**Witch Hunt**

**Diagramme de classe initial/final**

Nous allons de ce document expliquer de manière concise les changements opérés entre notre 1er diagramme de classes pour le projet Witch Hunt (réalisé pour le livrable 1) et le diagramme de classes final, correspondant à l’organisation des classes de notre jeu terminé.

Les diagrammes de classes se trouvent au format png dans le même dossier (doc) que ce document, sous les noms :

1er diagramme : diagramme\_classes\_witch\_hunt\_Lou\_Prevost\_Valentin\_Guitton\_livrable1.png

2ème diagramme : diagramme\_classes\_witch\_hunt\_Lou\_Prevost\_Valentin\_Guitton\_final.png

L’organisation générale de notre diagramme de classes est globalement restée inchangée mais des modifications ont cependant eu lieu.

On remarque sur le diagramme final que la classe « Partie » prend beaucoup plus de place qu’avant, ceci est lié à l’ajout de nombreuses méthodes permettant la gestion des effets, mais aussi à l’ajout d’attributs permettant par exemple de connaître l’identité du joueur actuel ou du joueur accusé.

Certaines méthodes présentes dans le 1er diagramme n’avaient en réalité pas d’utilité d’un point de vue du code et ont donc été supprimées.

La classe aléatoire, initialement présente a finalement été retirée car lors du livrable 1 nous n’avions pas eu le temps de la réaliser et il nous a été dit que le fait d’avoir un bot agressif fonctionnel était déjà suffisant et que nous devions surtout nous concentrer sur la correction d’un bug et sur l’interface graphique qui allait arriver.

Enfin, une autre modification a été la suppression des classes « EffetWitch » et « EffetHunt », qui avaient initialement pour rôle de recevoir une carte et d’activer son effet Witch ou Hunt en faisant appel aux méthodes de la classe « Partie ». Cependant, ces classes se sont avérées inutiles car nous utilisions une méthode « jouerCarte » dans la classe « Cartes Rumeurs », qui recevait le nom d’une carte ainsi qu’un booléen pour choisir entre effet hunt et effet witch, cette méthode appelait directement les méthodes de la classe « Partie » qui gèrent les effets. Cette méthode a donc remplacé les classes « EffetWitch » et « EffetHunt ».

Ces différents changements sont les seuls apportés à notre diagramme de classes.

A noter que le diagramme de classes correspond au Modèle du patron de conception MVC qui a été utilisé.