

BAB IV PEMBAHASAN DAN HASIL PENELITIAN

4.1 Alur Program

Gambar 4.1 menjelaskan alur program atau apa saja yang terjadi ketika game berjalan. Game terdiri dari tiga level masing masing level memiliki enam sampai tujuh soal, diawali dengan player yang mendapatkan soal kemudian player menjawab soal tersebut dan terjadilah proses perhitungan poin serta registrasi soal mana yang sering dijawab salah oleh player. Hingga pada akhir game total skor player akan menentukan bagaimana hasil yang didapatkan player, serta perpustakaan mendapatkan data yang berguna untuk literasi perpustakaan.

4.2 Perancangan Game

Game literasi VR ini dibuat dengan tujuan agar mahasiswa mendapat suasana yang baru ketika mempelajari tentang kegiatan perpustakaan. Sesuai dengan visinya perpustakaan menggunakan game berbasis virtual reality untuk mengikuti perkembangan zaman serta selera generasi milenial untuk bermain game sehingga kegiatan untuk meliterasikan tentang perpustakaan tidak monoton dan menarik minat mahasiswa untuk mengetahui lebih mengenai perpustakaan.

Game ini hanya dapat berjalan menggunakan alat bernama Oculus Go, pemilihan alat Oculus Go didasari dengan kesadaran pihak programmer selama mengerjakan project VR ini. Awalnya project ini direncanakan berjalan menggunakan VR Box biasa dengan media android namun Ketika program sudah jadi dan di test device android yang digunakan sangat panas dan dicemaskan menimbulkan radiasi Ketika pemain menggunakan terlalu lama, sehingga timbul ide untuk pengembangan Game ini ke dalam Oculus Go.

Selain sebagai sarana perpustakaan untuk meliterasikan kegiatan perpustakaan, Game ini juga akan memberikan data jumlah soal soal yang sering dijawab salah oleh player yang telah memainkan game ini. Hal ini dilakukan supaya perpustakaan dapat membuat materi yang tepat Ketika mengadakan acara literasi lainnya.

4.3 Desain Game

Sebelum game dibuat rancangan game didiskusikan bersama programmer perpustakaan dan kepala perpustakaan. Berdasarkan hasil diskusi muncul ide untuk membuat game kuis pilihan ganda berbasis Virtual Reality menggunakan android sesuai dengan visi misi perpustakaan yaitu mengikuti perkembangan teknologi yang ada. Selama pembuatan game terdapat usulan untuk membuat jawaban yang sebelumnya pilihan ganda menjadi pilihan benar atau salah sehingga permainan menjadi lebih sederhana dan mudah ditangkap oleh para pemain.

Setelah beberapa kali percobaan game terdapat masalah pada perangkat android yang digunakan lama kelamaan menjadi panas dan tidak nyaman digunakan. Dilakukan diskusi bersama dengan kepala perpustakaan untuk memecahkan masalah perangkat karena game ini akan standby selama perpustakaan buka sehingga tidak mungkin perangkat android bisa mendukung keperluan perpustakaan. Dilakukan pencarian perangkat bersama dengan programmer perpustakaan, dari hasil pencarian ditemukan sebuah alat Virtual Reality mandiri dengan Operating Sistem sendiri yaitu Oculus Go yang diklaim dapat mengatasi masalah overheating pada perangkat Virtual Reality sehingga diusulkan penggunaan Oculus Go kepada perpustakaan supaya game berjalan lebih maksimal dan kenyamanan player lebih terjamin.

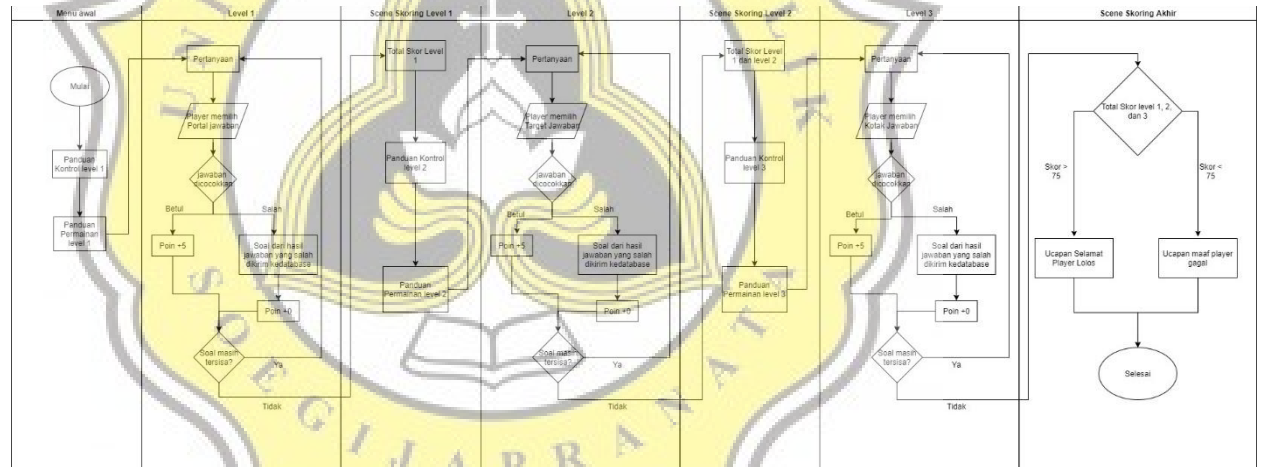
Selama pemindahan game dari android ke Oculus Go Kepala perpustakaan menambahkan fitur penghitungan jawaban salah yang bisa digunakan perpustakaan untuk mendapatkan data materi apa yang kurang dimengerti mahasiswa mengenai perpustakaan itu sendiri sehingga dapat mendukung pembuatan materi Ketika perpustakaan mengadakan acara literasi dalam bentuk lain. Fitur tersebut akhirnya ditambahkan dalam game dengan server yang disediakan oleh perpustakaan sehingga Game ini bisa bermanfaat untuk kedua belah pihak baik dari pihak pemain sebagai sarana untuk mendapatkan ilmu kepustakaan dan pihak perpustakaan untuk mendapatkan data dari pemain. Dosen pembimbing menyarankan penambahan level sebagai salah satu cara untuk menambah model permainan sehingga tidak permainan menjadi tidak membosankan.

Nantinya pustakawan akan berperan sebagai pemandu permainan serta pembuat soal sehingga pustakawan bisa meliterasikan informasi yang ingin sampaikan melalui game tersebut. Selain sebagai alat literasi game ini juga dapat digunakan sebagai alat untuk menguji seberapa paham materi yang diberikan pustakawan terhadap mahasiswa Ketika melakukan roadshow sehingga pustakawan bisa mengevaluasi apa saja materi yang kurang dipahami oleh mahasiswa roadshow melalui permainan Virtual Reality ini.

4.4 Gameplay

Permainan ini memiliki genre permainan kuis, seperti permainan kuis pada umumnya disini pemain diharuskan menjawab semua pertanyaan yang berhubungan dengan informasi seputar kegiatan Pustaka pada perpustakaan UNIKA Soegijapranata. Game terdiri dari 3 level dan dua puluh soal dengan pembagian sebagai berikut, pada level satu terdapat enam soal pemain harus menjawab pertanyaan yang disediakan

dengan cara berjalan masuk kedalam portal benar atau salah , pada level dua terdiri dari tujuh soal pemain menjawab pertanyaan yang disediakan dengan cara membidik dan menembak target benar atau salah, dan pada level tiga terdiri dari tujuh soal pemain menjawab pertanyaan dengan cara berjalan kemudian menyerang kotak target benar atau salah. Masing masing soal dengan jawaban benar akan mendapatkan poin 5 namun apabila jawaban salah tidak akan mendapatkan poin. Setiap jawaban salah terdapat penjelasan sehingga pemain bisa mengetahui informasi yang tepat mengenai materi yang diberikan. Permainan tidak menyediakan fitur untuk melewati pernyataan sehingga pemain harus menjawab semua pertanyaan yang ada. Batas skor untuk dinyatakan lolos permainan adalah sama dengan atau di atas 75.



Gambar 4.1 Alur permainan

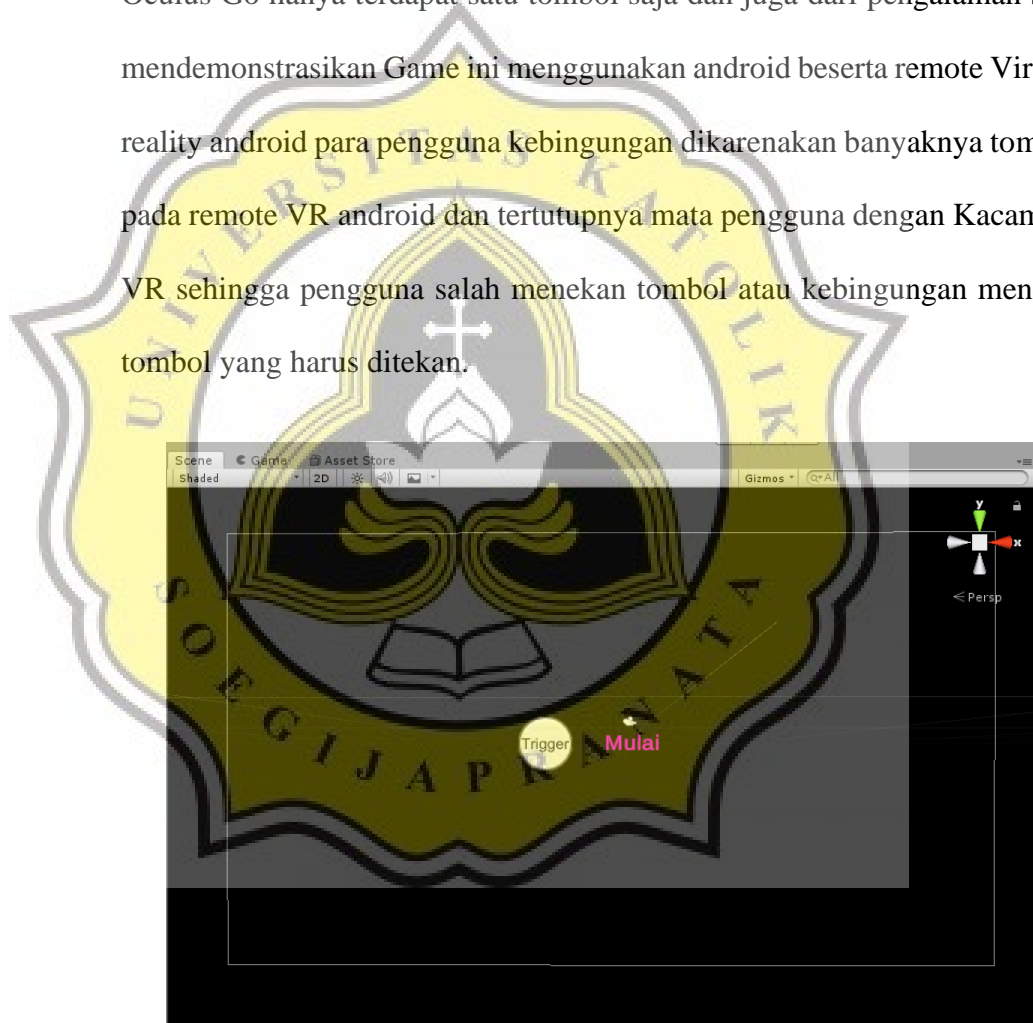
4.5 Pembuatan Game

Game literasi Virtual Reality ini dibuat menggunakan game engine Unity 3D, berbasis Virtual reality menggunakan Oculus Go. Game ini memiliki 2 fungsi yaitu sebagai sarana mahasiswa untuk mengetahui tentang kegiatan perpustakaan dan sarana

perpustakaan untuk mendapatkan data sebagai bahan pertimbangan pembuatan materi literasi pada kegiatan lainnya.

4.5.1 Menu Utama

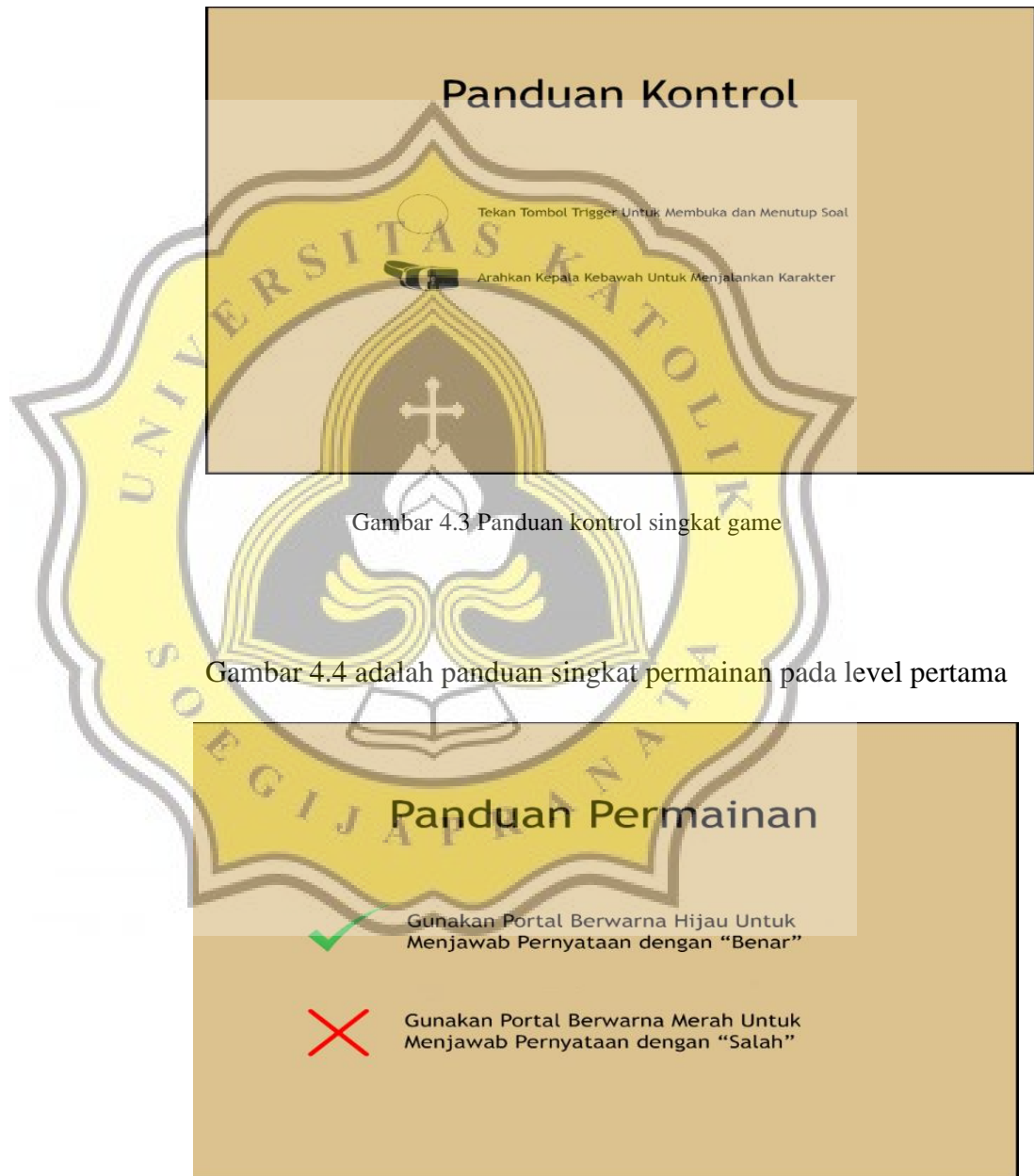
Tampilan utama bisa dilihat pada gambar 4.2. Tampilannya dibuat sangat sederhana dengan satu tombol dikarenakan remote kontrol pada Oculus Go hanya terdapat satu tombol saja dan juga dari pengalaman saat mendemonstrasikan Game ini menggunakan android beserta remote Virtual reality android para pengguna kebingungan dikarenakan banyaknya tombol pada remote VR android dan tertutupnya mata pengguna dengan Kacamata VR sehingga pengguna salah menekan tombol atau kebingungan mencari tombol yang harus ditekan.



Gambar 4.2 Tampilan menu utama game

4.5.2 Permainan Level 1

Ketika player menekan tombol mulai maka akan muncul dua panduan sederhana yaitu bagaimana cara menggerakkan player yang ada pada level pertama seperti pada Gambar 4.3 dan panduan singkat bermain game seperti pada Gambar 4.4.



Gambar 4.3 Panduan kontrol singkat game

Gambar 4.4 adalah panduan singkat permainan pada level pertama

Gambar 4.4 Panduan permainan

Kemudian pada gambar 4.5 menjelaskan script yang digunakan untuk melakukan perpindahan dari scene menu kemudian menampilkan panduan serta perpindahan ke scene permainan.

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

public class MainMenu : MonoBehaviour
{
    public GameObject tutor;
    public GameObject tutor1;
    int State = 0;
    void Update()
    {
        //untuk membuka tutorial
        if (Input.GetButtonDown("Fire1"))
        {
            Debug.Log(State);
            if (State == 0)
            {
                State = 1;
                tutor.SetActive(true);
            }
            //untuk memulai game
            else if (State == 1)
            {
                State = 2;
                tutor.SetActive(false);
                tutor1.SetActive(true);
            }
            else if (State == 2)
            {
                tutor1.SetActive(false);
                SceneManager.LoadScene(1);
                Skor.scoreValue = 0;
            }
        }
    }
}
```

Gambar 4.5 Script main menu

A. Movement Player

Untuk Movement player pada level pertama ini dibuat lebih interaktif dengan cara player berjalan dengan menundukkan kepalanya. Ketika player menundukkan kepalanya karakter dalam game akan berjalan maju. Script yang digunakan terdapat pada Gambar 4.6

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

public class VRLookWalk : MonoBehaviour
{
    public Transform vrCamera;
    public float toggleAngle = 30.0f;
    public float speed = 3.0f;
    public bool moveForward;

    private CharacterController cc;

    // Use this for initialization
    void Start()
    {
        cc = GetComponent<CharacterController>();
    }

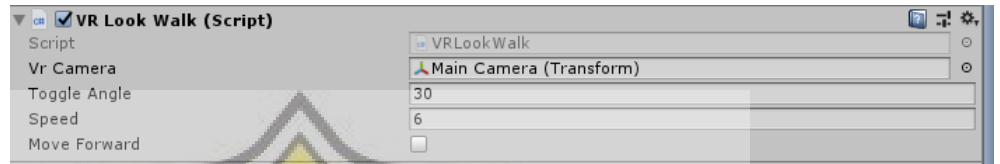
    // Update is called once per frame
    void Update()
    {
        if (GameManager.isMove)
        {
            //untuk menyetting syarat agar karakter bisa berjalan
            if (vrCamera.eulerAngles.x >= toggleAngle &&
                vrCamera.eulerAngles.x < 90.0f)
            {
                moveForward = true;
            }
            else
            {
                moveForward = false;
            }

            //memberikan kecepatan berjalan
            if (moveForward)
            {
                Vector3 forward =
                vrCamera.TransformDirection(Vector3.forward);

                cc.SimpleMove(forward * speed);
            }
        }
    }
}
```


Gambar 4.6 Script Movement player

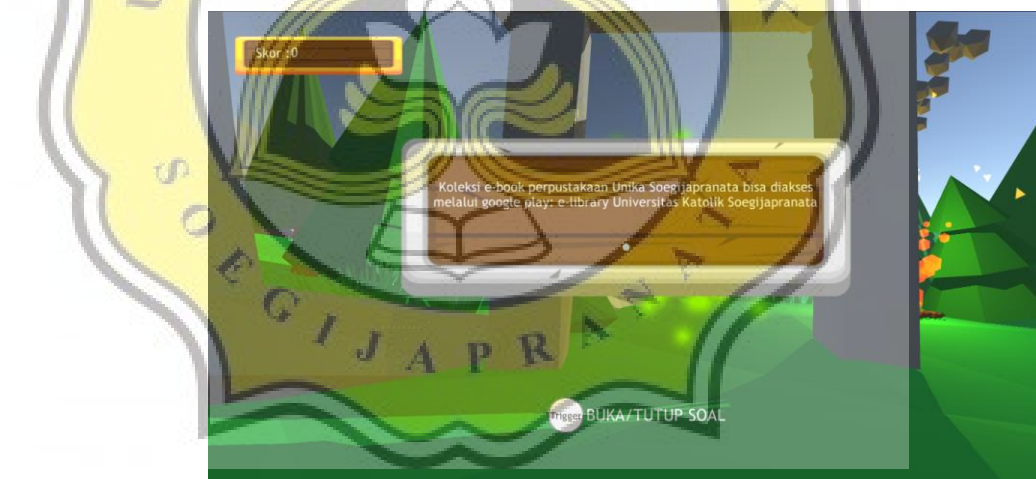
Pada Game Object “Player” Script ini hanya perlu ditambahkan dan masukkan “Main Camera” pada kolom “Vr Camera” yang terdapat pada Inspector seperti pada Gambar 4.7



Gambar 4.7 Inspector Player

B. Soal dan Kunci jawaban

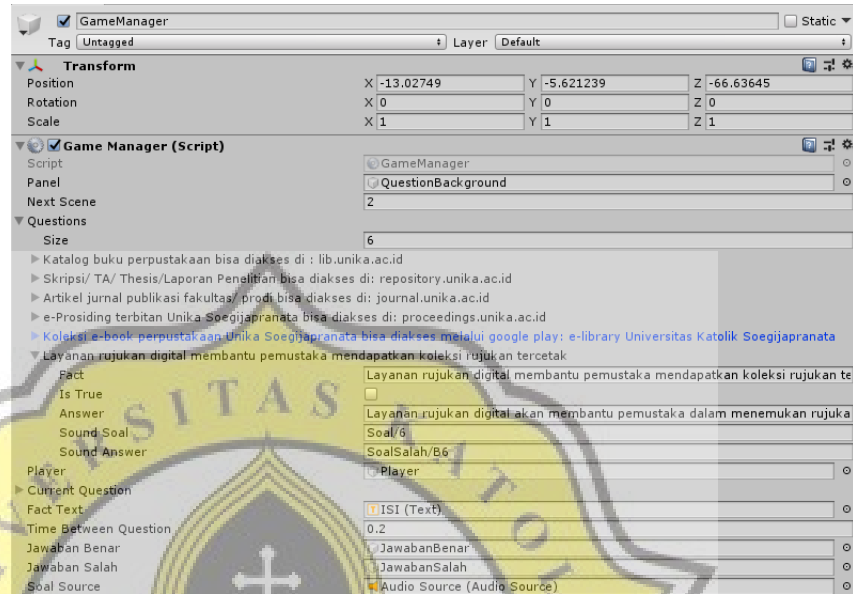
Ketika menekan tombol mulai maka akan muncul Papan soal seperti pada Gambar 4.8



Gambar 4.8 Tampilan masuk game

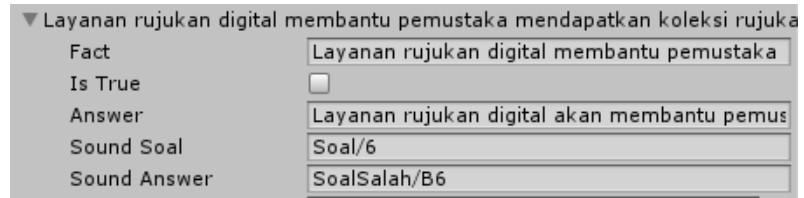
Soal yang disediakan pada level ini terdiri dari 6 soal namun bisa dibuat lebih atau kurang berdasarkan keinginan perpustakaan dikarenakan soal-soal yang disediakan pada game dapat diedit dengan

mudah melalui GameObject bernama “GameManager” dengan inspector seperti pada gambar 4.9.



Gambar 4.9 Inspector pada GameManager

Untuk mengubah soal dan kunci jawaban bisa dilakukan dengan mudah dengan cara membuka salah satu soal atau menambah size pada “Question” sehingga akan terlihat seperti pada Gambar 4.10. Cara untuk mengubah soal adalah pada kolom bagian “fact” untuk pernyataannya, apabila benar centang kolom “Is True” namun apa pernyataan tersebut salah kosongkan kolom “Is True” tersebut. Selain itu terdapat kolom “Sound Soal” dan “Sound Answer” yang fungsinya untuk memberikan audio atau suara yang berisi soal dan kebenaran dari pernyataan tersebut.



Gambar 4.10 Inspector pada GameManager

Kolom Answer yang terdapat pada Inspector GameManager berfungsi untuk memberikan keterangan soal secara tertulis dan lisan setelah player selesai menjawab sebuah pernyataan seperti pada gambar 4.11.



Gambar 4.11 Keterangan pernyataan

Pada Gambar 4.12 adalah script GameManager yang berfungsi sebagai script yang membuat sebuah list sebagai tempat untuk soal soal, memunculkan soal pada interface game, mengacak pertanyaan pada list, panel untuk membuka dan menutup soal, serta perpindahan scene Ketika soal sudah habis.

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

```

using UnityEngine.SceneManagement;

public class GameManager : MonoBehaviour
{
    public GameObject Panel;
    int State = 0;

    public int nextScene = 0;

    public Question[] questions;
    public GameObject player;
    private static List<Question> unansweredQuestions;
    //public static List<Question> unansweredQuestions;

    //private Question currentQuestion;
    public Question currentQuestion ;

    [SerializeField]
    private Text factText;

    [SerializeField]
    public float timeBetweenQuestion = 0.5f;

    public GameObject jawabanBenar;
    public GameObject jawabanSalah;

    public AudioSource soalSource;

    public static bool isMove;
    // Start is called before the first frame update
    void Start()
    {
        //mengisi soal pada list
        if(unansweredQuestions == null || unansweredQuestions.Count == 0)
        {
            unansweredQuestions = questions.ToList<Question>();
        }
        isMove = true;
        SetCurrentQuestion();
    }

    void SetCurrentQuestion()
    {
        //merandom pertanyaan
        int randomQuestionIndex = Random.Range(0,
            unansweredQuestions.Count);
        currentQuestion = unansweredQuestions[randomQuestionIndex];

        //mengeluarkan soal pada UI
        factText.text = currentQuestion.fact;
        soalSource.clip = Resources.Load<AudioClip>(currentQuestion.soundSoal);
        soalSource.Play();
    }
}

```

```

}

void Update()
{
}

public IEnumerator TransitionToNextQuestion()
{
    unansweredQuestions.Remove(currentQuestion);
    Debug.Log(unansweredQuestions.Count);

    //ketika pertanyaan habis , langsung lompat ke scene selanjutnya
    if (unansweredQuestions.Count==0)
    {
        yield return new WaitForSeconds(timeBetweenQuestion);
        yield return new WaitForSeconds(5f);
        SceneManager.LoadScene(nextScene);
        unansweredQuestions = questions.ToList<Question>();
    }

    yield return new WaitForSeconds(timeBetweenQuestion);
    soalSource.Stop();
    player.transform.position = new Vector3(-14.37f, -8.54f, -72.78f);
    isMove = false;
    yield return new WaitForSeconds(5f);
    isMove = true;
    jawabanBenar.SetActive(false);
    jawabanSalah.SetActive(false);
    Panel.SetActive(true);

    Start();
}
}

```

Gambar 4.12 Script Game Manager

Pada Gambar 4.13 adalah script untuk form soal dan kunci jawaban yang nantinya diulang berdasarkan berapa soal yang akan dibuat.

```

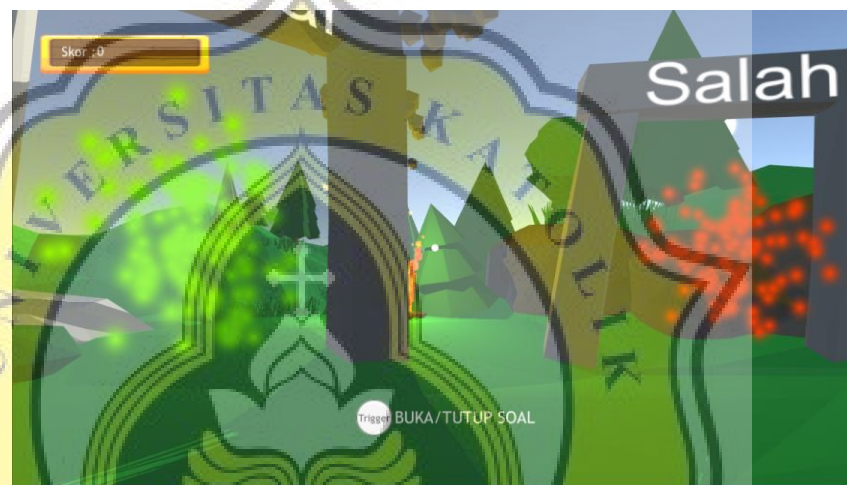
[System.Serializable]
public class Question
{
    //form untuk pertanyaan , kunci jawaban game
    public string fact;
    public bool isTrue;
    public string answer;
    public string soundSoal;
    public string soundAnswer;
}

```

Gambar 4.13 Script Form Soal

C. Tombol untuk buka tutup soal

Ketika di dalam permainan pemain akan langsung dihadapkan pada papan soal yang menutupi pandangan. Untuk menutup papan soal tersebut pemain dapat menekan tombol trigger untuk menutup papan soal tersebut seperti pada gambar 4.14.



Gambar 4.14 Menutup soal pada permainan level 1

Script untuk menutup soal bisa dilihat pada gambar 4.15.

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

public class Tombol : MonoBehaviour
{
    public GameObject Panel;
    int State = 0;

    void Start()
    {
    }

    void Update()
```

```

{
  //buka tutup soal
  if (Input.GetButtonDown("Fire1"))
  {
    //tutup soal
    Debug.Log(State);
    if (State == 0)
    {
      State = 1;
      Panel.SetActive(false);
    }
    //buka soal
    else if (State == 1)
    {
      Panel.SetActive(true);
      State = 0;
    }
  }
}
}
}

```

Gambar 4.15 Script untuk menutup soal pada permainan level 1

D. Portal

Ketika player menjawab sebuah pernyataan dengan memasuki sebuah portal Seperti pada gambar 4.16 maka game akan memeriksa jawaban tersebut apakah sesuai dengan kunci jawaban yang ada.

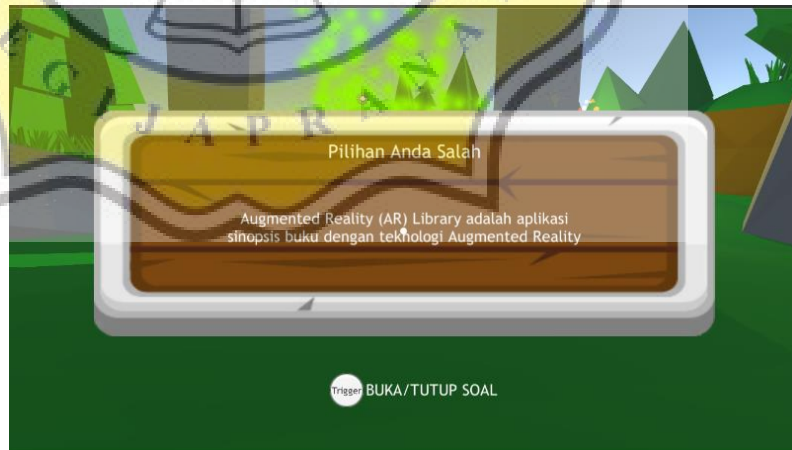


Gambar 4.16 Portal pilihan benar dan salah

Bila sesuai maka akan menambah skor sebesar 5 poin serta muncul keterangan secara tertulis dan lisan yaitu “Pilihan Anda Benar” seperti pada Gambar 4.17.



Gambar 4.17 Apabila jawaban sesuai dengan kunci jawaban namun bila tidak sesuai maka akan muncul keterangan secara tertulis dan lisan yaitu “Pilihan Anda Salah” Gambar 4.18.



Gambar 4.18 Apabila jawaban tidak sesuai dengan kunci jawaban

Untuk script pencocokan tersebut masing masing portal memiliki script yang berbeda, untuk script pada portal benar terdapat pada Gambar 4.19.

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

public class True : MonoBehaviour
{
    public string InputJawaban;

    GameManager gm;
    public GameObject jawabanBenar;
    public GameObject jawabanSalah;
    public Text jawabanBenarText;
    public Text jawabanSalahText;
    public AudioSource benarSource;
    public AudioSource salahSource;

    // Start is called before the first frame update
    void Start()
    {
        gm =
        GameObject.Find("GameManager").GetComponent<GameManager>();
    }

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

    IEnumerator AddJawab(string jawaban_salah)
    {
        WWWForm form = new WWWForm();
        form.AddField("addJawabanSalah", jawaban_salah);

        using (UnityWebRequest www =
        UnityWebRequest.Post("http://lib.unika.ac.id/skripsi/ricky/dataIn
sert.php", form))
        {
            yield return www.SendWebRequest();

            if (www.isNetworkError || www.isHttpError)
            {
                Debug.Log(www.error);
            }
        }
    }
}
```

```

    }
    else
    {
        Debug.Log("Form upload complete!");
    }
}
}

//ketika menyentuh gerbang
public void OnTriggerEnter(Collider coll)
{
    //apabila jawaban sesuai dengan kunci jawaban maka....
    if (gm.currentQuestion.isTrue)
    {
        jawabanBenar.SetActive(true);
        jawabanBenarText.text = gm.currentQuestion.answer;
        Skor.scoreValue += 5;
        benarSource.Play();

        StartCoroutine(gm.TransitionToNextQuestion());
    }
    //apabila jawaban tidak sesuai dengan kunci jawaban
    maka....
    else
    {
        StartCoroutine(AddJawab(gm.currentQuestion.fact));
        Debug.Log(gm.currentQuestion.fact);

        jawabanSalah.SetActive(true);
        jawabanSalahText.text = gm.currentQuestion.answer;
        salahSource.Play();

        StartCoroutine(gm.TransitionToNextQuestion());
    }
}
}
}

```

Gambar 4.19 Script portal benar

Untuk script pada portal salah terdapat pada Gambar 4.20.

```

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

public class Falsee: MonoBehaviour

```

```

{
    GameManager gm;
    public GameObject jawabanBenar;
    public GameObject jawabanSalah;
    public Text jawabanBenarText;
    public Text jawabanSalahText;
    public AudioSource benarSource;
    public AudioSource salahSource;

    // Start is called before the first frame update
    void Start()
    {
        gm =
        GameObject.Find("GameManager").GetComponent<GameManager>();
    }

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

    IEnumerator AddJawab(string jawaban_salah)
    {
        WWWForm form = new WWWForm();
        form.AddField("addJawabanSalah", jawaban_salah);

        using (UnityWebRequest www =
        UnityWebRequest.Post("http://lib.unika.ac.id/skripsi/ricky/dataIn
        sert.php", form ))
        {
            yield return www.SendWebRequest();

            if (www.isNetworkError || www.isHttpError)
            {
                Debug.Log(www.error);
            }
            else
            {
                Debug.Log("Form upload complete!");
            }
        }
    }
}

//ketika menyentuh gerbang
public void OnTriggerEnter(Collider coll)
{
    //apabila jawaban sesuai dengan kunci jawaban maka....
    if (!gm.currentQuestion.isTrue)
    {
        jawabanBenar.SetActive(true);
        jawabanBenarText.text = gm.currentQuestion.answer;
    }
}

```

```

        Skor.scoreValue += 5;
        benarSource.Play();
        StartCoroutine(gm.TransitionToNextQuestion());
    }
    else
    //apabila jawaban tidak sesuai dengan kunci jawaban
    maka....
    {
        StartCoroutine(AddJawab(gm.currentQuestion.fact));
        Debug.Log(gm.currentQuestion.fact);

        jawabanSalah.SetActive(true);
        jawabanSalahText.text = gm.currentQuestion.answer;
        salahSource.Play();

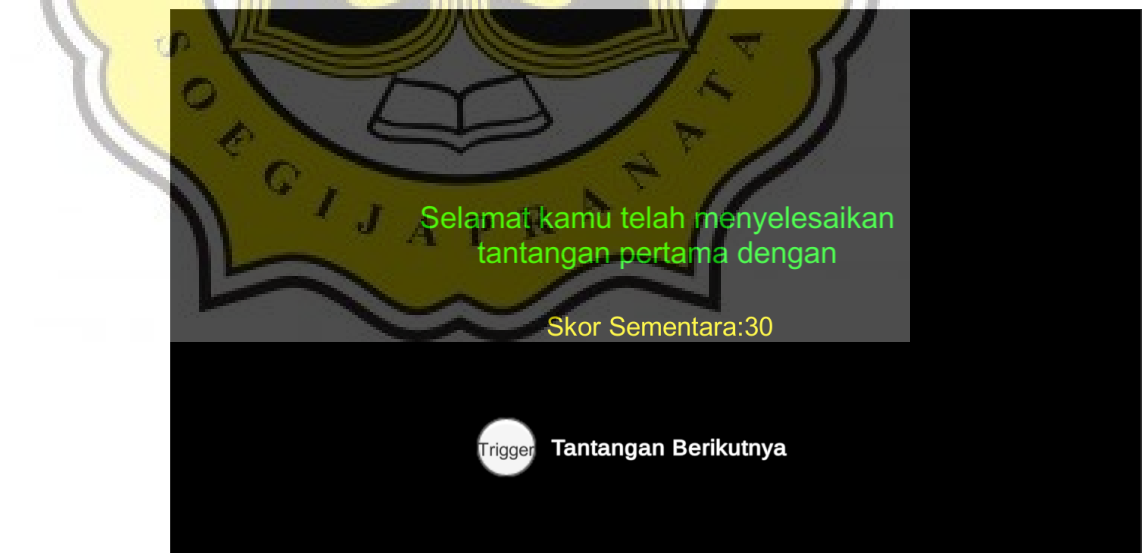
        StartCoroutine(gm.TransitionToNextQuestion());
    }
}
}
}

```

Gambar 4.20 Script Portal salah

E. Pergantian Level

Ketika selesai menjawab semua soal yang ada pada level 1 maka akan terjadi perpindahan scene yang menunjukkan hasil skor sementara pada level 1 seperti pada gambar 4.21.



Gambar 4.21 Scene yang menunjukkan hasil skor sementara level 1

Ketika player menekan tombol trigger akan muncul halaman tutorial untuk permainan level 2 seperti pada gambar 4.22.



Gambar 4.22 Tutorial kontrol karakter pada level 2

Selain itu terdapat panduan apa yang harus dilakukan pemain untuk menjawab benar atau salah pada level 2 seperti pada gambar 4.23.



Gambar 4.23 Panduan permainan pada level 2

Script yang digunakan untuk menampilkan skor sementara, panduan dan tutorial terdapat pada gambar 4.24.

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

public class SkorSementara : MonoBehaviour
{
    public Text scoreText;
```

```

public int nextScene;

public GameObject tutor;
public GameObject tutor1;
int astate = 0;

// Start is called before the first frame update
void Start()
{

}

// Update is called once per frame
void Update()
{
    if (Input.GetButtonDown("Fire1"))
    {
        Debug.Log(astate);
        if (astate == 0)
        {
            astate = 1;
            tutor.SetActive(true);
        }
        else if (astate == 1)
        {
            astate = 2;
            tutor.SetActive(false);
            tutor1.SetActive(true);
        }
        else if (astate == 2)
        {
            astate = 0;
            SceneManager.LoadScene(nextScene);
        }
    }
    //memberikan text score pada UI
    scoreText.text = "Skor Sementara:" + Skor.scoreValue;
}
}

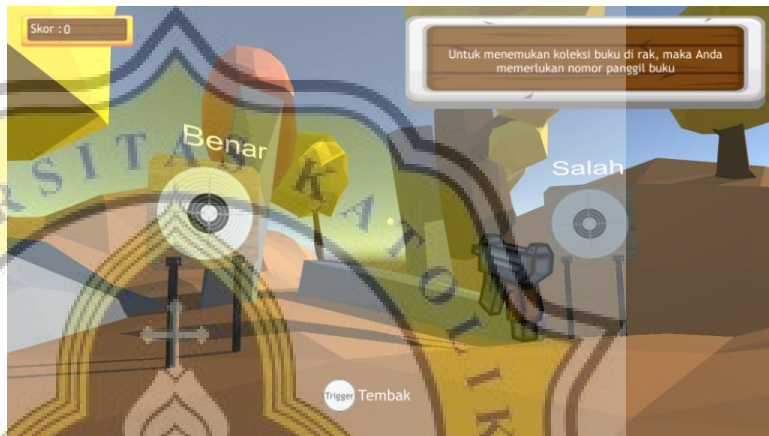
```

Gambar 4.24 Script untuk menampilkan skor sementara, tutorial, dan perpindahan level

4.5.3 Permainan Level 2

A. Menembak

Untuk level kedua player dibuat tidak bisa bergerak. Hal yang dilakukan pada level kedua ini adalah player membidik dan menembak target sasaran benar atau salah seperti pada gambar 4.25



Gambar 4.25 Level kedua pada permainan

Script yang digunakan untuk menembak terdapat pada gambar 4.26.

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

public class shoot : MonoBehaviour
{
    public float damage = 10f;
    public float range = 100f;

    public Camera fpsCam;
    public ParticleSystem muzzleFlash;
    public GameObject impactEffect;
    AudioSource shooting_sound;

    // Start is called before the first frame update
    void Start()
    {
        shooting_sound = GetComponent<AudioSource>();
    }
}
```

```

// Update is called once per frame
void Update()
{
    if (GameManager2.isMove)
    {
        if (Input.GetButtonDown("Fire1"))
        {
            shooting_sound.Play();
            Shoot();
        }
    }
}
void Shoot()
{
    muzzleFlash.Play();

    RaycastHit hit;
    if (Physics.Raycast(fpsCam.transform.position,
fpsCam.transform.forward, out hit, range))
    {
        Debug.Log(hit.transform.name);

        TrueDemo3 target =
hit.transform.GetComponent<TrueDemo3>();
        if (target != null)
        {
            target.TakeDamage(damage);
        }

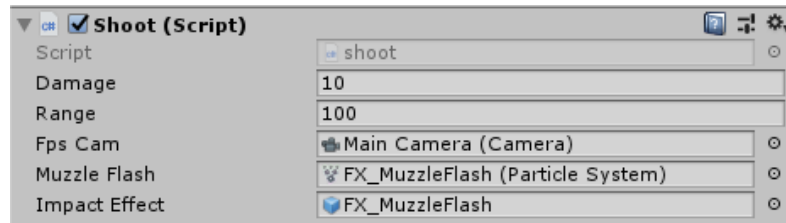
        FalseDemo3 target2 =
hit.transform.GetComponent<FalseDemo3>();
        if (target2 != null)
        {
            target2.TakeDamage(damage);
        }

        GameObject impactGO = Instantiate(impactEffect,
hit.point, Quaternion.LookRotation(hit.normal));
        Destroy(impactGO, 1f);
    }
}
}

```

Gambar 4.26 Script untuk menembak target

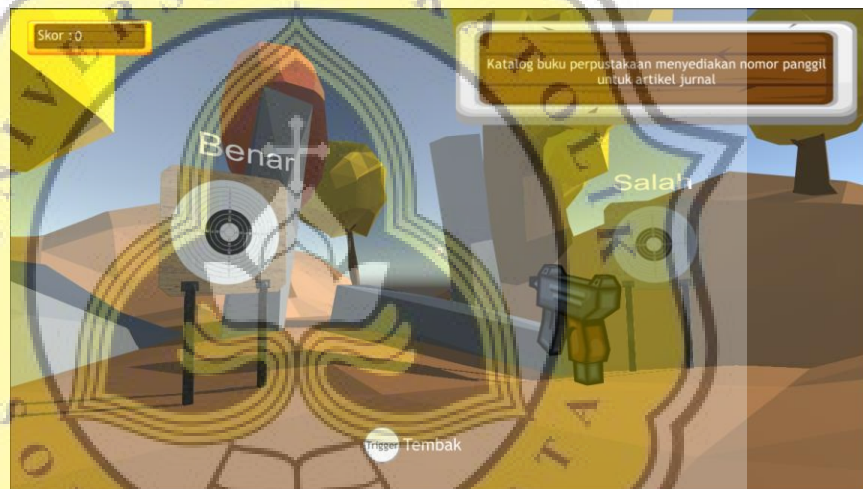
Untuk Fps cam diisi dengan “main camera”, sedangkan muzzle flash dan impact effect adalah efek saat menembak objek. Inspector pada script menembak terdapat pada gambar 4.27.



Gambar 4.27 Inspector script pada player

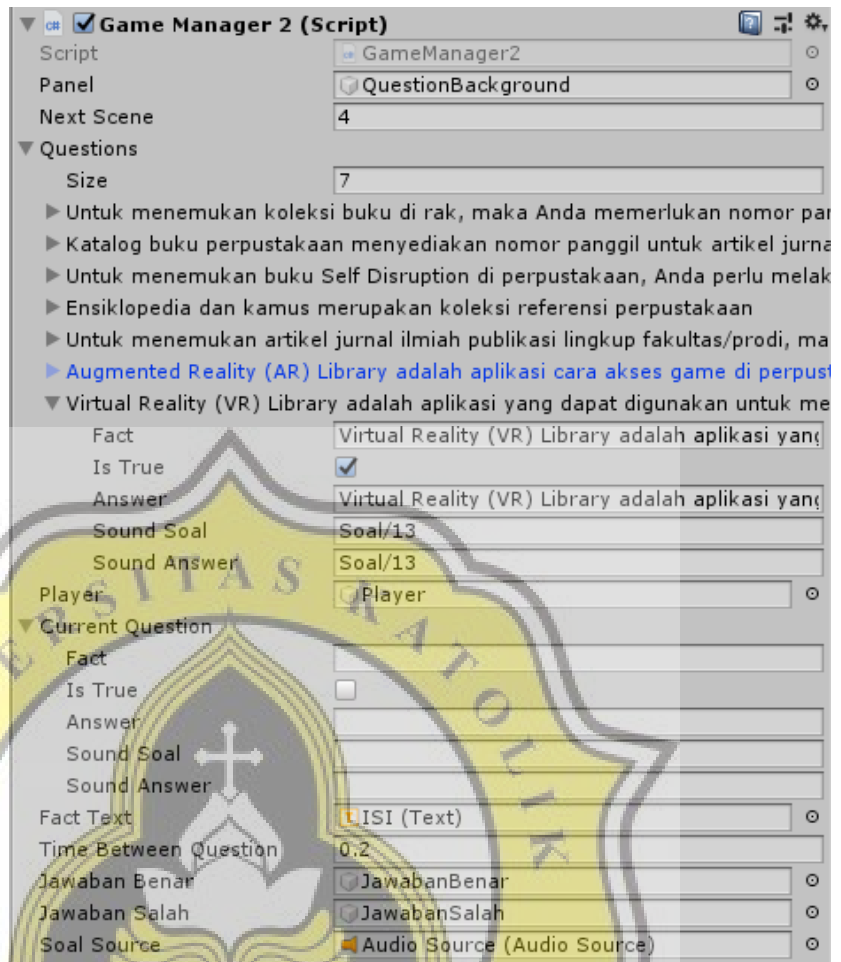
B. Soal dan Kunci Jawaban

Setelah pergantian level dan pemain menekan tombol trigger maka scene akan berlanjut ke permainan level 2 seperti pada gambar 4.28.



Gambar 4.28 Scene permainan level 2

Soal yang disediakan pada level ini terdiri dari 7 soal namun bisa dibuat lebih atau kurang berdasarkan keinginan perpustakaan dikarenakan soal-soal yang disediakan pada game dapat diedit dengan mudah melalui GameObject bernama “GameManager2” dengan inspector seperti pada gambar 4.29.



Gambar 4.29 Inspector GameManager2

GameManager pada level ini masih memiliki fungsi dan cara kerjanya sama dengan game manager level 1 script terdapat pada gambar 4.30.

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

public class GameManager2 : MonoBehaviour
{
    public GameObject Panel;
    int State = 0;

    public int nextScene = 0;
}
```

```

public Question[] questions;
public GameObject player;
private static List<Question> unansweredQuestions;
//public static List<Question> unansweredQuestions;

//private Question currentQuestion;
public Question currentQuestion ;

[SerializeField]
private Text factText;

[SerializeField]
public float timeBetweenQuestion = 0.5f;

public GameObject jawabanBenar;
public GameObject jawabanSalah;

public AudioSource soalSource;

public static bool isMove;
// Start is called before the first frame update
void Start()
{
    //mengisi soal pada list
    if(unansweredQuestions == null ||
unansweredQuestions.Count == 0)
    {
        unansweredQuestions = questions.ToList<Question>();
    }
    isMove = true;
    SetCurrentQuestion();
}
void SetCurrentQuestion()
{
    //merandom pertanyaan
    int randomQuestionIndex = Random.Range(0,
unansweredQuestions.Count);
    currentQuestion =
unansweredQuestions[randomQuestionIndex];

    //mengeluarkan soal pada UI
    factText.text = currentQuestion.fact;
    soalSource.clip =
Resources.Load<AudioClip>(currentQuestion.soundSoal);
    soalSource.Play();

}

void Update()
{
}

```

```

public IEnumerator TransitionToNextQuestion()
{
    unansweredQuestions.Remove(currentQuestion);
    Debug.Log(unansweredQuestions.Count);

    //ketika pertanyaan habis , langsung lompat ke scene 2
    if (unansweredQuestions.Count==0)
    {
        yield return new WaitForSeconds(timeBetweenQuestion);
        yield return new WaitForSeconds(5f);
        SceneManager.LoadScene(nextScene);
        unansweredQuestions = questions.ToList<Question>();
    }

    yield return new WaitForSeconds(timeBetweenQuestion);
    soalSource.Stop();
    player.transform.position = new Vector3(-14.37f, -8.54f,
-72.78f);
    isMove = false;
    yield return new WaitForSeconds(5f);
    isMove = true;
    jawabanBenar.SetActive(false);
    jawabanSalah.SetActive(false);
    Panel.SetActive(true);

    Start();
}
}

```

Gambar 4.30 Script game manager 2

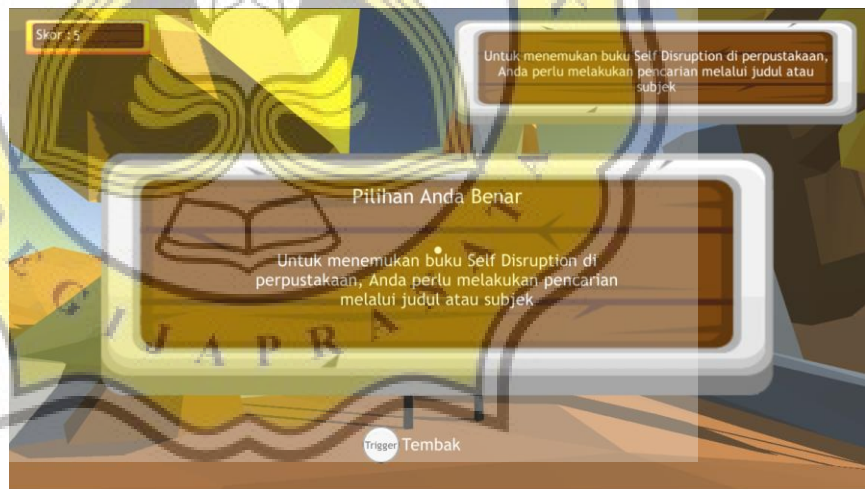
C. Target

Ketika player menjawab sebuah pernyataan dengan menembak sebuah target seperti pada gambar 4.31 maka game akan memeriksa jawaban tersebut apakah sesuai dengan kunci jawaban yang ada.



Gambar 4.31 Target pada permainan level 2

Bila sesuai maka akan menambah skor sebanyak 5 poin serta muncul keterangan secara tertulis dan lisan yaitu “Pilihan Anda Benar” seperti pada Gambar 4.32.



Gambar 4.32 keterangan benar pada level 2

Apabila tidak sesuai maka akan muncul keterangan secara tertulis dan lisan yaitu “Pilihan Anda Salah” Gambar 4.33.



Gambar 4.33 keterangan salah pada level 2

Untuk script pencocokan jawaban masing masing target memiliki script yang berbeda, untuk script pada target benar terdapat pada Gambar 4.34.

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

public class TrueDemo3 : MonoBehaviour
{
    public string InputJawaban;

    GameManager2 gm;
    public GameObject jawabanBenar;
    public GameObject jawabanSalah;
    public Text jawabanBenarText;
    public Text jawabanSalahText;
    public AudioSource benarSource;
    public AudioSource salahSource;

    // Start is called before the first frame update
    void Start()
    {
        gm =
        GameObject.Find("GameManager").GetComponent<GameManager2>();
    }

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

```

}

IEnumerator AddJawab(string jawaban_salah)
{
    WWWForm form = new WWWForm();
    form.AddField("addJawabanSalah", jawaban_salah);

    using (UnityWebRequest www =
UnityWebRequest.Post("http://lib.unika.ac.id/skripsi/ricky/dataIn
sert.php", form))
    {
        yield return www.SendWebRequest();

        if (www.isNetworkError || www.isHttpError)
        {
            Debug.Log(www.error);
        }
        else
        {
            Debug.Log("Form upload complete!");
        }
    }
}

public float health = 10f;
public void TakeDamage(float amount)
{
    health -= amount;
    if (health <= 0f)
    {
        if (gm.currentQuestion.isTrue)
        {
            jawabanBenar.SetActive(true);
            jawabanBenarText.text =
gm.currentQuestion.answer;
            Skor.scoreValue += 5;
            benarSource.Play();

            StartCoroutine(gm.TransitionToNextQuestion());
        }
        else
        {

StartCoroutine(AddJawab(gm.currentQuestion.fact));

            jawabanSalah.SetActive(true);
            jawabanSalahText.text =
gm.currentQuestion.answer;
            salahSource.Play();

            StartCoroutine(gm.TransitionToNextQuestion());
        }
    }
}

```

```

    }
}
}
}

```

Gambar 4.34 Script pada target benar

Untuk script pada target salah terdapat pada Gambar 4.35.

```

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

public class FalseDemo3 : MonoBehaviour
{
    GameManager2 gm;
    public GameObject jawabanBenar;
    public GameObject jawabanSalah;
    public Text jawabanBenarText;
    public Text jawabanSalahText;
    public AudioSource benarSource;
    public AudioSource salahSource;
    // Start is called before the first frame update
    void Start()
    {
        gm =
        GameObject.Find("GameManager").GetComponent<GameManager2>();
    }
    // Update is called once per frame
    void Update()
    {
    }
    IEnumerator AddJawab(string jawaban_salah)
    {
        WWWForm form = new WWWForm();
        form.AddField("addJawabanSalah", jawaban_salah);

        using (UnityWebRequest www =
        UnityWebRequest.Post("http://lib.unika.ac.id/skripsi/ricky/dataIn
        sert.php", form))
        {
            yield return www.SendWebRequest();

            if (www.isNetworkError || www.isHttpError)
            {
                Debug.Log(www.error);
            }
            else

```



```

        {
            Debug.Log("Form upload complete!");
        }
    }
}

public float health = 10f;

public void TakeDamage(float amount)
{
    health -= amount;
    if (health <= 0f)
    {
        if (!gm.currentQuestion.isTrue)
        {
            jawabanBenar.SetActive(true);
            jawabanBenarText.text =
gm.currentQuestion.answer;
            Skor.scoreValue += 5;
            benarSource.Play();

            StartCoroutine(gm.TransitionToNextQuestion());
        }
        else
        {
            StartCoroutine(AddJawab(gm.currentQuestion.fact));

            jawabanSalah.SetActive(true);
            jawabanSalahText.text =
gm.currentQuestion.answer;
            salahSource.Play();

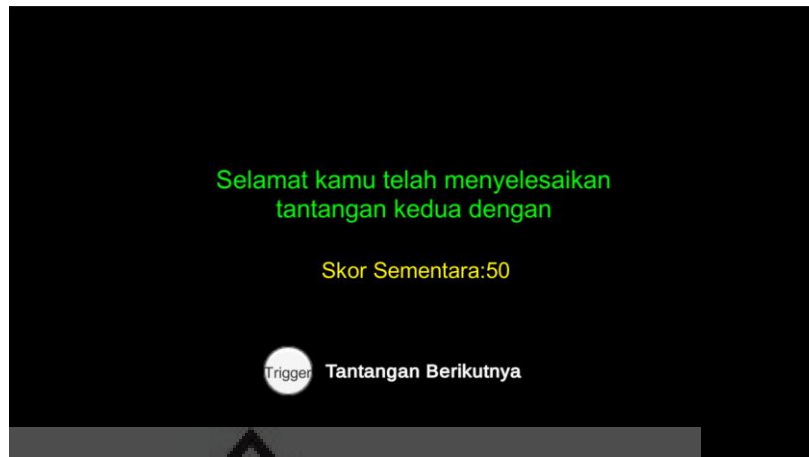
            StartCoroutine(gm.TransitionToNextQuestion());
        }
    }
}
}
}

```

Gambar 4.35 Script pada target salah

D. Pergantian Level

Ketika selesai menjawab semua soal yang ada pada level 2 maka akan terjadi perpindahan scene yang menunjukkan hasil skor sementara yang didapat pada level 2 seperti pada gambar 4.36.



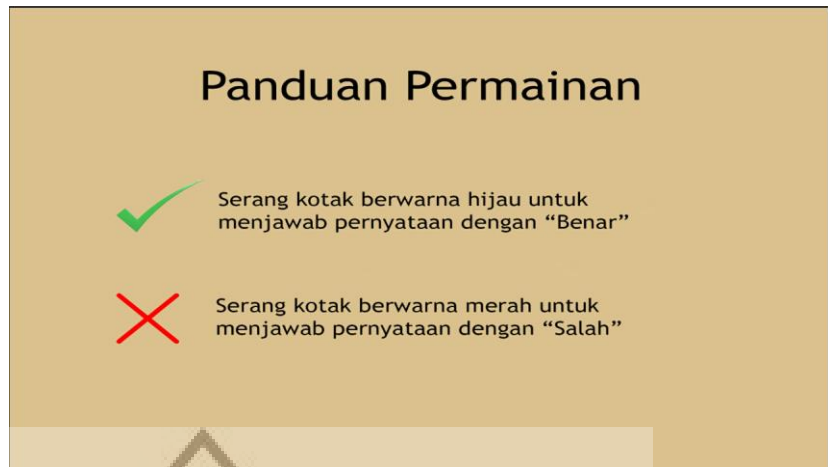
Gambar 4.36 Scene skor sementara level 2

Ketika player menekan tombol trigger akan muncul halaman tutorial untuk permainan level 3 seperti pada gambar 4.37.



Gambar 4.37 Tutorial kontrol karakter pada level 3

Selain itu terdapat panduan apa yang harus dilakukan pemain untuk menjawab benar atau salah pada level 3 seperti pada gambar 4.38.



Gambar 4.38 Panduan permainan pada level 3

Script yang digunakan untuk menampilkan skor sementara, panduan, dan tutorial terdapat pada gambar 4.39.

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

public class SkorSementara : MonoBehaviour
{
    public Text scoreText;
    public int nextScene;

    public GameObject tutor;
    public GameObject tutor1;
    int astate = 0;

    // Start is called before the first frame update
    void Start()
    {

    }

    // Update is called once per frame
    void Update()
    {
        if (Input.GetButtonDown("Fire1"))
        {
            Debug.Log(astate);
            if (astate == 0)
            {
                astate = 1;
                tutor.SetActive(true);
            }
            else if (astate == 1)
            {
                astate = 0;
                tutor1.SetActive(true);
            }
        }
    }
}
```

```

    {
        astate = 2;
        tutor.SetActive(false);
        tutor1.SetActive(true);
    }
    else if (astate == 2)
    {
        astate = 0;
        SceneManager.LoadScene(nextScene);
    }
}
//memberikan text score pada UI
scoreText.text = "Skor Sementara:" + Skor.scoreValue;
}
}

```

Gambar 4.39 Script untuk menampilkan skor sementara, tutorial, dan perpindahan level

4.5.4 Permainan Level 3

A. Movement Player

Pada level ketiga ini kontrol dibuat dengan cara player berjalan dan menyerang. Ketika player menundukkan kepalanya karakter dalam game akan berjalan maju. Script yang digunakan terdapat pada Gambar 4.40.

```

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

public class VRLookWalk : MonoBehaviour
{
    public Transform vrCamera;
    public float toggleAngle = 30.0f;
    public float speed = 3.0f;
    public bool moveForward;

    private CharacterController cc;

    // Use this for initialization
    void Start()
    {
        cc = GetComponent<CharacterController>();
    }

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

```

```

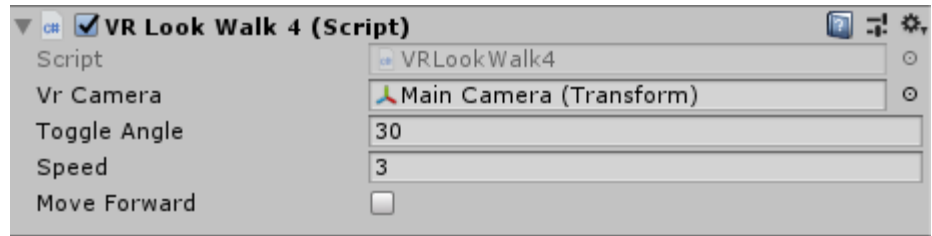
{
  if (GameManager.isMove)
  {
    //untuk menyetting syarat agar karakter bisa berjalan
    if (vrCamera.eulerAngles.x >= toggleAngle &&
vrCamera.eulerAngles.x < 90.0f)
    {
      moveForward = true;
    }
    else
    {
      moveForward = false;
    }

    //memberikan kecepatan berjalan
    if (moveForward)
    {
      Vector3 forward =
vrCamera.TransformDirection(Vector3.forward);
      cc.SimpleMove(forward * speed);
    }
  }
}

```

Gambar 4.40 Script Movement player

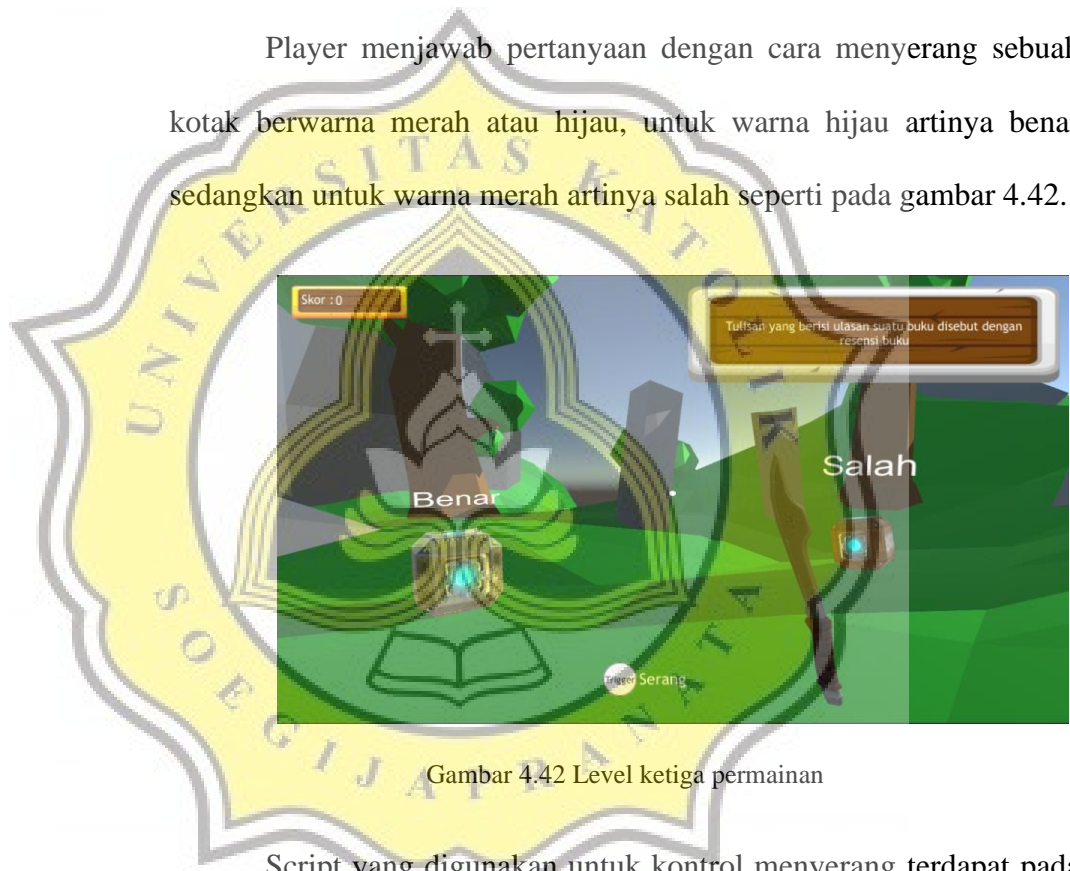
Pada Game Object “Player” Script ini hanya perlu ditambahkan dan masukkan “Main Camera” pada kolom “Vr Camera” yang terdapat pada Inspector seperti pada Gambar 4.41



Gambar 4.41 Inspector Player

B. Menyerang

Player menjawab pertanyaan dengan cara menyerang sebuah kotak berwarna merah atau hijau, untuk warna hijau artinya benar sedangkan untuk warna merah artinya salah seperti pada gambar 4.42.



Gambar 4.42 Level ketiga permainan

Script yang digunakan untuk kontrol menyerang terdapat pada gambar 4.43.

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

public class Slash : MonoBehaviour
{
    public float damage = 10f;
    public float range = 100f;
}
```

```

public Camera fpsCam;
public GameObject Sword;
public GameObject impactEffect;
AudioSource slashing_sound;

// Start is called before the first frame update
void Start()
{
    slashing_sound = GetComponent();
}

// Update is called once per frame
void Update()
{
    if (GameManager3.isMove)
    {
        if (Input.GetButtonDown("Fire1"))
        {
            slashing_sound.Play();
            slash();
        }
    }
}

void slash()
{
    Sword.GetComponent<Animator>().Play("sword");

    RaycastHit hit;
    if (Physics.Raycast(fpsCam.transform.position,
        fpsCam.transform.forward, out hit, range))
    {
        Debug.Log(hit.transform.name);

        TrueDemo4 target =
        hit.transform.GetComponent<TrueDemo4>();
        if (target != null)
        {
            target.TakeDamage(damage);
        }

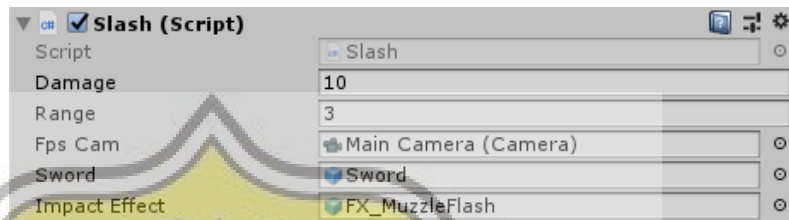
        FalseDemo4 target2 =
        hit.transform.GetComponent<FalseDemo4>();
        if (target2 != null)
        {
            target2.TakeDamage(damage);
        }

        GameObject impactGO = Instantiate(impactEffect,
        hit.point, Quaternion.LookRotation(hit.normal));
        Destroy(impactGO, 1f);
    }
}
}

```

Gambar 4.43 Script menyerang

Untuk fps cam diisi dengan “main camera”, muzzle flash dan impact effect adalah efek pada saat menyerang. Inspector pada player dapat dilihat pada gambar 4.44.



Gambar 4.44 Inspector script serang pada player

C. Soal dan Kunci jawaban

Setelah pergantian level dan pemain menekan tombol trigger maka scene akan berlanjut ke permainan level 3 seperti pada gambar 4.45.



Gambar 4.45 Scene permainan level 3

Soal yang disediakan pada level ini terdiri dari 7 soal namun bisa dibuat lebih atau kurang berdasarkan keinginan perpustakaan

dikarenakan soal-soal yang disediakan pada game dapat diedit dengan mudah melalui GameObject bernama “GameManager3” dengan inspector seperti pada gambar 4.46.



Gambar 4.46 Inspector GameManager3

GameManager pada level ini masih memiliki fungsi dan cara kerjanya sama dengan game manager level 1 dan level 2, script terdapat pada gambar 4.47.

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

```

using System.Linq;
using UnityEngine.SceneManagement;

public class GameManager3 : MonoBehaviour
{
    public GameObject Panel;
    int State = 0;

    public int nextScene = 0;

    public Question[] questions;
    public GameObject player;
    private static List<Question> unansweredQuestions;
    //public static List<Question> unansweredQuestions;

    //private Question currentQuestion;
    public Question currentQuestion ;

    [SerializeField]
    private Text factText;

    [SerializeField]
    public float timeBetweenQuestion = 0.5f;

    public GameObject jawabanBenar;
    public GameObject jawabanSalah;

    public AudioSource soalSource;

    public static bool isMove;
    // Start is called before the first frame update
    void Start()
    {
        //mengisi soal pada list
        if(unansweredQuestions == null ||
unansweredQuestions.Count == 0)
        {
            unansweredQuestions = questions.ToList<Question>();
        }
        isMove = true;
        SetCurrentQuestion();
    }

    void SetCurrentQuestion()
    {
        //merandom pertanyaan
        int randomQuestionIndex = Random.Range(0,
unansweredQuestions.Count);
        currentQuestion =
unansweredQuestions[randomQuestionIndex];

        //mengeluarkan soal pada UI
        factText.text = currentQuestion.fact;
    }
}

```

```

        soalSource.clip =
Resources.Load<AudioClip>(currentQuestion.soundSoal);
        soalSource.Play();

    }

    void Update()
    {

    }

    public IEnumerator TransitionToNextQuestion()
    {
        unansweredQuestions.Remove(currentQuestion);
        Debug.Log(unansweredQuestions.Count);

        //ketika pertanyaan habis , langsung lompat ke scene 2
        if (unansweredQuestions.Count==0)
        {
            yield return new WaitForSeconds(timeBetweenQuestion);
            yield return new WaitForSeconds(5f);
            SceneManager.LoadScene(nextScene);
            unansweredQuestions = questions.ToList<Question>();
        }

        yield return new WaitForSeconds(timeBetweenQuestion);
        soalSource.Stop();
        player.transform.position = new Vector3(-14.37f, -8.54f,
-72.78f);
        isMove = false;
        yield return new WaitForSeconds(5f);
        isMove = true;
        jawabanBenar.SetActive(false);
        jawabanSalah.SetActive(false);
        Panel.SetActive(true);

        Start();
    }
}

```

Gambar 4.47 Script game manager 3

D. Kotak Target

Ketika player menjawab sebuah pernyataan dengan menyerang sebuah kotak target seperti pada gambar 4.48 maka game akan memeriksa jawaban tersebut apakah sesuai dengan kunci jawaban yang ada.



Gambar 4.48 Kotak target pada permainan level 3

Bila player menjawab sesuai dengan kunci jawaban maka akan mendapatkan skor 5 poin serta muncul keterangan secara tertulis dan lisan yaitu “Pilihan Anda Benar” seperti pada Gambar 4.49.



Gambar 4.49 keterangan benar pada level 3

Apabila player menjawab tidak sesuai dengan kunci jawaban maka skor tidak bertambah dan akan muncul keterangan secara tertulis dan lisan yaitu “Pilihan Anda Salah” Gambar 4.50.



Gambar 4.50 keterangan salah pada level 3

Untuk script pencocokan jawaban masing masing kotak target memiliki script yang berbeda, untuk script pada target benar terdapat pada Gambar 4.51.

```

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

public class TrueDemo4 : MonoBehaviour
{
    public string InputJawaban;

    GameManager3 gm;
    public GameObject jawabanBenar;
    public GameObject jawabanSalah;
    public Text jawabanBenarText;
    public Text jawabanSalahText;
    public AudioSource benarSource;
    public AudioSource salahSource;

    // Start is called before the first frame update
    void Start()
    {
        gm =
        GameObject.Find("GameManager").GetComponent<GameManager3>;
    }

```

```

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

IEnumerator AddJawab(string jawaban_salah)
{
    WWWForm form = new WWWForm();
    form.AddField("addJawabanSalah", jawaban_salah);

    using (UnityWebRequest www =
UnityWebRequest.Post("http://lib.unika.ac.id/skripsi/ricky/dat
aInsert.php", form))
    {
        yield return www.SendWebRequest();

        if (www.isNetworkError || www.isHttpError)
        {
            Debug.Log(www.error);
        }
        else
        {
            Debug.Log("Form upload complete!");
        }
    }

    public float health = 10f;
    public void TakeDamage(float amount)
    {
        health -= amount;
        if (health <= 0f)
        {
            if (gm.currentQuestion.isTrue)
            {
                jawabanBenar.SetActive(true);
                jawabanBenarText.text =
gm.currentQuestion.answer;
                Skor.scoreValue += 5;
                benarSource.Play();

                StartCoroutine(gm.TransitionToNextQuestion());
            }
            else
            {
                StartCoroutine(AddJawab(gm.currentQuestion.fact));

                jawabanSalah.SetActive(true);
                jawabanSalahText.text =
gm.currentQuestion.answer;
                salahSource.Play();
            }
        }
    }
}

```

```

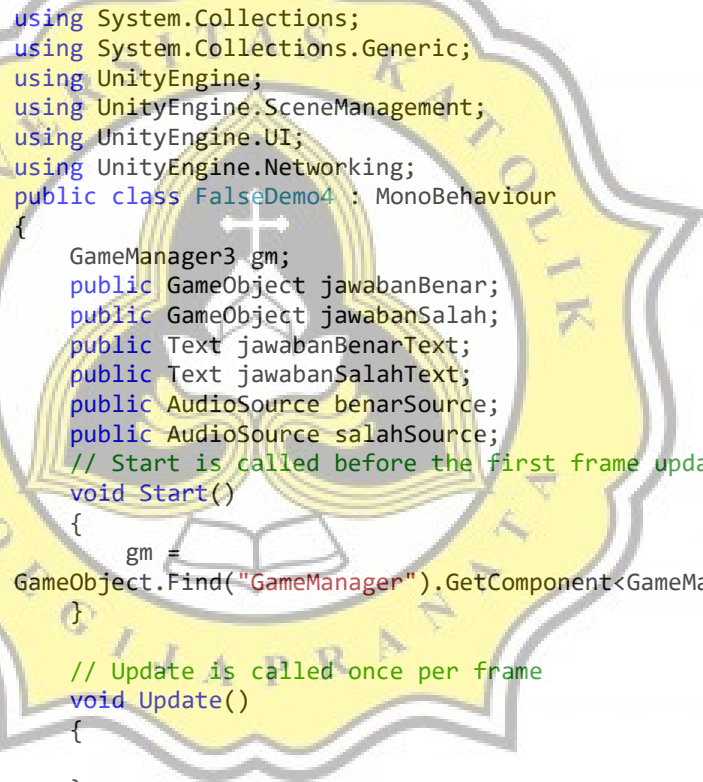
        StartCoroutine(gm.TransitionToNextQuestion());
    }
}
}

```

Gambar 4.51 Script pada kotak target benar

Untuk script pada kotak target salah terdapat pada Gambar

4.52.



```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.SceneManagement;
using UnityEngine.UI;
using UnityEngine.Networking;
public class FalseDemo4 : MonoBehaviour
{
    GameManager3 gm;
    public GameObject jawabanBenar;
    public GameObject jawabanSalah;
    public Text jawabanBenarText;
    public Text jawabanSalahText;
    public AudioSource benarSource;
    public AudioSource salahSource;
    // Start is called before the first frame update
    void Start()
    {
        gm =
        GameObject.Find("GameManager").GetComponent<GameManager3>();
    }

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

    }

    IEnumerator AddJawab(string jawaban_salah)
    {
        WWWForm form = new WWWForm();
        form.AddField("addJawabanSalah", jawaban_salah);

        using (UnityWebRequest www =
        UnityWebRequest.Post("http://lib.unika.ac.id/skripsi/ricky/dat
aInsert.php", form))
        {
            yield return www.SendWebRequest();

            if (www.isNetworkError || www.isHttpError)

```

```

        {
            Debug.Log(www.error);
        }
        else
        {
            Debug.Log("Form upload complete!");
        }
    }
}

public float health = 10f;

public void TakeDamage(float amount)
{
    health -= amount;
    if (health <= 0f)
    {
        if (!gm.currentQuestion.isTrue)
        {
            jawabanBenar.SetActive(true);
            jawabanBenarText.text =
gm.currentQuestion.answer;
            Skor.scoreValue += 5;
            benarSource.Play();

            StartCoroutine(gm.TransitionToNextQuestion());
        }
        else
        {
            StartCoroutine(AddJawab(gm.currentQuestion.fact));

            jawabanSalah.SetActive(true);
            jawabanSalahText.text =
gm.currentQuestion.answer;
            salahSource.Play();

            StartCoroutine(gm.TransitionToNextQuestion());
        }
    }
}
}
}

```

Gambar 4.52 Script pada target salah

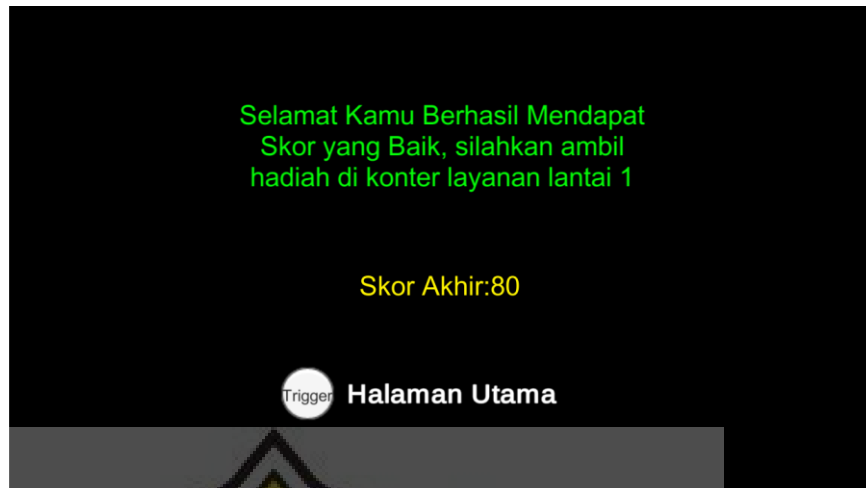
E. Akhir Game

Setelah pemain menyelesaikan ketiga level maka skor akhir akan ditampilkan. Apabila skor di bawah 75 maka akan muncul kalimat "Maaf Skor kamu kurang maksimal silahkan coba lagi" seperti pada gambar 4.53.



Gambar 4.53 Scene akhir Ketika skor <75

Namun apabila skor yang diperoleh 75 atau di atas 75 maka akan muncul kalimat "Selamat kamu berhasil mendapat skor yang baik, silahkan ambil hadiah di konter layanan lantai 1" seperti pada gambar 4.54.



Gambar 4.54 Scene akhir Ketika skor ≥ 75

4.5.5 Pengiriman soal yang dijawab salah oleh player ke database

Ketika player menjawab sebuah soal dan jawaban tidak sesuai dengan kunci jawabannya baik pada portal benar maupun portal salah, script juga akan mengirimkan soal tersebut ke database perpustakaan menggunakan script PHP 4.55. Data yang dikirimkan hanya jawaban yang dijawab player tidak sesuai dengan kunci jawaban karena data tersebut nantinya digunakan sebagai data untuk mengevaluasi materi Roadshow perpustakaan maupun update soal untuk permainan berikutnya.

```
<?php
include ('connect.php');

$jawabansalah = $_POST['addJawabanSalah'];

$sql = "insert into jawaban (jawaban_salah) values
('".$jawabansalah."')";
$result = mysqli_query($connect, $sql);
?>
```

Gambar 4.55 Script PHP Untuk mengirim soal ke database

dan script C# pada unity seperti pada Gambar 4.56.

```
IEnumerator AddJawab(string jawaban_salah)
```

```

{
    WWWForm form = new WWWForm();
    form.AddField("addJawabanSalah", jawaban_salah);

    using (UnityWebRequest www =
UnityWebRequest.Post("http://lib.unika.ac.id/skripsi/ricky/dataInsert.php", form ))
    {
        yield return www.SendWebRequest();

        if (www.isNetworkError || www.isHttpError)
        {
            Debug.Log(www.error);
        }
        else
        {
            Debug.Log("Form upload complete!");
        }
    }
}

```

Gambar 4.56 Script C# Untuk mengirimkan soal ke database

Gambar 4.57 adalah contoh data yang sudah masuk pada database perpustakaan.

id	jawaban_salah	waktu
3	Virtual Reality (VR) Library adalah aplikasi yang	2020-05-27 08:37:35
4	Dalam pembuatan sumber kutipan tipe American Psych.	2020-05-27 08:40:58
5	Abstrak merupakan teks lengkap pada publikasi ilmi.	2020-05-27 08:41:07
6	Katalog buku perpustakaan bisa diakses di lib.un...	2020-06-24 04:46:55
7	Penulisan kutipan dalam publikasi ilmiah harus dis...	2020-06-24 04:47:22
9	Skripsi/ TA/ Thesis/Laporan Penelitian bisa diakse...	2020-06-24 13:01:00
10	Augmented Reality (AR) Library adalah aplikasi car...	2020-06-24 13:02:45
11	Artikel jurnal ilmiah adalah publikasi yang dituli...	2020-06-24 13:02:59
12	Abstrak merupakan teks lengkap pada publikasi ilmi...	2020-06-24 14:14:21
13	e-Prosiding terbitan Unika Soegijapranata bisa dia...	2020-06-24 14:14:48
14	Katalog buku perpustakaan bisa diakses di : lib un...	2020-06-24 14:45:15
15	Untuk menemukan artikel jurnal ilmiah publikasi li...	2020-06-24 14:45:36
16	Untuk menemukan artikel jurnal ilmiah publikasi li...	2020-06-24 16:28:00
17	Skripsi/ TA/ Thesis/Laporan Penelitian bisa diakse...	2020-06-24 16:28:21
18	Artikel jurnal ilmiah adalah publikasi yang dituli...	2020-06-24 16:28:33
19	Tulisan yang berisi ulasan suatu buku disebut deng...	2020-06-25 11:04:57
20	Dalam pembuatan sumber kutipan tipe American Psych...	2020-07-01 22:22:50

Gambar 4.57 tampilan data pada database

Gambar 4.58 adalah tampilan Website sederhana yang jumlah soal yang sama sudah dihitung secara otomatis sehingga perpustakaan tidak perlu menghitung soal soal yang sering dijawab salah oleh para player secara manual. Website bisa diakses pada

<http://lib.unika.ac.id/skripsi/ricky/tampilan2.php>

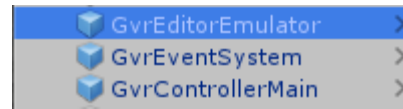
Tabel Hitung Jawaban Salah

Jawaban Salah	Hitung
Abstrak merupakan teks lengkap pada publikasi ilmiah	4
Artikel jurnal ilmiah adalah publikasi yang ditulis oleh seorang jurnalis	5
Augmented Reality (AR) Library adalah aplikasi cara akses game di perpustakaan	3
Dalam pembuatan sumber kutipan tipe American Psychological Association (APA) dicantumkan nama, tahun terbit buku, dan tanggal	4
e-Prosiding terbitan Unika Soegijapranata bisa diakses di: proceedings.unika.ac.id	2
Ensiklopedia dan kamus merupakan koleksi referensi perpustakaan	2
Katalog buku perpustakaan bisa diakses di : lib.unika.ac.id	2
Katalog buku perpustakaan menyediakan nomor panggil untuk artikel jurnal	1
Koleksi e-book perpustakaan Unika Soegijapranata bisa diakses melalui google play: e-library Universitas Katolik Soegijapranata	3
Layanan rujukan digital membantu pemustaka mendapatkan koleksi rujukan tercetak	3
Penulisan kutipan dalam publikasi ilmiah harus disertai sumber kutipan, catatan kaki (footnote) dan daftar pustaka	4
Plagiarisme adalah copy paste kalimat dari suatu publikasi tidak disertai sumber kutipan	1
Skripsi/ TA/ Thesis/Laporan Penelitian bisa diakses di: repository.unika.ac.id	4

Gambar 4.58 Tampilan sederhana pada website

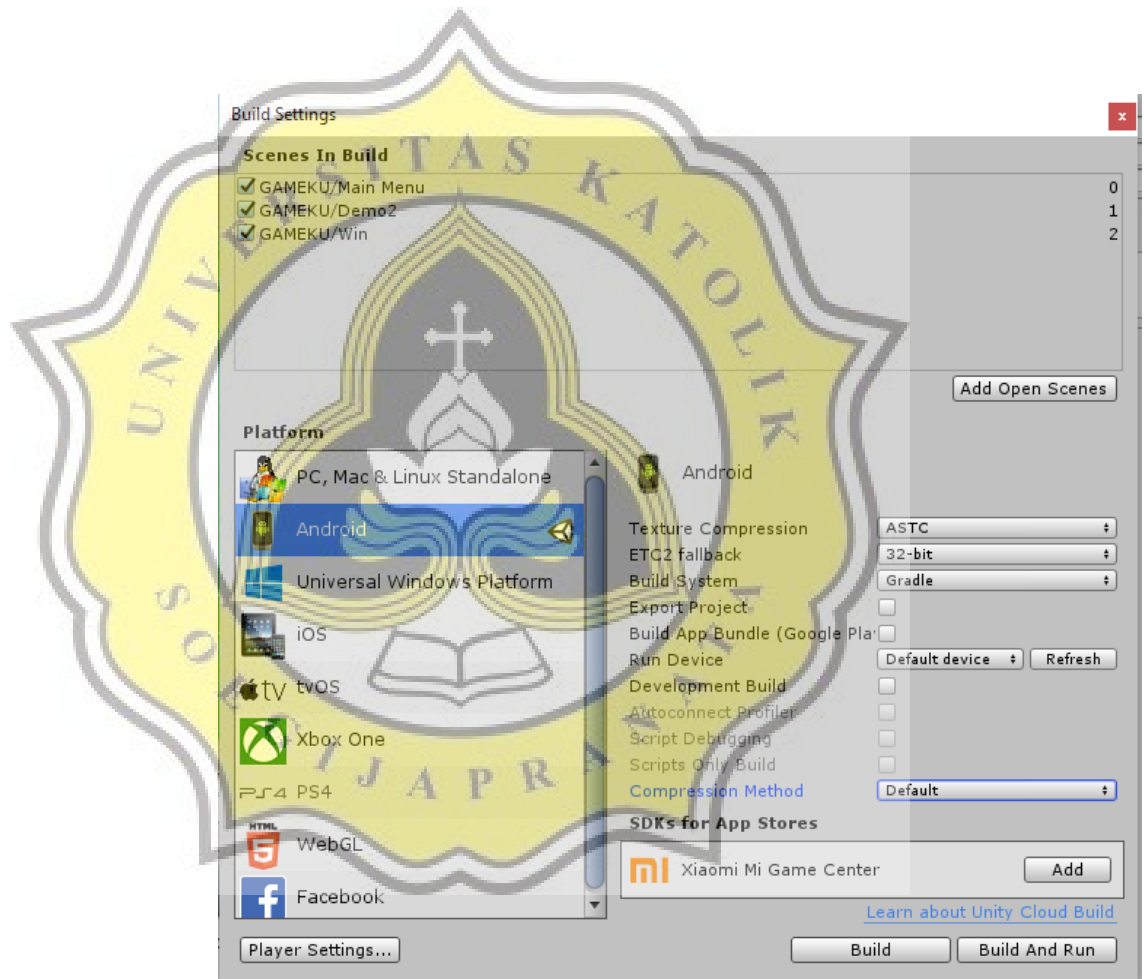
4.5.6 Build game

Untuk Build Game pada Oculus Go membutuhkan beberapa konfigurasi khusus karena sistem operasi yang berbeda dengan Android. Untuk SDK Menggunakan Google VR SDK seperti pada gambar 4.59.



Gambar 4.59 Google VR SDK

Pada build setting menggunakan platform Android dengan mengganti Texture Compression menjadi ASTC seperti pada gambar 4.60.



Gambar 4.60 Build Setting

Pada player settings gunakan "Auto Graphics API", "Multithreaded Rendering", "Static Batching", "Dynamic Batching", dan "Virtual Reality Supported" dengan SDK Oculus seperti pada gambar 4.61.



Gambar 4.61 Player Setting

Pada Quality setting dibuat seperti pada gambar 4.62.

Setting	Value
Pixel Light Count	1
Texture Quality	Full Res
Anisotropic Textures	Per Texture
Anti Aliasing	OVRManager -> Use Recommended MSAA Level
Soft Particles	Remove selection
Realtime Reflections Probes	Select
Billboards Face Camera	Select

Gambar 4.62 Quality Setting

4.6 Memasukan Game ke Oculus Go

Game yang masuk kedalam Oculus Go tidak bisa secara langsung dengan mengcopy file seperti pada android, namun memerlukan Langkah Langkah khusus melalui command prompt dan mengcopy file khusus pada komputer seperti Langkah Langkah dibawah ini

- Setting ADB Driver dengan cara mengcopy “Platform Tools” ke
C:/Users/*nama computer*
- Hubungkan Oculus Go ke PC menggunakan kabel
- Aktifkan cmd atau command prompt
- Ketik “adb”
- Kemudian ketik “adb devices” untuk memastikan device yang terhubung
- Ketik “adb install namaApk.apk” untuk menginstall aplikasi pada Oculus Go

4.7 Rangkuman Hasil Wawancara

No	Staff 1	Staff 2	Staff 3	Staff 4	Kepala Perpustakaan
1	Game menarik, karena	Game menarik, karena	Menarik, karena game ini	Sangat menarik, karena ini adalah	Game Menarik,

	menggunakan teknologi virtual reality yang memberi efek suasana baru berbeda dengan permainan biasa	teknologi yang digunakan belum pernah saya coba	sebagai salah satu media pembelajaran yang menggunakan teknologi yang dapat digunakan semua orang	cara baru memperkenalkan literasi	karena bisa menambah pengetahuan mengenai perpustakaan
2	Alat yang digunakan menarik serta memenuhi syarat, selain itu memberikan kesan immersive	Alatnya cukup menarik dan memenuhi syarat selain itu penampilan keren dan nyaman digunakan di kepala	Selain menarik alat ini memenuhi syarat karena lucu dan alatnya modern	Alat yang digunakan menarik, karena alat ini adalah alat baru yang belum banyak orang mencoba	Menarik dan memenuhi syarat
3	Pernah	Belum pernah	Belum pernah	Belum pernah	Belum pernah
4	Pengoperasian mudah dimengerti	Mudah, serta instruksi jelas	Pengoperasian mudah	Sangat mudah	Mudah

5	Panduan yang ada sudah jelas	Instruksi jelas dan tidak sulit untuk orang awam	Panduan sangat jelas	Panduan jelas dan simpel	Panduan sangat jelas
6	Enjoy	Enjoy	Enjoy	Sangat Enjoy	Sangat enjoy
7	Sangat membantu, karena game menjelaskan kegiatan perpustakaan	Sangat membantu, karena game ini membuat hal yang rumit menjadi mudah	Membantu, karena sebagai orang awam bisa belajar apa itu perpustakaan	Membantu, karena game mengenalkan apa yang ada di perpustakaan	Sangat membantu
8	Bisa, karena game ini lebih interaktif daripada sekedar membaca buku manual atau pengumuman poster	Bisa, karena game memiliki interaksi ketika memberi informasi sehingga tidak membosankan	Bisa, karena game ini memiliki keterangan jawaban lebih lanjut sehingga ketika salah ada penjelasannya	Bisa, karena mahasiswa gen Z membutuhkan model pembelajaran yang interaktif	Game bisa membantu mahasiswa untuk memahami materi koperasi informasi, menggunakan koleksi perpustakaan, dan lain lain.

9	Menyelesaikan, apabila dimainkan	Cukup menyelesaikan, karena game cocok untuk kaum milenial yang melek teknologi	Menyelesaikan, karena game ini bisa membuka sedikit wawasan mengenai perpustakaan	Menyelesaikan, karena dengan game ini mahasiswa bisa bermain sambil belajar	Cukup menyelesaikan
10	Sudah cukup mewakili namun diharapkan penambahan bank soal.	Cukup mewakili karena game mengikuti visi misi perpustakaan	Cukup mewakili	Sudah cukup mewakili	Cukup mewakili
11	Sangat disarankan, supaya mengikuti perkembangan teknologi	Sangat disarankan, karena game seru mahasiswa bisa mengerti kegiatan perpustakaan	Sangat disarankan, karena ini adalah satu permainan baru yang ada di perpustakaan unika serta game ini	Sangat disarankan, karena selain mengedukasi game ini mengenalkan perpustakaan secara fun	Ya, supaya pengetahuan mahasiswa mengenai literasi informasi meningkat

			mengedukasi mahasiswa		
12	-	-	-	-	Ya, akan digunakan dan dikembangkan dari sisi aplikasi dan konten

