

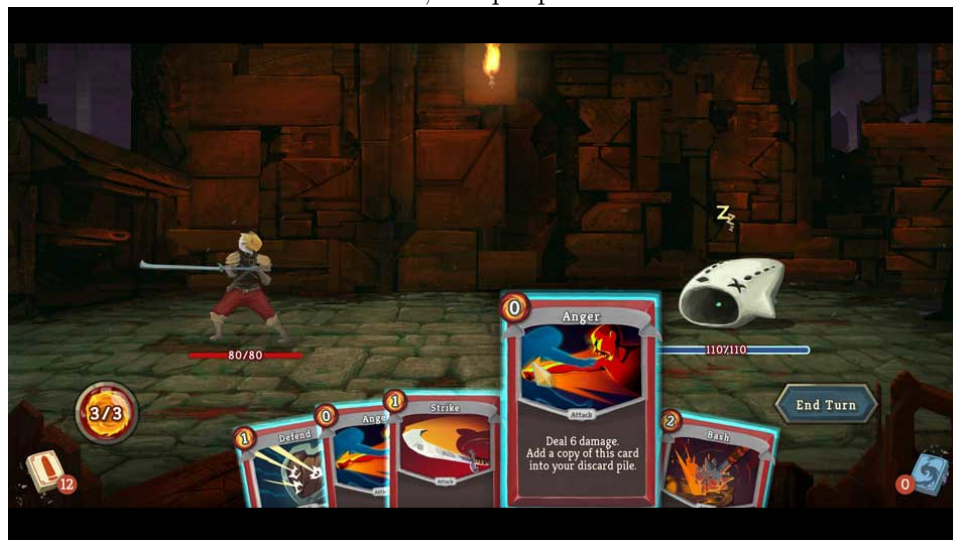
# Projet PO - Slay The Spire

18 novembre 2023

## 1 Introduction

Dans ce projet, vous devrez concevoir et implémenter une version simplifiée du jeu *Slay The Spire* : <https://www.megacrit.com/>.

Slay the Spire est un jeu dans lequel le joueur affronte des monstres en utilisant des *cartes* pour effectuer différentes actions (attaques, défense, soin...). Chaque partie du jeu se compose d'une suite de combats et de phases durant lesquelles le joueur choisit de nouvelles cartes pour la suite du jeu. C'est le principe des jeux de *deck-building*. En cas de défaite, le joueur doit recommencer une nouvelle partie en perdant toutes les cartes récoltées durant sa partie précédente. Chaque combat oppose le joueur à une nouvelle équipe de monstres choisie au hasard. Ainsi, chaque partie est différente.



## 2 Structure d'une partie

Durant chaque partie, le joueur incarne un héros qui jouera dans une suite de salles étant soit des combats, soit des points de repos. Une partie prend fin lorsque le héros perd toute sa vie ou qu'il bat le boss final.

Cette première section présente les éléments qui composent une partie.

### 2.1 Héros

Durant une partie, le héros dispose de :

- Points de vie (PV) maximum (70 par défaut).
- Point de vie actuels. Au début d'une partie, les points de vie actuels sont initialisés à la valeur maximale, et descendront au fur et à mesure des combats.
- Un *deck*. Le deck de carte de base du Héros est le même à chaque début de partie et se remplira de nouvelles cartes choisies par le joueur lors de la partie. Les cartes qui pourront rejoindre ce deck sont présentées dans l'annexe A.

### 2.2 Salles

Une partie est composée d'une suite de salles. Trois types de salles existent dans cette version simplifié :

- Les combats, durant lesquels le Héros affronte une équipe de monstres. A l'issue du combat, le joueur pourra choisir une récompense (Section 3.7).
- Les lieux de repos où le Héros se repose pour regagner des points de vie (30% de sa vie max).
- Une salle de boss final, unique à chaque partie, durant lequel le joueur affronte un monstre unique pour gagner la partie.

Le Héros passe à la salle suivante en appuyant sur entrée pour choisir la carte de récompense ou regagner des points de vie selon la salle. L'algorithme du choix des salles est pour l'instant simple : chaque partie est composée de **15** salles. On définit ici **C** pour Combat, **R** pour Repos et **B** pour Boss final. L'ordre des salles prédéfini est le suivant :

[C] -> [C] -> [R] -> [C] -> [C] -> [C] -> [R] -> [C] -> [C] -> [C] -> [R] -> [C] -> [C] -> [R] -> [B]



FIGURE 1 – Combat opposant le héros à un monstre. Plusieurs informations sont visibles, comme les points de vie, les cartes jouables ou encore l’intention du monstre.

### 3 Structure d’un combat

#### 3.1 Affichage du combat

Lors d’un combat, la scène doit afficher différentes informations comme montré sur Figure 1 :

- (1) Sur la partie gauche de l’écran, le héros avec sa barre de vie et ses statuts actuels. (Section 2.1) Ici, le héros possède 7 points de vies sur ces 72 points de vie maximum. Le statut présent en dessous de sa barre de fin correspond à *fragile* (cf. section 3.6).
- (2) Sur la partie droite de l’écran, l’équipe de monstre que le héros affronte. Chaque monstres affiche une barre de vie et ses statuts actuels. (Section 4). Ici, le monstre possède 39 points de vie restant sur ces 68 points de vie maximum. Il possède deux statuts : un qui n’est pas présenté dans ce sujet à gauche et le statuts *Vulnérable*.
- (3) De plus, chaque monstre doit afficher une image montrant son action du tour (intention) (Section 4). Ici, l’intention du monstre montre qu’il va infliger 18 points de dégâts au Héros le prochain tour.
- (4) L’énergie actuelle du héros et l’énergie qu’il récupère en début de tour. Ici, le joueur dispose actuellement de 3 points d’énergie et de 3 points d’énergie à chaque début de tour.
- (5) Les cartes dans la main du joueur. (max 10)

- (6) Le nombre de cartes dans la pioche. (Contient initialement le deck mélangé). Ici, la pioche contient 6 cartes.
- (7) Le nombre de cartes dans la défausse. (Contient les cartes jouées. Quand la pioche est vide et qu'on doit piocher, on mélange la défausse pour former une nouvelle pioche.) Ici, la pioche contient 0 cartes
- (8) Le nombre de cartes exilées. (Pile spéciale concernant certaines cartes). Ici, il y a 2 cartes exilées.

### 3.2 Librairie StdDraw

Afin d'afficher tous ces éléments, vous utiliserez la classe Affichage qui vous a été fournie avec le squelette du code. Cette classe interagit avec la librairie StdDraw. Si elle ne vous suffit pas, vous pouvez regarder directement le code de StdDraw.

### 3.3 Déroulement d'un combat

Chaque tour d'un combat se déroule comme suit :

Premièrement, le héros joue. Son tour commence en appliquant les règles suivantes :

- Le héros pioche 5 cartes. (Section 3.4)
- Son énergie actuelle devient son énergie par tour. (L'énergie restante du tour précédent est perdue)
- Tous ses points de blocage disparaissent. (Section 3.5)
- Les monstres affichent leur nouvelle intention. (Section 4)

Le joueur peut ensuite décider quelles cartes jouer. Pour cela, il doit payer son coût en énergie (l'énergie ne peut pas être inférieure à 0).

Il peut mettre fin à son tour quand il veut. A la fin de son tour, toutes les cartes de la main du héros vont dans la défausse.

Ensuite, les monstres doivent jouer. Leur tour se déroule en deux parties :

- Chaque monstre perd tous ses points de blocage
- Chaque monstre réalise son action selon son pattern (Section 4)

C'est ensuite à nouveau le tour du héros.

Le combat se termine quand le héros ou tous les monstres meurent et que toutes les actions à faire sont résolues (i.e. si une carte a plusieurs effets, on les applique tous avant de finir le combat). Si les deux camps meurent en même temps, le héros a perdu.

A la fin du combat, le deck du héros revient à son état initial (les cartes créées durant le combat disparaissent, les cartes exilés reviennent dans le deck, etc.)

### 3.4 Pioches

Le joueur dispose d'une pioche et d'une défausse. La pioche est initialisée au début du combat avec les cartes du deck et la défausse est vide. A chaque début de tour, le joueur tire 5 cartes au hasard dans la pioche.

Après avoir joué une carte, celle-ci rejoint généralement la **défausse**. Lorsque le joueur pioche et que la pioche est vide, on mélange la défausse pour en faire une nouvelle pioche. Il n'y a donc pas les cartes en main du joueur dans cette nouvelle pioche ; vous ne devez pas ré-utiliser le deck entier pour construire la pioche.

Certaines cartes doivent être mises en **exil** après avoir été jouées. Les cartes exilées ne peuvent être ré-utilisées pour les prochaines pioches.

### 3.5 Attaque et Défense

Certaines cartes ou actions de monstre dans le jeu permettent aux personnages de gagner des **points de blocage**. Quand un personnage **inflige des dégâts**, sa cible perd d'abord des points de blocage. Si elle n'a pas assez de points de blocage pour encaisser la totalité des dégâts, elle perd autant de PV que de dégâts non-bloqués. Par exemple, un personnage avec 3 points de blocage subissant une attaque de 7 points de dégâts verra ses points de blocage réduits à 0. Puis, les 4 points de dégâts restants sont enlevés à ses PV.

Quand une carte ou un effet dit qu'un personnage **perd des PV**, le blocage n'est pas compté. C'est le cas par exemple de la carte **Bloodletting**. Le nombre de PV perdus n'est pas modifié par les statuts décrits ensuite (Force, Vulnérable, etc.). Lorsqu'une carte ou un effet dit qu'un personnage **Applique des dégâts**, alors il faut considérer les points de blocage et les statuts.

### 3.6 Statut

Le héros et les monstres peuvent subir des effets durables dans le temps nommés *statuts*. Un statut est accompagné d'un certain nombre de points (sert à indiquer la durée ou l'intensité de l'effet). Si un personnage subit plusieurs fois le même statut, on additionne les points. Ainsi, si un monstre a 1 point de Fragile et est la cible d'une carte indiquant "subit Fragile 2", il aura désormais 3 points de Fragile.

Ces statuts modifient certains calculs du jeu (par exemple, les dégâts reçus) ou causent des effets en début ou fin de tour. Si plusieurs effets s'ap-

pliquent en même temps (ex : Force et Fragile), on les applique dans l'ordre de la consigne (Force avant Faiblesse).

- *Force* : Les points de Force du joueur ou du monstre sont ajoutés en valeur directe à ses points de dégâts infligés. La Force peut avoir un nombre de points négatif. Par exemple, une attaque qui inflige 5 points de dégâts lancée par un personnage avec 3 points de force devra infliger 8 points de dégâts.
- *Faiblesse* : Un personnage ayant le statut **Faiblesse** voit ses dégâts d'attaques réduits à 75%, arrondi à l'inférieur. À la fin de chaque tour, le nombre de points de Faiblesse du personnage baisse de 1. S'il atteint 0, alors l'effet prend fin. Par exemple, une attaque qui inflige 5 points de dégâts lancée par un personnage affaibli infligera 3 points de dégâts seulement.
- *Vulnérable* : Un personnage *Vulnérable* subit 50% (arrondi à l'inférieur) de dégâts supplémentaires. À la fin de chaque tour, le nombre de points de Vulnérabilité du personnage baisse de 1. S'il atteint 0, alors l'effet prend fin. Par exemple, une attaque qui inflige 7 points de dégâts à un personnage vulnérable lui infligera 10 points de dégâts.
- *Fragile* : Quand un personnage *Fragile* gagne du blocage, il gagne 75% (arrondi à l'inférieur) de la quantité normale. À la fin de chaque tour, le nombre de points de fragilité du personnage baisse de 1. S'il atteint 0, alors l'effet prend fin et les points de blocages restants sont totalement considérés pour les prochaines attaques. Par exemple, si le héros est Fragile et joue une carte lui donnant 5 points de blocage, il n'en gagnera que 3.

### 3.7 Récompenses

À la fin du combat, le joueur va pouvoir rajouter une nouvelle carte à son deck. Cette carte va donc rester avec lui pour tous les combats suivants. Pour se faire, le jeu va lui proposer entre **trois** cartes différentes. Le joueur doit en choisir une, ou aucune. Les cartes proposées en récompense par le jeu sont tirées de toutes les cartes existantes au sein du jeu. Les cartes disponibles dans le jeu sont présentées dans l'annexe A. Pour savoir quelles récompenses proposer, il faut d'abord choisir la rareté de chaque carte (Commun 60%, Non-commun 37%, Rare 3%). Ensuite, la carte est choisie avec une probabilité uniforme parmi les cartes de sa rareté.

## 4 Monstres

Chaque monstre est caractérisé par :

- des points de vie maximums.
- un pattern déterminant les prochaines actions par le monstre.

Il existe trois types de patterns :

- Aléatoire : la prochaine action du monstre est choisie aléatoirement parmi la liste des actions possibles du monstre. Certaines actions peuvent avoir plus de chances de se produire que d'autres.
- Cycle : le monstre effectue des actions selon un ordre prédéfini.
- Préparation : le monstre effectue une séquence fixée d'actions, puis il suit un pattern aléatoire ou cyclique.

Le pattern spécifique à chaque monstre ainsi que ses actions possibles sont détaillées dans l'annexe B.

## 5 Comment réaliser ce projet ?

Ce sujet vous présente l'intégralité des fonctionnalités, cartes et monstres qui composent le jeu. Il est trop difficile de réaliser du premier coup un projet aussi riche. Ainsi, nous vous proposons une démarche progressive, où vous allez d'abord réaliser une version très simple du jeu, puis la complexifier au fur et à mesure afin d'atteindre voire dépasser le jeu de base qui vous est demandé dans l'énoncé. À chaque étape, réfléchissez sur papier à votre hiérarchie de classes et **faites-la valider par le ou la chargé de TP**.

Pour cela, nous vous proposons de poursuivre **trois** grandes étapes :

- Premièrement, vous allez créer une version simplifiée des combats opposant Héros et un groupe de monstres.
- Puis, vous allez ajouter des cartes pour augmenter les possibilités de jeu lors d'un combat. De même, les monstres seront étendus pour implémenter les premiers comportements possible.
- Enfin, vous allez créer une première version du jeu avec plusieurs salles et un boss final.

### 5.1 Version simplifiée d'un combat

Pour commencer, nous allons modéliser un combat simple où notre héros affronte une équipe de monstres.

- Pour le héros, ses cartes jouables vont se limiter à *Attaque* et *Défense*. (cf. annexe A)

- Pour les monstres, un premier monstre lambda qui se contente d’attaquer à chaque tour suffit. Pour simplifier le combat, n’ajoutez qu’un seul monstre dans l’équipe de monstre dans un premier temps. Puis, vous pourrez gérer un combat avec de multiples monstres dans l’équipe.
- Si vous coincez sur l’implémentation des cartes, commencez par implémenter des méthodes que vous pouvez appeler dans le code au sein de la boucle de combat pour vérifier que les différentes mécaniques de combat fonctionnent bien. Puis proposez un objet qui puisse se reposer sur ces méthodes.

**Votre projet peut encore être en pur texte**, ou être passé en graphique. A noter que même en mode graphique il n’est pas important de ressembler exactement au jeu original : les cartes peuvent être mises en ligne et pas en arc de cercle (c’est plus facile), les points de vie peuvent être juste écrits et ne pas être une barre de vie ...

## 5.2 Implémentations des cartes

Le joueur peut affronter un ou plusieurs monstres, mais ces options de jeux sont assez limitées. Nous allons ajouter toutes les cartes de difficulté  $+$  et  $++$  comme précisé dans l’annexe A.

Faites attention particulièrement à :

- Quels sont les attributs de la carte ?
- Quel effet à ma carte ? Réfléchissez bien aux **cible(s)** de cet effet : Infliger des dégâts, mais à qui ? Soigner, mais qui ? Infliger un statut spécial, mais à qui ? etc...

Dans tous les cas, si vous bloquez trop sur une carte vous pouvez implémenter une version simplifiée de celle-ci. **Vous devrez cependant le noter dans votre README.** Enfin, faites attention au code redondant. Si vous devez ré-écrire plusieurs fois la même méthode à des points différents de votre code, vous devez sûrement revoir la conception objet de votre projet...

Maintenant, nous allons implémenter un monstre plus complexe : le **Mâchouilleur**. Celui-ci vous demandera d’ajouter une routine de sélection de son action. Ces actions vont aussi être plus diversifiées, avec certaines ajoutant des points de blocage ou effet de statut (cf. annexe B).

Faites attention à bien proposer une modélisation objet pouvant être ré-utilisée par la suite pour implémenter les autres monstres du jeu !



### 5.3 Création du jeu

Suite aux deux étapes précédentes, votre jeu propose des combats simplifiés. La dernière grande mécanique qui vous reste à mettre en place est la possibilité d’avoir plusieurs combats afin de mener une partie de bout en bout. De plus, le joueur doit pouvoir ajouter de nouvelles cartes à son deck qu’il va pouvoir garder entre ces combats.

Premièrement, vous devrez proposer une modélisation en classes des différentes salles qui compose le jeu. Une classe principale doit pouvoir utiliser les classes représentant les salles pour gérer le jeu en suivant le plan présenté dans la section 2.2. Dans cette classe principale, le joueur doit pouvoir changer de salle après chaque combat ou repos jusqu’à la dernière salle de boss. Pour l’instant, le boss se contentera d’être composé d’une équipe de 3 *Mâchouilleur*. Vous pouvez ainsi gérer les conditions de victoire (Boss battu) ou de défaite (combat perdu). Ne gérez pas la défaite par une simple fermeture puis re-ouverture du programme, ou création d’un nouveau *Jeu* : la classe principale doit pouvoir remettre le joueur au début du jeu, en ré-initialisant son deck à l’état de base.

Jusqu’ici, le deck ne s’est vu apporté aucune modification entre les combats. Désormais, le jeu doit pouvoir garder l’état du deck du joueur entre les différentes salles. De plus, celui-ci doit se remplir de nouvelles cartes après chaque combat en suivant la mécanique de récompenses (cf. section 3.7).

Finalement, vous aurez une version **jouable** du jeu dans son entièreté. Celle-ci ne représente pas toutes les mécaniques présentées dans le sujet, mais sera une base assez complète pour toutes les ajouter.

Pour la suite du projet, à vous de choisir quelle mécanique ajouter. Vous ajouterez de nouvelles cartes ou monstres en suivant les annexes A et B. Choisissez bien la difficulté d’implémentation selon le temps qu’il vous reste, il vaut mieux ajouter quelques éléments plus facile que rien du tout. De même qu’auparavant, vous pouvez **simplifier** leur fonctionnement, mais vous devez l’indiquer dans votre rapport.

## 6 Notation

Ce projet est à réaliser en binôme. Vous déposerez votre travail sur Moodle début janvier (voir Moodle pour la date exacte). Vous devez déposer un dossier compressé zip avec :

- le **code source** de votre jeu (ainsi que les images éventuelles).

- un fichier **README.txt**, donnant vos noms et une description de votre jeu : ce que vous avez fait, les fonctionnalités de votre jeu, les bonus que vous avez ajoutés. Tout ce à quoi vous voulez que votre évaluateur fasse attention, écrivez le ici ! Celui-ci est une feuille de route pour votre évaluateur dans votre projet, s'il est incomplet (Fonctionnalités non décrites mais présente dans votre code source) ou mal présenté (Pas de séparation en parties avec titres, fautes d'orthographes multiples...) cela risque de vous pénaliser.

Si deux (ou plus) binômes rendent des projets très similaires : tous les participants dans la triche auront la note de 0/20 (cette note compte pour 1/4 de votre moyenne totale de PO).

Vous aurez la note de **15/20** si vous implémentez un "jeu de base" correspondant exactement aux règles indiquées ci-dessus :

- La section 5.3 a été atteinte, et le joueur peut enchaîner différents combats et points de repos. Toutes les fonctionnalités non-évoqués dans les sections précédentes ne sont pas obligatoires.
- Le code doit présenter une conception objet propre et des commentaires Javadoc pour les classes et méthodes principales. Plus le code sera clairement commenté, plus il sera simple pour votre correcteur de connaître votre implémentation et la noter.
- L'implémentation doit impérativement utiliser des collections Java.

L'aspect **graphique** du jeu n'est pas obligatoire. Il est plus intéressant de rendre un projet avec une modélisation objet forte mais un affichage texte qu'un jeu beau visuellement mais creux dans ses fonctionnalités. Un projet en texte n'aura qu'un ou deux points en moins qu'un jeu graphique disposant des mêmes fonctionnalités.

## A Liste de cartes de base

Cette annexe décrit les cartes composant le jeu de base. Chaque carte dispose d'un coût et effet à appliquer. Selon ces effets, une complexité d'implémentation est spécifié dans une dernière colonne.

Au début du jeu, le héros dispose de 5 cartes Frappe, 4 Défense et 1 Heurt.

| <b>Cartes Communes</b>   |                |  |                             |
|--------------------------|----------------|--|-----------------------------|
| Nom                      | Coût (énergie) | Effet(s)   | Complexité d'implémentation |
| <b>Frappe</b>            | 1              | Inflige 6 dégâts à un ennemi   | ++                          |
| <b>Défense</b>           | 1              | Gagne 5 points de blocage  | +                           |
| <b>Heurt</b>             | 2              | Inflige 8 dégâts à un ennemi puis 2 de Vulnérable.   | +++                         |
| <b>Même pas mal</b>      | 1              | Gagne 8 points de blocage puis pioche 1 carte  | +                           |
| <b>Vague de fer</b>      | 1              | Gagne 5 points de blocage puis inflige 5 dégâts à un ennemi.   | ++                          |
| <b>Frappe du pommeau</b> | 1              | Inflige 9 dégâts à un ennemi puis pioche 1 carte   | ++                          |
| <b>Frappe double</b>     | 1              | Inflige 5 dégâts 2 fois à un ennemi.   | ++                          |
| <b>Enchaînement</b>      | 1              | Inflige 8 dégâts à tous les ennemis  | ++                          |
| <b>Epée boomerang</b>    | 1              | Inflige 3 dégâts à un ennemi au hasard. Cet effet s'applique 3 fois à la suite.                      | +++                         |
| <b>Manchette</b>         | 2              | Inflige 12 dégâts à un ennemi puis 2 de Faiblesse.   | +++                         |
| <b>Plaquage</b>          | 1              | Inflige x dégâts à un ennemi, avec x le nombre de points de blocage de l'entité qui utilise la carte | +++                         |

| Cartes Non-Communes   |   |   |      |
|-----------------------|---|---|------|
| <b>Saignée</b>        | 0 | Perdez 3 points de vie puis gagnez 2 points d'énergie.  | +    |
| <b>Hémokinésie</b>    | 1 | Perdez 2 points de vie, inflige 15 points de dégâts à un ennemi.  | ++   |
| <b>Uppercut</b>       | 2 | Inflige 13 points de dégâts à un ennemi, puis inflige 1 points de Faiblesse et Vulnérable.                              | +++  |
| <b>Volée de coups</b> | 1 | Inflige 2 dégâts 4 fois à un ennemi. Cette carte est mise en exil après avoir été jouée.                                | +++  |
| <b>Voir rouge</b>     | 1 | Gagne 2 points d'énergies. Cette carte est mise en exil après avoir été jouée.  | +++  |
| <b>Enflammer</b>      | 1 | Gagne 2 de Force. Cette carte est mise en exil après avoir été joué.  | +++  |
| <b>Désarmement</b>    | 1 | Fait perdre 2 de Force à un ennemi. Cette carte est mise en exil après avoir été jouée.                                 | +++  |
| <b>Onde de choc</b>   | 2 | Inflige 3 points de Faiblesse et de Vulnérables à tous les ennemis. Cette carte est mise en exil après avoir été jouée. | +++  |
| <b>Tenacité</b>       | 1 | Ajoute 2 cartes <b>Plaie</b> à sa main, gagne 15 points de blocage.   | ++++ |

| Cartes Rares            |   |  |      |
|-------------------------|---|--|------|
| <b>Gourdin</b>          | 3 | Inflige 32 points de dégâts à un ennemi.   | ++   |
| <b>Invincible</b>       | 2 | Gagne 30 points de blocage. Cette carte est mise en exil après avoir été jouée.  | +++  |
| <b>Offrande</b>         | 0 | Perds 6 points de vie, gagne 2 points d'énergie, puis pioche 3 cartes. Cette carte est mise en exil après avoir été jouée.   | +++  |
| <b>Forme démoniaque</b> | 3 | Donne au joueur 2 points d'un statut spécial <b>Forme démoniaque</b> . Au début de chaque tour le héros gagne un point de Force par point de Forme Démoniaque. Cette carte est mise en exil après avoir été jouée. | ++++ |

La carte **Plaie**, mentionné dans la carte **Tenacité** est une carte spécifique. Celle-ci ne peut-être ajouté au deck du joueur en récompense. Elles servent à "polluer" la main du joueur, et ne peuvent apparaître que durant un combat. Elles n'ont aucun effet et ne peuvent pas être joué. Cependant, elles rejoignent la pile de défausse à la fin du tour du joueur.

## B Liste de monstres

| Nom du monstre             | PV  | Mouvements   | Pattern  | Complexité d'implém. |
|----------------------------|-----|--|--|----------------------|
| <b>Petit slime piquant</b> | 12  | <b>Charge</b> : Inflige 5 dégât  | <b>Cycle</b> : Charge  | +                    |
| <b>Petit slime acide</b>   | 8   | <b>Charge</b> : Inflige 3 dégâts - <b>Lécher</b> : Inflige 1 Faible  | <b>Cycle</b> : Lécher - Charge   | ++                   |
| <b>Mâchouilleur</b>        | 40  | <b>Morsure</b> : Inflige 11 dégâts. <b>Charge</b> : Inflige 7 dégâts et gagne 5 points de blocage. <b>Gronder</b> : Gagne 6 points de blocage et 3 Force.  | <b>Préparation</b> : Charge. <b>Aléatoire</b> : Gronder 45% - Charge 30% - Morsure 25%                               | ++                   |
| <b>Cultiste</b>            | 50  | <b>Rituel</b> : Le monstre gagne 3 points d'un statut nommé <b>Rituel</b> . <b>Attaque</b> : Inflige 6 dégâts  | <b>Préparation</b> : Rituel. <b>Cycle</b> : Attaque.   | +++                  |
| <b>Slime piquant</b>       | 30  | <b>Crachat</b> : Inflige 8 dégât et ajoute une carte <b>Slime</b> dans la défausse - <b>Lécher</b> : Inflige 1 Fragile   | <b>Aléatoire</b> : Crachat 30% - Lécher 70%  | ++++                 |
| <b>Slime acide</b>         | 30  | <b>Crachat</b> : Inflige 7 dégâts et ajoute une carte <b>Slime</b> dans la défausse - <b>Lécher</b> : Inflige 1 Fragile - <b>Charge</b> : Inflige 10 dégât   | <b>Aléatoire</b> : Crachat 40% - Lécher 30% - Charge 30%   | ++++                 |
| <b>Hexaghost</b>           | 250 | <b>Chargement</b> : Aucun effet. <b>Divise</b> : Inflige H dégâts 6 fois avec $H = 1 + (\text{PV du héros}) / 12$ . <b>Enfer</b> : Inflige 6 dégâts 2 fois et ajoute 3 <b>Brûlure</b> dans la défausse. <b>Brûle</b> : Inflige 6 dégâts et ajoute 1 <b>Brûlure</b> dans la défausse. <b>Enflamme</b> : Gagne 12 points de blocage et 2 Force. <b>Charge</b> : Inflige 5 dégâts 2 fois. | <b>Préparation</b> : Chargement - Divise. <b>Cycle</b> : Brûle - Charge - Brûle - Enflamme - Charge - Brûle - Enfer. | +++++                |

Précisions :

- **Rituel** : Statut spécial du Cultiste. Le cultiste gagne un point de Force par point de Rituel à la fin de son tour.
- **Slime** : Carte servant à polluer le deck du joueur. Coûte une énergie à jouer. Son seul effet est d'être exilée après être jouée.
- **Brûlure** : Carte servant à polluer le deck du joueur. Impossible à jouer. A la fin du tour, elle inflige 2 dégâts au héros si elle est dans sa main.