eCampusExplorer: Discover NYUAD

Team Members: Aman Sunesh (as18181), Ashmit Shankar Mukherjee (asm8879), Richmond Agyenim-Boateng (ra4150)

1. Introduction:

Choosing your future college can be a daunting decision, especially if you've never had the opportunity to explore it firsthand. Many colleges boast a diverse student body representing numerous countries across the globe. It's indisputable that prospective students, hailing from various corners of the world, yearn for the chance to experience their future campus and explore its vibrant surroundings before finalizing their decision to apply to a specific university. However, the geographical distances that separate these institutions can pose a significant challenge to this aspiration.

In the era of immersive technology, we are excited to introduce eCampusExplorer, a revolutionary solution that transforms the way you explore potential campuses. Imagine a campus tour like no other, where the boundaries between the physical and digital realms seamlessly merge. This is precisely what eCampusExplorer offers – an opportunity for users to embark on an immersive journey through the NYUAD campus, regardless of their physical location, in a virtual environment.

Virtual campus tours are not only a response to the growing demand for innovative and accessible methods of exploration but also an evolution in the field of education and tourism. Our mission is to transcend geographical constraints and provide an unforgettable campus tour experience to prospective students, parents, faculty, professors, and curious minds worldwide. The scope encompasses the development of a user-friendly, interactive, and visually captivating application that leverages the Oculus headset's capabilities to offer an unparalleled virtual campus exploration experience. It's time to step into a new dimension of campus discovery, where you can don the Oculus VR headset and explore the future you've been dreaming of.

1.1 Problem Definition and Objectives:

Problem Statement:

The aim of this project is to develop software for a virtual campus tour of New York University Abu Dhabi, leveraging the Oculus VR Headset. The eCampusExplorer addresses a pressing concern faced by prospective students, parents, and educators globally in the realm of higher education. The problem revolves around physical limitations and geographical distances that often hinder individuals from experiencing potential campuses before making important decisions regarding their education. Many students find it challenging to travel long distances to visit a campus before finalizing their decision. eCampusExplorer aims to tackle this problem by providing an immersive virtual tour of the NYUAD campus, allowing users from around the world to explore it without the need for physical presence.

Objectives:

- **Immersive Campus Exploration:** The primary objective of eCampusExplorer is to deliver a fully immersive and interactive campus tour experience using virtual reality technology. It provides a lifelike representation of all significant campus facilities, enabling users to explore it as if they were physically present, thereby gaining a comprehensive understanding of the campus environment, facilities, and ambiance.

- **Accessibility:** To make campus tours accessible to anyone, regardless of their geographical location. The VR tour also incorporates accessibility features, including audio descriptions and text-to-speech capabilities, ensuring that all users, including those with disabilities, can fully enjoy the experience.

- **Information Provision:** The eCampusExplorer aims to offer relevant information and details about major campus locations, facilities, and resources. It strives to provide insights into the institution's background and heritage. Throughout the virtual tour, users can expect information pop-ups at major campus locations, providing comprehensive details about each spot, accompanied by narration to enhance the immersive experience.

- **User-Friendly Interface:** To design an intuitive user interface that can be easily navigated by users of all ages and technical abilities.

- **Realistic Visuals –** To create a realistic model of the campus in a virtual environment, allowing users to fully immerse themselves in the campus experience.

1.2 Significance of the eCampusExplorer Project

- **Overcoming Geographical Barriers:** The project addresses the significant challenge faced by prospective students, parents, and educators globally, allowing them to explore the NYUAD campus virtually, overcoming the constraints of physical distance.

- **Innovative Exploration Methods:** eCampusExplorer introduces an innovative and accessible method for individuals to explore potential campuses. It embraces immersive technology, offering a unique blend of physical and digital experiences, revolutionizing the conventional campus tour.

- **Enhancing Decision-Making:** Prospective students often face challenges in making informed decisions about their education due to the inability to visit campuses. This project facilitates a more informed decision-making process by providing a realistic and comprehensive virtual tour experience.

- **Inclusivity and Accessibility:** The virtual campus tour is designed to be inclusive, catering to users worldwide, including those with physical disabilities. By incorporating accessibility features such as audio descriptions and text-to-speech capabilities, the project ensures that everyone can participate in the exploration.

- **Advancing Education Technology:** eCampusExplorer contributes to the evolution of education and tourism by leveraging cutting-edge technology. It sets a precedent for the integration of virtual reality in education, showcasing the potential for immersive experiences beyond traditional learning methods.

2. Project Development:

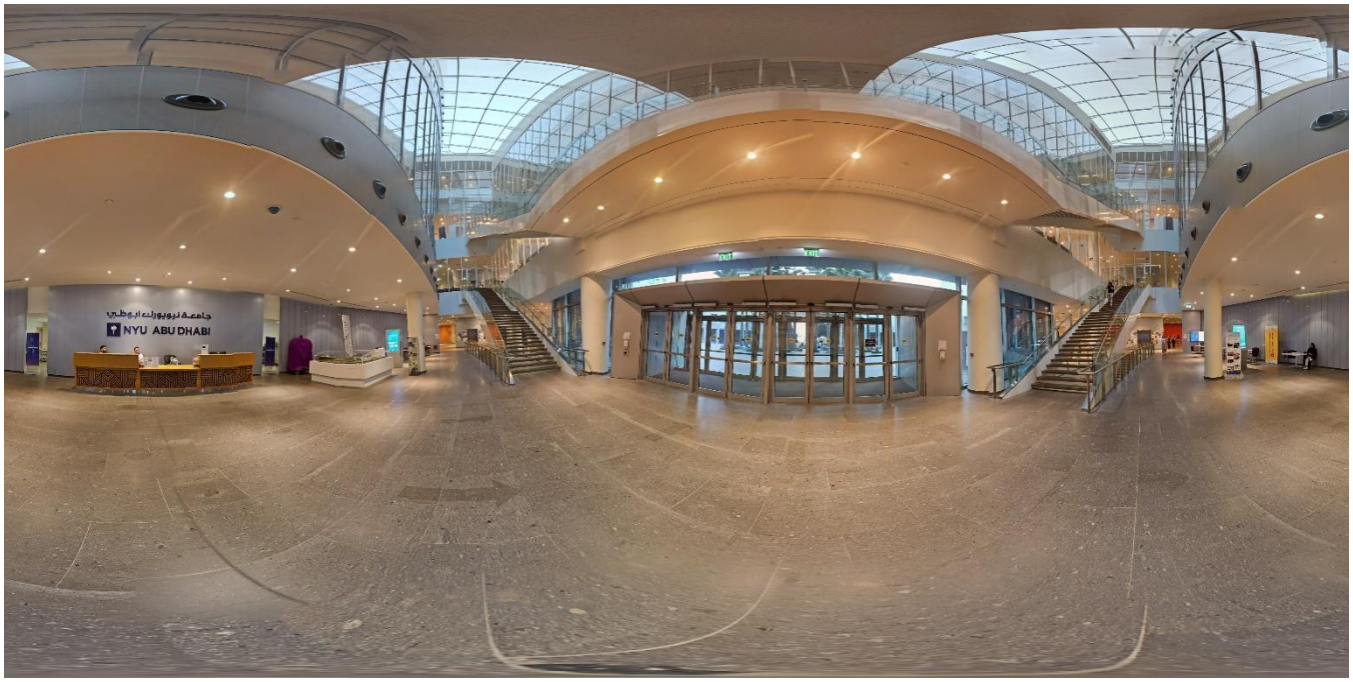
Embark on a comprehensive exploration of the meticulous development process that brought eCampusExplorer to life. This section delves into the technical intricacies, design philosophies, and challenges encountered during the construction of our virtual campus tour application. Utilizing Unity and C# programming, we engineered an immersive experience optimized for the Oculus Meta Quest VR headset (Quest 2, Quest Pro, or Quest 3). Note that this software is **only compatible on Unity Editor Version 2022.3.14f1**. From the architecture of our main menu to the sophisticated 360 Degree image capture techniques, this section unravels the layers of development that culminated in the creation of the eCampusExplorer. Here are some key aspects of our implementation:

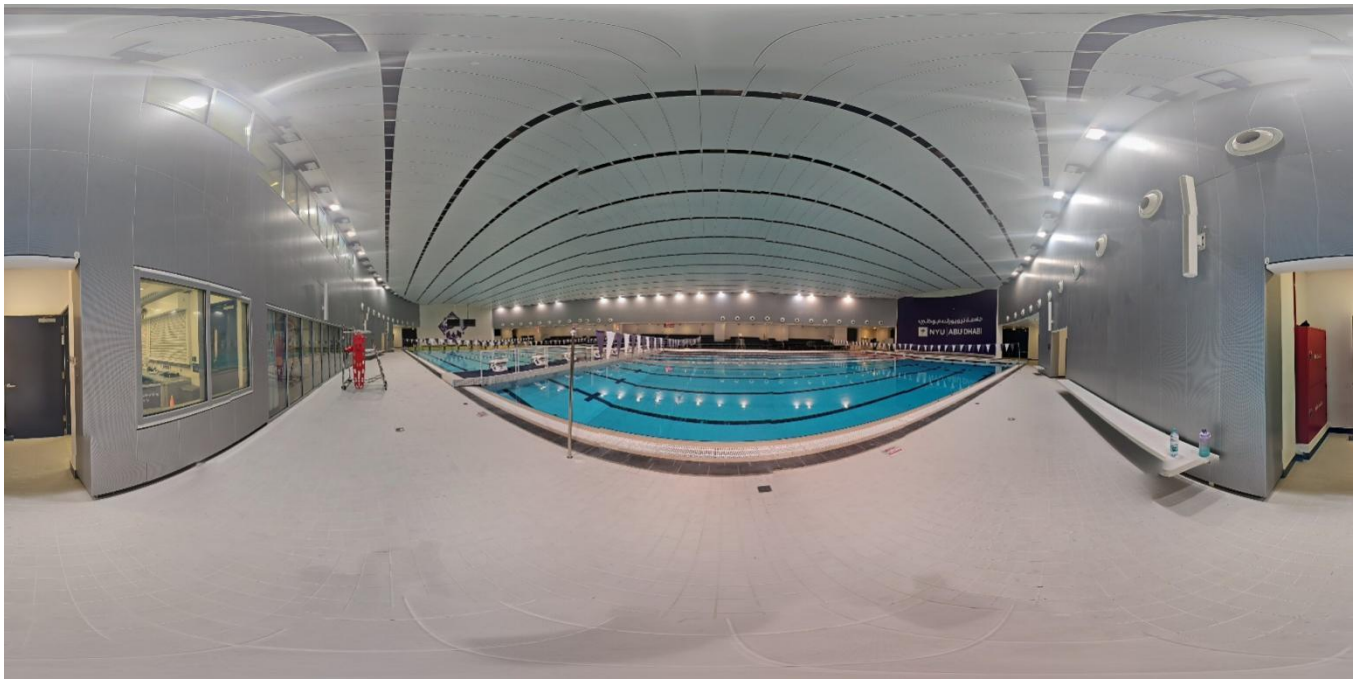
- **Hardware and Platform Compatibility:** Our eCampusExplorer is optimized to run seamlessly on the Meta Quest Pro VR headset or the Meta Quest 3, offering users an immersive campus tour experience. The Meta Quest Pro provides an advanced VR environment with enhanced capabilities for an unparalleled exploration journey.
    - **A) Unity Integration for Build and Run:** eCampusExplorer is developed using the Unity game engine, a powerful platform for creating interactive and visually stunning virtual reality applications. Unity enables us to build and run the software seamlessly on the Meta Quest Pro VR headset, ensuring a smooth and enjoyable experience for users.
    - **B) Navigation with In-Hand Controllers:** The software leverages the in-hand controllers **(YOU CAN ONLY USE THE RIGHT HAND CONTROLLER TO NAVIGATE)** of the Meta Quest Pro VR headset, allowing users to navigate through the virtual campus effortlessly. With the controllers, users can interact with the environment, making the exploration not just visually captivating but also physically engaging.

Here's an overview of the key steps involved in bringing this project to life:

- **STEP 1: 360 Degree Image Capture:** To intricately recreate the virtual campus, our team has extensively captured a series of high-quality images, spanning from the Sushi Counter to Nirvana and includes vital locations such as the entire ground floor of NYUAD (encompassing D1, D2, and entrances of A1, A2, A3, A4, A5, A6, C1, C3 buildings) and all four levels (B1, 1, 2, and 3) of C2. To ensure the utmost precision, all these images were meticulously taken utilizing the advanced GoPro Fusion 360 Camera. These comprehensive visuals have been seamlessly stitched together by the camera, guaranteeing fluid transitions between different locations within the virtual campus tour.

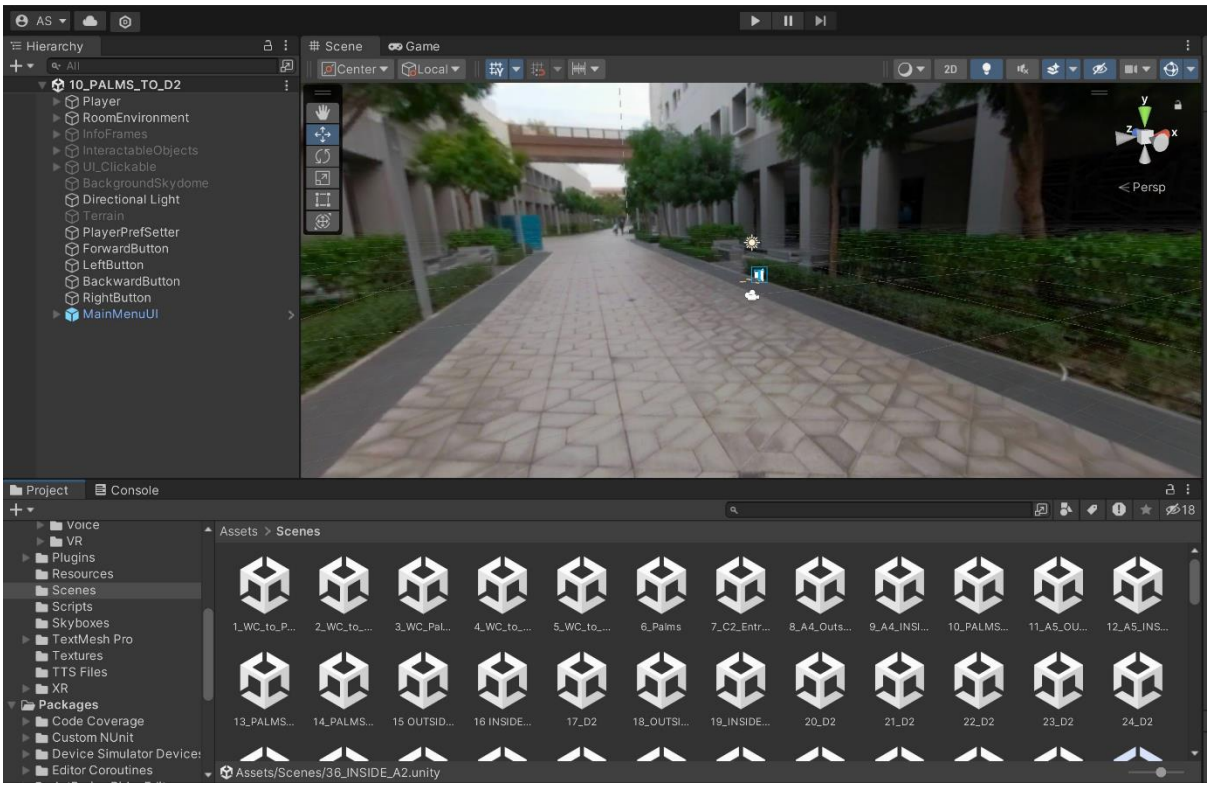
Given below are some sample 360 Degree pictures:




NYU ABU DHABI

- **STEP 2: Image Integration in Unity:** Following the image capture process, we seamlessly integrated all captured images into Unity, employing an organized naming system for efficient management. Each image became a vital piece in constructing a vivid representation of the campus.

  To enhance the immersive experience, we crafted a unique skybox for every location captured. These skyboxes serve as the visual backdrop, encapsulating the essence of each scene. With meticulous attention to detail, we meticulously designed and implemented over 120 scenes, each encapsulating a distinctive facet of the campus. For each of these skyboxes, we created a corresponding scene on Unity.



- **Step 3: Teleportation Script & Direction Control:** For the next stage, we drafted C# scripts 1) to set the global state, and 2) to facilitate teleportation between different scenes or areas in our Unity application, allowing users to navigate based on their chosen direction.

➢ The PlayerPrefSetter script in Unity is designed to set a global state value using PlayerPrefs. It includes the following functionality:

```
using UnityEngine;

public class PlayerPrefSetter : MonoBehaviour
{
    // This is the value that you can set in the Unity Editor.
    public string globalStateValue = "1";
```

```
    private void Awake()
    {
        SetGlobalState(globalStateValue);
    }

    public void SetGlobalState(string value)
    {
        PlayerPrefs.SetString("globalState", value);
    }
}
```

1) Awake() Function: The Awake method is called when the script instance is loaded. In this script, it sets the global state using the SetGlobalState method with the initial value specified in the Unity Editor.

2) SetGlobalState(string value) Function: The SetGlobalState method takes a string parameter (value) and uses PlayerPrefs to store this value under the key "globalState." PlayerPrefs is a way to store and retrieve persistent data between game sessions in Unity, making it suitable for saving and accessing global states.

- Below is the C# script for teleportation:

```
using System.Collections;
using System.Collections.Generic;
using System.Diagnostics;
using UnityEngine;
using UnityEngine.SceneManagement;

public class Teleporter : MonoBehaviour
{
    //List of all states (forward, backward, left, right)

    private string[] State1 = { "2_WC_to_Palms", "Disabled", "101_MYSK",
"106_WC" };
    private string[] State101 = { "102_BOBA", "1_WC_to_Palms", "Disabled",
"Disabled" };
    private string[] State102 = { "103_BLACKSMITH", "101_MYSK", "Disabled",
"Disabled" };
    private string[] State103 = { "104_SALOON_1", "102_BOBA", "Disabled",
"Disabled" };
    private string[] State104 = { "105_SALOON_2", "103_BLACKSMITH",
"Disabled", "Disabled" };
    private string[] State105 = { "110_Outside_Path", "104_SALOON_1",
"Disabled", "Disabled" };
    private string[] State110 = { "111_Sushi_Counter", "105_SALOON_2",
"Disabled", "Disabled" };
    private string[] State111 = { "Disabled", "110_Outside_Path",
"Disabled", "Disabled" };

    private string[] State106 = { "107_ARTS_GALLERY", "1_WC_to_Palms",
"Disabled", "Disabled" };
```

```
    private string[] State107 = { "108_PHARM", "106_WC", "Disabled",
"Disabled" };
    private string[] State108 = { "109_NIRVANA 5", "107_ARTS_GALLERY",
"Disabled", "Disabled" };
    private string[] State109 = { "Disabled", "108_PHARM", "Disabled",
"Disabled" };



    //Similarly, we have states for all other scenes


    // Start is called before the first frame update
    void Start()
    {

    }

    // Update is called once per frame
    void Update()
    {

    }

    void OnTriggerEnter(Collider other)
    {
        if (other.CompareTag("NavigationArrow"))
        {
            string globalState = PlayerPrefs.GetString("globalState", "A");
            string[] currentPossibleDirections;

            //All switch cases
            switch (globalState)
            {
            case "1":
                currentPossibleDirections = State1;
                break;
            case "101":
                currentPossibleDirections = State101;
                break;
            case "102":
                currentPossibleDirections = State102;
                break;
            case "103":
                currentPossibleDirections = State103;
                break;
            case "104":
                currentPossibleDirections = State104;
                break;
            case "105":
                currentPossibleDirections = State105;
                break;
            case "106":
                currentPossibleDirections = State106;
                break;
            case "107":
                currentPossibleDirections = State107;
```

```
        break;
    case "108":
        currentPossibleDirections = State108;
        break;
    case "109":
        currentPossibleDirections = State109;
        break;
    case "110":
        currentPossibleDirections = State110;
        break;
    case "111":
        currentPossibleDirections = State111;
        break;


    //Similarly, we have included cases for all other scenes


    default:
       currentPossibleDirections = State1; // Default to State1 or
       handle differently
        break;
    }


    //Handle the 4 arrows
    // Get the name of the GameObject that collided
    string ArrowName = other.gameObject.name;

    //Do something when the object with the "NavigationArrow" tag
    enters the trigger
    UnityEngine.Debug.Log(ArrowName);
    switch (ArrowName)
    {
    case "ForwardButton":
        UnityEngine.Debug.Log("The Forward button was pressed.");
        if (currentPossibleDirections[0] != "Disabled")
        {
            SceneManager.LoadScene(currentPossibleDirections[0]);
        }
        // Insert additional logic for ForwardButton here
        break;
    case "BackwardButton":
        UnityEngine.Debug.Log("The Backward button was pressed.");
        if (currentPossibleDirections[1] != "Disabled")
        {
            SceneManager.LoadScene(currentPossibleDirections[1]);
        }
        // Insert additional logic for BackwardButton here
        break;
    case "LeftButton":
        UnityEngine.Debug.Log("The Left button was pressed.");
        if (currentPossibleDirections[2] != "Disabled")
        {
            SceneManager.LoadScene(currentPossibleDirections[2]);
        }
        // Insert additional logic for LeftButton here
```

```
                break;
            case "RightButton":
                UnityEngine.Debug.Log("The Right button was pressed.");
                if (currentPossibleDirections[3] != "Disabled")
                {
                    SceneManager.LoadScene(currentPossibleDirections[3]);
                }
                // Insert additional logic for RightButton here
                break;
            default:
                UnityEngine.Debug.Log("Button name not recognized.");
                break;
            }
        }
    }
}
```

Central to our project is the above teleportation script, which facilitates movement within the virtual environment. This script is responsible for handling teleportation between scenes based on the user's selected direction. Every scene within the Unity project file is associated with an array specified in the C# teleportation script. This array defines the four possible directions: forward, backward, left, and right (formatted as {forward, backward, left, right}). It contains data that assigns the names of scenes each specific direction would teleport you to from a particular scene. If the initial location is designated as the entrance of the NYUAD campus, specifically the path from the welcome center to the palms, the 'State1' array in the C# code encompasses four potential directions: forward, backward, left, and right. In this context, choosing the forward arrow will transition you to the Palms, selecting the left arrow will navigate you to Mysk, and opting for the right arrow will lead you to the Welcome Center. However, the back arrow will remain hidden from the user in this particular scene, as this direction does not lead to any specific location.

The teleportation destinations and available directions are determined by the current "globalState." This script responds to triggers from objects with the "NavigationArrow" tag and performs the teleportation logic accordingly. In general, the script identifies the possible destinations or directions the user can travel to and prepares to move the user to the chosen location.

The teleportation script includes the following functionalities:

1) Scene Arrays: The script defines arrays (e.g., State1, State101, etc.) that represent possible teleportation directions for different global states. Each array contains scene names that can be loaded when a specific condition is met.

2) OnTriggerEnter: This function is called when a collider enters the trigger zone of the GameObject this script is attached to. It checks if the collider has the tag "NavigationArrow" before proceeding.

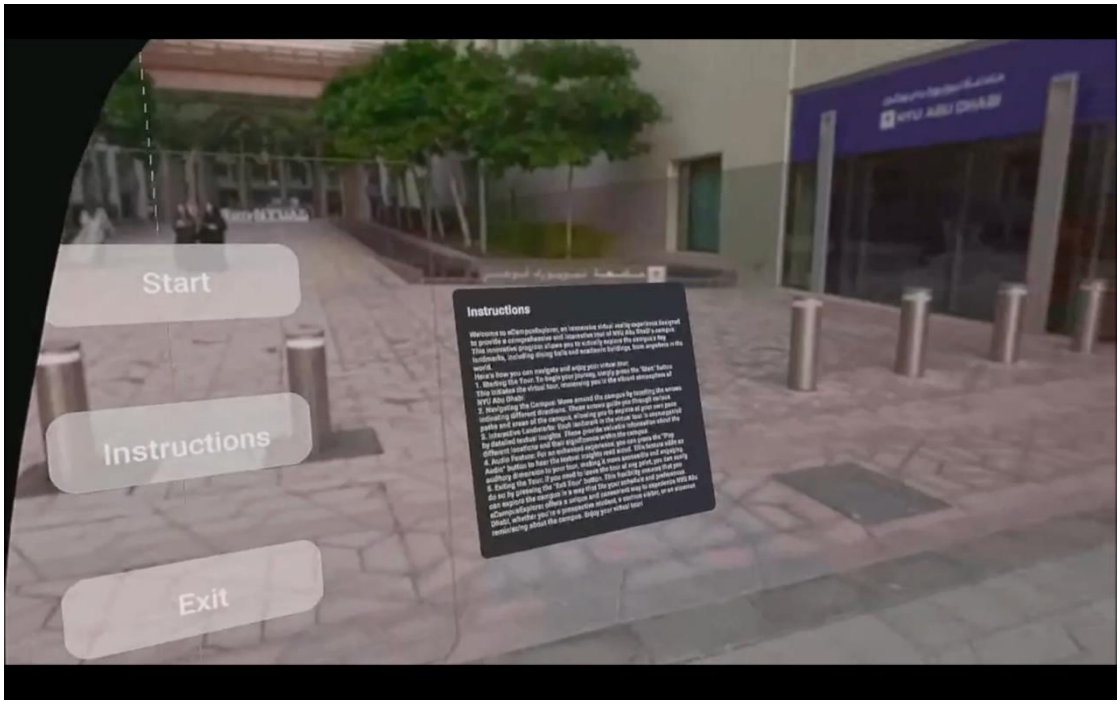
3) Global State Switch: It uses a switch statement to determine the current set of possible directions based on the global state retrieved from PlayerPrefs. If the global state is unrecognized, it defaults to State1. For example, if the user is currently in the third scene, the global state is saved as 3, and the switch case would assign the array containing the names of all possible scenes that can be accessed from Scene 3 (the array State3) to the array named 'currentPossibleDirections' .

4) Arrow Button Handling: It further checks which arrow button (Forward, Backward, Left, Right) triggered the collision. Depending on the button pressed, it loads the corresponding scene if the direction is enabled (not "Disabled"). For instance, suppose the user clicks on the 'ForwardButton', the script retrieves the name of the scene saved in the 0^{th} index in the 'currentPossibleDirections' array. Subsequently, it utilizes the LoadScene function of the Scene Manager module to teleport to that scene. Similarly, to teleport to the backward, left, and right scenes, the script loads the scene located at the 1st, 2nd, and 3rd indices in the 'currentPossibleDirections' array, respectively.

5) Debug Logging: It logs information to the Unity console, such as the name of the arrow button and a message if the button name is not recognized.

- **STEP 4: Creating Main Menu and Enhancing User Interaction with Accessibility Features:** In this pivotal step, we focused on crafting an intuitive and engaging main menu for our virtual campus tour application. Our attention to detail extended to integrating user interaction enhancements and accessibility features to ensure that the campus tour is inclusive and user-friendly for all. This section delves into the design and implementation that went into creating a main menu and additional features that enriches the overall user experience.

  - **Main Menu:** At the start of the program, you are in the main menu where you will be presented with 3 options:
    - **A) Start Tour:** The primary option is to begin the campus tour experience.
    - **B) Instructions:** By clicking on this option, the user can learn about the project and understand how to use the application effectively.
    - **C) Exit:** An option to exit the app.



- **Code:**

```
using UnityEngine;
using UnityEngine.SceneManagement;
#if UNITY_EDITOR
using UnityEditor;
#endif
public class ButtonFunctions : MonoBehaviour
{
    public GameObject AboutText;

    public void StartTour()
    {
        // Load the scene with the name "1_WC_to_Palms"
        SceneManager.LoadScene("1_WC_to_Palms");
    }

    public void About()
    {
        // Toggle the visibility of the AboutText
GameObject
        if (AboutText != null)
        {
            AboutText.SetActive(!AboutText.activeSelf);
        }
    }

 public void Exit()
    {
        // Logs the quit request in the Unity Editor
console
        Debug.Log("Quit requested");

#if UNITY_EDITOR
        // Stop playing the scene in the Unity Editor
        EditorApplication.isPlaying = false;
#else
        // Quit the application
        Application.Quit();
#endif
    }
}
```

This code is used to implement the main menu. It declares a public class that inherits from 'MonoBehaviour' and includes three public functions as follows:

1) StartTour Function:
The StartTour function in the ButtonFunctions script utilizes Unity's SceneManager to load the scene named "1_WC_to_Palms," enabling a transition to that specific scene.

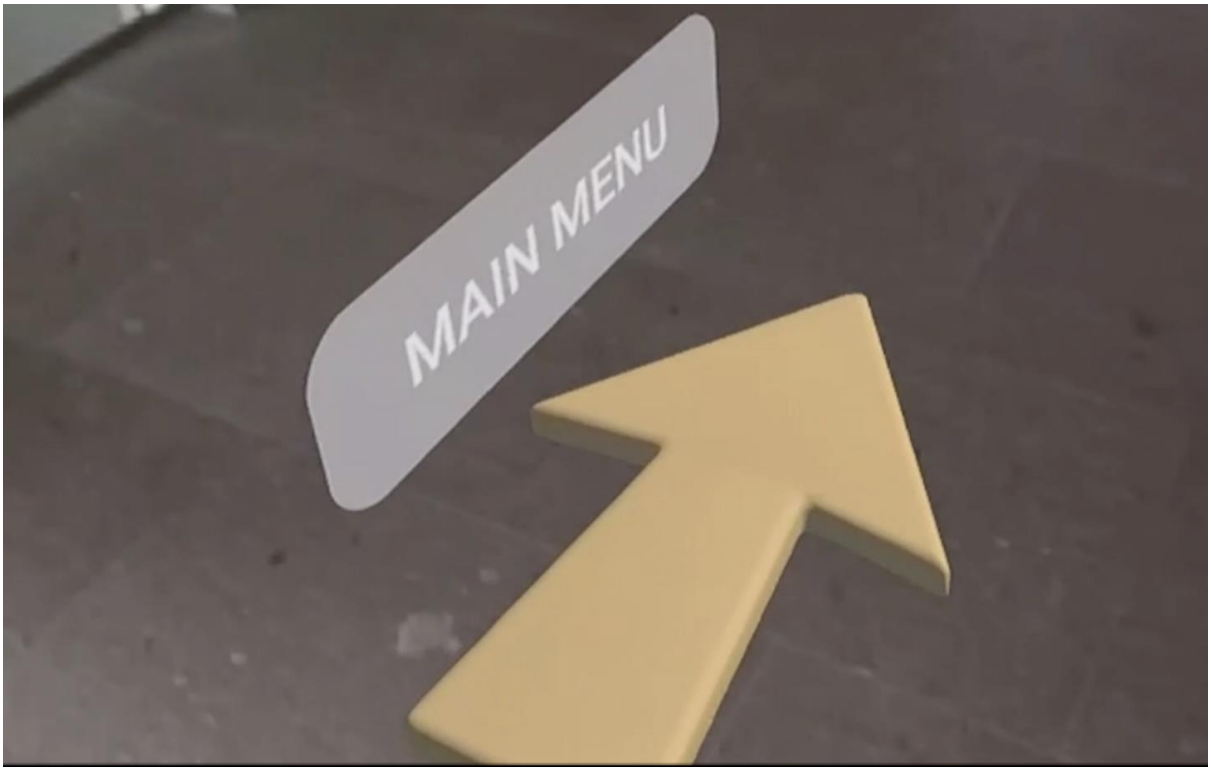
2) About Function:
The About function toggles the visibility of the specified AboutText GameObject. If the GameObject is not null, it switches between active and inactive states, controlling the visibility of an informational text box.

3) Exit Function:
The Exit function logs a quit request to the Unity Editor console. Additionally, it handles application termination differently based on whether the code is running within the Unity Editor or as a standalone application. In the editor, it stops playing the scene, while in the standalone application, it initiates an application quit.

- **Navigation Tips:**
  - **Observing & Navigating Through the Campus:** During the tour, take full advantage of the 360-degree view of the campus. Surrounding you, you'll notice arrows guiding you to specific locations. To move to the next scene, click on the arrow using your right-hand controller. Swipe your finger on the arrow from top to bottom at a moderate pace to avoid unintentionally moving through multiple scenes.

  - **Audio Option:** At locations featuring information boxes, an audio option is available. Utilize your right-hand controller's trigger to click on the option and listen to the accompanying audio. This adds an extra layer of detail and context to your virtual tour.

  - **Main Menu Access:** In every scene, locate the 'Main Menu' option positioned on the bottom right-hand side of your screen. Click on it using your controller to swiftly redirect back to the main menu. This option provides a convenient way to navigate and explore different areas of interest without interrupting the overall tour experience.



- **Code:**

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.SceneManagement;

public class MainMenuSceneSwitcher : MonoBehaviour
{
     // This public function can be called to switch to the
Main Menu scene
    public void GoToMainMenu()
    {
        SceneManager.LoadScene("Main Menu");
    }
    // Start is called before the first frame update
}
```

This script, MainMenuSceneSwitcher, is a simple Unity script that allows you to switch to the Main Menu scene. It declares a public class that inherits from MonoBehaviour and includes a public function named GoToMainMenu. This function utilizes Unity's SceneManager to load the scene named "Main Menu." The script is attached to the Main Main button (GameObject) in the Unity scene, and the GoToMainMenu function can be called by this button, to initiate the transition to the Main Menu scene.

- **User Interaction & Accessibility Features:**
  - **360 Degree View of Each Scene:** Our eCampusExplorer provides users with a captivating and immersive experience through a 360-degree view of each scene. Users have the freedom to look around and explore the virtual campus from every angle, gaining a comprehensive understanding of the campus's aesthetics and surroundings. This feature enhances the sense of presence, allowing users to feel as if they are physically present within each scene.

  - **Navigation Arrows for Seamless Transitions:** Within each scene, users encounter four interactive arrows (https://sketchfab.com/360 Degree-models/direction-arrow-6ef46718c7b242e39fcad7f27ee858a5) representing forward, backward, left, and right directions. These arrows serve as intuitive navigation tools, enabling users to seamlessly transition between scenes based on their chosen direction. Interacting with these arrows triggers a smooth transition to the next scene, providing a user-friendly and dynamic means of navigating through the virtual campus. This interactive navigation system ensures that users can explore the campus at their own pace and focus on areas of specific interest.

    The navigation system is implemented by associating arrows with the user's right hand. A virtual cylinder is inserted into the user's right-hand index finger. When the user interacts with an arrow by clicking it using their index finger, the associated cylinder on the finger is triggered, seamlessly transporting the player to the chosen destination within the virtual environment. When a player interacts with a virtual arrow in the VR world, the script determines which arrow was

selected. Then, it takes specific actions based on the selected arrow. Each arrow corresponds to a particular direction, like 'Forward,' 'Backward,' 'Left,' or 'Right.' Upon selecting an arrow, the script guides the player to a new location within the virtual world, aligning with the chosen direction specified by the arrow.

- **Information Boxes for Contextual Details:** To enrich the user experience, we have implemented information boxes strategically placed under all important locations. These boxes provide users with contextual details, describing the significance and functionality of each location. Users can access valuable information about the campus's landmarks, facilities, and historical background, enhancing their understanding of the virtual environment.

- **AI-Generated Audio Narration:** Adding another layer of interactivity, eCampusExplorer incorporates AI-generated audio for a more immersive tour. Utilizing the ElevenLabs website, we have created audio narratives associated with each important location. Users can click the audio button to trigger narrations that describe the place and provide insights into its usefulness or function. This feature adds a dynamic auditory dimension to the virtual tour, offering users a multisensory exploration of the campus.

- **STEP 5: Adding Final Touches and Testing:** As we approached the culmination of our project timeline, we dedicated this stage to refining and perfecting every detail. Here are the key actions undertaken during this phase:

    - **Skybox Alignment:** Carefully positioned and reoriented all skyboxes to ensure they align seamlessly, facilitating smooth transitions between scenes. This meticulous adjustment aimed to enhance the overall visual continuity of the virtual campus tour.

    - **Arrow Placement:** Positioned arrows precisely in their designated locations within scenes, ensuring intuitive navigation for users. Disabled arrows, which did not lead to any scene, were removed to eliminate confusion and streamline the user experience.

    - **Background Music:** To elevate the user experience, background music was incorporated into the main menu scene. This addition aimed to create a captivating and immersive ambiance for users embarking on the virtual tour.

    - **Main Menu Optimization:** All options within the main menu were strategically positioned for easy access and navigation. The layout was fine-tuned to enhance user interaction and provide a seamless entry point into the diverse features of the application.

    - **Testing and Error Correction:** Rigorously tested the application multiple times to identify and rectify any potential errors. This phase involved addressing points of concern, ensuring the application met the highest standards of performance, visual appeal, and user engagement.

This comprehensive approach ensures that our eCampusExplorer delivers an engaging and immersive virtual campus tour experience, offering users a realistic and interactive way to explore the campus's diverse locations.

The following link provides a comprehensive visual showcase of our VR Campus Tour project in action:

https://www.youtube.com/watch?v=dlHD-bC170Q

The following link includes the project file for your reference:

https://drive.google.com/file/d/1BvI5djKg6iHXhJ8XMbfT0xbi6Bv0-LT2/view?usp=sharing

Backup:

https://nyu0my.sharepoint.com/:u:/g/personal/asm8879_nyu_edu/EVWfmV8xHttPqYCYHkjJ0egBwryBN-oVjnPSggMOyXvmzQ?e=3hgLeL

3. Results and Evaluation:

3.1 Challenges

- **Photographic Quality Enhancement:**
  The process of capturing 360 Degree images posed an initial challenge. Our team initially used a mobile application to photograph scenes, but this approach generated a 360 Degree image for each scene by stitching together 26 different images. Unfortunately, this method resulted in minor defects and breaks within the 360 Degree images due to the stitching process. In response, we opted to enhance the photographic quality by acquiring a GoPro Fusion 360 camera. This decision aimed to ensure the incorporation of the highest-quality images into the software, overcoming the limitations of the initial mobile application.

- **Ethical Photography Practices:**
  A significant challenge in the photography phase revolved around ethical considerations. We were committed to respecting the privacy and consent of individuals present in the campus spaces we captured. To address this, we either patiently waited for spaces to clear up naturally or proactively sought consent from each individual in the close vicinity of our photography sessions. This ethical approach required additional time and effort but was essential to align our project with respectful and responsible photography practices.

- **Inclusive Accessibility Features:**
  Implementing comprehensive accessibility features proved to be a complex challenge. Addressing diverse user needs, including those with disabilities, required careful consideration of audio descriptions, and other features to make the virtual tour accessible to a wide audience. Considering this challenge, we have successfully incorporated audio descriptions and strategically placed text boxes under each important location. These additions ensure that users, including those with visual or auditory impairments, can engage with the virtual tour effectively, enhancing their overall experience and inclusivity.

- **User Engagement Retention:**
  Sustaining user engagement throughout the virtual campus tour was a challenge. Implementing interactive elements and captivating content to retain user interest and involvement required ongoing refinement and creativity.

3.2 Errors and Debugging

During the development and testing phases of eCampusExplorer, several errors and issues were identified and addressed to ensure a seamless and error-free user experience. Here are the key points related to errors and debugging:

- **Skybox Alignment Issues:**
  - **Issue:** Some skyboxes were initially positioned incorrectly, affecting the overall visual continuity and causing discomfort during transitions between scenes.
  - **Resolution:** During the testing process, all skyboxes' positions were carefully adjusted to ensure proper alignment. This meticulous adjustment aimed to enhance user convenience and facilitate smooth navigation throughout the virtual campus tour.

- **Teleportation Errors in Arrays:**
  - **Issue:** Errors were identified in the arrays, leading to issues with teleportation. Some scenes were not accessible initially due to these errors.
  - **Resolution:** Thorough testing of all scenes was conducted during the testing phase. Identified errors in teleportation arrays were systematically fixed to ensure that users could seamlessly transition between scenes. This debugging process aimed to provide a flawless and immersive exploration experience.

- **Arrow Placement and Player Position Adjustment:**
  - **Issue:** Arrows were initially placed too low for convenient reach during the first attempt to run the code. This hindered the user's ability to interact with the arrows easily.
  - **Resolution:** Player position in the game was adjusted to optimize the height of the arrows. This adjustment aimed to enhance user convenience, ensuring that the arrows are positioned at an accessible height for the player. The goal was to make arrow-clicking intuitive and user-friendly without obstructing the user's view.

These debugging efforts were crucial in addressing initial issues and refining the eCampusExplorer application. Thorough testing and systematic debugging ensured that users could navigate the virtual campus seamlessly, providing an immersive and error-free exploration experience.

4. Conclusion and Future Work:

In the ever-evolving landscape of higher education exploration, eCampusExplorer: Discover NYUAD emerges as a transformative tool, breaking down geographical barriers and ushering in a new era of virtual campus tours. Our team embarked on a mission to revolutionize the way prospective students, parents, and educators experience potential campuses, particularly the vibrant grounds of New York University Abu Dhabi (NYUAD).

Through the lens of immersive technology and virtual reality, eCampusExplorer extends an invitation to users globally to traverse the NYUAD campus seamlessly, regardless of their physical location. We recognized the inherent challenges faced by those eager to explore campuses firsthand but hindered by distance. In response, our virtual reality solution not only provides a visually captivating tour but also integrates features like audio descriptions, text-to-speech capabilities, and information pop-ups under key locations, ensuring accessibility for a diverse audience, including those with disabilities.

4.1 Project Achievements:

The journey of eCampusExplorer has been marked by significant accomplishments, reflecting the dedication and innovation of our team. As we set out to redefine the paradigm of campus exploration through virtual reality, several key achievements stand out:

- **Immersive Campus Exploration:** One of our primary objectives was to deliver a fully immersive and interactive campus tour experience. With the successful implementation of the Unity game engine and C#, users can now navigate the NYUAD campus seamlessly, experiencing a lifelike representation of significant facilities. The use of the Meta Quest Pro VR headset enhances the exploration journey, offering advanced capabilities for an unparalleled virtual experience.

- **Accessibility Integration:** Recognizing the importance of inclusivity, we prioritized the integration of accessibility features. The addition of audio descriptions, and strategically placed information boxes ensures that the virtual tour caters to a diverse audience, including individuals with visual or auditory impairments. This commitment aligns with our vision of making campus exploration accessible to anyone, regardless of their location or abilities.

- **User-Friendly Interface:** The design of an intuitive user interface was crucial to enable users of all ages and technical abilities to navigate the virtual environment effortlessly. The main menu offers clear options, including starting the tour, obtaining information, and exiting the application. This user-centric approach enhances the overall experience and facilitates seamless interaction.

- **Realistic Visuals:** The meticulous process of capturing high-quality 360 Degree images using the GoPro Fusion 360 camera and stitching them together has resulted in a realistic model of

the NYUAD campus in the virtual environment. Users can now enjoy a visually captivating tour, with fluid transitions between different locations, ensuring an engaging and authentic representation of the campus.

- **Teleportation & Direction Control:** Central to the user experience is the teleportation script, allowing users to navigate through scenes effortlessly. The script's design ensures smooth transitions between different areas, providing users with choices based on their preferred direction. This dynamic teleportation system contributes to the overall fluidity and interactivity of the virtual campus tour.

4.2 Challenges Addressed:

The acknowledgment and resolution of challenges throughout the project underscore our commitment to delivering a high-quality virtual tour. From enhancing photographic quality and adhering to ethical photography practices to addressing user engagement retention and inclusive accessibility features, each challenge was met with strategic solutions, resulting in a more refined and comprehensive product.

In summary, the eCampusExplorer stands as a testament to our collective efforts to transcend the boundaries of traditional campus exploration. Our achievements reflect not only technical prowess but a commitment to inclusivity, user satisfaction, and the continuous pursuit of excellence in the realm of virtual reality and educational technology.

4.3 Future Scope and Development:

As eCampusExplorer: Discover NYUAD lays the foundation for a transformative campus exploration experience, envisioning its future development opens avenues for even more immersive and comprehensive features. The roadmap for the evolution of eCampusExplorer includes:

- **Multi-Campus Expansion:**
  Extend the platform to encompass virtual tours of multiple campuses within the New York University (NYU) network. This expansion aims to provide prospective students with a broader perspective on the diverse campuses affiliated with NYU, offering unique insights into the academic, cultural, and architectural nuances of each institution.

- **Dynamic Day-Night Cycle:**
  Implement a system that allows users to experience the campus during different times of the day—morning, evening, and night. This feature aims to enhance the realism of the virtual environment, showcasing the campus in various lighting conditions. Users can explore the beauty of the campus at different hours, fostering a more immersive and dynamic experience.

- **Live Campus Events Integration:**
  Integrate a feature that enables users to virtually attend live events happening on the NYUAD campus. From lectures and performances to real-time campus tours, this addition provides an up-to-date and engaging experience. Users can participate in ongoing events, fostering a sense of connection with the vibrant life of the campus.

- **Interactive Learning Modules:**
  Develop interactive learning modules within the virtual tour, offering educational content related to specific academic departments, courses, or research initiatives. This addition serves as a valuable resource for prospective students, providing in-depth information about the academic offerings and research endeavors at NYUAD.

- **Interactive Map Navigation:**
  Create an interactive map for each location, displaying all the important landmarks. Users can click on these locations on the map to teleport directly, enhancing navigation efficiency and providing a comprehensive overview of the campus layout.

- **Student Ambassador Collaborations:**
  Collaborate with student ambassadors to integrate personalized narratives, experiences, and insights into the virtual tour. Adding a human touch to the exploration, prospective students can connect with real individuals within the NYUAD community, gaining authentic perspectives on campus life.

- **Interactive Tour Guide Character:**

  Design an interactive character within the software to act as a virtual tour guide. This character would accompany users, offering narration and insights as they navigate the campus. This interactive guide enhances the storytelling aspect of the virtual tour, making the exploration more engaging and informative.

- **Virtual Events within VR:**
  Organize virtual events within the VR environment, such as open houses, webinars, or Q&A sessions with faculty members. This immersive approach provides a unique way for prospective students to engage with the NYUAD community, fostering a sense of connection and belonging.

- **Language Localization Features:**
  Implement language localization features to make the virtual tour accessible to a global audience. Translate information, narration, and user interface elements into multiple languages, ensuring that users from diverse linguistic backgrounds can fully enjoy and understand the virtual exploration.

- **Virtual Events Calendar:**
  Develop a calendar of virtual events and activities happening on the NYUAD campus. Users can access information about upcoming seminars, cultural festivals, and guest lectures, enhancing their awareness of campus life and providing a dynamic view of the events shaping the NYUAD community.

The future development of eCampusExplorer aims to continually enhance the user experience, foster engagement, and provide a comprehensive and dynamic representation of the NYUAD campus life.

5. Reflection on Learning:

The journey of developing eCampusExplorer NYUAD has been a profound learning experience, providing insights and skills that extend beyond the realm of virtual campus tours. The team collectively embraced challenges, honed technical expertise, and gained a deep understanding of collaborative project development. The learning outcomes can be summarized as follows:

- **Technical Proficiency in VR Development:**
  The project required a comprehensive understanding of Virtual Reality (VR) development using the Unity game engine and C#. Team members developed proficiency in creating immersive environments, implementing interactive features, and optimizing the application for seamless VR experiences on the Meta Quest headset.

- **Photography and Imaging Enhancement:**
  The initial challenge of stitching 26 different images to create a 360 Degree image highlighted the importance of high-quality photography. Transitioning to the GoPro Fusion 360 camera showcased the significance of leveraging advanced imaging technology for capturing detailed and seamless 360 Degree representations.

- **Ethical Considerations in Project Execution:**
  The need for consent during campus photography sessions underscored the ethical considerations in technology projects. Learning to navigate the balance between project objectives and respecting the privacy and consent of individuals contributed to a nuanced understanding of responsible project execution.

- **Inclusive Design and Accessibility Features:**
  Addressing the challenge of implementing accessibility features heightened awareness of the importance of inclusive design. Incorporating audio descriptions and text boxes under each location underscored the commitment to making the virtual tour accessible to users with diverse needs.

- **Team Collaboration and Division of Responsibilities:**
  The project's success hinged on effective collaboration and the clear division of responsibilities among team members. Each member took on major and minor responsibilities, contributing to different facets such as coding, audio work, multimedia content creation, and 360 Degree image capture.

- **Project Management and Problem Solving:**
  The challenges encountered during 360 Degree image capture and ethical photography practices demanded effective problem-solving. Overcoming these challenges enhanced the team's project management skills and ability to adapt to unforeseen circumstances.

- **User-Centric Design Principles:**
  The development of user-friendly interfaces and interactive features emphasized the importance of user-centric design. Considering user experience in terms of navigation, information access, and engagement played a crucial role in shaping the virtual tour application.

- **Integration of AI-Generated Audio Narration:**
  Incorporating the AI-generated audio narration introduced the team to cutting-edge technologies. Integrating AI-generated content into the virtual tour enhanced the overall interactivity, showcasing the potential of emerging technologies in project development.

The learning outcomes of eCampusExplorer extend beyond the technical skills acquired, encompassing ethical considerations, collaborative dynamics, and a holistic understanding of user experience. As the project concludes, the team members carry forward these lessons, ready to apply them to future endeavours in technology and immersive experiences.

6. Contribution Summary:

The workload is equally divided among all the team members with each team member responsible for one major and one minor responsibility. The division of the work among the team members is as follows:

1) **Aman Sunesh (as18181):**

Major Responsibility: C# Script Development

- Develop the C# scripts required for the project.
- Implement interactive features, navigation, and user interface elements within the VR environment.

Minor Responsibility: Data Collection and Research

- Gather information about the university campus, such as historical facts, notable landmarks, and interesting details.

- Research best practices for VR development and user experience design to enhance the project. Research and implement features to make the VR tour accessible to a wide range of users, including those with disabilities.

2) **Ashmit Shankar Mukherjee (asm8879)**

Major Responsibility: Audio Work and Multimedia Content Creation

- Develop and produce a range of audio elements, including background music, sound effects, and voiceovers, to be integrated into the VR environment. These audio elements play a crucial role in setting the mood and enhancing the overall experience.

- Record and edit voiceovers that provide narration, commentary, or character dialogue within the VR tour. Ensure that voiceovers are clear, engaging, and synchronized with the on-screen actions.
  Ensure the 360 Degree models are optimized for VR integration and navigation.

Minor Responsibility: Creating Skyboxes, Graphic Design and User Interface (UI) Enhancements

- Create a skybox for each picture in unity.
- Create visually appealing icons, buttons, and UI elements for the VR application.
- Work on enhancing the overall aesthetics of the user interface to make it more engaging and user-friendly.

3) Richmond Agyenim-Boateng (ra4150)

Major Responsibility: Capturing 360 Degree Pictures of the University

- Capturing detailed 360 Degree images of the university's surroundings using the GoPro Fusion 360 camera, including buildings, outdoor areas, and notable landmarks. This process involves taking multiple photographs from different angles to create comprehensive 360 Degree representations.
- Ensure that the 360 Degree images captured are optimized for seamless integration into the VR environment. This includes converting the images into VR-compatible formats and making them compatible with navigation within the virtual tour.

Minor Responsibility: Script and Dialogue Writing

- Develop engaging dialogue for characters or narration within the VR experience.
- Creating the information pop-ups within the virtual campus tour, providing relevant details about key locations on campus.

7. References:

1) https://www.youtube.com/watch?v=q0T-4_zUv8Y&list=PLmxqIl8EWtFjCcSd-AFuzW2es30NAobqf

2) https://www.youtube.com/watch?v=ZoZcBgRR9ns

3) https://www.youtube.com/watch?v=02NI9kwOOKo

4) https://docs.unity3d.com/ScriptReference/SceneManagement.SceneManager.html

5) https://sketchfab.com/360 Degree-models/direction-arrow-6ef46718c7b242e39fcad7f27ee858a5