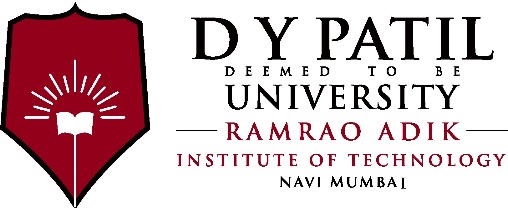
**RAMRAO ADIK INSTITUTE OF TECHNOLOGY**

**DEPARTMENT OF COMPUTER ENGINEERING**



**Project Report On**

**AR In Education**

**( Jet Turbine)**

**Team Members**

1. Om Dange. 21CE1195 TE-F2
2. Sahil Patankar. 21CE1271 TE-E2
3. Chirag Kunder. 21CE1197 TE-F2
4. Prajwal Maske. 21CE1181 TE-F2
5. Siddharth Singh. 21CE1174 TE-F2

Academic Year Even- 2023- 2024

**Index:**

|  |  |  |
| --- | --- | --- |
| Sr No | Particular | Page Number |
| 1. | Introduction | 3 |
| 2. | Background | 4 |
| 3. | Design | 5 |
| 4. | Development | 7 |
|  | 1. Program a basic environment | 7 |
|  | 1. Implement audio special effect | 8 |
|  | 1. Implement Visual effects | 9 |
|  | 1. Implement AR/VR User interfaces | 10 |
|  | 1. Implement AR/VR Animations | 11 |
|  | 1. Iterate on the AR/VR application development | 12 |
| 5. | Results (Screenshots of the project outcome) | 13 |

**1.Indroduction**

Jet turbines stand as marvels of engineering, powering industries ranging from aviation to energy generation. Yet, understanding their intricate workings has often been confined to textbooks and physical models, limiting the depth of comprehension and practical experience attainable. Augmented Reality (AR) technology presents a transformative opportunity to transcend these limitations, offering an immersive and interactive platform for learning and exploration.

This report delves into the creation and utilization of an AR jet turbine simulation—a pioneering endeavor poised to redefine engineering education and training. By seamlessly blending virtual elements with the real world, this simulation brings the complexities of jet turbine dynamics to life, empowering learners to engage with, manipulate, and comprehend these systems in unprecedented ways.

In recent years, Augmented Reality (AR) technology has emerged as a powerful tool for enhancing engineering education and training. This report presents a comprehensive overview of the development and implementation of an AR jet turbine simulation designed to revolutionize the learning experience for aspiring engineers and industry professionals.

**2. Background**

The AR jet turbine simulation offers a multitude of applications across various domains, each contributing to its significance and potential impact. This report provides a comprehensive analysis of the diverse applications of the simulation, highlighting its versatility and value in different contexts.

The primary application of the AR jet turbine simulation lies in engineering education and training. By providing an immersive and interactive learning experience, the simulation enhances comprehension and retention of complex turbine concepts among students and professionals. It serves as a valuable supplement to traditional teaching methods, enabling learners to gain practical skills and insights essential for success in turbine engineering.

Within the aerospace and energy industries, the simulation finds applications in product development and testing. Engineers can use the virtual environment to simulate performance characteristics, identify design flaws, and optimize turbine components and systems before physical prototyping. This accelerates the development process, reduces costs, and enhances the reliability and performance of turbine products.

**3. Design**

The Jet Turbine Package available on the Unity Asset Store is an exceptional resource for developers interested in integrating high-quality, realistic jet engine models and animations into their game or simulation projects. This package is meticulously designed to provide not only visual fidelity but also functional accuracy that can enhance the overall design aspect of aerospace or aviation-themed applications. It includes various models of jet turbines, each featuring detailed textures, accurate scaling, and comprehensive animations that mimic real-world engine behavior. This makes the package particularly valuable for developers looking to create immersive experiences, whether it be in flight simulators, action games, or educational tools that require a deep level of detail and realism.

In addition to its visual assets, the Jet Turbine Package offers extensive customization options, enabling designers to tailor the engines to fit specific requirements or scenarios within their projects. The package typically includes scripts and shaders that work seamlessly within the Unity environment, allowing for adjustments in aspects like thrust power, heat distortion effects, and particle emission, all of which contribute to a more dynamic and interactive design. Moreover, the integration of sound assets that replicate the unique roars and whistles of jet engines adds an extra layer of immersion, ensuring that users not only see but also hear the power of these incredible machines. By leveraging such a comprehensive package, developers can significantly reduce development time and costs while elevating the aesthetic and functional quality of their projects.

To streamline the user interface and enhance user interaction with the AR jet turbine simulation, we can simplify and refine the design by removing the specified components. Here's a description of the interface, focusing on a more integrated and seamless experience:

**Component-Specific Interaction Buttons**

* **Piston Button:** Opens a detailed view of piston-related components. Users can interact with 3D models of pistons, cylinders, and crankshafts, manipulating them to see their functions and relationships within the turbine.
* **Grid Button:** This button leads to the grid component view, focusing on elements like stators and vanes. Users can explore how these components manage airflow and their impact on turbine efficiency.
* **Tubes Button:** Accesses the tubes view, where users can examine conduits, ducts, and pipes. This helps in understanding the routing and management of fluids and gases, essential for efficient turbine operation.
* **Hull Button:** Reveals the hull or casing components of the turbine, allowing users to study its structural and protective features.
* **Pause Button:** Halts all audio like narrations and effects, aiding concentration and discussion on turbine details.
* **Un-pause Button:** Resumes audio playback from where it was paused, ensuring a seamless and continuous learning experience.

**4.Development**

For my augmented reality (AR) project focusing on jet turbines, I utilized Unity as the development platform to create an interactive and educational experience. To ensure high fidelity and accuracy in the visualization of jet turbines, I sourced a detailed model from the Unity Asset Store. This model, which is part of a meticulously curated Jet Turbine Package, includes finely textured components and dynamic animations that simulate real-world turbine behavior. Leveraging Unity's powerful rendering capabilities and the high-quality assets from the store, I was able to craft an immersive AR application that not only educates users about the mechanics and operation of jet turbines but also allows them to explore the complex components in a visually engaging manner. This approach significantly enhanced the user experience, making complex engineering concepts more accessible and engaging to a broader audience.

* **Program a basic environment**

To set up a basic environment for an Android APK app with AR configuration in Unity Hub, follow these simplified steps:

* **Install Unity Hub and Unity Editor:** Include Android Build Support during installation to ensure compatibility with Android devices.
* **Add ARCore SDK:** Import the ARCore SDK for Unity from Google's ARCore SDK GitHub page or the Unity Asset Store to enable AR capabilities.
* **Configure Android Settings:** In Unity, switch the project's target platform to Android and adjust the Player Settings to support the required API level for ARCore.
* **Enable XR Plugin Management:** Activate ARCore in XR Plugin Management within Unity’s project settings to integrate AR functionalities.
* **Set Up AR Components**: Add AR Session and AR Session Origin in your scene, along with other necessary AR management components.
* **Build and Deploy:** Connect an Android device, enable USB debugging, and use the Build and Run option in Unity to compile and deploy the app directly to the device.
* **Implement audio special effect**

To implement audio special effects involving text-to-speech in our project, we first converted the desired text into speech using an appropriate text-to-speech (TTS) tool. This step involved selecting voices and modulating parameters to ensure the audio was clear and matched the intended tone for our application.

Once we had the audio files, we imported them into Unity Hub. To do this, we placed the audio files in a designated folder within our project’s assets directory. In Unity, these audio clips were then brought into the scene by dragging and dropping them from the assets folder into the project hierarchy.

Next, we used Unity's built-in audio source component to attach these audio clips to various objects and UI elements in the scene. This setup allowed us to control playback, including when and how these clips are triggered during user interactions. We also explored Unity's Audio Mixer to apply additional effects, such as reverb and echo, enhancing the overall immersive experience of the AR application.

This method of integrating text-to-speech into Unity not only facilitated dynamic interaction but also enriched the educational and informational value of the application, making it more engaging for users.

* **Implement visual effects**

To implement visual effects that highlight the intricate mechanics of a jet turbine in our AR project, we used Unity's advanced graphics tools to create compelling and realistic visual enhancements. First, we employed Unity’s Particle System to simulate critical aspects of jet turbine operation, such as exhaust smoke and heat haze effects emanating from the turbine. These particle effects provided a visual indication of the turbine's functionality and power, adding depth and realism to the simulation.

Next, we explored Unity’s Shader Graph to develop custom shaders specifically for rendering metal surfaces and moving parts typical of a jet turbine. This allowed us to achieve a high level of visual detail on the turbine blades and casing, including realistic reflections and the subtle wear-and-tear expected from high-speed mechanical operations. These shaders helped in visually distinguishing different materials and components within the turbine, enhancing the educational aspect of the AR experience by clearly showing how each part contributes to the turbine’s operation.

By integrating these sophisticated visual effects directly into the AR environment, we were able to create a more immersive and interactive learning tool. Users can visually trace the airflow through the turbine and see firsthand the effects of temperature and pressure changes, making the complex operation of a jet turbine more accessible and understandable. The use of Unity's visual effects tools was crucial in bringing the dynamic and powerful nature of jet turbines to life in a virtual setting.

* **Implement VR User interfaces**

To enhance user interaction in our VR project focused on jet turbine mechanics, we implemented intuitive and immersive VR user interfaces using Unity. Our primary goal was to create an interface that was both user-friendly and informative, allowing users to engage deeply with the components and operation of the jet turbine.

* **3D Interactive Menus:** We designed 3D menus that users can access within the virtual environment. These menus provide options for exploring different parts of the jet turbine, such as the combustion chamber, turbine blades, and exhaust systems. By pointing and selecting with VR controllers, users can open detailed informational panels or trigger animations that demonstrate the function of each component.
* **Component-Specific Interaction Buttons:** We integrated buttons that appear when the user focuses on specific parts of the turbine. For instance, hovering over the turbine blades might bring up options to view the blade design up-close or see a simulation of the airflow through the blades. This contextual interface helps keep the user’s experience clean and focused, avoiding overwhelming them with too much information at once.
* **Implement VR Animations**

For the VR animations in our jet turbine project, we focused specifically on animating the turbine fan, which rotates continuously to demonstrate its functionality in real time. This animation is central to understanding how the turbine operates, as the fan's movement is crucial for air intake and the initiation of the compression process within the engine.

**Implementing the Fan Rotation Animation:**

* **Model Setup:** We began by ensuring the 3D model of the turbine fan was accurately designed and aligned with the rest of the turbine components in Unity.
* **Animation Mechanics**: Using Unity’s Animator component, we created a simple yet effective looping animation that continuously rotates the fan. The animation was designed to reflect realistic speeds and behaviors based on actual turbine operations.
* **Interactivity:** To enhance user engagement, we included controls that allow users to adjust the speed of the fan rotation. This interaction not only makes the experience more immersive but also educates users on the effects of different operational speeds on turbine performance.
* **Visual Feedback:** We added visual cues such as slight vibrations and changes in sound pitch to indicate increases or decreases in the fan’s rotational speed. These elements help convey the realism of the turbine's operation and provide immediate sensory feedback to the user’s inputs.
* **Integration with VR Controls:** Users can manipulate the fan speed using VR controllers, making the interaction intuitive and straightforward. This direct control mechanism helps users feel more connected to the simulation, enhancing the educational value of the experience.
* **Iterate on the VR application development**

To enhance our VR jet turbine application, we adopted an iterative development approach, focusing on continuous improvement through user feedback and testing. Here’s a concise overview:

1. **Prototype Development**: We built an initial prototype featuring basic functionalities like the 3D model and the fan animation.
2. **User Testing and Feedback**: We gathered insights from users, including students and professionals, to identify usability and engagement issues.
3. **Feedback Analysis and Implementation**: We analyzed the feedback to prioritize improvements, such as enhancing interactivity and graphical details.
4. **Performance Optimization**: We optimized the application for smooth performance across different VR devices to ensure a seamless user experience.
5. **Final Adjustments**: After multiple iterations, we refined and polished the application, focusing on enhancing educational value and user engagement.

**5.Results :-**

