

DAFTAR PUSTAKA

- Adhitya (2017). *Penggunaan Augmented Reality sebagai Media Promosi Pariwisata di Kabupaten Karanganyar*. Skripsi. Tidak Diterbitkan. Fakultas Ilmu Komunikasi dan Informatika. Universitas Muhammadiyah : Surakarta.
- Amril. (2012, Agustus 25). *Fort Rotterdam : Benteng Termegah di Asia*. Retrived from Daeng Battala: <https://daengbattala.com/2012/10/fort-rotterdam-benteng-pertahanan-termegah-di-asia/>
- Arifitama, Budi (ed). 2015. *Panduan Mudah Membuat Augmented Reality*. Tangerang Selatan: Penerbit Andi
- Azuma, Ronald T. *A Survey of Augmented Reality*. Presence: Teleoperators and Virtual Environments, 1997.
- Daryanto. 2015. *Media Pembelajaran*. Bandung : Satu Nusa.
- Febiharsa, Dhega, Djuniadi. 2017. *Pengembangan Media Pembelajaran Interaktif 3 Dimensi untuk Pembelajaran Materi Pengenalan Lingkungan Pada Anak Usia Dini di Indonesia*. Journal of SCE (Studies in Early Childhood Education). 75-84
- Flavell, L. (2010). *Beginning Blender: Open Source 3D Modeling, Animation, and Game Design*. New York: Springer Science Business Media.
- Intan. (2013, September 4). *Mengulas 1 Buah Software Terbaru*. Retrived from Tugasku : <https://tugasku.forumid.net/t928p15-mengulas-1-buah-software-terbaru#4262>
- Nugraha I.S, Satoto K.I., Martono K.T. (2016). *Pemanfaatan Augmented Reality untuk Pembelajaran Pengenalan Alat Musik Piano*. *Teknologi dan Sistem Komputer*. Vol.2, No.1, 62-70.
- Machfoedz, Mahmud. (2010). *Komunikasi Pemasaran Modern*. Yogyakarta : Cakra Ilmu. Promosi.
- Mahendra, I. B. (2016). *Implementasi Augmented Reality (AR) Menggunakan Unity 3D dan Vuforia SDK* . *Jurnal Ilmiah ILMU KOMPUTER* Vol.9, No.1, 1-5.
- Martono, K. T. (2011). *Augmented Reality Sebagai Metafora Baru dalam Teknologi Interaksi Manusia dan Komputer* . *JURNAL SISTEM KOMPUTER* Vol.1, No.2 , 60-64.

- Mediaty, U.A., Wibawanto H., Azzizah L.N. (2019). *Membuat Augmented Reality(AR) dengan Unity 3D*. Semarang: Penerbit Andi
- Morrison. (2010). *Periklanan : Komunikasi Pemasaran Terpadu*. Jakarta : Kencana Prenada Media Group.
- Mustaqim I., & Nanang K. (2017). *Pengembangan Media pembelajaran Berbasis Android. Edukasi Elektro. Vol.1, No.1, 36-48*.
- Purwanto, E. (2014, Mei 12). *Cerita di Balik Software 3D Blender*. Retrieved from Bpptik: <https://bpptik.kominfo.go.id/2014/05/12/419/cerita-di-balik-software3d-blender/> diakses pada 21 Agustus 2020
- Ratno, 2012. Elearning Center. Yogyakarta : s.n. unity
- R. Silva, J. C. Oliveira, G. A. Giraldo (2000, September 10). *Introduction to Augmented Reality*. Retrieved from Incc : [http://www.incc.br/~jauvane/papers/Re latorioTecnicoLNCC-2503.pdf](http://www.incc.br/~jauvane/papers/Re%20latorioTecnicoLNCC-2503.pdf)
- Safaat, Nazruddin H. (2012). *Pemrograman Aplikasi Mobile Smartphone dan Tablet PC berbasis android*, Cetakan Pertama, Edisi Revisi, Penerbit Informatika Bandung. Bandung.
- Vaughan, T. 2006. *Multimedia : Making It Work*. Yogyakarta: Andi
- Widiya. (2019, Agustus 30). *Teknologi Augmented Reality*. Retrieved from Jagoan Hosting: <https://www.jagoanhosting.com/blog/teknologi-augmented-reality/> diakses pada 21 Agustus 2020
- Widyaningsih N.H. (2016). *Perancangan Augmented Reality sebagai Media Promosi Benteng Vestenburg di Kota Solo*. Fakultas Seni Rupa dan desain. Institut Seni Indonesia : Surakarta.
- Yunita C. U., Virginia T., Brave A. S. (2016). *Rancang Bangun Film Animasi 3 Dimensi Universitas Sam Ratulangi. Teknik Informatika. Vol .9, No.1, 1-7*
- Zulkarnaen, Rizky, 2010. *Perancangan Aplikasi Viewer Model 3D Interaktif Berbasis Web dengan Teknologi Augmented Reality*, Universitas Sumatera Utara, Medan, Indonesia.

LAMPIRAN

Lampiran 1. Script perpindahan scene

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

public class scenenext : MonoBehaviour
{
    public void LoadScene(string)
    {
        SceneManager.LoadScene("new");
    }
}
```

Lampiran 2. Script perpindahan panel

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

public class NextPanel : MonoBehaviour
{
    public GameObject panelAwal;
    public GameObject panelTujuan;

    public void GantiPanelBaru()
    {
        panelAwal.SetActive(false);
        panelTujuan.SetActive(true);
    }
}
```

Lampiran 3. Script exit device android

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

public class exit : MonoBehaviour
{
    public void QuitGame()
    {
        Application.Quit();
        Debug.Log("Exit");
    }
}
```

Lampiran 4. Script button info denah

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

public class info : MonoBehaviour
{
    public GameObject PanelInfo;

    private bool show = false;

    public void ShowHideInfo()
    {
        if (!show)
        {
            PanelInfo.SetActive(true);
            show = true;
        }
        else
        {
            PanelInfo.SetActive(false);
            show = false;
        }
    }
}
```

Lampiran 5. Script kembali dengan tombol device android

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.SceneManagement;
public class backbuttondevice : MonoBehaviour
{
    int sceneIndex;
    // Use this for initialization
    void Start ()
    {
        sceneIndex = SceneManager.GetActiveScene().
        buildIndex;
    }

    // Update is called once per frame
    void Update ()
    {
        if (Input.GetKeyDown(KeyCode.Escape))
            SceneManager.LoadScene("mainmenu");
    }
}
```

Lampiran 6. Script On Click Scale

```
using UnityEngine;
using System.Collections;

public class onclickscale : MonoBehaviour
{
    void OnMouseDown()
    {
        scalezoom.ScaleTransform = this.transform;
    }
}
```

Lampiran 7. Script Zoom Out In

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

public class scalezoom : MonoBehaviour
{

    public float initialFingersDistance;
    public Vector3 initialScale;
    public static Transform ScaleTransform;

    void Update()
    {
        int fingersOnScreen = 0;

        foreach (Touch touch in Input.touches)
        {
            fingersOnScreen++; //Count fingers (or rather touches) on screen as
            you iterate through all screen touches.

            //You need two fingers on screen to pinch.
            if (fingersOnScreen == 2)
            {

                //First set the initial distance between fingers so you can
                compare.
                if (touch.phase == TouchPhase.Began)
                {
                    initialFingersDistance =
                    Vector2.Distance(Input.touches[0].position, Input.touches[1].position);
                    initialScale = ScaleTransform.localScale;
                }
                else
                {
                    float currentFingersDistance =
                    Vector2.Distance(Input.touches[0].position, Input.touches[1].position);

                    float scaleFactor = currentFingersDistance /
                    initialFingersDistance;
                }
            }
        }
    }
}
```



```

{
#region ROTATE
private float _sensitivity = 1f;
private Vector3 _mouseReference;
private Vector3 _mouseOffset;
private Vector3 _rotation = Vector3.zero;
private bool _isRotating;

#endregion

void Update()
{
    if (_isRotating)
    {
        // offset
        _mouseOffset = (Input.mousePosition - _mouseReference);

        // apply rotation
        _rotation.y = -(_mouseOffset.x + _mouseOffset.y) *
        _sensitivity;

        // rotate
        gameObject.transform.Rotate(_rotation);

        // store new mouse position
        _mouseReference = Input.mousePosition;
    }
}

void OnMouseDown()
{
    // rotating flag
    _isRotating = true;

    // store mouse position
    _mouseReference = Input.mousePosition;
}

void OnMouseUp()
{
    // rotating flag
    _isRotating = false;
}
}

```

Lampiran 10. Script Virtual Button

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using Vuforia;

public class vbInteraction : MonoBehaviour, IVirtualButtonEventHandler
{
    private GameObject firstPanel, Panel2, Panel3, Panel4, Panel5, Panel6,

```

Panel17, Panel18, Panel19, Panel10, Panel11, Panel12, Panel13, Panel14, Panel15,
Panel16, Panel17, Panel18, Panel19, Panel20, Panel21, Panel22, Panel23,
Panel24, Panel25, Panel26, Panel27, Panel28, Panel29, Panel30, cube;

```
void Start()
{
    firstPanel = GameObject.Find("VirtualButton");

    firstPanel.GetComponent<VirtualButtonBehaviour>().RegisterEventHandler(
this);
    Panel2 = GameObject.Find("VirtualButton1");

    Panel2.GetComponent<VirtualButtonBehaviour>().RegisterEventHandler(this
);
    Panel3 = GameObject.Find("VirtualButton2");

    Panel3.GetComponent<VirtualButtonBehaviour>().RegisterEventHandler(this
);
    Panel4 = GameObject.Find("VirtualButton3");

    Panel4.GetComponent<VirtualButtonBehaviour>().RegisterEventHandler(this
);
    Panel5 = GameObject.Find("VirtualButton4");

    Panel5.GetComponent<VirtualButtonBehaviour>().RegisterEventHandler(this
);
    Panel6 = GameObject.Find("VirtualButton5");

    Panel6.GetComponent<VirtualButtonBehaviour>().RegisterEventHandler(this
);
    Panel7 = GameObject.Find("VirtualButton6");

    Panel7.GetComponent<VirtualButtonBehaviour>().RegisterEventHandler(this
);
    Panel8 = GameObject.Find("VirtualButton7");

    Panel8.GetComponent<VirtualButtonBehaviour>().RegisterEventHandler(this
);
    Panel9 = GameObject.Find("VirtualButton8");

    Panel9.GetComponent<VirtualButtonBehaviour>().RegisterEventHandler(this
);
    Panel10 = GameObject.Find("VirtualButton9");

    Panel10.GetComponent<VirtualButtonBehaviour>().RegisterEventHandler(thi
s);
    Panel11 = GameObject.Find("VirtualButton10");

    Panel11.GetComponent<VirtualButtonBehaviour>().RegisterEventHandler(thi
s);
    Panel12 = GameObject.Find("VirtualButton11");

    Panel12.GetComponent<VirtualButtonBehaviour>().RegisterEventHandler(thi
s);
    Panel13 = GameObject.Find("VirtualButton12");

    Panel13.GetComponent<VirtualButtonBehaviour>().RegisterEventHandler(thi
s);
    Panel14 = GameObject.Find("VirtualButton13");
```



```
Panel14.GetComponent<VirtualButtonBehaviour>().RegisterEventHandler(this);
    Panel15 = GameObject.Find("VirtualButton14");

Panel15.GetComponent<VirtualButtonBehaviour>().RegisterEventHandler(this);
    Panel16 = GameObject.Find("VirtualButton15");

Panel16.GetComponent<VirtualButtonBehaviour>().RegisterEventHandler(this);
    Panel17 = GameObject.Find("VirtualButton16");

Panel17.GetComponent<VirtualButtonBehaviour>().RegisterEventHandler(this);
    Panel18 = GameObject.Find("VirtualButton17");

Panel18.GetComponent<VirtualButtonBehaviour>().RegisterEventHandler(this);
    Panel19 = GameObject.Find("VirtualButton18");

Panel19.GetComponent<VirtualButtonBehaviour>().RegisterEventHandler(this);
    Panel20 = GameObject.Find("VirtualButton19");

Panel20.GetComponent<VirtualButtonBehaviour>().RegisterEventHandler(this);
    Panel21 = GameObject.Find("VirtualButton20");

Panel21.GetComponent<VirtualButtonBehaviour>().RegisterEventHandler(this);
    Panel22 = GameObject.Find("VirtualButton21");

Panel22.GetComponent<VirtualButtonBehaviour>().RegisterEventHandler(this);
    Panel23 = GameObject.Find("VirtualButton22");

Panel23.GetComponent<VirtualButtonBehaviour>().RegisterEventHandler(this);
    Panel24 = GameObject.Find("VirtualButton23");

Panel24.GetComponent<VirtualButtonBehaviour>().RegisterEventHandler(this);
    Panel25 = GameObject.Find("VirtualButton24");

Panel25.GetComponent<VirtualButtonBehaviour>().RegisterEventHandler(this);
    Panel26 = GameObject.Find("VirtualButton25");

Panel26.GetComponent<VirtualButtonBehaviour>().RegisterEventHandler(this);
    Panel27 = GameObject.Find("VirtualButton26");

Panel27.GetComponent<VirtualButtonBehaviour>().RegisterEventHandler(this);
    Panel28 = GameObject.Find("VirtualButton27");

Panel28.GetComponent<VirtualButtonBehaviour>().RegisterEventHandler(this);
```

```

        Panel29 = GameObject.Find("VirtualButton28");

Panel29.GetComponent<VirtualButtonBehaviour>().RegisterEventHandler(this);
        Panel30 = GameObject.Find("VirtualButton29");

Panel30.GetComponent<VirtualButtonBehaviour>().RegisterEventHandler(this);
        cube = GameObject.Find("Canvas");
        cube.SetActive(false);
    }

public void OnButtonPressed(VirtualButtonBehaviour vb)
    {
        cube.SetActive(true);
    }

public void OnButtonReleased(VirtualButtonBehaviour vb)
    {
        cube.SetActive(false);
    }
}

```

Lampiran 11. Script Find Vumark

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using Vuforia;

public class VuMarkEvent : MonoBehaviour
{
    public List<GameObject> modelList;
    public List<string> modelIdList;
    private int modelIN;
    private VuMarkManager vuMarkManager;

    void Start()
    {
        // Set VuMarkManager
        vuMarkManager =
TrackerManager.Instance.GetStateManager().GetVuMarkManager();
        // Set VuMark detected and lost behavior methods
        vuMarkManager.RegisterVuMarkDetectedCallback(onVuMarkDetected);
        vuMarkManager.RegisterVuMarkLostCallback(onVuMarkLost);

        // Deactivate all models
        foreach (GameObject item in modelList)
        {
            item.SetActive(false);
        }
    }

    void Update()
    {

```

```

        /*          foreach (var vmb in
vuMarkManager.GetActiveBehaviours()) {
                    Debug.Log ("ID: "+
getVuMarkID(vmb.VuMarkTarget));
                }
        */
    }

private string getVuMarkID(VuMarkTarget vuMark)
{
    switch (vuMark.InstanceId.DataType)
    {
        case InstanceIdType.BYTES:
            return vuMark.InstanceId.HexStringValue;
        case InstanceIdType.STRING:
            return vuMark.InstanceId.StringValue;
        case InstanceIdType.NUMERIC:
            return vuMark.InstanceId.NumericValue.ToString();
    }

    return null;
}

public void onVuMarkDetected(VuMarkTarget target)
{
    Debug.Log("Detected ID: " + getVuMarkID(target));
    //          Debug.Log (target.Template.VuMarkUserData);

    // Find and activate model by VuMark ID
    for (int i = 0; i < modelIdList.Count; i++)
    {
        if (modelIdList[i] == getVuMarkID(target))
        {
            modelList[i].SetActive(true);

            // Set model number
            modelN = i;
        }
    }
}

public void onVuMarkLost(VuMarkTarget target)
{
    Debug.Log("Lost ID: " + getVuMarkID(target));
    // Deactivate model by model number
    modelList[modelN].SetActive(false);
}
}

```

Lampiran 12. Script Information Building

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using Vuforia;

```

```

public class datatarget : MonoBehaviour
{
    private GameObject bg_label1;
    private GameObject bg_label2;
    private GameObject bg_label3;
    private GameObject bg_label4;
    private GameObject bg_label5;
    private GameObject bg_label6;
    private GameObject bg_label7;
    private GameObject bg_label8;
    private GameObject bg_label9;
    private GameObject bg_label10;
    private GameObject bg_label11;
    private GameObject bg_label12;
    private GameObject bg_label13;
    private GameObject bg_label14;
    private GameObject bg_label15;
    private GameObject bg_label16;
    private GameObject groundd;

    // Use this for initialization
    void Start()
    {
        bg_label1 = GameObject.Find("bglabel");
        bg_label1.SetActive(false);
        bg_label2 = GameObject.Find("bglabel2");
        bg_label2.SetActive(false);
        bg_label3 = GameObject.Find("bglabel3");
        bg_label3.SetActive(false);
        bg_label4 = GameObject.Find("bglabel4");
        bg_label4.SetActive(false);
    }
}

```

```

    bg_label15 = GameObject.Find("bglabel5");
    bg_label15.SetActive(false);
    bg_label16 = GameObject.Find("bglabel6");
    bg_label16.SetActive(false);
    bg_label17 = GameObject.Find("bglabel7");
    bg_label17.SetActive(false);
    bg_label18 = GameObject.Find("bglabel8");
    bg_label18.SetActive(false);
    bg_label19 = GameObject.Find("bglabel9");
    bg_label19.SetActive(false);
    bg_label10 = GameObject.Find("bglabel10");
    bg_label10.SetActive(false);
    bg_label11 = GameObject.Find("bglabel11");
    bg_label11.SetActive(false);
    bg_label12 = GameObject.Find("bglabel12");
    bg_label12.SetActive(false);
    bg_label13 = GameObject.Find("bglabel13");
    bg_label13.SetActive(false);
    bg_label14 = GameObject.Find("bglabel14");
    bg_label14.SetActive(false);
    bg_label15 = GameObject.Find("bglabel15");
    bg_label15.SetActive(false);
    bg_label16 = GameObject.Find("bglabel16");
    bg_label16.SetActive(false);
    groundd = GameObject.Find("gd");
    groundd.SetActive(false);
}

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

```

```

if ((Input.GetTouch(0).phase == TouchPhase.Stationary) ||
    (Input.GetTouch(0).phase == TouchPhase.Moved &&
    Input.GetTouch(0).deltaPosition.magnitude < 1.2f))
{
    Ray ray = Camera.main.ScreenPointToRay(Input.GetTouch(0).position);
    RaycastHit hit;
    if (Physics.Raycast(ray, out hit))
    {
        if (hit.collider.gameObject.name == "bangunanA")
        {
            bg_label.SetActive(true);
            bg_label2.SetActive(false);
            bg_label3.SetActive(false);
            bg_label4.SetActive(false);
            bg_label5.SetActive(false);
            bg_label6.SetActive(false);
            bg_label7.SetActive(false);
            bg_label8.SetActive(false);
            bg_label9.SetActive(false);
            bg_label10.SetActive(false);
            bg_label11.SetActive(false);
            bg_label12.SetActive(false);
            bg_label13.SetActive(false);
            bg_label14.SetActive(false);
            bg_label15.SetActive(false);
            bg_label16.SetActive(false);
            groundd.SetActive(false);
        }
        if (hit.collider.gameObject.name == "bangunanB")

```

```

{
    bg_label1.SetActive(false);
    bg_label2.SetActive(true);
    bg_label3.SetActive(false);
    bg_label4.SetActive(false);
    bg_label5.SetActive(false);
    bg_label6.SetActive(false);
    bg_label7.SetActive(false);
    bg_label8.SetActive(false);
    bg_label9.SetActive(false);
    bg_label10.SetActive(false);
    bg_label11.SetActive(false);
    bg_label12.SetActive(false);
    bg_label13.SetActive(false);
    bg_label14.SetActive(false);
    bg_label15.SetActive(false);
    bg_label16.SetActive(false);
    groundd.SetActive(false);
}
if (hit.collider.gameObject.name == "bangunanC")
{
    bg_label1.SetActive(false);
    bg_label2.SetActive(false);
    bg_label3.SetActive(true);
    bg_label4.SetActive(false);
    bg_label5.SetActive(false);
    bg_label6.SetActive(false);
    bg_label7.SetActive(false);
    bg_label8.SetActive(false);
    bg_label9.SetActive(false);

```

```

        bg_label10.SetActive(false);
        bg_label11.SetActive(false);
        bg_label12.SetActive(false);
        bg_label13.SetActive(false);
        bg_label14.SetActive(false);
        bg_label15.SetActive(false);
        bg_label16.SetActive(false);
        groundd.SetActive(false);
    }
    if (hit.collider.gameObject.name == "bangunanD")
    {
        bg_label1.SetActive(false);
        bg_label2.SetActive(false);
        bg_label3.SetActive(false);
        bg_label4.SetActive(true);
        bg_label5.SetActive(false);
        bg_label6.SetActive(false);
        bg_label7.SetActive(false);
        bg_label8.SetActive(false);
        bg_label9.SetActive(false);
        bg_label10.SetActive(false);
        bg_label11.SetActive(false);
        bg_label12.SetActive(false);
        bg_label13.SetActive(false);
        bg_label14.SetActive(false);
        bg_label15.SetActive(false);
        bg_label16.SetActive(false);
        groundd.SetActive(false);
    }
    if (hit.collider.gameObject.name == "bangunanE")

```



```
{
    bg_label1.SetActive(false);
    bg_label2.SetActive(false);
    bg_label3.SetActive(false);
    bg_label4.SetActive(false);
    bg_label5.SetActive(true);
    bg_label6.SetActive(false);
    bg_label7.SetActive(false);
    bg_label8.SetActive(false);
    bg_label9.SetActive(false);
    bg_label10.SetActive(false);
    bg_label11.SetActive(false);
    bg_label12.SetActive(false);
    bg_label13.SetActive(false);
    bg_label14.SetActive(false);
    bg_label15.SetActive(false);
    bg_label16.SetActive(false);
    groundd.SetActive(false);
}
if (hit.collider.gameObject.name == "bangunanF")
{
    bg_label1.SetActive(false);
    bg_label2.SetActive(false);
    bg_label3.SetActive(false);
    bg_label4.SetActive(false);
    bg_label5.SetActive(false);
    bg_label6.SetActive(true);
    bg_label7.SetActive(false);
    bg_label8.SetActive(false);
    bg_label9.SetActive(false);
```

```

        bg_label10.SetActive(false);
        bg_label11.SetActive(false);
        bg_label12.SetActive(false);
        bg_label13.SetActive(false);
        bg_label14.SetActive(false);
        bg_label15.SetActive(false);
        bg_label16.SetActive(false);
        groundd.SetActive(false);
    }
    if (hit.collider.gameObject.name == "bangunanG")
    {
        bg_label1.SetActive(false);
        bg_label2.SetActive(false);
        bg_label3.SetActive(false);
        bg_label4.SetActive(false);
        bg_label5.SetActive(false);
        bg_label6.SetActive(false);
        bg_label7.SetActive(true);
        bg_label8.SetActive(false);
        bg_label9.SetActive(false);
        bg_label10.SetActive(false);
        bg_label11.SetActive(false);
        bg_label12.SetActive(false);
        bg_label13.SetActive(false);
        bg_label14.SetActive(false);
        bg_label15.SetActive(false);
        bg_label16.SetActive(false);
        groundd.SetActive(false);
    }
    if (hit.collider.gameObject.name == "bangunanH")

```

```
{
    bg_label1.SetActive(false);
    bg_label2.SetActive(false);
    bg_label3.SetActive(false);
    bg_label4.SetActive(false);
    bg_label5.SetActive(false);
    bg_label6.SetActive(false);
    bg_label7.SetActive(false);
    bg_label8.SetActive(true);
    bg_label9.SetActive(false);
    bg_label10.SetActive(false);
    bg_label11.SetActive(false);
    bg_label12.SetActive(false);
    bg_label13.SetActive(false);
    bg_label14.SetActive(false);
    bg_label15.SetActive(false);
    bg_label16.SetActive(false);
    groundd.SetActive(false);
}
if (hit.collider.gameObject.name == "bangunanI")
{
    bg_label1.SetActive(false);
    bg_label2.SetActive(false);
    bg_label3.SetActive(false);
    bg_label4.SetActive(false);
    bg_label5.SetActive(false);
    bg_label6.SetActive(false);
    bg_label7.SetActive(false);
    bg_label8.SetActive(false);
    bg_label9.SetActive(true);
```

```

        bg_label10.SetActive(false);
        bg_label11.SetActive(false);
        bg_label12.SetActive(false);
        bg_label13.SetActive(false);
        bg_label14.SetActive(false);
        bg_label15.SetActive(false);
        bg_label16.SetActive(false);
        groundd.SetActive(false);
    }
    if (hit.collider.gameObject.name == "bangunanJ")
    {
        bg_label1.SetActive(false);
        bg_label2.SetActive(false);
        bg_label3.SetActive(false);
        bg_label4.SetActive(false);
        bg_label5.SetActive(false);
        bg_label6.SetActive(false);
        bg_label7.SetActive(false);
        bg_label8.SetActive(false);
        bg_label9.SetActive(false);
        bg_label10.SetActive(true);
        bg_label11.SetActive(false);
        bg_label12.SetActive(false);
        bg_label13.SetActive(false);
        bg_label14.SetActive(false);
        bg_label15.SetActive(false);
        bg_label16.SetActive(false);
        groundd.SetActive(false);
    }
    if (hit.collider.gameObject.name == "bangunanK")

```

```
{
    bg_label.SetActive(false);
    bg_label2.SetActive(false);
    bg_label3.SetActive(false);
    bg_label4.SetActive(false);
    bg_label5.SetActive(false);
    bg_label6.SetActive(false);
    bg_label7.SetActive(false);
    bg_label8.SetActive(false);
    bg_label9.SetActive(false);
    bg_label10.SetActive(false);
    bg_label11.SetActive(true);
    bg_label12.SetActive(false);
    bg_label13.SetActive(false);
    bg_label14.SetActive(false);
    bg_label15.SetActive(false);
    bg_label16.SetActive(false);
    groundd.SetActive(false);
}
if (hit.collider.gameObject.name == "bangunanL")
{
    bg_label.SetActive(false);
    bg_label2.SetActive(false);
    bg_label3.SetActive(false);
    bg_label4.SetActive(false);
    bg_label5.SetActive(false);
    bg_label6.SetActive(false);
    bg_label7.SetActive(false);
    bg_label8.SetActive(false);
    bg_label9.SetActive(false);
```

```

        bg_label10.SetActive(false);
        bg_label11.SetActive(false);
        bg_label12.SetActive(true);
        bg_label13.SetActive(false);
        bg_label14.SetActive(false);
        bg_label15.SetActive(false);
        bg_label16.SetActive(false);
        groundd.SetActive(false);
    }
    if (hit.collider.gameObject.name == "bangunanM")
    {
        bg_label1.SetActive(false);
        bg_label2.SetActive(false);
        bg_label3.SetActive(false);
        bg_label4.SetActive(false);
        bg_label5.SetActive(false);
        bg_label6.SetActive(false);
        bg_label7.SetActive(false);
        bg_label8.SetActive(false);
        bg_label9.SetActive(false);
        bg_label10.SetActive(false);
        bg_label11.SetActive(false);
        bg_label12.SetActive(false);
        bg_label13.SetActive(true);
        bg_label14.SetActive(false);
        bg_label15.SetActive(false);
        bg_label16.SetActive(false);
        groundd.SetActive(false);
    }
    if (hit.collider.gameObject.name == "bangunanN")

```

```
{
    bg_label1.SetActive(false);
    bg_label2.SetActive(false);
    bg_label3.SetActive(false);
    bg_label4.SetActive(false);
    bg_label5.SetActive(false);
    bg_label6.SetActive(false);
    bg_label7.SetActive(false);
    bg_label8.SetActive(false);
    bg_label9.SetActive(false);
    bg_label10.SetActive(false);
    bg_label11.SetActive(false);
    bg_label12.SetActive(false);
    bg_label13.SetActive(false);
    bg_label14.SetActive(true);
    bg_label15.SetActive(false);
    bg_label16.SetActive(false);
    groundd.SetActive(false);
}
if (hit.collider.gameObject.name == "bangunan0")
{
    bg_label1.SetActive(false);
    bg_label2.SetActive(false);
    bg_label3.SetActive(false);
    bg_label4.SetActive(false);
    bg_label5.SetActive(false);
    bg_label6.SetActive(false);
    bg_label7.SetActive(false);
    bg_label8.SetActive(false);
    bg_label9.SetActive(false);
```

```

        bg_label10.SetActive(false);
        bg_label11.SetActive(false);
        bg_label12.SetActive(false);
        bg_label13.SetActive(false);
        bg_label14.SetActive(false);
        bg_label15.SetActive(true);
        bg_label16.SetActive(false);
        groundd.SetActive(false);
    }
    if (hit.collider.gameObject.name == "bangunanP")
    {
        bg_label1.SetActive(false);
        bg_label2.SetActive(false);
        bg_label3.SetActive(false);
        bg_label4.SetActive(false);
        bg_label5.SetActive(false);
        bg_label6.SetActive(false);
        bg_label7.SetActive(false);
        bg_label8.SetActive(false);
        bg_label9.SetActive(false);
        bg_label10.SetActive(false);
        bg_label11.SetActive(false);
        bg_label12.SetActive(false);
        bg_label13.SetActive(false);
        bg_label14.SetActive(false);
        bg_label15.SetActive(false);
        bg_label16.SetActive(true);
        groundd.SetActive(false);
    }
    if (hit.collider.gameObject.name == "ground")

```



```

        if (vuforia != null)
            vuforia.RegisterVuforiaStartedCallback(StartAfterVuforia);
    }

    private void StartAfterVuforia()
    {
        mVuforiaStarted = true;
        SetAutofocus();
    }

    void OnApplicationPause(bool pause)
    {
        if (!pause)
        {
            // App resumed
            if (mVuforiaStarted)
                // App resumed and vuforia already started
                // but lets start it again...
                SetAutofocus(); // This is done because some android devices
lose the auto focus after resume
                // this was a bug in vuforia 4 and 5. I haven't checked 6, but
the code is harmless anyway
        }
    }

    private void SetAutofocus()
    {
        if
            (CameraDevice.Instance.SetFocusMode(CameraDevice.FocusMode.FOCUS_MODE_CONT
INUOUSAUTO))
            Debug.Log("Autofocus set");

        else
            // never actually seen a device that doesn't support this, but just in
case
            Debug.Log("this device doesn't support auto focus");
    }
}

```

Lampiran 14. Script AR Camera

```

using UnityEngine;
using Vuforia;

public class DefaultInitializationErrorHandler : VuforiaMonoBehaviour
{
    #region Vuforia_lifecycle_events

    public void OnVuforiaInitializationError(VuforiaUnity.InitError initError)
    {
        if (initError != VuforiaUnity.InitError.INIT_SUCCESS)
        {
            SetErrorCode(initError);
            SetErrorOccurred(true);
        }
    }
}

```

```

    }
}

#endregion // Vuforia_lifecycle_events

#region PRIVATE_MEMBER_VARIABLES

string mErrorText = "";
bool mErrorOccurred;

const string headerLabel = "Vuforia Engine Initialization Error";

GUIStyle bodyStyle;
GUIStyle headerStyle;
GUIStyle footerStyle;

Texture2D bodyTexture;
Texture2D headerTexture;
Texture2D footerTexture;

#endregion // PRIVATE_MEMBER_VARIABLES

#region UNITY_MONOBEHAVIOUR_METHODS

void Awake()
{
    // Check for an initialization error on start.
    VuforiaRuntime.Instance.RegisterVuforiaInitErrorCallback(OnVuforiaInitializationError);
}

void Start()
{
    SetupGUIStyles();
}

void OnGUI()
{
    // On error, create a full screen window.
    if (mErrorOccurred)
        GUI.Window(0, new Rect(0, 0, Screen.width, Screen.height),
        DrawWindowContent, "");
}

/// <summary>
///     When this game object is destroyed, it unregisters itself as event
handler
/// </summary>
void OnDestroy()
{
    VuforiaRuntime.Instance.UnregisterVuforiaInitErrorCallback(OnVuforiaInitializationError);
}

#endregion // UNITY_MONOBEHAVIOUR_METHODS

#region PRIVATE_METHODS

```

```

void DrawWindowContent(int id)
{
    var headerRect = new Rect(0, 0, Screen.width, Screen.height / 8);
    var bodyRect = new Rect(0, Screen.height / 8, Screen.width,
Screen.height / 8 * 6);
    var footerRect = new Rect(0, Screen.height - Screen.height / 8,
Screen.width, Screen.height / 8);

    GUI.Label(headerRect, headerLabel, headerStyle);
    GUI.Label(bodyRect, mErrorText, bodyStyle);

    if (GUI.Button(footerRect, "Close", footerStyle))
    {
#ifdef UNITY_EDITOR

                UnityEditor.EditorApplication.isPlaying = false;
#else
                Application.Quit();
#endif
    }
}

void SetErrorCode(VuforiaUnity.InitError errorCode)
{
    switch (errorCode)
    {
        case VuforiaUnity.InitError.INIT_EXTERNAL_DEVICE_NOT_DETECTED:
            mErrorText =
                "Failed to initialize the Vuforia Engine because this " +
                "device is not docked with required external hardware.";
            break;
        case VuforiaUnity.InitError.INIT_LICENSE_ERROR_MISSING_KEY:
            mErrorText =
                "Vuforia Engine App key is missing. Please get a valid key
" +
                "by logging into your account at developer.vuforia.com " +
                "and creating a new project.";
            break;
        case VuforiaUnity.InitError.INIT_LICENSE_ERROR_INVALID_KEY:
            mErrorText =
                "Vuforia Engine App key is invalid. " +
                "Please get a valid key by logging into your account at " +
                "developer.vuforia.com and creating a new project. \n\n" +
                getKeyInfo();
            break;
        case
VuforiaUnity.InitError.INIT_LICENSE_ERROR_NO_NETWORK_TRANSIENT:
            mErrorText = "Unable to contact server. Please try again
later.";
            break;
        case
VuforiaUnity.InitError.INIT_LICENSE_ERROR_NO_NETWORK_PERMANENT:
            mErrorText = "No network available. Please make sure you are
connected to the Internet.";
            break;
        case VuforiaUnity.InitError.INIT_LICENSE_ERROR_CANCELED_KEY:
            mErrorText =

```

```

        "This App license key has been cancelled and may no longer
be used. " +
        "Please get a new license key. \n\n" +
        getKeyInfo();
        break;
    case
VuforiaUnity.InitError.INIT_LICENSE_ERROR_PRODUCT_TYPE_MISMATCH:
        mErrorText =
        "Vuforia Engine App key is not valid for this product.
Please get a valid key " +
        "by logging into your account at developer.vuforia.com and
choosing the " +
        "right product type during project creation. \n\n" +

```

```

        getKeyInfo() + " \n\n" +
        "Note that Universal Windows Platform (UWP) apps require "
+
        "a license key created on or after August 9th, 2016.";
        break;
    case VuforiaUnity.InitError.INIT_NO_CAMERA_ACCESS:
        mErrorText =
            "User denied Camera access to this app.\n" +
            "To restore, enable Camera access in Settings:\n" +
            "Settings > Privacy > Camera > " + Application.productName
+ "\n" +
            "Also verify that the Camera is enabled in:\n" +
            "Settings > General > Restrictions.";
        break;
    case VuforiaUnity.InitError.INIT_DEVICE_NOT_SUPPORTED:
        mErrorText = "Failed to initialize Vuforia Engine because this
device is not supported.";
        break;
    case VuforiaUnity.InitError.INIT_ERROR:
        mErrorText = "Failed to initialize Vuforia Engine.";
        break;
    }

    // Prepend the error code in red
    mErrorText = "<color=red>" + errorCode.ToString().Replace("_", " ") +
"</color>\n\n" + mErrorText;

    // Remove rich text tags for console logging
    var errorTextConsole = mErrorText.Replace("<color=red>",
"").Replace("</color>", "");

    Debug.LogError("Vuforia Engine initialization failed: " + errorCode +
"\n\n" + errorTextConsole);
}

void SetErrorOccurred(bool errorOccurred)
{
    mErrorOccurred = errorOccurred;
}

string getKeyInfo()
{
    string key = VuforiaConfiguration.Instance.Vuforia.LicenseKey;
    string keyInfo;
    if (key.Length > 10)
        keyInfo =
            "Your current key is <color=red>" + key.Length + "</color>
characters in length. " +
            "It begins with <color=red>" + key.Substring(0, 5) + "</color>
" +
            "and ends with <color=red>" + key.Substring(key.Length - 5, 5)
+ "</color>.";
    else
        keyInfo =

```

```

        "Your current key is <color=red>" + key.Length + "</color>
characters in length. \n" +
        "The key is: <color=red>" + key + "</color>.";
    return keyInfo;
}

void SetupGUIStyles()
{
    // Called from Start() to determine physical size of device for text
sizing
    var shortSidePixels = Screen.width < Screen.height ? Screen.width :
Screen.height;
    var shortSideInches = shortSidePixels / Screen.dpi;
    var physicalSizeMultiplier = shortSideInches > 4.0f ? 2 : 1;

    // Create 1x1 pixel background textures for body, header, and footer
bodyTexture = CreateSinglePixelTexture(Color.white);
headerTexture = CreateSinglePixelTexture(new Color(
    Mathf.InverseLerp(0, 255, 220),
    Mathf.InverseLerp(0, 255, 220),
    Mathf.InverseLerp(0, 255, 220))); // RGB(220)
footerTexture = CreateSinglePixelTexture(new Color(
    Mathf.InverseLerp(0, 255, 35),
    Mathf.InverseLerp(0, 255, 178),
    Mathf.InverseLerp(0, 255, 220))); // RGB(35, 178, 220)

    // Create body style and set values
bodyStyle = new GUIStyle();
bodyStyle.normal.background = bodyTexture;
bodyStyle.font = Resources.GetBuiltinResource<Font>("Arial.ttf");
bodyStyle.fontSize = (int) (18 * physicalSizeMultiplier * Screen.dpi /
160);
bodyStyle.normal.textColor = Color.black;
bodyStyle.wordWrap = true;
bodyStyle.alignment = TextAnchor.MiddleCenter;
bodyStyle.padding = new RectOffset(40, 40, 0, 0);

    // Duplicate body style and change necessary values
headerStyle = new GUIStyle(bodyStyle);
headerStyle.normal.background = headerTexture;
headerStyle.fontSize = (int) (24 * physicalSizeMultiplier * Screen.dpi
/ 160);

    // Duplicate body style and change necessary values
footerStyle = new GUIStyle(bodyStyle);
footerStyle.normal.background = footerTexture;
footerStyle.normal.textColor = Color.white;
footerStyle.fontSize = (int) (28 * physicalSizeMultiplier * Screen.dpi
/ 160);
}

```

```
Texture2D CreateSinglePixelTexture(Color color)
{
    // Called by SetupGUIStyles() to create 1x1 texture
    var texture = new Texture2D(1, 1, TextureFormat.ARGB32, false);

    texture.SetPixel(0, 0, color);
    texture.Apply();
    return texture;
}

#endregion // PRIVATE_METHODS
}
```