

Rapport Python

Lucas WU

Sofiane MOUHOUB

Sommaire:

I/ Guide utilisateur

II/ Guide développeur

III/ Perspectives de développement

I/ Guide utilisateur :

Scenario :

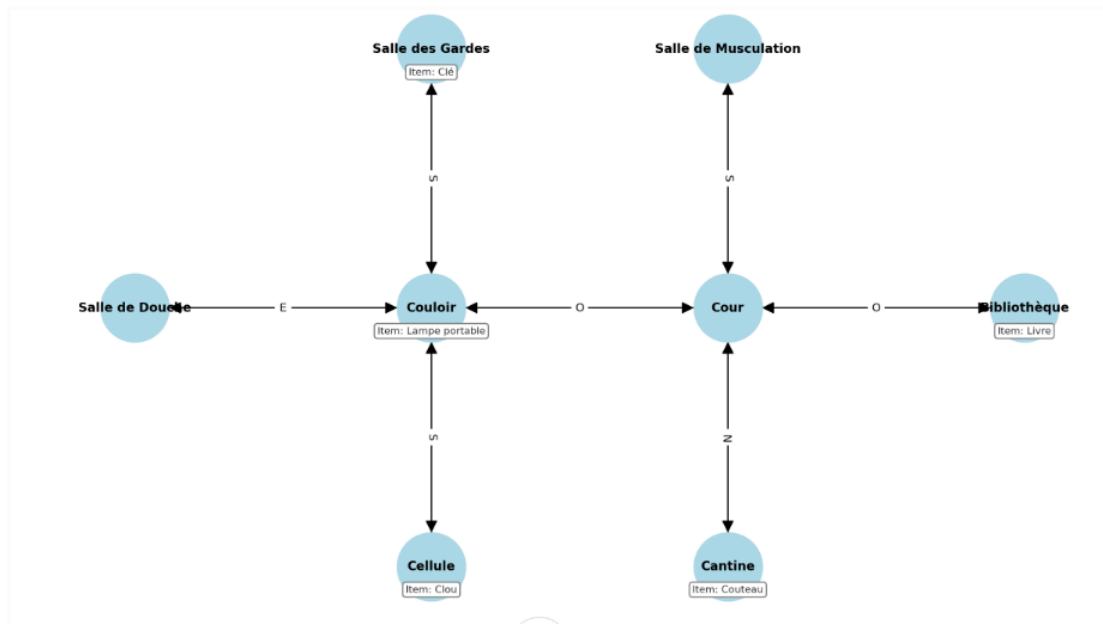
Notre jeu a été inspiré par la série culte américaine Prison Break qui consiste à s'évader d'une prison. Le joueur incarne un prisonnier accusé à tort. Chaque nuit, il explore des recoins de sa cellule pour trouver des objets utiles. Il doit résoudre des énigmes, pour avoir des objets afin d'amadouer au maximum le garde qu'il le laissera potentiellement sortir de prison.

Différents lieux :

Pour cela, on a voulu créer une prison classique avec différents lieux :

1. Cellule
2. Couloir
3. Salle des Gardes
4. Salle de douche
5. Bibliothèque
6. Salle de musculation
7. Une cour
8. Une cantine

Carte du jeu :



Comment jouer ?

Le jeu commence dans la cellule et on peut se déplacer dans les pièces que l'on souhaite en allant au nord, sud, est et l'ouest et prendre les objets permettant de sortir de la prison.

Condition de victoire :

Afin de gagner, il faut parler à Sébastien en ayant sûr soit un couteau et la clé issue de la salle des gardes.

Condition de défaite :

Si le joueur parle au garde George tout en ayant un couteau, vous perdez.

Les Commandes:

help : afficher cette aide

```
> help
```

Voici les commandes disponibles:

- help : afficher cette aide
- quit : quitter le jeu
- go<direction>: se déplacer dans une direction (N,E,S,O)
- back<direction> : retour à la dernière position
- inventaire<direction> :affiche l'inventaire
- look<direction> : affiche les objets de la salle
- take<direction> : Prend un objet
- drop<direction> : Dépose un objet
- talk<description> : Parle à un personnage

quit : quitter le jeu

```
> quit
```

Merci lucas d'avoir joué. Au revoir.

go, "<direction>: se déplacer dans une direction (N,E,S,O) "

```
> go N
```

Vous êtes Un long passage éclairé par une lampe au mur.

Sorties: N, S, E, O

back, "<direction> : retour à la dernière position"

```
> back
```

Vous êtes dans une cellule, petite pièce sombre avec un lit superposé .

Sorties: N

Vous avez déjà visité les pièces suivantes :

inventaire, "<direction> :affiche l'inventaire

```
> inventaire
```

Vous disposez des items suivants:

clou : Permet d'ouvrir la porte de la cellule. (0.14 kg)

look "<direction> : affiche les objets de la salle"

```
> look
```

La pièce contient :

- clou : Permet d'ouvrir la porte de la cellule.

take "<direction> : Prend un objet"

```
> take
```

```
La commande 'take' prend 1 seul paramètre.
```

```
drop "<direction> : Dépose un objet"
```

```
> drop
```

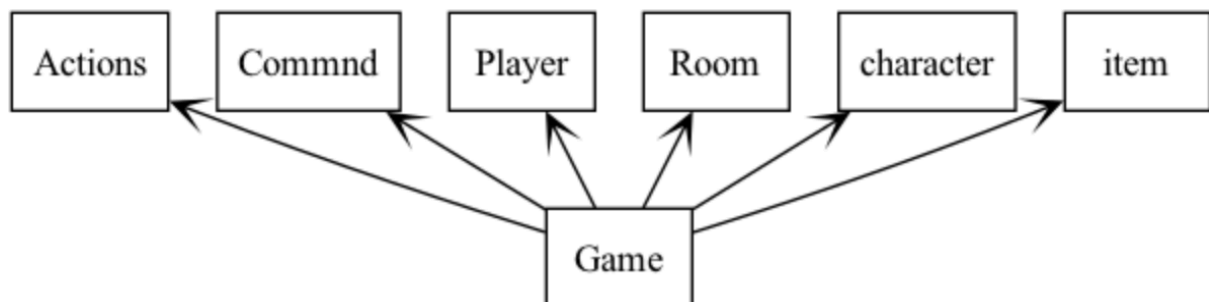
```
La commande 'drop' prend 1 seul paramètre.
```

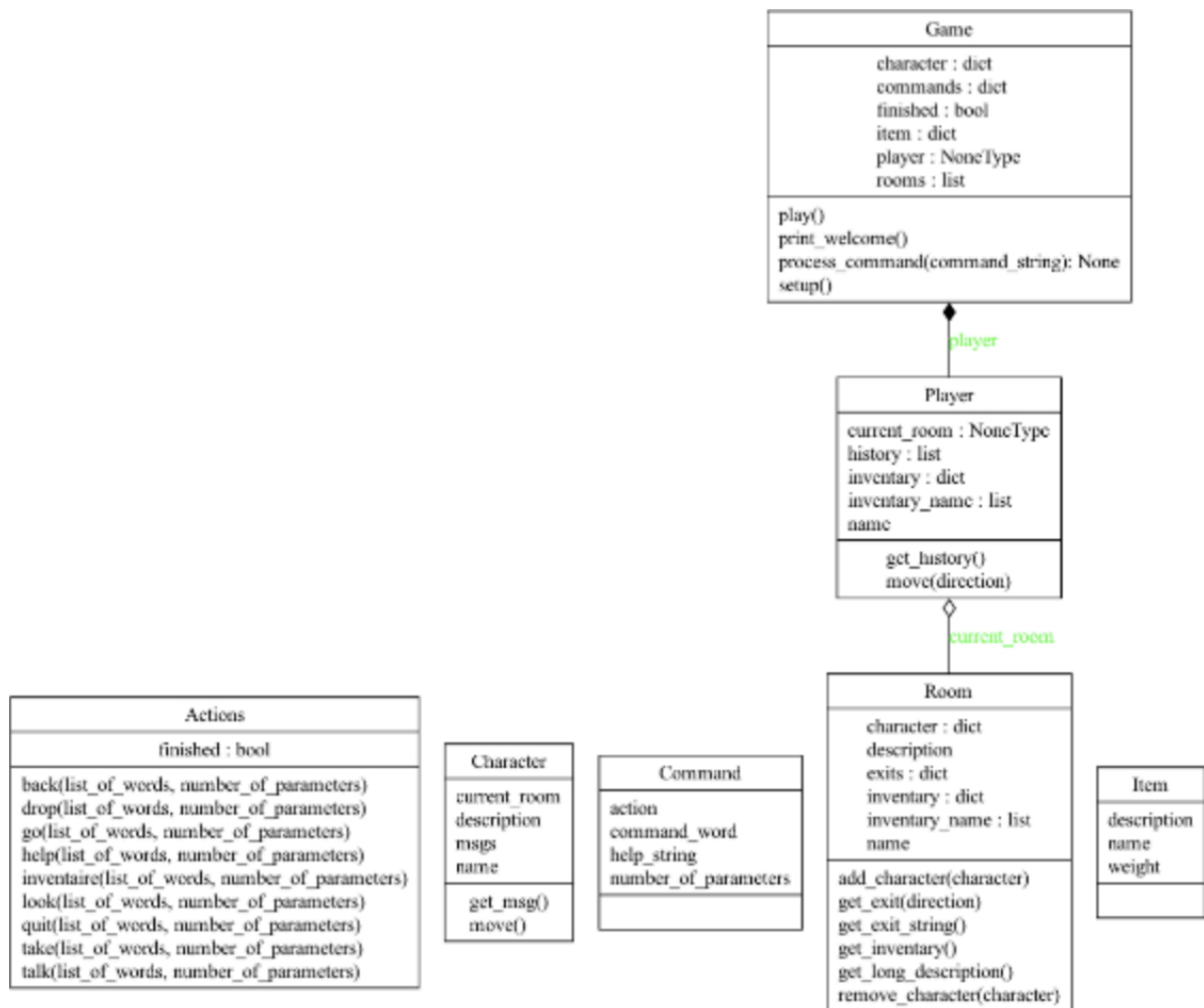
```
talk "<description> : Parle à un personnage"
```

```
> talk
```

```
La commande 'talk' prend 1 seul paramètre.
```

II/Diagramme des classes (utilisation de Pyreverse et Graphviz)





III/ Perspectives de développement

Au niveau du jeu, on pouvait l'améliorer de différentes façons :

Premièrement, on pourrait améliorer l'immersion grâce à des dialogues complexes influençant sur le déroulement de l'histoire, en créant potentiellement des énigmes dans chaque pièce pour le rendre plus "amusant". De plus, on pouvait rendre le jeu plus visuel tout en restant dans un cadre Python avec notamment la création d'une interface graphique. On ajoutera notamment plus de personnages pour plus de dialogues pour que ça soit un peu plus 'vivant'. Et pour finir une condition de victoire améliorée où le prisonnier sort véritablement de prison sans l'aide de garde tout en réussissant les défis qui lui ont été proposés.

Ces améliorations rendront le jeu plus immersif, interactif et engageant tout en exploitant les capacités offertes par Python.