

Borderline Racing Group Project Retrospective

By James (Jamie) Marland – 23100010

PR5001 – May 2025 – Word count: 3072



Figure 1: Cinematic Trailer.

1. Introduction

1.1. Overview

This project retrospective covers my experience of the PR5001 Module - Industry Studio Project. This project aimed to mimic working as a team in industry to make a game. Specifically, we were tasked with creating a vertical slice racing game based on an AAA game in Unreal Engine 5. The team chose Borderlands 3, which is known for its iconic inked art style and chaotic gameplay (Gearbox Software, 2020). The project took over fourteen weeks, but the project time was twelve weeks with breaks removed. That time was broken down into:

- Four Weeks of pre-production.
- Five Weeks of production.
- Two Weeks of polish.
- One Week of reflection.

1.2 Deliverables

On top of the project itself, we had a fair amount of other deliverables, some of which are shown in Figure 2:

- Art bible - The game's visual style and expression are outlined to communicate the game's art direction easily (Polycount, 2025).
- Asset list - A list of all the assets made for the game.
- Game Design Document - A living document that showcases the game's mechanics and design principles, helping break down the game (Ashby, 2024).

- Playtesting evidence - A showcase of consistently getting player feedback compiled and referencing it for future decisions.
- Schedule - The week-to-week team-wide plan.
- Technical documentation - Any tools made for development, and how to use them.
- Treatment - Game's concept and design pillars.
- Beauty shots - High-res screenshots in-game.
- Playthrough recordings - Showcase of gameplay.
- Cinematic - Finally rendered cinematic trailer.
- Contribution list - Each individual team member's contributions.

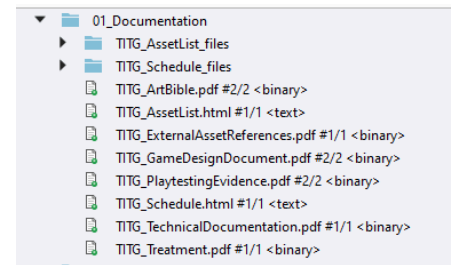


Figure 2: Documentation deliverables.

2. Engagement

2.1. Role

My role in the team project was a mixture of multiple disciplines:

- As the team's technical artist, this was my first role; I was in charge of all the VFX, from environmental to items and vehicles shown in Figure 3. I also created tools for the artists, since most of the team's artists wanted a lot of creative control and did not need many complex tools. I also helped organise the asset pipeline and limitations required for the artists. Materials were another key area in which I set up all master materials, added features as requested by the artists, and organised them into artist-friendly parameters. Finally, as a technical artist, I oversaw the Unreal project's optimisation, including nanite, LODs, shader complexity, and more. I also managed the project sub-level system and folder structure, and kept assets in the style guide.

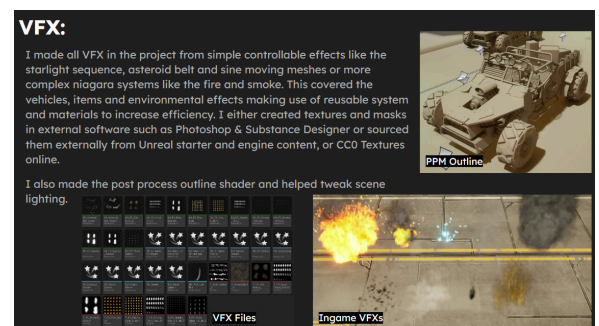


Figure 3: Individual contribution list - VFX.

- I also worked as a programmer for the team. I was involved in developing most mechanics from the behaviour of vehicles and items, such as item handling, damage components, the vehicle parts visual and stat systems shown in Figure 4. I also helped compile feedback from playtesting sessions, as shown in Figure 5.

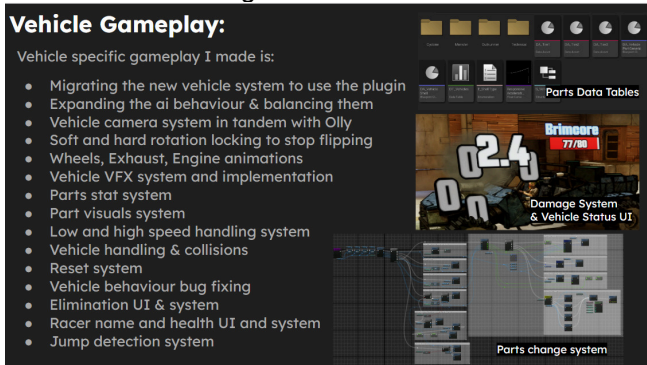


Figure 4: Individual contribution list - Vehicle Gameplay.

13/05 Playtesting Feedback session:

- Bots are too fast at accelerating compared to the player
- The start position is too cramped
- AI doesn't go for one item box enough
- The item box's hitbox is too large
- Lose too much momentum when hitting another vehicle
- Gravity is too floaty
- The selection ui needs to be clearer
- Need a podium area
- The UI layout needs to align correctly
- Menu camera shots
- Loading screen

18/05 Playtesting Feedback session:

UI:

- Vehicle stats in the UI Menu
- Game Explanation
- Vehicle Parts Explanation
- Item Explanation
- Vehicle Parts
- Item UI
- Item Button ingame prompt UI
- Character name changes

Item:

- Item behaviour needs to be more telegraphed
- Item use cases need to be clearer

Vehicle:

- The vehicle needs to have different stats and a base speed increase
- Cyclone keeps falling over
- Vehicles shouldn't be able to drive through each other
- Potential Boost meter
- AI are too robotic
- AI drive into other vehicles
- Air Control
- A Warmup Lap to get players accustomed to the layout before elimination starts

Figure 5: Playtesting evidence.

- I was also the team's producer, which meant I was in charge of keeping the team on track over the 14 weeks and organising SCRUM meetings and deliverables on deadlines as shown in Figure 6. While the creative decisions were always made as a team, they needed to be thought through, and that was a big part of my responsibility. Resolving and coordinating technical or team issues also fell to me.

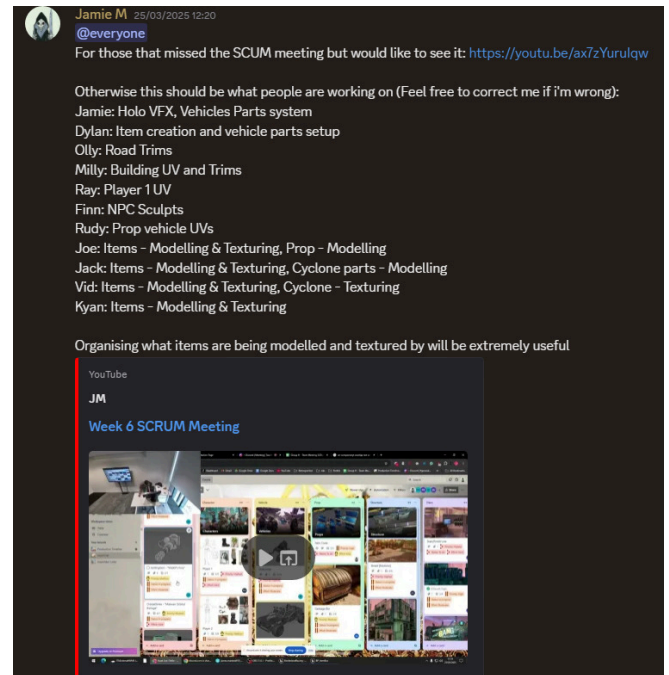


Figure 6: Result of SCRUM.

It was essential that I handled them in order of priority for the project, so I coordinated with the team to focus on the most critical elements of art and gameplay first, building layer by layer, tools and gameplay shown below in Figure 7. So, one of my first focuses was getting an MVP of gameplay mechanics and vehicle behaviour and making initial artist tools like the cables and master materials. Then later in the project, I was able to work on optimisation and VFX, and then on final gameplay polish again.



Figure 7: Screenshot from MVP.

2.1. Workflow

I approached tasks with a focus on industry practice, which meant we worked as a team to prioritise important tasks and create iterations. We focused on making an MVP for our project, which means a minimum viable product where iteration is the primary focus. The goal is to get a playable version as early as possible, with consistent internal and external feedback that drives decision-making (Rouse, 2020).

As a result, I developed my work with that in mind, first gathering references from the game and breaking them down into parts, building iterations of assets as needed, effectively keeping my work on track and showing results. I would consistently update my team on my progress and stay in line with the team. An example of a tool that was quickly scrapped due to a lack of artists' needs is shown below in Figure 8.

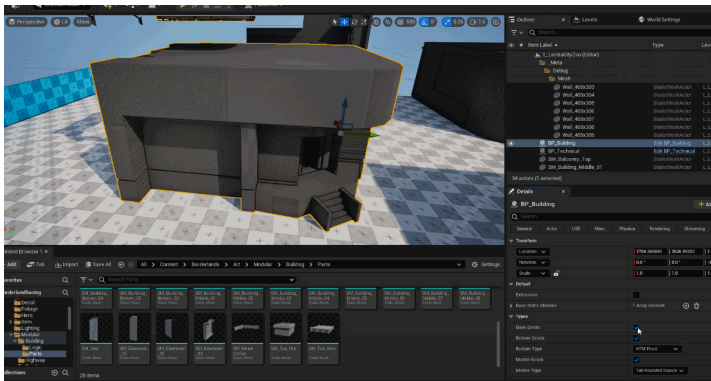


Figure 8: Scrapped building tool.

I consistently stepped back and reapproved priorities, pushing less important aspects back and focusing on more important ones, like the master materials. As is my role in the team, I updated them frequently with artist requests for use while balancing their optimisation as shown in Figure 9. This workflow proved effective at keeping goals in line with the team's.

Optimisation:

I was in charge of all optimisation in the project by tweaking asset properties, project settings, using nanite and lumen settings, LODs and more. This involved a careful balance of optimising performance and preserving detail. Assets that could use nanite did and if they couldn't due to shader model they used traditional LOD methods. Shader complexity was kept low on the main material despite a range of uses and assets with high shader complexity were optimised in other ways to maintain performance. The final result using a range of optimisation techniques to keep a stable fps despite a highly detailed scene contain 2,500+ Actors at high settings level allowing for a smooth player experience.



Figure 9: Individual contribution list - Optimisation.

My work as a producer also involved tracking and documenting team-wide progress as shown in Figure 11, adjusting our project scale to align with our timeframe, and balancing each member's workload. After feedback sessions, I would compile and incorporate our feedback into our following weeks of work, balancing the scale of work and priority of new tasks into existing plans as shown in Figure 10. These behaviours align with industry practice, which was a central point of this project and successfully helped our project be completed efficiently.



Figure 10: Priority of aspects from the production timeline.

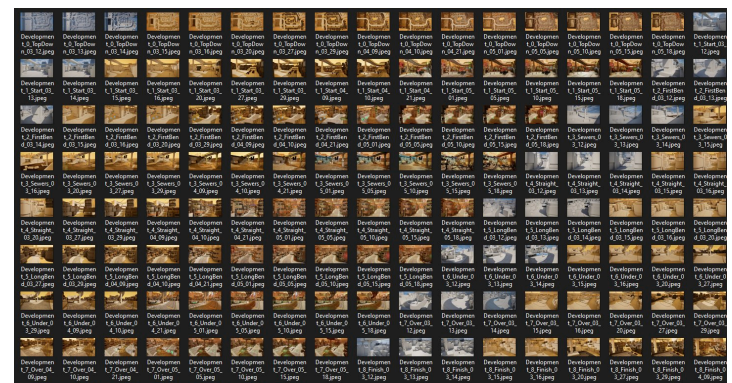


Figure 11: Compiled development screenshots.

Balancing my personal commitments with what the team needed and tracking it was a challenging but rewarding task.

2.2. Adapting to feedback

Both as a team and individually, we adapted throughout the project; first, as a team, the initial setup of roles was never designed to be hard-set, and that proved to be for the best, as teammates collaborated across various tasks and areas.

Having some initial setup for production proved helpful as having a Trello which is a tool for product's development and listing required tasks (Duffy and Vatu, 2024) was beneficial to laying out what would be done across the length of the project and what needed to be done, but became crowded towards the end and resorted to having one big to-do list on the screen for everyone with what needed to be done by who proved helpful in organising in the final two-week sprint as shown in Figure 12, especially when multiple assets were being completed in a day it aligned with the quicker assignment needed at the end of production and resulted in a highly effective sprint.

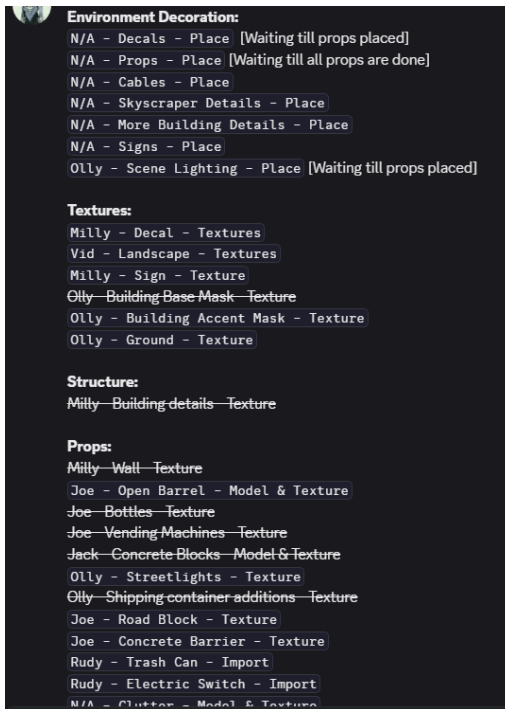


Figure 12: Final sprint team-wide to-do list.

Individually, I adapted to use a personal to-do list on top of Trello, for a similar reason to the team final sprint; it was useful to have only my tasks laid out and to focus on my goals.

External feedback was also valuable in refreshing what tasks were priorities and effectively realigning our goals. After each feedback session, the compiled feedback influenced what was prioritised, as shown in Figure 13. Vehicle physics, Track borders, and layout were all pushed forward or created as tasks after feedback, and many other examples of this exist in the project.

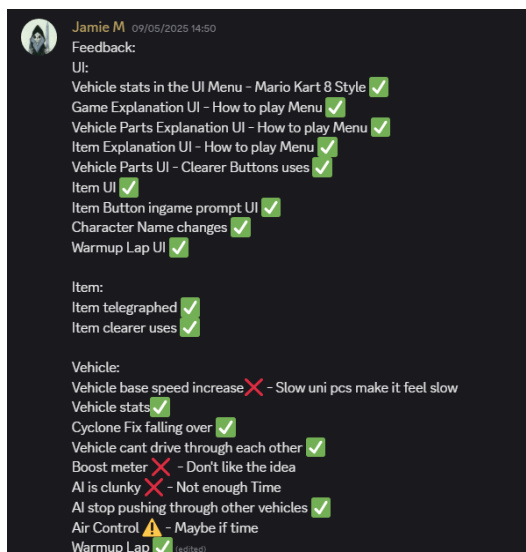


Figure 13: Compiled feedback.

The most outstanding example of our response to feedback was the vehicle movement extremely early on, both the team and playtesters were not happy with how the vehicles responded to player input, the nail in the coffin was that chaos physics vehicles do not support changing parameters at runtime as they are private classes while modular chaos vehicles exist its in a newer version in beta (Epic Games, 2025). Luckily, however, I was able to resolve this by changing our vehicle's underlying physics system to a third-party plugin called the Arcade vehicle system as shown in Figure 14. The plugin allowed for adapting stats at runtime and also much more responsive controls due to not being based on real physics. This was one of our group's most significant achievements for gameplay, and was shown in our feedback by immediately receiving much more positive feedback from playtesting sessions on our gameplay.



Figure 14: Fab store page (Furious Production LTD, 2021).

3. Evaluate

3.1. Team Decisions

This project stayed aligned with the industry by using Agile project management and SCRUM. Every Tuesday, the team would meet and discuss what had been worked on since the previous week and what tasks would be the focus for the following week.

As the producer and SCRUM master, I was in charge of directing this, and it was a great tool to figure out what the project needed and keep the team on the same page. Most issues were resolved with a plan of action in these SCRUMs, and decisions as to the best approaches were team-wide and influenced by every member's input, leading to a solid team-wide perspective on issues and goals.

Although the team didn't face too many issues, we did have an issue with two team members' participation, but with tutor support and project scale, we could work with it. The initial buffer time (Doknic, 2024) added to the project scale by me as producer was an effective tool as it gave room for these issues and caused much less stress for individual team members' workload, as we then could distribute work evenly while allowing team members a break who needed it for a multitude of reasons.

Again, the Trello production timeline was an effective schedule tool for tracking team members' tasks, defining what people needed to do in the week outlined on Discord, and keeping track of our team's plans as shown in Figure 15.



Figure 15: Trello production timeline.

3.2. Individual Decisions

Solving issues individually in my own production required consistently consulting the Borderlands games' design decisions, breaking them down and consulting tutors and team members for informal feedback on decisions. As shown in Figure 16, I was typically able to resolve issues quickly and iterate quickly, as well as track my progress throughout.



Figure 16: Comparison of the final asset after multiple iterations (left) vs Borderlands 3 (right) for a holo sign.

When contributing to the project, I communicated with my team to ensure that my contributions were an effective use of time and resources and would positively impact the final product. I also worked hard to keep my work organised and optimised for real-time. I also worked to make sure this is what a shared vision for the artists is, setting requirements for assets in terms of optimisations like poly counts and material slots and organising them accordingly into the right folders, sub-levels, and naming conventions. As shown in Figure 17.

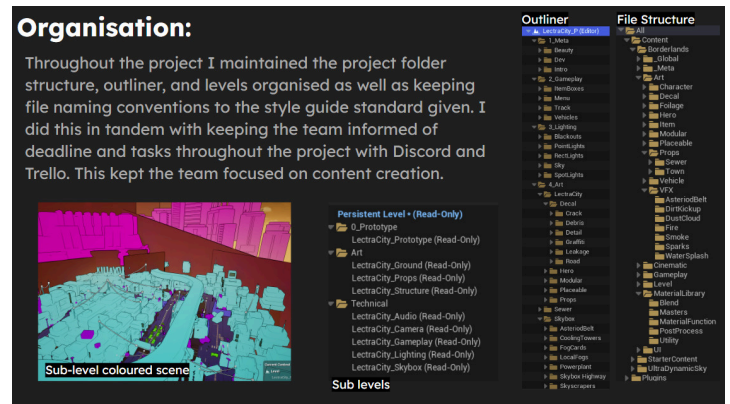


Figure 17: Individual contribution list - Organisation.

3.3. Industry Practice

Industry practice required following a specific set of stages in the project, with pre-production being a critical way to align the team's vision and project expectations:

- First, we created our game's Treatment shown in Figure 18, where we defined our game's core design pillars, core engagement loop, how it would play, and the experience we wanted to create, as well as a tagline summarising the game in one sentence. We had to make three ideas to flesh out and experiment with each team member's desire. Eventually, landing on the Borderlands idea as it held the right amount of initial comfort for artists while still pushing and learning new concepts.

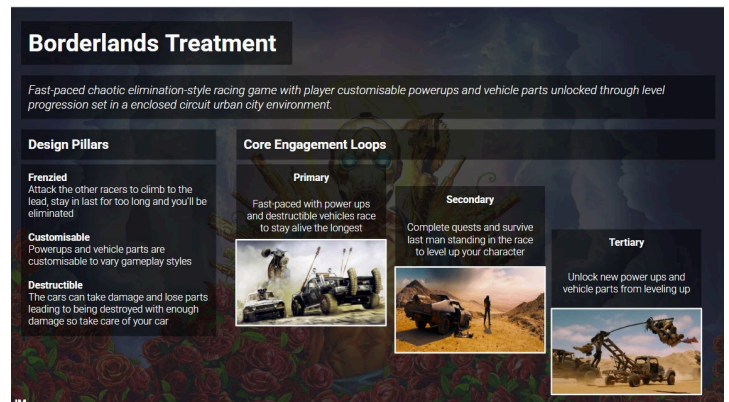


Figure 18: Our treatment.

- Then, we created the Art bible shown in Figure 19 for Borderlands, where we defined the characters, items, and props in the games and made top-down art direction decisions about how the game should look.



Figure 19: Art bible - VFX.

- Finally, an asset list and schedule decided the number of assets and which ones we would create, sorting them into the following categories: Items, Characters, Vehicles, Props, Structures, Heroes, and decals. These categories influenced initial team member roles and responsibilities. Shown in Figure 20.

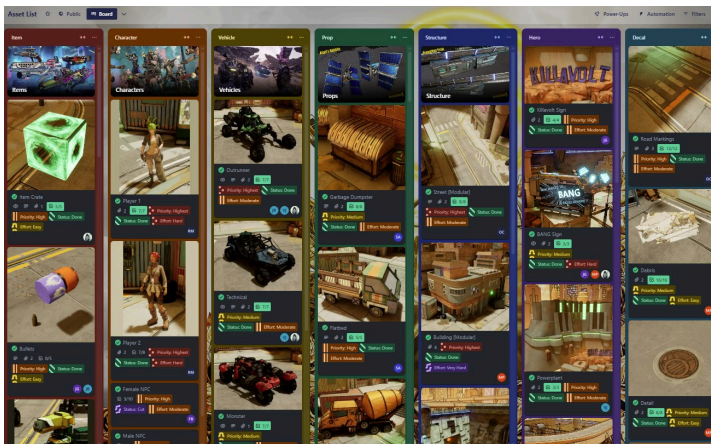


Figure 20: Asset List.

Another tool that helped us stay in line was our daily SCRUM notes, shown in Figure 21, which showed what each team member was working on and any problems faced; these were updated frequently to keep everyone up to date with what I was working on and had completed.

[illegible]

Figure 21: SCRUM notes.

Documentation is another industry practice that I made sure to align to not only did I do my SCRUM notes to document my progress I also created the technical documentation shown in Figure 23 for the tools made in the project to help artists understand how to use what was made, which was directly inspired by industry practice and is a crucial aspect of making high-quality tools and being an effective team member. I did so thoroughly and adapted tools dynamically to needs. For instance, the artists requested a vertex blending material, and I set up the system and explained thoroughly how to use it, as shown in Figure 22.



Figure 22: Final Vertex painting.

Skyscraper - [/Cable/BP_Skyscraper]

A placeable skyscraper that is randomly selected from a variety of skyscrapers and chooses a random tint, size, height, rotation, and whether the lights are on based on the world position and seed. It has a couple of controls:

- **Reset** - Turns off the skyscraper system, which is useful when moving the BP around.
- **Seed** - Allows for a different skyscraper variation even at the same point.
- **Extension On** - Creates a mirrored version below to extend its length.
- **Scale Variation Range** - The range the skyscraper scales between.
- **Height Variation Range** - The range the skyscraper height is between.
- **Rotation Variation Range** - The range the skyscraper rotates between.



Buildings - [/Building/BP_ModularBuilding]

A placeable building that has no initial meshes, as the artist requested direct control of it; however, the trims' colours used by the building can be changed to allow for artistic control over the randomised building colours

- **Color** - Turns the building system on with the building types.
- **Reset** - Turns off the building system, which is useful when moving the BP around.
- **Seed**- Allows for a different building variation.
- **Base Trims Pool** - Pool to choose the building's base colours from.
- **Accent Trims Pool** - Pool to choose the building's accent colours from.
- **Final Base Material** - Allows for manual override of Base.
- **Final Accent Material** - Allows for manual override of Accent.

Material: [/Borderlands/MaterialLibrary/Masters/...]

Opaque = [/M_MasterOpaque]

The main master material controls are self-explanatory; it automatically removes ink lines from ORM based on the Diffuse and allows for macro overlay textures to Diffuse and Normal, an Emissive mask can be used and adjusted, and an override for roughness and tinting of textures.

Sign - [/M_MasterSign]

The main master sign allows for holographic-style materials with customisable size, distortion, jitter, texture, scanlines, noise, pulse, and faults. The controls are self-explanatory.

Vertex painted layer blend - [/M_MasterBlend]

On meshes with the material blend setup this allows for artistic control over four layers on top of a base material that can be painted in the mesh point mode, painting should be done in one channel at a time, Red, Green, Blue, Alpha with a world aligned material that aligns textures seamlessly and has a world aligned noise textures to hide low vertex meshes with control similar to the opaque master material.

Vertex painting

External tools

Figure 23: Technical Documentation.

4. Reflection

4.1. Effectiveness

We produced a stunning final product as a team, and I am thrilled to have been involved in its development. The art, especially, was a significant point of compliments in feedback. This resulted in being personally very proud of our product. By setting a good initial framework, we also effectively responded to feedback and solved problems early, so as not to lead to major blocks later on. Our team's planning was a project highlight and avoided needing to scrap too much from our original goals.

Most of our team was also highly effective at communicating and showcasing progress made, resulting in informed and up-to-date decisions. Fast iteration of our team's game aspects was one of our biggest strengths and resulted in effective development.

Individually, I was an effective leader and team member in our team's development by keeping goals realistic and in scale and communicating team-wide problems as they came up. I also worked hard on contributing and committed considerable effort to accommodate team member requests and personal goals in the project, on top of mine and the team's.

Next time, we would consider a greater emphasis on gameplay depth, as this feels like the weakest area of our final product. A different brief would have been better, as the AAA art style racing game was a limiting factor in the creative expression of what we could make and the tools I could develop, the genre is known for being creatively stagnant and niche, being only 74th in revenue and largely held by only a few franchises as shown in Figure 24.

73	Automobile Sim	1,864	-\$31 billion	-\$2.2 million	-\$1,800	100%	100%	100%	100%	100%
74	Immersive Sim	6,902	-\$31 billion	-\$700,000	-\$1,400	100%	100%	100%	100%	100%
75	Massively Multiplayer	3,249	-\$3 billion	-\$1.3 million	-\$0	100%	100%	100%	100%	100%
76	Horror	5,248	-\$3 billion	-\$710,000	-\$170	100%	100%	100%	100%	100%
77	Colorful	26,824	-\$3 billion	-\$160,000	-\$480	100%	100%	100%	100%	100%
78	Team-Based	2,591	-\$2.8 billion	-\$1.8 million	-\$380	100%	100%	100%	100%	100%
79	Multiple Endings	10,268	-\$2.8 billion	-\$410,000	-\$1,600	100%	100%	100%	100%	100%

Figure 24: Steam chart tags (Gamestats, 2025).

4.2. Improvements

As a team, we have little to improve on; however, I think we could have worked more on gameplay, as while the gameplay we have is acceptable for a vertical slice, it lacks the depth we initially planned in our Treatment. We had to scrap our tertiary loop altogether because the team needed to focus on other aspects of the development. This means our gameplay is only fit for limited playtime, as there's no greater reason to return to it. With better planning and team collaboration, we could have had this as well.

Another point of contention would have been researching more about the tools used in the project. The chaos vehicle system was unsuitable for our project aim and was only realised a couple of weeks in, which was quick enough to resolve. Figuring this out even earlier would have been better, as our game's ability to change stats was a core design pillar.

We also faced challenges with two team members not contributing, and had to adjust the project accordingly. Overall, this was fine for our project, but it was a point of contention and stress for the team. We resolved this as a team in our SCRUM meetings when deciding what needed to be scrapped or distributed from the tasks. This resulted in some extension tasks lacking inclusion in our final product, but they were handled.

Individually, I would suggest the opposite. I worked on many aspects and gave myself a lot to do to carry the weight of the project as required. This led to a small amount of burnout, which could have been avoided with greater

consideration of what I involved myself in. Even if it worked out in this module, it would not have been sustainable for a longer project or industry.

4.3. Ethical/Legal

The most obvious legal concern is that our game was directly inspired by the existing IP Borderlands, and would mean it could not be published commercially, as we don't have the rights to it. We could work within the fair use framework for the purpose of educational use (Intellectual Property Office, 2014). As required for our project, though we did not use copyrighted works from Borderlands directly, any external assets, such as textures and tools, were clearly labelled in our individual asset list, and the team's external reference was submitted alongside our project.

All referenced assets were used within their licenses, most often CC0, meaning the initial copyright holder forfeits their copyright protections and allows use without attribution (Creative Commons, 2014). Tutors were made aware of external assets, especially paid ones, and were permitted to use them in our project.

4.4. Learning

I came into the project intending to develop my technical art skillset, and this project was the first time I had actively worked with artist requests and broken down requests into their parts to figure out what an artist wishes to achieve and what a tool needs to do.

I also wanted to develop my project development skills and successfully do so by keeping the project in scope throughout development and consistently updating the team as to changes.

I think next time, I would like to be more focused on my initial technical artist role, as I couldn't push myself in this area as much as I would have liked too by taking on two additional roles even if I did enjoy them and the work I did for them it was a lot of extra stress.

I'd like to develop more external tools for Unreal via scripts developed in Python, as I did not get the time needed to do so in this project, and the artist did not require much inside Unreal for tools as they will typically request most if not all artistic control over placement and appearance of assets, most technical tools were not very complex, as shown in Figure 25.

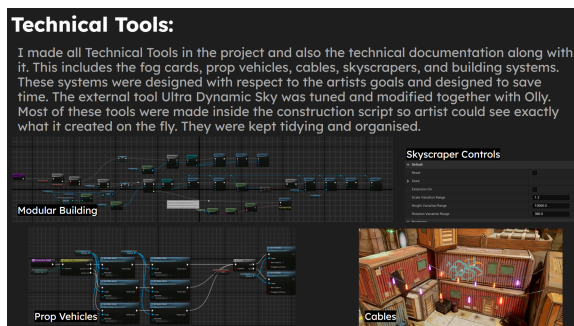


Figure 25: Individual contribution list - Technical Tools.

A good example of this would have been a style guide tool that took the artist's final assets, automatically recognised what textures were given, and assigned them the proper naming conventions, saving both the artist and me time on project organisation. Pipeline tools are a key area I wish to explore further next time and develop in my following modules.

4.5. Conclusion

I conclude that the project has been an excellent way for me to practice industry-standard techniques and strategies for project management, and I have been able to lead both myself and my team in the project effectively. We have produced a great vertical slice, as showcased in Figure 24, and I am very proud of my team and personal contribution to the project. I have learned a lot about how to conduct myself and the quality of assets I produce. As well as pushing myself out of my comfort zone to learn new techniques and set myself more advanced goals.



Figure 26: Beauty shot 1.

5. Bibliography

5.1. Reference List

Furious Production LTD (2021). *Arcade Vehicle System* | Fab. [online] Fab.com. Available at: <https://www.fab.com/listings/d27622ff-91af-4e87-a092-b3e4cd5012bb> [Accessed 23 May 2025].

Ashby, S. (2024). *How to Write a Game Design Document - with Examples and Template* - GitBook Blog. [online] www.gitbook.com. Available at: <https://www.gitbook.com/blog/how-to-write-a-game-design-document>.

Creative Commons (2014). CC0. [online] Creative Commons. Available at: <https://creativecommons.org/public-domain/cc0/>.

Doknic, A. (2024). *What Is Buffer Time & How to Find Your Sweet Spot* | Memtime. [online] Memtime. Available at: <https://www.memtime.com/blog/buffer-time>.

Duffy, J. and Vatu, G. (2024). *Trello Review*. [online] PCMAG. Available at: <https://www.pcmag.com/reviews/trello>.

Epic Games (2025). *Chaos Modular Vehicles* | Unreal Engine 5.5 Documentation | Epic Developer Community. [online] Epic Games Developer. Available at: <https://dev.epicgames.com/documentation/en-us/unreal-engine/chaos-modular-vehicles> [Accessed 23 May 2025].

Gamestats (2025). *Top Steam Tags by Sales and Revenue as of 2023* - Games-Stats.com. [online] games-stats.com. Available at: <https://games-stats.com/steam/tags/>.

Gearbox Software (2020). *Borderlands 3 on Steam*. [online] store.steampowered.com. Available at: https://store.steampowered.com/app/397540/Borderlands_3/.

Intellectual Property Office (2014). *Exceptions to copyright: Education and Teaching*. [online] Available at: https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/375951/Education_and_Teaching.pdf.

Polycount (2025). *Art Bible*. [online] Polycount.com. Available at: http://wiki.polycount.com/wiki/Art_Bible [Accessed 23 May 2025].

Rouse, M. (2020). *What Is a Minimum Viable Product (MVP)? - Definition from Techopedia*. [online] Techopedia.com. Available at: <https://www.techopedia.com/definition/27809/minimum-viable-product-mvp>.