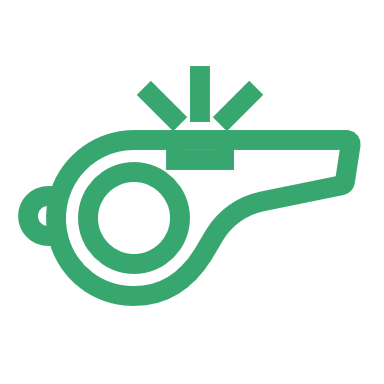
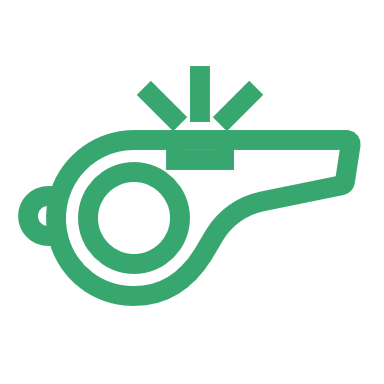
Laboratoire (partie 1)

Continuer le jeu *Forêt magique*



**Suivez bien les étapes.**

**Et lâchez votre cellulaire/Youtube/autre distraction.**

**Je ne vous dis pas tout quoi faire… c’est normal.**

# Chasseur

Créez une maquette du Chasseur. Le mot maquette (*mockup* en anglais) signifie dans le contexte que l’ennemi aura du visuel mais pas/peu de comportement.

## Animation

A pixelated image of a purple and yellow character

Description automatically generated

Créez, à la manière de *Robin*, les animations pour bouger dans les 4 directions. Utilisez les animations d’arbres *TreeAnt*. Pour tester votre système, utilisez le script *Robin.cs*, **NE PAS CRÉEZ DE NOUVEAU SCRIPT.**

## Comportement Tirer – version primitive

Robin doit tirer des flèches. Pour le moment, il va seulement tirer des flèches qui vont de gauche à droite. Nous les compléterons au prochain cours.

A pixel art of a arrow

Description automatically generated

### Création de la flèche

Utiliser l’animation de la flèche fournie sur mon site web et créez un objet animé avec une *BoxCollider2D* et un *RigidBody2D* dynamique.

### Mouvement de la flèche

Ajoutez un script *BougeVitesseConstante.cs* pour que la flèche devra bouge de gauche à droite avec une vitesse constante comme dans le jeu défenseur du sous-sol. La vitesse doit être configurable via l’éditeur.

### Tirer la flèche

A video game graphics of a cartoon character

Description automatically generated

Position du nœud Canon

Sur *Robin*, créez un nœud enfant nommé *Canon*. Créez un comportement *Tirer.cs* contenant une fonction publique Tirer() et qui expose les variables *Transform Canon* et *Transform ExemplaireProjectile*. Modifiez le script *ControleJoueur.cs* et lorsqu’on appuie sur une touche au choix, appeler la fonction Tirer() pour lancer la flèche à partir de la position de *Transform Canon*.

## Élimination du chasseur

### Explosion magique

Utilisez la spritesheet *ExplosionMagique.png* se trouvant sur le site web pour créer une animation d’explosion. Ajoutez un script *EliminationFinAnimation.cs* et faite qu’elle disparaisse à la fin de son animation.

### Élimination

Finalement, créez un comportement *Vivant.cs*. Les variables *int NombreDePointDeVie* comprise entre [1,20], *Transform ExemplaireExplosionLorsElimination* et *bool DoitDevenirRouge* doivent être exposées via l’éditeur. L’entité qui a ce script assigné doit devenir un peu plus rouge à chaque fois qu’il se fait toucher selon la valeur de *DoitDevenirRouge*, à la manière des *Viscos*.

## Obstacle

A square wooden box on a blue wall

Description automatically generated

Créez un obstacle simple qui pourra être poussé par votre joueur. Pour ce faire, créez un object avec le sprite *WoodCrate*, un *BoxCollider2D* ainsi qu’un *RigidBody2D* dynamique. Ajustez la propriété *Drag* du *RigidBody2D* pour ralentir l’obstacle lorsqu’il est poussé par *Robin*. La boîte ne doit pas tourner lorsqu’on la pousse.