

Computer Games Development

Project Thesis

Year IV

**Winter Wonder – VR Multiplayer Survival**

By

**Filip Rusiecki** [C00250721]

Submitted to the Department of Computing  
in partial fulfilment of the requirements for the degree of  
Bachelor of Science in Games Development  
at the  
South East Technological University  
May 2023  
© South East Technological University 2023. All rights reserved.

Author . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .  
 Department of Computing  
 May 1, 2023  
Certified by. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .  
 Lei Shi  
 Assistant Professor  
 Thesis Supervisor

[Declaration form to be attached]

**Contents**

[Acknowledgements 2](#_Toc54714373)

[Project Abstract 2](#_Toc54714374)

[Project Introduction and/or Research Question 2](#_Toc54714375)

[Literature Review 2](#_Toc54714376)

[Evaluation and Discussion 3](#_Toc54714377)

[Conclusions 3](#_Toc54714378)

[References 4](#_Toc54714379)

[Appendices 4](#_Toc54714380)

# 

# Acknowledgements

I would like to thank the following people who assisted in completing this project including:

-Ben Williamson (Art Student 4th year) for creating 3D Models for my game.

# Project Abstract

The VR game genre became popular in recent years, slowly more and more people try to get their hands on VR technologies this allows users to plunge into virtual reality environment. The emergence of new and modern VR headsets such as HTC VIVE, Oculus Rift and many more stimulated the rapid VR development in the game industry in recent years.

On the other hand, Survival game genre has been on the market for a long while yet always brining something new to excite people and making the companies have competition on which survival game is better yet always surprising players by brining something different to the market. Some survival games after realising they end up doing a version of the game for a VR which always attracts new players. With my game being made in VR I want to bring my own game to a great standard that will have some new features that will make my game unique.

I chose to create this game in a game Engine that is very known and it is Unity game Engine, I will try to create a very optimised game that will run smoothly on any VR headset and I was also wanting to bring something new to that game genre.

Thesis Supervisor: Lei Shi  
Title: Assistant Professor

# Project Introduction and/or Research Question

The Project I have decided to pick and do is a VR Multiplayer Survival game.

**Why VR?** I have never worked with VR or never seen how that process is made. This project also was chosen because I wanted to learn about VR games as in spare time, I’m trying to develop games/ modifications for different games and releasing them online.

**Why Survival game for the first game in VR?** Having learning about game development I wanted to do something that would challenge me and at the same time educate me and something that would expand my knowledge.

The game will present the player with a cold environment in which they will have to survive and beat few goals of the game to complete it. In the game the user will face different puzzles and different Advanced AI that will try to stop the player from completing the goals. The players main goal is to fix a crashed plane that they will use to escape the dangerous area in which they are trapped in.

**Why multiplayer?** I have never coded multiplayer game in unity engine. While playing any game almost definitely every game is better when you play it multiplayer. Its great to interact with other players and friends in a virtual environment. It is going to be challenging to make as VR and Multiplayer are hard to implement into games but I believe the I will manage to get the two systems working with each other.

**How will this project challenge me?** This project will challenge me in several different ways. First one is that I never worked with VR technology. This will require me to research into the topic on how to create a project file from scratch to suit a specific VR set (HTC VIVE), Input system will be difficult and controlling of any object by the player.

Another challenge that I will face is creating advanced options and controls for the player and also AI that player will be facing against. The AI will need different edge cases not to get stuck or bugged out to provide the player with smooth gameplay.

A big challenge that I will face is the graphics of the game, as many VR games if not done correctly they can make the user feel motion sickness so I have to make sure that the graphics of the game or camera movement will not cause it. Also while creating a game I need to remember that the game is created for VR, that cant be overloaded with very high graphics in order to run smoothly without interruptions.

Another part of the project which I will find challenging to do is multiplayer as I also never really looked into making games for multiplayer using unity engine.

Implementing Unity Mirror Multiplayer can be challenging for game developers, especially for those who are new to multiplayer game development. The technology requires a solid understanding of networking, synchronization, and game optimization, which can be difficult to master. Game developers need to ensure that the game is optimized for multiplayer, and that the game logic is properly synchronized across all players. They also need to consider the potential for lag or network interruptions, which can affect the gameplay experience. However, with the right knowledge and expertise, implementing Unity Mirror Multiplayer can provide a highly rewarding multiplayer gaming experience.

# Literature Review

Replace this text with an appropriate Literature Review.

The literature review places your research in context. You aren’t the first person to investigate or research a particular topic. Present a short literature review with the following goals:

* Give the reader a good overview of the key concepts;
* Describe the most relevant work (in your own words) that other people have done in this area;
* Use proper academic writing with references.
* Show how the existing work influenced your project.

# Evaluation and Discussion

Replace this text with Results and Discussion.

Describe the results using diagrams such as graphs etc. as appropriate, and discuss what the results mean.

Example: Results indicate that once the threshold gets over a certain point it significantly reduces player performance and player experience

**Project Milestones**

**Reaching the first milestone** of a project is always an exciting achievement. In this particular project, the first milestone was accomplished after successfully adding Steam VR to the project. This was a major milestone because it opened up a whole new world of possibilities for the project, allowing for the creation of an immersive and interactive virtual reality experience. The process of integrating Steam VR into the project was not an easy one. It required a significant amount of research and experimentation to ensure that the player was able to move around the virtual environment without any issues. One of the primary concerns was making sure that the player did not fall through the ground, which required careful attention to the physics engine and the creation of a new player controller.

After the initial setup was completed, the team was able to create a simple test area to ensure that everything was working as intended. This involved testing the physics engine to ensure that the player could move around the environment without clipping through objects or experiencing any strange behaviour. This was an important step in the development process, as it allowed the team to identify any potential issues early on and make adjustments as necessary. In order not to clip through the objects It was decided to make a script for the VR camera so that the player collider will follow the players camera which is players head in our situation. Overall, reaching the first milestone of the project was a significant accomplishment that required a lot of hard work and dedication. It laid the foundation for the rest of the project and set the team up for success in creating an engaging and immersive virtual reality experience.

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated with medium confidence

**Reaching the second milestone** was a crucial step in the development of the game. After successfully integrating Steam VR into the project, it was time to create a more immersive gameplay experience for the player by allowing them to shoot with guns in the game.

In order to achieve this, I created a customizable gun script that would allow me to create any weapon I desired. This script saved me a lot of time as it could be attached to any object, making it a weapon with any desired properties. It was a versatile tool that could be used in various ways and added to different weapons to make them unique.

However, creating the gun script came with its own set of challenges, particularly in regards to VR input. As the game was being developed for VR, it was essential that the guns were controllable with VR input. I had to consider how the gun would be held and fired in VR, and how the controls would feel to the player.

To overcome these challenges, I had to experiment with different input mappings and configurations. I had to adjust the settings until I found the perfect balance between intuitive controls and realistic gameplay. It was a trial and error process that required a lot of patience and perseverance, but ultimately, it paid off.

With the customizable gun script and the VR input controls in place, the player could now shoot in the game, adding a new level of interactivity and excitement to the gameplay. It was a significant achievement that brought the game one step closer to completion.

A screenshot of a computer program

Description automatically generated with low confidenceA screenshot of a video game

Description automatically generated

A screen shot of a computer program

Description automatically generated with medium confidence

A screen shot of a computer program

Description automatically generated with medium confidence

**Reaching the third milestone** of the project involved designing and creating the game environment, which was an enjoyable experience for the development process. The decision was made to create a large-scale world for players to explore, with a focus on creating an immersive experience. However, I had to be cautious about the performance impact of rendering large amounts of foliage in a VR headset, and thus ensured that all the foliage had a low level of detail when viewed from a distance.

To optimize the game's performance, I also created the project with low poly models, which not only made the game run more smoothly, but also added to its overall aesthetic appeal. The process of creating the world was challenging yet rewarding, as it allowed the me to experiment with various design elements and explore different artistic styles.

One of the most exciting aspects of creating the world was the opportunity to collaborate with a 4th year art student who provided some of the models. This collaboration allowed the me to access a wider range of assets and to interact with someone who had a different perspective and approach to art and game design. This experience helped the developer to learn and grow as a game developer.

In summary, the process of creating and designing the game world was a rewarding experience for the me as a developer. It provided an opportunity to experiment with different techniques and styles while also learning how to optimize performance for VR. The end result was a beautiful and immersive world that players could explore and enjoy.

**Reaching the fourth milestone** of the project was a critical moment for me as it marked a significant leap forward in the game's design and functionality. The decision was made to create an interactable plane that players could control and fly, adding a new level of excitement and immersion to the game.

To ensure the plane was easy to control and enjoyable to fly, It was put a lot of thought into the details. For example, they designed the plane to have a turning speed that wasn't too fast, as this would make the game difficult to control and potentially frustrating for players. Additionally, it was made sure that the plane's movements weren't too extreme - for example, the plane wouldn't be able to spin vertically upwards or downwards fully. This was done to prevent players from feeling disorientated or getting motion sickness from playing the game.

Creating a plane that was both fun and comfortable to fly required a lot of experimentation and trial-and-error. I also had to think creatively and come up with innovative solutions to ensure that players wouldn't get nauseous or feel sick when playing the game. Also had to consider the various factors that could contribute to motion sickness, such as the plane's speed, altitude, and orientation.

One of the most exciting aspects of designing the plane was the opportunity to experiment with new technologies and control schemes. For example, the implementation of VR touchpad system that allowed players to control the plane using specific positions on the touchpad. This was a new experience for the creation process, and it took a lot of time mapping different positions on the touchpad to different functions within the game. However, the effort paid off, and the VR touchpad system added a new level of immersion and excitement to the game.

Overall, reaching the fourth milestone of the project was a significant achievement for the development. The creation of an interactable plane added a new dimension to the game and required a lot of thought and experimentation to get right. However, by carefully considering the various factors that could contribute to motion sickness and implementing innovative control schemes, the team was able to create a plane that was both fun and comfortable to fly.

**Reaching the fifth milestone** of the project was a critical moment for the development as it marked a significant leap forward in the game's design and functionality. The decision was made to enable multiple players to connect to a single world, allowing them to play together in real-time. Players would be able to connect to a host using IP. While this feature added a new level of excitement and social interaction to the game, it also presented some significant challenges for the development. Another challenge was designing the game world in a way that would be interesting and engaging for multiple players. However, building this functionality was not without its challenges, and I had to invest a significant amount of time and effort into ensuring a smooth and secure multiplayer experience.

**Reaching the sixth milestone** of the project was a crucial moment for the development, as decided to implement a waiting lobby that would improve the overall user experience of the game. This feature enabled players to either choose to host a game or connect to a game using an IP address.

The waiting lobby provided a central hub where players could gather and interact with each other before the game started. Once players were connected to the waiting screen, they had to ready up before the game could start. This feature ensured that all players were prepared and ready to play, preventing any delays or disruptions during the game.

To make the waiting lobby even more user-friendly, the team implemented a "Ready" button that appeared next to each player's name. Players could click this button to indicate that they were ready to start the game. When all players had clicked the button, the game would start automatically.

To prevent any unfairness or confusion during the waiting period, I made sure that players could only ready up themselves and not other players. This ensured that each player had control over their own readiness status, and that no one could start the game prematurely.

The waiting lobby also provided an opportunity for players to interact with each other and build anticipation for the game. This social aspect of the game added an extra layer of enjoyment and excitement for players, as they could chat, make plans, and get to know each other before the game started.

Overall, the implementation of the waiting lobby was a significant achievement for the development. This feature improved the overall user experience of the game, ensuring that players could connect and prepare for the game without any delays or disruptions. By making the waiting lobby user-friendly and social, I was happy to add an extra layer of enjoyment to the game, making it more engaging and exciting for players.

**Seventh syncing objects**

Replace

this text with Project Milestones.

Key project milestone dates and measurement on schedule, was project schedule adhered to, effectively planned for delivery on-time or ahead of schedule if appropriate.

**Major Technical Achievements**

What are your major technical achievements?

**Project Review**

As I continued to work on the project, I encountered a plethora of obstacles that I had not anticipated when I first began. The complexities of working with VR and multiplayer technologies became more apparent, and I realized that I needed to dig deeper and learn more in order to succeed. It was a challenging journey, but it ultimately helped me to grow as a developer.

One of the biggest challenges that I faced during the project was related to the Fish-net Networking library that I had initially chosen to use. While it had been recommended to me as a popular choice among Unity developers, I found that it was quite complex and difficult to work with. It was particularly frustrating to encounter the issue with cloning a copy from GitHub that caused my entire project to break. I was disheartened and felt like I was taking steps backward rather than moving forward.

(This error has fixes to it but no tutorial or fix was working in my situation) A screenshot of a computer

Description automatically generated

However, I did not let this setback defeat me. Instead, I decided to switch to the Mirror Networking library, which was a more user-friendly and straightforward option. This change required significant modifications to my code, but I felt it was worth it. I was pleased to see that the project was no longer crashing, and I was able to proceed with implementing multiplayer functionality.

Another significant challenge that I faced was related to testing the game in VR with parallel sync. It was a complex and time-consuming process to ensure that everything worked correctly in both VR and multiplayer. It was particularly frustrating to discover that all the players were trying to use the same VR headset, which made testing impossible. However, I persevered and found a workaround that allowed me to continue testing despite this difficulty.

Despite the challenges, I am grateful for the experience that this project gave me. I learned a great deal about working with VR and multiplayer technologies, and I gained valuable insights into network synchronization and latency. I am confident that this knowledge will be useful in future projects, and I am excited to continue exploring these fields further.

During my project, I encountered another significant challenge related to the integration of Steam VR and Mirror networking libraries. I had originally planned to use both of these libraries to enhance the VR and multiplayer aspects of the game. However, as I began to implement both libraries, I quickly realized that they were clashing and not working together as I had hoped. The main issue was that both libraries had their own references to the keyword "Player" in their respective scripts. As a result, when I tried to run the project, there were conflicts between the two libraries as they were trying to access the same properties simultaneously. This made it impossible to run the project without first addressing this issue. To solve the problem, I had to make significant changes to the VR properties in the Steam VR scripts. This was a daunting task as I had to go through each script and modify it to ensure that there were no conflicts with the Mirror networking library. It was a time-consuming process, but it was necessary to get both libraries working together and to ensure that the project was functional. In retrospect, this challenge taught me the importance of careful planning and consideration when integrating multiple libraries and frameworks. It highlighted the importance of understanding the intricacies of each library and the potential conflicts that may arise when using them together. While it was a difficult experience, it was also a valuable lesson that will help me to approach future projects with more caution and forethought. (Check appendices for explanation and snippets)

Overall, I am proud of the work that I accomplished and the progress that I made. It was a journey filled with ups and downs, but I emerged from it with a greater understanding of these technologies and a renewed sense of determination.

What went right? What went wrong? What (if anything) is still outstanding/missing (i.e., still left to do)? If starting again, how would you approach this project differently? What advice would you have for someone attempting a similar project in the future? Were your technology choices the right or wrong ones? If you chose the wrong technology, provide justifications for why you think this. What were the implications of your technology choices?

# Conclusions

In conclusion, Unity Mirror Multiplayer technology has revolutionized the multiplayer gaming experience in VR, providing game developers with the tools they need to create immersive and engaging multiplayer games. However, implementing this technology is not without its challenges, as game developers must navigate the complexities of networking, synchronization, and game optimization.

One of the biggest challenges of developing multiplayer games in VR is the need to ensure seamless gameplay. Even minor delays or interruptions in network connectivity can lead to a poor user experience, which can be frustrating for players. As a result, game developers must work tirelessly to account for potential lag or network interruptions and ensure that their games are optimized for performance.

Additionally, creating multiplayer games in VR requires a different set of skills and expertise than developing single-player games. Game developers must familiarize themselves with new VR systems, tools, and technologies and must work to optimize their games for the unique requirements of the VR environment.

Despite these challenges, Unity Mirror Multiplayer technology has enabled game developers like myself to create engaging and immersive multiplayer games that provide players with a highly rewarding experience. By leveraging the power of this technology, we can create games that allow players to connect with each other in real-time and explore new worlds together.

Of course, developing multiplayer games in VR is not without its setbacks. During the development process, I encountered several challenges, including issues with networking and game optimization. However, with perseverance and determination, I was able to overcome these obstacles and deliver a game that I am proud of.

In retrospect, I believe that my lack of prior knowledge of VR or multiplayer technologies made this project even more challenging. However, with the help of online resources and forums, I was able to learn the necessary skills and knowledge to bring this project to fruition.

Overall, I believe that Unity Mirror Multiplayer technology has opened up a world of possibilities for game developers, enabling us to create innovative and immersive multiplayer games in VR. Despite the challenges, I believe that the rewards are well worth the effort, as multiplayer games in VR provide players with an unparalleled gaming experience that they will never forget.

**Future Work**

If a student were to undertake a project in the area of Unity VR multiplayer, there are several interesting next steps that they could consider exploring:

1. Allowing for players to play with any input they want such as Controller, Keyboard or VR. So that any players playing the game could have a choice between different inputs they want to play on so the game wouldn’t be limited to only play it with VR.

2. Proper AI integration: The integration of AI into Unity Mirror Multiplayer could provide an exciting opportunity for game developers to create highly engaging and challenging multiplayer experiences. The implementation of AI-controlled non-player characters could enhance the realism of the game and create new challenges for players. Even creating a bot that can do anything that player can.

3. Dynamic environments: The development of Unity Mirror Multiplayer games that feature dynamic environments, such as changing weather or destructible terrain, could provide an exciting and engaging multiplayer experience. This could involve the implementation of advanced physics systems, such as Unity's Physics System or NVIDIA's PhysX, to create more realistic and interactive environments. Procedural world generation could be included into this which would create a much better experience for the players as it would continuously create a random chunks of world for the player to explore. This feature would add to the replay ability to the game.

# References

# Appendices

Using the VR input system was a trail and error as things didn’t go as easy as plannedA screenshot of a computer

Description automatically generated with medium confidence

The VR scripts I had to change in order to run the project.

[PlayerVRRR.cs]

A screenshot of a computer program

Description automatically generated with medium confidence

[Hand.cs]

A screen shot of a computer program

Description automatically generated with medium confidence

A screenshot of a computer program

Description automatically generated with medium confidence

A screen shot of a computer

Description automatically generated with low confidence

A picture containing text, screenshot

Description automatically generated

[HandPhysics.cs]

A picture containing text, screenshot, font, software

Description automatically generated

[SnapTurn.cs]

A screen shot of a computer program

Description automatically generated with low confidenceA picture containing text, screenshot, font, software

Description automatically generatedA picture containing text, screenshot, software, font

Description automatically generated

[DebugUI.cs]

A screenshot of a computer program

Description automatically generated with medium confidence

[RenderModels.cs]

A screen shot of a computer code

Description automatically generated with medium confidence

[Throwable.cs]

A screenshot of a computer program

Description automatically generated with medium confidence

[ControllerButtonHints.cs]

A screen shot of a computer

Description automatically generated with low confidence

A picture containing screenshot, text

Description automatically generated

[Arrow.cs]

A picture containing screenshot, text

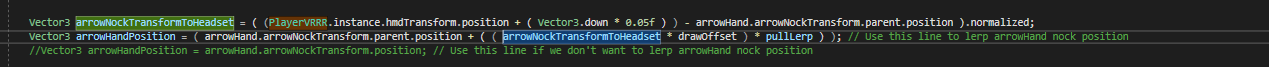
Description automatically generated

A picture containing text, screenshot

Description automatically generatedA picture containing text, screenshot, font

Description automatically generated

[Longbow.cs]



[JoeJeffGestures.cs]

A picture containing text, screenshot, software, multimedia software

Description automatically generated

[ControllerHintsExamples.cs]

A screen shot of a computer

Description automatically generated with medium confidence

[FlowerPlanted.cs]

A picture containing text, screenshot, font, software

Description automatically generated

[Planting.cs]

A picture containing text, screenshot, font

Description automatically generated

[SkeletonUIOptions.cs]

A screen shot of a computer program

Description automatically generated with medium confidence

[SnapTurn.cs]

A screen shot of a computer program

Description automatically generated with medium confidence

[Teleport.cs]

A screen shot of a computer

Description automatically generated with low confidenceA picture containing text, screenshot, font

Description automatically generated

[TeleportArc.cs]

A screen shot of a computer program

Description automatically generated with low confidence

[TeleportPoint.cs]



A screen shot of a computer

Description automatically generated with low confidence

Replace this text with Appendices.

This might include ethics application and other relevant material e.g. copy of any questionnaires used.