Développement de jeux vidéo

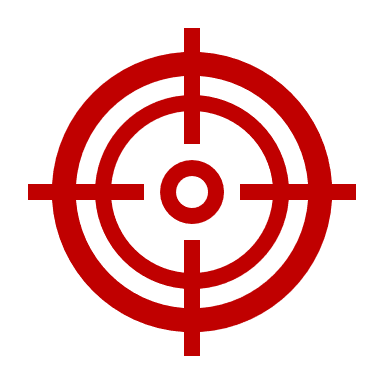
TP01 – Vilain poulet et la grange maudite



Créez un jeu complet par vous-même!

Contents

[ Présentation 2](#_Toc160282832)

[Modalités 3](#_Toc160282833)

[Livrable 3](#_Toc160282834)

[ Résumé des éléments à réaliser 4](#_Toc160282835)

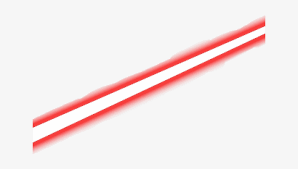
[Déroulement du jeu 5](#_Toc160282836)

[Environnement 5](#_Toc160282837)

[Description des acteurs 5](#_Toc160282838)

[ **Vilain poulet** 5](#_Toc160282839)

[**Laser** 6](#_Toc160282840)



[ **Lama** 6](#_Toc160282841)

[**Cochon** 7](#_Toc160282842)



[Diagram

Description automatically generated**Caisse compagnon** 7](#_Toc160282843)

[A picture containing histogram

Description automatically generated **Oeuf en or** 7](#_Toc160282844)

[A picture containing light

Description automatically generated **Douzaine d’oeufs en or** 7](#_Toc160282845)

[A cloud in the sky

Description automatically generated **Nuages** 7](#_Toc160282846)

[Ressources supplémentaires 8](#_Toc160282847)

# Présentation

*Vilain poulet et la grange maudite* est l’histoire d’un poulet qui en a assez de vivre dans sa grange extérieure et veut traverser la route. Il débute sa quête mais au même moment la ferme devient maudite…

# Target with solid fillModalités

Des ressources vous seront fournies et vous aurez à faire l’assemblage ainsi que de la conception de niveaux.

* Vous pouvez ajouter/modifier les image.
* Date de remise **mardi 2 avril 2024 sur Léa avant la fin de la journée.**
  + Chaque journée de retard = 10% de pénalité.
* Vaut 30% de la session.
* Équipe de 2 ou 3. Les attentes pour une équipe de 3 sont ajustées en conséquence.
* Si deux copies sont partiellement ou complètement identiques, c’est du plagiat et les 2 équipes ont la note 0.
* Vous êtes encouragé à ajouter du contenu, mais assurez-vous de faire ce qui est demandé.

# Present outlineLivrable

Nom du projet : ***Poulet\_NOM\_PRENOM\_NOM\_PRENOM***.

Remise de…

* Projet Unity au complet sans fichier intermédiaire (penser gitignore)
* Version publiée sous Windows.

Le tout remis via Léa dans un gros dossier compressé.

Si votre jeu est trop difficile, injouable ou trop buggé, il est possible que vous ayez la note 0%.

**\* Ne pas se conformer aux conventions vous pénalisera de 10%. \***

# Clipboard Checked with solid fill Résumé des éléments à réaliser

* Scènes
  + Niveau **Cours intérieur**
  + Niveau **Cours**
  + Niveau **Route**
* Ennemis
  + Lama
  + Laser
  + Cochon
* Joueur
  + Vilain poulet
* Tuilages
* Séquences
  + Fondu au noir
  + Fondu au jeu
  + Célébration
  + Mort (du Vilain poulet)
* Items spéciaux
  + Œuf en or
  + La douzaine d’œufs en or
  + Caisse compagnon
* Atmosphère
  + Nuages

# Solar system with solid fillDéroulement du jeu

Après un Fondu au jeu, le Vilain poulet débute sa quête dans le fond de sa **Cours intérieur** **sur son foin**. Après avoir relevé des défis, à la sortie, il pourra récupérer un Œuf en or qui déclenchera la séquence de sortie. Elle consiste en un fondu au noir, un changement de niveau, puis un fondu pour se rendre au prochain niveau.

Il apparaît à la sortie du niveau **Cours**. Il devra récupèrer à nouveau un œuf en or pour se rendre au niveau final avec le même effet de transition: la **Route**.

Le Vilain Poulet doit traverser la rout. L’autre côté de la route, il récupérera une La douzaine d’œufs en or qui déclenchera la Célébration.

Grosseur des zones accessibles par le Vilain Poulet :

* **Cours intérieur : 15x15 tuiles**
* **Cours : 20x20 tuiles**
* **Route**: **30x15 tuiles ou 15x30 tuiles**

1 tuile= 1 unité de Unity. Une tuile doit être approximativement grosse comme le Vilain Poulet.

On doit pouvoir aller sur les bords des niveaux sans voir d’artefacts visuels.

## Difficulté

Si vous n’êtes pas en mesure de terminer chaque niveau en 3 vies ou moins, vous risquez la note 0.

## Défis

Chaque niveau est organisé en *défis*. Un *défi* est une section délimitée dans le niveau dans laquelle le joueur doit surmonter une épreuve.

* **Cours intérieur : au moins 2 défis**
* **Cours : au moins 2 défis**
* **Route**: **au moins 3 défis**

### Construction de défis

Pour créer des défis, agencez ensemble au moins deux différents acteurs du jeu :

* Caisse compagnon
* Lama
* Cochon
* Laser fixe
* Laser rotatif

Ex : Dans une section, deux lamas rodent et j’ai une caisse compagnon pour me cacher derrière.

Ex : Deux lasers rotatifs tournent dans le sens opposé et j’ai une caisse compagnon pour m’aider à traverser la section.

Laissez aller votre imagination!

#### Étape 1 : conception

Pour chaque niveau, faites un plan AVANT de construire le niveau des défis. Par exemple, pour la **Cours intérieur :**

* Défi 1 : Un laser fix bloque le passage. Le joueur doit utiliser une caisse compagnon dans les environs.
* Défi 2 : Sur une grande surface, un laser rotatif à 2 pales placé au centre tourne rapidement et un Lama rode. Le joueur doit traverser de gauche à droite.

#### Étape 2 : maquette

Dans votre scène Unity, créez le sol et ensuite, sur la couche de collision, créer 2 sections CLAIREMENT DÉLIMITÉS et connectez-les avec un petit couloir. Placez vos ennemis et testez!

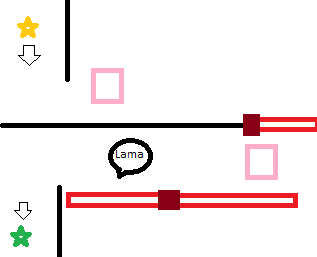


Figure 1: Maquette du défi 1

##### Étape 3 : décoration

Quand vous êtes satisfaits, placez des doodads, arbres et autre éléments décoratifs.

## Environnement

Le dossier Tuilage contient toutes les tuiles dont vous avez besoin. Il doit y avoir au moins…

* **Sol** : les acteurs peuvent circuler par-dessus
* **Doodad** : les acteurs peuvent circuler par-dessus
* **Trou** : lacs et trous qui bloquent les acteurs sauf le laser.
* **Terrestre** : bloquent tous les acteurs.
* **Ciel** : les acteurs peuvent circuler en dessous.

# Drama with solid fillDescription des acteurs

## Vilain poulet

Joueur contrôlable avec les flèches.

### Fonctionnement

Il doit pouvoir *marcher* dans toutes les directions mais utilisez seulement les animations de 4 côtés. Lorsque le poulet est immobile, il doit jouer son animation de *picossement au sol* de manière aléatoire. Il doit également faire face dans la bonne direction une fois la touche relâchée. Par exemple, s’il marchait vers le haut, il doit picosser vers le haut lors de l’arrêt.

### Séquence de Mort

Lorsque le vilain poulet à un ennemi, il est éliminé. Il doit se produire au minimum les éléments suivants :

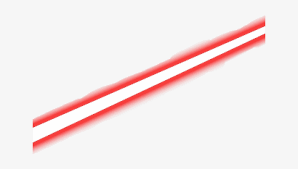
1. Le joueur perd le contrôle et disparaît.
2. Apparition d’une des animations d’explosion.
3. À la fin de l’animation, un poulet cuit apparaît. 
4. Pause d’une demie seconde.
5. Fondue au noir.
6. Recommencement au niveau **actuel**.

### Séquence de Célébration

Lorsque la douzaine d’œufs en or est atteinte, la séquence de célébration démarre en boucle.

1. Le joueur perd le contrôle.
2. Le poulet doit marcher dans une direction au choix sur une courte distance.
3. Il doit ensuite picosser le sol.
4. Il marche dans la direction opposée et picosse encore.

## Laser



Le laser est émis par . Il doit pouvoir être émis dans toutes les directions si est tourné. Il est arrêté uniquement par les éléments suivants :

* Murs de l’environnement
* Caisse compagnon

Autrement, il n’a pas de limites de longueur.

Pour l’affichage du laser, utilisé le *package* nommé [Volumetric Lines](https://assetstore.unity.com/packages/tools/particles-effects/volumetric-lines-29160) sur le marché d’Unity. Copier *VolumetricLines/BuiltInRenderPiepeline/Prefabs/SingleLine-LightSaber.prefab* dans votre dossier de projet. Ajoutez un laser dans la scène et prenez le temps de l’ajuster : rapetissez-le et ajustez la propriété *LightSaberFactor*.

Ajoutez ensuite un script sur le *prefab*. Voici un pseudocode comme exemple définir le départ et l’arrivé du laser.

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

public class ControleLaser : MonoBehaviour

{

private VolumetricLines.VolumetricLineBehavior laser;

void Start()

{

laser = GetComponent<VolumetricLines.VolumetricLineBehavior>();

//Les positions sont données en coordonnées monde.

laser.StartPos = new Vector3(0.0f, 0.0f, 0.0f);

laser.EndPos = new Vector3(1.0f, 0.0f, 0.0f);

}

}

## Lama

### Errance

Le Lama doit être immobile pour un aléatoire au choix pendant un certain temps (suggéré entre 1 et 2 secondes). Il doit ensuite bouger dans une direction aléatoire pendant un certain temps (suggéré entre 1 et 2 secondes). Il doit effectuer ce cycle sans arrêt.

### Chasse

Lorsque le poulet approche et est visible (i.e. pas caché derrière un mur), le Lama chasse le Vilain Poulet. Le Vilain Poulet doit pouvoir courir assez vite pour se sauver. Lorsque le Vilain Poulet s’éloigne suffisamment ou se cache, le Lama revient au comportement d’errance. Les lamas sont invincibles aux autres ennemis.

Lorsque le Lama se met en chasse, un effet spécial (fourni dans les ressources) de poussière démarre à ses pieds.

## Cochon



Les cochons sont toujours en marche. Il est seulement nécessaire d’intégrer ces directions.

**OU**  

Ils doivent être au moins présent sur la route en traversant dans 2 directions. Évidement, l’espacement entre les cochons doit être suffisant pour que le Vilain Poulet passe sans être trop facile.

### Émetteur et destructeur

Le cochon doit être émis et disparaître SOUS un A picture containing indoor, wood, grill, cooking

Description automatically generated . De plus, il doit entrer SOUS un A picture containing indoor, wood, grill, cooking

Description automatically generated et sera éliminé une fois complètement à l’intérieur. On ne doit pas voir le cochon se détruire.

## Diagram Description automatically generatedCaisse compagnon

Merci Guillaume pour la suggestion… C’est un objet de physique qui peut être poussé par le joueur et optionnellement les lamas et cochons. Il sert surtout à bloquer les lasers ou à se cacher des lamas.

## A picture containing histogram Description automatically generated Oeuf en or

C’est simplement un sprite sans animation qui, lorsqu’atteint par le joueur, démarre le changement de niveau.

L’œuf doit être accompagnée d’une animation de scintillement (fournie dans les ressources) qui joue à tout moment pour le mettre en évidence.

## A picture containing light Description automatically generated Douzaine d’oeufs en or

C’est simplement un sprite sans animation qui, lorsqu’atteint par le joueur, démarre la Célébration . La douzaine doit être accompagnée d’une animation de scintillement (fournie dans les ressources) qui joue à tout moment pour le mettre en évidence.

## A cloud in the sky Description automatically generated Nuages

Des nuages bougeront lentement dans le ciel pendant toute la partie pour les 3 niveaux. Vous devriez avoir un émetteur et une zone de destruction similairement aux cochons. Les nuages doivent…

* Utiliser chacun des 3 modèles de nuages au hasard.
* Être de grosseur variable
* Être de vitesse variable
* Ne pas nuire à la partie (i.e. s’il est trop gros et qu’on ne voit plus rien)

# Ressources supplémentaires

Normalement, vous avez ce qu’il faut pour compléter le jeu sans accéder à ces liens, mais vous pouvez modifier les images si vous le voulez.

Animaux

<https://opengameart.org/content/lpc-style-farm-animals>

Poulet

<https://opengameart.org/content/chicken-sprites>

Mort

<https://opengameart.org/content/chicken-and-pork-icon-pack>

<https://opengameart.org/content/more-lpc-meals>

Environnement et magie

<https://opengameart.org/content/lpc-farming-tilesets-magic-animations-and-ui-elements>