

The role of .NET in the development of computer games: an analysis of the capabilities of Unity

Mykola Smorgun

*Master of Finance and Credit, CEO of Lumighost, Inc,
<https://orcid.org/0009-0008-8962-6611>*

Abstract: The .NET framework is used to create an environment for computer game development industrialization. The specification and architecture of a video game product line are described and developed using software factory assets such as a graphic designer focused on a business-specific language, semantics validators, and code generators. The goal of this paper is to examine the possibilities of the Unity game design engine for people working in the IT business, specifically those employing the .NET platform. This study used qualitative research methods. Unity has been determined to be an innovative and dynamic game design platform, and it is presently in use. Unity inspires multiple industries that could have a positive impact on professional progress, career opportunities, and so forth. Unity has multiple benefits; it is a basic and clear environment in which to learn game creation and a powerful tool that professionals like.

Keywords: benefits, challenges, game development, .NET platform, Unity

1. Introduction

Game design has been an ongoing academic issue. The computer gaming business has risen by enormous amounts, to become a major application development sector with annual revenue in the billions of millions. The gaming sector has also been a major driving force in the development of cutting-edge modern systems including multi-core, highly threaded processors, extremely high graphics processors, enhanced audio devices, and amazing HCI (human-computer interaction) devices including virtual reality (VR) helmets and brain transmitter caps. Computer games are similar to books, movies, and museums [1].

Unity, developed by Unity Technologies, is a cross-platform game engine that was launched as a limited-edition computer game in 2005 June at World Wide Apple Inc.'s Conference (Mac OS X). The engines were expanded to serve over 20 platforms in 2018. This game platform is used to construct augmented reality (AR), VR, and both 2D and 3D video games, in addition to simulators and other applications. In the twenty-first century, this game engine has been used by firms other than video games, such as film, architecture, cars, construction, and engineering [2]. The Unity3D Game Engine is a complete development platform used to create dynamic features such as computer games, architecture modeling, and real-time 3-dimensional animations. Its editor is available for both versions of Windows and Mac OS X. Despite the fact that it only operates on multiple systems, it has the capacity to build programs for a variety of platforms, including Windows, Windows Mobile, Macintosh, iOS, UNIX, Android, Web Player, and others. All that is required to build the application on the software provided is the software development kit (SDK) [3].

A. Research Aim

To raise knowledge and provide information on Unity for the advantage of individuals working in the IT industry specifically for the .NET platform. To complete the goal, three objectives and questions are constructed, as shown below.

B. Research Objective

- To investigate the advantages of the Unity game design engine.
- Examine the potential issues that could occur in Unity.
- Outline the optimal Unity ways for boosting the capabilities of Unity developers using the .NET framework.

C. Research Paper Questions

- What are the advantages of the Unity game design engine?
- To what level have Unity's issues disturbed those employed in the game development sector on the .Net platform?
- What are the best techniques for developing Unity abilities for Unity developers utilizing the .Net platform?

2. Literature review

This chapter of the study includes a review of the literature. Fig. 1 depicts the documentation of significant works as well as the evaluation of the materials acquired. Also, Fig. 1 was generated utilizing the PRISMA [4] approach, which is detailed below. As a consequence, the literature study has come to be considered one of the most essential parts of a research paper because it facilitates both the author and the audience in learning and studying several features that are significant to the current topic and have been done before.

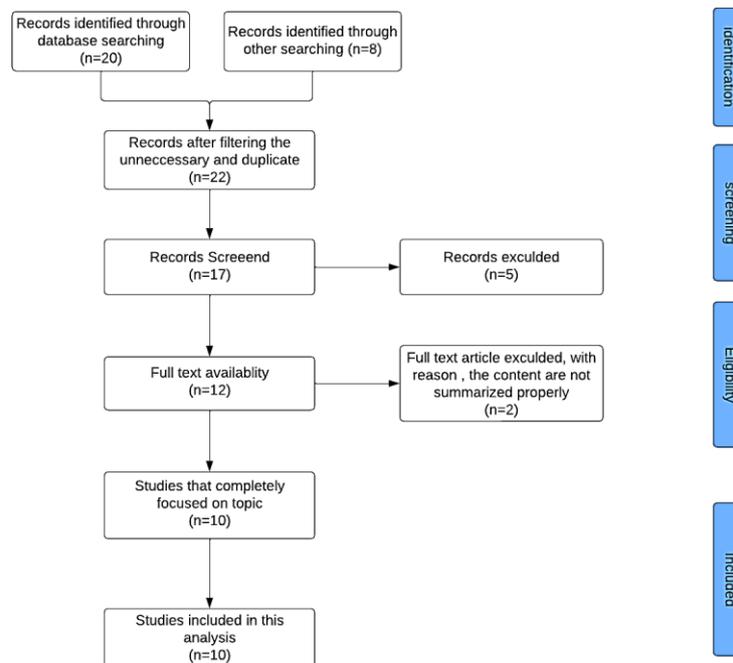


Figure 1 PRIMA chart

Unity is a multi-platform game engine that may be used to develop both 2D and 3D computer games, in addition to graphics and non-game interactive simulations. Moreover, Unity is a significant free and accessible game engine that is currently popular among developers because of its affordability, adaptability, efficiency, and power consumption. The Unity Editor has a variety of features that enable rapid iteration and change during development cycles, such as intelligent displays and real play modes [5]. Unity is also accessible for Windows, Linux, and Mac, and it offers an artist-friendly toolkit for building immersive environments and gaming environments, as well as a complete developer tools package for generating gameplay and game intelligence [6]. Moreover, Unity supports both two-dimensional and three-dimensional development using functions and features that meet particular demands.

Similarly, Eriksen, Nielsen, and Pittelkow [7] stated that Unity offers a navigation system that enables you to create NPCs who navigate logically within the gaming universe. Navigational mesh, which is dynamically created out of scene geometry or, in certain cases, dynamical obstacles, has been used in the technology to alter the character's movement during runtime. Unity Prefabs, also referred to as predefined game components, offer operational adaptability and effectiveness, enabling confident work without the risk of time-consuming mistakes [8]. According to Hastawan, Nashiroh, Firdaus, and Rossa [9], the Unity-designed UI architecture enables the construction of the UI straightforwardly and understandably while spending less time. To enable high-performance and realistic gameplay, Unity employs the breakthrough DOTS-based NVIDIA PhysX and Physics technologies in Box2D. Developers may add capabilities to the Unity Editor to customize team operations. It also makes it easier to construct and customize extensions with many assets, plugins, and resources for faster project completion [10].

Unity is a platform that incorporates a variety of powerful components, including the PhysX physics engine, the Mechanim animation system, a separate landscape editor, and much more. It is also seamlessly integrated with the Mono-develop code editor, allowing any changes made in Mono-develop to be transparently built and loaded into the gameplay via Unity's C# or Javascript compilers. The compilation errors are displayed in the Unity terminal window [11].

JavaScript and C# are used to create the game engine. The scripting in Unity is built on Mono, an accessible .NET Framework implementation. MonoDevelop is an integrated development environment (IDE) for Unix, Macintosh X, and Windows. The following programming languages are supported: Boo, C, C++, C#, CIL, D, Oxygene, F#, Java, Python, Vala, and VB.NET. The scriptis employed to react to player input and to organize gaming events so that they occur at the proper moments. They may also be used to create graphical effects, change the structure and behavior of objects, and even create a custom AI system for computer games [11].

The Unity Asset Server is also included, which is a version control system for the developer's game components and scripts. It has a PostgreSQL database backend and an audio system built using the Frequency-modulated (FMOD) toolkit. There is also a terrain and vegetation engine [12].

A platform for creating and hosting client-side JavaScript applications. Although the Lively Kernel is not intended to be a game engine, it has gaming apps and implements several game engine characteristics. The current version includes features such as user chat and user identity (registration) [13].

The previously stated Facebook platform offers a diverse range of gaming apps. According to Di Loreto and Abdelkader, in the Facebook environment, playfulness is associated with the merging of personal and social features. The presence of pals on Facebook is a motivator for players to return to the program. Finally, it is the users that generate fun [14].

3. Methodology

In this work, a qualitative research approach was used to collect data from previously published studies such as peer-reviewed journals, papers, books, and other sources. The data was gathered using a secondary search method that consisted of studying and analyzing past research directed at Unity game production, namely publications and conference proceedings. To make this method more efficient, researchers imposed several constraints. For example, data is exclusively taken from credible databases like Google Scholar, Scopus IEEE Xplore, Academia, Research Gate, and ProQuest from 2019-2023 using keywords like Unity, Difficulties, Advantages, Information Systems, .NET platform, .NET framework, and Game Development. An inductive research technique, in conjunction with the interpretative research philosophy, was used to effectively demonstrate the insights and key aspects. As a result, the discussion and analysis were carried out using a content analysis approach in which significant topics related to the study objectives were developed to meet the research paper's target and objectives [15;16].

As a result, SLR (System or Systematic Literature Review) provides further support for the methodology's aforementioned qualities. The SLR technique seeks to locate, evaluate, and summarize certain issues. SLR allows for more restricted database design and evaluation with fewer biases than standard reviews [17]. As a result, SLR consists of five stages: the development of a review question; a collection of relevant previous research works; inspection of the gathered sources; data synthesis and summarization; and finally, the analysis of the results. As a result, Fig. 2 shows a clear representation of SLR phases.

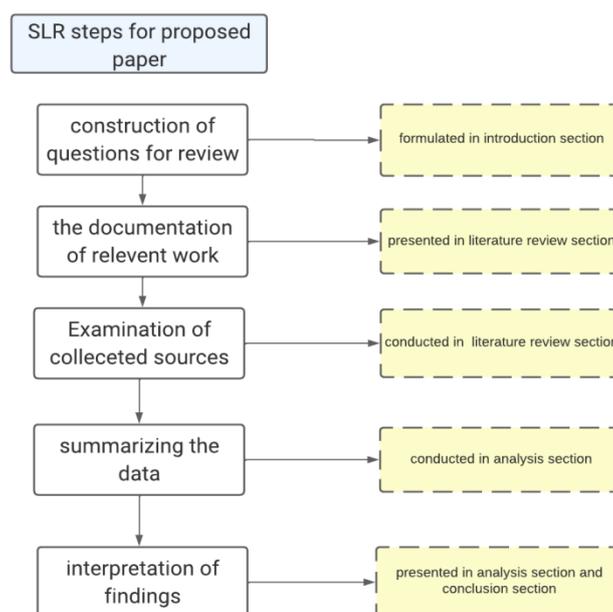


Figure 2 SLR steps for proposed paper

4. Analysis and discussion

Discussion and analysis have been undertaken in this part based on the study goals. Appropriate topics have been developed, which will eventually assist in effectively presenting the study objectives and in reaching the research paper's aim.

D. Advantages of using Unity: game development engine

It has been revealed that Unity 3D provides certain benefits that are not clearly m, and the author of this research work feels that this capability should be advertised. Unity's game creation features, such as processing, monitoring systems, scripting, and physics, serve to reduce game development expenses and duration while also allowing for project execution across several platforms [8]. The Unity game engine has risen to the top of the multi-platform game creation food chain, according to experts. According to Sarosa, Chalim, Suhari, Sari, and Hakim [12], on a development platform that is easy to use, this powerful cross-platform games engine made it possible to create gaming apps for 27 different devices and platforms. It offers ready-to-use parts, fundamental tools, tutorials, comprehensive manuals, and an online forum for creating outstanding free 3-D game models. The Unity 3D Platform has a roughly 45 percent global share of the game engine industry, and 47% of game developers have chosen Unity as an intelligent game development tool, according to a survey conducted in Hussain, Shakeel, Hussain, Uddin, and Ghouri [18]. It is simple to share the Unity-developed and delivered application technique across Windows, mobile devices, and online platforms. In addition, Kröger, Raschke, Campbell, and Ullrich [19] explains that the agile structure of Unity enables rapid release and prototyping, thereby accelerating game development. The Unity IDE also provides a text editor for coding. To avoid confusion, developers may use a different code editor. Additionally, the IDE provides useful game production tools and supports C# and JavaScript programming. According to Hocking [20], the Unity engine allows for high-quality music and visual effects, which aid in intelligent and productive game development. Without losing or altering picture quality, videos may be adapted for all devices and monitors.

Also, new developers demand significant documentation, which Unity provides. Practically every item is thoroughly detailed in the substantial documentation. Modification and debugging are much faster with Unity game production because all game elements and variables are visible during gaming, allowing programmers to debug the technique while it is running.

E. Issues of Unity with .NET platform

According to Comber, Motschnig, Mayer, and Haselberger [21], Unity 3D offers a number of unique qualities that allow developers to quickly construct a variety of mobile games. Yet, some of its feature documentation is out of date, and certain functions are unavailable. Up to Unity 5.0, the game production engine has been tuned for 32-bit operation. This means that when a programmer runs out of storage, the browser may crash unexpectedly. The engine is also unsuited for creating AAA games. There hasn't been a recent mention of a new OpenGL update for 4.X. As a result, Mac and Linux users cannot use features like Computer Shader and Geometry Shader. Even the licensed Unity version contains full mobile functionality. In this case, users will need to spend an extra \$1500 to \$3000 to maintain the Mobile Pro subscription. This looks to be rather costly. The visuals of the engine have been inverted. It does not have as many capabilities for creating realistic images as other game development engines. The Unity 5 game design engine's built-in compatibility with the PhysX physics engine has several performance concerns, and it lacks key important characteristics necessary to produce world-class gaming software [22].

F. Best approaches of Unity for enhancing the skills of Unity developers using .Net Framework

Microsoft itself. The entire NET ecosystem has developed, adding new licensing options and documentation that is not related to Windows. They could make Unity better. NET Mono Environment in 2018 and, as a result of this effort, enabled more recent C# language versions (7.0+). In the same year, they released the first version of the Burst compiler, which was the first to make it easy to create native C# code quickly. Because of this discovery, Unity was able to envision a scenario in which users might be able to use C# in other vital parts of the system without having to build these parts in C++, which led to the creation of the DOTs runtime.

New is included in both Unity 2020 LTS and Unity 2021 LTS. NET APIs and new versions of the C# programming language. Parallel to this, we've seen significant performance enhancements with the introduction of SDK-style csproj and the expanding NuGet ecosystem. NET environment and a programming model that is easier to use. Now that, we may begin upgrading the Unity run-time in various ways. Unity 2021 LTS supports NET Standard 2.1. We are working on two improvements at the moment [20].

1) Enhancement of the async/await programming models

A crucial programming paradigm for developing gaming code that must wait for an asynchronous operation to finish without obstructing the engine's main loop is `async/await`. Solidarity upheld non concurrent tasks in 2011 utilizing iterator-based co-solidarity schedules before `async/await` became well known. NET, even though this approach is less effective and incompatible with `async/await`. Meanwhile, `async/await` compatibility in C# and .NET Standard 2.1 has been improved. With Value Task and Async Method Builder, users can now make use of these advancements and work on incorporating `async/await` into Unity's current asynchronous activities. NET by enabling improved efficient workflow of `async/await` events. As a first step, we are enhancing the functionality for canceling pending asynchronous processes when a Mono Behavior is destroyed or Play mode is ended using the cancellations token. To ensure that developers can take advantage of these new capabilities, developers have also been closely collaborating with the most outstanding community initiatives, such as the creator of UniTask [3].

To reduce memory allocations and duplicates, SpanT> is being utilized. There is a lot of data between Unity and its C# programming layer, which is a C++ engine. Since it typically requires either the creation of newly managed objects or the transmission of data, this is inefficient.

SpanT> is currently standard and was created in C# 7.2 to improve comparable instances. NET 2.1 Standard It's likely that users have heard or read about a lot of significant speed enhancements to the. SpanT> was used to implement the NET Runtime in recent years. The user wanted to use it in Unity because it will improve the overall efficiency of several APIs and reduce allocations and, as a result, garbage collection latencies [23].

The success of the Unity game engine, which employs C# as its language, is most likely to blame for the increase of C# and probably the collapse of C++ in the Languages category. The rise in the popularity of JavaScript might be attributed to the progress of web technology. Yet, the significance of web development for a computer game designer/programmer was rated fairly low. Visual Language was considered highly essential among our additional elements. This is most probably due to the increasing ubiquity of Unreal Engine, but it is also due to graphical languages in other prominent technologies, such as Unity's Shader Graph, and content production tools such as Blender and Arcweave [24].

2) Prefab system

Users may use Unity to build reusable assets, which allows for faster game development. The Prefab system in Unity enables users to build complicated hierarchies that are simple to maintain, modify, and update across numerous levels. A prefab resource is a design that does not have to be similar to the one that was previously generated.

Users may quickly alter current prefab assets and build new ones. When it comes to asset change, the extension is accomplished through association rather than hierarchical inheritance, allowing for logical and code consistency. Also, when an object is modified, it does not disrupt the entire network of objects of the same type, but only alters the present object without causing any problems with the code. The Unity Prefab system substantially accelerates prototyping [25].

5. Conclusion

To sum up, the goal of this exploration was to bring issues to light and give data in regard to Solidarity to individuals working in the IT region. There are three goals set for achieving the goal: investigate the advantages of Unity, investigate the potential issues with Unity, and describe the most effective Unity methods for enhancing the abilities of Unity developers. Three questions are posed in response to the three objectives, one of which asks, "What are the advantages of Unity?" How much of an impact do Unity issues have on game designers? What are the best ways for Unity developers to increase their capabilities?

The paper employs qualitative research supported by secondary data throughout. To gather insights and intelligently discuss them, the strategy and search approach are also connected to an interpretive research methodology and an inductive method of study to help readers develop relevant knowledge and comprehension. The point investigation data was assessed utilizing the substance examination method. As a consequence of this, it has been shown that Unity has been working in the field of game development and that its efficacy and effectiveness, as well as the usage of Unity and the industry's devotion to it, are growing over time. As a result, it has several positive effects on game development, such as boosting career opportunities, increasing the value of game developers, and improving career prospects.

References

- [1]. C. Vohera, H. Chheda, D. Chouhan, A. Desai, and V. Jain, Game Engine Architecture and Comparative Study of Different Game Engines. *12th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, 2021, 1-6.
- [2]. A. Juliani, V.-P. Berges, E. Teng, A. Cohen, J. Harper, C. Elion, et al., Unity: A general platform for intelligent agents, *arXiv preprint arXiv: 1809.02627*, 2018.
- [3]. A. Chia, B. Keogh, D. Leorke, and B. Nicoll, Platformisation in game development, *Internet Policy Review*, vol. 9, 2020, 1-28.
- [4]. A. A. Selçuk, A guide for systematic reviews: PRISMA, *Turkish archives of otorhinolaryngology*, vol. 57, 2019, 57.
- [5]. B. Nicoll, and B. Keogh, *The Unity game engine and the circuits of cultural software* (Cham: Palgrave Pivot, 2019).
- [6]. [R. Xie, S. Zlatanova, and J. Lee, 3D indoor environments in pedestrian evacuation simulations, *Automation in Construction*, vol. 144, 2022, 104593.
- [7]. K. Eriksen, B. E. Nielsen, and M. Pittelkow, Visualizing 3D molecular structures using an augmented reality app, *Journal of Chemical Education*, 97 (5), 2020, 1487-1490.
- [8]. N. Nurym, R. Sambetova, M. Azybaev, and N. Kerimbayev, Virtual Reality and Using the Unity 3D Platform for Android Games, *2020 IEEE 10th International Conference on Intelligent Systems (IS)*, 2020, 539-544.
- [9]. A. F. Hastawan, P. K. Nashiroh, A. A. Firdaus, and H. Rossa, Designing Educational Game of Indonesian Traditional Musical Instruments Based on Android Using Unity 3D, *1st Vocational Education International Conference (VEIC 2019)*, 2019, 92-100.
- [10]. K. Sung, and S. Gregory, *Basic Math for Game Development with Unity 3D: A Beginner's Guide to Mathematical Foundations*, (Berkeley, CA: A press, 2019).
- [11]. Anonymous. (2021, 12/3). *Unity User Manual 2021.3 (LTS)*. <https://docs.unity3d.com/Manual/index.html>
- [12]. M. Sarosa, A. Chalim, S. Suhari, Z. Sari, and H. Hakim, Developing augmented reality based application for character education using unity with Vuforia SDK, *Journal of Physics, Conference Series*, 2019, 012035.
- [13]. A. Gallidabino, *Liquid Web Applications - Design and Implementation of the Decentralized Cross-Device Web*, doctoral diss., Università della Svizzera italiana, Lugano, 2020.
- [14]. D. Loreto Ines, and G. Abdelkader G. A. Di Loreto Ines, Facebook Games: Between Social and Personal Aspects, *International Journal of Computer Information Systems and Industrial Management Applications*, vol. 3, 2011, 713-723.
- [15]. D. Silverman, *Qualitative Research*, (London: SAGE Publications, 2020).
- [16]. K. L. Rieger, Discriminating among grounded theory approaches, *Nursing inquiry*, vol. 26, issue 1, 2019, e12261.
- [17]. S. Kraus, M. Breier, and S. Dasí-Rodríguez, The art of crafting a systematic literature review in entrepreneurship research, *International Entrepreneurship and Management Journal*, vol. 16, 2020, 1023-1042.
- [18]. A. Hussain, H. Shakeel, F. Hussain, N. Uddin, and T. L. Ghouri, Unity Game Development Engine: A Technical Survey, *Univ. Sindh J. Inf. Commun. Technol*, vol. 4, 2020, 73-81.
- [19]. [J. L. Kröger, P. Raschke, J. P. Campbell, and S. Ullrich, Surveilling the gamers: Privacy impacts of the video game industry, *Entertainment Computing*, vol. 44, 2023, 100537.
- [20]. J. Hocking, *Unity in action: multiplatform game development in C* (Simon and Schuster, 2022).
- [21]. O. Comber, R. Motschnig, H. Mayer, and D. Haselberger, Engaging students in computer science education through game development with unity, *2019 IEEE Global Engineering Education Conference (EDUCON)*, 2019, 199-205.
- [22]. Potenza. (2022, 12/3). *5 Rarely Known Advantages and Disadvantages of Unity Game Development 2022*. <https://blog.potenzaglobalsolutions.com/5-rarely-known-advantages-and-disadvantages-of-unity-game-development>
- [23]. C. G. Diaz, P. Perry, and R. Fiebrink, Interactive machine learning for more expressive game interactions, *2019 IEEE Conference on Games (CoG)*, 2019, 1-2.
- [24]. K. Takoordyal, Introduction to unity, *Beginning unity android game development: From beginner to Pro* (Berkeley, CA: Apress, 2020) 29-58.
- [25]. J. Halpern, *Developing 2D Games with Unity* (Berkeley, CA: Apress, 2018).