

**Deuxième année licence informatique :
2018/2019**

Rapport projet programmation orientée objet 2eme partie

MALUEKI-NGOMBI Jérémie 11605194

Université Paris 13 Institut Galilée
99 Avenue Jean Baptiste Clément
93430 Villetaneuse

Présentation générale :

Le jeu se nomme **Bavia**. Bavia est un jeu en mode texte qui se joue dans une invite de commande et reçoit les instructions par ce dernier.

Dans ce jeu votre but est de récupérer tout les objets royaux perdus a travers le royaume afin d'être couronné roi du royaume.

Ce jeu est une modification du jeu Zork dans le cadre du projet de programmation orientée objet. Ceci est la seconde partie du projet et le jeu a subit des modification

J'ai donc dû implémenter au jeu des classes importantes celle du "joueur" ,celle des objets nomme "ObjetZork". La classe « EtreAnime » hérite de celles des ObjetZork

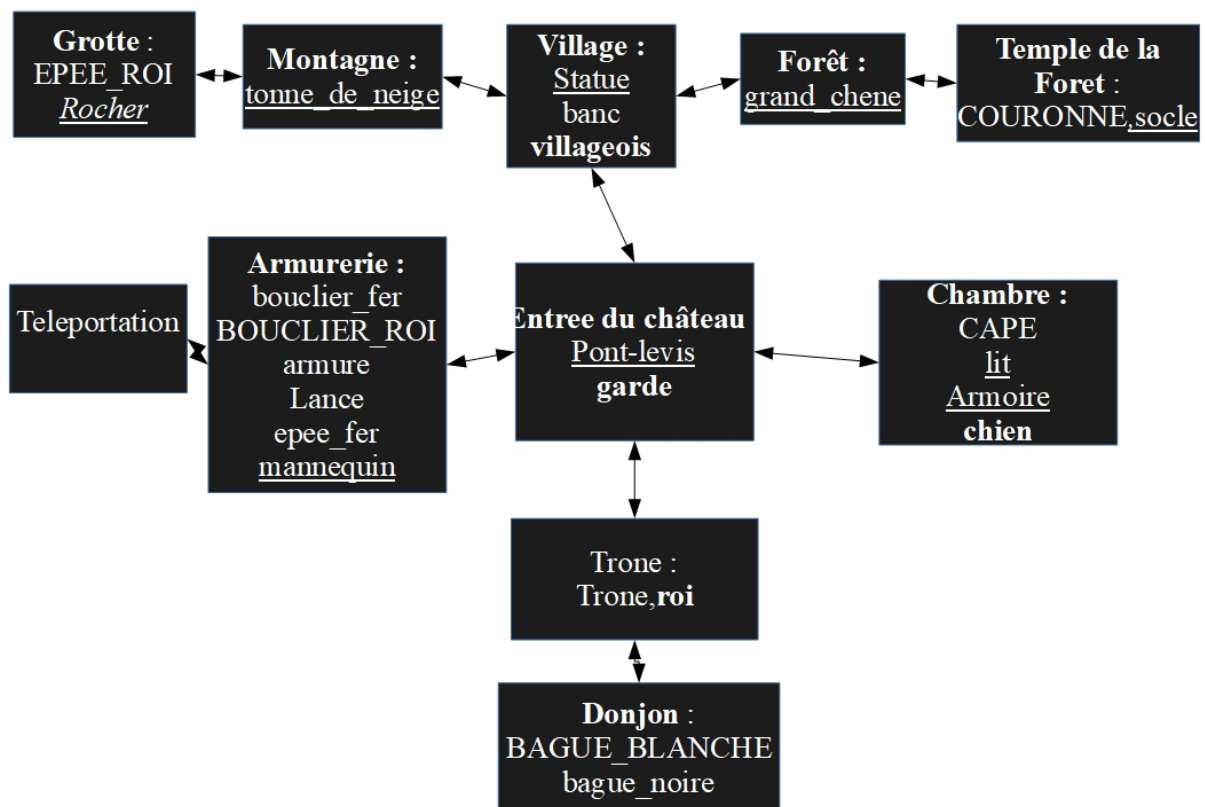
Listes des Commandes :

* (...) = paramètre de la commande

Commandes :	Description
<i>aller (nord,est,sud,ouest)</i>	Permet au joueur de changer de pièce
quitter	Permet de quitter le jeu
aide	Affiche une aide au joueur ; affiche toutes les commandes possibles et rappel au joueur son objectif.
prendre (Objet)	Permet de prendre un objet transportable dans le jeu renvoie une erreur sinon (ici un message est affiché)
deposer (Objet)	Permet de déposer un objet provenant de l'inventaire du

	joueur renvoie une erreur sinon (ici un message est affiché)
retour	Permet au joueur de revenir dans la pièce précédente
etat	Permet de voir les objets dans l'inventaire du joueur et le poids total des objets transportés
regarder	Permet de connaître les différent objets présents dans la pièce courante et les donne des indication sur le lieu
donner	Permet de donner un objet a un EtreAnime

-Plan du Jeu



les objets en majuscule font gagner le jeu et ceux souligner sont non transportable, enfin la bague noire fait perdre la partie

-Listes des Objets

Nom	Transportable	Lieux
epee_roi	Oui poids(3)	grotte
Rocher	non	grotte
Tonne_de_neige	Non	Montagne
Fontaine	Non	Village
grand_chene	Non	Foret
couronne	Oui	Temple de la foret
Socle	Non	Temple de la foret
Pont-levis	Non	Entre du château
cape	Oui	Chambre
Lit	Non	Chambre
Armoire	Non	Chambre
bouclier_fer	Oui	Armurerie
bouclier_roi	Oui	Armurerie
Lance	Oui	Armurerie
epee_fer	Oui	Armurerie
mannequin	Non	Armurerie
bague_blanche	Oui	Donjon
trone	Non	trone
bague_noire	Oui	Donjon
Armure	Oui	armurerie
roi	Non(EtreAnime)	trone
garde	Non(EtreAnime)	chateau,
Chien	Non(EtreAnime)	chambre

*Les objets en gras sont des objets qui permettent de gagner le jeu

-Scenario de test

il y a deux scenarii de test un perdant et un autre gagnant.

Scenario perdant : pour perdre il suffit de prendre la bague noire situé dans le donjon

depuis le château taper les commandes :

> aller sud ; x2

> prendre bague_noire

Le jeu vous informera que vous avez perdu puis quitte le jeu.

Scenario gagnant: Pour gagner il faut récupérer les objets gagnant, les ramener sur soi a la salle du trône et donner la couronne au roi.

Les objets sont : « bague_blanche », « bouclier_roi, » « cape », « epee_roi », et la « couronne »

Pour gagner taper les commandes suivante depuis le château

>aller est

>prendre cape

>aller ouest

>aller ouest

>prendre bouclier_roi

>aller est

>aller sud

>aller sud

>prendre bague_blanche

>aller nord

>aller nord

>aller est

>aller est

>prendre couronne

>aller ouest

>aller ouest

>aller ouest

>aller ouest

>prendre epee_roi

>aller est

>aller est

>aller sud

>aller sud

>donner couronne roi ;

Le jeu affichera un message vous indiquant que vous avez gagné puis quitte le jeu.

modifications apportées au logiciel :

-Listes des nouvelles classes :

la classe ObjetZork : la Classe centrale du jeu car elle permet la création des objets dans le jeu. Les objets sont définis par 3 arguments : leurs poids leur nom et si ils sont transportable ou non

la classe joueur : a été crée afin d'initialiser un joueur et de pouvoir gérer les objets. La classe permet également au joueur d'avoir un inventaire sous la forme d'une ArrayList d'ObjetZork.

Le joueur est caractérisé par son nom le poids actuel le poids maximum prédéfini a l'avance et une ArrayList d'ObjetZork.

l'interface « Conteneur » et la classe « ArrayListConteneur » : la classe ArrayListConteneur a été ajoutée afin de factoriser le code du jeu et amener à une meilleure compréhension de celui-ci. Elle contient une ArrayList d'ObjetZork qui est utilisée dans les classes héritées.

Elle regroupe donc toutes les fonctions permettant de manipuler les conteneurs d'objets des classes « Piece » et « Joueur » et ces deux classes héritent de celle-ci.

EtreAnime:

la classe EtreAnime hérite de la classe ObjetZork. Cette classe peut transporter des ObjetZork grâce à une ArrayList et peut les déplacer dans le jeu.

-Listes des classes modifiées :

Les classes Piece et Joueur : ont été modifiées elles héritent désormais de ArrayListConteneur. Les fonctions qu'elles avaient en commun (ajouter,retirer,contient,getNbObjet...) sont désormais

définies dans sa classe mère ArrayListConteneur.

Les anciennes méthodes prendre et depose qui servaient a l'échange d'objets entre le joueur et les pièces ont été remplacé par la méthode : `public ObjetZork chercheObjet()`

Méthode qui cherche un ObjetZork present dans le conteneur a l'aide d'une chaîne de caractères qui correspond a la description de l'ObjetZork.

Renvoie l'ObjetZork si celui-ci est présent dans le conteneur

La classe Jeu :

La classe qui a sans doute subit le plus de modification. Les nouvelles méthodes retour etatJoueur ,prendre, déposer, regarder correspondent aux nouvelle commandes ajoutées

`public void etatJoueur()`

la méthode affiche l'état du joueur c'est a dire le poids transporté et son inventaire

`public void regarder()`

sert à connaître les objets présents dans une pièce et donne en plus, des commentaires sur la pièce courante. Correspond à la commande regarder.

`public boolean gagner()`

Méthode qui vérifie si le joueur réunis les condition pour perdre

`public boolean perdre()`

méthode qui vérifie si le joueur réunis les condition pour perdre ici le joueur doit ramasser l'objet « bague_noire » afin de perdre

`public boolean teleportation()`

la méthode permettant au joueur de se téléporter dans une pièce du jeu au hasard. Pour cela il faut se déplacer dans la pièce a l'ouest de l'armurerie.

les méthode gagner et perdre sont utilisées dans la méthode jouer et sont dans même boucle while qui permet au jeu d'exécuter les commandes entrées par l'utilisateur, jusqu'à ce que la commande choisie soit la commande "quitter".
Donc si le joueur réunis les condition de victoire ou de défaite le jeu se fini.

La méthode traiter commande j'ai ajouter les commande a vu précédemment

La classe MotCommande :
J'y ai seulement ajouter les nouvelles commandes :retour etat ,prendre, déposer, regarder et donner dans le tableau commandesValides

Bug rencontré et remarques :

Je n'ai pas réussi a implémenté le mouvement des êtres animés