiter dans une même maison. Notre population est donc limitée à 15 personnages maximum pour le mode normal (car 15 batiments de travail) et 60 en mode autonome (car 60 maisons) Si l'on décide d'augmenter la taille de la ville ou d'enlever les limitations de personnages par maison, la ville pourrait contenir une plus grande population. Cependant, une trop grande population provoquera des ralentissement de l'interface graphique mais le moteur de jeu est parfaitement capable de faire tourné une grande population.

Mode normale	Mode autonome
niveau easy : 1	[1 - 60]
niveau normale : 3	
niveau hard : 5	
niveau pro : 15	

TABLE 2 – Taille de la population

Création des personnages Un personnage est composé d'information de base comme d'un nom, d'un prénom, d'un sexe , d'un age et d'un numéro d'identité.

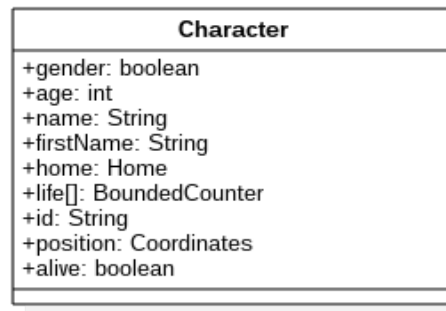


FIGURE 1 – Classe abstraite Character

Pour le nom, le prénom et le sexe du personnage, ils sont initialisés à partir de fichier CSV. Le premier fichier contient une liste de 300 noms de familles et le second une liste de 200 prénoms, 100 masculin et 100 féminin. Lors de la création du personnage, le programme va prendre au hasard un nom dans le fichier name.csv et un prénom (associé à un sexe) dans le fichier firstName.csv. L'âge du personnage est simplement choisis aléatoirement entre 10 et 100 ans. Le numéro d'identité du personnage est unique, il est calculé à partir de toutes les informations du personnage grâce à un code de hashage. Ce code nous permettra de reconnaître le personnage. Ce grand nombre de choix nous permet de garantir une grande diversité au sein de la population.

Comme le montre la figure 1, un personnage possède également une maison. Cet élément est également attribué aléatoirement via une recherche dans la liste de lieux d'habitation de la carte de jeu.

Un personnage possède également un tableau de jauges contenant :

- une jauge d'émotion
- une jauge d'argent
- une jauge de fatigue

Ces jauges peuvent varier de 0 à 100. Ces jauges représentent les critères de vie du personnage. Elle est initialisé à 75 en début de partie mais elle variera en fonctions des actions des personnages. Si l'une des jauges du personnage arrive à 0 il meurt.

Nous avons voulu rendre nos personnages le moins statique possible. Ainsi d'une partie à l'autre, la chance de tomber sur des personnages avec les mêmes propriétés est très faible.

Les différents types de personnages

4 Manuel Utilisateur

Cette section est dédiée au manuel utilisateur.

5 Déroulement du projet

Dans cette section, nous décrivons comment la réalisation du projet s'est déroulée au sein de l'équipe de projet. La répartition des tâches, la synchronisation du travail et l'utilisation du temps seront abordées.

6 Conclusion

Dans cette section, nous résumons la réalisation du projet et nous présentons également les extensions et améliorations possibles du projet.

Références

- [1] L. M. Haas, E. T. Lin, and M. A. Roth. Data integration through database federation. *IBM Syst. J.*, 41(4) :578–596, 2002.