Unity Game Development Engine: A Technical Survey

Article · October 2020

CITATIONS
40

READS
24,315

5 authors, including:

Faizan Hussain
Hamdard University
11 PUBLICATIONS **67** CITATIONS

SEE PROFILE

University of Sindh Journal of Information and Communication Technology (USJICT)
Volume 4, Issue 2, July 2020
ISSN-E: 2523-1235, ISSN-P: 2521-5582
Website: http://sujo.usindh.edu.pk/index.php/USJICT/

Unity Game Development Engine: A Technical Survey

Afzal Hussain[1], Haad Shakeel[1], Faizan Hussain[1], Nasir Uddin[1], and Turab Latif Ghouri[1]

[1]Department of Computer Science, Faculty of Engineering Science and Technology (FEST), Hamdard University Karachi, Pakistan
afzal.hussain@hamdard.edu.pk, haadshakeel140@gmail.com, faizanhussain535@outlook.com, nsrkh5585@gmail.com, and turabghori@gmail.com

Abstract: Paper aims to develop relevant understanding and knowledge about the Unity game development engine for the better conception of the people allied with the IT sector. The main audience targeted in this paper are those who are from a non-technical and technical background and due to the augmentation of technology, they want to pursue their careers in game development. Moreover, in this paper, qualitative research methods have been adopted. It is found that Unity is a smart and active game development platform that is playing an operative role nowadays. Different industries are inspired by Unity that may impact positively such as in career growth, job opportunities, and in other regards. Unity comes with many benefits; it is an easy and simple platform to learn game development and is a powerful tool that is preferred by professionals.

Keywords: Unity, Information Technology, Benefits, Game Development, and Challenges.

I. INTRODUCTION

Unity has been developed under the umbrella of Unity-Technologies that is a cross-platform game engine released in 2005 June at World Wide Apple Inc's Conference as an exclusive game engine (Mac OS X). In 2018, the engine was extended and facilitated more than 20 platforms. This game engine can be used to develop the augmented reality, virtual reality, two-dimensional, and three-dimensional, games along with simulations and for other practices. This game engine in the 21st Century has been adopted by businesses outside video gaming like film, architecture, automotive, construction, and engineering [1]. This paper is presenting survey on the Unity as in the current epoch amid the IT (Information Technology) sector the surge of gaming and its development has emerged enormously; however, Unity is playing actively in game development and thus, the researchers of this paper deem that there must be some comprehensive survey that supports the people working in IT sector to connect with the benefits, challenges, and appropriate solutions of those challenges that emerged in the Unity. This ultimately helps the readers and the people allied with the IT sector in the development of relevant knowledge and understanding of Unity.

A. Research Paper Aim

To develop relevant understanding and knowledge about Unity for the better comprehension of the people allied with the IT sector. To attain the aim, three objectives and questions are developed that are depicted below.

B. Research Paper Objectives

1. To examine the benefits of Unity game development engine.
2. To inspect the challenges that may emerge in Unity.
3. To outline the best approaches of Unity for enhancing the skills of Unity developers

C. Research Paper Questions

1. What are the benefits of Unity game development engine?
2. To what extent the challenges of Unity affect the people working in the game development domain?
3. What are the best approaches to Unity skills for enhancing the skills of Unity developers?

D. Research Paper Layout

To achieve the aim and objectives the research paper has been segregated into five sections. Section 1 covers the introduction, section 2 comprised of the literature review, section 3 depicts the methods and material adopt to conduct the study, section 4 covers the discussion. Section 5 covers the conclusion and future recommendations.

II. LITERATURE REVIEW

In this section of the proposed paper, the literature review has been conducted. The documentation of relevant works and examination of the collected sources fundamentally is demonstrated in Fig 1. Moreover, Fig 1 is constructed with the help of a technique named as PRISMA, depicted below. Therefore, the literature review has been considered as one of the vital aspects of the research paper as its supports authors as well the readers in understanding and examining several

facets that the relevant to the current topic and are conducted previously.

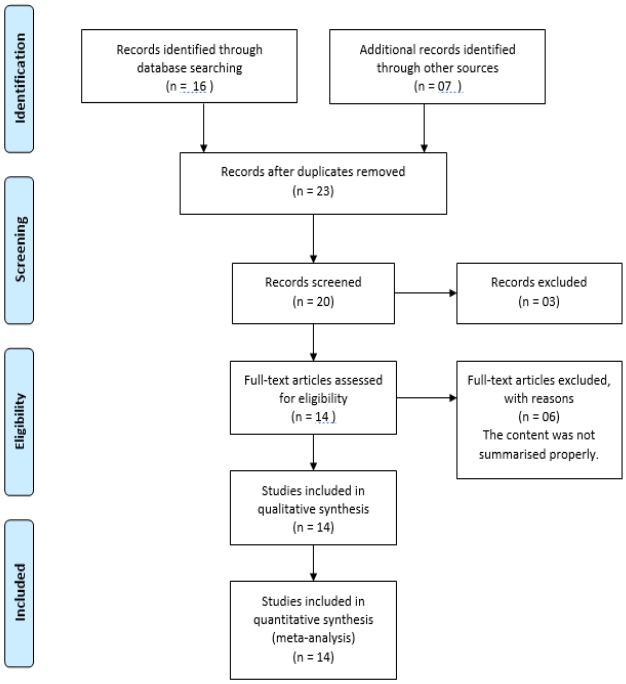

Figure 1. PRISMA Chart

Unity, a multi-platform game engine, commercially available and is used for 2d and 3D video games production accompanied by visualizations and non-game interactive simulations. Moreover, Unity is one of the popular engines for games that is easily accessible, and in the current epoch is communal amid the developers due to its ease, flexibility, efficiency, and power consumption. Multiple tools have been featured in the Unity Editor that permits rapid iteration and editing in the cycles of development comprised of smart previews play mode in real-time [2]. Furthermore, Unity is offered on Mac, Linux, and Windows it contains an artist-friendly range of tools for immersive designing and game worlds, in addition to a strong developer tools suite for implementing high-performance gameplay and game logic [3]. Moreover, Unity supports 3D and 2D development with functionalities and features for the specific needs that are further focused on categories.

Similarly, [4] has quoted that the navigation system has also been included in Unity that permits to create NPCs that logically move around the world of gaming [28]. navigation meshes have been used in the system that is automatically created from the scene geometry, or sometimes even from the dynamic obstacles, to change the character's navigation at runtime. Unity Prefabs that are known as the pre-configured objects of gaming provide the workflow flexibility and efficiency that permit in the confident working without nerve-wracking about the time-consuming errors [5]. According to [6] it has been found that Unity built-in UI system permits to create the UI intuitively and smartly with less consumption of time. Unity takes Box2D advantage the novel DOTS-based NVIDIA PhysX and Physics system that aids in the provision of high-performance and high realistic gameplay. Developers can extend the Editor of Unity with tools need to match the workflow of teams. It also supports the creation and customizing of the extensions that feature many possessions, extensions, and tools for the sake of speediness of the projects [29].

According to [2] one of the feature highlights of this discharge for Android is the accessibility of a seen form of Adaptive Performance for Samsung Galaxy leads. In contrast to PC and consoles, gaming on cell phones characteristically impediment of warmth the board and power utilization. Lovely looking and smooth-messing around have escalated handling needs, which can rapidly warm up your gadget. PC and consoles handle this issue through their dynamic cooling frameworks, yet since telephones don't include dynamic cooling equipment (yet), the telephone winds up throttling execution to hold the temperature within proper limits. The issue turns out to be much progressively risky considering the wide scope of equipment accessible, and the shifting execution and throttling situations. In addition, some of the key features of unity are related to the simple workflow that allows developers to quickly combine scenes in a workspace referred to as (intuitive editor). It also supports creating high-quality games like AAA images, high definition sound, and action at full speed with no gaps on the screen [32]. Also, some extra features are [33] access the components, coroutine and return types, creating and destroying GameObjects, dealing with vector variables and timing variables, events for GameObject, and physics-oriented events. Primarily, Unity supports scripting in C#, and there are two ways to design C# scripts in Unity: object-oriented design, which is the more traditional and widely used approach, and data-oriented design, which is now possible in Unity [34].

Moreover, according to [1] game designers handle this issue through two fundamental methodologies: guaranteeing the most extreme similarity by giving up realistic loyalty and casing rate, or by foreseeing equipment conduct, which is hard to execute. Solidarity and Samsung have teamed up for a component called "Versatile Performance", which gives a superior method to oversee thermals and execution of games continuously. After you introduce Adaptive Performance through the Unity Package Manager, Unity will consequently add the Samsung GameSDK subsystem to your undertaking. During runtime and on upheld gadgets [7]. Unity will make and begin an Adaptive Performance Manager which will give criticism about the warm condition of the gadget [30].

Engineers would then be able to decide to buy into occasions or inquiry the data from the Adaptive Performance Manager during runtime to make responses progressively concerning warm patterns. For example, when the gadget started throttling in the beginning periods, the game could tune quality settings, target outline rate, and different parameters to guarantee that the game can squeeze out increasingly continued execution. When the temperature

begins declining once more, parameters could be changed by and by to convey better ongoing interaction execution. By watching out for the warm presentation, one can abstain from throttling all together by modifying execution dependent on constant criticism. This will prompt a gradual unsurprising edge rate and interactivity experience and lower warm buildup [1, 2, 8].

Therefore, based on the studies examined in the literature review, a conceptual framework is formulated exposed below (See Fig 2). Hence, this supports the readers in analysing the gap that may have a reflection on delineating the concept of the proposed paper.

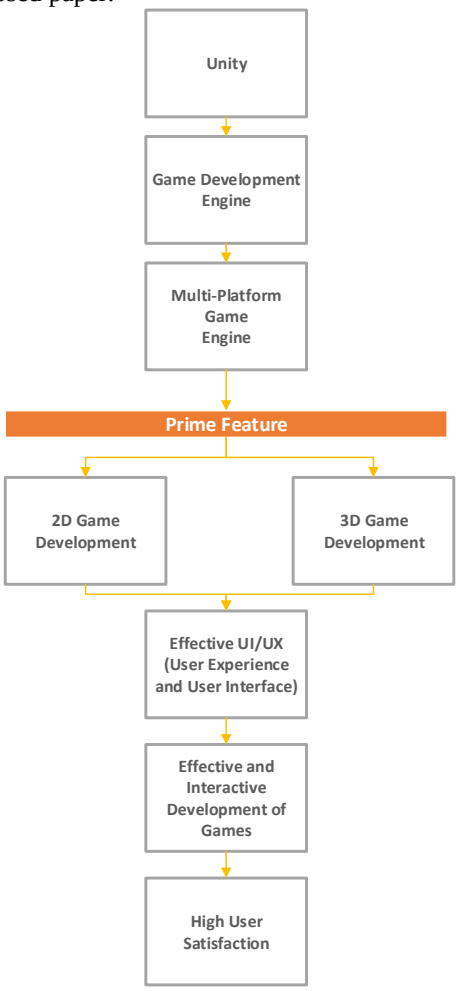

Figure 2. Conceptual Framework

III. METHODOLOGY

Qualitative research methodology has been adopted in this paper in which the data is gathered from already available researches available in the form of peer-reviewed journals, articles, books, and other sources. The collection of data has been done with the help of a secondary search strategy that has been done in the form of examining and analysing the studies that are conducted previously aimed towards Unity game development mainly journals and proceedings of conferences. To make this process more effective researchers have set some restrictions. For example, the data is only gathered from 2008-2020 via using several keywords like Unity, Challenges, Benefits, Information Technology, and Game Development from reliable databases like Google Scholar, Academia, IEEE Xplore, Scopus, ProQuest, and Research Gate. An inductive research approach along with the interpretivism research philosophy has been adopted for the demonstration of the insights and vital facets passably. Therefore, the discussion and analysis have been conducted through a content analysis technique in which relevant themes in accord with the research objectives are formulated to achieve the research paper aim and objectives [9,10, 11]. Therefore, the above-mentioned aspects of the methodology are further supported by SLR (System or Systematic Literature Review). SLR method aims towards identifying, evaluating, and summarizing the state-of-the-art on a specific topic. SLR permits for the restrictive compilation of databases and enable analysis with less bias than traditional reviews [31]. Hence, there are five stages of SLR such as the construction of a question for the review; the documentation of relevant works; examination of the collected sources; synthesis of data and summarizing; and lastly, interpretation of the findings. Hence, see Fig 3 for a clear demonstration of SLR steps.

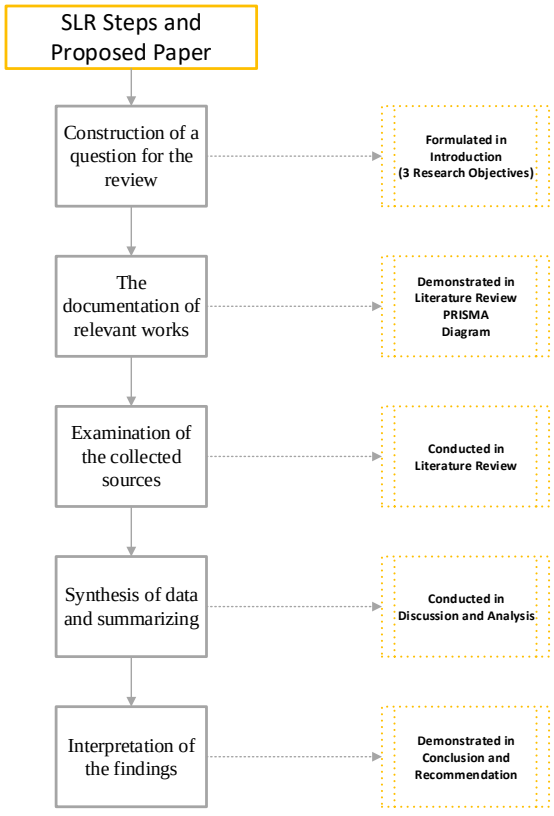

Figure 3. SLR Steps and Proposed Paper

IV. DISCUSSION

In this section, based on the research objectives discussion and analysis have been conducted. Relevant themes have been formulated that ultimately support in discussing the research objectives properly and also aids in achieving the research paper aim.

A. To Examine The Benefits Of Unity Game Development Engine

With reference to [8], it is found that Unity 3D comes with several benefits that are not highlighted passable and, on such facet, the researchers of this paper deem that this aspect must be disclosed. Unity aids in processing, asset tracking, scripting, and physics are some of Unity's game development features that diminish game development costs and time and offer elasticity when implementing projects on more than one platform. Professionals have stacked the Unity engine at the top of the ladder of multiplatform game development. According to [12] it is found that this powerful cross-platform game development engine permitted the development of game applications for 27 diverse platforms and devices in a user-friendly development environment. It bids affluence of resources like ready-to-use elements, intuitive tools, tutorials, clear documentation, and the online community for creating amazing free 3D content in games. According to the survey conducted in [13], Unity 3D Engine has a global market share of Game Engine which is around 45%, while 47% of game developers prefer Unity as a smart game development tool. The developed and implemented application process via Unity can easily be shared amid PC, mobile, and web platforms. In addition, it is found from [3] the nature of Unity has been grounded on the agile methodology that allows constant release and rapid prototyping, which speeds up game development. Likewise, Unity IDE provides the text editor for writing the code. However, sometimes developers use a separate code editor to avoid confusion. Moreover, the IDE supports C# and JavaScript for scripting and offers important functions that are suitable for game development. According to [14] it if found that the Unity engine supports high-quality video and audio effects that facilitate game development smartly and effectively. Videos can be adjusted on all devices and screens without compromise or distortion on picture quality. Moreover, new developers need easy-to-understand documentation that has been provided by Unity in detail. The thorough documentation contains an explanation of almost every single unit. The tweaking and debugging are easier with game development accomplished via Unity remarkably as all the game units and variables are depicted through gameplay that permits the developers to debug the procedure at runtime. The benefits of game development that are easily available through Unity are depicted in Table 1 and 2 in which some scenarios have been listed and how through Unity such aspects can be attained that comparatively not easy or passably achievable by other platforms.

TABLE I. SCENARIOS, UNITY GAME AND ENGINE AND ANALYSIS [2,5,9,11, 27]

The scenario of Game Design	Solution through Unity game engine	Comparative analysis with different games engines
Create a 3D environment that reconciles visual quality with game performance	Maximum visual quality for the display in the real world that is shadow and light, texture map, alpha channel, independent animation time	Less graphic quality structures (VRML, Flash-based)
Create navigation system of which permits the user to discover maximum freedom degree	Permit maximum freedom degree to navigate and explore in the virtual environment	High degree of the navigation system (Bentley-3D PDF), less flexible freedom of movement (QuickTime VR) view direction, and pre-programmed animation sequence (3D animation)
Create a navigation system which permits end-user to scrutinize a particular interest object in many perspectives way	Method to augment spatial understanding	High degree navigation system, view direction, (Bentley-3D PDF)
Create a method to integrate several information types	Rich content and other data leaping technique	Require server-based interaction and script language (Flash-based)
Create a method for flexible 3D data exchange	Method for 3D data identical to the external tool	Less supple to accomplish data synchronization (other animation packages and Flash-based)
Create a translate conventional analysis study method	The basic visual-based analysis method is present	Less flexible (Bentley-3D, Flash-based, and QuickTime VR)
Flexibility for the expansion of game design	Possible due to object-oriented programming	Not possible because of structural programming (Bentley-3D, Flash-based, 3D animation and QuickTime VR)

TABLE II. SCENARIOS, UNITY GAME AND ENGINE AND ANALYSIS [1,6,11,24,25, 27]

Engine Name	Lighting	Geometries I/O	Texture Mapping
Unreal Engine 2	Dynamic shadow and lightening HDR Rendering	Built-in editor, any data CAD essential to be converted	3D shader management vegetation generator.
Source-Half-Life 2	Dynamic shadow and lightening HDR Rendering	No built-in editor, any data CAD essential to be converted	Optimization for big open area
Gamebryo-Oblivion	Dynamic lighting and shadow	Built-in libraries with the editor, any data CAD essential to be converted	3D vegetation generator
Unity	Subsidize direct light to the scene and support in updating every frame. directional, point and spot are Realtime	Built-in libraries with the editor, optimized for simple geometry construction, but also detailed processing and UV recursion	The texture map is a list of 2D UV coordinates assigned to its vertex counterparts 3D vertices in three dimensions (x, y, z).

Therefore, some of the benefits of Unity in terms of 2D, 3D, VR (Virtual Reality), and AR (Augmented Reality) has been depicted below in Fig 4, 5, 6, and 7.

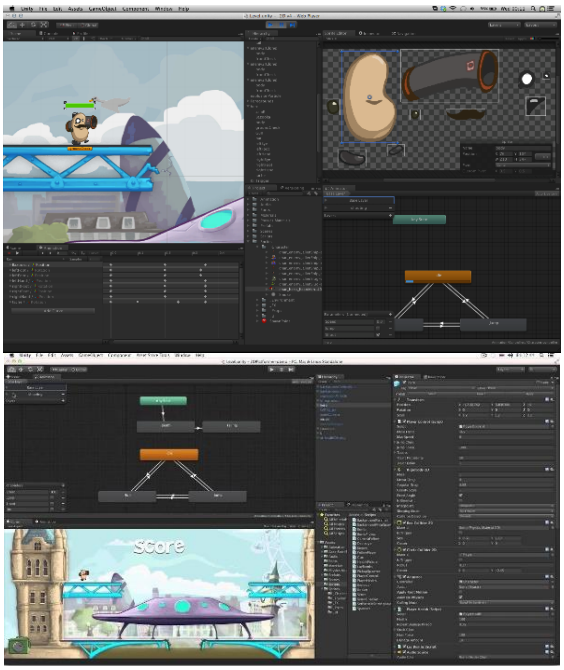

Figure 4. Unity Game Development Engine and 2D-Image

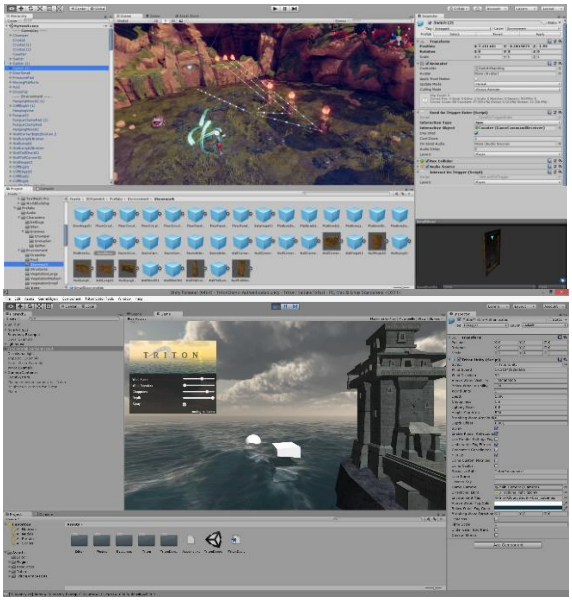

Figure 5. Unity Game Development Engine and 3D-Image

Figure 6. Unity Game Development Engine and VR-Image

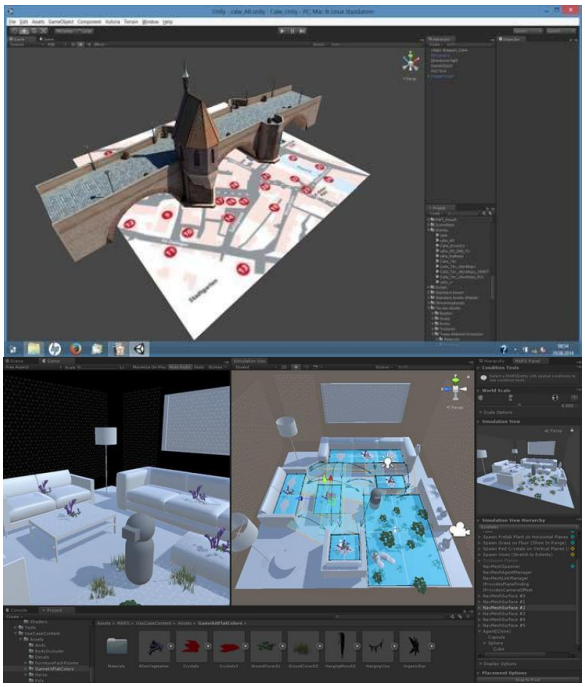

Figure 7. Unity Game Development Engine and AR-Image

B. To inspect the challenges that may emerge in Unity.

With reference to [15], it is found that Unity 3D offers many exclusive features that permit developers to develop diverse mobile games time-effectively. However, the documentation for some of its functions is out of date and not available for some functions. Up to Unity 5.0, the game development engine is designed for 32-bit operation. This means that the editor may crash wordlessly if a developer is

run out of memory. The engine is also not appropriate for the development of AAA games. No new update for OpenGL support to 4.X has been announced recently. Therefore, functions like Computer Shader or Geometry Shader for OSX or Linux are not available. Even the Unity licensed version does not offer all mobile functions. In this case, you will need a supplementary investment of $1500 to $3000 to operate your Mobile Pro license. This seems a bit steep. The engine is graphical upside down. It doesn't offer several other tools for creating prudent graphics than other game development engines [16]. In the Unity 5 game development engine, built-in support for the PhysX physics engine has some challenges regarding the performance, and it lacks in some key features that need to be added to build the world-class gaming app. Similarly, developers must have licensed to get the best deployment, graphics, and performance advances. Likewise, buying these licenses is expensive. The use of buffer support, rendering, template support, and many other functions increases the development costs due to expensive licenses [15-16].

Additionally, it has been found from the [17-18] study that the code is more stable in Unity than other engines and has a fantastic architecture that advances the performance of the game application. Yet, due to source code unavailability, it is difficult to fix, find, and address performance issues. The Unity engine influence more memory in accordance with the game development that ultimately causes debugging issues and OOM errors in the applications that have been developed under the umbrella of Unity. Simultaneously, in the field of game application development, organizations rank the Unity 3D engine for mobile game development on a priority basis or it can also be stated as on the top of the list despite its several challenges. Unity updated versions have incessantly advanced the game development engine and fixed the exiting challenges on a passable scale, which can also be considered as one of the reasons behind their upsurge and immense use in the domain of game development. Hence, each coin has two sides; henceforth, even though there are challenges there are also benefitting that is supporting the developers to attain high-quality outcomes.

C. To Outline the Best Approaches of Unity for enhancing the skills of Unity developers

The role of Unity Developer is comparatively new and tougher to describe than some of its precursors. Some developers focus the Unity only on the art side whereas some prefer it for the coding that may help in the process of game development. Nevertheless, [19] has quoted that somewhere amid them there is more to do like building tools to aid out their more exceptionally focused allies. As of this diversity, it's vital to look at the Unity role from an upper level to classify the four wide-ranging approaches and skills depicted in Fig 8 that benefit virtually in the context of accomplishing the high-quality outcomes.

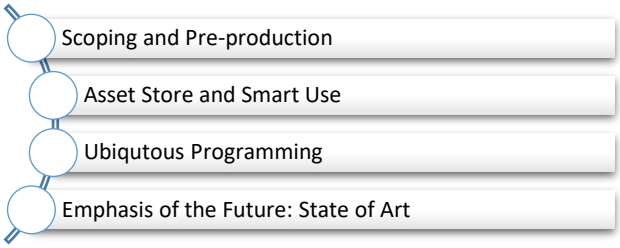

Figure 8. Approaches and Skills [19]

- Scoping and Pre-production; it has been considered as the most vital aspect of Unity. It is also found from [19] study scoping and pre-production are the two most vigorous skills that must be catered and considered by the Unity developer in order to be successful. A sign of a good Unity developer is being able to reliably ship goods, and one of the chief project killers is feature creep. Moreover, Feature creep has been well-defined by [20] that it occurs when a developer or any other team member has another great idea that needs to be almost completely integrated into the game, which often leads to the next or equally great idea to solve the problem. The most operative way to evade feature creep is to create a project production roadmap and plan) pre-production) before starting the proper development in such facet a developer must have to design and define exactly which functions they want it in the endgame and what the end product will look like in quantifiable terms, for instance, some game characters, special characters, and others. However, this does not mean that things do not change during production, things might get changed in the development process; nevertheless, it depends on the methodology that is adapted for game development. Therefore, in this regard, for instance, to mitigate the sudden change in the game development professionals and experts have set some tips and tricks that are allied to testified and validated process that are present as per the methods, guidelines, and approaches and amid some beneficial facets scoping has been considered as the vital aspects in every development arena whether it is connected to technical and non-technical. Scoping helps in pre-production facet that may occur in the form of gathering and understanding the needs of the end-users and by proper documentation such gathered aspects are jotted it down that help in crystalizing and transparency factors which eventually support in the mitigation of feature creep and other challenges that may impact on the overall game development process. In addition, it is found from the [21] study that the importance of scoping and pre-production is often rejected by inexperienced teams or those from other industries, but for the game development scoping and pre-production is considered critical for the success of the projects or products. Pre-production does not have to be a formal or extended process. Just give enough time to clearly define project goals and by locking it adequately quality outcomes can be accomplished.

Therefore, the researchers deem that a game developer who is developing the game under the umbrella of Unity must focus on this approach. It is also advised that give pre-production time as long it requires that ultimately help in accomplishing the quality outcomes in the end-product. It might seem like a slow process; however, if you are working on a project or product for six months or even two years and have inevitably lost track of what you are working on, it is very helpful to have a document that helps you adequately throughout the process of development as it helps in the mitigation of assumption based development.

- Asset Store and Smart Use; Using the Asset Store is often a bad name due to the increasing spread of flip asset games that are flooding online sales platforms like Steam. Developers can buy or purchase assets in the Unity Asset Store and use the launched demo scenes as the base for their own game. Even non-developers have noticed that the store and some of its most popular assets are used to flip. Also, the end-user is generally unaware of how often legitimate dev. successfully uses the asset store. A good example of this is Blizzard's online card game Hearthstone, which uses PlayMaker, a pictorial scripting tool accessible from the Asset Store. Other popular games with Playmaker are Inside and The Forest, Hollow Knight. In these cases, it can be stated this approach of asset store and its smart use has been considered as an active method that has also been suggested to the Unity developer to adopt for better retrieving the services [19]. Therefore, it also ensures the success of high-quality outcomes. In addition, the Asset Store is a very powerful tool for Unity game developers throughout the production stages if it is used intelligently. A great way to imagine the Asset Store is with an up-to-date resource library that an art studio like Disney needs to support its team. From sound effects to moving reference films, these tools are used as an early point for an artist's creative endeavour. The key to making good use of the Unity asset store is to look at it as a way to rebuild the wheel. As long as game developers make sure that what they have downloaded from the Unity asset store aimed at setting up the best class in the project despite adapting the things generally [22].

- Ubiquitous Programming; Although Unity may not be programmable, there is ultimately no way to create more multifaceted projects. However, you need to know how to code C# and JavaScript as Unity is supporting such two-programming language depicted earlier. There are some workplaces but the good news is that everyone can program easily which makes Unity as ubiquitous programming platform that means programing for everyone. Fans, artists, old, young, everyone can learn to write C# and JavaScript and can work on Unity and it is a famous quote that practice makes you perfect and if someone had the potential, he/she can easily work and gain expertise in Unity game development easily [19]. However, this approach has been suggested for both technical and non-technical individuals that help in the success factor. On the other side, it is found from the [23] study that there is a widespread misunderstanding that it takes a certain kind of mind to understand the scriptures. However, solving problems with methods and variables is itself a creative undertaking. Learning a scripting language and writing your logic is more than learning a set of rules and thinking in a certain way. All it takes is patience and a desire to learn. It is recommended that people interest in the game development must work in accordance with the Unity as it comes with ubiquitous programming that may also help in their career growth as technology currently and in the forthcoming time may play an active role all over the globe and it is evident that global job market will have a big ratio of tech-based jobs. Therefore, for technical and non-technical people it is advised that they start learning if they want to advances in the game development arena for this, they may follow some script tutorials then challenge yourself to create something unique. Therefore, the more scripts an individual write, the more they understand [24].

- The emphasis of the future: State of Art; being a Unity developer is exciting with almost all Unity engines that are released. Unity crosses the boundaries of what modern game engines can do. Some of these new features are still in the progression stage; however, they are already in high demand as gaming companies use new technologies to reap the benefits [19]. The new main features that Unity developers should address as quickly as possible are depicted in Fig 9.

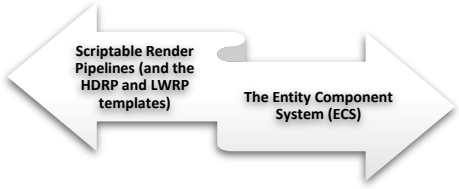

Figure 9. Future and Unity

Scriptable Render Pipelines, it is inspected from [25] that with the release of Unity 2018.1, end-users were able to write custom rendering pipes through C#. This means that developers can now regulate and cope with how things are carried out to the screen smartly. This can be helpful if the project has a dedicated look or desires to be optimized highly. They also released two templates of rendering pipeline: the HD rendering pipeline for high-end consoles and PCs where graphics quality pushed to its limits, and the lightweight rendering pipeline for low-end devices like a cell phone and platforms with special requirements like virtual reality and a highly polished appearance that surpasses the old rendering flow. Both templates give

access to new features like Shader Graph, a node-based visual system for developing custom shaders. This permits artists to get involved in creating specialized shaders that reserved for programmers exclusively.

ECS, it is a fundamental change in the way Unity projects are developed, particularly in the way things are written and scripted. Every developer who has registered with Unity is familiar with object-oriented programming [26]. However, ECS uses data-oriented programming. Data-oriented programming is unique because it offers integrated deep code optimization. If it can be run, it is fully optimized. This approach of Unity that has been based on the current tech-based trends permits Unity projects written with the ECS to push the boundaries, as newly revealed in Unite LA Prime 2018 [19]. Unity's development plans with a roadmap for public access are fairly open and often reveal new features long before they are fully published in the editor. It is good practice for Unity developers to check the YouTube channel, official blog, and roadmap itself regularly. While gaming has been the main focus of Unity, more industries are paying attention to what the Unity engine can do. Therefore, it can be stated that connecting with the Unity under the best practices and approaches is considered as the great opportunity for developers looking to expand their careers in the future, as Unity developers have more job opportunities than ever before that is evidentially and scientifically discussed in this paper under the qualitative research.

V. CONCLUSION AND FUTURE RECOMMENDATIONS

To conclude, the paper aims was to develop relevant understanding and knowledge about Unity for the better comprehension of the people allied with the IT sector. To achieve the aim, three objectives are designed such as, to examine the benefits of Unity, to inspect the challenges that may emerge in the Unity, and to outline the best approaches of Unity for enhancing the skills of Unity developers. Based on the three objectives, three questions are also formulated such as; What are the benefits of Unity? To what extent the challenges of Unity affect the people working in the game development domain? What are the best approaches to Unity skills for enhancing the skills of Unity developers?

The method that has been used throughout the paper is qualitative research that is further backed up by secondary data. Moreover, the method and search strategy are connected with inductive research approach, and interpretivism research philosophy for the achievement of insights and discussing it smartly that helps in developing the relevant understanding and knowledge of the readers. The content analysis technique has been used to analyze the data that has been done via thematic analysis. Therefore, it is found that Unity has been playing an active role in the arena of game development, and with the passage of time and effectiveness and efficiency as well as the usage and devotion of industries to Unity is amplifying. Thus, it comes with several positive impacts in the arena of game development like firming careers, job opportunities, and upsurging the value of game developers intensively.

To recommend, the important thing a Unity developer can do is to work on improving their skills continuously. The technology industry is never static, and the work landscape in five years will be very different from what it is today. A top Unity developer is forever a student who learns everything about Unity's new features as well as its time-to-time releases and works eagerly to improve what they already know about the engine so they can be prepared as per the current needs, wants and demands that also support in the career growth. Even though Unity has certain limitations, its advantages outweigh its disadvantages. As such, it is undoubtedly an effective podium for creating games. To choose a powerful tool that gives great features is obligatory to develop an adequate game that is available via Unity. Unity 3D is also easy to learn and use as discussed in this paper. It offers a very cheap pricing solution to meet the developer’s needs. A free version of Unity offers most of the functions. Though, if developers need advanced features that can always use the paid version.

REFERENCES

[1] Juliani, Arthur, Vincent-Pierre Berges, Esh Vckay, Yuan Gao, Hunter Henry, Marwan Mattar, and Danny Lange. "Unity: A general platform for intelligent agents." arXiv preprint arXiv:1809.02627 (2018).

[2] Haas, John K. "A history of the unity game engine." (2014).

[3] Canossa, Alessandro. "Interview with nicholas francis and thomas hagen from unity technologies." In-Game Analytics, pp. 137-142. Springer, London, 2013.

[4] Becker-Asano, Christian, Felix Ruzzoli, Christoph Hölscher, and Bernhard Nebel. "A multi-agent system based on unity 4 for virtual perception and wayfinding." Transportation Research Procedia 2 (2014): 452-455.

[5] Goldstone, Will. Unity 3. x game development essentials. Packt Publishing Ltd, 2011.

[6] Patil, Pratik P., and Ronald Alvares. "Cross-platform Application Development using Unity Game Engine." Int. J 3, no. 4 (2015).

[7] Zioma, Renaldas, and Aras Pranckevičius. "Unity: iOS and Android: cross-platform challenges and solutions." In ACM SIGGRAPH 2012 Mobile, pp. 1-1. 2012.

[8] Kim, Sung Lee, Hae Jung Suk, Jeong Hwa Kang, Jun Mo Jung, Teemu H. Laine, and Joonas Westlin. "Using Unity 3D to facilitate mobile augmented reality game development." In 2014 IEEE World Forum on the Internet of Things (WF-IoT), pp. 21-26. IEEE, 2014.

[9] Silverman, David, ed. Qualitative research. Sage, 2016.

[10] Burney, Aqil. "Inductive and deductive research approach." Department of Computer Science, University of Karachi, Pakistan (2008): 22.

[11] Ryan, Gemma. "Introduction to positivism, interpretivism and critical theory." Nurse researcher 25, no. 4 (2018): 41-49.

[12] Craighead, Jeff, Jennifer Burke, and Robin Murphy. "Using the unity game engine to develop sarge: a case study." In Proceedings of the 2008 Simulation Workshop at the International Conference on Intelligent Robots and Systems (IROS 2008). 2008.

[13] Blackman, Sue. Beginning 3D Game Development with Unity 4: All-in-one, multi-platform game development. Apress, 2013.

[14] Thorn, Alan. Learn unity for 2d game development. Apress, 2013.

[15] 11 Pros & Cons to Know Before Choosing Unity 3D - GreatSoftLine.com", GreatSoftLine.com, 2020. [Online]. Available: https://www.greatsoftline.com/11-pros-cons-to-know-before-choosing-unity-3d/. [Accessed: 25- May- 2020].

[16] 5 Rarely Known Advantages And Disadvantages Of Unity Game Development | Potenza Global Solutions", Potenzaglobalsolutions.com, 2020. [Online]. Available: https://www.potenzaglobalsolutions.com/blogs/5-rarely-known-advantages-and-disadvantages-of-unity-game-development. [Accessed: 25- May- 2020].

[17] Buyuksalih, Ismail, Serdar Bayburt, Gurcan Buyuksalih, A. P. Baskaraca, Hairi Karim, and Alias Abdul Rahman. "3D Modelling and Visualization Based on the Unity Game Engine–Advantages and Challenges." ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences 4 (2017): 161.

[18] Doran, John P. Unity Game Development Blueprints. Packt Publishing Ltd, 2014.

[19] 4 essential skills for Unity developers", Pluralsight.com, 2020. [Online]. Available: https://www.pluralsight.com/blog/software-development/4-skills-unity-devs. [Accessed: 25- May- 2020].

[20] Shukla, Sakshi, Rayan Dutta, and Rudrani Mishra. "GAME DEVELOPMENT: PROBLEMS."

[21] Jensen, Bjarne Fisker, and Jacob Boesen Madsen. "CameraTool: Pipeline Optimization for Camera Setup in the Unity Game Engine." (2011).

[22] Kim, AeHyun, and JaeHwan Bae. "Development of mobile game using multiplatform (Unity3D) game engine." International Journal of Intelligent Information Processing 5, no. 1 (2014): 29.

[23] Bond, J. G. (2014). Introduction to Game Design, Prototyping, and Development: From Concept to Playable Game with Unity and C. Addison-Wesley Professional.

[24] Oak, Ji Won, and Jae Hwan Bae. "Development of smart multiplatform game app using UNITY3D engine for CPR education." International Journal of Multimedia and Ubiquitous Engineering 9, no. 7 (2014): 263-268.

[25] Messaoudi, Farouk, Gwendal Simon, and Adlen Ksentini. "Dissecting games engines: The case of Unity3D." In 2015 international workshop on network and systems support for games (NetGames), pp. 1-6. IEEE, 2015.

[26] Raffaillac, Thibault, and Stéphane Huot. "Polyphony: Programming Interfaces and Interactions with the Entity-Component-System Model." Proceedings of the ACM on Human-Computer Interaction 3, no. EICS (2019): 1-22.

[27] Indraprastha, Aswin, and Michihiko Shinozaki. "The investigation on using Unity3D game engine in the urban design study." Journal of ICT Research and Applications 3, no. 1 (2009): 1-18.

[28] Nurym, Nurdaulet, Raushan Sambetova, Muhit Azybaev, and Nurassyl Kerimbayev. "Virtual Reality and Using the Unity 3D Platform for Android Games." In 2020 IEEE 10th International Conference on Intelligent Systems (IS), pp. 539-544. IEEE, 2020.

[29] Sung, Kelvin, and Gregory Smith. Basic Math for Game Development with Unity 3D. Apress, 2019.

[30] Wenar, Leif. "The Development of Unity." Journal of Human Development and Capabilities 21, no. 3 (2020): 211-222.

[31] Kraus, Sascha, Matthias Breier, and Sonia Dasí-Rodríguez. "The art of crafting a systematic literature review in entrepreneurship research." International Entrepreneurship and Management Journal (2020): 1-20.

[32] Andrade, António. "Game engines: a survey." EAI Endorsed Transactions on Serious Games 2, no. 6 (2015).

[33] Borg, Markus, Vahid Garousi, Anas Mahmoud, Thomas Olsson, and Oskar Stalberg. "Video Game Development in a Rush: A Survey of the Global Game Jam Participants." IEEE Transactions on Games (2019).

[34] Malete, Thabo N., Kabo Moruti, Tsaone Swaabow Thapelo, and Rodrigo S. Jamisola. "EEG-based Control of a 3D Game Using 14-channel Emotiv Epoc+." In 2019 IEEE International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation and Mechatronics (RAM), pp. 463-468. IEEE, 2019.