

30 AVRIL 2025

UQAC

Université du Québec  
à Chicoutimi

# POST-MORTEM

REALISATION DE JEUX VIDEO

PIERRE VANHOVE

VANP26010500

Equipe ZOULOU

# Sommaire

---

|   |    |
|---|----|
| Sommaire .....                                    | 1  |
| Section 1 : Présentation.....                     | 2  |
| Titre du Projet : .....                           | 2  |
| Argumentaire éclair :.....                        | 2  |
| Brève description du projet.....                  | 2  |
| Les fonctionnalités que vous avez réalisées.....  | 2  |
| Vaisseau.....                                     | 3  |
| Fuel .....  | 3  |
| Deathzone et Checkpoint .....                     | 3  |
| Debug .....                                       | 4  |
| Section 2 : Retour sur la tranche verticale ..... | 5  |
| La qualité générale .....                         | 5  |
| Le « Game Feel », le « Juciness » .....           | 5  |
| L'architecture .....                              | 6  |
| Points positifs de la production.....             | 8  |
| Points négatifs de la production.....             | 8  |
| Suggestions à appliquer en production .....       | 9  |
| Section 3 : Scoping .....                         | 10 |
| Mécaniques supplémentaires et variations .....    | 10 |
| Première histoire utilisateurs .....              | 10 |
| Seconde histoire utilisateurs.....                | 10 |
| Environnement.....                                | 10 |
| Hub.....  | 10 |
| Niveaux additionnels.....                         | 11 |
| Section 4 : Conclusion .....                      | 12 |

Voir annexe en fin de document

## Section 1 : Présentation

---

### Titre du Projet :

Ziggy Found Life (But Ran Out of Fuel on His Way Home)

### Argumentaire éclair :

Parcourez des planètes mystérieuses avec votre tuyau d'essence extensible, récoltez du carburant et trouvez le chemin optimal avant de manquer de longueur et de rester piégé à jamais !

### Brève description du projet

Ziggy Found Life (But Ran Out of Fuel on His Way Home) est un jeu d'exploration de planètes en vue à la troisième personne. Vous incarnez Ziggy, un astronaute maladroit qui, après avoir découvert la vie extraterrestre, doit maintenant trouver du carburant pour regagner son vaisseau spatial et annoncer sa grande découverte.



Pour y parvenir, Ziggy devra manœuvrer un long tuyau d'essence à travers divers obstacles, en cherchant toujours le chemin le plus direct. Attention : si le tuyau atteint sa longueur maximale, Ziggy risque de rester coincé à jamais !

### Les fonctionnalités que vous avez réalisées

J'ai eu la possibilité de travailler sur différentes fonctionnalités tout au long de la tranche verticale. Cependant, la fonctionnalité principale que j'ai développée et qui sera la plus détaillée dans ce rapport est le fuel, que nous étudierons plus tard.

## Vaisseau

Pendant la tranche verticale, je me suis occupé de développer la logique autour du vaisseau. Cette dernière n'était pas une fonctionnalité très complexe ; elle servait surtout de zone de fin du jeu et me servait également à centraliser certains delegates d'UI ou à centraliser certaines fonctions et propriétés d'autres actors via des delegates également.

## Fuel

Cette fonctionnalité est la plus importante que j'ai réalisée. Je devais être capable de créer des sources de fuel que le joueur pouvait aspirer. Je pensais au début pouvoir faire un plan avec un VFX pour l'aspiration, mais les artistes m'avaient donné comme exemple les liquides dans le jeu « Mais, où est Swampy ? », qui possède une gestion des liquides très différente. En effet, le jeu donne l'impression que toutes les particules sont indépendantes et réagissent entre elles.

J'ai donc décidé d'utiliser des Metaballs afin de faciliter les calculs et d'obtenir l'effet escompté. Le problème était que toutes les Metaballs recevaient différentes forces par frame (gravité, répulsion entre les Metaballs, restriction à l'intérieur d'un objet). Après plusieurs essais, j'ai eu de nombreuses difficultés à optimiser les Metaballs lorsque leur nombre était très élevé. La solution a alors été d'augmenter la taille de chaque Metaball afin de réduire leur nombre en jeu.

L'avantage des Metaballs est qu'ils avaient bien le comportement souhaité et qu'ils sont relativement simples à utiliser.

En revanche, ils m'ont généré de nombreux bugs, dont beaucoup causaient l'arrêt du jeu. J'ai passé une bonne partie de mon temps à supprimer ces bugs.



## Deathzone et Checkpoint

Cette fonctionnalité était très simple, bien que j'aie eu plusieurs problèmes de build avec cette dernière. En effet, nous avons fait le choix de ne pas avoir de mesh visible dans le jeu pour indiquer les checkpoints, ce qui avait pour conséquence de rendre l'actor plus difficile à placer dans le jeu. J'ai donc voulu simplifier la tâche des level designers en

ajoutant un billboard pour qu'il soit visible dans le level et disparaisse au lancement du jeu. J'avais pour cela utilisé les macros de précompilation (#if – #endif) afin que les billboards n'apparaissent que dans l'éditeur. Cependant, pour une raison que j'ignore, lorsque j'effectuais un build, l'actor disparaissait complètement.

Pour régler ce problème, j'ai simplement supprimé la précompilation et spécifié à Unreal que le billboard devait être « hidden in game ».

## Debug

Ceci n'est pas vraiment une fonctionnalité, mais étant donné l'état du projet la semaine précédant la remise, et en prenant en compte que, même en tant que Firefighter, j'ai passé la majorité de mon temps à supprimer les bugs ou à aider les artistes à accéder à certaines variables ou autres, je trouvais cela important de le préciser. En effet, je suis repassé sur presque toutes les fonctionnalités du projet afin de supprimer les bugs rapidement, pour que le projet avance bien et que les artistes puissent continuer de travailler facilement.

## Section 2 : Retour sur la tranche verticale

---

### La qualité générale

Je pense que nous avons réussi à atteindre nos objectifs, même si cela semblait impossible une semaine avant la remise. Nous avons rencontré bon nombre de difficultés, que je détaillerai par la suite dans les [points négatifs de la production](#). Malgré tout, je pense que l'état actuel du projet, bien que, je l'espère, suffisant pour le travail demandé, aurait pu être meilleur, notamment en termes de « Game Feel ». En effet, d'après ma propre expérience et les retours que j'ai eus lors des playtests, certains comportements sont « étranges » et ne correspondent pas vraiment à l'idée initiale. Je pense par exemple au câble du vaisseau, qui présente toujours plusieurs bugs de collision, ainsi qu'à la rétractation, qui n'est pas vraiment naturelle et possède également des bugs de collision.

Cependant, je suis personnellement content d'avoir réussi à rendre le jeu d'une telle qualité et sans aucune fonctionnalité manquante. Je pense malgré tout que la tranche verticale permet de refléter les intentions que nous souhaitions transmettre pour le jeu complet.

Aucun playtest officiel n'a pu être réalisé dans ce projet, ce qui a énormément impacté la qualité générale de la tranche verticale. La partie technique de nos mécaniques principales n'était pas fonctionnelle avant quelques jours précédant la fin de la tranche verticale. Il était donc impossible pour nous de faire tester le jeu à des personnes extérieures au projet, et donc d'ajuster les mécaniques en fonction des retours des playtests.

En conclusion, je dirais que même si le projet a rencontré de nombreux problèmes et possède toujours des bugs dans le build, je suis satisfait de ce que nous avons réussi à produire avec l'équipe.

### Le « Game Feel », le « Juciness »

Selon moi et selon les tests, la mécanique principale du jeu manque de finition et, par conséquent, le « Game Feel » n'est pas incroyable. Nous n'avons pas l'impression que le câble se rétracte de manière naturelle. De la même manière, comme le câble présente des bugs de collision, ce n'est pas toujours très plaisant de le voir s'enrouler autour de l'environnement. D'autres fonctionnalités mériteraient d'être retravaillées pour améliorer le

feeling du jeu. Je pense par exemple à la plateforme, qui fonctionne bien mais qui peut parfois créer des bugs avec le câble, ou encore au fuel, qui mériterait d'être passé en compute shader pour augmenter leur nombre comme souhaité, sans perte de FPS.

En conclusion, je pense que le jeu mériterait quelques perfectionnements afin d'améliorer le « juiciness » du jeu. Si aucun bug n'apparaît pendant la partie, en revanche, je trouve que le jeu est plutôt bon et agréable à jouer.

## L'architecture

Toutes les fonctionnalités que j'ai développées ont été pensées pour être facilement modifiables par les artistes et les designers. Mon objectif était de leur laisser le plus de liberté possible pour modifier les variables nécessaires à l'utilisation de la fonctionnalité en question.

La fonctionnalité qui m'a posé le plus de soucis est le fuel avec les metaballs. Les metaballs me semblaient être la solution la plus simple pour gérer le fuel et obtenir le résultat escompté.

L'approche que j'ai utilisée avec les metaballs est la suivante :

- Le générateur de fuel crée le nombre de metaballs nécessaire pour obtenir le fuel.
- Chaque metaball se voit attribuer, toutes les 0,1 seconde, trois forces différentes :
  - Une force de gravité de la planète,
  - Une force de répulsion entre les metaballs proches,
  - Une force de « clamping » à l'intérieur d'une capsule component afin d'éviter que les metaballs ne s'échappent.
- Lorsque le joueur aspire une metaball, un delegate est envoyé afin de signaler sa suppression avant de détruire l'acteur.
- Chaque metaball porte un nombre prédéfini de fuel qui s'ajoute à la somme totale de carburant.

Cette approche fonctionne très bien, mais elle est très coûteuse en performances, et j'ai donc dû limiter aux designers le nombre de metaballs à générer.

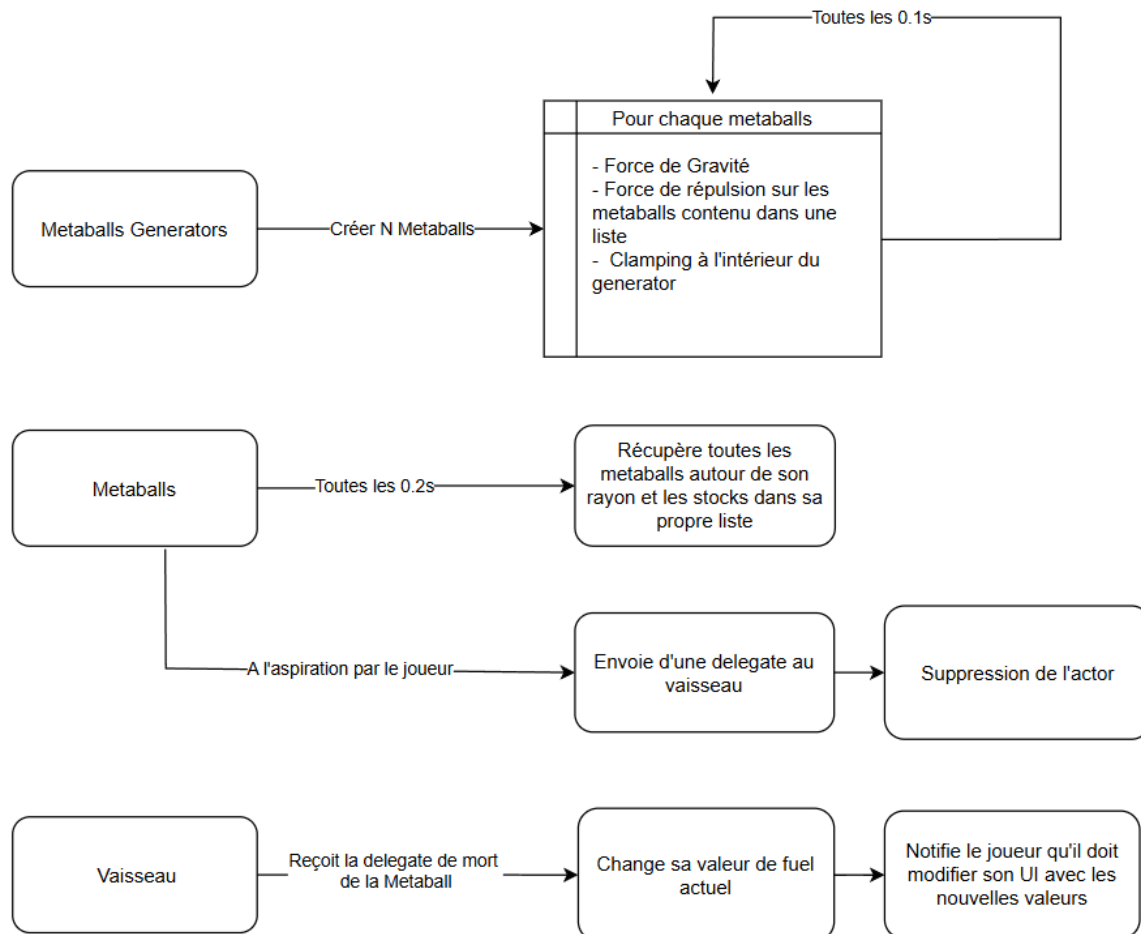
Cette fonctionnalité est celle qui m'a confronté aux plus grandes difficultés. J'ai dû la retravailler jusqu'à la fin de la tranche verticale.

Sur cette fonctionnalité uniquement, j'ai effectué 21 pushes Perforce différents en combinant ceux sur Main et sur Dev. Tous les pushes correspondent soit à des ajouts, soit à des modifications du comportement des metaballs. Au total, j'ai réalisé trois ou quatre modifications complètes du comportement pour essayer d'atteindre les objectifs fixés.

Par exemple, j'ai essayé de changer la méthode de clamping, qui est de loin la plus coûteuse des forces. En effet, comme les designers souhaitaient une capsule plutôt qu'une sphère, j'ai dû modifier la fonction de clamping pour correspondre au nouveau design. Le problème est que faire des vérifications dans une capsule est bien plus coûteux que dans une sphère, et cela fut la principale raison de ma perte de FPS dans le jeu.

Pour optimiser les metaballs, j'ai essayé plusieurs méthodes, notamment le time slicing et la parallélisation. Cependant, la parallélisation a créé plusieurs problèmes, notamment des crashes. Pour cette raison, j'ai décidé de la supprimer.

Actuellement, le comportement des metaballs correspond au schéma ci-dessous :





En général, sur toutes les fonctionnalités que j'ai pu développer, j'ai utilisé les fonctionnalités suivantes d'Unreal :

- Delegates
- Billboard
- Macro de précompilation (#IF WITH\_EDITOR)
- Calculs physique
- Shaders pour créer les metaballs
- Timer pour remplacer Tick
- Time slicing

## Points positifs de la production

Malgré les difficultés que nous avons eues pendant la tranche verticale, je trouve tout de même des points positifs. Premièrement, je trouve que, pendant la dernière semaine de travail, la communication avec les autres membres de l'équipe s'est grandement améliorée, en raison de l'urgence. J'ai surtout échangé avec les artistes, qui trouvaient régulièrement des bugs et avaient besoin de quelqu'un pour les corriger rapidement.

En plus de cela, tout le reste de l'équipe a beaucoup travaillé pour s'assurer que nous puissions remettre un travail fini et propre.

En résumé, je pense que l'équipe a beaucoup appris pendant la dernière semaine et a corrigé la majorité des problèmes que nous avons rencontrés pendant le reste de la production.

## Points négatifs de la production

Pendant la production, de nombreux points ont ralenti le travail et la création des builds. En parlant des builds justement, seulement trois d'entre eux ont été réalisés jusqu'à la dernière semaine de développement. La principale raison vient des développeurs. En effet, tant que nous n'étions pas sûrs d'avoir corrigé les bugs pouvant provoquer des fatal errors, nous ne faisions pas de merge vers Main.

De plus, nous avons rencontré des problèmes avec Perforce pendant un merge, ce qui a beaucoup ralenti la production. En plus de cela, certains conflits ont eu lieu entre les artistes et un membre de l'équipe de développement.

Finalement, je soulignerai une dernière chose : le temps qu'il a fallu attendre pour qu'une fonctionnalité soit disponible dans le projet. La personne concernée a eu toute la durée de la tranche verticale pour fournir cette feature, et nous ne l'avons reçue que le mercredi soir avant la remise. J'ai dû décaler deux builds pour attendre que cette feature soit disponible, et elle n'était fonctionnelle qu'après que Valère a modifié son comportement pour en faciliter la mise en place.

## Suggestions à appliquer en production

Pour la production, je pense qu'il faut d'abord retravailler certaines fonctionnalités pour qu'elles soient plus intéressantes, mais aussi s'assurer que la communication entre les développeurs et les artistes soit opérationnelle dès le début de la production.

À part cela, je pense que le jeu représente bien ce que nous souhaitons réaliser.

Je pense que la prochaine équipe de développeurs devra fusionner son travail avec la branche Main plus régulièrement afin de produire des builds fréquents. Peut-être qu'avoir un Firefighter désigné pour réaliser les builds dès le début pourrait aider à atteindre l'objectif fixé pendant la production.

## Section 3 : Scoping

---

Selon moi, soit le scoping du jeu était certes un peu trop ambitieux, mais je pense qu'avec le nombre de développeurs que nous avons, il était tout à fait possible de réaliser toutes les mécaniques que nous avons prévues.

En effet, nous avons au maximum une fonctionnalité par développeur, ce qui, selon le temps disponible, était à l'origine possible.

Cependant, avec la tournure qu'a prise le projet par la suite, le scope s'est effectivement révélé être trop grand pour le rendu final.

### Mécaniques supplémentaires et variations

#### Première histoire utilisateurs

En tant que joueur, je dois pouvoir choisir les planètes sur lesquelles je souhaite me rendre à partir de mon vaisseau.

#### Seconde histoire utilisateurs

En tant que joueur, je veux que certaines planètes soient inaccessibles tant que je n'ai pas trouvé suffisamment de carburant, afin d'encourager l'exploration complète de chaque planète.

### Environnement

#### Hub

Un hub simple et fonctionnel sera conçu (possiblement basé sur l'intérieur du vaisseau spatial de Ziggy), permettant au joueur de choisir quelle planète visiter une fois assez de carburant collecté.

## Niveaux additionnels

### *Planète 1 :*

Prototype actuel, basé sur des kits d'assets déjà existants (rochers, champignons géants, ruines).

### *Planète 2 :*

Nouvelle planète naine/tutoriel (ex : Lune)

### *Planète 3 :*

Nouvelle planète thématique (ex : planète de glace, planète de sable, planète industrielle abandonnée, etc).

## Section 4 : Conclusion

---

En conclusion, malgré les nombreux défis rencontrés tout au long de la production, je suis satisfait du résultat que nous avons réussi à atteindre en équipe. Le projet, même s'il reste perfectible en termes de « Game Feel » et de finition, reflète globalement l'expérience de jeu que nous souhaitons transmettre.

La dernière semaine, bien que très intense, a permis de rendre un travail satisfaisant pour la tranche verticale.

Si le projet devait se poursuivre, je recommanderais de concentrer les efforts sur l'amélioration du « Juiciness » du gameplay, de mieux structurer les builds et de renforcer la communication dès les premières semaines de développement. Avec ces ajustements, je suis convaincu que le jeu pourrait atteindre un niveau de qualité bien supérieur.

# Annexe 1 : Composition de l'équipe

## **Artistes :**

- William Charbonneau
- Alexandre Adam-Goyette
- Jean-Félix Côté
- Vincent Galipeau
- Tonine Ghanous
- Youssef Jazmati
- Gabriel Mailloux
- Sandrine Pelletier

## **Développeurs :**

- Hüseyin Çakar
- Camille Girodengo
- Martin Queval
- Hugo Vincente
- Quentin Bouëtel
- Pierre Vanhove