

3. ANALISIS DAN DESAIN SISTEM

3.1 Analisa Permasalahan

Pada tahap ini, penulis akan melakukan analisis terhadap permasalahan yang ada pada penelitian ini. Masalah yang akan diidentifikasi mencakup masalah-masalah yang terkait dengan pemahaman bahasa pemrograman, ketertarikan belajar, dan efisiensi belajar. Survey dilakukan dengan penyebaran angket yang di isi sebanyak 21 responden oleh mahasiswa Prodi informatika yang sedang belajar pemrograman dasar. Berikut Hasil *survey* Analisa masalah dapat dilihat pada Tabel 3.1.

Tabel 3. 1 Tabel Hasil *Survey* analisis Masalah

No.	Pertanyaan	Jawaban Pengguna (%)				
		Sangat Setuju	Setuju	Cukup	Tidak Setuju	Sangat Tidak Setuju
1	Apakah Anda Percaya diri dalam membuat program sederhana menggunakan bahasa pemrograman?	4.8	19	38.1	38.1	10
2	Apakah Anda pernah mengalami kesulitan dalam memahami konsep dasar pemrograman seperti variabel, kondisi, perulangan, dan fungsi?	9.5	71.4	14.3	4.8	0
3	Apakah Anda pernah merasa tidak tertarik atau bosan ketika belajar pemrograman?	9.5	33.3	42.9	14.3	0
4	Apakah Anda percaya bahwa membuat pemrograman lebih menarik dengan cara tertentu dapat meningkatkan minat Anda dalam belajar?	42.9	28.6	28.6	0	0

5	Apakah Anda merasa perlu bantuan tambahan atau sumber daya yang lebih baik untuk mendukung pembelajaran pemrograman Anda?	38.1	47.6	14.3	0	0
---	---	------	------	------	---	---

Berdasarkan hasil kuisioner yang telah dianalisis, beberapa masalah utama dalam pembelajaran pemrograman telah teridentifikasi. Pertama, setengah dari responden menunjukkan tingkat percaya diri yang rendah dalam membuat program sederhana menggunakan bahasa pemrograman (4.8% sangat tidak setuju, 38.1% tidak setuju dan 38.1% cukup). Ini menunjukkan bahwa beberapa mahasiswa merasa kurang percaya diri dalam kemampuan pemrograman, yang dapat mempengaruhi motivasi mereka untuk belajar lebih lanjut. Kemudian, responden mengalami kesulitan dalam memahami konsep dasar pemrograman seperti variabel, kondisi, perulangan, dan fungsi sebesar 71.4% menjawab setuju. Ini mengindikasikan adanya tantangan dalam pemahaman dasar-dasar pemrograman yang perlu diatasi dalam pembelajaran.

Selanjutnya, sebagian besar responden percaya bahwa membuat pemrograman lebih menarik dengan cara tertentu dapat meningkatkan minat mahasiswa dalam belajar, hasil survey sebesar 42.9% menunjukkan bahwa responden sangat setuju dan 28.6% menjawab setuju. Hampir seluruh responden merasa perlu bantuan tambahan atau sumber daya tambahan untuk mendukung pembelajaran pemrograman. Hal ini membuktikan bahwa mahasiswa perlu metode pembelajaran tambahan yang lebih interaktif untuk membantu mahasiswa dalam pembelajaran pemrograman dasar.

3.2 Analisa Kebutuhan

Berdasarkan analisa permasalahan yang ada, maka dibuat sebuah pembelajaran berbasis visual novel. Untuk memudahkan interaksi dalam penggunaan sistem, maka dibuatlah beberapa fitur untuk memudahkan pengguna dalam menggunakan sistem:

a. Kebutuhan Fungsional

Kebutuhan fungsional adalah Kebutuhan yang terkait dengan fasilitas yang dibutuhkan oleh sistem secara umum. Kebutuhan fungsional dari sistem yang akan dibangun meliputi:

- 1) Login
 - 2) Main Menu
 - 3) Achievement pembelajaran
 - 4) Play stories
 - 5) Minigames
- b. Kebutuhan non Fungsional

Kebutuhan non fungsional adalah Batasan-batasan penggunaan yang dimiliki aplikasi. Kebutuhan non fungsional dari aplikasi ini diantaranya:

- 1) Visual novel mudah digunakan dan navigasi dalam aplikasi intuitif.
- 2) Sistem yang dikembangkan dapat dijangkau melalui website.
- 3) Aplikasi dapat diakses oleh mahasiswa kapan saja.

Berdasarkan hasil survei pada Tabel 3.1, sejumlah mahasiswa mengalami kesulitan dalam pemahaman konsep pemrograman. Sebagian besar mahasiswa merasa perlu bantuan tambahan untuk mendukung pembelajaran mereka. Fitur pencapaian pembelajaran dapat memberikan motivasi tambahan dengan memberikan penghargaan atau pengakuan kepada mahasiswa ketika mereka mencapai tujuan tertentu dalam pemahaman pemrograman. Sebagian besar responden merasa memiliki ketertarikan saat belajar pemrograman, akan tetapi terdapat mahasiswa yang merasa tidak tertarik atau bosan saat belajar pemrograman. Fitur "Play Stories" dapat memecah monoton pembelajaran dengan menyediakan cerita-cerita menarik yang berhubungan dengan pemrograman. Ini akan meningkatkan minat belajar mereka. Mahasiswa mengalami kesulitan dalam pemahaman konsep dasar pemrograman. Minigames dapat digunakan sebagai alat pembelajaran interaktif yang akan membantu mahasiswa memahami konsep pemrograman dengan cara yang lebih menyenangkan dan menarik.

Dengan mengintegrasikan kebutuhan fungsional ini dalam pembelajaran berbasis visual novel, akan memungkinkan solusi yang lebih efektif untuk mengatasi permasalahan yang diidentifikasi, seperti kurangnya minat, kesulitan pemahaman, dan kebutuhan akan motivasi tambahan dalam belajar pemrograman dasar. Dengan demikian, aplikasi ini dapat menjadi solusi yang efektif untuk meningkatkan pemahaman pemrograman dasar dan mengatasi masalah yang ada.

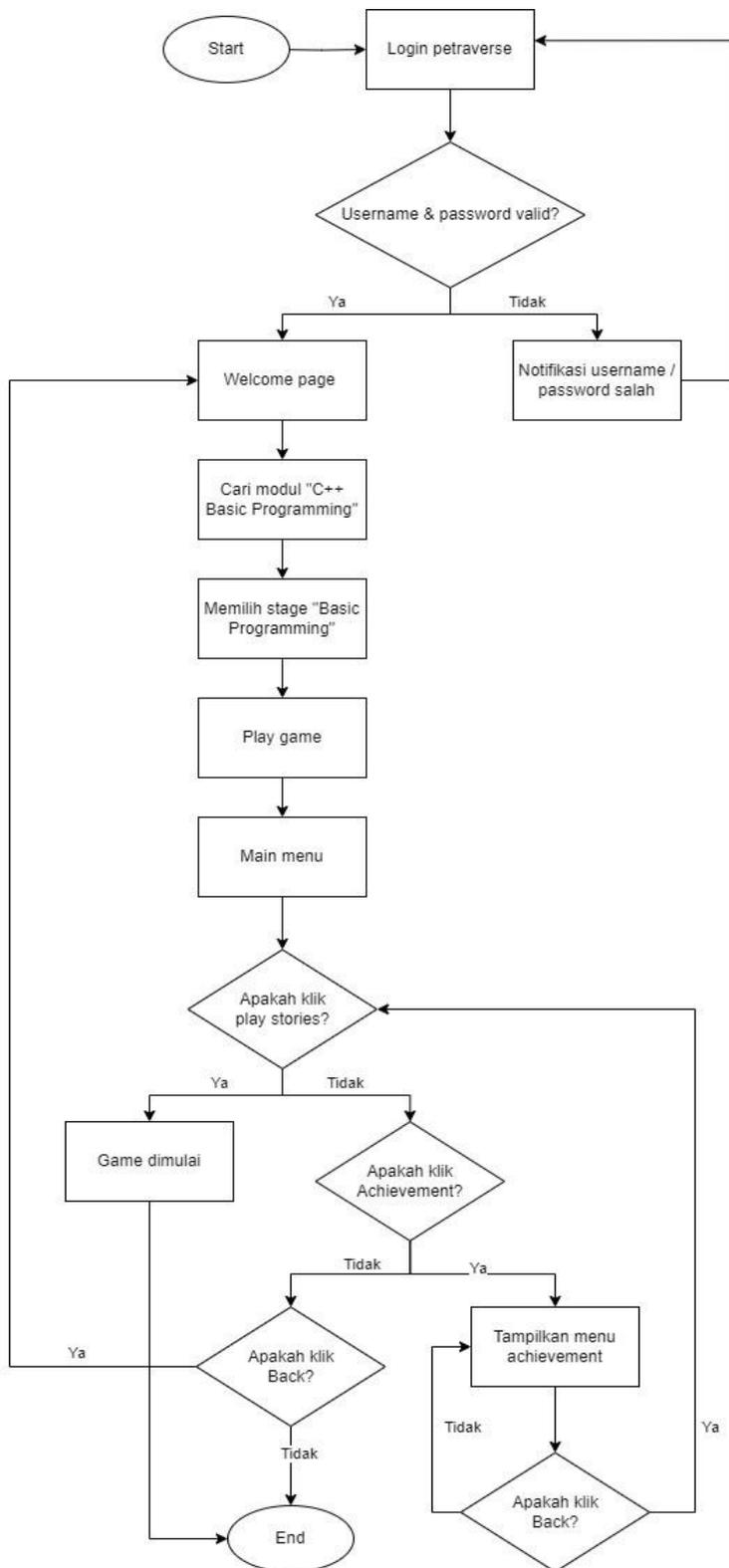
3.3 Desain Sistem

Ketika pemain pertama kali membuka game, pemain wajib melakukan login akun petaverse. Setelah itu, pemain memilih judul game visual novel, lalu memilih stage “Basic Programming” dan melakukan klik pada tombol play untuk memulai memainkan permainan. Setelah memilih stage, game akan di *load* dan pemain akan melihat menu utama yang terdapat tombol untuk memulai permainan, *achievement*, dan *back*. Jika pemain memilih *Achievement*, pemain akan melihat kumpulan dari *achievement* yang sudah didapat oleh pemain. Jika pemain memilih untuk memulai permainan, pemain akan melihat tampilan utama game yang berisi *dialogue box*, tombol kembali ke main menu, dan gambar *character*.

Pada saat memainkan *game* visual novel, pemain akan menemukan pertanyaan yang akan mempengaruhi hasil akhir permainan. Jika pemain berhasil menjawab pertanyaan dengan benar, pemain akan mendapatkan *achievement*. Pada akhir permainan, pemain dapat memainkan minigame yang disediakan di *website* terpisah.

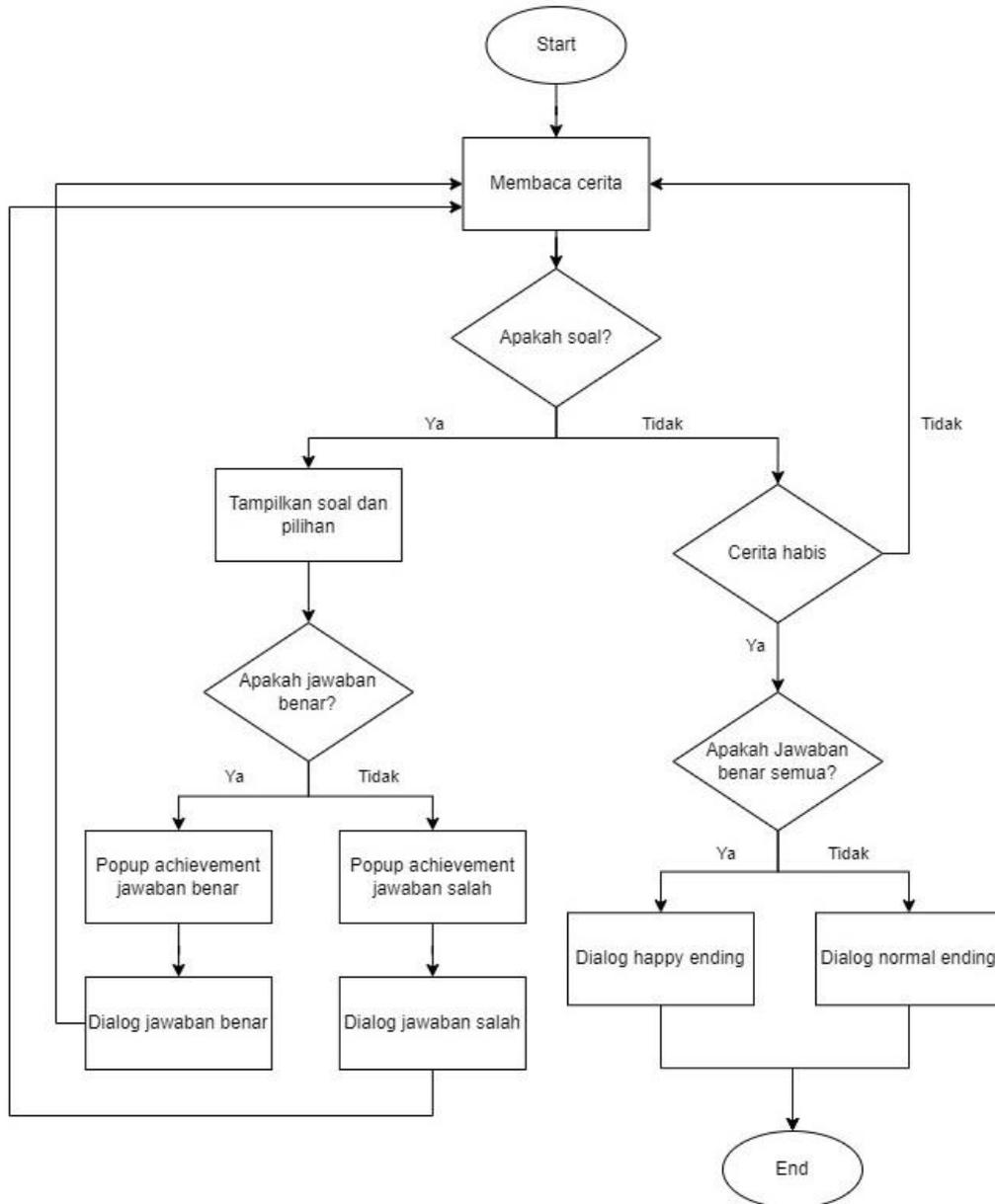
Saat memulai minigame, pemain akan melihat main menu minigame. Pada main menu, pemain dapat melihat tombol play dan back. Saat memilih tombol play, akan muncul *pop up* untuk memilih stage 1 sampai 3. Setelah memilih stage, pemain dapat memainkan minigame berupa *code block programming*. Jika pemain berhasil mencapai goal, akan muncul *pop up* berisi pertanyaan. Jika pemain berhasil menjawab pertanyaan dengan benar, pemain dapat melanjutkan ke stage berikutnya. Tetapi jika pemain tidak dapat menjawab pertanyaan dengan benar, akan muncul *pop up* salah menjawab pertanyaan dan pemain harus memainkan ulang minigame.

Gambar 3.1 merupakan penjelasan menggunakan flowchart untuk login kedalam petraverse sampai pemain dapat membuka halaman game.



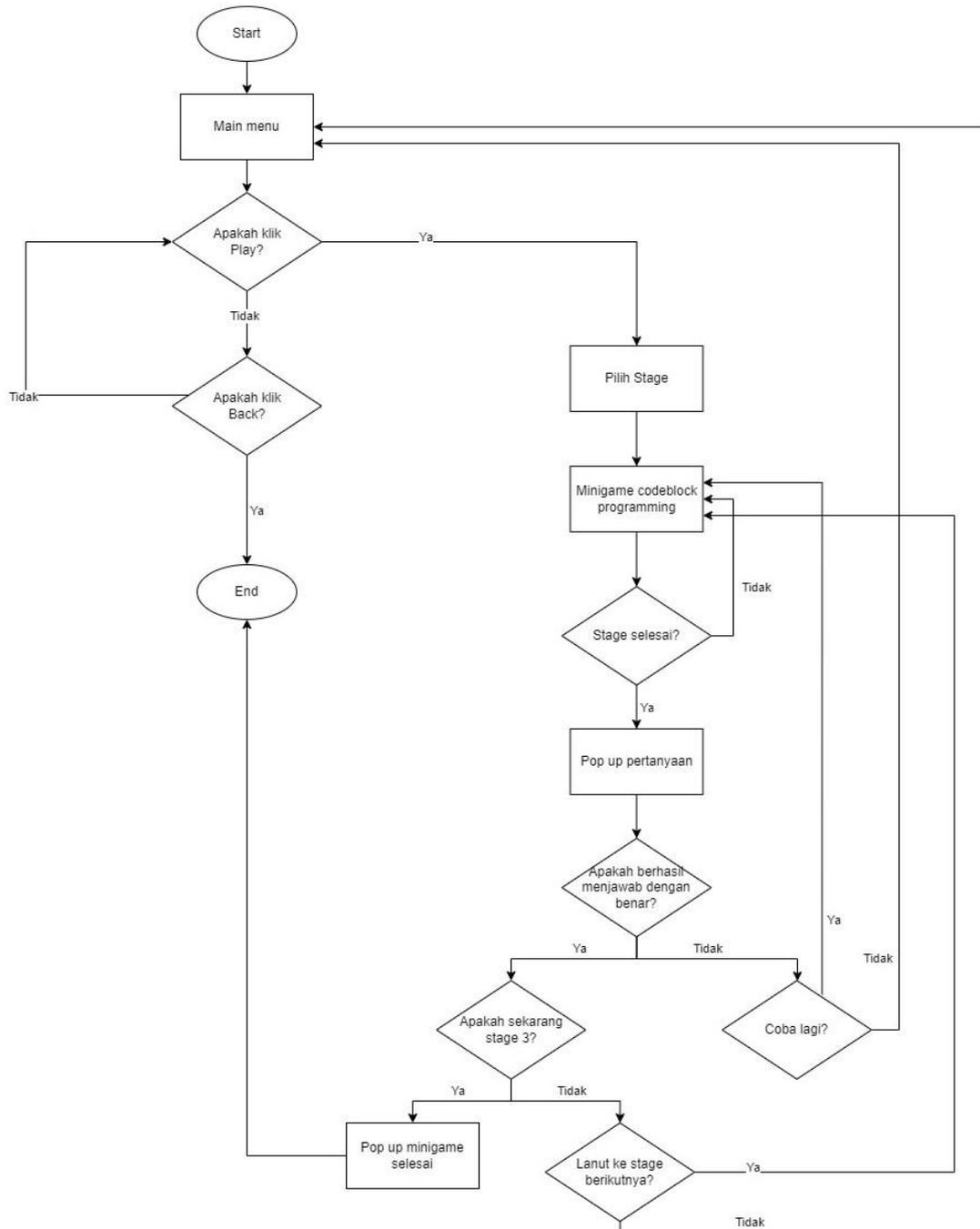
Gambar 3.1 Flowchart Login Petraverse dan membuka game

Gambar 3.2 merupakan penjelasan menggunakan flowchart untuk memainkan game.



Gambar 3.2 Flowchart Game Visual Novel

Pada Gambar 3.3 merupakan penjelasan menggunakan flowchart untuk *minigame code block programming*



Gambar 3.3 Flowchart Sistem Minigame

3.3.1 Game Design Document

Background Game:

Mahasiswa berperan sebagai tokoh utama bernama Josh. Cerita dimulai dari Josh seorang mahasiswa semester 1 Prodi informatika yang ketiduran di kelas, dengan keadaan kelas sudah terkunci. Disaat kebingungan, laptop Josh tiba-tiba mengatakan bahwa Josh tidak bisa keluar dari kelas sampai dia berhasil memahami konsep dasar pemrograman dasar. Karena beberapa hal, akhirnya laptop terpaksa membantu Josh memahami konsep pemrograman dasar agar Josh dapat keluar dari kelas.

Objective:

Pemain harus membaca cerita, dan memilih jawaban mengenai materi pemrograman dasar yang benar agar dapat keluar dari ruang kelas.

Gameplay:

1. Pemain membaca cerita dari awal sampai akhir.
2. Pemain harus memilih semua jawaban dengan benar agar pada akhir cerita pemain dapat keluar dari kelas.
3. Jika pemain menjawab pertanyaan materi dengan benar, pemain akan mendapat *achievement*.
4. Jika pemain memilih jawaban yang salah, pemain tidak akan mendapat *achievement* dan pada akhir cerita pemain tidak dapat keluar kelas.
5. Seluruh permainan dapat dilakukan dengan menggunakan klik mouse kiri.
6. Setelah selesai membaca cerita, pemain dapat melakukan minigames.

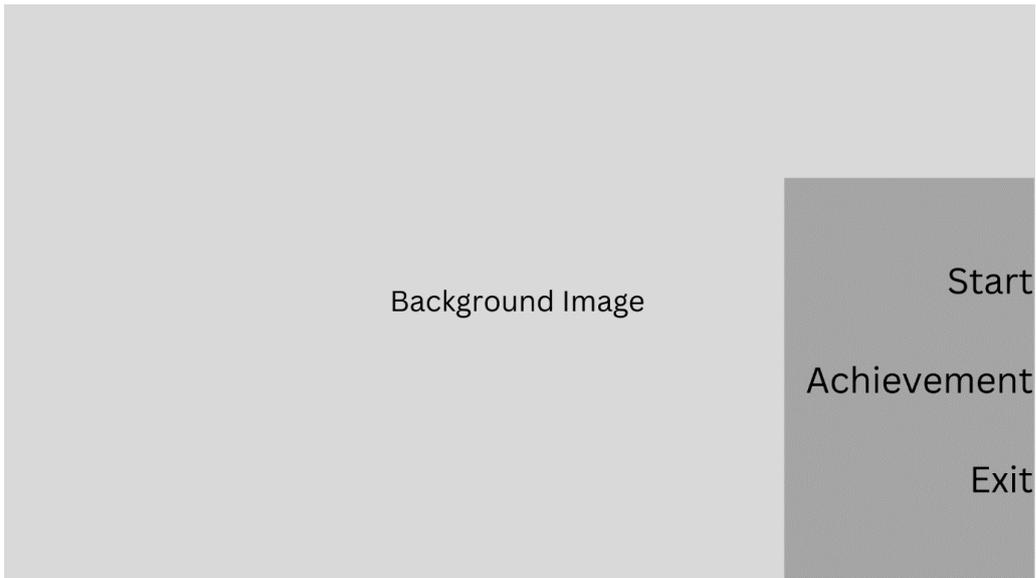
Detail Minigames:

1. Minigames berupa *code block programming* dimana pemain dapat melakukan drag and drop pada tombol anak panah yang sudah disediakan untuk menggerakkan *character* ke *goals*.
2. Terdapat 3 stage minigame yang dapat dimainkan oleh pemain.

3.4 Desain Interface

3.4.1 Desain Main Menu

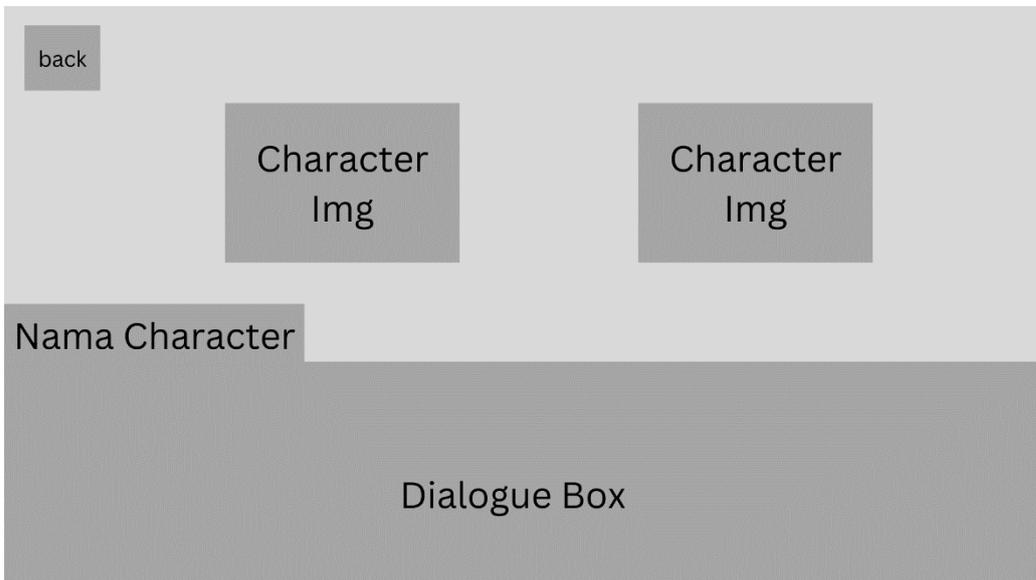
Pada gambar 3.4 pemain akan melihat main menu jika telah melakukan login. Pada main menu, pemain dapat memilih tombol start untuk memainkan game, achievement untuk melihat achievement yang sudah didapat, dan exit untuk keluar dari permainan.



Gambar 3.4 Desain *Main menu*

3.4.2 Desain Tampilan Utama Game

Gambar 3.5 Merupakan tampilan game utama yang didalamnya terdapat dialogue box, gambar character, dan tombol back untuk Kembali ke main menu.

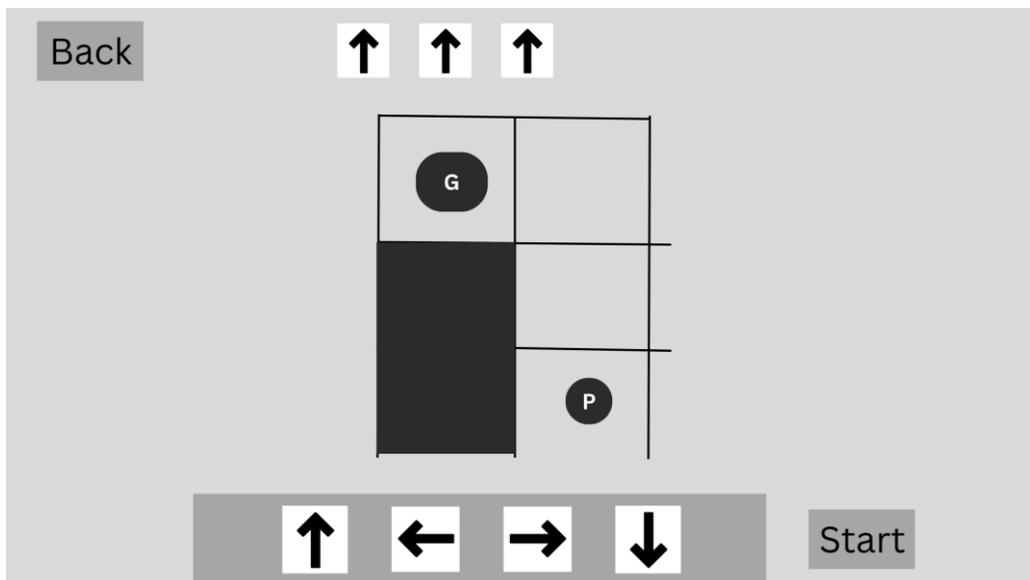


Gambar 3.5 Desain *Start Game*

3.4.3 Desain Minigame

Gambar 3.6 merupakan contoh desain minigame pada stage 1. Pemain dapat melakukan klik pada tanda panah yang disediakan pada kotak abu-abu agar character yang diibandingkan dengan lingkaran P dapat mencapai goal yang dilambangkan dengan lingkaran G.

Saat pemain menekan tombol tanda panah ke atas sebanyak 3 kali, maka akan muncul tanda panah dengan jumlah dan arah yang sama pada bagian atas layar permainan sebagai pengingat bahwa pemain sudah menekan tombol panah sebanyak 3 kali. Pemain tidak dapat menghapus perintah yang sudah diberikan sehingga jika pemain salah melakukan Langkah, pemain harus menekan tombol start lalu melakukan reload pada pop up atau back untuk kembali ke menu utama minigame.



Gambar 3.6 Desain *Minigame*

3.5 Materi Pembelajaran Pada game

Pada skripsi ini, materi pembelajaran terdapat pada dialog antar karakter diaman karakter light akan berperan sebagai guru atau tutor yang akan mengajari tokoh utama agar dapat keluar dari kelas yang terkunci. Berikut merupakan beberapa contoh dialog yang memiliki unsur pembelajaran.

3.5.1 Dialog Materi If

Light: Oke kalau gitu langsung aja ke fungsi if. Fungsi if itu ibaratnya kalau kamu lagi undian. Kamu pasti dapat vocer yang isinya nomor undian terus kalau nomor mu sama kaya yang di sebutin berarti kamu dapat hadiahnya.

Light: Contohnya gini.

```

if(nomorUndian == 777)
{
    cout << "Menang undian";
}
else
{
    cout << "Kurang beruntung";
}

```

Gambar 3.7 Contoh Kode If

Light: Dari gambar ini, kamu bisa lihat kalau misalnya nomor undian kamu = 777 kamu menang undian. Selain itu ya kamu gak dapat apa-apa.

Light: Terus ada juga yang namanya else if. Itu sama aja kaya hadiah ke 2, ke 3, dst. Ya kaya pilihan lainnya gitu.

3.5.2 Dialog Materi For If

Light: Kamu juga bisa gabungin if dan loop mu. Itu sama aja kaya kamu di hukum lari 20 kali dan tiap kali kamu selesai 10 putaran, kamu boleh istirahat sebentar.

Josh: Aku tau itu pasti...

Opsi A

```

int putaran = 1;
if(putaran == 10)
{
    cout << "Istirahat";
    for (putaran <= 20; putaran++)
    {
        cout << putaran << endl;
    }
}

```

Gambar 3.8 Contoh Kode For If Option A

Opsi B

```
for (int putaran = 1; putaran <= 20; putaran++)
{
    cout << putaran << endl;
    if (putaran == 10)
        cout << "Istirahat";
}
```

Gambar 3.9 Contoh Kode For If Option B

Rute ending A

Light: Aku gak tau kenapa kamu bisa pilih jawaban yang ini.

Light: Emang sih bisa dipakai if baru di dalamnya ada loop. Tapi dalam kasus kali ini, kamu harus taroh if dalam loop seperti jawaban B karena kan kamu harus keliling lapangan dulu dan kalau misalnya kamu sudah hitung sampai 10 baru boleh istirahat.

Light: Kalau misalnya pertanyaannya kamu harus makan siang dulu sebelum lari keliling lapangan 20 kali baru benar if dulu. If sudah makan siang, print sudah makan siang baru tulis fungsi loop nya. Else, belum makan siang.

Josh: Oh iya, ya.

Light: Next time lebih teliti dengar in apa yang aku jelasin.

Josh: Siap pak!

End rute A

Rute ending B

Light: Nice! Kupikir kamu bakal terkecoh karena ifnya gak ada bracketsnya. Kayanya kamu ada kemajuan nih.

Josh: Hehehe, ayo puji aku lagi

Light: Baru dipuji dikit udah belagu...

End rute B