**DJ Simulator**

***Individual Project Dissertation***

# Table of contents

[Table of contents 2](#_Toc196229546)

[Introduction 3](#_Toc196229547)

[Aims and objectives 3](#_Toc196229548)

[Personal developmental goals 3](#_Toc196229549)

[Summary of upcoming analysis 3](#_Toc196229550)

[State of the Art Review (Literature Review) 4](#_Toc196229551)

[Tools and Technologies 7](#_Toc196229552)

[Analysis, requirements specification and development methodology 7](#_Toc196229553)

[Game Design: Conceptual 8](#_Toc196229554)

[Game Design: Technical 8](#_Toc196229555)

[Implementation 8](#_Toc196229556)

[Testing and Evaluation 8](#_Toc196229557)

[Legal, Social, Security and Ethical Issues 8](#_Toc196229558)

[Critical Review and Conclusion 8](#_Toc196229559)

[Technical Appendices 8](#_Toc196229560)

[References 8](#_Toc196229561)

# Introduction

DJ Simulator is a strategic simulation game made in Unity 2023 LTS. The game has been developed mainly on macOS, however anticipated platform deployment is both Windows and macOS. The game is set in a low-budget nightclub, where the player plays as the DJ who must entertain the crowd with good music choices.

# Aims and objectives

The primary aim of the game is to perform excellent DJ sets that score plenty of points within a 5-minute time limit. In order to gain points, the player must ensure that they are selecting tracks that are of a genre that is the most popular within the crowd.

The main game objective is to score the most points within the 5-minute time limit. In order to score the most points, the player must first identify the most popular genre within the crowd. Once identified, the player must play a track of that genre in order to start accumulating points. The crowd will react accordingly. The player must continue to monitor the crowd, due to the fact that the crowd will change in numbers. As the popular genre changes within the crowd, the player must choose new music that fits accordingly. Once the 5-minute timer elapses, the game will notify the player of their score. Not selecting appropriate genres, or not changing the music based on the crowd’s changes in favourite genres will not award many points.

# Personal developmental goals

When embarking on this project, I had 3 main goals in mind:

* To create a convincing simulation of tasks & experiences of a typical beginner DJ
* To realistically simulate crowd behaviours that can be observed within nightclubs
* To simulate a low-budget nightclub environment that feels ‘alive’ and responsive to the player’s actions.

Being a DJ myself, I intended to create a game that is able to simulate and showcase the typical workflow and tasks that a beginner DJ in a low-budget venue would experience. This is an experience that many DJs can relate to, and in order to deliver on a genuine representation of this experience, I set the 3 main goals above as my developmental goals since I believe that achieving these well will result in a realistic depiction of the intended experience.

# Summary of upcoming analysis

This report will delve into an analysis of how this project was developed and formed to achieve the personal developmental goals outlined above. It is important to analyse both the infancy and late development stages of the project thoroughly.

The early planning stages will be covered in the form of a Literature Review, where sources of inspiration will be explored. The information collected from this research will then be thoroughly analysed and translated into a development methodology and requirements specification, where a clear developmental plan can be established.

At this point, clear requirements and goals will be apparent, therefore conceptual and technical game design can be discussed. A deep insight into the reasoning of how the game is designed and why those design choices have been made is crucial, due to the many intentional mechanics that are implemented in order to create an immersive simulation. Many of those design choices are implemented in unique and abstract way, therefore a detailed analysis of the technical implementation is also necessary.

Since this game contains a diverse simulated system, thorough testing is necessary. The report will detail how this has been achieved to ensure that the game’s simulation systems are robust, and as many simulation cases as possible have been tested. Legal and moral issues will also be considered here.

Finally, the report will conclude with a critical evaluation and conclusion of the information and research conducted during the development of this project. This will be followed by technical appendices and references.

# State of the Art Review (Literature Review)

DJ Simulator aims to depict the humble beginnings of a new DJ. Most DJs, when starting their career, will perform in low-budget, run-down venues [7], therefore it is important that this project depicts the gritty atmosphere well. There are certain industry standards within the DJ world, such as the hardware, which often tends to be Pioneer CDJs or Technics SL-1200s [8]. It is important that the game represents those expected industry standards faithfully in order not to alienate the player – which is highly likely to be (or aspiring to be) a DJ.



*Figure 1 – A typical Pioneer DJ setup.*

Not many gaming products exist within the niche of the DJing world. There does not seem to be a strong link between video games and DJing. Despite that, several gaming products and generic niche conventions have been found and analysed. While no singular product has satisfied the entirety of the developmental objectives of this project, the following products each contain features that have proven insightful for the development of this project. Identifying those, and adapting them for the needs and goals of this project will help create a cohesive DJing experience. Beat Saber VR has been selected as an example of reactive environment design, Vinyl Reality has been selected as an example of realistic DJ hardware simulation, and finally GTA Online has been selected as an example of a realistic crowd simulation system.

***Beat Saber VR***



*Figure 2 – In-game screen shot of Beat Saber VR.*

Beat Saber VR is a music rhythm game where the player must break blocks with their sword in beat with the music. This game features heavily adaptable environments, which change based on the player’s actions and song choice. Visual effects may appear or morph when the player triggers an event system [1]. This means that the game adapts visually based on what they player is doing and how well the player is doing.

This level of visual feedback is a great device for signalling important gameplay messages to the player, and creating an immersive, engaging and ‘breathing’ in-game world [2]. Based on this observation, it has been decided that in aid of improving environmental immersion and player feedback, audio reactive lighting will be implemented into the project.

Audio reactive lighting is a very typical element of most conventional nightclubs [3], therefore including it in the game is a great way of combining visual feedback, environmental immersion and generic convention within the DJ niche; most DJs would have had experience performing in a venue with reactive lighting.

***Vinyl Reality***



*Figure 3 - In-game screen shot of Vinyl Reality (via Steam Store page).*

Vinyl Reality is a game that deeply focuses on track selection, and the performance of said track. It is a deep emulation of a vinyl-based DJ system. Being developed by professional DJs [4], this product provides an excellent representation of typical tasks for a performing DJ. Most of the expected DJ system mechanics are thoroughly and realistically portrayed within this product.

The implementation of allowing the player to drag-and-drop music [5] on the turntable is an excellent way of giving the player control over the in-game world. This product is a strong source of inspiration for the simulation of the interactable in-game systems within the project. It allows for realistic interfacing and control over the in-game world, while retaining it within a format that is very synonymous within the DJ niche.

The turntable depicted within Vinyl Reality is the Technics SL-1200. It is a turntable that is a staple within the world of DJing, and has been around since the 1970’s, making it a crucial piece of hardware ad heritage within the DJ world [6]. More importantly, the selection of such widely recognised hardware means that most DJs can easily interact with the in-game depiction of the turntable, since it is highly likely that they have used the real-life counterpart. Using this renowned system reduces the risk of confusion or alienation of the player, by utilising a familiar layout and system.

***GTA Online – Nightclub Missions***



*Figure 4 – A crowd and a DJ inside the Music Locker nightclub in GTA Online.*

The nightclub missions form a relatively small element within Grand Theft Auto Online’s expansive and busy world. They are however a perfect example of excellent crowd simulation within a nightclub environment. There is a multitude of NPCs that have unique behavioural patterns [9], and enhance the mise-en-scene of the nightclub drastically.

While in GTA Online the crowd is mostly utilised for an aesthetic purpose, within DJ Simulator the crowd will provide visual aid for the player. The player should be able to identify popular music genres by looking at the crowd.

A realistic implementation of crowd simulation within the nightclub can greatly benefit the project by highly enhancing the environment (and its realism) and also as a means of providing visual feedback to the player on their performance and areas of improvement.

To summarise, based on this analysis, utilising the reactive environmental systems of Beat Saber, the realistic crowd simulation systems from GTA Online nightclubs, along with the detailed, recognisable and robust hardware simulation systems of Vinyl Reality, is highly beneficial for creating a lively and detailed simulation of a DJ setup in a nightclub. Combining these systems creates a synergy between functional, immersive, ‘breathing’ environments and recognisable and realistic DJ hardware. These sources of inspiration provide an insight into the ingredients required for a robust, practical and non-alienating simulation of a DJ's environment within a nightclub.

# Tools and Technologies

There are several essential features that the game engine of choice must provide for this project:

* High performance – large crowds and real-time visual effects can be computationally expensive. Therefore, a well optimised engine is needed that will not suffer from performance issues when simulating detailed scenes.
* Powerful scripting – there are many features that will need to be implemented that require complex logic to work. The engine must provide useful tools and libraries in order to make this process easier.
* Stability – since this project is not utilising any novel/evolving concepts (such as generative AI), an engine version with the greatest stability is of highest priority.
* Versatile animation system – in order to achieve realistic crowd simulation, an engine that can handle and calculate animations well is important.
* Environmental sensing – due to the large number of stimuli within the scene, the engine must be able to identify and act accordingly to player and environmental triggers.

Based on this list of requirements, two prospective engines have been considered: Unity 2023 LTS and Unreal Engine 5. Both of these engines satisfy the majority of these requirements much better than other game engines such as Godot or GameMaker, which either lack the necessary functionality, or are not as capable with it.

Unity is known for excellent performance, even with complex scenes. Therefore, scenes with large crowds should not cause performance issues for this engine. C# is a very powerful scripting language and is the one used by Unity. C#, along with MonoBehaviour [10] provide extremely powerful scripting opportunities. The 2023 LTS version of Unity provides all necessary features while being very stable. No preview (or more recent) version is needed here, and the project will benefit from using the very stable LTS branch. Unity’s Animator is very powerful and will allow for complex NPC simulations. Unity’s colliders and map constructing systems work very well with events and triggers.

On the other hand, Unreal Engine provides high fidelity graphics at ease [11], albeit with a large performance cost. The usage of C++, which is a very powerful language, along with the versatile and practical Blueprint system offers huge scripting capabilities. Unreal Engine offers no LTS solution, and crashes (especially on macOS) are very frequent. Unreal Engine provides a very powerful [12] suite of animation and simulation tools. They are very detailed, however overkill for this application. Unreal Engine’s environment sensing is excellent, especially when paired with the Blueprint system.

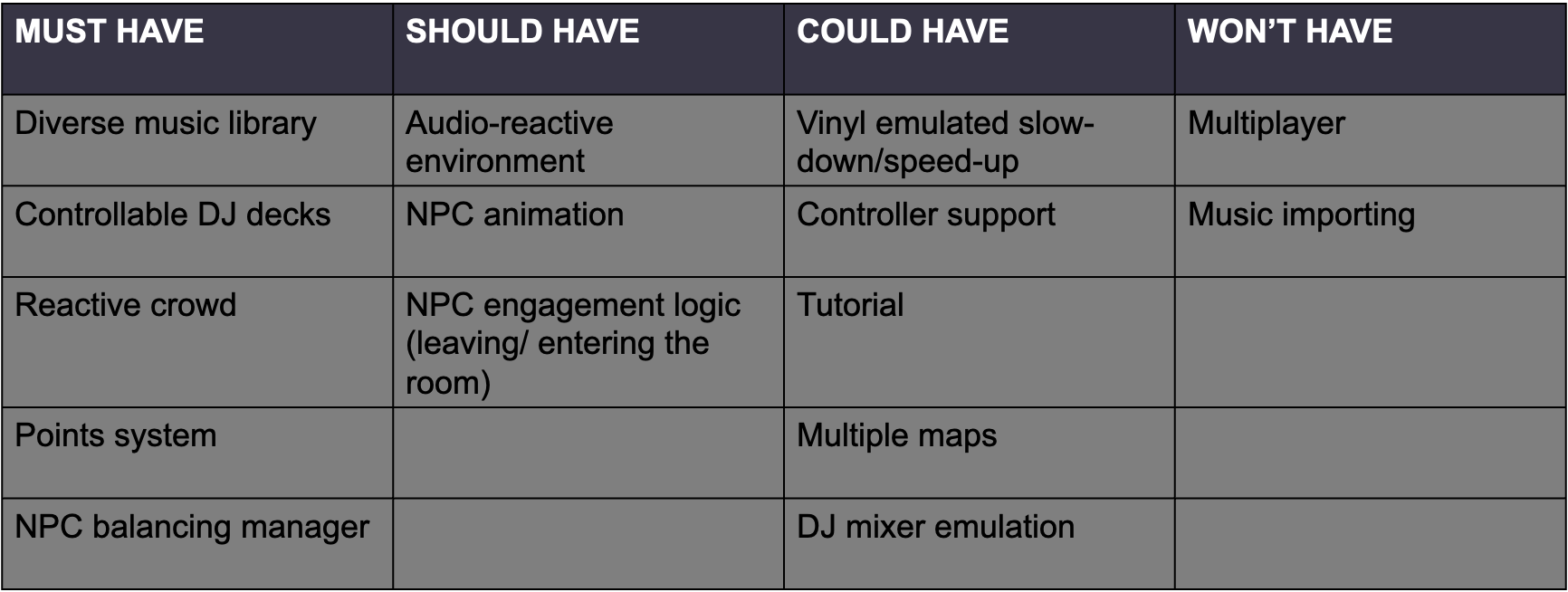
# Analysis, requirements specification and development methodology

While both engines provide excellent opportunities for developing a good DJ simulation game, the engine of choice is Unity 2023 LTS. A table weighing the pros and cons of each engine illustrates the greater advantages of Unity over Unreal Engine for this application.

|  |  |  |
| --- | --- | --- |
|  | **Unity 2023 LTS** | **Unreal Engine 5** |
| **Performance** | Due to lesser (unnecessary for this use case) graphical fidelity, this engine provides better performance | The high-fidelity graphics are very computationally heavy. |
| **Scripting** | C# and MonoBehaviour – provides all necessary functionality for this project. The scripts compile quickly on play. | C++ and Blueprints – provides plenty of additional functionality. However, it is not needed for this project. C# in-engine compilation is not straightforward. |
| **Stability** | LTS version offered – provides excellent stability and no unexpected updates. | No LTS version. Very unstable on macOS. |
| **Animation/simulation systems** | Robust suite of animation tools. Everything needed for a realistic crowd/NPC simulation. | Overwhelming suite of tools. Many of which are not needed for this application. |
| **Environmental sensing** | Great interaction between colliders, triggers and MonoBehaviour scripts. | Great implementation of Blueprints. |

Looking at the data in this table, it is easily observable that Unity 2023 LTS should be the engine of choice. It fulfils all requirements, without much extra bloat. Being an LTS release, it should not cause down-time or issues related to unexpected behaviours or updates.

Due to this project being a DJ Simulator game, audio analysis algorithms are also very important here. Unity offers the GetSpectrumData function, which provides the block of audio frequencies (spectrum data) of the AudioSource that is currently playing [13]. This is essential for the reactive crowd and lighting that is envisioned within this game.



The MoSCoW system is a great way to quickly organise priorities. Based on the analysis of this project, it is important to focus on core gameplay and simulation features. The game must have a purpose, and a means of executing that purpose.

# Game Design: Conceptual

# Game Design: Technical

# 

# Implementation

# 

# Testing and Evaluation

# 

# Legal, Social, Security and Ethical Issues

# 

# Critical Review and Conclusion

# 

# Technical Appendices

# References

[1] Ler digital studio (2025). Beat Saber - VR rhythm game. [online] Beatsaber.com. Available at: https://beatsaber.com/documentation/environment-specific-lighting-attributes-and-explanations/index.html [Accessed 22 Apr. 2025].

[2] Sunil Khobragade (2024). Bringing Stories to Life in Casual Gaming Experiences [online] Linkedin.com. Available at: https://www.linkedin.com/pulse/visual-storytelling-casual-game-development-sunil-khobragade-xjvhf/ [Accessed 22 Apr. 2025].

[3] Venue Lights. (2024). Nightclub Lighting | Venue Lights. [online] Available at: https://venuelights.co.uk/lights/venue-lighting/nightclub-lighting/ [Accessed 22 Apr. 2025].

[4] Vinyl Reality. (2021). Vinyl Reality - DJ in VR. [online] Available at: https://www.vinyl-reality.com/ [Accessed 22 Apr. 2025].

[5] Vinyl-reality.com. (2019). audio\_analysis [Vinyl Reality Wiki]. [online] Available at: https://vinyl-reality.com/wiki/doku.php?id=audio\_analysis [Accessed 22 Apr. 2025].

[6] Technics (n.d.). SL-1200 ‒ Heritage - Technics. [online] www.technics.com. Available at: <https://www.technics.com/global/home/sl1200/heritage.html>.

[7] www.djgym.co.uk. (n.d.). How Do DJs Get Their First DJ Gig: A Guide For New DJs. [online] Available at: https://www.djgym.co.uk/post/how-do-djs-get-their-first-dj-gig-a-guide-for-new-djs.

[8] Simoes, R. (2024). A Complete Guide To The DJ Club Standard Setup For The Industry In 2024 - If You’re In Search Of DJ Gigs, Here’s What You Should Know - Learningtodj.com. [online] Learning to DJ. Available at: <https://learningtodj.com/blog/the-club-standard-dj-setup-in-2024/>.

[9] Fandom.com. (2025). GTA Online Nightclubs. [online] Available at: https://gta.fandom.com/wiki/Nightclubs [Accessed 22 Apr. 2025].

[10] Unity Technologies (2024). Unity - Scripting API: MonoBehaviour. [online] Unity3d.com. Available at: <https://docs.unity3d.com/6000.0/Documentation/ScriptReference/MonoBehaviour.html>.

[11] Epic Games Developer. (2025). Designing Visuals, Rendering, and Graphics | Unreal Engine 4.27 Documentation | Epic Developer Community. [online] Available at: <https://dev.epicgames.com/documentation/en-us/unreal-engine/designing-visuals-rendering-and-graphics?application_version=4.27>.

[12] Unreal Engine. (2024). Animation. [online] Available at: <https://www.unrealengine.com/en-US/uses/animation>.

[13] Unity Technologies (2025). Unity - Scripting API: AudioSource.GetSpectrumData. [online] Unity3d.com. Available at: https://docs.unity3d.com/6000.0/Documentation/ScriptReference/AudioSource.GetSpectrumData.html [Accessed 22 Apr. 2025].

‌