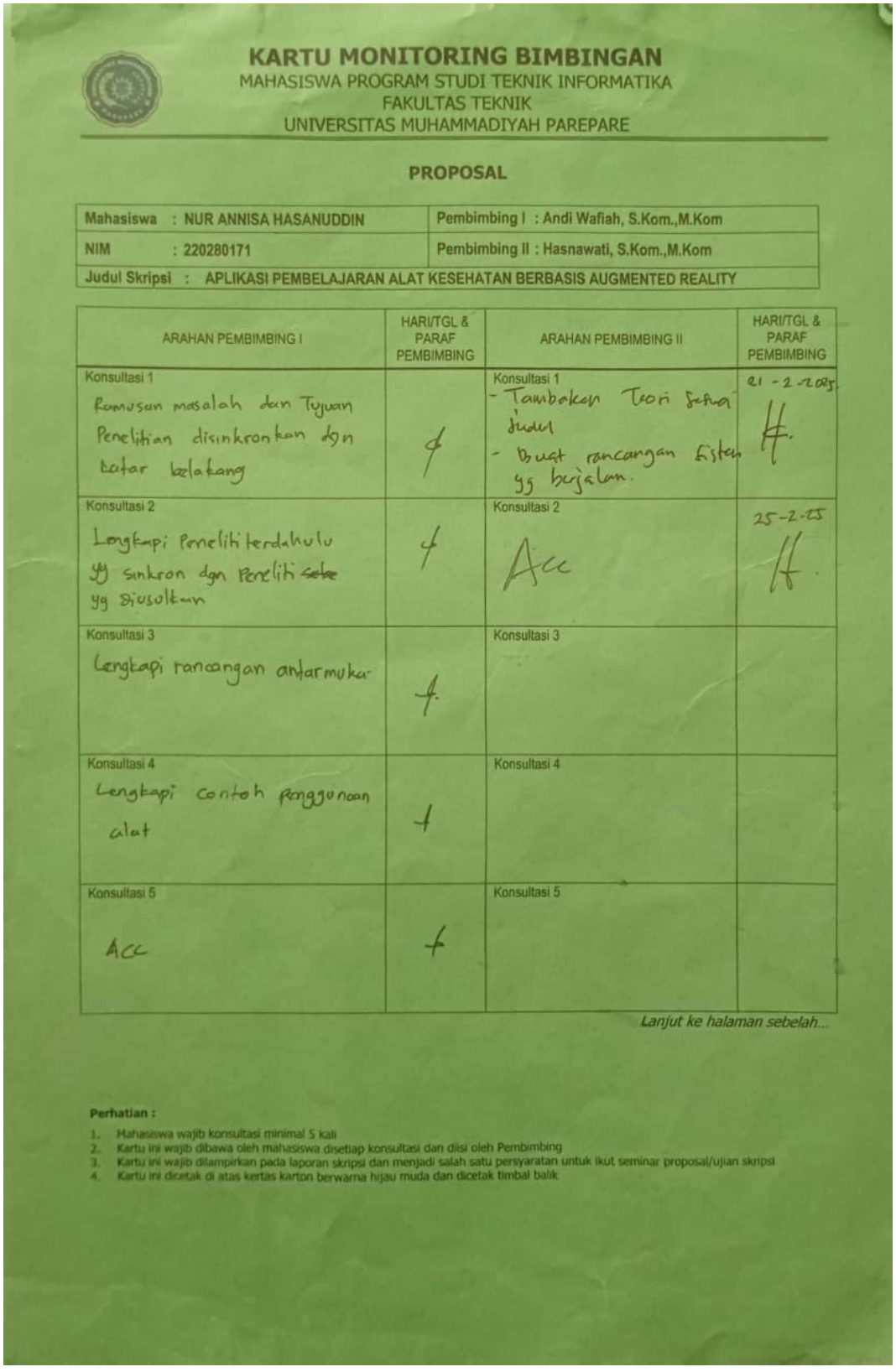
LAMPIRAN

Kartu Monitoring Proposal



KARTU MONITORING BIMBINGAN
MAHASISWA PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS MUHAMMADIYAH PAREPARE

PROPOSAL

Mahasiswa : NUR ANNISA HASANUDDIN	Pembimbing I : Andi Wafiah, S.Kom.,M.Kom
NIM : 220280171	Pembimbing II : Hasnawati, S.Kom.,M.Kom
Judul Skripsi : APLIKASI PEMBELAJARAN ALAT KESEHATAN BERBASIS AUGMENTED REALITY	

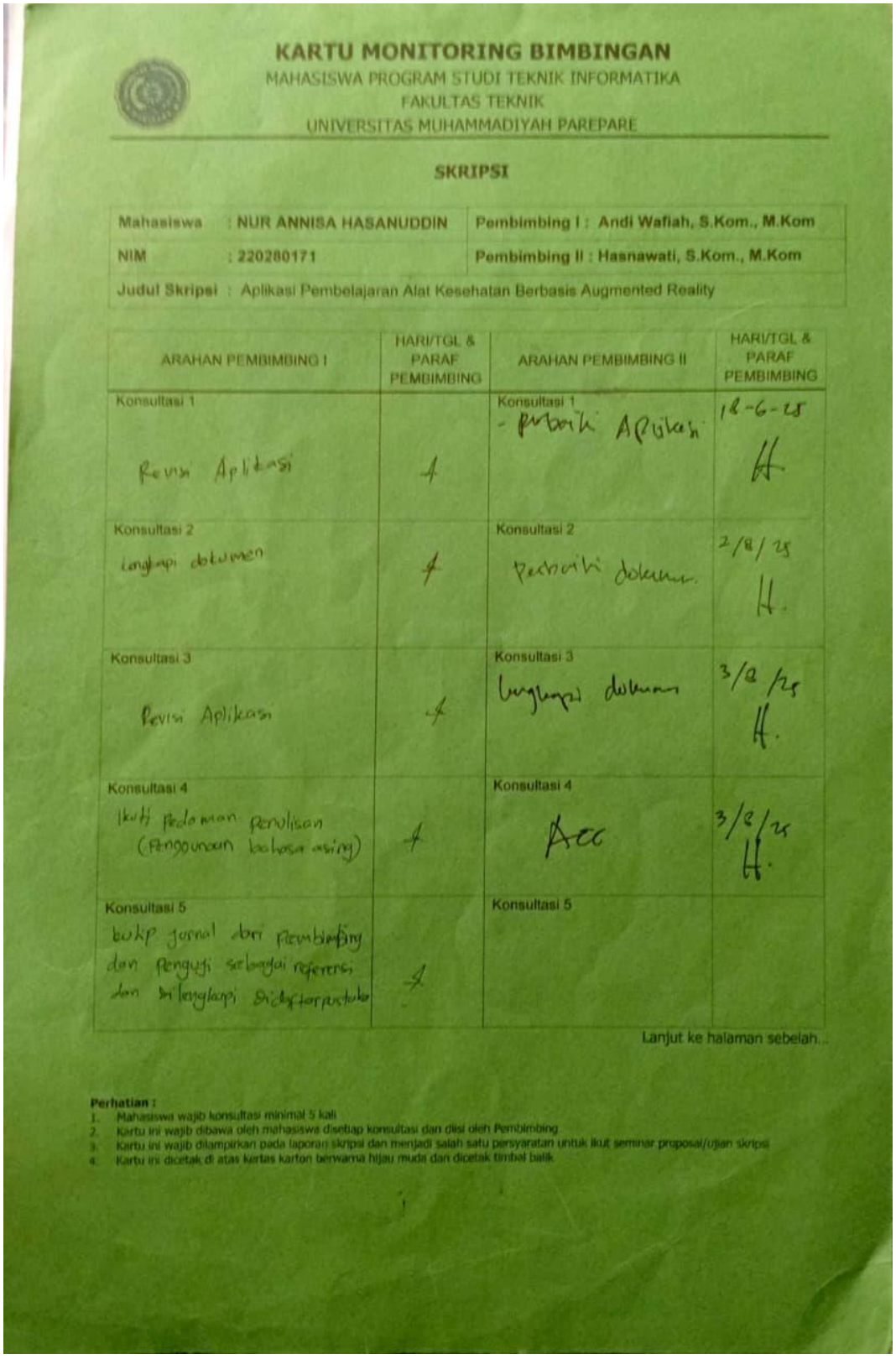
ARAHAN PEMBIMBING I	HARI/TGL & PARAF PEMBIMBING	ARAHAN PEMBIMBING II	HARI/TGL & PARAF PEMBIMBING
Konsultasi 1 Rumusan masalah dan Tujuan Penelitian disinkronkan dgn latar belakang		Konsultasi 1 - Tambahkan Teori Sesuai judul - Buat rancangan Sistem yg berjalan.	21-2-2025
Konsultasi 2 Lengkapi Penelititerdahulu yg sinkron dgn Penelitisetse yg disusulkan		Konsultasi 2 Acc	25-2-25
Konsultasi 3 Lengkapi rancangan antarmuka		Konsultasi 3	
Konsultasi 4 Lengkapi contoh penggunaan alat		Konsultasi 4	
Konsultasi 5 Acc		Konsultasi 5	

Lanjut ke halaman sebelah...

Perhatian :

1. Mahasiswa wajib konsultasi minimal 5 kali
2. Kartu ini wajib dibawa oleh mahasiswa disetiap konsultasi dan diisi oleh Pembimbing
3. Kartu ini wajib dilampirkan pada laporan skripsi dan menjadi salah satu persyaratan untuk ikut seminar proposal/ujian skripsi
4. Kartu ini dicetak di atas kertas karton berwarna hijau muda dan dicetak timbal balik

Kartu Monitoring Skripsi



KARTU MONITORING BIMBINGAN

MAHASISWA PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS MUHAMMADIYAH PAREPARE

SKRIPSI

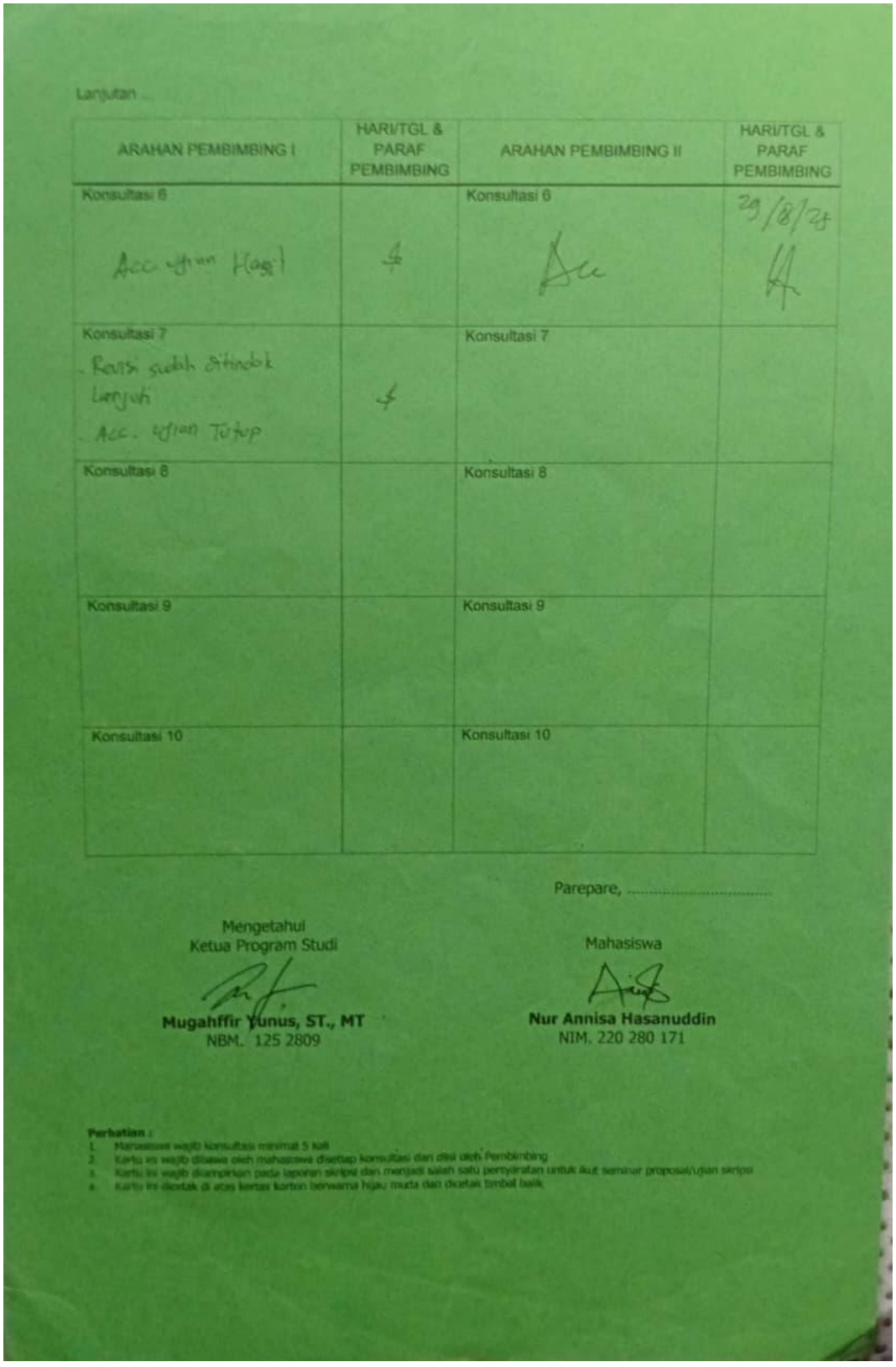
Mahasiswa	: NUR ANNISA HASANUDDIN	Pembimbing I :	Andi Wafiah, S.Kom., M.Kom
NIM	: 220280171	Pembimbing II :	Hasnawati, S.Kom., M.Kom
Judul Skripsi	: Aplikasi Pembelajaran Alat Kesehatan Berbasis Augmented Reality		

ARAHAN PEMBIMBING I	HARI/TGL & PARAF PEMBIMBING	ARAHAN PEMBIMBING II	HARI/TGL & PARAF PEMBIMBING
Konsultasi 1 Revisi Aplikasi		Konsultasi 1 - perbaiki Aplikasi	18-6-25
Konsultasi 2 lengkapi dokumen		Konsultasi 2 perbaiki dokumen	2/8/25
Konsultasi 3 Revisi Aplikasi		Konsultasi 3 lengkapi dokumen	3/8/25
Konsultasi 4 ikuti pedoman penulisan (penggunaan bahasa asing)		Konsultasi 4 Acc	3/8/25
Konsultasi 5 bukp jurnal dari Pembimbing dan penguji sebagai referensi dan di lengkapi Sidang terakhir		Konsultasi 5	

Lanjut ke halaman sebelah..

Perhatian :
1. Mahasiswa wajib konsultasi minimal 5 kali
2. Kartu ini wajib dibawa oleh mahasiswa disetiap konsultasi dan diisi oleh Pembimbing
3. Kartu ini wajib dilampirkan pada laporan skripsi dan menjadi salah satu persyaratan untuk ikut seminar proposal/ujian skripsi
4. Kartu ini dicetak di atas kertas karton berwarna hijau muda dan dicetak timbel balik

Kartu Monitoring Skripsi



Lanjutan ...

ARAHAN PEMBIMBING I	HARI/TGL & PARAF PEMBIMBING	ARAHAN PEMBIMBING II	HARI/TGL & PARAF PEMBIMBING
Konsultasi 6 Acc ujian Hasil	4	Konsultasi 6 Acc	29/8/24
Konsultasi 7 - Revisi sudah ditindak lanjuti - Acc. ujian Tutup	4	Konsultasi 7	
Konsultasi 8		Konsultasi 8	
Konsultasi 9		Konsultasi 9	
Konsultasi 10		Konsultasi 10	

Parepare, ...

Mengetahui
Ketua Program Studi

Mugahffir Yunus, ST., MT
NBM. 125 2809

Mahasiswa

Nur Annisa Hasanuddin
NIM. 220 280 171

Perhatian :
1. Mahasiswa wajib konsultasi minimal 5 kali
2. Kartu ini wajib diisi oleh mahasiswa disetiap konsultasi dan diisi oleh Pembimbing
3. Kartu ini wajib diampirkan pada laporan skripsi dan menjadi salah satu persyaratan untuk ikut seminar proposal/ujian skripsi
4. Kartu ini dicetak di atas kertas karton berwarna hijau muda dan dicetak timbal balik

Lampiran -2 Script Program

- **Script Back**

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

public class Back : MonoBehaviour
{
    public void BackDeskripsi()
    {
        SceneManager.LoadScene("MenuDeskripsi");
    }

    public void BackAr()
    {
        SceneManager.LoadScene("MenuAr");
    }

    public void BackMain()
    {
        SceneManager.LoadScene("MainMenu");
    }

    public void BackMenu()
    {
        SceneManager.LoadScene("MenuUtama");
    }

    public void BackQuiz()
    {
        SceneManager.LoadScene("MenuQuiz");
    }
}
```

- **Script Jawab**

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

public class jawab : MonoBehaviour
{
    public GameObject feed_benar, feed_salah;

    void Start()
    {

    }

    public GameObject panelHasil;

    public void jawaban(bool jawab)
    {
        if (jawab)
        {
            feed_benar.SetActive(false);
```

```
        feed_benar.SetActive(true);

        int skor = PlayerPrefs.GetInt("skor")
+ 20;
        PlayerPrefs.SetInt("skor", skor);
    }
    else
    {
        feed_salah.SetActive(false);
        feed_salah.SetActive(true);
    }
```

- **Script Lihat Daftar Skor**

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

[System.Serializable]
public class ScoreEntry
{
  public string nama;
  public int skor;
}

[System.Serializable]
public class ScoreList
{
  public List<ScoreEntry> skorList = new
List<ScoreEntry>();
}
```

```
    gameObject.SetActive(false);
    transform.parent.GetChild
(gameObject.transform.GetSiblingIndex () +
1).gameObject.SetActive (true);
  }

  void Update()
  {

  }
}
```

```
public class LihatDaftarSkor :
MonoBehaviour
{
  public GameObject panelHasil;
  public GameObject panelDaftarSkor;
  public TextMeshProUGUI
teksDaftarSkor;

  public void TampilkanDaftarSkor()
  {
    panelHasil.SetActive(false);
    panelDaftarSkor.SetActive(true);

    string json =
PlayerPrefs.GetString("daftarSkor", "{}");
    ScoreList scoreList =
JsonUtility.FromJson<ScoreList>(json);

    var skorTerurut = scoreList.skorList
      .OrderByDescending(s => s.skor)
```

```
            .ToList();


        string tampilan = "<b>High Score</b>\n\n";
        int urutan = 1;


        foreach (var entry in skorTerurut)
        {
            // Tambahkan nomor urut dan padding agar rata kiri
            tampilan += $"{urutan,2}. {entry.nama,-15} {entry.skor,5}\n";
            urutan++;
        }


        teksDaftarSkor.text = tampilan;
        Debug.Log("JSON dari PlayerPrefs: " + json);
    }
}
```

- **Script Links**

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


public class Link : MonoBehaviour
{


    public void Ambubag()
    {
        Application.OpenURL("https://youtu.be/S31mww2efcY?si=q-hwDlxWOE08YOIs");
    }
    public void Bak()
    {
        Application.OpenURL("https://youtu.be/cXwLYa-CUws?si=5VpFGbhov4kFbFsN");
    }


    public void Bola()
    {
        Application.OpenURL("https://youtu.be/Ajrp3z7jHXE?si=YyAAzjFix7y6UAk8");
    }


    public void Cervical()
    {
        Application.OpenURL("https://youtu.be/FLWK35-Mwfw?si=tAceeZiMViLz6OUN");
    }


    public void Abocath()
    {
        Application.OpenURL("https://youtu.be/ZwTzRpf-fhc?si=AG2POlZNzoppLxAO");
    }


    public void Ett()
    {
        Application.OpenURL("https://youtu.be/zrXbtmMhEH8?si=ENSdrFB4DaS2wYDL");
```

```
    }

    public void Inkubator()
    {
        Application.OpenURL("https://youtube
.com/shorts/tV7-e-
Wp2H8?si=kxBKhA9o0YpRrS2_");
    }

    public void Defibrillator()
    {
        Application.OpenURL("https://youtu.b
e/pp7m3Hhenk0?si=GWUxAnAvU_7sF09s
");
    }

    public void Laringiscop()
    {
        Application.OpenURL("https://youtube
.com/shorts/k5AZRDlKDEg?si=Z02e9EN0
QMpePQwV");
    }

    public void Spine()
    {
        Application.OpenURL("https://youtu.b
e/_VTgy7CbT4I?si=ubsY1DO48Kp2BtxQ")
;
    }

    public void Mayo()
    {
        Application.OpenURL("https://youtube
.com/shorts/jHVTBJmKXVQ?si=6lXz7Y4Z
APnYKIFs");
    }

    public void Oksigen()
    {
        Application.OpenURL("https://youtu.b
e/VIOWviu1xW0?si=NHtCg4uScpwmKmN
R");
    }

    public void Refleks()
    {
        Application.OpenURL("https://youtube
.com/shorts/7DyqcjQws6s?si=hsxx4vOc93w
556Hg");
    }

    public void Stetoskop()
    {
        Application.OpenURL("https://youtube
.com/shorts/MqCD0djsJp4?si=wlJuycaSJiT0
xUKt");
    }

    public void Suction()
    {
        Application.OpenURL("https://youtu.b
e/lVXCWS2Psjo?si=1lKp3XRcUhIeMJTH"
);
    }

    public void Tensi()
    {
        Application.OpenURL("https://youtu.b
e/cq6zS61fONg?si=NIcZO5pmnXeltd2q");
    }
```

```
    public void Termometer()
    {
        Application.OpenURL("https://youtu.be/9Kfoy5VbJWk?si=T_Vi5_DUESLX2m_7");
    }

    public void Saturasi()
    {
        Application.OpenURL("https://youtu.be/0aVXhd4PlRk?si=zNH4YWOCMAC65Mq2");
    }

    public void Troly()
    {
        Application.OpenURL("https://youtube.com/shorts/1taNWEzypj4?si=12J2itu6uKWql-Ru");
    }

    public void Walker()
    {
        Application.OpenURL("https://youtu.be/BEmaklIN5_I?si=CRfrGMGQ8mTJ65Aa");
    }

}
```

- **Script MainMenu**

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

public class MainMenu : MonoBehaviour
{

    public void Exit() {
        Application.Quit();
        Debug.Log("Aplikasi Telah Keluar");
    }

    public void About()
    {
        SceneManager.LoadScene("About");
    }

    public void MenuUtama()
    {
        SceneManager.LoadScene("MenuUtama");
    }

}
```

- **Script Ar**

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

```
using UnityEngine;
using UnityEngine.SceneManagement;

public class MenuAr : MonoBehaviour
{
    public void ArAmbubag()
    {
        SceneManager.LoadScene("ArAmbubag");
    }

    public void Back()
    {
        SceneManager.LoadScene("MenuUtama");
    }
}
```

- **Script Deskripsi**

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

public class MenuDeskripsi : MonoBehaviour
{
    public void DesAmbubag()
    {
        SceneManager.LoadScene("DesAmbubag");
    }

    public void DesBak()
    {
        SceneManager.LoadScene("DesBak");
    }

    public void DesBola()
    {
        SceneManager.LoadScene("DesBola");
    }

    public void DesCervical()
    {
        SceneManager.LoadScene("DesCervical");
    }

    public void DesAbocath()
    {
        SceneManager.LoadScene("DesAbocath");
    }

    public void DesEtt()
    {
        SceneManager.LoadScene("DesEtt");
    }

    public void DesInkubator()
    {
```

```
    SceneManager.LoadScene("DesInkubator");
    }

    public void DesDefibrillator()
    {
    SceneManager.LoadScene("DesDefibrillator");
    }

    public void DesLaringiscop()
    {
    SceneManager.LoadScene("DesLaringiscop");
    }

    public void DesMayo()
    {
    SceneManager.LoadScene("DesMayo");
    }

    public void DesOksigen()
    {
    SceneManager.LoadScene("DesOksigen");
    }

    public void DesRefleks()
    {
    SceneManager.LoadScene("DesRefleks");
    }

    public void DesSpine()
    {
    SceneManager.LoadScene("DesSpine");
    }

    public void DesStetoskop()
    {
    SceneManager.LoadScene("DesStetoskop");
    }

    public void DesSuction()
    {
    SceneManager.LoadScene("DesSuction");
    }

    public void DesTensi()
    {
    SceneManager.LoadScene("DesTensi");
    }

    public void DesTermometer()
    {
    SceneManager.LoadScene("DesTermometer");
    }

    public void DesSaturasi()
    {
    SceneManager.LoadScene("DesSaturasi");
```

```
    }

    public void DesTroly()
    {
        SceneManager.LoadScene("DesTroly")
;
    }

    public void DesWalker()
    {
        SceneManager.LoadScene("DesWalker
");
    }

    public void Back()
    {
        SceneManager.LoadScene("MenuUtam
a");
    }

}
```

- **Script MenuQuiz**

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

public class MenuQuiz : MonoBehaviour
{
    public void Play()
    {
        SceneManager.LoadScene("Quiz");
    }

    public void Back()
    {
        SceneManager.LoadScene("MenuUtam
a");
    }
}
```

- **Script MenuUtama**

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

public class MenuUtama : MonoBehaviour
{
    public void MenuAr()
    {
        SceneManager.LoadScene("ArMenu");
    }

    public void MenuDeskripsi()
    {
        SceneManager.LoadScene("MenuDesk
ripsi");
    }
```

```
    public void MenuQuiz()
    {
        SceneManager.LoadScene("MenuQuiz"
);
    }
```

- **Script NewBehaviour**

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

public class NewBehaviourScript :
MonoBehaviour
{
    public float rotationSpeed = 100f;
    public float zoomSpeed = 0.5f;
    public float minZoom = 2f;
    public float maxZoom = 15f;

    private Vector2 startTouchPosition,
endTouchPosition;
    private Camera mainCamera;

    void Start()
    {
        mainCamera = Camera.main;
    }

    void Update()
    {
        // ======= ROTASI DENGAN
KEYBOARD =======
        if (Input.GetKey(KeyCode.LeftArrow))
            transform.Rotate(0, -rotationSpeed *
Time.deltaTime, 0);
        else if
(Input.GetKey(KeyCode.RightArrow))
            transform.Rotate(0, rotationSpeed *
Time.deltaTime, 0);
        else if
(Input.GetKey(KeyCode.UpArrow))
            transform.Rotate(-rotationSpeed *
Time.deltaTime, 0, 0);
        else if
(Input.GetKey(KeyCode.DownArrow))
            transform.Rotate(rotationSpeed *
Time.deltaTime, 0, 0);

        // ======= ZOOM DENGAN
SCROLL MOUSE (Editor) =======
        float scroll = Input.GetAxis("Mouse
ScrollWheel");
        if (scroll != 0f)
        {
            float distance =
Vector3.Distance(transform.position,
mainCamera.transform.position);
            float zoomAmount = scroll *
zoomSpeed * 10f;

            if ((scroll > 0 && distance >
minZoom) || (scroll < 0 && distance <
maxZoom))
```

```
    public void Back()
    {
        SceneManager.LoadScene("MainMenu
");
    }
}
```

```
            mainCamera.transform.Translate(0
, 0, zoomAmount);
        }

        // ======= ROTASI DENGAN
SWIPE (HP) =======
        if (Input.touchCount == 1)
        {
            Touch touch = Input.GetTouch(0);

            if (touch.phase ==
TouchPhase.Began)
                startTouchPosition =
touch.position;

            if (touch.phase ==
TouchPhase.Ended)
            {
                endTouchPosition =
touch.position;
                Vector2 swipeDelta =
endTouchPosition - startTouchPosition;

                if (swipeDelta.magnitude > 50f)
                {
                    if (Mathf.Abs(swipeDelta.x) >
Mathf.Abs(swipeDelta.y))
                    {
                        if (swipeDelta.x > 0)
                            transform.Rotate(0,
rotationSpeed * Time.deltaTime * 10f, 0);
                        else
                            transform.Rotate(0, -
rotationSpeed * Time.deltaTime * 10f, 0);
                    }
                    else
                    {
                        if (swipeDelta.y > 0)
                            transform.Rotate(-
rotationSpeed * Time.deltaTime * 10f, 0, 0);
                        else
                            transform.Rotate(rotationS
peed * Time.deltaTime * 10f, 0, 0);
                    }
                }
            }
        }

        // ======= ZOOM DENGAN PINCH
(HP) =======
        if (Input.touchCount == 2)
        {
            Touch touchZero =
Input.GetTouch(0);
            Touch touchOne =
Input.GetTouch(1);

            Vector2 prevTouchZero =
touchZero.position -
touchZero.deltaPosition;
            Vector2 prevTouchOne =
touchOne.position - touchOne.deltaPosition;

            float prevMagnitude =
(prevTouchZero -
prevTouchOne).magnitude;
            float currentMagnitude =
(touchZero.position -
touchOne.position).magnitude;

            float difference = currentMagnitude -
prevMagnitude;
```

```
        float distance =
Vector3.Distance(transform.position,
mainCamera.transform.position);

        float zoomAmount = difference *
zoomSpeed * Time.deltaTime;

        if ((difference > 0 && distance >
minZoom) || (difference < 0 && distance <
maxZoom))

            mainCamera.transform.Translate(0
, 0, zoomAmount);

      }

   }

}
```

- **Script PlayVideoOnTrack**

```
using UnityEngine;

using Vuforia;


public class PlayVideoOnTrack :
MonoBehaviour

{

   private ObserverBehaviour
observerBehaviour;

   private UnityEngine.Video.VideoPlayer
videoPlayer;



   void Start()

   {

      // Ambil komponen ObserverBehaviour
dari ImageTarget

      observerBehaviour =
GetComponent<ObserverBehaviour>();

      videoPlayer =
GetComponentInChildren<UnityEngine.Vid
eo.VideoPlayer>();



      if (observerBehaviour != null)

      {

         // Daftarkan event saat status target
berubah

         observerBehaviour.OnTargetStatusC
hanged += OnTargetStatusChanged;

      }

   }


   private void
OnTargetStatusChanged(ObserverBehaviour
behaviour, TargetStatus status)

   {

      if (status.Status == Status.TRACKED ||
status.Status ==
Status.EXTENDED_TRACKED)

      {

         // 🎬 Marker terdeteksi → mainkan
video

         if (videoPlayer != null &&
!videoPlayer.isPlaying)

            videoPlayer.Play();

      }

      else

      {

         // ⏸ Marker hilang → pause video

         if (videoPlayer != null &&
videoPlayer.isPlaying)

            videoPlayer.Pause();
```

```
        // kalau mau restart dari awal, ganti Pause() jadi Stop()
    }
```

- **Script Rotasi3D**

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

public class Rotasi3D : MonoBehaviour
{
    public bool Terputar = false;
    public Vector3 KecepatanPutaran;


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

        if (Terputar)
        {
            transform.Rotate(
                KecepatanPutaran.x * Time.deltaTime * 10,
                KecepatanPutaran.y * Time.deltaTime * 10,
                KecepatanPutaran.z * Time.deltaTime * 10
            );
        }

    }


    private void OnMouseDown()
    {
        if(!Terputar)
        {
            Terputar = true;
        }
        else
        {
            Terputar = false;
        }
    }
}
```

- **Script Skor**

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using TMPro; // Wajib untuk TextMeshPro

public class skor : MonoBehaviour
{
    void Start()
    {
        PlayerPrefs.SetInt ("skor", 0);
```

```
    }

    void Update()
    {
```

- **Script SkorVisabilityManager**

```
using UnityEngine;

public class SkorVisibilityManager : MonoBehaviour
{
    public GameObject skorLabel;  // Drag objek Label/skor ke sini di Inspector

    public GameObject[] panelSoal; // Semua panel soal (Soal1, Soal2, dst)

    void Update()
    {
        bool isSoalAktif = false;

        foreach (var panel in panelSoal)
        {
            if (panel.activeInHierarchy)
            {
                isSoalAktif = true;
                break;
            }
        }

        skorLabel.SetActive(isSoalAktif);
    }
}
```

- **Script Tampilkan Hasil**

```
using UnityEngine;
using TMPro;

public class TampilkanHasil : MonoBehaviour
{
    public TextMeshProUGUI hasilText;

    void OnEnable()
    {
        string username = PlayerPrefs.GetString("username", "Player");
        int skor = PlayerPrefs.GetInt("skor", 0);

        hasilText.text = "Nama : " + username + "\nSkor Anda : " + skor;

        // Ambil data sebelumnya
        string json = PlayerPrefs.GetString("daftarSkor", "{}");
        GetComponent<TextMeshProUGUI>().text = PlayerPrefs.GetInt("skor").ToString();
    }
}
```

```
        ScoreList daftar =
JsonUtility.FromJson<ScoreList>(json);
        if (daftar == null || daftar.skorList ==
null)
            daftar = new ScoreList();

        // Tambah data baru
        daftar.skorList.Add(new ScoreEntry {
nama = username, skor = skor });

        // Simpan kembali
        string jsonBaru =
JsonUtility.ToJson(daftar);
        PlayerPrefs.SetString("daftarSkor",
jsonBaru);
        PlayerPrefs.Save();

        Debug.Log("Data skor disimpan: " +
jsonBaru);
    }
}
```

- **Script UsernameManager**

```
using UnityEngine;
using UnityEngine.UI;
using TMPro;
public class UsernameManager :
MonoBehaviour
{
    public TMP_InputField inputUsername;
    public GameObject panelInput;
    public GameObject
panelPertanyaanPertama;

    public void MulaiKuis()
    {
        string username =
inputUsername.text.Trim();
        if (!string.IsNullOrEmpty(username))
        {
            PlayerPrefs.SetString("username",
username);
            PlayerPrefs.SetInt("skor", 0);
            panelInput.SetActive(false);
            panelPertanyaanPertama.SetActive(tr
ue);
        }
        else
        {
            Debug.Log("Silakan isi username
terlebih dahulu.");
        }
    }
```