**Documentation : Explication du fonctionnement de Bomerpy**

1. **Prérequis :**

Veuillez avoir déjà installé Python 3.10 ou une version plus récente, ainsi que les modules Ursina et Requests à l’aide des lignes de code suivante :

* pip install ursina
* pip install requests

1. **Modes de jeu :**

On accède au jeu en lançant le programme *bbm.py*. Le mode de jeu par défaut est le Bomberman classique où il faut être le dernier survivant sur la carte.

Une image contenant capture d’écran, texte, carré, Rectangle

Description générée automatiquement

*Document 3 : Mode par défaut du Bomberman*

Pour accéder à l’autre mode de jeu qui est le mode football, il faut procéder à un changement de la ligne 25 du code en remplaçant « br » par « foot ». Il suffit de procéder au changement inverse pour revenir au mode par défaut.

Une image contenant capture d’écran, texte, Logiciel multimédia, Police

Description générée automatiquement

Une image contenant capture d’écran, texte, Logiciel multimédia

Description générée automatiquement

*Document 4 : Remplacement de la ligne de code pour accéder à la ligne de code*

On obtient alors l’écran suivant où le but du jeu est de faire rentrer le ballon dans la cage adverse en le déplaçant à l’aide des bombes.

Une image contenant texte, capture d’écran, jeux

Description générée automatiquement

*Document 5 : Mode football du Bomberman*

1. **Le Jeu**

On peut y faire jouer jusqu’à quatre personnes. Chaque joueur possède trois vies. La seule façon d’en perdre une est de se faire toucher par une bombe (adverse comme la sienne), dans ce cas il réapparaitra à sa position de départ. Si un joueur se fait toucher trois fois, il n’apparaitra plus et ne pourra donc plus jouer. Pour détruire un bloc, il faut que celui-ci se trouve sur le rayon d’explosion d’une bombe. Un bloc peut libérer aléatoirement un item utile au joueur. Les power-ups sont les suivants :

*  augmente le nombre de bombes pouvant être posé simultanément sur la carte par le joueur
*  augmente la puissance d’une bombe et accroit son rayon d’explosion
*  augmente la vitesse du joueur

Les contrôles par défaut (sur un clavier français) sont les suivants :

* Joueur 1 (en bas à gauche) : ZQSD pour les mouvements et A pour l’attaque
* Joueur 2 (en bas à droite) : flèches directionnelles pour les mouvements et shift droit pour l’attaque
* Joueur 3 (en haut à gauche) : contrôle par manette, croix directionnelle pour le mouvement et X pour l’attaque
* Joueur 4 (en haut à droite) : UHJK pour les mouvements et espace pour l’attaque

1. **Proposition pour l’implémentation d’un réseau**

Au départ, notre principal but était de créer un Bomberman multijoueur pouvant utiliser le réseau local. Par manque de temps, nous avons dû abandonner cette fonctionnalité.

Notre programme aurait utilisé le module python ursina.networking directement installé avec ursina. Voilà ce qui aurait dû être utilisé :

* Il faut d’abord utilisé la classe RPCPear du module ursina.networking pour pouvoir créer une connexion.
* La page d’accueil permettrait de choisir si l’ordinateur doit être l’hôte ou s’il doit y être relié. On peut choisir entre localhost ou l’adresse IP de l’hôte.
* Il faut également créer des fonctions de sérialisation qui pourront envoyer en information plus petite les entrées du joueur ainsi que les positions en un temps très court. Ces fonctions seront associées à des fonctions de désérialisation pour réobtenir des informations exploitables par l’ordinateur.
* En fonction des entrées et différentes positions du joueur, l’animation ne se fera plus à l’aide d’une méthode de la classe Joueur, mais avec une fonction update qui déterminera la trajectoire prise, ainsi que les sprites qui seront associés.
* Il faut également créer une classe spéciale pour la position et l’animation des bombes et des items.
* Penser à envoyer la carte utilisée par le joueur 1 au joueur 2.

Bon jeu !!!