

CALISTUNG: MEDIA PEMBELAJARAN INTERAKTIF MEMBACA, MENULIS DAN BERHITUNG BERBASIS *TOUCH SCREEN* DAN *AUGMENTED REALITY*

mrizalumami27@gmail.com

<https://github.com/RizalUmami/CALISTUNG.git>

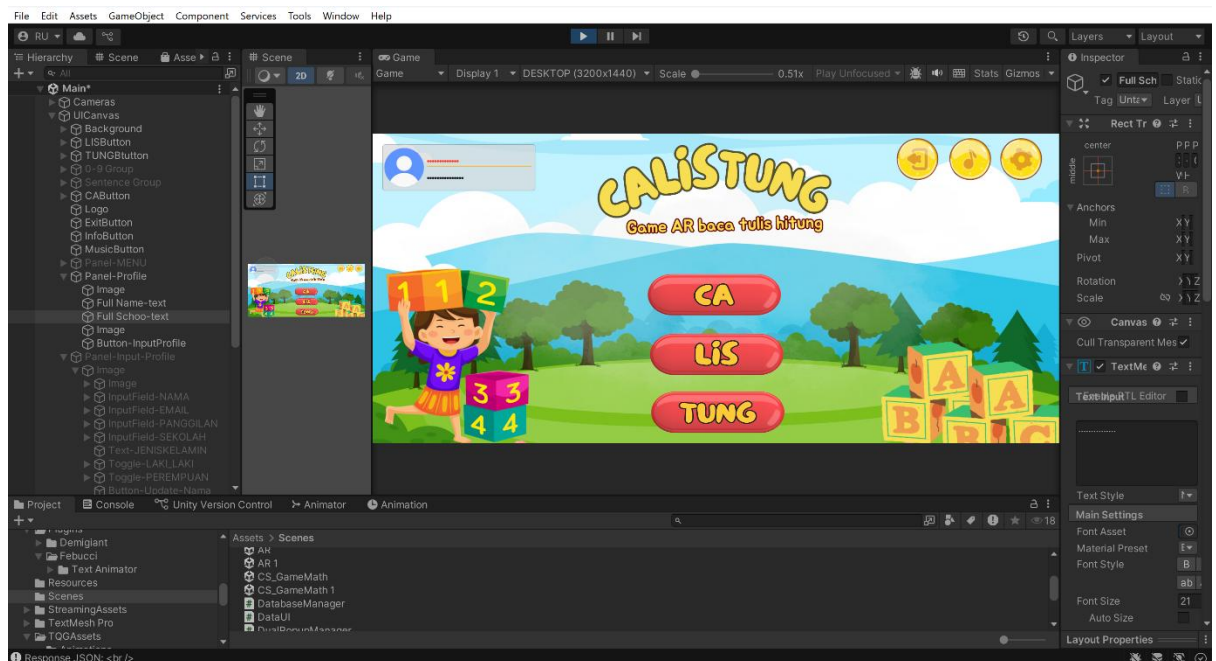
Game edukatif calistung adalah sebuah permainan edukatif berbasis *Augmented Reality* (AR) yang dirancang khusus untuk anak-anak usia 4- 7 tahun. Game ini bertujuan untuk membantu anak-anak belajar membaca, menulis, dan berhitung (calistung) dengan cara yang interaktif dan menyenangkan.

Menggabungkan teknologi AR dengan metode belajar yang sederhana, anak-anak dapat belajar melalui pengalaman visual dan interaktif yang menarik. Dalam permainan ini, anak-anak menggunakan perangkat (seperti tablet atau smartphone), dan akan menampilkan konten 3D, animasi, dan suara yang memperkaya pembelajaran.

Dengan berbagai tugas interaktif seperti menyusun kata, menebalkan huruf, dan menjawab soal berhitung, anak-anak diajak untuk belajar sambil bermain.

EVENT DRIVEN

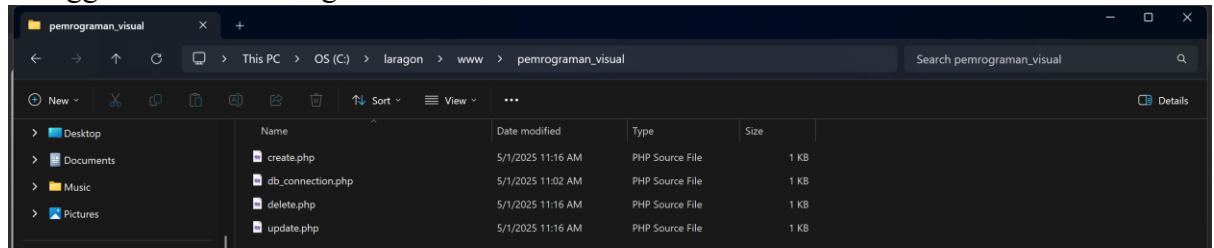
A. Menu



Di Scene Menu terdapat 7 Button yang dapat berfungsi

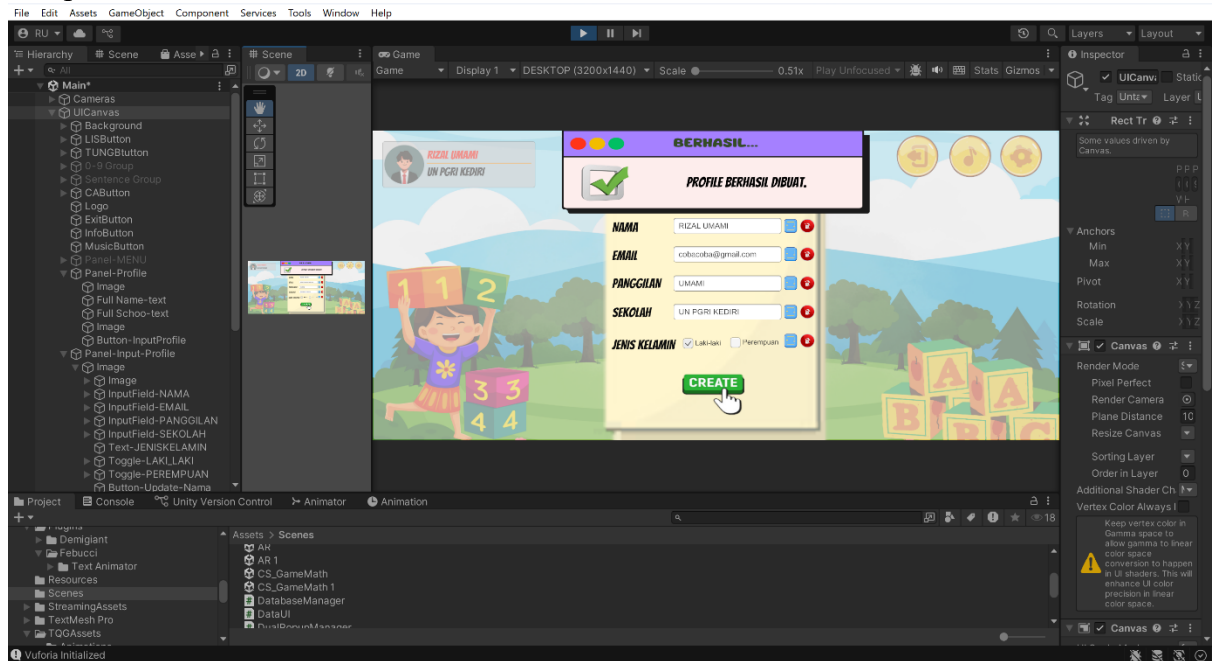
1. ButtonCA > Masuk ke SceneBACA
2. ButtonLI > Masuk ke SceneTULIS
3. ButtonTUNG > Masuk ke SceneTUNG
4. ButtonEXIT > Fungsi PanelExit()
5. ButtonMUSIC > Switch On/ Off Ketika di klik, Fungsi AudioManager()
6. ButtonINFO > Masuk ke SceneINFO
7. ButtonPROFILE > Masuk ke PanelPROFILE
Dibagian PROFILE (CRUD/ koneksi localhost)

1. Menggunakan PHP sebagai koneksi ke database



Terdapat create.php, db_connection.php, delete.php, dan update.php

2. Tampilan UI Profile



Pengguna/siswa dapat menambahkan data baik dari nama, email, namapanggilan, asalsekkolah, dan jenis kelamin.

Ketika berhasil dan di isi semua kolom akan menampilkan popup berhasil yang di animasikan dari atas kebawah, dan otomatis tersimpan ke database db_calistung, jika pengguna/siswa klik toggle Laki-laki maka akan tampil di profile menu akan berubah menjadi Sprite Laki-Laki begitu juga sebaliknya, yang semulanya default sprite.

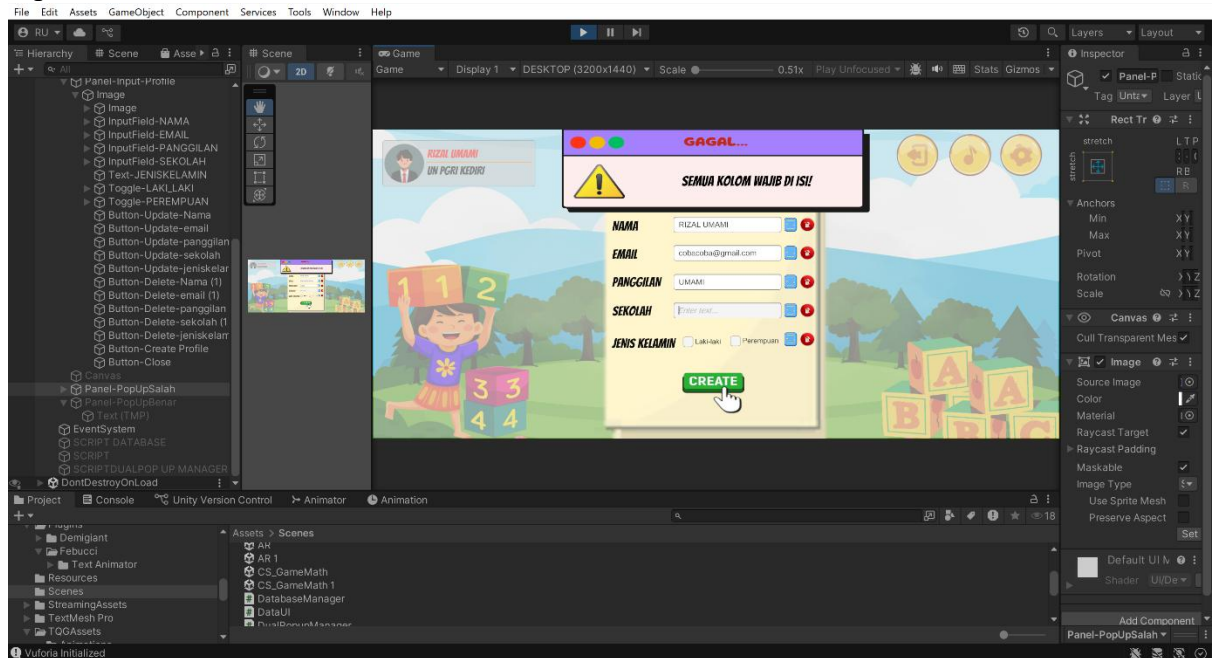
Ketika Sudah melakukan penambahan data pengguna/siswa dapat melakukan perubahan data di masing-masing kolom dan juga melakukan penghapusan data/ setiap kolomnya serta menampilkan popup berhasil/ tidaknya.

Dibagian Profile terdapat event-event

1. InputTextFieldNAMA | ButtonUpdateNAMA | ButtonDeleteNAMA
2. InputTextFieldEMAIL | ButtonUpdateEMAIL | ButtonDeleteEMAIL
3. InputTextFieldPANGGILAN | ButtonUpdatePANGGILAN | ButtonDeletePANGGILAN
4. InputTextFieldSEKOLAH | ButtonUpdateSEKOLAH | ButtonDeleteSEKOLAH

5. InputTextFieldGENDER | ButtonUpdateGENDER | ButtonDeleteGENDER
6. ButtonCreatePROFILE
7. Fungsi ClosePanelPROFILE()
8. Fungsi PopupMessageTrue()
9. Fungsi PopupMessageFalse()

Jika gagal melakukan apapun maka akan tampil Fungsi PopupMessageFalse() seperti contoh dibawah



3. Program Utama CRUD

A. DataBaseManager.cs

```

1. using UnityEngine;
2. using UnityEngine.Networking;
3. using System.Collections;
4.
5. [System.Serializable]
6. public class ServerResponse
7. {
8.     public string status, message;
9. }
10.
11. [System.Serializable]
12. public class ProfilData
13. {
14.     public string nama, gender;
15.     public bool lengkap;
16. }
17.
18. [System.Serializable]
19. public class CekDataDiriResponse
20. {
21.     public string status;
22.     public ProfilData data;
23. }
24.
25. public class DataBaseManager : MonoBehaviour
26. {

```

```

27.     public DualPopupManager popupManager;
28.     private const string BASE = "http://localhost/pemrograman_visual/";
29.
30.     public void Create_Siswa(string nama, string email, string panggilan, string asal,
string gender)
31.     {
32.         var json = JsonUtility.ToJson(new {
33.             nama, email, namapanggilan = panggilan, asalsekolah = asal, gender
34.         });
35.         StartCoroutine(Post("create.php", json));
36.     }
37.
38.     public void Update_Siswa(string kolom, string nilai, int id)
39.     {
40.         var json = JsonUtility.ToJson(new { id, kolom, nilai });
41.         StartCoroutine(Post("update.php", json));
42.     }
43.
44.     public void Delete_Siswa(string kolom, int id)
45.     {
46.         var json = JsonUtility.ToJson(new { id, kolom });
47.         StartCoroutine(Post("delete.php", json));
48.     }
49.
50.     public void CekDataDiri_Siswa(string email, System.Action<ProfilData> cb)
51.     {
52.         var json = JsonUtility.ToJson(new { email });
53.         StartCoroutine(PostForProfil("cek_data_diri.php", json, cb));
54.     }
55.
56.     IEnumerator Post(string ep, string json)
57.     {
58.         var req = new UnityWebRequest(BASE + ep, "POST");
59.         byte[] body = System.Text.Encoding.UTF8.GetBytes(json);
60.         req.uploadHandler = new UploadHandlerRaw(body);
61.         req.downloadHandler = new DownloadHandlerBuffer();
62.         req.SetRequestHeader("Content-Type", "application/json");
63.         yield return req.SendWebRequest();
64.
65.         if (req.result == UnityWebRequest.Result.Success)
66.         {
67.             var res = JsonUtility.FromJson<ServerResponse>(req.downloadHandler.text);
68.             if (res.status == "success") popupManager.ShowSuccess(res.message);
69.             else popupManager.ShowError(res.message);
70.         }
71.         else
72.         {
73.             popupManager.ShowError("Gagal koneksi server");
74.         }
75.     }
76.
77.     IEnumerator PostForProfil(string ep, string json, System.Action<ProfilData> cb)
78.     {
79.         var req = new UnityWebRequest(BASE + ep, "POST");
80.         byte[] body = System.Text.Encoding.UTF8.GetBytes(json);
81.         req.uploadHandler = new UploadHandlerRaw(body);
82.         req.downloadHandler = new DownloadHandlerBuffer();
83.         req.SetRequestHeader("Content-Type", "application/json");
84.
85.         yield return req.SendWebRequest();
86.
87.         if (req.result == UnityWebRequest.Result.Success)
88.         {
89.             // Log the raw response
90.             Debug.Log("Response JSON: " + req.downloadHandler.text);
91.

```

```

92.         try
93.         {
94.             var res =
JsonUtility.FromJson<CekDataDiriResponse>(req.downloadHandler.text);
95.
96.             if (res.status == "exists" && res.data != null)
97.             {
98.                 cb(res.data);
99.             }
100.            else
101.            {
102.                popupManager.ShowError("Data tidak ditemukan");
103.            }
104.        }
105.        catch (System.Exception e)
106.        {
107.            popupManager.ShowError("Error parsing JSON: " + e.Message);
108.        }
109.    }
110.    else
111.    {
112.        popupManager.ShowError("Gagal koneksi server");
113.    }
114. }
115.
116. IEnumerator Get(string ep, System.Action<ProfilData> cb)
117. {
118.     var req = UnityWebRequest.Get(BASE + ep);
119.     yield return req.SendWebRequest();
120.
121.     if (req.result == UnityWebRequest.Result.Success)
122.         cb(JsonUtility.FromJson<ProfilData>(req.downloadHandler.text));
123.     else
124.         popupManager.ShowError("Gagal koneksi server");
125. }
126. }
127.

```

B. DataUI.cs

```

1.  using UnityEngine;
2.  using UnityEngine.UI;
3.  using TMPPro;
4.
5.  public class DataUI : MonoBehaviour
6.  {
7.      [Header("Profil Inputs")]
8.      public TMP_InputField namaInput;
9.      public TMP_InputField emailInput;
10.     public TMP_InputField panggilanInput;
11.     public TMP_InputField asalInput;
12.
13.     [Header("Gender Selection")]
14.     public Toggle maleToggle;
15.     public Toggle femaleToggle;
16.
17.     [Header("Profile Image")]
18.     public Image profileImage;
19.     public Sprite defaultImage;
20.     public Sprite maleImage;
21.     public Sprite femaleImage;
22.
23.     [Header("Manager")]
24.     public DatabaseManager db;
25.

```

```

26.     [Header("ID Input (for Update/Delete)")]
27.     public TMP_InputField idInput;
28.
29.     public void OnCreateClicked()
30.     {
31.         string gender = maleToggle.isOn ? "laki-laki" : femaleToggle.isOn ? "perempuan" :
        "";
32.         if (string.IsNullOrEmpty(namaInput.text) ||
33.             string.IsNullOrEmpty(emailInput.text) ||
34.             string.IsNullOrEmpty(panggilanInput.text) ||
35.             string.IsNullOrEmpty(asalInput.text) ||
36.             string.IsNullOrEmpty(gender))
37.         {
38.             db.popupManager.ShowError("Semua kolom wajib diisi!");
39.             return;
40.         }
41.
42.         profileImage.sprite = gender == "laki-laki" ? maleImage : femaleImage;
43.
44.         db.Create_Siswa(
45.             namaInput.text,
46.             emailInput.text,
47.             panggilanInput.text,
48.             asalInput.text,
49.             gender
50.         );
51.         db.popupManager.ShowSuccess("Profile berhasil dibuat.");
52.     }
53.
54.     public void OnUpdateField(string kolom, string nilai)
55.     {
56.         if (string.IsNullOrEmpty(idInput.text))
57.         {
58.             db.popupManager.ShowError("Masukkan ID terlebih dahulu!");
59.             return;
60.         }
61.
62.         if (int.TryParse(idInput.text, out int id))
63.             db.Update_Siswa(kolom, nilai, id);
64.         else
65.             db.popupManager.ShowError("ID tidak valid!");
66.     }
67.
68.     public void OnDeleteField(string kolom)
69.     {
70.         if (string.IsNullOrEmpty(idInput.text))
71.         {
72.             db.popupManager.ShowError("Masukkan ID terlebih dahulu!");
73.             return;
74.         }
75.
76.         if (int.TryParse(idInput.text, out int id))
77.             db.Delete_Siswa(kolom, id);
78.         else
79.             db.popupManager.ShowError("ID tidak valid!");
80.     }
81.
82.     public void OnUpdateNama() {
83.         db.popupManager.ShowSuccess("Mengupdate nama...");
84.         OnUpdateField("nama", namaInput.text);
85.     }
86.     public void OnDropNama() {
87.         db.popupManager.ShowSuccess("Menghapus nama...");
88.         OnDeleteField("nama");
89.     }
90.     public void OnUpdateEmail() {

```

```

91.         db.popupManager.ShowSuccess("Mengupdate email...");
92.         OnUpdateField("email", emailInput.text);
93.     }
94.     public void OnDropEmail() {
95.         db.popupManager.ShowSuccess("Menghapus email...");
96.         OnDeleteField("email");
97.     }
98.     public void OnUpdatePanggilan() {
99.         db.popupManager.ShowSuccess("Mengupdate panggilan...");
100.        OnUpdateField("namapanggilan", panggilanInput.text);
101.    }
102.    public void OnDropPanggilan() {
103.        db.popupManager.ShowSuccess("Menghapus panggilan...");
104.        OnDeleteField("namapanggilan");
105.    }
106.    public void OnUpdateAsal() {
107.        db.popupManager.ShowSuccess("Mengupdate asal sekolah...");
108.        OnUpdateField("asalsekolah", asalInput.text);
109.    }
110.    public void OnDropAsal() {
111.        db.popupManager.ShowSuccess("Menghapus asal sekolah...");
112.        OnDeleteField("asalsekolah");
113.    }
114.    public void OnUpdateGender() {
115.        string gender = maleToggle.isOn ? "laki-laki" : femaleToggle.isOn ? "perempuan" :
116.        "";
117.        if (string.IsNullOrEmpty(gender)) {
118.            db.popupManager.ShowError("Pilih gender terlebih dahulu!");
119.            return;
120.        }
121.        db.popupManager.ShowSuccess("Mengupdate Jenis Kelamin...");
122.        OnUpdateField("gender", gender);
123.    }
124.    public void OnDropGender() {
125.        db.popupManager.ShowSuccess("Menghapus Jenis Kelamin...");
126.        OnDeleteField("gender");
127.    }
128.

```

C. DualPopupmanager.cs

```

1.  using UnityEngine;
2.  using UnityEngine.UI;
3.  using DG.Tweening;
4.  using TMPro;
5.
6.  public class DualPopupManager : MonoBehaviour
7.  {
8.      [Header("Success Popup")]
9.      public GameObject successPanel;
10.     public RectTransform successRect;
11.     public TMP_Text successText;
12.
13.     [Header("Error Popup")]
14.     public GameObject errorPanel;
15.     public RectTransform errorRect;
16.     public TMP_Text errorText;
17.
18.     Vector2 hidden = new Vector2(0, 500);
19.     Vector2 visible = new Vector2(0, -30);
20.
21.     void Awake()
22.     {
23.         successPanel.SetActive(false);
24.         errorPanel.SetActive(false);

```

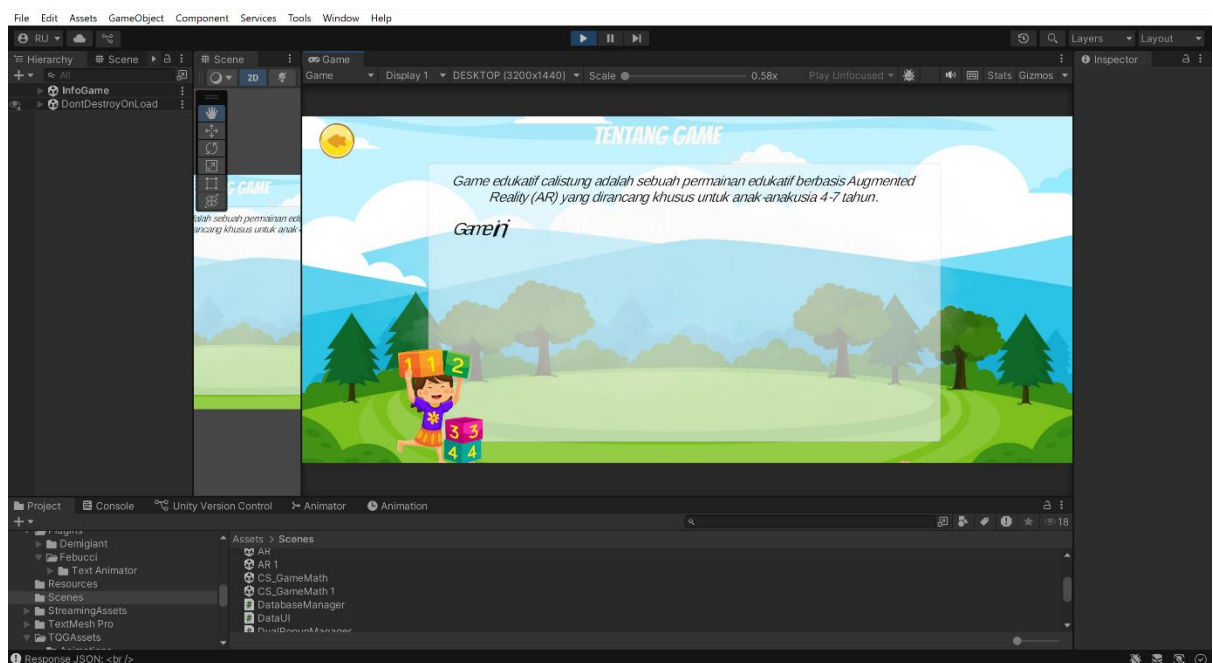


```

25.         successRect.anchoredPosition = hidden;
26.         errorRect.anchoredPosition = hidden;
27.     }
28.
29.     public void ShowSuccess(string msg)
30.     {
31.         successText.text = msg;
32.         Animate(successRect, successPanel);
33.     }
34.
35.     public void ShowError(string msg)
36.     {
37.         errorText.text = msg;
38.         Animate(errorRect, errorPanel);
39.     }
40.
41.     void Animate(RectTransform rt, GameObject panel)
42.     {
43.         panel.SetActive(true);
44.         rt.DOAnchorPos(visible, 0.5f).OnComplete(() => {
45.             DOVirtual.DelayedCall(2f, () => {
46.                 rt.DOAnchorPos(hidden, 0.5f).OnComplete(() => {
47.                     panel.SetActive(false);
48.                 });
49.             });
50.         });
51.     }
52. }
53.

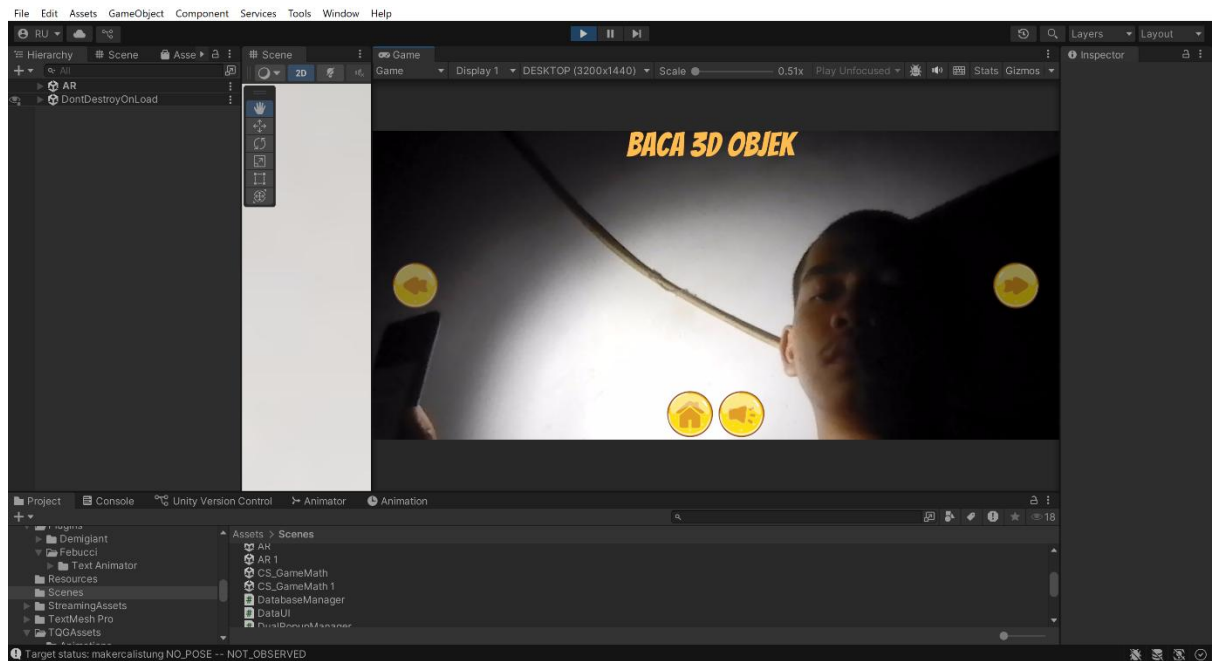
```

B. Info

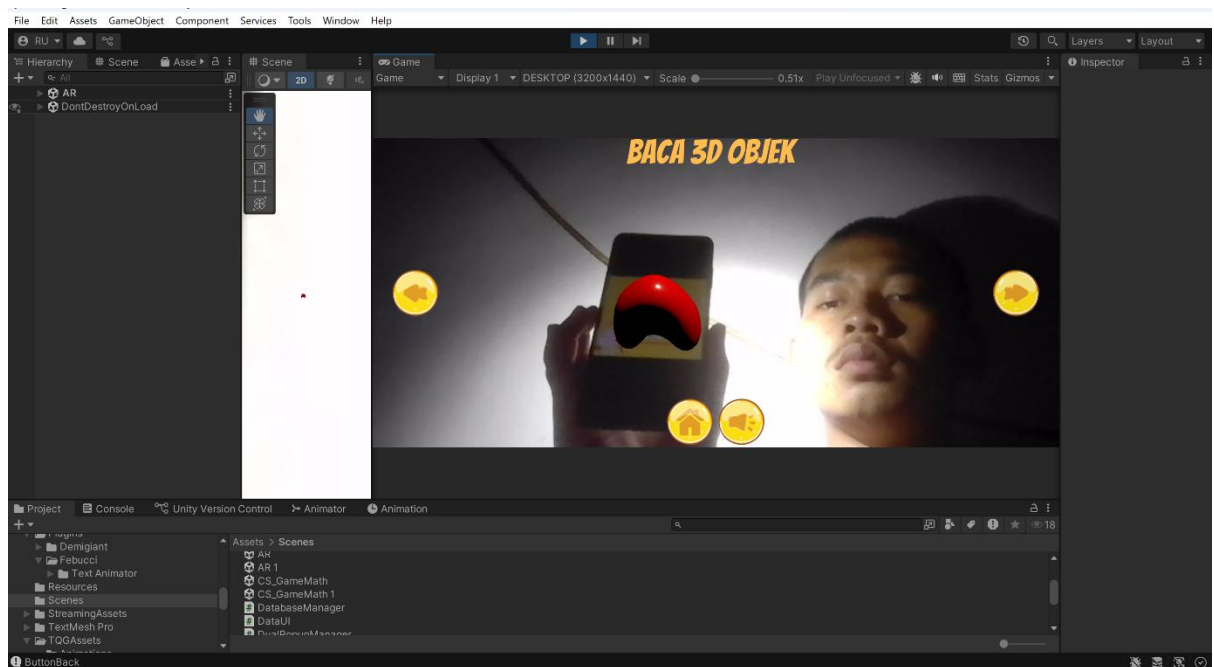


Dibagian SceneINFO hanya menampilkan text berjalan yang ditampilkan melalui animasi DOTween, serta terdapat ButtonBACK

C. SceneAR



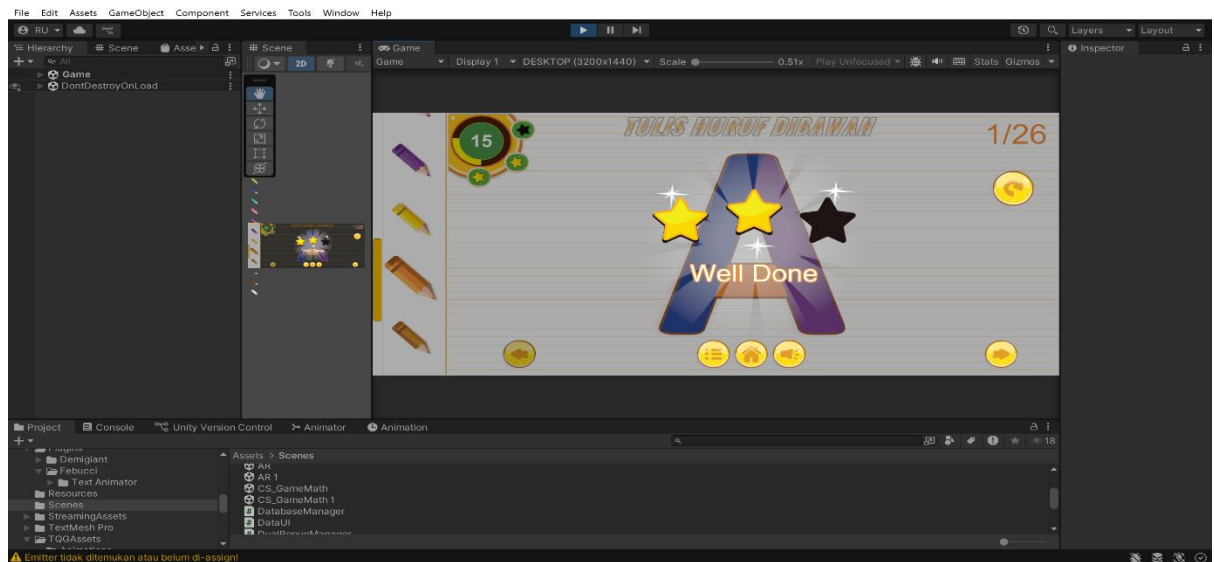
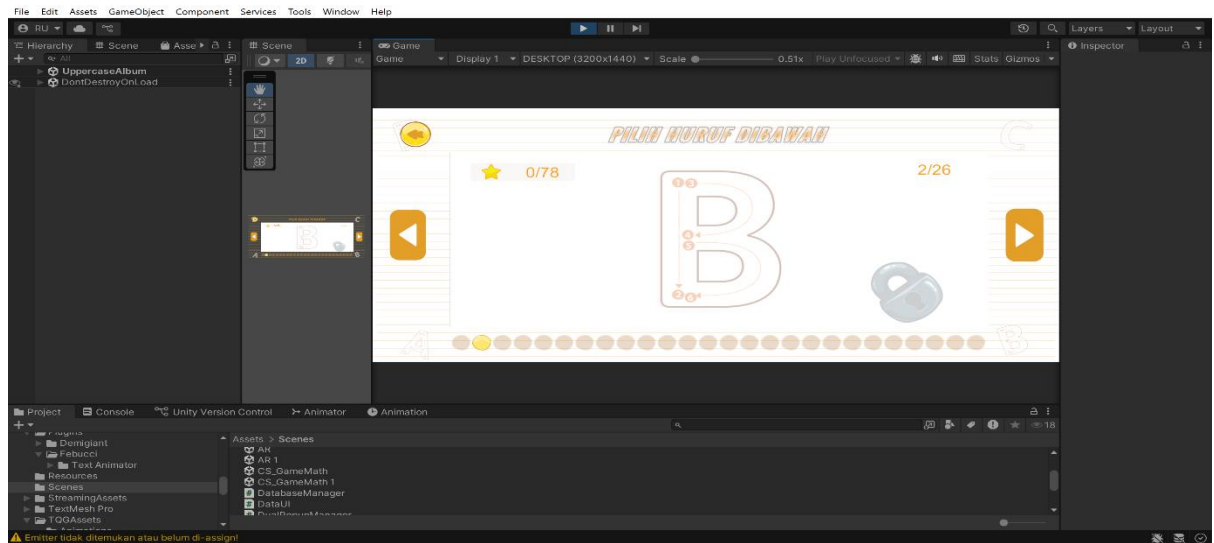
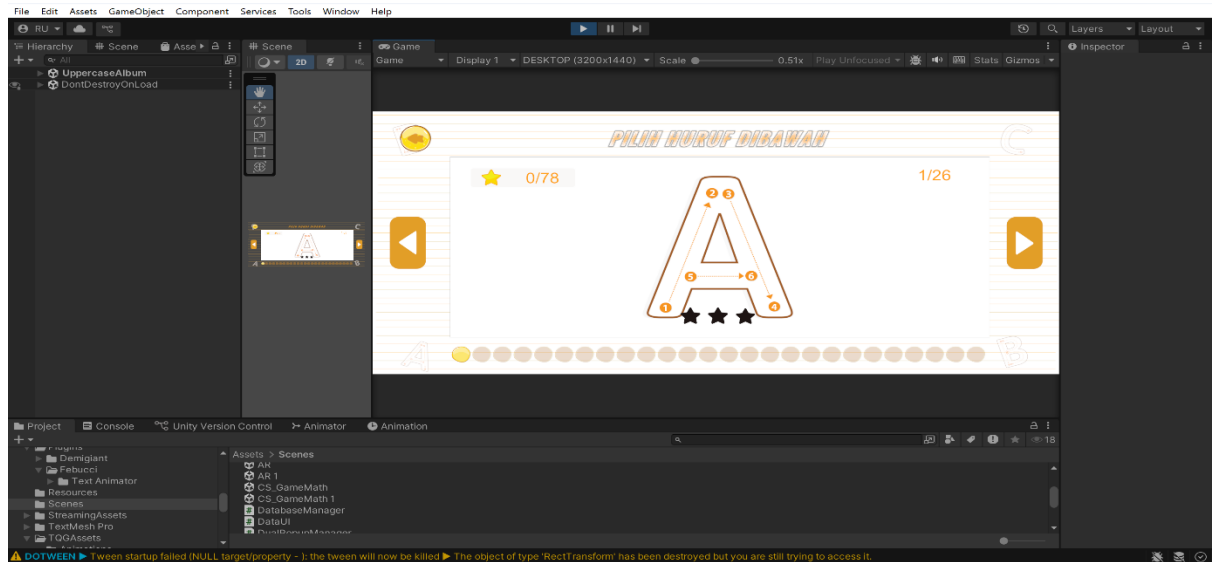
Di Scene AR terdapat interaksi jika maker belum terdeteksi maka Button-button tidak akan aktif, jika sudah terdeteksi maka button bisa digunakan, serta harus ada maker terlebih dahulu yang sebelumnya harus ada di vuforia



Terdapat 4 Button di SceneAR

1. ButtonHOME > Kembali ke SceneMENU
2. ButtonMusicObject > Menampilkan suara
3. ButtonNEXT dan BACK, jika maker sudah terdeteksi maka button NEXT dan BACK bisa digunakan dari 3d Huruf A-Z

D. SceneTULIS

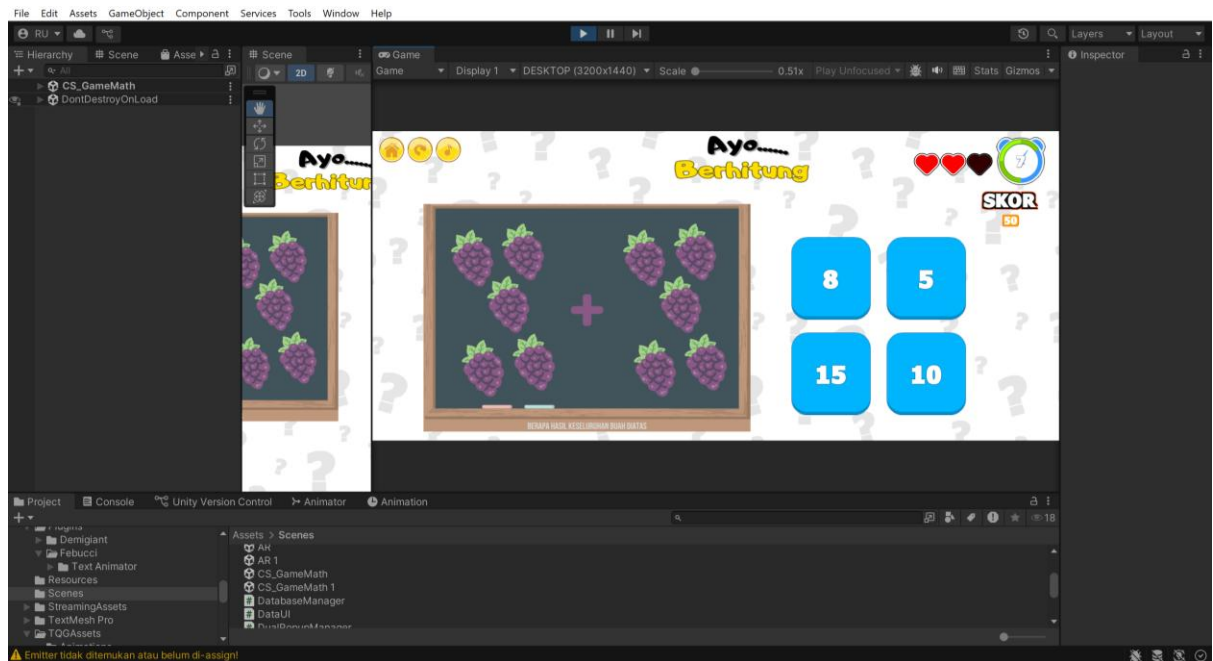


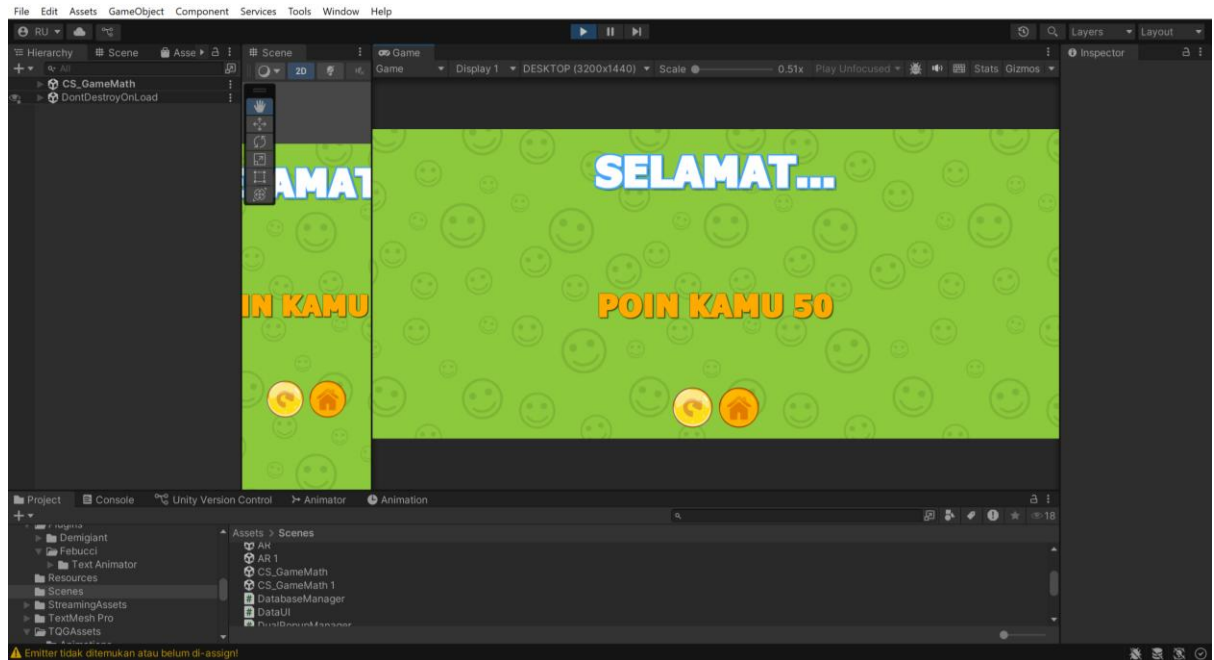
Di scene tulis terdapt PanelUppercase yang berisikan huruf-huruf yang sudah dan belum di gunakan, jika belum di gunakan/ latihan maka huruf berikutnya terkunci, untuk membuka kunci maka harus latihan terlebih dahulu yang dimana dimulai dari huruf A-Z

Event-event di SceneTULIS

1. ScrollbarA-Z
2. ButtonBACK
3. ButtonBACK2
4. ButtonNEXT2
5. MazkingView
6. ScrollbarPEN
7. ButtonBACK3
8. ButtonNEXT3
9. ButtonRESET
10. ButtonHOME
11. ButtonMUSIC
12. ButtonINFO

E. SceneHITUNG





DiScene HITUNG terdapat soal-soal 1-5 soal yang dapat dikerjakan oleh pengguna/ siswa terdapat waktu dan nyawa apabila diperlukan.

Event-Event diScene HITUNG

1. ButtonHOME
2. ButtonRESET
3. ButtonMUSIC
4. ButtonScore
5. ButtonRESET2
6. ButtonHOME2