

LAMPIRAN

Load Scene	Keluar
<pre> using UnityEngine; using System.Collections; using UnityEngine.SceneManagement; public class ButtonManager : MonoBehaviour { public void startbtn(string newGameLevel) { SceneManager.LoadScene(newGameLevel); } } </pre>	<pre> using UnityEngine; using System.Collections; using UnityEngine.UI; public class script : MonoBehaviour { public Canvas start; public Button exit; public void keluar() { Application.Quit(); } } </pre>
Vuforia	
<pre> using UnityEngine; namespace Vuforia { public class VuforiaBehaviour : VuforiaAbstractBehaviour { protected override void Awake() { IUnityPlayer unityPlayer = new NullUnityPlayer(); if (Application.platform == RuntimePlatform.Android) unityPlayer = new AndroidUnityPlayer(); SetUnityPlayerImplementation(unityPlayer); gameObject.AddComponent<ComponentFactoryStarterBehaviour>(); base.Awake(); } private static VuforiaBehaviour mVuforiaBehaviour= null; public static VuforiaBehaviour Instance { get { if (mVuforiaBehaviour == null) mVuforiaBehaviour = FindObjectOfType<VuforiaBehaviour>(); return mVuforiaBehaviour; } } } } </pre>	

DefaultTrackablehandler

```
using UnityEngine;
using UnityEngine.UI;

namespace Vuforia
{
    /// <summary>
    /// A custom handler that implements the ITrackableEventHandler interface.
    /// </summary>
    public class DefaultTrackableEventHandler : MonoBehaviour,
                                             ITrackableEventHandler
    {
        public Transform TextTargetName;
        public Transform ButtonAction;
        public Transform TextDescription;
        public Transform PanelDescription;

        #region PRIVATE_MEMBER_VARIABLES

        private TrackableBehaviour mTrackableBehaviour;

        #endregion // PRIVATE_MEMBER_VARIABLES

        #region UNITY_MONOBEHAVIOUR_METHODS

        void Start()
        {
            mTrackableBehaviour = GetComponent<TrackableBehaviour>();
            if (mTrackableBehaviour)
            {
                mTrackableBehaviour.RegisterTrackableEventHandler(this);
            }
        }

        #endregion // UNITY_MONOBEHAVIOUR_METHODS

        #region PUBLIC_METHODS

        /// <summary>
        /// Implementation of the ITrackableEventHandler function called when the
        /// tracking state changes.
        /// </summary>
        public void OnTrackableStateChanged(
            TrackableBehaviour.Status previousStatus,
            TrackableBehaviour.Status newStatus)
        {
            if (newStatus == TrackableBehaviour.Status.DETECTED ||
                newStatus == TrackableBehaviour.Status.TRACKED ||
                newStatus == TrackableBehaviour.Status.EXTENDED_TRACKED)
            {

```

```

        OnTrackingFound();
    }
    else
    {
        OnTrackingLost();
    }
}

#endregion // PUBLIC_METHODS

#region PRIVATE_METHODS

private void OnTrackingFound()
{
    Renderer[] rendererComponents =
GetComponentInChildren<Renderer>(true);
    Collider[] colliderComponents =
GetComponentInChildren<Collider>(true);

    // Enable rendering:
    foreach (Renderer component in rendererComponents)
    {
        component.enabled = true;
    }

    // Enable colliders:
    foreach (Collider component in colliderComponents)
    {
        component.enabled = true;
    }

    Debug.Log("Trackable " + mTrackableBehaviour.TrackableName + "
found");
}

private void OnTrackingLost()
{
    Renderer[] rendererComponents =
GetComponentInChildren<Renderer>(true);
    Collider[] colliderComponents =
GetComponentInChildren<Collider>(true);

    // Disable rendering:
    foreach (Renderer component in rendererComponents)
    {
        component.enabled = false;
    }

    // Disable colliders:
    foreach (Collider component in colliderComponents)
    {
        component.enabled = false;
    }
}

```

```

    }

    Debug.Log("Trackable " + mTrackableBehaviour.TrackableName + " lost");
    //Evertime the target lost / no target found it will show "???" on the
    TextTargetName. Button, Description and Panel will invicible (inactive)

    TextTargetName.GetComponent<Text>().text = "A R A H K A N
    K E M A R K E R ";
    ButtonAction.gameObject.SetActive(false);
    TextDescription.gameObject.SetActive(false);
    PanelDescription.gameObject.SetActive(false);

    }

    #endregion // PRIVATE_METHODS
}
}

```

Data Target Trackable

```

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

namespace Vuforia
{
    public class dataTarget : MonoBehaviour
    {

        public Transform TextTargetName;
        public Transform TextDescription;
        public Transform ButtonAction;
        public Transform PanelDescription;

        public AudioSource soundTarget;
        public AudioClip clipTarget;

        // Use this for initialization
        void Start()
        {
            //add Audio Source as new game object component
            soundTarget = (AudioSource)gameObject.AddComponent<AudioSource>();
        }

        // Update is called once per frame
        void Update()
        {
            StateManager sm = TrackerManager.Instance.GetStateManager();
            IEnumerable<TrackableBehaviour> tbs =
            sm.GetActiveTrackableBehaviours();

            foreach (TrackableBehaviour tb in tbs)
            {
                string name = tb.TrackableName;
            }
        }
    }
}

```

```

ImageTarget it = tb.Trackable as ImageTarget;
Vector2 size = it.GetSize();

Debug.Log("Active image target:" + name + " -size: " + size.x +
", " + size.y);

//Evertime the target found it will show "name of target" on the
TextTargetName. Button, Description and Panel will visible (active)

TextTargetName.GetComponent<Text>().text = name;
ButtonAction.gameObject.SetActive(true);
TextDescription.gameObject.SetActive(true);
PanelDescription.gameObject.SetActive(true);

//If the target name was "zombie" then add listener to
ButtonAction with location of the zombie sound (locate in Resources/sounds folder)
and set text on TextDescription a description of the zombie

if (name == "BADAQ")
{
ButtonAction.GetComponent<Button>().onClick.AddListener(delegate {
playSound("sounds/Badak_01"); });
TextDescription.GetComponent<Text>().text = "Populasi Badak
yang berada di Sumatera berjumlah antara 60-80 ekor.Banyak sekali perburuan liar
yang terjadi dilakukan oleh pihak yang tidak bertanggung jawab.";
}

if (name == "BERUANG")
{
ButtonAction.GetComponent<Button>().onClick.AddListener(delegate {
playSound("sounds/Beruang_01"); });
TextDescription.GetComponent<Text>().text = "Populasi Beruang
Madu di Balai Konservasi Sumber Daya Alam Aceh (BKSDA) diperkirakan tinggal 50
ekor.Beberapa organ tubuhnya yang begitu diminati dan dihargai cukup tinggi
seperti kuku, taring, otak, hati, dan empedu. Untuk hati dan empedu dihargai
setara dengan emas.";
}

if (name == "HARIMAU")
{
ButtonAction.GetComponent<Button>().onClick.AddListener(delegate {
playSound("sounds/Harimau_01"); });
TextDescription.GetComponent<Text>().text = "Di sumatera
berjumlah 400 ekor dihutan Konservasi, 100 ekor diluar Habitat, termasuk Kebun
Binatang.Perburuan untuk satwa peliharaan, Kulit, Cakar, Kumis, Tulang, Penis, dan
Gigi untuk Souvenir atau dijadikan Obat-obatan.";
}

if (name == "GAJAH")
{
ButtonAction.GetComponent<Button>().onClick.AddListener(delegate {
playSound("sounds/Gajah_01"); });
TextDescription.GetComponent<Text>().text = "Populasi gajah
yang berada di sumatera berjumlah sekitar 2800-4800 dengan 44 populasi.Ancaman

```

```

paling nyata terhadap kelangsungan hidup satwa ini adalah perburuan dan Degradasi
habitat.";
    }
    if (name == "CEMPAKA")
    {
ButtonAction.GetComponent<Button>().onClick.AddListener(delegate {
playSound("sounds/Cempaka_01"); });
        TextDescription.GetComponent<Text>().text = "Bunga Cempaka
Wangi mengeluarkan aroma yang harum. Biasanya, hiasan Bunga Cempaka Wangi
digunakan sebagai riasan untuk menyambut tamu.";
    }
    if (name == "RAFFLESIA")
    {
ButtonAction.GetComponent<Button>().onClick.AddListener(delegate {
playSound("sounds/Rafflesia_01"); });
        TextDescription.GetComponent<Text>().text = "Populasi
Rafflesia yang ada pada tahun 2016 tinggal sekitar 11 bunga, tersebar di beberapa
lokasi di provinsi itu. Ancaman juga datang dari masyarakat yang merusak dan
mengambil putik bunga raflesia untuk dimanfaatkan sebagai obat tradisional.";
    }

    }
}

//function to play sound
void playSound(string ss)
{
    clipTarget = (AudioClip)Resources.Load(ss);
    soundTarget.clip = clipTarget;
    soundTarget.loop = false;
    soundTarget.playOnAwake = false;
    soundTarget.Play();
}

}
}

```