

BAB IV HASIL DAN ANALISA

1.1. Implementasi Permainan

Game Colour Resistor ini dibuat dengan menu-menu dan konsep program yang telah dijelaskan pada bab tiga dan hasil dari program game yang telah dibuat tersebut akan diuraikan pada bab ini. *Game Colour Resistor* ini tidak dibuat dalam bentuk bundel instalasi pada PC, tetapi merupakan program portabel yang dapat dijalankan di semua PC dengan baik, minimum *operating system* yang bisa digunakan adalah Windows 7. Berikut adalah uraian hasil pembuatan dari *game* tersebut.

1.2. Penulisan Kode Program

Penulisan kode program digunakan untuk mendesain proses jalannya permainan sebelum ditampilkan pada antarmuka aplikasi.

1.2.1. Penulisan Script Menu Utama

Kode program pada gambar 4.1 dan 4.2. di bawah ini menjelaskan pembuatan menu utama dari aplikasi *resistor colour game*.

```
private GameObject qM;
void Start () {
    if (GameObject.Find ("quitMenu") != null) {
        qM = GameObject.Find ("quitMenu");
        qM.SetActive (false);
    }
    if (!PlayerPrefs.HasKey ("jumlahGelang")) {
        PlayerPrefs.SetInt ("jumlahGelang", 4);
    }
    for (int i = 0; i < 10; i++) {
        if (!PlayerPrefs.HasKey ("Highscore4" + i)) {
            PlayerPrefs.SetInt ("Highscore4" + i, 0);
            PlayerPrefs.SetString ("Name4" + i, "Empty");
        }
    }
}
```

Gambar 4.1. *Script Menu Utama*

```

        if (!PlayerPrefs.HasKey ("Highscore5" + i)) {
            PlayerPrefs.SetInt ("Highscore5" + i, 0);
            PlayerPrefs.SetString ("Name5" + i, "Empty");
        }
    }
}
public void Ganti_Scenes(string scene){
    SceneManager.LoadScene (scene);
}
public void Quit(){
    Application.Quit ();
}
public void quitMenu(bool sets){
    qM.SetActive (sets);
}
}

```

Gambar 4.2. *Script Menu Utama*

1.2.2. Penulisan *Script Menu Help*

Berikut ini merupakan *script code* yang menjalankan perintah menu *help* atau bantuan ketika tombol *help* diklik pada aplikasi *resistor colour game*. Lebih jelasnya dapat dilihat pada gambar 4.3. di bawah ini.

```

public void gantiHalaman(int pg){
    if (pg == 1) {
        page += 1;
    } else {
        page -= 1;
    }
    for(int i=0;i<Pages.transform.childCount;i++){
        if(i==page){
            Pages.transform.GetChild (i).gameObject.Set
Active (true);
            PagesL.transform.GetChild(i).gameObject.Set
Active (true);
        } else{
            Pages.transform.GetChild (i).gameObject.Set
Active (false);
            PagesL.transform.GetChild(i).gameObject.Set
Active (false);
        }
    }
}
public void Ganti_Scenes(string scene){
    SceneManager.LoadScene (scene);
}
}

```

Gambar 4.3. *Script Menu Help*

1.2.3. Penulisan *Script Menu Highscore*

Berikut ini merupakan penulisan *script* untuk menjalankan menu *highscore* yang diklik dari aplikasi *resistor colour game*. Lebih jelasnya dapat dilihat pada gambar 4.4. di bawah ini.

```

void Start () {
    tombolEmpat = GameObject.Find ("Empat");
    tombolLima = GameObject.Find ("Lima");
    tampil = PlayerPrefs.GetInt ("jumlahGelang");
    for (int i = 0; i < 10; i++) {
        nama[i] = GameObject.Find ("Nama" + i).GetComponent<T
ext> ();
        skor[i] = GameObject.Find ("Skor" + i).GetComponent<T
ext> ();
        if (!PlayerPrefs.HasKey ("Highscore4" + i)) {
            PlayerPrefs.SetInt ("Highscore4" + i, 0);
            PlayerPrefs.SetString ("Name4" + i, "Empty");
        }
        if (!PlayerPrefs.HasKey ("Highscore5" + i)) {
            PlayerPrefs.SetInt ("Highscore5" + i, 0);
            PlayerPrefs.SetString ("Name5" + i, "Empty");
        }
    }
    PlayerPrefs.Save ();
}
void Update(){
    for (int i = 0; i < 10; i++) {
        nama [i].text = PlayerPrefs.GetString ("Name" + tampil + i);
        skor [i].text = PlayerPrefs.GetInt ("Highscore" + tampil + i).ToString ();
    }
}

```

Gambar 4.4. Script Menu Highscore

1.2.4. Penulisan *Script Menu Settings*

Berikut ini merupakan penulisan *script* untuk menjalankan menu *setting* pada aplikasi *resistor colour game*. Dengan melakukan klik pada menu *setting*, pemain dapat melakukan pengaturan *game*. Lebih jelasnya dapat dilihat pada gambar 4.5. di bawah ini.

```

private GameObject tombolEmpat, tombolLima, hapusMenu;
private Slider sliderVolume;
public int no;

void Start () {
    tombolEmpat = GameObject.Find ("Empat");
    tombolLima = GameObject.Find ("Lima");
    hapusMenu = GameObject.Find ("hapusMenu");
    sliderVolume = GameObject.Find ("volume").GetComponent<
Slider> ();
    sliderVolume.GetComponent<Slider> ().value = Audioliste
ner.volume;
    hapusMenu.SetActive (false);
}

void Update () {
    if (PlayerPrefs.GetInt ("jumlahGelang") == 4) {
        tombolEmpat.GetComponent<Button>().interactable = false;
        tombolEmpat.GetComponent<AudioSource> ().enabled = false;
        tombolLima.GetComponent<Button>().interactable = true;
        tombolLima.GetComponent<AudioSource> ().enabled = true;
    } else {
        tombolEmpat.GetComponent<Button>().interactable = true;
        tombolEmpat.GetComponent<AudioSource> ().enabled = true;
        tombolLima.GetComponent<Button>().interactable = false;
        tombolLima.GetComponent<AudioSource> ().enabled = false;
    }
    AudioManager.volume = sliderVolume.value;
}

```

Gambar 4.5. *Script Menu Setting.*

1.3. Tampilan Antar Muka

Tampilan ini berisikan *screenshot* dari menu serta bentuk permainan pada aplikasi yang telah dibuat, tampilan ini dapat dilihat lebih jelas sebagai berikut.

4.3.1. Layar Utama

Tampilan layar utama game dapat dilihat pada gambar 4.1. berikut ini

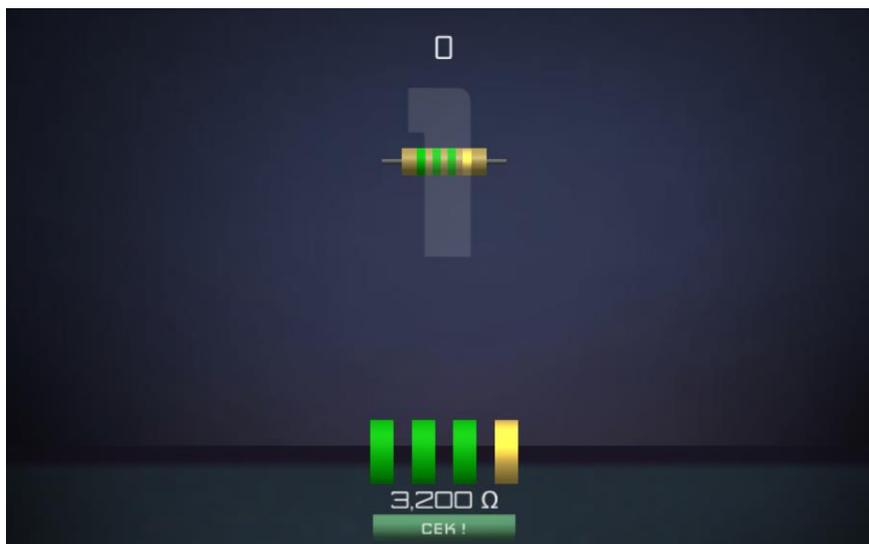


Gambar 4.6. Tampilan layar utama *resistor game*.

Pada tampilan gambar 4.6. di atas dapat dijelaskan bahwa layar utama ini terdapat menu *Play*, *Highscores*, *Help*, *Setting* dan *Exit*. Dimana menu *Play* untuk memulai permainan, menu *Highscores* untuk melihat record perolehan nilai *score*, menu *Help* untuk melihat panduan permainan, menu *Setting* untuk menyeting permainan serta menu *Exit* untuk keluar dari permainan. Dengan mengklik menu *Play*, maka akan membuka layar permainan yang nampak pada gambar 4.7.

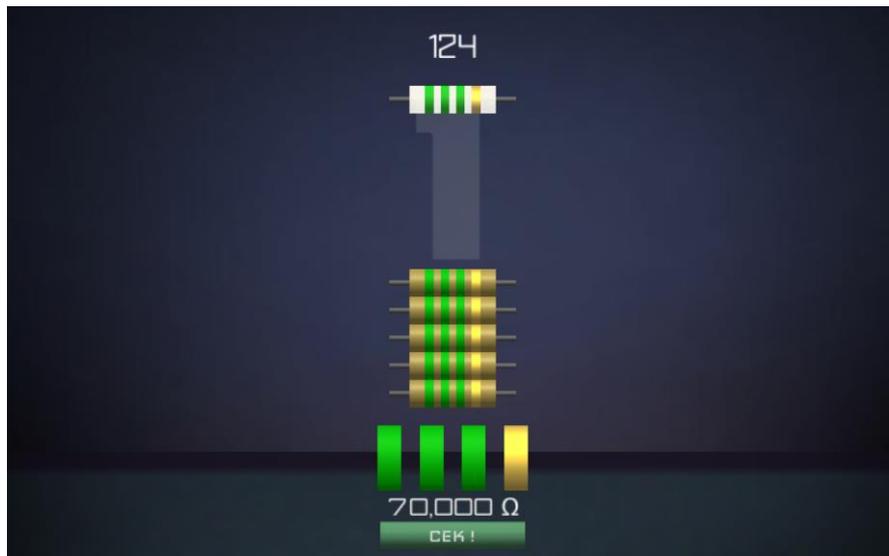
4.3.2. Layar *Play*

Pada layar *play*, posisi angka yang paling atas adalah nilai perolehan *score* dari permainan. Pengaturan pemberian nilai *score* pada permainan ini adalah setiap pemberian warna yang benar dari *resistor score* adalah 100 ditambah dengan prosentase posisi dimana *resistor* tersebut berhasil dijawab warnanya. Misalnya posisi *resistor* saat berhasil dijawab warnanya tepat ditengah antara posisi awal dengan dasar, maka nilai *score* yang diperoleh adalah 150. Dari penjelasan ini maka nilai maksimum dalam satu kali jawab benar adalah 200 (walaupun demikian ini tidak mungkin terjadi, karena dalam proses pemberian warna, posisi *resistor* tentu sudah bergerak turun) sedangkan nilai minimum adalah 100 (posisi jawab benar bertepatan dengan sampai didasar).



Gambar 4.7. Tampilan layar Play

Berdasarkan gambar 4.7. di atas dapat dijelaskan, angka *watermark* besar yang ada pada layar adalah posisi level permainan. Untuk mengubah-ubah warna pada *resistor* yang turun, digunakan kotak warna pemandu yang ada pada bagian bawah. Untuk mengubah warna dilakukan dengan melakukan klik pada warna tersebut. Pemilihan warna pada kotak warna disesuaikan dengan pertanyaan nilai *resistor* yang ada pada bagian bawah kotak warna. Jika pemilihan warna telah selesai dan pemain sudah yakin dengan jawabannya, maka tombol cek harus diklik. Bila jawaban pemain benar, maka *resistor* akan hilang dan *score* bertambah, bila salah maka *resistor* tersebut akan jatuh kebawah membentuk tumpukan. Bentuk tumpukan dari game dapat dilihat pada gambar 4.8. di bawah ini.



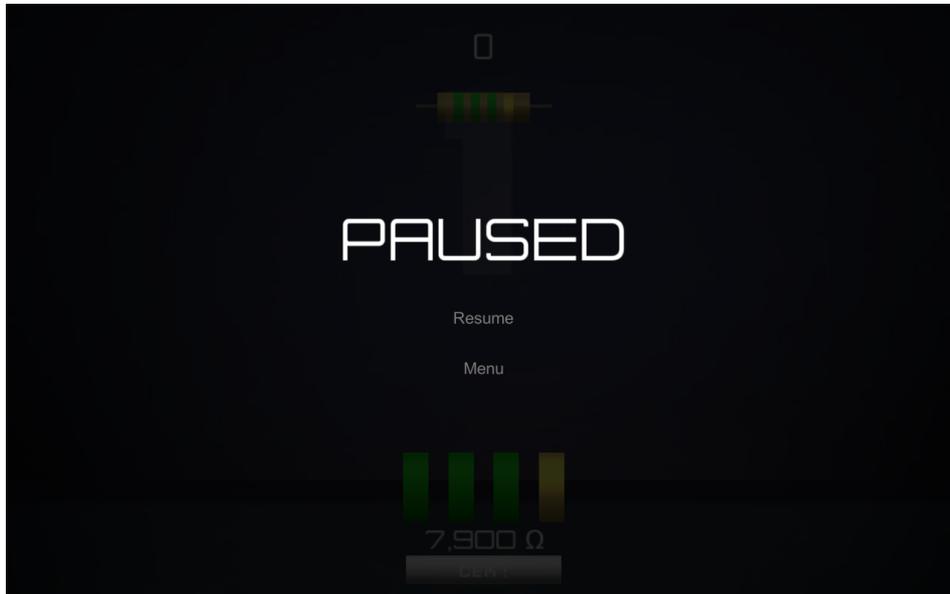
Gambar 4.8. Tampilan *resistor* yang membentuk tumpukan

Dalam permainan ini juga diberikan *resistor* bonus, bila *resistor* bonus ini benar dalam menjawabnya, maka tumpukan yang telah terjadi akan dihilangkan. Bentuk *resistor* bonus ini adalah warna dasar *resistornya* berkilip putih (*blink*) seperti yang dapat dilihat pada gambar 4.8. di atas. Adapun tampilan dari hilangnya tumpukan *resistor* dapat dilihat pada gambar 4.9. berikut ini.



Gambar 4.9. Saat tumpukan *resistor* dihilangkan

Saat permainan berlangsung, *game* juga dapat dilakukan *pause* dengan mengklik *icon pause* pada pojok kanan atas halaman permainan. Hasil pengklikan *icon* ini dapat dilihat pada gambar 4.10. di bawah ini.



Gambar 4.10. Tampilan paused game

Saat *game pause* ini terdapat dua menu yang dapat diklik, yaitu *resume* yang akan mengembalikan keposisi semula (permainan) dan menu yang akan mengembalikan ke menu awal. Bila yang diklik menu, maka ini sama juga dengan membatalkan permainan, dan ini akan membuat *score* yang sudah diperoleh tidak ter-*record* bila nilai *score* masuk kategori *highscore*.

Saat pemain mencapai “*game over*”, sistem akan memeriksa apakah nilai pemain memperoleh *record highscore*, bila iya, maka akan muncul layar untuk memasukkan nama pemain untuk dicatat dalam perolehan *highscore*. Tampilan pengisian nama *highscore* nampak pada gambar 4.11. di bawah ini.



Gambar 4.11. Tampilan pengisian nama perolehan highscore

4.3.3. Layar *Highscores*

Tampilan layar *highscores* digunakan untuk mencatat perolehan nilai-nilai tertinggi dari pemain. Pada layar *Highscores* ini terdapat 3 *icon* yang dapat diklik yang terdapat pada bagian bawah layar. *Icon* yang paling kiri adalah untuk kembali ke menu utama. *Icon* dengan angka 4 adalah untuk melihat perolehan *highscore* pada permainan dengan 4 gelang warna dan *icon* dengan angka 5 adalah untuk menampilkan *highscores* pada permainan 5 gelang warna. Tampilan layar *Highscores* nampak pada gambar 4.12. di bawah ini.

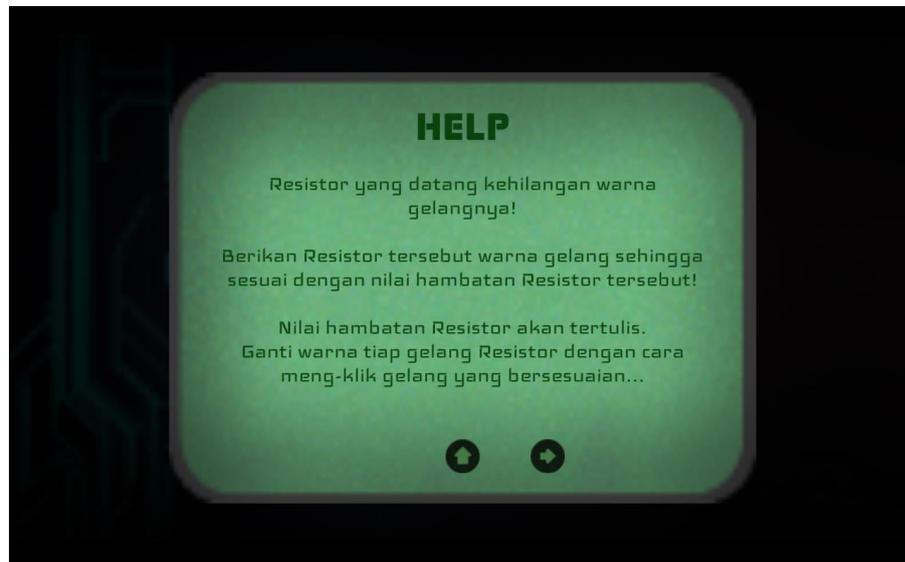


Gambar 4.12. Tampilan record highscore

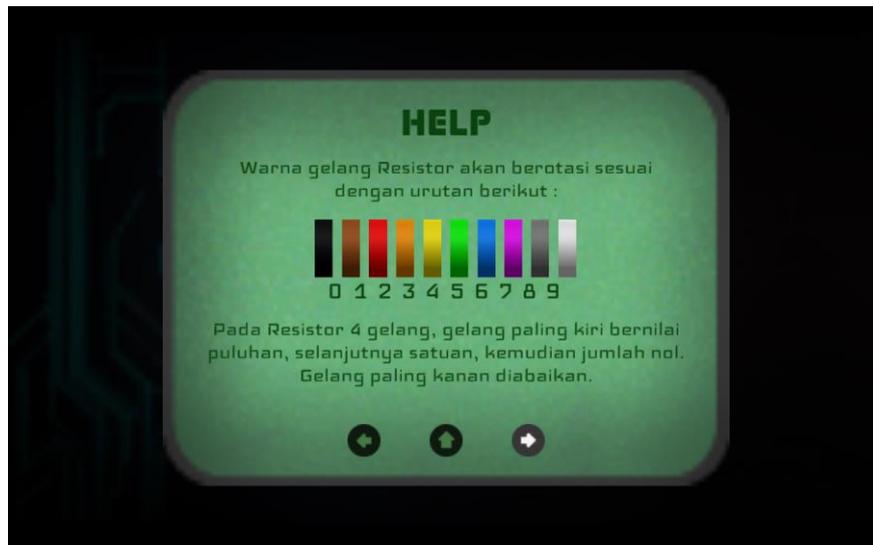
4.3.4. Layar *Help*

Seperti pada *game-game* yang sudah ada, *game resistor* yang dibuat ini juga diberikan menu panduan dalam permainan (*Help*), ini digunakan untuk pemain dalam memahami permainan. Layar panduan dalam game ini dibuat sebanyak 6 halaman yang berisi cara bermain, cara menentukan warna pada nilai *resistor* yang ditanyakan, petunjuk nilai-nilai warna pada *resistor* serta contoh gambar warna pada *resistor*.

Pada layar *Help* ini terdapat 3 *icon* dibagian bawah, yaitu *icon home* untuk kembali ke menu utama, *icon* panah kiri untuk *preview* halaman sebelumnya dan *icon* panah kanan untuk *preview* halaman berikutnya. Pada halaman awal layar *Help* tidak ada *icon* panah kiri dan pada halaman akhir layar *Help* tidak ada *icon* panah kanan, ini sekaligus untuk menandakan halamn awal dan halam akhir layar *Help*. Berikut adalah tampilan-tampilan layar *help* mulai dari gambar 4.13. hingga 4.18. yang dapat dilihat di bawah ini.



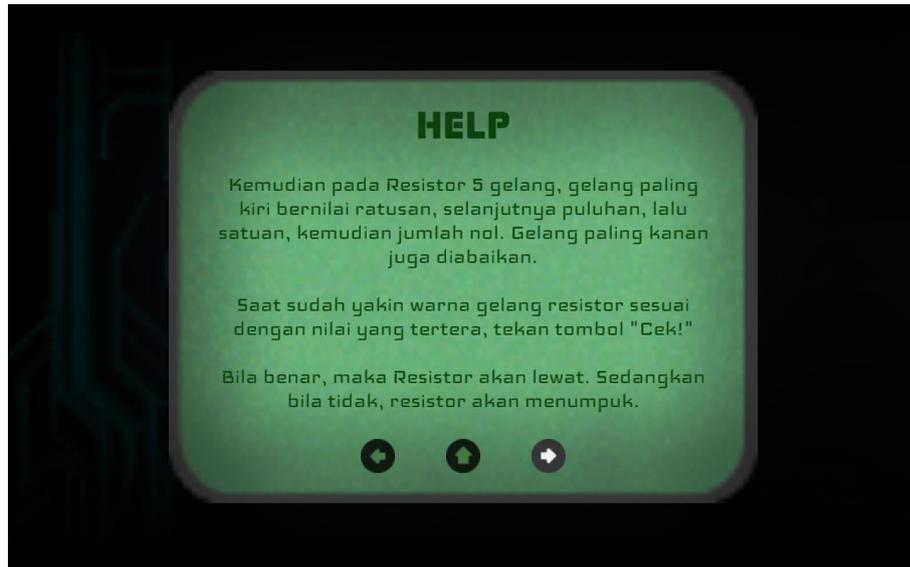
Gambar 4.13. Layar *Help* halaman pertama



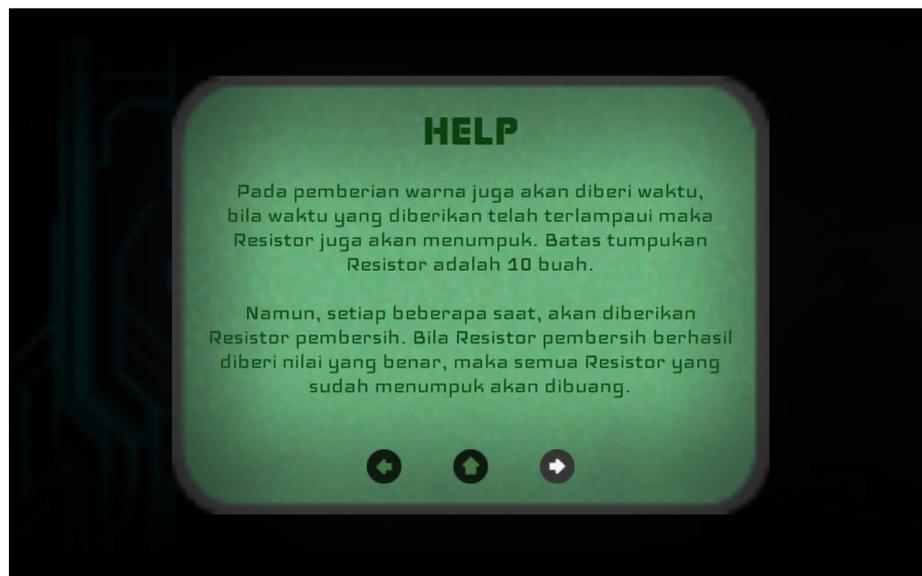
Gambar 4.14. Layar *Help* halaman kedua



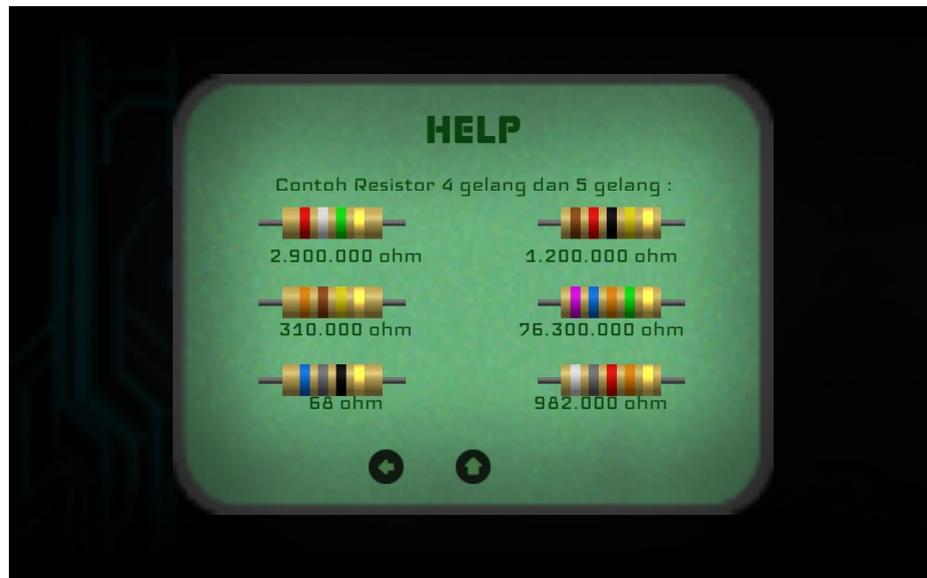
Gambar 4.15. Layar *Help* halaman ketiga



Gambar 4.16. Layar *Help* halaman keempat



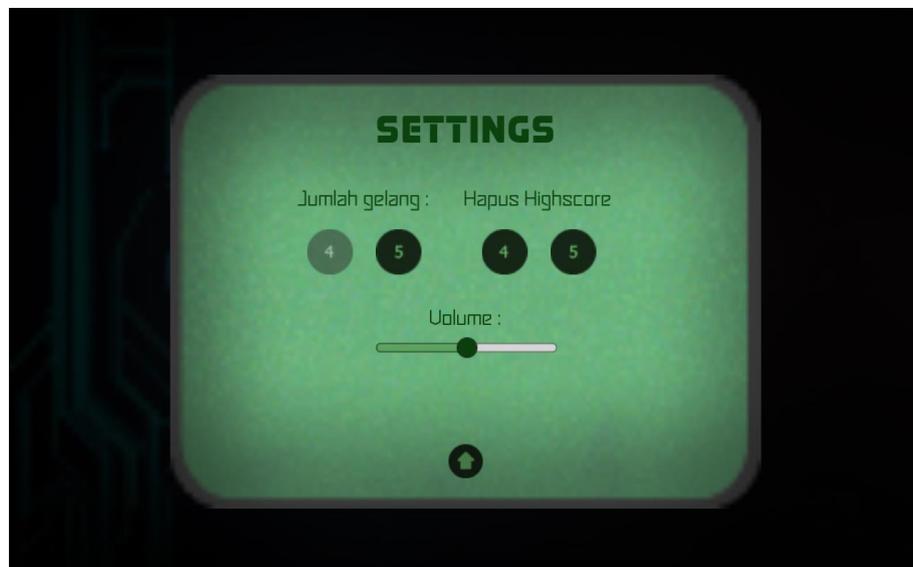
Gambar 4.17. Layar *Help* halaman kelima



Gambar 4.18. Layar *Help* halaman keenam

4.3.5. Layar *Setting*

Tampilan layar *Setting* dari resistor colour game dapat dilihat pada gambar 4.19. di bawah ini.

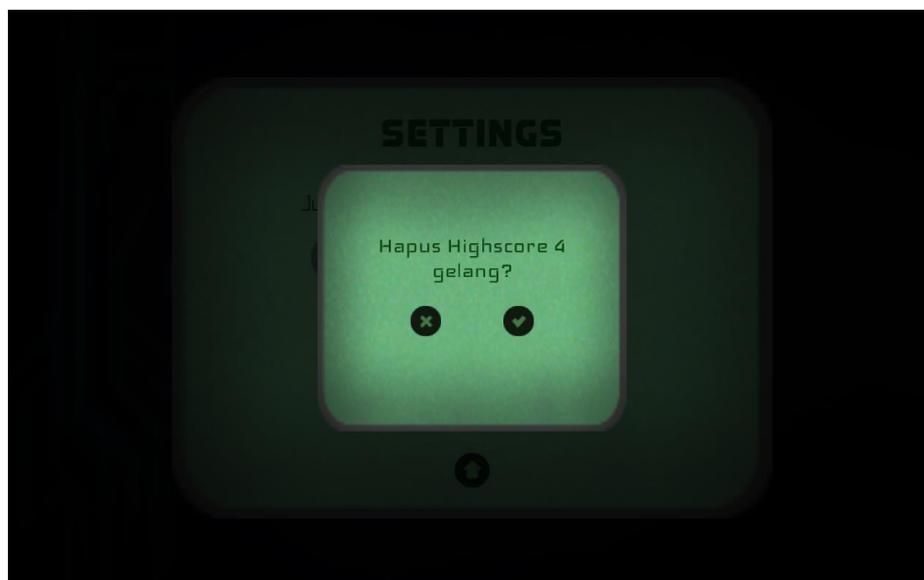


Gambar 4.19. Tampilan layar *Setting*

Terdapat 6 *icon* yang dapat diklik pada tampilan layar *Setting*. Pada posisi jumlah gelang terdapat dua *icon* dengan angka 4 dan 5. *Icon* ini digunakan untuk

memilih jenis permainan warna *resistor*, yaitu 4 dan 5 gelang warna. Jenis yang terpilih adalah yang terlihat dengan warna *icon* yang abu-abu.

Pada posisi hapus *Highscore* juga terdapat dua *icon* dengan angka 4 dan 5. *Icon* ini digunakan untuk menghapus *record* perolehan *Highscores* pada catatan jenis permainan yang bersangkutan, dimana *icon* dengan angka 4 digunakan untuk menghapus *record* permainan jenis 4 gelang warna, sedangkan *icon* dengan angka 5 untuk menghapus *record* permainan jenis 5 gelang warna. Bila user ingin menghapus *record*, maka program akan memunculkan layar untuk memastikan perintah hapus. Untuk lebih jelasnya dapat dilihat tampilannya pada gambar 4.20 di bawah ini.



Gambar 4.20. Tampilan layar pemastian penghapusan *record highscores*

Pada layar ini terdapat dua *icon* dengan gambar tanda silang dan centang, bila *icon* dengan tanda silang yang diklik, maka *record* tidak jadi dihapus dan kembali ke layar sebelumnya. Bila tanda centang yang diklik, maka *record* akan dihapus dan tampilan perolehan *highscores* semua akan bernilai 0.

Pengaturan selanjutnya yang tersedia pada layar *setting* adalah pengaturan *volume* suara yang ada dalam permainan seperti musik, klik dan animasi. Pengaturan *volume* ini dilakukan dengan mengklik tombol pada *volume*, tahan, dan menggesernya ke kiri atau ke kanan sesuai dengan tingkat *volume* yang diinginkan.

Untuk keluar dari layar *setting* dan kembali ke menu utama dilakukan dengan mengklik *icon home* pada bagian bawah layar *Setting*.

4.3.6 Keluar Permainan

Untuk keluar dari aplikasi *Resistor Colour Game*, pada menu utama terdapat tombol *Exit* yang bila di klik akan muncul layar pertanyaan yang dapat dilihat seperti pada gambar 4.21. di bawah ini.



Gambar 4.21. Tampilan layar pemastian keluar dari permainan

Berdasarkan gambar 4.21. di atas, dapat dijelaskan bagi pemain yang ingin keluar dari permainan, pemain dapat mengklik *icon* dengan tanda centang. Bila tidak jadi, dapat mengklik *icon* dengan tanda silang.

4.4. Hasil Pengujian *Black Box*

Pengujian secara fungsional yang digunakan untuk menguji sistem adalah metode alpha. Metode yang digunakan pada pengujian aplikasi *resistor colour game* ini adalah pengujian *black box* yang berfokus pada fungsional dari sistem yang dibangun.

1.4.1. Pengujian Menu Utama

Kelas Uji	Skenario Uji	Hasil yang diharapkan	Kesimpulan
Menu Utama	Memilih Tombol Mulai <i>Game</i>	Menampilkan Area Permainan	Berhasil
	Memilih Tombol <i>Highscore</i>	Menampilkan Hasil <i>Highscore</i>	
	Mengklik Tombol Bantuan (<i>Help</i>)	Menampilkan Cara Bermain	
	Mengklik Tombol Pengaturan (<i>Setting</i>)	Menampilkan Pengaturan <i>Game</i>	
	Mengklik Tombol keluar	Keluar dari permainan	

Tabel 4.1. Pengujian Menu Utama

1.4.2. Pengujian Mulai Game

Kelas Uji	Skenario Uji	Hasil yang diharapkan	Kesimpulan
Mulai <i>Game</i>	Ketika pemain mengklik tombol cek	Jika jawaban benar akan menambah skor	Berhasil
	Ketika pemain mengklik gelang warna pada <i>resistor</i>	Warna gelang pada <i>resistor</i> berubah	
	Tombol jeda permainan	Jeda Permainan	

Tabel 4.2. Pengujian Mulai *Game*

1.4.3. Pengujian Menu Highscore

Kelas Uji	Skenario Uji	Hasil yang diharapkan	Kesimpulan
<i>Highscore</i>	Ketika pemain memilih tombol <i>highscore</i>	menampilkan hasil skor permainan	Berhasil
	Tombol kembali (<i>Home</i>)	Kembali ke menu utama	

Tabel 4.3. Pengujian Menu *Highscore*

1.4.4. Pengujian Menu Bantuan (Help)

Kelas Uji	Skenario Uji	Hasil yang diharapkan	Kesimpulan
<i>Help</i>	Ketika pemain memilih tombol bantuan (<i>Help</i>)	Menampilkan papan petunjuk	Berhasil
	Tombol kembali (<i>Home</i>)	Kembali ke menu utama	

Tabel 4.4. Pengujian Menu Bantuan

1.4.5. Pengujian Menu pengaturan (*Setting*)

Kelas Uji	Skenario Uji	Hasil yang diharapkan	Kesimpulan
<i>Settings</i>	Ketika pemain memilih tombol pengaturan (<i>Setting</i>)	Menampilkan papan pengaturan permainan	Berhasil
	Ketika pemain memilih tombol hapus <i>highscore</i>	Data <i>highscore</i> yang ada kembali kosong	

	Tombol kembali	Kembali ke menu utama	
--	----------------	-----------------------	--

Tabel 4.5. Pengujian Menu Pengaturan

1.4.6. Pengujian Menu Keluar

Kelas Uji	Skenario Uji	Hasil yang diharapkan	Kesimpulan
Menu Keluar	Ketika pemain memilih tombol keluar	Keluar dari aplikasi	Berhasil

Tabel 4.6. Pengujian Menu Keluar

4.5. Analisa

Dari hasil pengamatan dalam menjalankan program yang dibuat, terlihat bahwa setiap bagian dari program berjalan dengan baik dan sesuai dengan perencanaan konsep *game* yang telah disusun, seperti yang disampaikan pada bab tiga, mulai dari awal (menu utama) hingga keluar dari program. Demikian pula animasi-animasi gambar dan suara yang dibuat juga berjalan dengan baik.

Dari proses pembuatan program *game resistor* ini yaitu dengan perpaduan pemrograman bahasa C# dan visual animasi 3D Unity, penulis dapat melakukan penarikan kesimpulan bahwa dua program ini merupakan sarana pembuatan program *game* animasi yang cukup handal untuk digunakan kalangan pembuat *game*.