iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** IV **Month of publication:** April 2026

DOI: https://doi.org/10.22214/ijraset.2026.79685

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

Development of an Interactive 3D Virtual Campus for Educational Navigation

Yash Waghamare, Yash Dighade, Sujal Patil, Raghav Deshmukh, Shantanu Sinkar, Srikant Akarte
Department of Computer Science and Engineering, Prof. Ram Meghe Institute of Technology and Research (PRMITR) Sant Gadge Baba Amravati University, Amravati, India

Abstract: ***The speedy progress of computer graphics and simulation technologies will be enabling the growth of mutual virtual environments for educational applications. This paper represents a 3D virtual model of our college campus and its area made to give a view by using our appealing platform to explore campus view in virtual mode. The system developed using the Unity 3D game engine using C# scripting, and 3D model is being created using tools such like Blender and SketchUp by us.***
The system offered by us allows viewers to navigate and explore the campus environment, its view, buildings as per their multiple perspectives and give a clear grasp of the campus layout. In comparison to other traditional static maps, or interfaces our system offers a more engaging and informative experience by using interaction and visualization.
This our model can be acting as a tool for students, visitors and stakeholders who wants overlook of campus and can helps in college promotions, student orientation and educational presentation. This work shows a merging of 3D visualization and engaging technologies for displaying environments.

I. INTRODUCTION

The presentday improvement in computer graphics and visualization technology has been outstandingly change the growth of interactive virtual environments. Technologies such as 3D modeling and virtual simulation are globally used amongst domains that include gaming, architecture, education and urban planning. These technologies help users to interact with nearly realistic digital environments and improving both understanding and engagement.

In the field of educational institutions, the use of digital solutions is increasing and adopt for improving accessibility as well as communication of infrastructure and services. Large campus frequently present practice of navigation challenges for new students and visitors. As traditional methods like signboards, printed maps, and physical tours which providing limited spatial understanding and lack of interactivity. Moreover, there is a growing need for remote accessibility allowing users to explore institutional facilities without being physically present.

To attend these types of challenges, our work provides an interactive 3D virtual campus application for our college. The system targets in providing a real digital presentation of the campus enabling users to explore buildings, departments and facilities through a deeply engaging virtual environment. The solution is developed using the Unity 3D game engine with C# scripting while 3D models are created by us using tools such as Blender and SketchUp.

The propose of this system allows users to navigate freely within the virtual campus, view infrastructure from multiple perspectives, and access relevant information about different locations. Comparing to traditional static representations, the system offers a more engaging and informative experience by combining real-time rendering with interactive navigation.

The primary objective of this work is to design and develop a user-friendly virtual campus model that improves accessibility and enhances user understanding of campus layout. The system
also demonstrates the practical application of computer graphics, game engine technologies, and interactive simulation in solving real-world problems.

Furthermore, the proposed approach contributes to digital transformation in education by providing a scalable and effective platform for campus exploration. It can be utilized for student orientation, institutional promotion, and educational demonstrations, thereby improving the overall user experience and institutional outreach.

II. RELATED WORK

The development of such type of environments and 3D visualization systems has been gained major attention in past few years because of development in computer graphics, simulation technologies and interactive systems.

©IJRASET: All Rights are Reserved

Researchers and developers have extensively exploring the use of 3D model and game engine for creating a virtual representations of real-world environments such as campuses, cities, and architectural structures. These technologies giving immersive and interactive experiences that enhance users understanding of spatial layouts and infrastructure.

On the word of Stallings [1], newly visualization technologies allows the creation of greatly realistic digital environments by using advanced type of rendering techniques and graphical processing methods. These technologies are being used widely applied in educational simulations, architectural visualization, and virtual tours.

Similarly according to Gregory [2] highlights the importance of game engine an architecture in building interactive 3D environments. Game engines type of Unity and Unreal Engine giving features like real-time rendering, physics simulation, animation systems as well as scripting support which are making them greatly suitable for simulation-based applications.

As Hocking [3] says the using of the Unity game engine for making interactive applications and virtual environments. Unity giving cross-platform compatibility in addition to advanced graphical tools along with scripting support through C# that which making it a largely popular option for 3D application development. Similar as per, Adams [4] highlighting how C# scripting used in Unity making navigation systems together with camera control along with object interaction in virtual environments.

The creation of a realistic 3D modeling is also an another essential aspect of virtual an environment development. Marschner and Shirley [5] explaining as that modern computer graphics techniques, along with tools like Blender and SketchUp, allow developers to design in detailed and more accurate representations of real-world objects by the help of textures, lighting, and material properties.

In this field of educational technology virtual campuses systems is been widely adopting for improve accessibility and to provide the digital campus tours. These systems allowing users to exploring buildings, departments, and facilities interactively without being physically visiting or presenting in the the campus.

In addition this, research in Augmented Reality (AR) and Virtual Reality (VR) demonstrating that immersive technologies more improve user engagement and learn experiences [6]. Such technologies can be valuably integrated into virtual campus systems to providing a more real and interactive experience.

On the basis of the analysis of existing research, it is evidences that the integration of 3D modeling tools, game engines, and interactive programming techniques provides a strong foundation to made virtual campus systems. These technologies enable the creation of realistic, interactive, and user-friendly environments.

Hence this system applies these concepts and technologies to develop an interactive virtual campus. This system is integrates 3D modeling, real-time rendering and user interaction features to deliver an immersive campus exploration experience.

III. PROPOSED SYSTEM METHODOLOGY

A. System Overview

The proposed system is a 3D virtual model of the PRMITR College campus. It is made to show the college infrastructure in a digital way so that users can explore it easily. Users can move around the campus in a virtual environment and see different places, which makes it feel more real and interactive. The system uses 3D models and real-time rendering along with user interaction to make navigation simple.

For developing the system, Unity 3D is used because it helps in creating interactive environments. C# is used for writing the scripts, mainly for movement and user controls. This allows users to move smoothly inside the virtual campus and explore different areas without much difficulty.

B. Methodology Framework

The development of the system is done in different steps:

1) Data Collection and Analysis

In this step, we collect information about the campus layout and building details. We also identify important infrastructure and understand what the user needs for navigation and interaction.

2) 3D Modeling

Here, 3D models of the campus are created using Blender or SketchUp. Textures and materials are added to improve the look, and we try to keep the models similar to real structures.

After that, the models are imported into Unity. They are placed according to the real campus layout. Extra things like trees, roads, lighting, and shadows are also added to make it look better.

3) Navigation and Interaction Design

In this part, user movement is implemented using C#. Camera controls like rotation and zoom are added. Users can move around easily and also interact with objects.

4) Information Integration

Buildings are connected with some information. When a user clicks or interacts, the related details are shown.

5) User Interface Design

Simple UI is designed so that users can use the system easily. Menus and controls are added in a basic way.

6) System Integration

All parts are combined together to form the complete system. It is checked that everything works properly.

7) Testing and Validation

Finally, the system is tested. Navigation, performance, and user experience are checked, and errors are fixed.

C. System Architecture

The system works in three basic parts.

First is the input part, where the user uses keyboard and mouse to move around.

Second is the main part where Unity and C# handle whatever input is given and decide what to do next, like movement or showing things.

Last is the output part, where everything is displayed on the screen as a 3D campus.

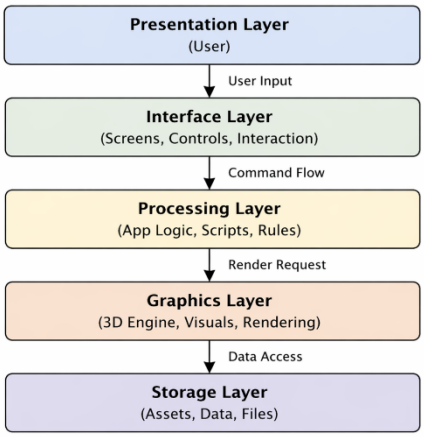


Figure 1: High-Level System Design

D. Module Design

The 3D modeling module is used to create the structures of the campus.

The scene development module is used to build the environment in Unity.

The navigation module handles user movement and camera control.

The interaction module detects objects and responds when the user interacts.

The information module is used to show details of buildings.

The user interface module manages how the user interacts with the system and its controls.

E. Algorithm

Virtual Campus Navigation Algorithm:

1) Start the system and load the Unity environment
2) Load the 3D models and set up the virtual scene
3) Add textures, lighting, and other environment settings
4) Take input from the user through keyboard or mouse
5) Use C# scripts to handle the input
6) Update the camera position and movement based on input
7) Check if the user is interacting with any object

8) Show the related information if interaction happens
9) Keep rendering the scene continuously
10) Stop the system when the user exits

F. Justification of Methodology

The method that is used in our project is depend on 3D modeling with a game engine for building the system. Instead of static methods this method allows users to interact in real time that's why it becomes easier to understand the campus layout.
Unity is used because it works well for rendering and running the system smoothly. The design is kept modular so that changes can be made easily later. Overall, this method is useful for building simple interactive systems for educational purposes.

IV. WORKFLOW

The workflow simply shows how the system is built and how it runs step by step. It starts from taking user input and then shows how that input is used to display the 3D campus on the screen.
Step 1: 3D Model Creation
The work starts by making 3D models of the campus. For this, Blender or SketchUp is used. Different parts like buildings, roads and trees are included in the models. The idea is to make them look similar to the real campus.
Step 2: Importing Models into Unity
After making the models, they are put into Unity. Unity is used as the main place where the system is built. All the files are then arranged properly so it becomes easier to use them later.
Step 3: Scene Development
In Unity, the campus is built step by step. Buildings are placed as per the real layout. Then textures and materials are added. Lighting and shadows are also set. Some surroundings are added so it looks a bit real.
Step 4: Navigation and Camera Setup
C# is used for navigation. With this, the user can move around the campus. The camera can also be rotated, and zoom in or out when needed.
Step 5: Interaction Handling
The system also keeps checking user actions. When the user goes near a building or clicks on it, the system understands which object it is and then does the required action.
Step 6: Information Display
Details about the buildings are shown to the user. This can include things like department name, its purpose, and the facilities available.
Step 7: Rendering the Environment
Unity handles everything and shows the final scene. The graphics are displayed in real time. The view keeps changing based on what the user does.
Step 8: User Interface Control
UI is also there to guide the user. It shows things like instructions, controls for movement, and some menu options.
Step 9: Continuous System Execution
The system runs continuously. It takes user input, does the required processing, and updates what is shown. This keeps happening, so the user can keep moving around the campus.
Step 10: System Exit
It ends when the user exits the app.

V. ANALYTICAL DISCUSSION

This part is about building the 3D model of PRMITR College into a working system. The idea was turned into an application where users can explore the campus. Unity, C#, and modeling tools were used to create the environment, so it looks close to the real one.

1) Analysis of Implementation Approach

The system was developed by the use of a modular approach also it worked quite well. By dividing it into parts such as 3D modeling, scene setup, navigation, interaction and user interface made things easier to manage and debug.
The 3D modeling part helped in making the structures display accurate but also it took a lot of time and effort.

The scene integration part was used to bring all the assets together though some performance issues came when handling bigger models.

The navigation part allowed smooth movement that is important for the user experience.The interaction part made the system more useful by showing information, not just visuals.The user interface part made it easier for normal users to use the system without much difficulty.

Overall dividing the system like this made it easier to maintain and also allowed future improvements.

2) Performance Evaluation

The system works in proper manner on normal hardware. The 3D view in Unity was smooth and movement was also fine most of the time. While interacting with objects, there was not much lag, especially when the models were simple.

Though performance is not always same. It depends on how complex models are as well as how much lighting is used and the system hardware. If models are more detailed or lighting is too much then it can be slow down.

To remove or avoid this some of basic optimizations can be used such as reducing extra details, using smaller textures and keeping assets organized.

3) Usability and User Experience Analysis

The system is simple to use and the controls are easy to understand. Users can move using keyboard and mouse without much problem. The campus view is clear, and showing information on interaction helps a bit in learning.

There are some good points in it. Movement is easy and the layout is visible properly. The system is not confusing to use.

But there are also some limitations. New users may need a little guidance at the beginning. There is no guided navigation, so users have to explore by themselves. Also, there is no voice or AI support in the system.

Overall, the system works fine and is easy to use, but it can be improved by adding some extra features later.

4) Effectiveness of Technology Used

The technologies used in this project were suitable for the work. Unity was helpful for building the environment and showing everything properly. C# was used for handling movement and interaction in the system. Blender or SketchUp were used to create the 3D models.

These tools are also easy to access since most of them are free. They can be used for bigger projects later as well. Also, they are commonly used, so it becomes easier to learn and improve further.

5) System Reliability and Testing Analysis

During testing, we checked if all models load properly. Navigation was also working fine and there were no major issues. Interactions were happening as expected. The system ran stable and no crashes were seen.

But testing on low-end systems was not done much. Also, if the scene is large, it can take more time to load.

So overall, the system works fine and is reliable, but it still needs some improvement for better performance on different systems..

6) Results Interpretation

The system was able to meet its main objectives. A 3D campus was created and users can explore it. This helps in understanding the layout better.

From this, it is clear that this is better than normal maps. Users can see the spacing and positions more clearly. It also makes the system a bit more interesting and easier to use.

7) Limitations Identified

No project is perfect, and this one also has some limitations.

There is no VR or AR feature yet. The interaction is quite basic, mostly just clicking or going near objects. Real-time data is also not included. The performance depends on the system hardware, and it is not available on the web.

Still, identifying these points shows that the system can be improved further.

8) Future Improvement Insights

From the analysis, some improvements can be made. VR or AR can be added. AI-based guides can also be included. Better navigation can be given using path guidance. Performance can be improved for low-end systems. It can also be made available on web and mobile.

VI. CONCLUSION

The 3D model of the college project is made to show the campus in virtual way. The main aim is helping users to see the college without going there or being physically present in real life. It shows buildings and other areas in 3D, so users can get the basic idea of how the campus looks.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

While developing this project, we followed steps like understanding the requirement, then designing, coding and testing. These steps helped us to finish the project in a proper manner.

Unity 3D is used to create the environment and C# is used for movement and controls. For making models, Blender or SketchUp is used. Because of this, the result looks close to the real campus, but still not exactly same.

This system users can move around and can see different places of the campus. Also they can look at buildings from different sides which is not possible in by using maps. So it become more easier for understand.

This project also helping us in learning how 3D graphics works in real applications. We also got some type of practice in coding and modeling.

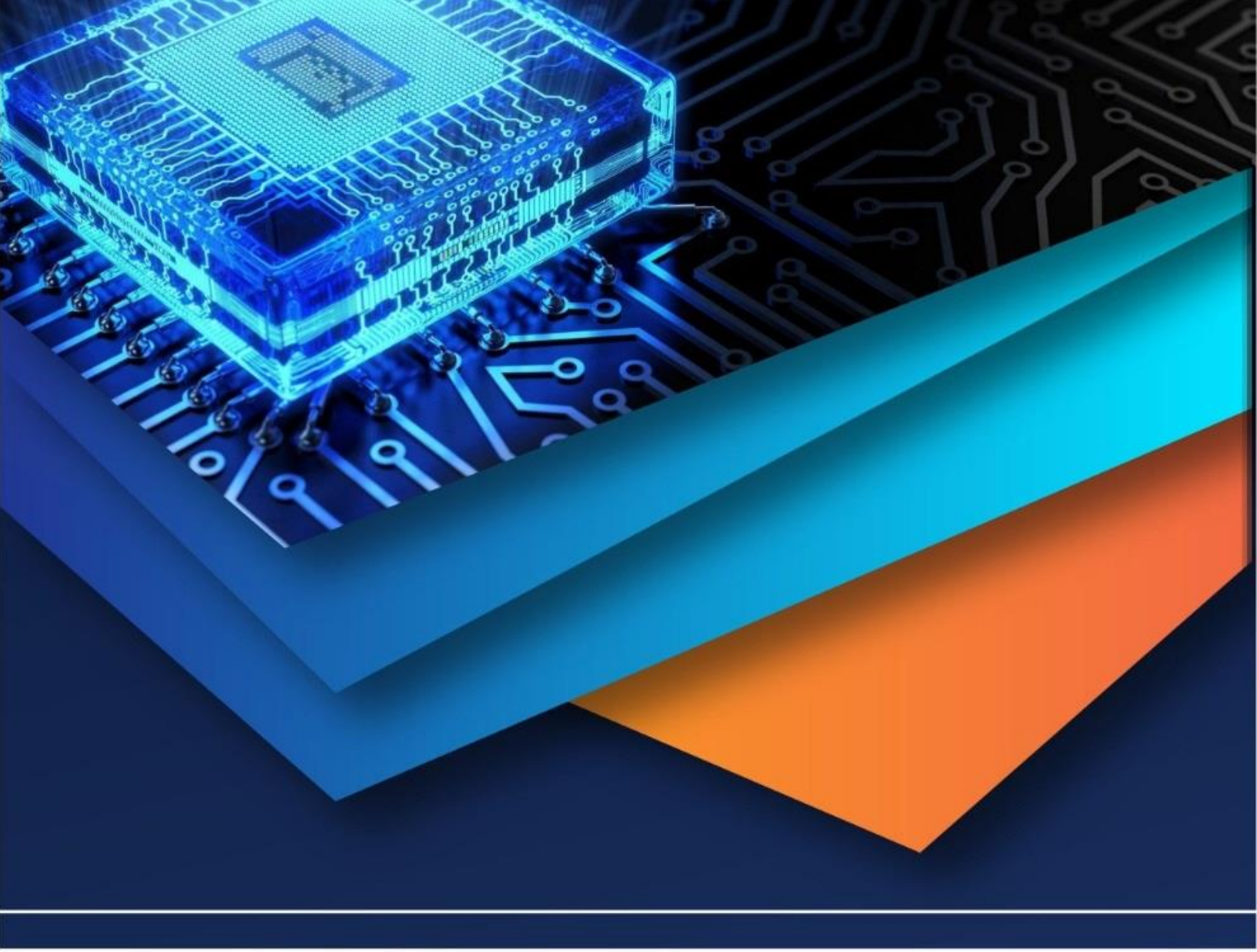
There are some limitations also. It does not include features like VR, guided navigation or online access. These things can be added in future to improve it.

Overall the project is very simple and works perfectly fine. It also give a digital view of the campus and can improved later.

REFERENCES

[1] W. Stallings, Computer Graphics and Visualization Systems. New York, NY, USA: Pearson, 2017.
[2] J. Gregory, Game Engine Architecture, 3rd ed. Boca Raton, FL, USA: CRC Press, 2018.
[3] J. Hocking, Unity in Action: Multiplatform Game Development in C#, 2nd ed. Shelter Island, NY, USA: Manning Publications, 2018.
[4] A. Adams, Learning C# by Developing Games with Unity 2021. Birmingham, U.K.: Packt Publishing, 2021.
[5] S. Marschner and P. Shirley, Fundamentals of Computer Graphics, 5th ed. Boca Raton, FL, USA: CRC Press, 2021.
[6] D. Schmalstieg and T. Höllerer, Augmented Reality: Principles and Practice. Boston, MA, USA: Addison-Wesley, 2016.
[7] T. Akenine-Möller, E. Haines, and N. Hoffman, Real-Time Rendering, 4th ed. Boca Raton, FL, USA: CRC Press, 2018.
[8] D. Shreiner, G. Sellers, J. Kessenich, and B. Licea-Kane, OpenGL Programming Guide, 9th ed. Boston, MA, USA: Addison-Wesley, 2017.
[9] R. Szeliski, Computer Vision: Algorithms and Applications, 2nd ed. Cham, Switzerland: Springer, 2022.
[10] I. Millington, Artificial Intelligence for Games, 3rd ed. Boca Raton, FL, USA: CRC Press, 2019.
[11] B. Furht, Handbook of Augmented Reality. New York, NY, USA: Springer, 2011.
[12] A. Dix, J. Finlay, G. Abowd, and R. Beale, Human–Computer Interaction, 3rd ed. Harlow, U.K.: Pearson, 2004.
[13] J. Vince, Introduction to Virtual Reality. London, U.K.: Springer, 2017.
[14] K. Perlin and E. Hoffert, “Hypertexture,” in Proc. ACM SIGGRAPH Conf., 1989, pp. 253–262.
[15] S. M. LaValle, Virtual Reality. Cambridge, U.K.: Cambridge University Press, 2019.
[16] M. Slater and M. Usoh, “Presence in immersive virtual environments,” Presence: Teleoperators and Virtual Environments, vol. 6, no. 6, pp. 603–616, 1997.
[17] R. T. Azuma, “A survey of augmented reality,” Presence: Teleoperators and Virtual Environments, vol. 6, no. 4, pp. 355–385, 199

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)