Development of a Realistic 3D Ocean Cleaning Game using JavaScript and Three.js

Sakibul Hasan[1],Md. Morshed Alam Shanto[2]
[1,2]*Department of Computer Science and Engineering*
[1,2]*Ahsanullah University Of Science and Technology (AUST)*, Dhaka, Bangladesh
Email: 190104074@aust.edu[1], 190104075@aust.edu[2]

I. Abstract

This report outlines the methodology and instructions for creating a 3D realistic ocean-cleaning game using JavaScript and Three.js. The game aims to provide an immersive experience where players control a boat to clean up trash in the ocean. The game's key features include loading a boat model, implementing a Boat class for player control, handling key-press events, creating a Trash class for trash objects, implementing collision detection, and incorporating multiple trash objects for game-play variety.

II. Introduction

The development of a 3D realistic ocean-cleaning game using JavaScript and Three.js presents an opportunity to create an engaging and immersive experience that promotes environmental awareness. The game revolves around the concept of cleaning up trash in a virtual ocean environment [1]. By leveraging JavaScript and Three.js, the game offers a visually immersive experience to players. In this game, players assume the role of a boat captain tasked with cleaning up trash floating in the ocean. By maneuvering the boat through the virtual environment, players must navigate challenging waters, detect and collect scattered trash objects, and ultimately contribute to a cleaner ocean. The game incorporates realistic 3D graphics and physics using the Three.js library, allowing players to explore a visually captivating and dynamic ocean environment.

III. Methodology

The development journey of this project, powered by Three.js, unfolds through a structured sequence of steps aimed at crafting an immersive 3D experience. Our initial stride entails the meticulous setup of the Three.js environment, meticulously creating a canvas to serve as the canvas for our rendering endeavors. Subsequently, the project's core takes form as a boat model is gracefully integrated into the scene via Three.js's capable loader. Guided by the blueprint of a Boat class, a vessel to encapsulate the player's interaction, we anchor attributes such as position, rotation, and the boat's nimble cruising pace. To bestow life to this interaction, the Boat class becomes the receptacle of keypress event handling, seamlessly weaving user input into the boat's fluid motion.The narrative deepens with the introduction of the Trash class, an embodiment of the marine debris plaguing our virtual ocean. This class inherits a repository of attributes, including position and size, and springs to life with a rendering mechanism. The heart of the project beats with the integration of collision detection, a sentinel guarding against unholy unions between the boat and trash. Upon identification of such a collision, the script orchestrates the triumphant exit of the offending trash from the aquatic stage, concurrently tuning the game's vital signs.

In pursuit of variety and realism, the stage welcomes multiple incarnations of the Trash class, their positions harmoniously choreographed across the aqueous expanse. Ingeniously, their initial positions are carefully crafted to preclude premature tango. And then, our project becomes more than a pursuit of points. The deep commitment to a cleaner ocean is reflected in the scoring system, where players' virtuous actions are rewarded with points as they shepherd the discarded flotsam to their rightful place.The symphony of visual storytelling crescendos with the implementation of a camera perspective that dances in harmony with the boat's motion. The stage whispers the immersive secrets of life as the boat meanders - the very environment swaying to the cadence of the boat's journey. In conclusion, the development odyssey etched through the lines of code becomes an ode to craftsmanship. Guided by Three.js's mighty wand, the project weaves an enchanting tapestry of interaction, realism, and environmental stewardship.

IV. Conclusion

The development and examination of the Three.js ocean-cleaning game have provided significant interactive with 3D environments. Besides user gets real time experience with the gaming environment. User easily interacts in the game. After all, by this game we want to give a strong message, to keep our environment safe and pollution free. It keeps all of us safe from the pollution.

V. LIMITATIONS AND FUTURE WORK

In addition to the aforementioned limitations, there are several areas for future work in the development of the 3D realistic ocean cleaning game. One potential avenue is to incorporate educational content into the game. By providing informative pop-ups or mini-lessons throughout gameplay, players can learn about the impact of ocean pollution, the importance of

recycling, and sustainable practices. Furthermore, expanding the game with additional levels, challenges, and objectives can provide extended gameplay and keep players engaged. Additionally, integrating a scoring system, leaderboards, and achievements would introduce a competitive aspect, encouraging replayability and fostering a sense of accomplishment among players. Finally, considering the potential for virtual reality (VR) support or compatibility with other platforms could open up new avenues for an even more immersive and interactive ocean cleaning experience.

REFERENCES

[1] Chen, Ge, et al. "Design and implementation of a 3D ocean virtual reality and visualization engine." Journal of Ocean University of China 11 (2012): 481-487.