

# Manuel Godot Schmup

## Principes

Le Schmup est un projet Godot. Pour le lancer, il faut ouvrir le projet et faire F5. On se déplace avec les flèches, on tire avec Espace.

Le jeu a lieu sur la scene Main. Il n'y a rien à modifier dessus.

Le joueur joue un vaisseau capable de se déplacer sur tout l'écran et de tirer. Il affronte des vagues (Wave) d'ennemis (Enemy) qu'il peut détruire à l'aide de ses tirs (Bullet). Les ennemis peuvent tirer ou non. Ils tirent à l'aide d'un ou plusieurs BulletGenerator, selon un pattern sélectionné dans une liste. Si le joueur est touché par un tir ennemi pour par un ennemi, il est détruit et perd une vie. Il réapparaît au bout de quelques secondes. Quand le joueur n'a plus de vie, la partie est finie.

Un ennemi a une phase d'entrée, où il sort d'un bord de l'écran et vient se placer, une phase principale où il reste à l'écran, et une phase de sortie.

## Séquencement des vagues

Les vagues se paramètrent dans le fichier LD.gd, accessible à la racine du projet. Double cliquer dessus pour ouvrir le script. En haut du code, une liste de dictionnaires :

```
var waves = [  
    { "id": 0, "path": "waves/Wave0.tscn", "timing": 2 },  
    { "id": 1, "path": "waves/Wave1.tscn", "timing": 10 }  
]
```

Pour ajouter une vague, il suffit d'ajouter un élément à la liste (sans oublier la virgule). Le paramètre *id* permet de savoir si la vague a déjà été déclenchée : il doit être unique. Le paramètre *path* donne l'adresse du fichier de la vague. Le paramètre *timing* donne le nombre de secondes au bout desquelles se déclenche la vague.

## Création d'une vague

Pour créer une vague, faire Scene / Nouvelle scène héritée. Choisir le fichier waves/Wave.tscn . Cela crée une nouvelle scène de vague. Enregistrer immédiatement dans le dossier waves (par exemple, appeler le nouveau fichier Wave2.tscn).

Une fois la vague créée, aller sur l'onglet 2D. On peut déposer des ennemis dans le cadre qui représente la vague. Pour ce faire, cliquer sur le dossier objects dans le système de fichier. Des fichiers apparaissent juste en dessous. Vous pouvez prendre le fichier Enemy.tscn et le déplacer sur la fenêtre 2D pour le placer. Placez autant d'ennemis que vous le souhaitez.

Chaque ennemi est paramétrable.

## Paramétrage des ennemis standards

Pour paramétrer un ennemi, cliquer sur l'ennemi sur la fenêtre 2D. Tous les paramètres de l'ennemi sont modifiables dans l'inspecteur en bas à droite. Notamment :

Lives	Nombres de points de vie de l'ennemi. Par défaut le joueur fait un dégât par tir, mais cela est paramétrable.
Time on screen	Durée de la phase principale de l'ennemi, après son entrée et avant sa sortie
Entry	Côté de l'écran où apparaît l'ennemi. En cas de None, l'ennemi apparaît à sa position.
Exit	Côté de l'écran où sort l'ennemi. En cas de None, l'ennemi n'a pas de phase de sortie et disparaît seulement quand il est détruit, où quand il sort de l'écran pendant la phase principale (grâce à sa Main speed).
Entry Order	Non valide pour l'instant. Doit servir à déclencher l'apparition des ennemis progressivement.
Transition Speed	Vitesse de déplacement l'ennemi pour entrer et sortir de l'écran
Main Speed	Vitesse de déplacement de l'ennemi (dans les deux directions) pendant sa phase principale
Bullet Cooldown	Temps entre chaque tir. Si cette valeur vaut 0, l'ennemi ne tire pas.

## Création de nouveaux ennemis

Pour créer de nouveaux types d'ennemi, faire Scene / Nouvelle scene héritée. Choisir de cloner le fichier objects/Enemy.tscn. Sauver sous un nouveau nom.

Dans la scène ainsi créée, on peut utiliser la hiérarchie (en haut à droite) pour modifier le sprite de l'ennemi. Si on veut modifier la collision shape, le plus simple est d'en ajouter une et de désactiver celle qui est héritée de Enemy.

La modification du sprite est surtout intéressant pour changer la texture de l'ennemi.

Pour modifier la Collision Shape (donc les collisions de l'ennemi) utiliser les poignets internes qui apparaissent sur la scene 2D.

On peut placer les nouveaux ennemis dans des vagues de la même manière que les ennemis standards. Si l'on souhaite modifier le comportement des nouveaux ennemis, faire hériter un nouveau script du script scripts/Enemy.gd

Si on souhaite que l'ennemi ait plusieurs générateurs de tirs, on peut en faire glisser un depuis le système de fichier, en choisissant object/BulletGenerator.tscn. On le place où on le souhaite sur l'ennemi. Chaque générateur de bullets est paramétrable.

## Paramétrage des BulletGenerators

Bullet Cooldown	Fréquence des tirs en secondes
Type	Type de tir. None : l'ennemi ne tire pas. Line : l'ennemi tire devant lui avec un certain angle

	Circle : l'ennemi tire des bullets réparties en cercle autour de lui Fan : l'ennemi tire des bullets en éventail dans un certain angle devant lui.
Line Angle	Angle de tir fixe pour le type de tir Line
Circle Bullet Number	Nombre de bullets à répartir autour de l'ennemi pour un tir de type Circle
Fan Max Angle	Angle maximum pour des tirs de type Fan, calculé depuis l'horizontale.
Fan Angle Speed	Vitesse de changement de l'angle pour un tir de type Fan

## Paramétrage du joueur

Ouvrir la scène Main.tscn à la racine du dossier. Dans la scène 2d, cliquer sur le joueur (en haut à droite de la scène). Modifier les valeurs comme désiré :

Max Speed	Vitesse maximum du joueur
Acceleration	Rapidité de l'augmentation de la vitesse du joueur
Deceleration factor	Multiplicateur de réduction de la vitesse du joueur quand on n'appuie plus sur les flèches. Doit être inférieur à 1, ou le joueur ne décélère pas.
Lives	Nombre de fois où le joueur peut être touché par un tir ou un ennemi avant de perdre une vie. 1 par défaut.
Damage	Nombre de dégâts que le joueur inflige par tir.
Cooldown	Temps minimum en seconde entre chaque tir.

Pour définir le nombre de vie du joueur, cliquer sur le nœud Main dans la hiérarchie (en haut à droite). Dans l'inspecteur s'affiche la variable Numbers of tries. La modifier comme souhaité.

## Paramétrage des parallaxes

Dans la scène Main, si on clique sur le nœud Parallax Background, on peut définir la vitesse générale de défilement des parallaxes.

Pour modifier les parallaxes, ouvrir la scene objects/Background.tscn. C'est le paramètre Scale qui définit la vitesse relative de défilement des parallaxes. On peut modifier la texture du sprite pour changer l'image.