Significance of State-of-Art Search Engine in Game Development

Ritu Dewan
Department of Computer Science and Engineering
Galgotias College of Engineering and Technology
Greater Noida, India
dewanritu22@gmail.com

Rohith Polishetty
Sr. software engineer, Migo India limited.
Hyderabad, India.
rohith.polishetty@gmail.com

Naveen Jagadam
CEO, N2force Technologies Inc.
Naperville, IL, USA 60563
Naveen@n2force.com

Mayank Kumar Goyal
Department of Computer Science & Engineering
School of Engineering & Technology
Sharda University
Greater Noida, India
mayankrkgit@gmail.com

Kiran Kumar Ravulakollu
School of Technology
Woxsen University
Hyderabad, India
kiran.r@woxsen.edu.in

Bhagwati Sharan
School of Engineering and Sciences
Department of Computer Science & Engineering
SRM University-AP
Amaravati, India
Bhagwati.sharan001@gmail.com

***Abstract*— Video game development process is different from other software projects while yet being related to them in software engineering. It is distinctive in that it incorporates the work of teams from several fields (art, music, acting, programming, etc.), and that it seeks to create a compelling gaming experience by using prototypes and iterations. This research study focuses on game development and how it is accomplished utilizing a game engine. The process of creating/developing games from a concept is known as game development. Game engines are software programs that help game developers create games. The literature study discusses about the various game engine. The utility of the "Unity" gaming engine is investigated in this paper. Unity may be utilized in different fields as study subjects, but there can be certain frequent benefits & obstacles of utilizing it in different case studies. Students learn which game engine to use based on their specific requirements.**

***Keywords*— Game Development, Game Engine, Unity, Utility, Software**

I. INTRODUCTION

Game development is the process of creating and releasing video games. Game development is the art of translating a developer's vision or idea into an actual playable game. Individually or as a group, game development is possible. Game creation is more than just a way to make games to play for pleasure; it's also an artform that uses the storylines of games to bring people together [1]. Fictional and real-life storylines are both possible. Game developers are also software developers with strong creativity, cooperation, 3D graphics, rendering, modelling, and sculpting abilities, among other things. Producers, publishers, development teams, designers, artists, programmers, level designers, sound engineers, and testers are all needed in the game production process.

A. History of Game Development

The first ever game was 'Tennis for two'. It was invented by the William Higginbotham in 1950s. The first computer game was 'Space war!'. It is developed by 'Steve Russell' and was released in 1962 [2].

B. The Rise of Computer Games

The second generation of computers appeared in the1980s, and included the Commodore VIC-20 and 64, the MSX series, and others. People began to acquire computers since they were less expensive and more games grew as a result [3]. The government also encourages the usage of computers in the county. Computer games and development grew as a result of this factory.

C. Importance of Game Development

1) Problem Solving Skill

During game creation, you must plan for many scenarios, such as when a player may appear in the game, and how to deal with them using the proper resources. Game creation necessitates problem-solving abilities, or the ability to think like a problem solver, since you will encounter instances when you will need to immerse yourself in the game or place a game character in the real world and then consider a solution to the problem [4]. Java, C++, Python, JavaScript, and more programming languages, as shown in Fig. 1. You can start studying any of them and build problem-solving abilities, as well as tackle fundamental programming problems and data structure and algorithm issues [5].

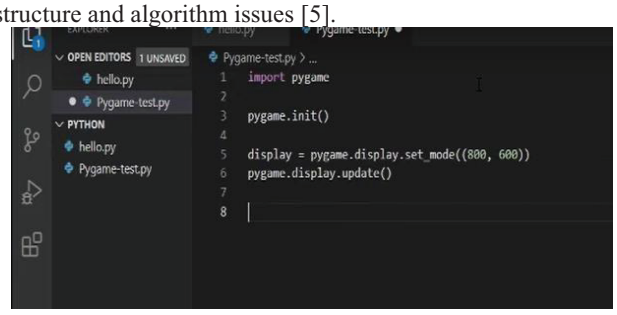

Fig. 1. VSCode Supporting Python

2) Growth Over Years

A software game is a program in which one or more people use resources to make decisions or choices in order to attain a certain objective. The importance of the game production business may be shown in the fact that software games earned three times as much as any other software in 2012. Every year, the video game business expands, with sales rising 13% annually to $6.7billion in 2016 [6]. This demonstrates that gaming technology is becoming more widely available to individuals of all ages and is being welcomed. Game production is a difficult process that spans numerous disciplines, while being one of the fastest-growing industries.

3) Team-Working Skills

Heavy-games cannot be made by a single person; they require a team of people to work on various aspects such as design, graphics, and music. For a machine to run well, all of its pieces must communicate with one another. Otherwise, the mechanism would come apart. As a result, both the creation and play of the game necessitate collaboration [7].

4) Self-Learning

You may turn your ideas into games and make an interactive résumé with them. When compared to a typical resume, this provides you more control over your initial impression in an interview and allows you to standout. You don't need to go to college to learn how to makegames. You may go at your own speed when learning game creation. Do you have any idea? One of the most significant skills that every successful person have is the ability to self-learn [13]. Almost everything in a game is created from scratch, including the characters, plot, format, and structure, among other things. Many skills, such as how to build game assets, how to pick or design the correct sound effects for a game, and how to tell your game plot, may be learned on a game development tour. You may use them to perform a variety of professions, including tester, designer, artist and programmer. To horn yourself in today's technological world, you'll need more than one ability. You may also discover how the game will operate and how it will flow from beginning to conclusion, shown in Fig. 2 [8].

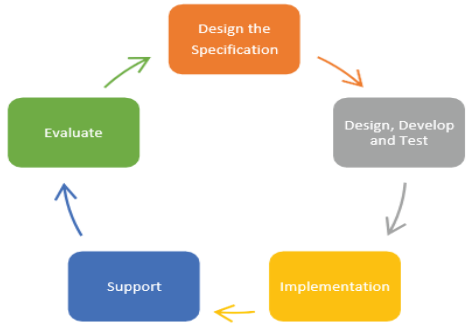

Fig. 2. Life Cycle of Game Development

5) Story/idea

You also learn about the game's design in addition to programming. You'll learn how to combine design and programming in game production and display them in a real-world setting. It must have nice visuals and be entertaining in order to become a popular game. The atmosphere was designed with considerable care by the developers. Even minor objects like a pen and a grenade get skins in COD, demonstrating how much effort has gone into the game. The power to create new things and our creativity are enhanced through game creation. It should be a one-of-a-kind game to grab attention. The more realistic the game is, the more engaging and endearing it becomes [9].

6) Earn Money

If you know how to design games, you can upload them to the Google Play Store and sell them for a reasonable fee to the general audience.

For that, you may make games. You might also work for a game company. You may earn extra money or pocket money as a college student. Servers and in- game services are how gaming firms make money.

II. Game Engine

It was formerly impossible to construct a game on one's own; it required high-specification systems, a crew, and tools, and making the game realistic was a challenging undertaking. Different tools/software are required for different tasks, however now that 'Game Engines' are available, everything is conceivable [10].

A. Specific Search Engines

We propose a software component description model to improve the performance of search engines and retrieval engines by utilizing categorization information (SCDM).

For identifying software components, this paradigm uses two different categorization schemas. The faceted categorization is the basis for one classification scheme. The systematic categorization is the basis for the other classification scheme A search engine-friendly categorization system is used.

According to this standard, the component developer's component descriptor allows re-user to search for components using a search engine [11].

The term "game engine" refers to a set of tools that enable game creators to make games more quickly and easily by allowing them to utilize commands that previously required a lot of code via the game engine's plug-in or asset provider. Previously, game play was constrained by hardware limitations. The first arcade games were developed in 1966, and one of the most well-known arcade games, Pong, was presented. This game took use of advancements in arcade game technology that were brought to consoles. Around 1990, as technology advanced, the game was converted to the PC platform.

Fig. 3. Different Game Engines

The phrase "game engine" was coined about this period. Torque 3D, UDK, Unity3D, and Cry Engine are just few of the game engines that have been launched since then. As shown in Fig. 3. Lewis and Jacobson proposed the design of a game engine.

B. Unity

Unity Technologies is a company that creates a cross-platform gaming engine. Unity was first launched on June 25, 2005. For the game's creation, Unity used the C# programming language, shown in Fig. 4 [12].

C. Unreal

The Unreal Engine was created by Tim Sweeney. Epic Game Technology was founded by Tim Sweeney, and in 1998, they released Unreal Engine. For game creation, Unreal Engine uses C++. If you aren't acquainted with C++, the Unreal Engine offers the Blueprint capability. With the use of Blueprints, a developer may create a game without coding, shown in Fig. 4 [13].

Fig. 4. Unity & Unreal Game Engine

III. Literature Survey

Computer and video games have grown highly popular in the lives of children and teenagers, and they play an important part in their culture. Games may now be played on computers, smart phones, game consoles (mobile and stationary), set-top boxes, and other digital devices in technology-rich locations. It is thought that the natural drive that young people have for games may be merged with educational material and objectives to create what Prensky refers to as "digital game-based learning" as a result of this phenomena.

Aside from the prevalence of games in the lives of young students, game production technology has progressed and become more advanced than previously. The entire game creation process may be separated into different areas and jobs based on existing game development technologies, such as game programmers, 3D models developers, game designers, composers, animators, and play writers, shown in Fig. 5.

A. The Torque Game Engine (TGE)

Garage Games publishes TGE, a gaming engine. Torque began as a firm known as Dynamic, but after purchasing the tribes 2 source codes from Source, the name was changed to Torque 3D. The engine's primary script is written in C++, which allows for easier modification by the user. Torque 3D supports PhysX, powerful delayed lighting mode, and contemporary shader capabilities. TGE is available for Windows, Mac OS X, and Linux. TGE also has guidance for those that want to make and build their own games. This game engine's database utilizes Torque script and supports MySQL [14].

B. Unreal Development Kit (UDK)

Epic Games Inc. created UDK, which is a free Unreal Engine 3 development kit. Professional companies and independent developers have utilized it to build well-known games like Gears of War, Xmen Origins, and others. UDK offers the same features as the premium edition to developers.

The main script in the UDK is Unreal Script, which enables for visual scripting using Unreal Kismet. UDK provides variety of features, including skeletal animations [14].

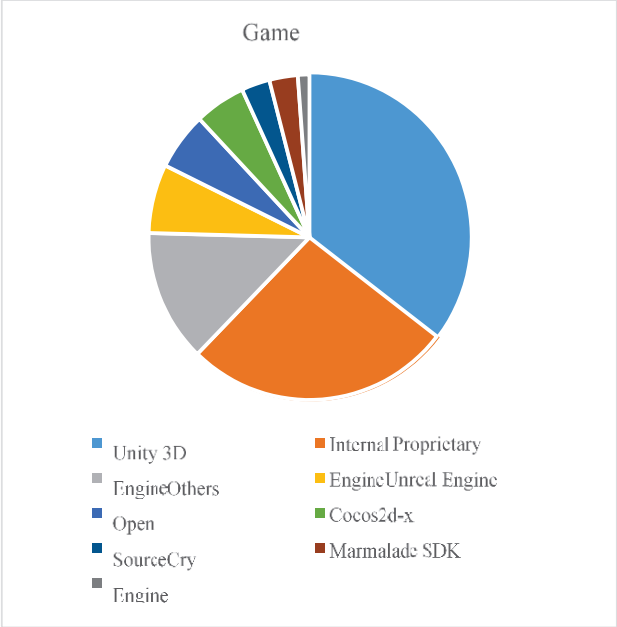

Fig. 5. Best chosen game engine 2022

IV. Unity

Unity is a fantastic tool for creating interactive visualizations and prototypes for games. In this section, we go have discussed the almost everything you need to know to start working with Unity.

A. Workflow and ecosystem

The Unity editor allows you to create apps using scenes. Scripts may be used to add various 3D models, textures, and application logic in scenes. Game objects are the basic building components of a scene. A gaming object may be compared to a container that holds various 3D models and other content. Components can be added to a game object to change it. Components are usually scripts that change how a game item behaves [15].

B. License

"Unity" apps are freely available software, and this technology does not collect any fees for software creations done by using the editor "[Unity Technologies, 2021h]". The developer owns patent to the program, among other things who created software using editor "[Unity Technologies, 2021i]".

Unity Technologies requires a license to publish software created using Unity [Unity Technologies, 2021f]. Users can choose from four different licensing tiers, as shown in Table1.

A. Challenges

By using various file formatting tools more efficiently data can be saved in various types. Files describing 3D models, for example, might be in a variety of file formats. Software that uses the same type of data may be unable to read the same file formats in some cases [16]. Unity 2021d supports a variety of model file types from a variety of tools, primarily mainstream modelling packages. Various file formatting tools are typically used in specialist.

B. Documentations

Documentation is essential in software development since it serves as a user guide for a particular technology. Tutorials and sample projects serve to fill in the gaps in the documentation and make it more understandable. In a few case studies, these aspects were explored.

Unity was not created with a specific game in mind, giving them a "blank slate" to work with while creating their simulation. They contrast Unity with Unreal Engine, claiming that the latter was created with first-person shooters in mind.

V. Case Study

This section provides an overview of the case studies that are part of this systematic literature review. Bioinformatics, engineering and information sciences, and architecture and environmental modelling, and were the three areas in which the works were categorized. The results are shown in Table2.

A. Bioinformatics

This study includes visualization, modelling of biology data and simulation and other compounds, are included in the bioinformatics category.

- C. M. Nakano et al. [17] present iBET, a VizBET extension for simulating and visualizing electron-transfer.
- E. M. Ratamero et al. [18] a software pipeline that describes and allows protein structure to be embedded into VR applications using composition of open-source software, conventional hardware & a few custom-written scripts.
- M. Wiebrands et al. [19] created a bio molecular visualization tool for use with Curtin University's HIVE Cylinder, a 3- meter-high 180-degree circular 3D screen.
- BioVR, a virtual reality platform for visualizing and analyzing protein sequences, RNA, DNA structure is presented by B. Turhan et al. [20].

TABLE I. Price of Unity

NAME	PRICE	REVENUE Allowed in last 12month
PRIVATE/PARTICUL AR	Free	(Less than) $100 000
ADD	$399 per yr.	(Less than) $200 000
EXPERIENCED	$1800 per yr.	(More than) $200000
ENDEAVOUR/PURS UIT	$200 per month.	(More than) $200000

B. Environmental & Architecture model

The studies in this category features initiatives that use elements of architectural and environmental modelling to some level.

Some researcher had discussed the benefits and drawbacks of utilizing “Unity” in three- D model and visualization for 2-projects: Istanbul underground utility mapping & the solar Vesuvius explosion, and building of effective replica of a Pompeian dwelling.

C. Engineering & information sciences

- Unity is mostly used in three different phases of engineering & information sciences fields i.e., simulate environmental phenomena & mechanical.
- J. C. Dai et al. [21] investigate, use of client-level haptic tools for the cost-effective training option as the development of medical- based surgical skills.

D. Advantages of “Unity”

This section summarizes the benefits & drawbacks of using it in academic work based on the case study. The majority of research failed to indicate any benefits or drawbacks. As a result, data collecting necessitated interpretation.

The benefits and problems were addressed in three different ways in the case studies. To begin, several studies detailed their Unity experiences, generally in the form of a list of benefits and drawbacks. These studies didn't require much interpretation, and the data could be used straight away. Second, the majority of research stated, Unity was taken as the case study.

The reasons were viewed as Unity's benefits. Finally, several surveys contained comments about specific aspects & property of Unity that may be taken as either good or reject feedback. These were regarded as opportunities and obstacles, accordingly.

E. Features-Built-In

This paragraph focus on Unity's built-in functionality. These include Unity's editor capabilities as well as standard game engine components like graphics, audio, and scene-construction. It's fair to infer that all of the researching this literature review utilized them in some way, even if they weren't explicitly specified.

According to T. Nieminen et al. the Unity engine was chosen for visualization since it has various built-in features that benefits heat in urban land research & complex scene creation [22]. The built-in capabilities of Unity, according to E. M. Ratamero et al. allow users without programming skills to create their ideas with minimum training [23].

F. V-R Capabilities

Unity was utilized to create VR apps in three of the studies that used it. Unity was chosen by C. M. Nakano et al. [17] because of its compatibility the Oculus interface. E. M. Ratamero et al. [23] also used Oculus goods to create the program. All of the research in this area are from the discipline of bioinformatics, indicating that the field is experimenting with novel.

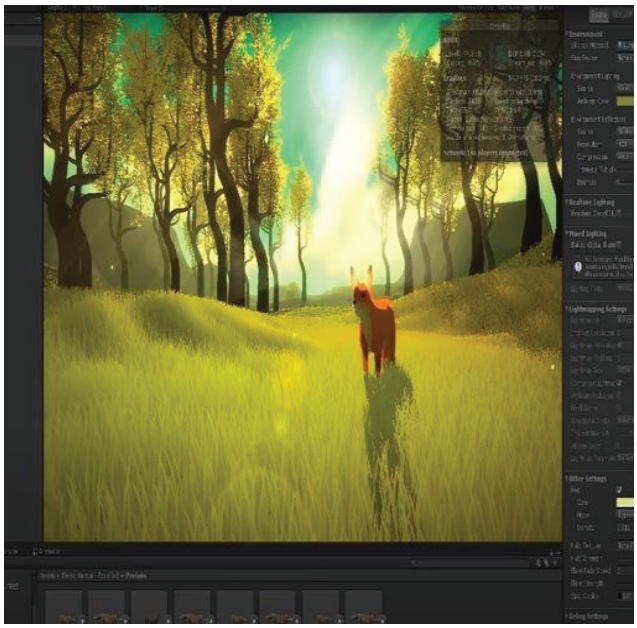

Fig. 6. Showing VR Working of UNITY

Visualization approaches with the HIVE Cylinder display, P. C. Patil et al. unveiled a novel molecular data visualization tool a bioinformatics visualization project, shown in Fig. 6 [24].

G. Coordinate System

Every software that deals with 2D & 3D data uses coordinate systems. However, they may be discrepancies in how each software implements the coordinate system, such as how the coordinate axes are named. Unity, according to C. Y. Kuo et al., employs a coordinate system in which the z and x axes constitute the plane surface, with the y axis pointing up, shown in Fig. 7 [25].

Fig. 7. Showing Working and capabilities of Unity

VI. Conclusion and Future Scope

Finally, the literature review responds to research issue by offering applicable case studies from a variety of sectors, as well as analyzing and reporting on the expertise with “Unity”. This paper's findings might be relevant to program searching for a software-based visualization, modelling platform, or simulation.

Unity is a 3D and 2D application development platform, most notably for video games. Unity offers a large variety of operations in numerous areas of study, such as visualization and engineering simulations, as demonstrated in this work. In certain case study, “Unity” supports exterior benefits VR, deployment, as well as cross-platform, was cited as crucial. The future scope of this research paper can be extended by gathering information about game development and knowing all game engines & doing comparisons between them.

References

[1] C. M. Kanode and H. M. Haddad, “Software Engineering Challenges in Game Development,” 2009 Sixth International Conference on Information Technology: New Generations. IEEE, 2009. doi: 10.1109/itng.2009.74.

[2] X. Zeiler and S. Mukherjee, “Video Game Development in India: A Cultural and Creative Industry Embracing Regional Cultural Heritage(s),” Games and Culture, vol. 17, no. 4. SAGE Publications, pp. 509–527, Sep. 24, 2021. doi: 10.1177/15554120211045143.

[3] Y. H. Kim, J. Naurright, and C. Suveatwatanakul, “The rise of E-Sports and potential for Post-COVID continued growth,” Sport in Society, vol. 23, no. 11. Informa UK Limited, pp. 1861–1871, Sep. 14, 2020. doi: 10.1080/17430437.2020.1819695.

[4] D. Topalli and N. E. Cagiltay, “Improving programming skills in engineering education through problem-based game projects with Scratch,” Computers & Education, vol. 120. Elsevier BV, pp. 64–74, May 2018. doi: 10.1016/j.compedu.2018.01.011.

[5] R. Wang, S. Yang, Y. Yan, Y. Tian, and P. Wang, “Internet Gaming Disorder in Early Adolescents: Gender and Depression Differences in a Latent Growth Model,” Healthcare, vol. 9, no. 9. MDPI AG, p. 1188, Sep. 09, 2021. doi: 10.3390/healthcare9091188.

[6] A. Fatima, T. Rasool, and U. Qamar, “GDGSE: Game Development with Global Software Engineering,” 2018 IEEE Games, Entertainment, Media Conference (GEM). IEEE, Aug. 2018. doi: 10.1109/gem.2018.8516498.

[7] E. Riivari, M. Kivijärvi, and A.-M. Lämsä, “Learning teamwork through a computer game: for the sake of performance or collaborative learning?,” Educational Technology Research and Development, vol. 69, no. 3. Springer Science and Business Media LLC, pp. 1753–1771, Jun. 2021. doi: 10.1007/s11423-021-10009-4.

[8] R. Ramadan and Y. Widyani, “Game development life cycle guidelines,” 2013 International Conference on Advanced Computer Science and Information Systems (ICACSIS). IEEE, Sep. 2013. doi: 10.1109/icacsis.2013.6761558.

[9] D. Vlachopoulos and A. Makri, “The effect of games and simulations on higher education: a systematic literature review,” International Journal of Educational Technology in Higher Education, vol. 14, no. 1. Springer Science and Business Media LLC, Jul. 10, 2017. doi: 10.1186/s41239-017-0062-1.

[10] A. Andrade, “Game engines: a survey,” EAI Endorsed Transactions on Game-Based Learning, vol. 2, no. 6. European Alliance for Innovation n.o., p. 150615, Nov. 05, 2015. doi: 10.4108/eai.5-11-2015.150615.

[11] C. Seacord, A. Hissam, K. Wallnau, “Agora: a search engine for software components” Internet Computing, Vol. 2 No. 6, pp. 62-70, December 1998.

[12] P. P. Patil, R. Alvares, “Cross-platform Application Development using Unity Game Engine”, International Journal of Advance Research in Computer Science and Management Studies, Volume 3, Issue 4, April 2015

[13] X. Zhang, J. Liu, Qing Chen, H. Song, Q. Zhan, and J. Lu. A 3D virtual weft-knitting engineering learning system based on Unreal Engine 4. Computer Applications in Engineering Education, 26(6):2223–2236, 2018. DOI: arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/cae.22030, doi:10.1002/cae.22030.

[14] T. Hinrichs and B. Buth, "Can AI-based Components be Part of Dependable Systems?," 2020 IEEE Intelligent Vehicles Symposium (IV), 2020, pp. 226-231, doi: 10.1109/IV47402.2020.9304740.

[15] Craighead, J., Burke, J., & Murphy, R. (2008, September). 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) (Vol. 4552).

[16] M. M. Wijaya, "THE UNITY OF SCIENCE PARADIGM, CHALLENGES, AND SOLUTIONS IN PANDEMIC ERA," Living Islam: Journal of Islamic Discourses, vol. 4, no. 1. Al-Jamiah Research Centre, May 25, 2021. doi: 10.14421/lijid.v4i1.2561.

[17] C. M. Nakano et al., "iBET: Immersive visualization of biological electron-transfer dynamics," J. Mol. Graph. Model., vol. 65, pp. 94–99, 2016.

[18] K. Xu et al., "VRmol: an integrative web-based virtual reality system to explore macromolecular structure," Bioinformatics, vol. 37, no. 7. Oxford University Press (OUP), pp. 1029–1031, Aug. 03, 2020. doi: 10.1093/bioinformatics/btaa696.

[19] M. Wiebrands, C. J. Malajczuk, A. J. Woods, A. L. Rohl, and R. L. Mancera, "Molecular Dynamics Visualization (MDV): Stereoscopic 3D Display of Biomolecular Structure and Interactions Using the Unity Game Engine," Journal of Integrative Bioinformatics, vol. 15, no. 2. Walter de Gruyter GmbH, Jun. 21, 2018. doi: 10.1515/jib-2018-0010.

[20] B. Turhan and Z. H. Gümüş, "A Brave New World: Virtual Reality and Augmented Reality in Systems Biology," Frontiers in Bioinformatics, vol. 2. Frontiers Media SA, Apr. 06, 2022. doi: 10.3389/fbinf.2022.873478.

[21] J. C. Dai, T. S. Lendvay, and M. D. Sorensen, "Crowdsourcing in Surgical Skills Acquisition: A Developing Technology in Surgical Education," Journal of Graduate Medical Education, vol. 9, no. 6. Journal of Graduate Medical Education, pp. 697–705, Dec. 01, 2017. doi: 10.4300/jgme-d-17-00322.1.

[22] Nieminen, Topi. "UNITY GAME ENGINE IN VISUALIZA-TION, SIMULATION AND MODELLING." (2021).

[23] E. M. Ratamero, D. Bellini, C. G. Dowson, and R. A. Römer, "Touching proteins with virtual bare hands," Journal of Computer-Aided Molecular Design, vol. 32, no. 6. Springer Science and Business Media LLC, pp. 703–709, Jun. 2018. doi: 10.1007/s10822-018-0123-0.

[24] P. C. Patil and T. Batra, "Virtual reality in bioinformatics," Open Access Journal of Science, vol. 3, no. 2. MedCrave Group, LLC, Mar. 29, 2019. doi: 10.15406/oajs.2019.03.00132.

[25] C. Y. Kuo et al., "The landslide stage of the Hsiaolin catastrophe: Simulation and validation," Journal of Geophysical Research, vol. 116, no. F4. American Geophysical Union (AGU), Oct. 20, 2011. doi: 10.1029/2010jf001921.