

# RAPPORT DE STAGE :

## Développer Unity : *Réalisation de jeux 3D sur Unity*

---

*Stage de 5<sup>e</sup> année*

---

**Nom** : Ghislain LEVREAU

**Formation** : 5<sup>e</sup> année Ingénieur Polytech Marseille

**Année** : 2023-2024

**Entreprise** : Jungle VR

**Adresse** : 29 rue Etienne Dolet 94140 ALFORTVILLE FRANCE

**Stage du** : 7 février au 28 juillet 2024

---

*Partie Crazy Cook Fusion*

---

# Crazy Cook Fusion

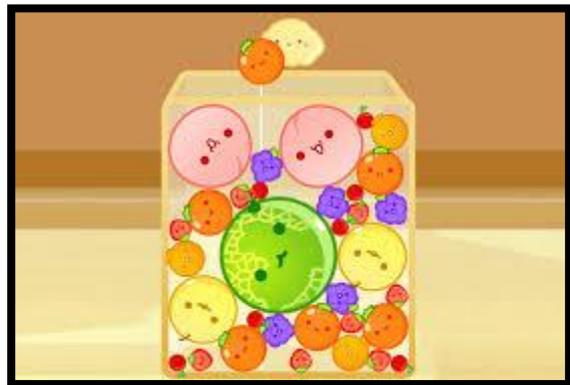
## I. Genèse de Crazy Cook Fusion

### a) Origine de l'idée

L'idée de créer *Crazy Cook Fusion* est née de la volonté de tirer parti des assets de haute qualité développés pour *Crazy Cook VR*. Ce dernier jeu, qui se distingue par son univers visuel unique et son style cartoon, regorge de modèles 3D animés, de menus interactifs bien conçus, et de sons distinctifs. L'opportunité de capitaliser sur ces ressources existantes plutôt que de repartir de zéro s'est rapidement imposée. En réutilisant ces assets, il était possible de créer un nouveau jeu qui non seulement bénéficierait d'une esthétique cohérente avec *Crazy Cook*, mais qui permettrait également de réduire considérablement le temps et les coûts de développement. Cette approche stratégique a permis de conserver l'identité visuelle et sonore distincte de *Crazy Cook* tout en offrant une nouvelle expérience de jeu.

### b) Adaptation du Suika Game

*Crazy Cook Fusion* est une adaptation innovante du célèbre *Suika Game*, un jeu qui a connu un immense succès il y a quelques mois grâce à son concept simple mais captivant. Dans ce type de jeu, les joueurs doivent fusionner des objets identiques pour créer de nouvelles entités plus grandes et plus complexes, avec pour objectif final de continuer à fusionner jusqu'à remplir l'espace de jeu ou atteindre une fusion ultime. *Crazy Cook Fusion* reprend ce concept et l'amène en 3D, tout en intégrant l'univers visuellement riche et excentrique de *Crazy Cook*.



*Suika Game*

Le jeu propose une expérience où le joueur doit envoyer des ingrédients dans une marmite à l'aide d'un canon. Les ingrédients similaires fusionnent pour gagner des points et libérer de l'espace, mais le défi réside dans la gestion de cet espace limité : si les ingrédients débordent de la marmite, la partie est perdue. En plus de la mécanique de fusion, *Crazy Cook Fusion* introduit un système de recettes, ajoutant une couche de stratégie et d'objectif supplémentaire au jeu. En combinant ces éléments, *Crazy Cook Fusion* offre une expérience unique qui allie le gameplay addictif du *Suika Game* avec l'univers thématique et ludique de *Crazy Cook*.

### c) Transfert et adaptation des assets

Pour lancer le développement de *Crazy Cook Fusion*, la première étape a été de copier le projet existant de *Crazy Cook VR* et de récupérer les assets nécessaires pour le nouveau jeu. Tous les modèles 3D, animations, effets sonores, et éléments d'interface utilisateur ont été importés dans le nouveau projet. Une fois transférés, ces assets ont été soigneusement triés et adaptés pour répondre aux besoins spécifiques du gameplay du *Suika Game*. Certains éléments ont été réutilisés tels quels, tandis que d'autres ont nécessité des ajustements pour s'adapter au nouveau contexte de jeu. Cette stratégie a permis de préserver l'identité visuelle de *Crazy Cook* tout en offrant une expérience de jeu renouvelée.

## II. Fusion des ingrédients

### a) Préparation des ingrédients

Pour *Crazy Cook Fusion*, j'ai sélectionné 10 ingrédients, inspirés des *Suika Games*, en définissant un ordre de fusion précis. Dans ce système, deux ingrédients identiques se transforment automatiquement en l'ingrédient suivant, créant ainsi une progression logique et visuelle.



*Liste dans l'ordre des ingrédients choisis*

Avant d'intégrer ces ingrédients, plusieurs étapes de préparation ont été nécessaires :

1. **Nettoyage des Ingrédients** : J'ai commencé par nettoyer les modèles 3D des ingrédients pour m'assurer qu'ils étaient optimisés pour leur nouvelle utilisation dans le jeu. Cela a impliqué la suppression des éléments superflus, l'ajustement des textures, et la préparation des modèles pour une intégration efficace dans le moteur de jeu.
2. **Ajout de Colliders et de Rigidbody** : Afin que chaque ingrédient puisse interagir physiquement avec son environnement, j'ai ajouté des *Colliders* et des *Rigidbody* à chaque modèle. Les *Colliders* ont été ajustés pour correspondre précisément à la forme des ingrédients, tandis que les *Rigidbody* ont été configurés pour permettre des interactions réalistes avec les autres objets du jeu.
3. **Correction des Effets de Particules** : Les effets de particules, essentiels pour illustrer la fusion des ingrédients, ont été configurés pour correspondre précisément à chaque transition entre les ingrédients.

La prochaine étape a consisté à uniformiser la taille des ingrédients, en tenant compte de leur volume. Pour cela, j'ai calculé le volume de chaque ingrédient en utilisant les *CapsuleColliders* associés. Ensuite, j'ai appliqué un facteur d'échelle de 1,5 pour chaque niveau de fusion, de manière que le volume du nouvel ingrédient fusionné soit environ 1,5 fois plus grand que celui des ingrédients précédents.

```
# Les volumes cibles sont calculés pour garantir que V1 * 1.5 = V2

# Exemple de sortie
Onion: Scale factor = 1.000
Tomato: Scale factor = 1.082
Potato: Scale factor = 1.296
Courgette: Scale factor = 1.953
Basilic: Scale factor = 1.434
EggPlant: Scale factor = 3.325
Crocnananas: Scale factor = 2.067
BlobFish: Scale factor = 2.542
```

*Facteurs multipliés*



*Ingrédients avec la bonne taille*

Pour faciliter la gestion des informations liées aux ingrédients et à leur fusion, un *Scriptable Object* a été créé pour chaque ingrédient. Ce *Scriptable Object* permet de centraliser toutes les données essentielles concernant les ingrédients, rendant ainsi la gestion et la configuration de chaque ingrédient beaucoup plus simples et structurées.

Chaque *Scriptable Object* inclut des informations telles que le nom de l'ingrédient, le prefab associé, les points de spawn, les points de fusion, et l'ingrédient suivant dans la chaîne de fusion. De plus, il contient des références aux effets visuels de fusion, à la probabilité de spawn, au prefab de prévisualisation, et au sprite de l'ingrédient.

```
public class FusionIngredient : ScriptableObject
{
    public string ingredientName;
    public GameObject prefab;
    public int spawnPoints;
    public int fusionPoints;
    public FusionIngredient nextIngredient;

    public GameObject fusionVFX;

    public float SpawnProba;
    public GameObject PreviewPrefab;

    public Sprite ingredientSprite;
}
```

*Scriptable Object des ingrédients*

### b) Système de fusion

Le système de fusion dans *Crazy Cook Fusion* est au cœur du gameplay, où deux ingrédients identiques se combinent pour former un nouvel ingrédient de niveau supérieur. Ce processus de fusion est géré par le script *FusionManager* qui contrôle l'ensemble des interactions entre les ingrédients.

Lorsqu'un ingrédient entre en collision avec un autre, la fonction *OnCollisionEnter* est appelée. Cette fonction vérifie si les deux objets en collision sont des ingrédients identiques et si aucun d'eux n'est déjà en cours de fusion (vérification avec les variables *isFusing* et *hasFused*). Si toutes les conditions sont réunies, la fonction *HandleFusion* est déclenchée pour gérer la fusion.

La fusion est gérée par la coroutine HandleFusion. Voici les étapes principales du processus :

- **Début de la Fusion** : Un délai est introduit avec `delayBeforeMerge` pour éviter une fusion instantanée après la collision. Pendant ce temps, les deux ingrédients sont marqués comme étant en cours de fusion (`isFusing = true`).
- **Création de l'Ingrédient Fusionné** : Si les conditions de fusion sont toujours valides après le délai, un nouvel ingrédient (de niveau supérieur) est instancié à la position médiane entre les deux ingrédients d'origine.
- **Gestion des Scores** : Le système ajoute des points au score du joueur pour chaque fusion réussie, en utilisant le `ScoreManager`.
- **Effets Visuels de Fusion** : Pour renforcer l'impact visuel de la fusion, un effet de particules est joué au moment où le nouvel ingrédient apparaît. Cet effet est configuré à partir des objets du jeu et instancié à la position de fusion.
- **Nettoyage** : Les deux ingrédients d'origine sont ensuite détruits pour laisser place à l'ingrédient fusionné. La fusion est marquée comme complète (`hasFused = true`) et le système attend un délai (`delayAfterMerge`) avant de permettre une nouvelle fusion avec le nouvel ingrédient.

### c) Problèmes de fusion

Les premières versions du système de fusion dans *Crazy Cook Fusion* ont rencontré plusieurs problèmes majeurs qui rendaient le jeu difficilement jouable. Parmi ces défis, certains ingrédients traversaient la marmite après une fusion, perturbant gravement le gameplay. De plus, l'apparition soudaine du nouvel ingrédient provoquait souvent des explosions imprévues, éjectant les autres ingrédients hors de la marmite. Un autre problème concernait l'ingrédient "Courgette", qui, en raison de sa forme, ne s'intégrait pas harmonieusement avec les autres ingrédients, nécessitant finalement sa suppression du jeu.

Pour résoudre ces problèmes, plusieurs solutions ont été mises en place :

- **Apparition Progressive du Nouvel Ingrédient** : Plutôt que de faire apparaître instantanément le nouvel ingrédient à sa taille finale, celui-ci est désormais généré en taille réduite et grossit progressivement grâce à `GrowIngredient`. Cette approche permet de réduire l'impact physique de l'apparition sur les autres objets environnants, évitant ainsi les explosions qui pouvaient éjecter les ingrédients en dehors de la marmite.
- **Positionnement en Hauteur** : Le nouvel ingrédient est également généré légèrement en hauteur par rapport au sol de la marmite. Cette méthode garantit qu'il ne passe pas à travers le sol ou ne cause pas d'autres problèmes de collision lors de son apparition.

### III. Mécanique du canon

#### a) Contrôles du canon

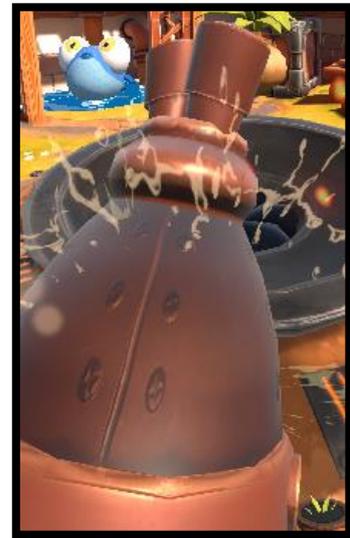
Le système de contrôle du canon dans *Crazy Cook Fusion* a été conçu pour être aussi simple et intuitif que possible. Le joueur n'a qu'à toucher l'écran pour commencer le processus de tir, contrôlé par l'*inputManager*. Une fois que le joueur appuie sur l'écran, le canon commence à se charger, montré par une animation et un son. Une fois que la charge atteint un niveau suffisant, le joueur peut relâcher l'écran pour tirer l'ingrédient. Cette approche minimaliste rend les contrôles accessibles à tous les joueurs.



*Canon IDLE*



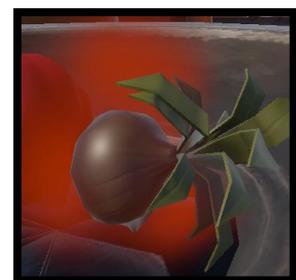
*Canon CHARGED*



*Canon SHOOT*

Pendant la charge, le joueur peut également faire glisser son doigt pour ajuster la visée du canon. Cela permet de cibler précisément les zones où les ingrédients doivent être envoyés dans la marmite.

Une fonctionnalité supplémentaire a été ajoutée pour introduire un élément stratégique au jeu. Si le joueur maintient le canon en charge trop longtemps, le canon commence à surchauffer. Cette surchauffe se manifeste par des tremblements du canon. Si le joueur ne relâche pas l'écran à temps, le canon tire automatiquement un ingrédient brûlé, qui ne peut pas être fusionné.



*Ingrédient brûlé*

Dans les versions précédentes, la rotation du canon était contrôlée manuellement par les inputs du joueur. Cependant, cette méthode a été remplacée pour être encore plus intuitive. Désormais, le canon tourne automatiquement autour de la marmite, permettant au joueur de se concentrer sur le timing et la visée plutôt que sur la rotation. Cette rotation automatique permet également d'atteindre des ingrédients cachés ou situés à des endroits difficiles d'accès.

## *b) Fonctionnement*

Le tir du canon dans *Crazy Cook Fusion* est une mécanique centrale du jeu fonctionné par *CanonController*. Lorsque le joueur relâche l'écran après avoir appuyé pour charger le canon, l'ingrédient sélectionné est projeté avec une force. Cette force est modulée par les paramètres du canon, notamment la force de tir (shotForce) et la masse de l'ingrédient (ingredientMass). Le canon utilise un système de Rigidbody pour simuler la physique du lancement, et des effets visuels comme les particules sont déclenchés lors du tir pour ajouter de l'impact visuel.

Le système de sélection des ingrédients repose sur une file d'attente qui est initialisée au début du jeu. À chaque tir, un nouvel ingrédient est tiré au hasard en fonction des probabilités définies pour chaque ingrédient (SpawnProba). Cet ingrédient est alors placé dans la file d'attente visuelle, et la file d'attente est mise à jour. Le joueur voit toujours les ingrédients qui seront les prochains à être lancés, ce qui lui permet de planifier ses actions stratégiquement. Lors du lancement, l'ingrédient sélectionné est instancié au niveau du canon et projeté dans la marmite. Si le joueur garde le doigt sur l'écran trop longtemps, une texture de brûlé est appliquée à l'ingrédient spécial.



*Ingrédient prêt à être tiré*



*Prochains ingrédients*

Les prochains ingrédients à venir sont affichés à l'écran pour que le joueur puisse anticiper et planifier ses tirs. Chaque fois qu'un ingrédient est lancé, la file d'attente des ingrédients est déplacée, et un nouvel ingrédient est ajouté à la fin de la file. Cette file d'attente visuelle permet au joueur de toujours savoir quels ingrédients sont disponibles pour les prochains tirs. Les animations sont utilisées pour déplacer les ingrédients dans la file d'attente, créant un flux visuel dynamique qui aide à maintenir l'engagement du joueur. Le système garantit également que les ingrédients sont toujours orientés vers la caméra pour une meilleure visibilité.

### c) Calcul de la Trajectoire

Pour aider le joueur à mieux planifier ses tirs, *Crazy Cook Fusion* intègre un système de calcul et d'affichage de la trajectoire des ingrédients projetés dans le script [TrajectoryLine](#). Ce système utilise un `LineRenderer` pour visualiser la trajectoire prévue d'un ingrédient avant même qu'il ne soit lancé, offrant ainsi un retour visuel précieux.

Le calcul de la trajectoire repose sur deux méthodes principales : `GetPosition` et `GetTangent`, qui prennent en compte à la fois la gravité et la résistance de l'air (drag).



*Trajectoire*

**1. Méthode `GetPosition` :** Cette méthode calcule la position future d'un ingrédient à un instant donné. Elle tient compte de la gravité qui attire constamment l'ingrédient vers le bas, ainsi que du drag, qui ralentit l'ingrédient en fonction de sa vitesse.

- **Sans Drag :** Si le drag est négligeable, la trajectoire est une parabole classique, calculée par la formule :

$$\text{position} = \text{position initiale} + \text{vitesse initiale} \times t + \frac{1}{2} \times \text{accélération} \times t^2$$

Ici, l'accélération est principalement due à la gravité.

- **Avec Drag :** Lorsque le drag est pris en compte, la position est ajustée pour refléter la résistance de l'air. Le drag réduit la vitesse au fil du temps, ce qui affecte la trajectoire en la courbant plus rapidement vers le sol. La position est alors calculée en intégrant ces effets, en utilisant une combinaison de logarithmes et de puissances pour moduler l'impact du drag sur la position.

**2. Méthode `GetTangent` :** Cette méthode calcule la vitesse d'un ingrédient à un moment donné, en tenant compte du drag.

- **Sans Drag :** La vitesse tangentielle est simplement la vitesse initiale, modifiée par l'accélération due à la gravité sur le temps écoulé.
- **Avec Drag :** Avec le drag, la vitesse diminue de manière exponentielle. La formule ajustée prend en compte la réduction progressive de la vitesse en fonction du drag, ainsi que l'effet continu de la gravité.

Les méthodes `GetPosition` et `GetTangent` sont utilisées dans la fonction `CalculateTrajectoryLine`, qui génère une série de points représentant la trajectoire prévue de l'ingrédient. Le `LineRenderer` utilise ces points pour dessiner une ligne visible à l'écran, permettant au joueur de voir où l'ingrédient devrait atterrir une fois lancé. De plus, quand le canon commence à surchauffer, il devient rouge.



*Surchauffe*

## IV. Recettes et modes de jeu

### a) Systeme de recettes

Le système de recettes, du script *RecipeManager*, dans *Crazy Cook Fusion* ajoute une couche de stratégie et de progression au jeu, en permettant aux joueurs de débloquer et de compléter des recettes spécifiques.

Le cœur du système de recettes repose sur la présence d'ingrédients spécifiques dans la marmite. Chaque recette dans le jeu est composée d'une série d'ingrédients que le joueur doit fusionner correctement pour compléter la recette. Par exemple, pour réaliser une soupe, le joueur devra combiner des oignons, des tomates, et des pommes de terre dans la marmite en respectant les quantités requises.

Le système vérifie constamment les ingrédients présents dans la marmite, en comparant ceux-ci avec les ingrédients requis par la recette en cours. Si tous les ingrédients nécessaires sont présents et dans les bonnes proportions, la recette est validée, et le joueur est récompensé avec des points, ainsi qu'un feedback visuel et sonore.

Pour gérer efficacement les recettes, un Scriptable Object est utilisé pour définir chaque recette. Ce Scriptable Object contient toutes les informations nécessaires, y compris les ingrédients requis, les points accordés, et les effets visuels associés à la recette.

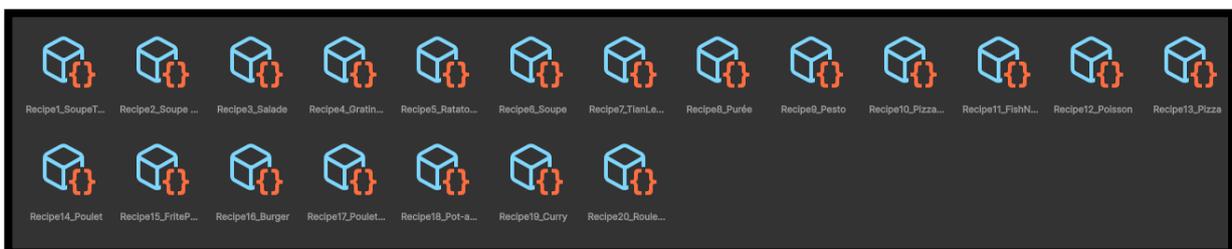


*Image de recette*

```
public class Recipe : ScriptableObject
{
    public string recipeName;
    public FusionIngredient[] requiredIngredients;
    public Sprite presentationSprite; // Sprite de présentation de la recette
    public int pointsEarned; // Nombre de points gagnés en réalisant cette recette
    public string lore; //texte de lore
}
```

*Scriptable Object des recettes*

Ce Scriptable Object permet de centraliser la gestion des recettes, facilitant ainsi leur intégration et leur modification. 20 recettes ont été créés avec les 9 ingrédients différents.



*Scriptable Object des recettes*

## b) Systeme de score

Le système de score dans *Crazy Cook Fusion* est un élément essentiel qui permet au joueur de suivre sa progression au cours de la partie, de comparer ses performances avec ses précédentes tentatives, et de se fixer des objectifs de score plus élevés.

Le score du joueur est mis à jour en temps réel pendant la partie et est affiché de manière visible à l'écran. Le score est accumulé à travers différentes actions, telles que le lancement d'ingrédients, leur fusion, et la réalisation de recettes. Le système de score permet ainsi de quantifier les performances du joueur et de lui offrir un retour immédiat sur ses actions. Pour encourager la progression et la rejouabilité, le meilleur score du joueur est sauvegardé localement à l'aide de `PlayerPrefs`. Cela permet au joueur de voir son meilleur score atteint et de se fixer des objectifs pour améliorer ses performances.

Le joueur gagne des points à travers plusieurs mécanismes dans le jeu :

1. **Lancement des Ingrédients** : Chaque fois qu'un ingrédient est lancé par le canon, le joueur reçoit un certain nombre de points basé sur l'indice de l'ingrédient. Par exemple, un oignon (indice 1) rapporte 1 point, tandis qu'un cochon (indice 9) rapporte 9 points.
2. **Fusion des Ingrédients** : Lorsque deux ingrédients identiques fusionnent, le joueur gagne des points selon une formule exponentielle :  $2^{(n-1)}$ . Cela incite le joueur à réaliser des fusions pour augmenter significativement son score.
3. **Réalisation des Recettes** : La réalisation d'une recette donne également des points, calculés comme la somme des points de fusion de chaque ingrédient, multipliée par 1,5. Ce système récompense les joueurs qui parviennent à suivre les recettes tout en gérant l'espace limité de la marmite.

Pour renforcer l'engagement et offrir un retour positif au joueur, des éléments de feedback visuel ont été intégrés :

**Affichage des Points lors des Fusions** : Lorsque deux ingrédients fusionnent, les points gagnés sont affichés directement dans la scène, toujours orientés vers le joueur. Cela permet de visualiser instantanément la récompense de chaque fusion.

**Affichage des Points des Recettes** : Lorsque le joueur complète une recette, les points obtenus sont affichés sous la recette dans l'interface utilisateur, offrant un feedback clair sur la réussite de l'action.



*Points recette*

*Points fusion*

**Animation du Texte de Score** : Chaque mise à jour du score est accompagnée d'une animation légère du texte, le faisant grossir puis revenir à sa taille normale, pour attirer l'attention du joueur sur l'augmentation du score. Cela crée une sensation de dynamisme et de satisfaction à chaque gain de points.

### c) Modes de jeu

**Crazy Cook Fusion** propose trois modes de jeu distincts, chacun offrant une expérience unique et des défis variés pour les joueurs. Bien que chaque mode ait ses spécificités, l'objectif commun reste le même : *éviter que les ingrédients ne débordent de la marmite.*



#### 1. Mode Survival

Le mode *Survival* est le mode classique qui teste l'endurance et la stratégie du joueur. Le but est de continuer à fusionner les ingrédients aussi longtemps que possible tout en évitant que la marmite ne déborde.

- Lorsque le joueur commence une partie en mode *Survival*, le jeu débute avec une recette aléatoire.
- Une fois cette recette complétée, une nouvelle recette est sélectionnée, avec une difficulté accrue si la partie est avancée.
- Le joueur doit fusionner les ingrédients pour gagner des points et faire de la place dans la marmite.
- Le jeu continue jusqu'à ce qu'un ingrédient sorte de la marmite, mettant fin à la partie.



*Survival : en jeu*

#### 2. Mode Recipe

Dans le mode *Recipe*, le joueur choisit la recette qu'il souhaite réaliser et se concentre sur la fusion des ingrédients spécifiques pour la compléter.

- Le joueur sélectionne une recette parmi celles disponibles avant de commencer la partie.
- Le but est de rassembler et de fusionner les ingrédients requis pour compléter la recette.
- Le joueur peut obtenir de 0 à 3 étoiles en fonction de sa performance. Si le joueur utilise trop d'ingrédients non requis, il obtiendra moins d'étoiles.
- Chaque recette a son propre lore, ajoutant une dimension narrative et immersive au jeu.



*Recipe : Choix  
recette*

### 3. Mode Daily

Le mode *Daily* propose un défi quotidien unique, où le joueur doit réaliser une recette spécifique en utilisant trois ingrédients choisis aléatoirement.

- Chaque jour, une nouvelle recette est générée aléatoirement et ne peut être jouée que le jour même. Cette recette est composée de trois ingrédients choisis aléatoirement parmi tous les ingrédients disponibles dans le jeu.
- Une fois générée, la recette du jour est sauvegardée à l'aide de *PlayerPrefs*. Cela permet non seulement de jouer la recette le jour même, mais aussi de conserver un historique des recettes jouées ou ratées durant la semaine précédente.
- Le système vérifie chaque jour s'il existe déjà une recette générée pour la date actuelle. Si ce n'est pas le cas, une nouvelle recette est créée et sauvegardée. Ce processus assure que chaque joueur reçoit un défi quotidien unique
- Lorsque le joueur complète la recette du jour, son état est marqué comme "complété", et le temps de réalisation est enregistré. Cela permet de conserver une trace des performances du joueur pour chaque recette quotidienne.



*Daily : Choix du jour*

Ces trois modes offrent une variété d'expériences adaptées à différents styles de jeu, qu'il s'agisse de défis prolongés, d'objectifs spécifiques, ou de défis quotidiens renouvelés.

## V. Expérience utilisateur

### a) Menus

Les menus de *Crazy Cook Fusion* sont conçus pour être simples, intuitifs et accessibles, peu importe la langue des joueurs. Plutôt que de recourir à des textes, les menus s'appuient sur des icônes compréhensibles et universelles. Chaque icône est soigneusement choisie pour représenter clairement son action, facilitant ainsi la navigation pour tous les joueurs, quels que soient leur âge ou leur langue.

Pour éviter que les menus ne paraissent statiques et monotones, des animations dynamiques sont intégrées. Par exemple, lorsque le joueur interagit avec un bouton, celui-ci réagit par un léger mouvement pour indiquer qu'il a bien été activé. L'affichage des recettes, lui, bénéficie d'une animation subtile où il semble légèrement pivoter, comme s'il était emporté par une brise légère, ajoutant une touche de vie et de dynamisme à l'interface. De plus, lorsque le joueur gagne des points, le score à l'écran grandit temporairement pour souligner l'accomplissement. Enfin, des animations spéciales accompagnées de particules visuelles sont déclenchées lors de l'obtention des étoiles à la fin d'une partie, rendant le moment plus gratifiant et mémorable.

### b) Musique et son

Pour enrichir l'expérience utilisateur, une attention particulière a été portée à l'environnement sonore de *Crazy Cook Fusion*. Chaque action en jeu est accompagnée de sons distincts, renforçant l'immersion et le feedback pour le joueur. Par exemple, un son spécifique est joué lors de la fusion des ingrédients, un autre lors de l'augmentation du score, et d'autres encore pour le chargement du canon, le tir, ou lorsqu'il tire en surchauffe.

Les musiques de *Crazy Cook VR* ont été réutilisées dans ce jeu, apportant une cohérence sonore et rappelant l'univers de base. Le menu principal est accompagné d'une musique en boucle, tandis que le jeu propose 11 morceaux différents qui se jouent aléatoirement pour éviter la répétition et maintenir une ambiance dynamique. Le joueur a également la possibilité de régler le volume des effets sonores et des musiques séparément dans les paramètres, offrant un contrôle total sur leur expérience auditive.

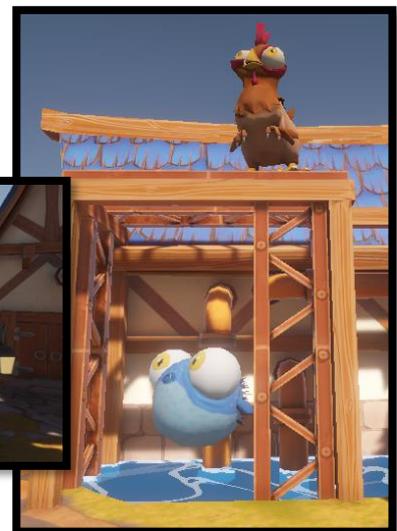


*Paramètres des sons*

### c) Environnement et décors

L'environnement de *Crazy Cook Fusion* a été soigneusement travaillé pour offrir une expérience visuellement riche et vivante. De nombreux éléments décoratifs ont été ajoutés, tels que des maisons, des caisses, des lampadaires et des arbres, afin de créer un monde cohérent et immersif. Ces décors ne sont pas simplement posés en arrière-plan, mais s'intègrent de manière fluide à l'ensemble du jeu, contribuant à l'ambiance générale.

En outre, pour rappeler constamment le gameplay, des ingrédients animés sont disposés à divers endroits de l'environnement. Ces animations ajoutent une touche d'interactivité et de dynamisme, rendant le monde du jeu plus vivant et engageant pour les joueurs. L'ensemble de ces éléments contribue à immerger le joueur dans l'univers loufoque et coloré de *Crazy Cook Fusion*, enrichissant ainsi l'expérience globale.



#### d) Lancement

Le lancement de *Crazy Cook Fusion* a été une étape cruciale du projet. Pour monétiser le jeu, des publicités Unity ont été intégrées, permettant de générer des revenus.

En parallèle, le jeu a été soumis à Google Play Console pour être publié sur le Play Store. Ce processus a nécessité la saisie de nombreuses informations et le remplissage de plusieurs formulaires complexes. Chaque détail, de la description du jeu à la configuration des paramètres de confidentialité, a été soigneusement renseigné pour répondre aux exigences de la plateforme.

Cependant, malgré ces efforts, le jeu a été refusé en raison d'un SDK qui collectait des données non conformes aux politiques de Google. Par manque de temps, je n'ai pas pu résoudre ce problème, ce qui a empêché la sortie du jeu sur les plateformes. Néanmoins, cette expérience m'a permis de mieux comprendre les exigences et les défis liés à la publication d'un jeu mobile, tout en soulignant l'importance de la conformité et de la gestion des SDK dans le développement de jeux.



## VI. Résumé de Crazy Cook Fusion

*Crazy Cook Fusion* est un projet qui illustre parfaitement l'intégration créative de concepts existants dans un nouveau contexte, tout en exploitant pleinement les ressources disponibles. Le jeu réinterprète le concept populaire du *Suika Game* en 3D, le combinant avec l'univers visuel et sonore de *Crazy Cook VR* pour créer une expérience unique et immersive.

L'idée de base a été de transformer un jeu simple mais addictif en une version plus riche et complexe, en ajoutant des éléments de gameplay comme la gestion des recettes, un système de fusion d'ingrédients, et plusieurs modes de jeu pour varier les défis. Le tout a été enveloppé dans une présentation soignée, avec des menus intuitifs, une bande sonore captivante, et un environnement visuellement dynamique.

*Crazy Cook Fusion* représente non seulement une évolution technique et créative, mais aussi un aboutissement dans la création d'une expérience de jeu cohérente et engageante. Ce projet m'a permis de renforcer mes compétences en développement de jeux mobiles, en gestion de projet et en design d'expérience utilisateur, tout en offrant un jeu qui, je l'espère, continuera à divertir et à captiver les joueurs.

# Annexes

## I. Liens de téléchargement

*Télécharger Crazy Cook Fusion*

