GENERAL VIEW

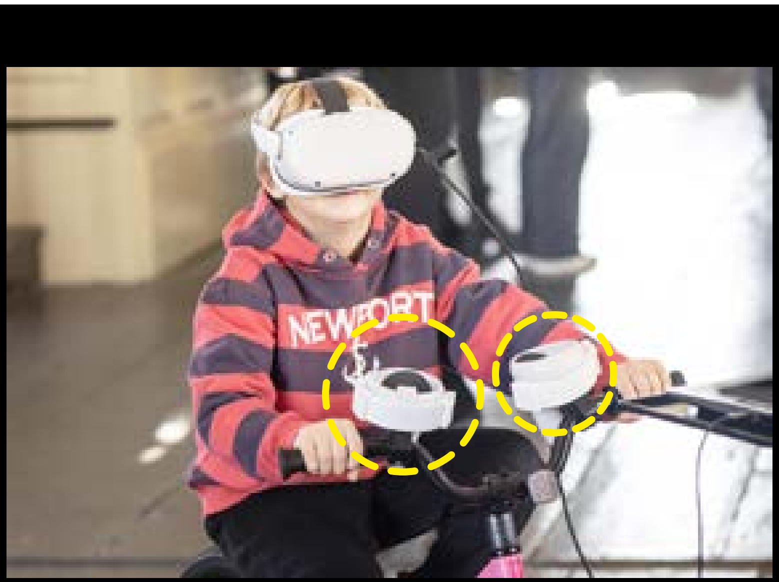
VIRTUAL RIDE PROGRAM. USERS EXPERIENCE RIDING ON THE MODIFIED BRIDGE ON BICYCLES.

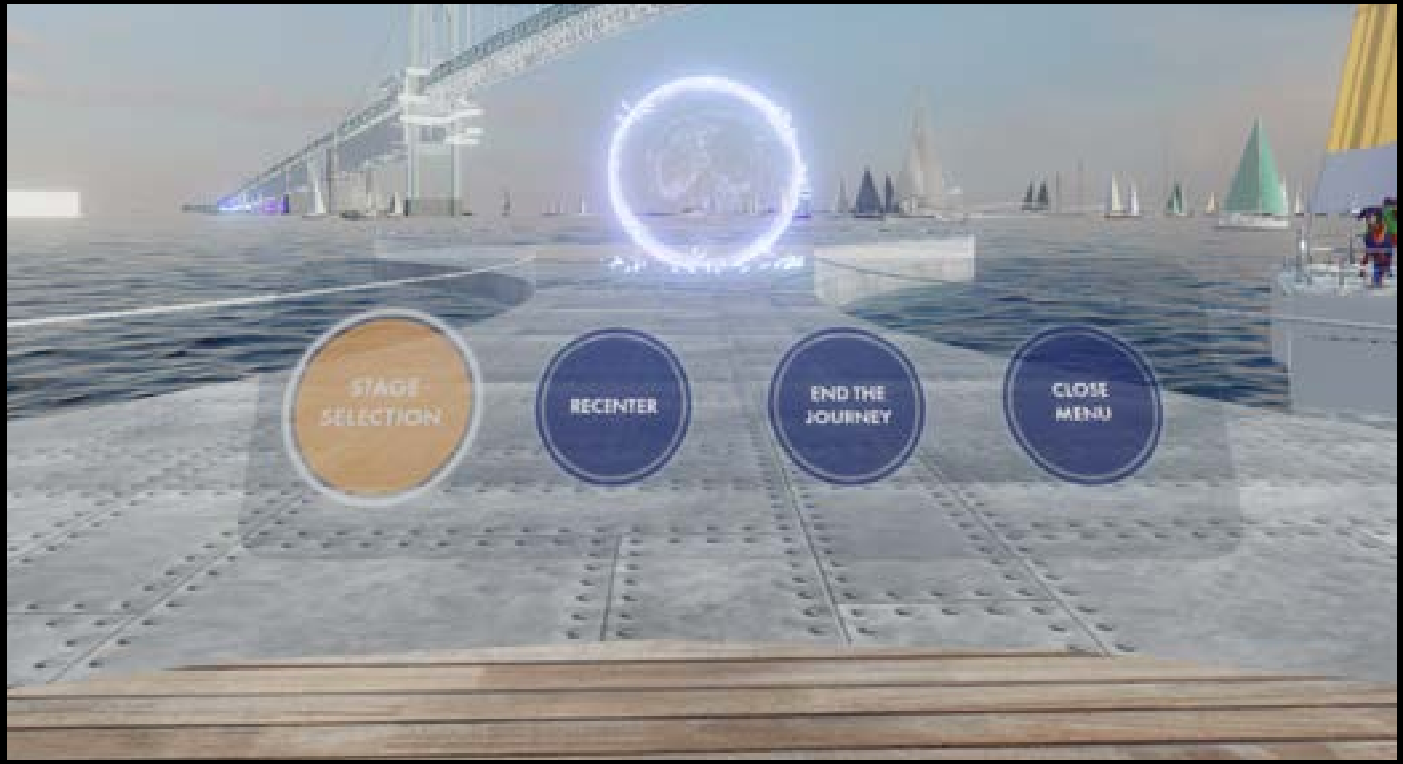
HTTPS://WWW.RISD.EDU/NEWS/STORIES/RISD-INTERIOR-ARCHITECTURE-STUDENTS-REIMAGINE-PELL-BRIDGE

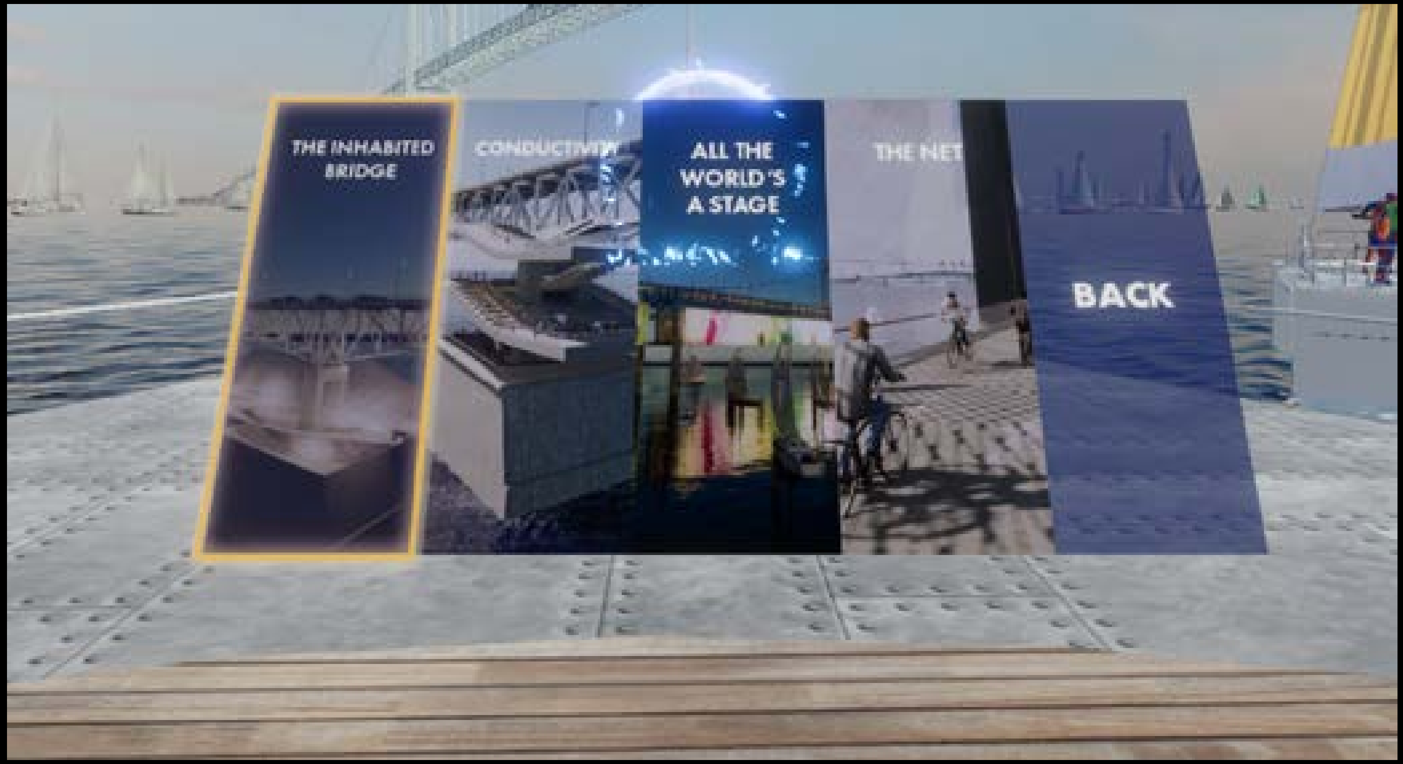
HTTPS://WWW.NEWPORTRI.COM/STORY/NEWS/LOCAL/2022/12/01/NEWPORT-PELL-BRIDGE-VR-EXPERIENCE-AT-OLD-COLONY-HOUSE/69683633007/
HTTPS://BIKENEWPORTRI.ORG/RISD-CROSSING-THE-PELL/

MENU FLOW DEVELOPMENT: CURRENT OPTION

A MENU IS DEVELOPED TO ALLOW THE USER TO SWITCH SCENES OR ADJUST THE POSITION OF THE HEADSET.



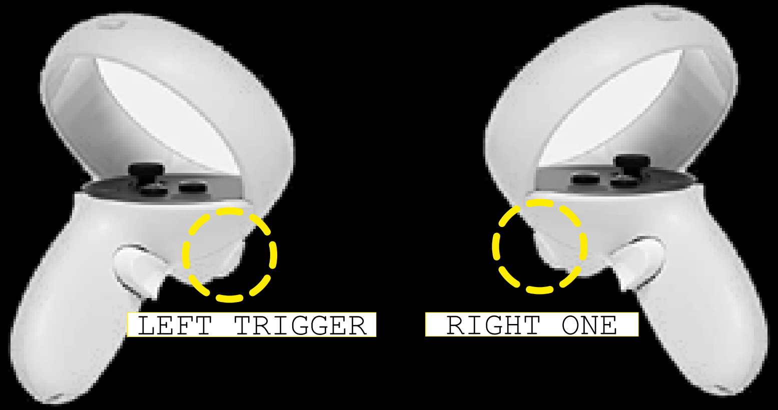




CHALLENGING INPUT MANAGEMENT

TWO CONTROLLERS ARE FIXED => NO RAYCASTING IN THE GAME
ONLY TWO TRIGGERS ARE USED =>

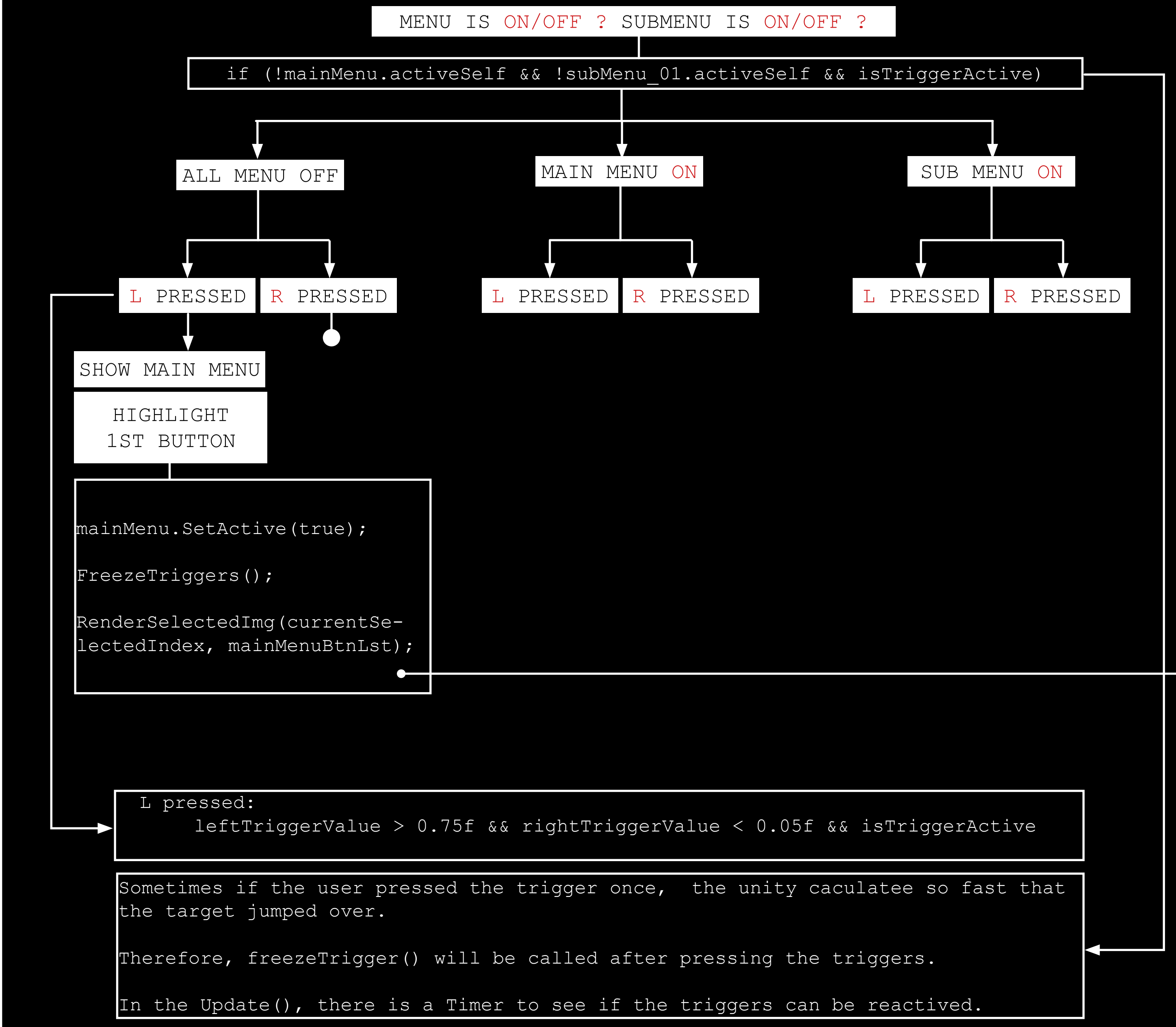
LFET TRIGGER: CALL MENU
CONFIRM THE BUTTON
RIGHT TRIGGER: SWITCH TO THE NEXT BUTTON



SPECIFIC TECHNICAL FLOW

KEEP GETTING THE INPUT FROM THE OCULUS CONTROLLERS

```
[SerializeField] InputActionProperty leftTriggerBtn;
[SerializeField] InputActionProperty rightTriggerBtn;

...

void Update()
{
    ...
    leftTriggerValue = leftTriggerBtn.action.ReadValue<float>();
    rightTriggerValue = rightTriggerBtn.action.ReadValue<float>();
    ...
```



MENU IS ON/OFF ? SUBMENU IS ON/OFF ?

```
if (!mainMenu.activeSelf && !subMenu_01.activeSelf && isTriggerActive)
```

```
mainMenu.SetActive(true);

FreezeTriggers();

RenderSelectedImg(currentSe-
lectedIndex, mainMenuBtnLst);
```

```
 L pressed:
     leftTriggerValue > 0.75f && rightTriggerValue < 0.05f && isTriggerActive
```

Sometimes if the user pressed the trigger once, the unity caculatee so fast that the target jumped over.

Therefore, freezeTrigger() will be called after pressing the triggers.

In the Update(), there is a Timer to see if the triggers can be reactivated.

CLASS VARIABLES

```
[SerializeField] InputActionProperty leftTriggerBtn;
[SerializeField] InputActionProperty rightTriggerBtn;
[SerializeField] InputActionProperty btnA;
[SerializeField] languageBool myLanguageBool;
```

```
List<GameObject> mainMenuBtnLst;
List<GameObject> sceneImgLst;
GameObject mainMenu;
GameObject subMenu_01;
GameObject leftController;
GameObject rightController;
const float SELECTEDIMGSCALERATIO = 1.15f;
const float TRIGGER_FREEZETIME = 0.3f;
float triggersFreezeStartTime;
bool isTriggerActive;

int currentSelectedIndex; // “cursor” position
float currentTime;
float leftTriggerValue;
float rightTriggerValue;
```

mainMenuBtnLst:
get all 4 buttons
in a list

HIGHTLIGHT THE BUTTON

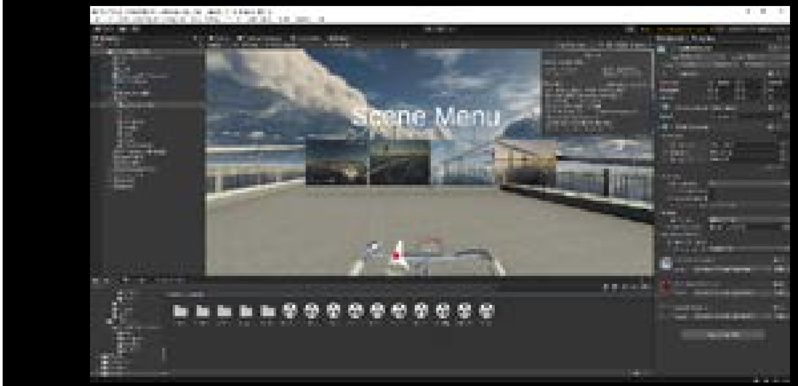
```
void RenderSelectedImg(int index, List<GameObject> imgList)
    {
        if(sceneImgLst.Count != 0)
        {
            // first, reset all img to (1f,1f,1f)
            foreach (GameObject obj in imgList)
            {
                obj.GetComponent<UnityEngine.UI.Image>().enabled = true;
            obj.transform.GetChild(1).GetComponent<SpriteRenderer>().enabled = false;
            }
            // then

            GameObject selectedSceneImg = imgList[index];
            selectedSceneImg.GetComponent<UnityEngine.UI.Image>().enabled = false;
            selectedSceneImg.transform.GetChild(1).GetComponent<SpriteRenderer>().enabled =
true;
            //selectedSceneImg.transform.localScale = new Vector3(SELECTEDIMGSCALERATIO, SELEC-
TEDIMGSCALERATIO, SELECTEDIMGSCALERATIO);
        }

    }
```

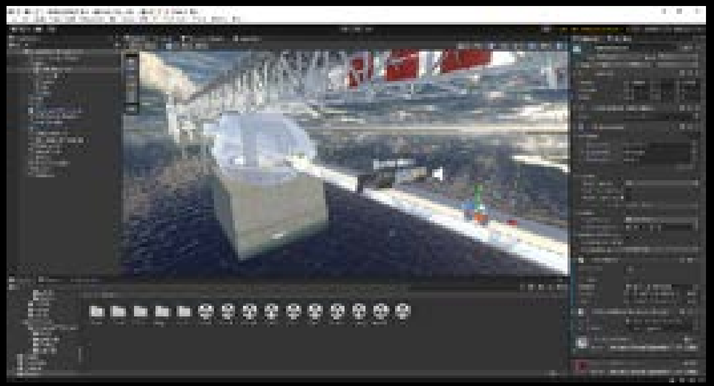
INPUT MANAGER C# FILE

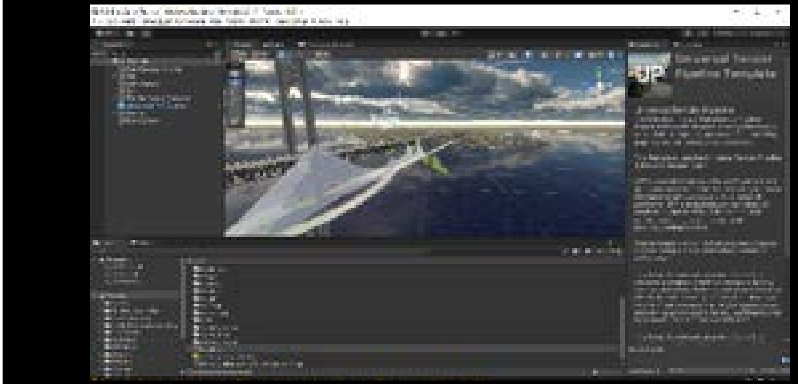
SOURCE CODE FILES
HTTPS://GITHUB.COM/DESIGNASABAKER/PELLUNITYSCRIPT

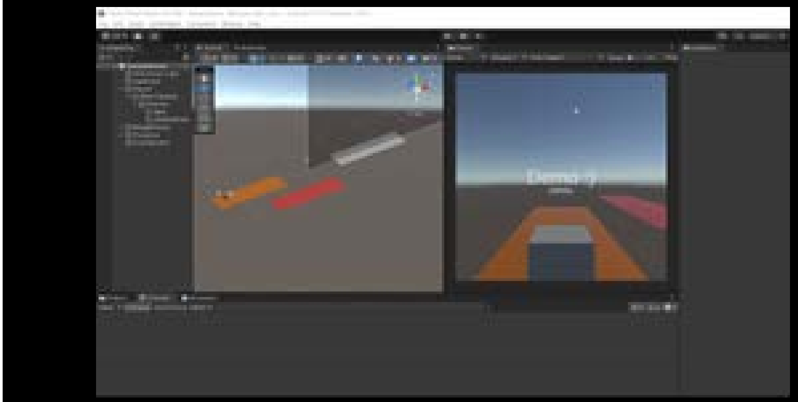
DEVELOPMENT

DURING THE DEVELOPMENT, MANY DIFFICULT PROBLEMS WERE SOLVED. IN ADDITION TO MENUS, THERE ARE PORTALS AND ANIMATIONS. THESE ARE TEST FILES WHILE DEVELOPING.









```
USING SYSTEM;
USING SYSTEM.COLLECTIONS;
USING SYSTEM.COLLECTIONS.GENERIC;
USING TMPRO;
USING UNITYENGINE;

PUBLIC CLASS PORTALMANAGER : MONOBEHAVIOUR
{
    DICTIONARY<GAMEOBJECT, GAMEOBJECT> DICTPORTALINOUT = NEW DICTION-
ARY<GAMEOBJECT, GAMEOBJECT>();

    [SERIALIZEFIELD] LIST<GAMEOBJECT> PORTALINLST = NEW LIST<GAMEOB-
JECT>();
    [SERIALIZEFIELD] LIST<GAMEOBJECT> PORTALOUTLST = NEW LIST<GAMEOB-
JECT>();
    // START IS CALLED BEFORE THE FIRST FRAME UPDATE
    VOID START()
    {
        DICTPORTALINOUT.CLEAR();
        INT PORTALLSTLEN = MATHF.MIN(PORTALINLST.COUNT, PORTALOUTLST.
COUNT);
        // IMPLEMENT THE DICTIONARY<GAMEOBJECT, GAMEOBJECT> DICTPORTAL-
INOUT
        FOR (INT I = 0; I < PORTALLSTLEN; I++) DICTPORTALINOUT.ADD(POR-
TALINLST[I], PORTALOUTLST[I]);
    }

    PUBLIC GAMEOBJECT LOOKUPPORTALOUT(GAMEOBJECT PORTALIN)
    {
        GAMEOBJECT PORTALOUT;
        PORTALOUT = DICTPORTALINOUT[PORTALIN];
        RETURN PORTALOUT;
    }
}
```

PORTAL MANAGER C# FILE

```
USING SYSTEM;
USING SYSTEM.COLLECTIONS;
USING SYSTEM.COLLECTIONS.GENERIC;
USING UNITYENGINE;
USING TMPRO;
USING UNITY.VISUALSCRIPTING;
USING UNITYEDITOR.RENDERING;
USING UNITYENGINE.SCENEMANAGEMENT;

PUBLIC CLASS TUTURIALMENUMANAGER20 : MONOBEHAVIOUR
{
    #REGION FIELD
    [SERIALIZEFIELD] CONTROLLERINPUTMANAGEMENT30 CONTROLLERINPUTMANAGER;
    [SERIALIZEFIELD] TEXTMESHPROUGUI TUTURIALMSGINFO;
    [SERIALIZEFIELD] LANGUAGEBOOL MYLB;

    DICTIONARY<STRING, STRING> MESSAGEINFODICT = NEW DICTIONARY<STRING,
STRING>();
    DICTIONARY<STRING, STRING> MESSAGEINFODICTEN = NEW DICTION-
ARY<STRING, STRING>();
    DICTIONARY<STRING, STRING> MESSAGEINFODICTSP = NEW DICTION-
ARY<STRING, STRING>();

    BOOL ISEN;

    INT HIGHLIGHTFONTSIZE = 36;

    BOOL HASSCENEMENUSHOWED;
    BOOL HASRIGHTTRIGGERCLICKEN;
    BOOL HASLEFTTRIGGERCLICKEN;
    BOOL HASMENUCLOSED;

    STRING LEFTTRIGGERCOLOR = “#DC143C”;
    STRING RIGHTTRIGGERCOLOR = “#ADD8E6”;

    // BOOL HASSUBSCENEMENUSHOWED;
    #ENDREGION
    #REGION METHOD
    // START IS CALLED BEFORE THE FIRST FRAME UPDATE
    VOID START()
    {
        ISEN = MYLB.ISEN;
        #REGION CHECKNULL
        IF (CONTROLLERINPUTMANAGER != NULL)
        {
            DEBUG.LOG($”---SUCCESS: CONTROLLERINPUTMANAGER LOADED---”);
        }
        ELSE
        {
            DEBUG.LOG($”---FAIL: CONTROLLERINPUTMANAGER NOT LOADED---”);
        }
        #ENDREGION
        IF (SCENEMANAGER.GETACTIVESCENE().NAME == “0_STARTINGSCENE 1”)
        {
            IF (ISEN)
            {
                STARTCOROUTINE(UPDATETUTURIALMSGBOX($”HELLO, WELCOME TO
THE MENU TUTORIAL. \NPLEASE PRESS <B><COLOR=RED><SIZE={HIGHLIGHTFONT-
SIZE}>LEFT</SIZE></COLOR></B> TRIGGERS TO CALL MENU”));
            }
            ELSE
            {
                STARTCOROUTINE(UPDATETUTURIALMSGBOX($”HOLA, BIENVEN-
IDO AL MEN□ TUTORIAL\NPRESIONE LOS DISPARADORES <B><COLOR={LEFTTRIG-
GERCOLOR}><SIZE={HIGHLIGHTFONTSIZE}>IZQUIERDOS</SIZE></COLOR></B> PARA
ACCEDER AL MEN□ <B>LLAMAR</B>”));
            }

        }
        ELSE {
            // IN OTHER SCENES, THERE IS NO NEED TO SHOWUP THE TUTORIAL
TEXTS
            TUTURIALMSGINFO.TEXT = “”;
        }
        #REGION IMPLEMENTDICT
        MESSAGEINFODICTEN.ADD(“TEACH TO SHOW THE MENU”,
            $”HI \NPLEASE PRESS <B><COLOR={LEFTTRIGGERCOLOR}><-
SIZE={HIGHLIGHTFONTSIZE}>LEFT</SIZE></COLOR></B> TRIGGERS TO \N<B>CALL</
B> MENU”);
        MESSAGEINFODICTEN.ADD(“TEACH TO SWITCH THE BUTTON”,
            $”GREAT! \NPRESS <B><COLOR={RIGHTTRIGGERCOLOR}><SIZE={HIGH-
LIGHTFONTSIZE}>RIGHT</SIZE></COLOR></B> TRIGGER TO \N<B>SWITCH</B> THE
SELECTED BUTTON”);
        MESSAGEINFODICTEN.ADD(“TEACH TO CONFIRM AND CLICK THE BUTTON”,
            $”AWESOME!\NPLEASE PRESS <B><COLOR={LEFTTRIGGERCOLOR}><-
SIZE={HIGHLIGHTFONTSIZE}>LEFT</SIZE></COLOR></B> TRIGGERS TO \N<B>LOAD</
B> THE SELECTED BUTTON”);
        MESSAGEINFODICTEN.ADD(“TEACH IN THE SUB MENU SCENE”,
            $”AWESOME!\NPLEASE PRESS <B><COLOR={LEFTTRIGGERCOLOR}><-
SIZE={HIGHLIGHTFONTSIZE}>LEFT</SIZE></COLOR></B> TRIGGERS TO \N<B>LOAD</
B> THE SELECTED BUTTON \N PRESS <B><COLOR=#ADD8E6><SIZE={HIGHLIGHTFONT-
SIZE}>RIGHT</SIZE></COLOR></B> TRIGGER TO \N<B>SWITCH</B> THE SELECTED
BUTTON”);

        // TO DO: IMPLEMENT THE SPANISH MENU TEXTS
        MESSAGEINFODICTSP.ADD(“TEACH TO SHOW THE MENU”,
            $”HOLA \NPRESIONE LOS DISPARADORES <B><COLOR={LEFTTRIGGER-
COLOR}><SIZE={HIGHLIGHTFONTSIZE}>IZQUIERDOS</SIZE></COLOR></B> PARA
ACCEDER AL MEN□ <B>LLAMAR</B>”);
        MESSAGEINFODICTSP.ADD(“TEACH TO SWITCH THE BUTTON”,
            $”□EXCELENTE! \NPRESIONE EL DISPARADOR <B><COLOR={RIGHT-
TRIGGERCOLOR}><SIZE={HIGHLIGHTFONTSIZE}>RIGHT</SIZE></COLOR></B> PARA \
N<B>CAMBIAR</B> EL BOT□N SELECCIONADO”);
        MESSAGEINFODICTSP.ADD(“TEACH TO CONFIRM AND CLICK THE BUTTON”,
            $”□IMPRESIONANTE!\NPRESIONE <B><COLOR={LEFTTRIGGERCOLOR}><-
SIZE{HIGHLIGHTFONTSIZE}>LEFT</SIZE></COLOR></B> DISPARADORES PARA \N<B>-
CARGAR</B> EL BOT□N SELECCIONADO”);
        MESSAGEINFODICTSP.ADD(“TEACH IN THE SUB MENU SCENE”,
            $”□IMPRESIONANTE!\NPRESIONE <B><COLOR={LEFTTRIGGER-
COLOR}><SIZE={HIGHLIGHTFONTSIZE}>LEFT</SIZE></COLOR></B> DISPARADORES
PARA \N<B>CARGAR</B> EL BOT□N SELECCIONADO \N PRESIONE EL DISPARADOR
<B><COLOR=#ADD8E6><SIZE={HIGHLIGHTFONTSIZE}>RIGHT</SIZE></COLOR></B>
PARA \N<B>CAMBIAR</B> EL BOT□N SELECCIONADO”);
        #ENDREGION
        MESSAGEINFODICT = ISEN ? MESSAGEINFODICTEN : MESSAGEINFODICTSP;
    }

    // UPDATE IS CALLED ONCE PER FRAME
    VOID UPDATE()
    {
        IF (CONTROLLERINPUTMANAGER != NULL)
        {
            HASSCENEMENUSHOWED = CONTROLLERINPUTMANAGER.HASSCENEMENUSH-
OWED;
            HASRIGHTTRIGGERCLICKEN = CONTROLLERINPUTMANAGER.HASRIGHT-
TRIGGERCLICKEN;
            HASLEFTTRIGGERCLICKEN = CONTROLLERINPUTMANAGER.HASLEFTTRIG-
GERCLICKEN;
            HASMENUCLOSED = CONTROLLERINPUTMANAGER.HASMENUCLOSED;
        }

        IF (!HASSCENEMENUSHOWED)
        {
            TUTURIALMSGINFO.TEXT = MESSAGEINFODICT[“TEACH TO SHOW THE
MENU”];
        }
        ELSE IF(HASSCENEMENUSHOWED && !HASRIGHTTRIGGERCLICKEN &&
!HASLEFTTRIGGERCLICKEN)
        {
            TUTURIALMSGINFO.TEXT = MESSAGEINFODICT[“TEACH TO SWITCH THE
BUTTON”];
        }
        ELSE IF(HASSCENEMENUSHOWED && HASRIGHTTRIGGERCLICKEN &&
!HASLEFTTRIGGERCLICKEN)
        {
            TUTURIALMSGINFO.TEXT = MESSAGEINFODICT[“TEACH TO CONFIRM AND
CLICK THE BUTTON”];
        }
        ELSE IF(HASSCENEMENUSHOWED && HASRIGHTTRIGGERCLICKEN && HASLEFT-
TRIGGERCLICKEN && !HASMENUCLOSED)
```

TUTORIAL MANAGER C# FILE

SOURCE CODE FILES
HTTPS://GITHUB.COM/DESIGNASABAKER/PELLUNITYSCRIPT