# **Interactive Photo Frame in VR using Unity**

## Table of Contents

1. **Project Overview**
2. **Features**
3. **Technical Requirements**
4. **Implementation Details**
   1. Scene Setup
   2. UI Design
   3. Script Overview
   4. Interaction Mechanism
5. **Loading Photos**
6. **User Instructions**
7. **Conclusion**

## 1. Project Overview

This project involves developing an interactive photo frame in Unity that allows users to grab, move, scale, and rotate the frame. The photo frame features two buttons: one to toggle a floating effect and another to change the displayed photo. Photos are loaded at runtime from the device's storage (persistent data path).

## 2. Features

* **Interactive Photo Frame**: Users can interact with the frame using Controllers or Hands in VR.
* **Floating Effect**: A toggle button enables the frame to float in the air.
* **Change Photo Button**: Users can switch the displayed photo from loaded .png images.
* **Dynamic Photo Loading**: Images (.png) are fetched from the device's persistent storage.

## 3. Technical Requirements

* **Unity Version**: 2021.3.40
* **Target Platforms**: Android (Virtual Reality) - Oculus Quest, Quest2, Quest3
* **Required Packages**: Unity UI, OpenXR interaction toolkit

## Implementation Details

### Scene Setup

* 1. **Create a New Unity Project**.
  2. Ensure you have a photo frame 3D model and UI elements.
  3. **Scene Hierarchy**:
     1. Create an empty GameObject named GameController.
     2. Removed defulat MainCamera and Add XR Interaction Setup prefab in the hierarchy
     3. Create some default environtment for virtual reality experience
     4. Add the photo\_frame GameObject.
     5. Create UI elements (Buttons and warning pop up) under a UICanvas object.

### UI Design

* **Buttons**:
  + - **Float Button**: Toggles the floating state.
    - **Change Photo Button**: Cycles through the available photos.
* Warning popup to display when no .png images are available in the given path
* Text to display the loaded images path (for reference)

### Script Overview

Create a script named GameController with the following main key components:

* ***Public Variables****:*
  + GameObject PhotoFrameObj,WarningPopup
  + List Textures : To load and add all the .png images into this list in Start() method
* ***Private Variables****:*
  + int CurrentImage : Current Image number
  + Vector3 initPostion, initRotation : To save initial position and angle of the PhotoFrame
* ***Methods****:*
  + Start(): Initializes variables, loads textures from storage.
  + LoadTextures(): Loads PNG images from the device storage.
  + ChangeImage(): Changes the displayed photo.
  + FloatFunction(): Toggles the floating behavior.
  + FixedUpdate():To enable the floating action.

1. Interaction Mechanism

* **Controllers and Hands Input**:
  + Use Unity's OpenXR Input for handling grabbing, moving, scaling, and rotating the photo frame.
  + Utilize Rigidbody for physics interactions and Transform for manipulating position, rotation, and scale.

## **Loading Photos**

* Photos are loaded from the device's persistent storage using the Directory.GetFiles method to locate PNG files.
* Implement a coroutine to load all the images at once in the start function.

### Code Snippet for Loading Images

private IEnumerator LoadFile(string filePath)

{

if (string.IsNullOrEmpty(filePath))

yield break;

if (System.IO.File.Exists(filePath))

{

byte[] bytes = File.ReadAllBytes(filePath);

Texture2D loadedTexture = new Texture2D(1024, 1024, TextureFormat.RGB24, false);

loadedTexture.filterMode = FilterMode.Trilinear;

loadedTexture.LoadImage(bytes);

textures.Add(loadedTexture);

}

material.SetTexture("\_BaseMap", textures[0]); // Set the first texture

}

## **User Instructions**

* 1. **Interacting with the Frame**:
     1. Use single controller grip button to grab and move the photo frame.
     2. To scale and rotate by using two controllers grip buttons.
  2. **Toggle Floating**:
     1. Press on the “Float” button by using Trigger button in any controller to make the frame float.
     2. press again to restore its original position.
  3. **Change Photo**:
     1. Press on the “Change Photo” button by using Trigger button in any controller to cycle through the available images.

## **Conclusion**

The Interactive Photo Frame project offers a VR engaging experience, showcasing how to use Unity's features to create dynamic user interactions. With the ability to load images from device storage and manipulate the photo frame physically.