BAB IV
PEMBAHASAN DAN HASIL

4.1 Implementasi Aplikasi

Implementasi aplikasi *Game* ini adalah proses pembuatan aplikasi yang sesuai dengan rancangan. Setelah pembuatan perancangan maka dapat dipresentasikan hasil dan perancangan yang telah dibuat.

Batasan Implementasi Aplikasi

Dalam implementasinya, aplikasi *game* ini memiliki beberapa keterbatasan yaitu:

a. *Game* ini hanya dapat dimainkan dalam mode single *player* (satu pemain).
b. Aplikasi ini bersifat *offline* atau tidak terhubung ke internet.
c. Aplikasi ini hanya ditujukan untuk sistem operasi *Android*.

4.2 Implementasi Proses Pembuatan Aplikasi

Implementasi proses pembuatan aplikasi ini menunjukkan teknik dan *tools* yang digunakan untuk membuat "*Game Edukasi Tertib Lalu Lintas Roda Empat*". Langkah-langkahnya adalah sebagai berikut:

4.2.1 Implementasi Pemodelan 3D

Model 3D

Dari sebagian banyak objek 3D yang akan digunakan dalam aplikasi *game* ini, ada beberapa objek 3D yang mempunyai fungsi penting dalam aplikasi *game* ini. Seperti model mobil, jalan raya, rambu lalu lintas, dan lain-lain. Setelah ditemukan objek-objek yang akan digunakan, tahap selanjutnya adalah membuat sampel model mobil. Pembuatan model mobil ini dengan menggunakan *software Sketchup*. Pembuatan model mobil dilakukan dengan bantuan model dari mobil Toyota Avanza, sehingga memudahkan dalam membentuk model mobil.

***Exporting* Model 3D**

Tahap ini adalah tahap untuk merubah file model dari *Sketchup* agar dapat digunakan kedalam *Unity*. File model *Skethup* yang memiliki format *.skp* diekspor ke dalam format *.fbx* agar dapat terbaca sebagai aset di *Unity*.

4.2.2 Implementasi *Unity*

Tahap Implementasi *software Unity* ini merupakan tahap yang paling utama. Seluruh konten *game* berupa model 3D dan script disatukan dan dibuat dalam *software Unity* ini kemudian dilakukan penggabungan seluruh *asset* hingga membentuk satu kesatuan utuh yang siap dijalankan berbentuk aplikasi *stand alone* atau aplikasi independen.

***Importing* Objek 2D**

Objek 2D yang telah diexport dalam bentuk .png kemudian diimpor ke dalam *Unity* dalam folder *asset* sehingga akan terbaca sebagai objek.

Scene

Tahap berikutnya adalah pemberian scene. Scene disini berfungsi sebagai tempat untuk memisahkan *event* (kejadian). Dalam game ini terdapat beberapa scene antara lain, scene menu awal, scene dalam game.

Build

Pada tahap *build* ini, scene yang sudah dibuat semua akan diurutkan sesuai eksekusinya dan akan diekspor ke dalam *platform* yang akan digunakan. Pada *game* "*Game Edukasi Tertib Lalu Lintas Roda Empat*" ini menggunakan *platform Android*.

4.2.3 Implementasi Antarmuka

Implementasi antarmuka dari aplikasi "*Game Edukasi Tertib Lalu Lintas Roda Empat*" ini terdiri dari beberapa halaman yang memiliki proses berdasarkan fungsi dari halaman tersebut. Fungsi-fungsi tersebut merupakan penjabaran dari hasil analisis kebutuhan sistem. Implementasi antarmuka dari aplikasi "*Game Edukasi Tertib Lalu Lintas Roda Empat*" adalah sebagai berikut.

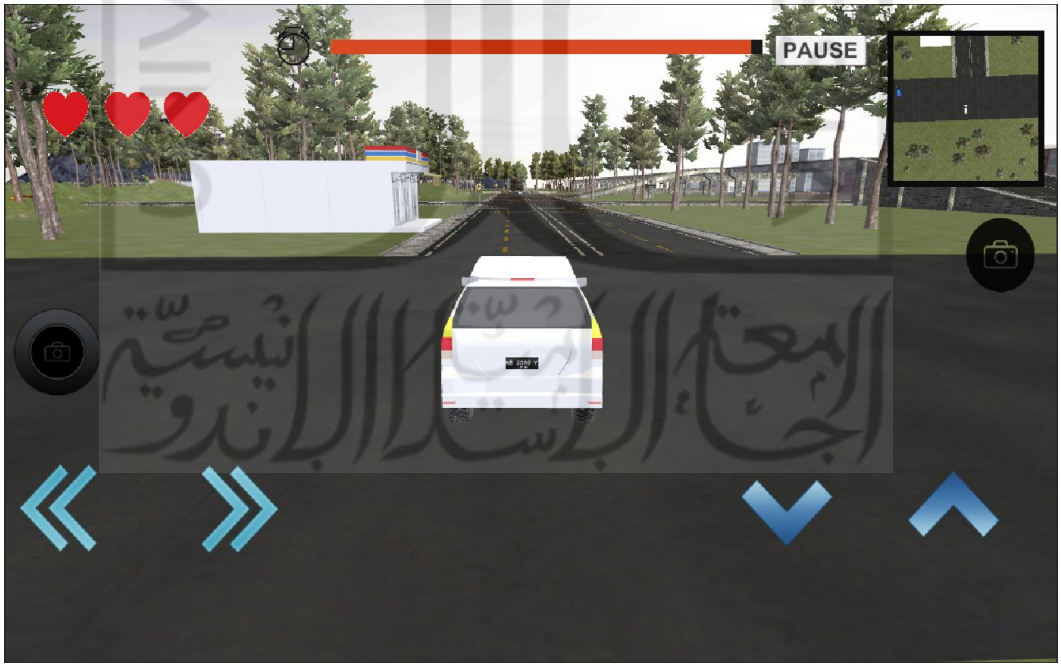
Implementasi Halaman Start Menu

Halaman start menu merupakan halaman awal yang mempunyai tiga tombol menu yaitu *Start Game*, dan Keluar. Halaman ini diimplementasikan seperti pada Gambar 4.1.

Gambar 4.1 Halaman Start Menu

Implementasi Halaman Dalam *Game*

Implementasi Halaman dalam *Game* adalah halaman untuk memainkan permainan. Halaman ini diimplementasikan seperti Gambar 4.2.



Gambar 4.2 Halaman dalam *Game*

Implementasi Halaman *Finish*

Implementasi Halaman *Finish* adalah halaman yang akan tampil jika pemain berhasil menyelesaikan suatu level. Halaman ini diimplementasikan seperti Gambar 4.3.

Gambar 4.3 Halaman *Finish*

Implementasi Halaman *Game Over*

Implementasi Halaman *Game Over* adalah halaman yang akan tampil jika pemain gagal menyelesaikan suatu level. Halaman ini diimplementasikan seperti Gambar 4.4.

Gambar 4.4 Halaman *Game Over*

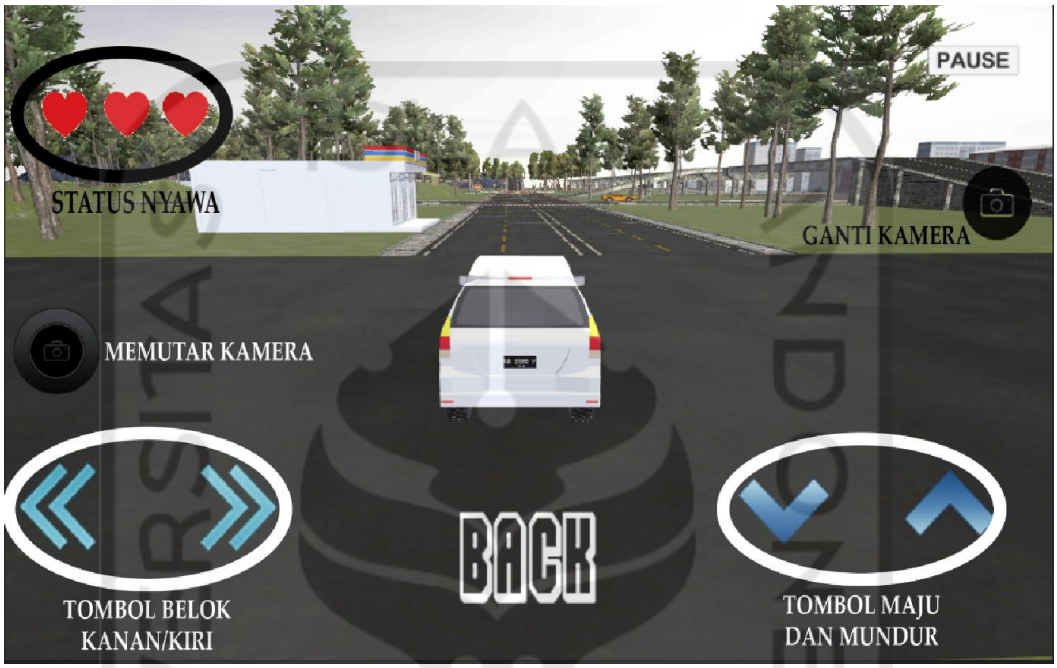
Implementasi Halaman *Pause*

Implementasi Halaman *Pause* adalah halaman yang akan tampil jika pemain *pause* dalam *game*. Halaman ini diimplementasikan seperti Gambar 4.5.

Gambar 4.5 Halaman *Pause*

Implementasi Halaman *Control Mapping*

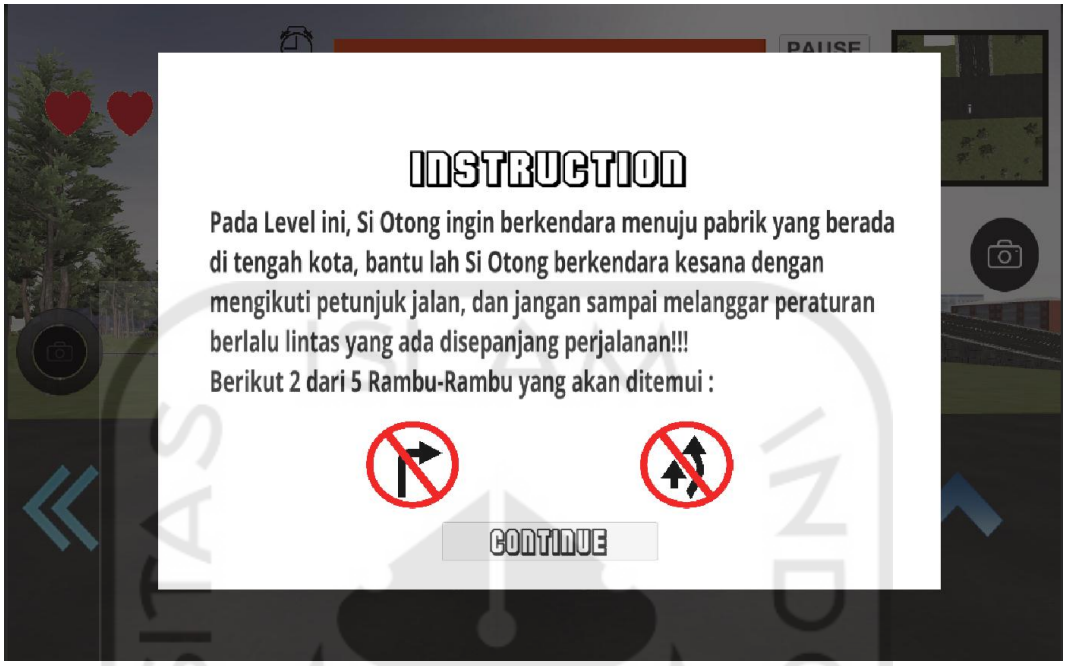
Implementasi Halaman *Pause* adalah halaman yang menampilkan penjelasan tombol-tombol yang ada dalam *game*. Halaman ini diimplementasikan seperti Gambar 4.6.



Gambar 4.6 Halaman *Control Mapping*

Implementasi Instruksi Awal

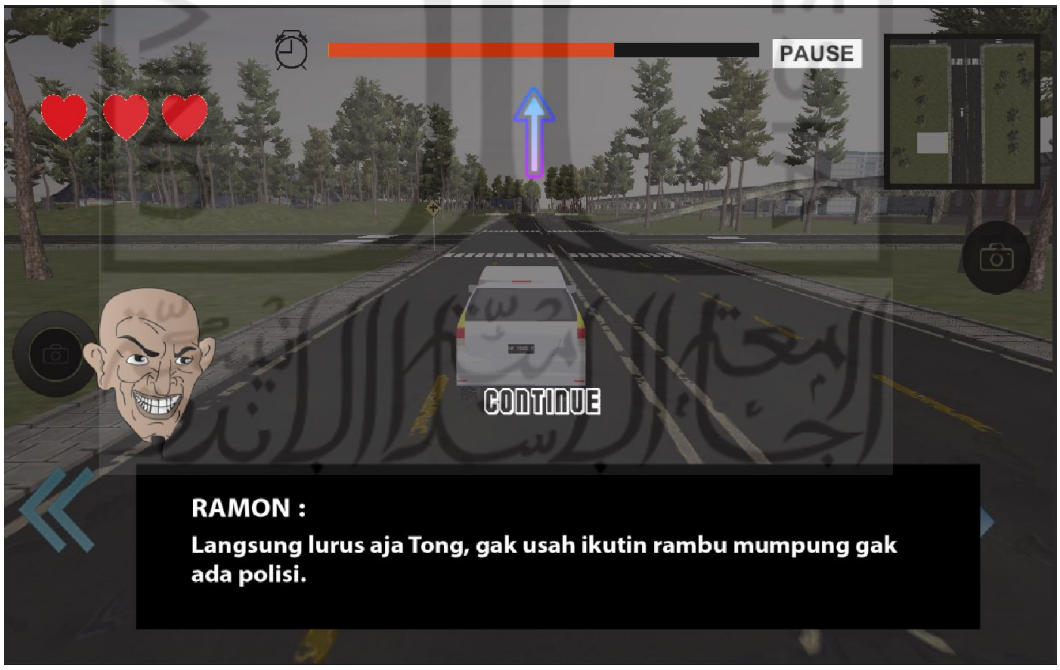
Instruksi awal ini digunakan untuk memberitahu pemain tugas yang harus dimainkan pada level tersebut. Tampilan ini diimplementasikan seperti Gambar 4.7.



Gambar 4.7 Tampilan Instruksi Awal

Implementasi Tampilan *Pop Up* Ramon

Pop up Ramon ini akan muncul untuk mengajak Otong agar melanggar peraturan lalu lintas. Tampilan ini diimplementasikan seperti Gambar 4.8.



Gambar 4.8 Tampilan *Pop Up* Ramon

Implementasi Halaman *Times Up*

Tampilan *Times Up* akan muncul jika pemain telah kehabisan waktu ketika berada di dalam game. Tampilan ini diimplementasikan seperti Gambar 4.9.

Gambar 4.9 Halaman *Times Up*

Kode Dalam *Game*

Berikut ini kode program yang digunakan dalam *game* :

a. Kode *Level Loader*

Kode ini digunakan untuk berpindah ke *scene* lain dengan *loading screen*. Kode ini ditunjukkan seperti pada Gambar 4.10.

```
1   using System.Collections;
2   using UnityEngine;
3   using UnityEngine.SceneManagement;
4   using UnityEngine.UI;
5
6   public class LevelLoader : MonoBehaviour {
7
8       public GameObject loadingScreen;
9       public Slider slider;
10      public Text progressText;
11
12      public void LoadLevel (int sceneIndex)
13      {
14          StartCoroutine(LoadAsynchronously(sceneIndex));
15      }
16
17      IEnumerator LoadAsynchronously (int sceneIndex)
18      {
19          AsyncOperation operation = SceneManager.LoadSceneAsync(sceneIndex);
20
21          loadingScreen.SetActive(true);
22
23          while (!operation.isDone)
24          {
25              float progress = Mathf.Clamp01(operation.progress / .9f);
26              slider.value = progress;
27              progressText.text = progress * 100f + "%";
28              yield return null;
29          }
30      }
31
32  }
33
```

Gambar 4.10 Kode *Level Loader*

b. Kode *Game Control*

Kode ini digunakan untuk memberi aturan dalam *game*, aturan tersebut berkaitan dengan nyawa pemain dalam *game*. Kode ini ditunjukkan seperti pada Gambar 4.11.

```
11      void Start () {
12          Health = 3;
13          Nyawa.gameObject.SetActive(true);
14          Nyawa1.gameObject.SetActive(true);
15          Nyawa2.gameObject.SetActive(true);
16          gameOver.gameObject.SetActive(false);
17      }
18
19      // Update is called once per frame
20      void Update () {
21          if (Health > 3)
22              Health = 3;
23
24          switch (Health)
25          {
26              case 3:
27                  Nyawa.gameObject.SetActive(true);
28                  Nyawa1.gameObject.SetActive(true);
29                  Nyawa2.gameObject.SetActive(true);
30
31                  break;
32              case 2:
33                  Nyawa.gameObject.SetActive(true);
34                  Nyawa1.gameObject.SetActive(true);
35                  Nyawa2.gameObject.SetActive(false);
36                  break;
37              case 1:
38                  Nyawa.gameObject.SetActive(true);
39                  Nyawa1.gameObject.SetActive(false);
40                  Nyawa2.gameObject.SetActive(false);
41                  break;
42              case 0:
43                  Nyawa.gameObject.SetActive(false);
44                  Nyawa1.gameObject.SetActive(false);
45                  Nyawa2.gameObject.SetActive(false);
46                  gameOver.gameObject.SetActive(true);
```

Gambar 4.11 Kode *Game Control*

c. Kode Nyawa

Kode ini digunakan untuk melakukan *trigger* kepada *player* dan mengurangi nyawa *player* jika menyentuh *trigger* dalam kode ini. Kode ini ditunjukkan seperti pada Gambar 4.12.

```
5   public class Nyawa : MonoBehaviour {
6
7
8       void OnTriggerEnter(Collider other)
9       {
10          if (other.CompareTag("Obstacle"))
11          {
12              GameControlScript.Health -= 1;
13          }
14      }
15      private void OnTriggerExit(Collider other)
16      {
17          if (other.CompareTag("Obstacle"))
18          {
19              GameControlScript.Health -= 0;
20          }
21      }
22
23  }
24
```

Gambar 4.12 Kode Nyawa

d. Kode *Trigger* Rambu

Kode ini digunakan untuk melakukan *trigger* berupa menampilkan informasi tentang rambu yang akan dilewati jika pemain menyentuh *trigger* yang ada dalam kode ini. Kode ini ditunjukkan seperti pada Gambar 4.13.

```
6   public class UIAppear : MonoBehaviour
7   {
8       [SerializeField] private Image customImage;
9
10      void OnTriggerEnter(Collider other)
11      {
12          if (other.CompareTag("Player"))
13          {
14              customImage.enabled = true;
15          }
16      }
17      private void OnTriggerExit(Collider other)
18      {
19          if (other.CompareTag("Player"))
20          {
21              customImage.enabled = false;
22          }
23      }
24  }
25
```

Gambar 4.13 Kode *Trigger* Rambu

e. Kode *Pause*

Kode ini digunakan untuk memanggil halaman *pause* dan fungsi-fungsi dalam halaman *pause*. Kode ini ditunjukkan seperti pada Gambar 4.14

```
4
5   public class PauseGame : MonoBehaviour {
6       public Transform canvas;
7       public Transform Player;
8
9
10
11
12      void Start()
13      {
14          canvas.gameObject.SetActive(false);
15
16      }
17      public void pauseButton() {
18
19
20          Pause();
21
22      }
23      public void Pause()
24      {
25          if (canvas.gameObject.activeInHierarchy == false)
26          {
27              canvas.gameObject.SetActive(true);
28              Time.timeScale = 0;
29
30              AudioListener.volume = 0;
31
32          }
33          else
34          {
35              canvas.gameObject.SetActive(false);
36              Time.timeScale = 1;
37
38              AudioListener.volume = 1;
39          }
40      }
41  }
```

Gambar 4.14 Kode *Pause*

f. Kode Tampilan Instruksi Awal

Kode ini digunakan untuk menampilkan instruksi yang harus dilakaukan pemain pada level tersebut. Kode ini ditunjukkan seperti pada Gambar 4.15.

```
1   using UnityEngine;
2   using System.Collections;
3   using UnityStandardAssets.Characters.FirstPerson;
4
5   public class InstruksiAwal : MonoBehaviour
6   {
7       public Transform canvas;
8
9       void Start()
10      {
11          canvas.gameObject.SetActive(true);
12          Time.timeScale = 0;
13
14          AudioListener.volume = 0;
15      }
16
17      public void lanjut()
18      {
19          canvas.gameObject.SetActive(false);
20          Time.timeScale = 1;
21
22          AudioListener.volume = 1;
23      }
24  }
25
```

Gambar 4.15 Kode Tampilan Instruksi Awal

g. Kode Tampilan *Minimap*

Kode ini digunakan untuk menampilkan minimap yang akan mengikuti pemain pada level tersebut. Kode ini ditunjukkan seperti pada Gambar 4.16.

```
1   using System.Collections;
2   using System.Collections.Generic;
3   using UnityEngine;
4
5   public class MiniMapScript : MonoBehaviour {
6
7       public Transform player;
8
9       private void LateUpdate()
10      {
11          Vector3 newPosition = player.position;
12          newPosition.y = transform.position.y;
13          transform.position = newPosition;
14
15          transform.rotation = Quaternion.Euler(90f, player.eulerAngles.y, 0f);
16      }
17  }
18
```

Gambar 4.16 Kode Tampilan *Minimap*

h. Kode *Timer*

Kode ini digunakan untuk membuat *timer* yang digunakan untuk memberi batasan waktu pemain untuk melewati level tersebut. Kode ini ditunjukkan seperti pada Gambar 4.17.

```
6   public class Timer_script : MonoBehaviour {
7
8       Image timerBar;
9       public float maxTime = 10f;
10      float timeLeft;
11      public GameObject timesUpText;
12
13      void Start () {
14          timesUpText.SetActive(false);
15          timerBar = GetComponent<Image>();
16          timeLeft = maxTime;
17      }
18
19      void Update () {
20          if(timeLeft > 0)
21          {
22              timeLeft -= Time.deltaTime;
23              timerBar.fillAmount = timeLeft / maxTime;
24          }
25          else
26          {
27              timesUpText.SetActive(true);
28              Time.timeScale = 0;
29              AudioListener.volume = 0;
30          }
31      }
```

Gambar 4.17 Kode *Timer*

i. Kode Dialog Ramon

Kode ini digunakan untuk menampilkan dialog Ramon yang digunakan untuk membujuk pemain untuk melanggar lalu lintas. Kode ini ditunjukkan seperti pada Gambar 4.18.

```
1   using System.Collections;
2   using System.Collections.Generic;
3   using UnityEngine;
4   using UnityEngine.UI;
5
6   public class OrangJahadAppear : MonoBehaviour
7   {
8       [SerializeField] private Image customImage;
9       public Button lanjut;
10      void Start()
11      {
12          lanjut.gameObject.SetActive(false);
13      }
14
15      void OnTriggerEnter(Collider other)
16      {
17          if (other.CompareTag("Player"))
18          {
19              customImage.enabled = true;
20              Time.timeScale = 0;
21              lanjut.gameObject.SetActive(true);
22              AudioListener.volume = 0;
23          }
24      }
25      private void OnTriggerExit(Collider other)
26      {
27          if (other.CompareTag("Player"))
28          {
29              customImage.enabled = false;
30              Time.timeScale = 1;
31              lanjut.gameObject.SetActive(false);
32              AudioListener.volume = 1;
33          }
34      }
35  }
```

Gambar 4.18 Kode Dialog Ramon

4.3 Analisis Kinerja *Game*

Analisis Kinerja *game* ini digunakan untuk menguji sejauh mana *game* ini dapat berinteraksi dengan pengguna setelah diberi tindakan oleh pengguna *game* ke dalam sistem.

Seusai dengan rancangan dalam urutan pembuatan aplikasi "*Game Edukasi Tertib Lalu Lintas Roda Empat*", analisis yang dilakukan terhadap teori pembuatan aplikasi yang akan dibangun adalah:

a. *Game* Simulasi

Program menggunakan tampilan 3D yang menggambarkan suatu keadaan mengemudi menggunakan kendaraan roda empat. Pemain diberikan tantangan mengendarai mobil sesuai peraturan yang ada untuk menuju ke lokasi yang telah ditentukan didalam game.

b. Instruksi untuk Pengguna *Game*

Untuk memudahkan pengguna dalam memainkan *game* ini, dihalaman sebelum masuk kedalam *game* akan ada petunjuk permainan yang memberikan informasi bagaimana cara memainkan *game*.

c. Interaksi Objek

Objek yang dikendarakan oleh pengguna dapat berinteraksi dengan objek-objek pada *game*. Seperti interaksi menabrak pembatas, pohon, bangunan dan lain-lain.

4.4 Hasil Pengujian

a. Pengujian Kelebihan dan Kekurangan

Hasil dari pengujian sistem dibuat setelah proses pengujian sistem dilakukan. Hasil dari pengujian sistem didapat melalui penyebaran kuisioner kepada responden. Berikut ini adalah data responden yang telah mengisi kuisioner "*Game Edukasi Tertib Lalu Lintas Roda Empat*" seperti pada Tabel 4.1.

Tabel 4.1 Data Responden

No.	Jenis Kelamin	Usia	Pekerjaan
1.	Laki-Laki	15	Siswa
2.	Laki-Laki	16	Siswa
3.	Perempuan	15	Siswa
4.	Laki-Laki	16	Siswa
5.	Laki-Laki	17	Siswa
6.	Laki-Laki	19	Mahasiswa
7.	Perempuan	20	Mahasiswi
8.	Laki-Laki	19	Mahasiswa
9.	Perempuan	21	Mahasiswi
10.	Laki-Laki	23	Mahasiswa
11.	Laki-Laki	22	Mahasiswa
12.	Laki-Laki	23	Mahasiswa
13.	Laki-Laki	24	Mahasiswa
14.	Laki-Laki	23	Mahasiswa
15.	Perempuan	26	Karyawati
16.	Laki-Laki	30	Karyawan
17.	Perempuan	25	Karyawati
18.	Laki-Laki	37	Karyawan
19	Perempuan	33	Karyawati
20.	Laki-Laki	27	Karyawan

Dari kuisioner tersebut diberikan beberapa nilai untuk memudahkan perhitungan hasil analisisnya, yaitu:

a. Nilai 1 untuk jawaban *sangat kurang.*
b. Nilai 2 untuk jawaban *kurang.*
c. Nilai 3 untuk jawaban *cukup.*
d. Nilai 4 untuk jawaban *baik.*
e. Nilai 5 untuk jawaban *sangat baik.*

Berikut adalah daftar pertanyaan yang diajukan beserta jumlah kuisioner yang memberikan jawaban masing-masing pertanyaan dan rata-rata nilainya seperti pada Tabel 4.2.

Tabel 4.2 Hasil Kuisioner Kinerja Game

NO	Pertanyaan	1	2	3	4	5	Rata-Rata
1.	Apakah *game* ini menambah pengetahuan anda mengenai peraturan berlalu lintas?			6	10	4	3,9
2.	Apakah desain antarmuka *game* ini menarik?			3	12	5	4,1
3.	Bagaimana informasi petunjuk permainan pada *game* ini?			5	8	7	4,1
4.	Bagaimana kemudahan penggunaan aplikasi *Game* ini?			3	14	3	4
5.	Bagaimana kesesuaian rintangan pada setiap ujian dalam *game*?			4	6	10	4,3
6.	Bagaimana tampilan visual pada permainan ini?			1	4	15	4,7
7.	Apakah fitur-fitur pada *game* ini sudah cukup baik?			6	10	4	3,9

Dari kuisioner yang telah diberikan kepada responden, maka dapat diambil hasil analisisnya. Hasil analisis adalah sebagai berikut:

- Menambah pengetahuan pengguna

Data yang diperoleh dari responden mengenai tingkat menambah pengetahuan responden terhadap aplikasi, enam responden menjawab cukup, sepuluh responden menjawab baik, dan empat responden menjawab sangat baik. Dari hasil didapat bahwa aplikasi *game* ini baik berguna menambah pengetahuan karena rata-rata penilaiannya mencapai 3,9.

- Tampilan desain antarmuka

Data yang diperoleh dari responden mengenai tampilan desain antarmuka pada aplikasi, terdapat tiga responden menjawab cukup, dua belas responden menjawab baik, dan lima responden menjawab sangat baik. Dari hasil didapat bahwa tampilan desain antarmuka *game* baik karena rata-rata penilaiannya mencapai 4,1.

- Tingkat kepuasan terhadap petunjuk permainan

Data yang diperoleh dari responden mengenai tingkat tengkat kepuasan terhadap petunjuk permaian, terdapat lima responden menjawab cukup, delapan responden menjawab baik, dan tujuh responden menjawab sangat baik. Dari hasil didapat bahwa aplikasi *game* ini memiliki petunjuk permainan baik karena rata-rata penilaiannya mencapai 4,1.

- Kemudahan penggunaan aplikasi

Data yang diperoleh dari responden mengenai kemudahan penggunaan aplikasi, terdapat tiga responden menjawab cukup, empat belas responden menjawab baik, dan tiga responden menjawab sangat baik. Dari hasil didapat bahwa kemudahan penggunaan baik karena rata-rata penilaiannya mencapai 4.

- Kesesuaian rintangan di setiap level

Data yang diperoleh dari responden mengenai kesesuaian rintangan disetiap level, terdapat empat responden menjawab cukup, enam responden menjawab baik, dan sepuluh responden menjawab sangat baik. Dari hasil didapat bahwa kesesuaian rintangan disetiap level baik karena karena rata-rata penilaiannya mencapai 4,3.

- Tingkat kepuasan terhadap tampilan visual

Data yang diperoleh dari responden tingkat kepuasan terhadap tampilan visual, terdapat satu responden menjawab cukup, empat responden menjawab baik, dan lima belas responden menjawab baik. Dari hasil didapat bahwa tingkat kepuasan terhadap tampilan visual *game* sangat baik karena karena rata-rata penilaiannya mencapai 4,7.

- Kepuasan terhadap fitur yang ada pada *game*

Data yang diperoleh dari responden mengenai kepuasan terhadap fitur yang ada pada *game*, terdapat enam responden menjawab cukup, sepuluh responden menjawab baik, dan empat responden menjawab sangat baik. Dari hasil didapat bahwa kepuasan terhadap fitur yang ada pada *game* baik karena karena rata-rata penilaiannya mencapai 3,9.

b. Pengujian Pre Test dan Post Test

Hasil dari pengujian sebelum memainkan "*Game Edukasi Tertib Lalu Lintas Roda Empat*" ditampilkan pada tabel 4.3. Hasil dari pengujian setelah memainkan "*Game Edukasi Tertib Lalu Lintas Roda Empat*" ditampilkan pada tabel 4.4.

Tabel 4.3 Hasil Pre Test

NO	Pertanyaan	Benar	Salah
1.	Apakah arti dari rambu ini?	6	14
2.	Apakah arti dari rambu ini?	3	17
3.	Apakah arti dari rambu ini?	9	11
4.	Apakah arti dari rambu ini?	13	7
5.	Warna rambu apakah yang memiliki arti peringatan?	17	3

Tabel 4.4 Hasil Post Test

NO	Pertanyaan	Benar	Salah
1.	Apakah arti dari rambu ini?	18	2
2.	Apakah arti dari rambu ini?	20	0
3.	Apakah arti dari rambu ini?	20	0
4.	Apakah arti dari rambu ini?	20	0
5.	Warna rambu apakah yang memiliki arti peringatan?	20	0

4.5 Analisis Kelebihan dan Kekurangan Sistem

Dalam membangun suatu aplikasi tentunya terdapat kelebihan dan kekurangan seperti yang terdapat pada aplikasi "*Game Edukasi Tertib Lalu Lintas Roda Empat*". Kelebihan dan kekurangan ini merupakan representasi dari proses yang berjalan dalam suatu aplikasi. Setelah dilakukan pengujian ke beberapa

pengguna yang senang bermain *game* ber*genre* simulasi maka terdapat kelebihan dan kekurangannya sebagai berikut:

a. Kelebihan

Kelebihan dari "*Game Edukasi Tertib Lalu Lintas Roda Empat*" yang telah dibuat adalah sebagai berikut:

1. Dengan adanya *game* "*Game Edukasi Tertib Lalu Lintas Roda Empat*" akan menambah wawasan seputar berkendara sesuai dengan aturan yang berlaku.
2. Aplikasi ini mudah digunakan karena memiliki tampilan yang sederhana namun jelas.
3. Aplikasi ini memiliki grafis yang menarik.
4. Dengan adanya aplikasi ini bisa menjadi referensi dalam pembuatan *game* sejenis.

b. Kekurangan

Kekurangan dari "*Game Edukasi Tertib Lalu Lintas Roda Empat*" yang telah dibuat adalah sebagai berikut:

1. Membutuhkan spesifikasi *mobile phone* yang tinggi.
2. NPC kurang responsif dan terlihat kaku.