|  |
| --- |
| EMT – INF3A – Atelier |
| **Amongus-Jumper** |
| **Rapport Technique** |

Une image contenant texte

Description générée automatiquement

**AMONGUS-JUMPER**

Rapport Technique



**Amongus-Jumper**

**Rapport Technique**

Table des matières

[1 Préambule 1](#_Toc93870540)

[2 Description 1](#_Toc93870541)

[3 Convention de nommage 1](#_Toc93870542)

[3.1 Commentaire 1](#_Toc93870543)

[3.2 Fonction 1](#_Toc93870544)

[3.3 Variables 1](#_Toc93870545)

[3.3.1 Sprites 1](#_Toc93870546)

[3.3.2 Pointeur 1](#_Toc93870547)

[3.3.3 Membres 1](#_Toc93870548)

[3.3.4 Boolean 1](#_Toc93870549)

[3.3.5 Constantes 2](#_Toc93870550)

[3.3.6 Data 2](#_Toc93870551)

[3.4 Liste 2](#_Toc93870552)

[3.5 Type énuméré 2](#_Toc93870553)

[4 UML 2](#_Toc93870554)

# Préambule

Dans le cadre de l’atelier de 31-Programmation OO enseigné par Jérôme Conus, nous devons réaliser un  
rapport technique qui permet d’expliquer à des utilisateurs compétents le projet que nous avons conçu pour qu’ils puissent le reprendre. Il sera par la suite évalué

# Description

**Amongus-Jumper** est un jeu vidéo dit Platformer. Un Platformer est un sous-genre du jeux vidéo d’action qui consiste à contrôler un personnage qui doit sauter sur des plateformes dans les airs et éviter des obstacles.

# Convention de nommage

## Commentaire

Les commentaires sont écrits en français.

## Fonction

Les Fonctions sont écrites en anglais.

## Variables

### Sprites

Les éléments composant le niveau comme les lance flammes, caisses, ennemis sont écrit en français.

Ils portent leur nom abrégé suivit d’un numéro.

*Exemples :*

Sprite \* plateFormeMoyenne1 = new Sprite(GameFramework::imagesPath() + «PlatformeMyenneV2.png»);

Sprite \* plateFormeMoyenne2 = new Sprite(GameFramework::imagesPath() + «PlatformeMyenneV2.png»);

Sprite \* plateFormeMoyenne3 = new Sprite(GameFramework::imagesPath() + «PlatformeMyenneV2.png»);

### Pointeurs

Les variables étants des pointeurs possèdent un p devant leur nom.

*Exemple :*

Character\* pCharacter = new Character();

### Membres

Les variables étants membre d’une classe possède le préfixe m\_ .

*Exemple :*

Ground\* m\_Ground = new Ground();

### Boolean

Les booléennes possèdent le préfixe Is au début de leur nom.

*Exemple :*

Bool IsDeath = true;

### Constantes

Les constantes sont écrites en anglais et en majuscule.

*Exemple :*

Const int PLAYER\_SPEED = 10 ;

### Data

Les sprites ayant des interactions particulières possède des datas qui leur sont propres.

Généralement, les sprites en question possède une data qui définit leur type, parfois elle en possède une deuxième qui définit un sous-type si besoin de précision.

*Exemple :*

|  |
| --- |
| platM3->setData(1,"sol"); |
|  |

platM3->setData(2,"plateforme");

## Liste

Les listes sont écrites en anglais avec un suffixe L majuscule à la fin.

*Exemple :*

|  |
| --- |
| QList<Entity\*>m\_EntityL; |
|  |

## Type énuméré

Les énumérés sont écrites en minuscule mais leurs valeurs sont en majuscule.

*Exemple :*

//Type d'animation du joueur

enum animation{

BASE = 0,

DEPLACEMENT = 1,

SAUT = 3,

ATTAQUE = 4,

};

# Diagramme de classe

