

BAB IV

ANALISIS DAN PERANCANGAN

4.1 ANALISIS GAME

4.1.1 Analisis Domain Masalah *Game*

Pada Sekolah SMP Xaverius 1 Jambi, siswa atau siswi SMP tersebut belum dapat mengetik sepuluh jari dengan cepat dan benar. Di sekolah ini belum terdapat pelajaran mengetik, dan seharusnya lebih ditujukan khususnya untuk siswa-siswi SMP Xaverius 1 Jambi yang berguna melatih kemampuan dalam hal pengetikan. Sehingga kemampuan siswa siswi dalam mengetik sepuluh jari dengan cepat dan benar sangatlah kurang memuaskan berdasarkan hasil wawancara yang dilakukan kepada guru di sekolah SMP Xaverius 1 Jambi.

Untuk itu perlu dikembangkan sebuah *game* bertema edukasi yang dapat digunakan sebagai media pembelajaran yang dapat memotivasi siswa agar tertarik dalam belajar pengetikan sekaligus juga menambah kemampuan siswa-siswi dalam mengetik cepat menggunakan sepuluh jari. Ditambah dengan tampilan-tampilan yang menarik sehingga anak tidak mudah bosan dan lebih tertantang dalam bermain *game*.

4.1.2 Tujuan *Game*

Adapun tujuan pengembangan *game* ini yaitu menambah kemampuan siswa-siswi SMP Xaverius 1 Jambi dalam mengetik sepuluh jari dengan cepat dan benar, sehingga siswa-siswi tersebut mendapatkan bekal yang berguna pada saat dunia kerja yang berhubungan dengan pengetikan dalam komputer.

4.1.3 Konsep Cerita *Game*

Pada game edukasi ini, dilakukan perancangan game mengetik adventure dimana terdapat tokoh, monster dan jebakan-jebakan yang ada. Di *game* ini juga terdapat *stage*, *level* yang berbeda. User harus memainkan *stage* dan *level* berbeda agar bisa memasuki *stage* yang lain. Setiap *stagenya* memiliki 10 menit untuk mengetik.

Pada Stage 1 *user* melarikan diri dari kejaran monster, dan harus memecahkan balok yang menghalangi jalan *user* tersebut dengan mengetikkan kalimat panjang yang muncul.

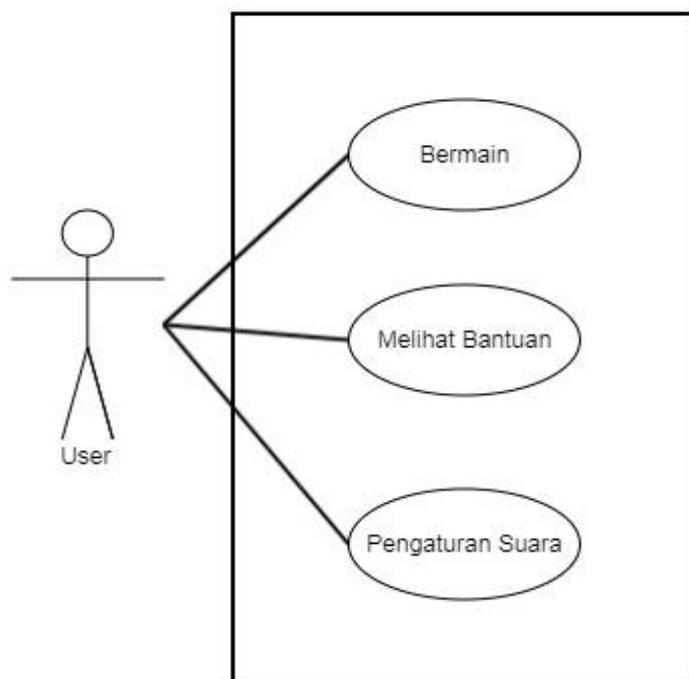
Pada Stage 2 *user* harus melepaskan diri dari jebakan blok didepan dan harus mengetik kalimat atau kata yang muncul dengan benar untuk lari dan menghancurkan blok lain.

Pada Stage 3 *user* diharuskan lari dari zombie beberapa detik setelah berjalan, pemain mengetikkan kalimat atau kata yang muncul agar mencapai tempat aman dan permainan tersebut selesai, jika waktu diberikan habis maka reruntuhan atap tersebut akan terjatuh kebawah dan permainan selesai atau gagal.

4.2 ANALISIS FUNGSI

4.2.1 Diagram *Use Case* yang Diusulkan

Diagram *use case* memperlihatkan hubungan-hubungan yang terjadi antara aktor-aktor dengan *use case* dalam sistemnya. Adapun *use case* digram pada *game* edukasi yang diusulkan adalah sebagai berikut:



Gambar 4.1 Use Case Diagram Proses Game Edukasi

Dari Diagram *Usecase* diatas, dibuat diagram berdasarkan fungsionalitas dari game edukasi tersebut, yang dimana pemain dapat memilih bermain, melihat bantuan, dan melakukan pengaturan suara. Sehingga pada saat pemain memilih bermain, maka ia akan diarahkan pada *stage* yang tersedia. Jika memilih melihat bantuan, maka akan diarahkan pada menu bantuan untuk melihat bagaimana cara bermain. Jika memilih pengaturan suara, akan diatur untuk menghidupkan musik latar permainan atau mematikannya.

4.2.2 Deskripsi Use Case

Untuk memudahkan dalam menganalisa skenario yang akan kita gunakan pada fase-fase selanjutnya dengan melakukan pemilihan dan penilaian terhadap skenario tersebut. Adapun skenario *Use Case* pengenalan *game* edukasi ini dalam sistem yang diusulkan adalah sebagai berikut :

Tabel 4.1 Deskripsi Usecase Bermain

Nama	Bermain
Aktor	User
Deskripsi	Untuk memulai menu permainan
Exception	Gagal memilih menu permainan
Pre Condition	Berhasil masuk ke menu level
Aktor	Sistem
Skenario Normal	
1. Aktor memilih menu Bermain	
	2. Sistem menampilkan menu Level
3. Aktor memilih Level	
	4. Sistem menampilkan permainan
	5. Sistem menampilkan Soal.
6. Aktor mengetik huruf atau kalimat dari soal yang keluar.	
	7. Sistem memvalidasi soal dan jawaban. 8. Jika Jawaban sama akan melanjutkan ke rintangan selanjutnya.
	9. Karakter user berjalan menyelamatkan diri.
10. Aktor Kembali mengetik huruf atau kalimat dari soal selanjutnya.	
	11. Sistem memeriksa apakah karakter mencapai garis finish atau tidak. 12. Sistem menampilkan pesan permainan selesai.
Skenario Alternatif	
	8a. Jika jawaban beda, aktor tidak dapat melanjutkan rintangan berikutnya. 8b. Aktor diberi kesempatan untuk mengetik huruf atau kalimat dengan sisa waktu yang ada. 8c. Jika waktu telah habis, maka karakter kalah atau mati.
Post Condition	Aktor berhasil menyelesaikan soal-soal yang disediakan.

Tabel 4.2 Deskripsi Usecase Melihat Bantuan

Nama	Melihat Bantuan	
Aktor	User	
Deskripsi	User melihat bantuan	
Exception	Gagal membuka menu bantuan	
Pre Condition	Aplikasi telah terinstal	
	Aktor	Sistem
Skenario Normal		
	1. Aktor berhasil masuk ke menu utama	
		2. Sistem menampilkan menu awal
	3. Aktor memilih pilihan Bantuan	
		4. Sistem menampilkan bantuan
Post Condition	Aktor berhasil melihat bantuan	

Tabel 4.3 Deskripsi Usecase Melakukan Pengaturan Suara

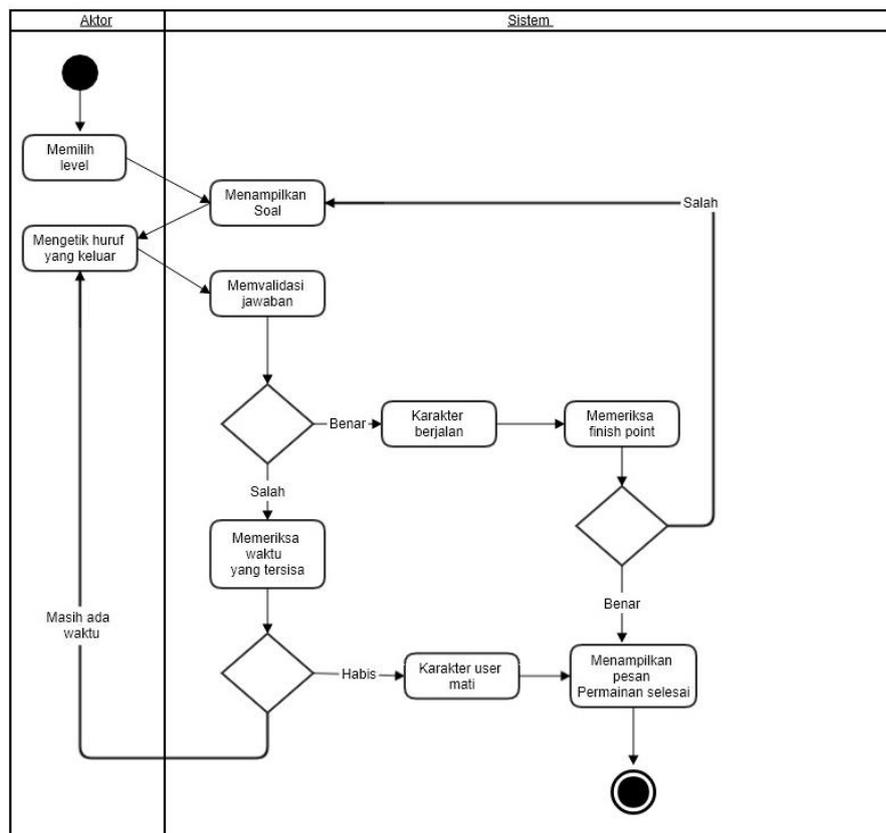
Nama	Melakukan Pengaturan Suara	
Aktor	User	
Deskripsi	User melakukan pengaturan suara	
Exception	Gagal mengatur suara	
Pre Condition	Aplikasi telah terinstal	
	Aktor	Sistem
Skenario Normal		
	1. Aktor berhasil masuk menu utama.	
		2. Sistem menampilkan menu awal.
	3. Aktor memilih pengaturan suara.	
		4. Sistem menghidupkan atau mematikan suara.
Post Condition	Aktor berhasil mengatur suara	

4.2.3 Diagram Aktivitas (*Activity Diagram*) yang diusulkan

Activity Diagram diperlukan untuk menggambarkan apa yang terjadi ketika aksi atau aktifitas dari suatu *state* diselesaikan, ketika aliran kendali akan menuju keaksi berikutnya atau aktifitas. Diagram aktifitas ini didesain

menggunakan *balsamiq mockups* karena lebih mudah dan efisien dibanding aplikasi lainnya. Balsamiq Mockups adalah alat khusus yang memungkinkan orang membuat dan meluncurkan aplikasi mereka sendiri dari desktop.

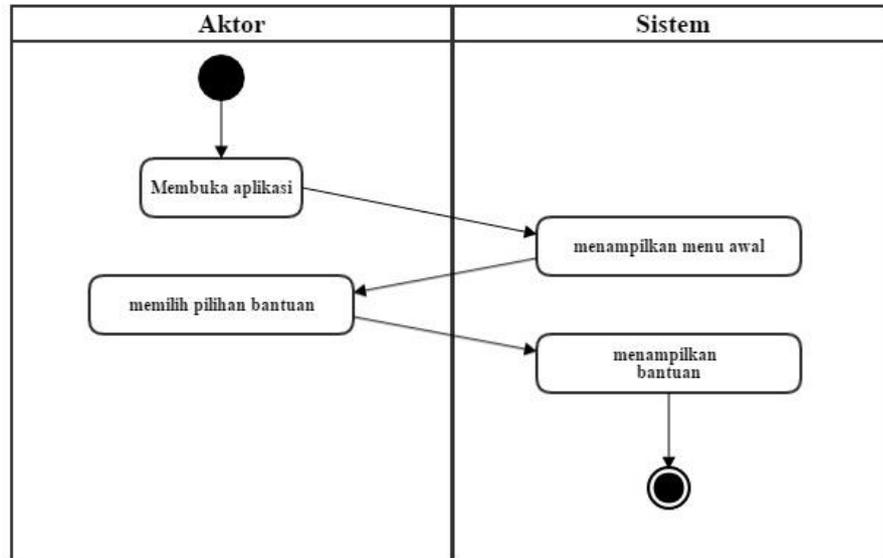
4.2.3.1 Diagram Aktivitas Bermain



Gambar 4.2 Diagram Aktivitas Bermain

Gambar diatas merupakan Diagram aktivitas dari deskripsi *usecase* Bermain pada tabel 4.1. Pada diagram ini menjelaskan bagaimana user menjalankan sistem atau program pada saat memilih menu bermain.

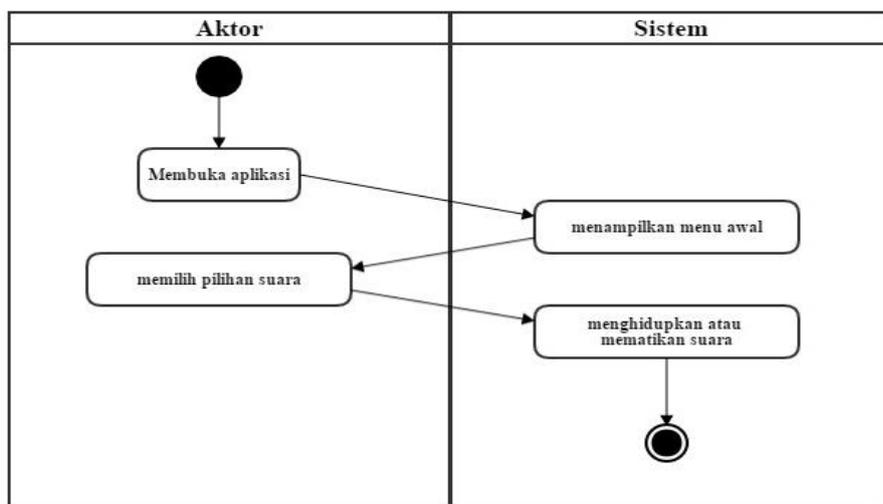
4.2.3.2 Diagram Aktivitas Melihat Bantuan



Gambar 4.3 Diagram AktivitasMelihatBantuan

Gambar diatas merupakan Diagram aktivitas dari deskripsi *usecase* Melihat Bantuan pada tabel 4.2. Pada diagram ini menjelaskan bagaimana user menjalankan sistem atau program pada saat memilih bantuan.

4.2.3.3 Diagram Aktivitas Pengaturan Suara

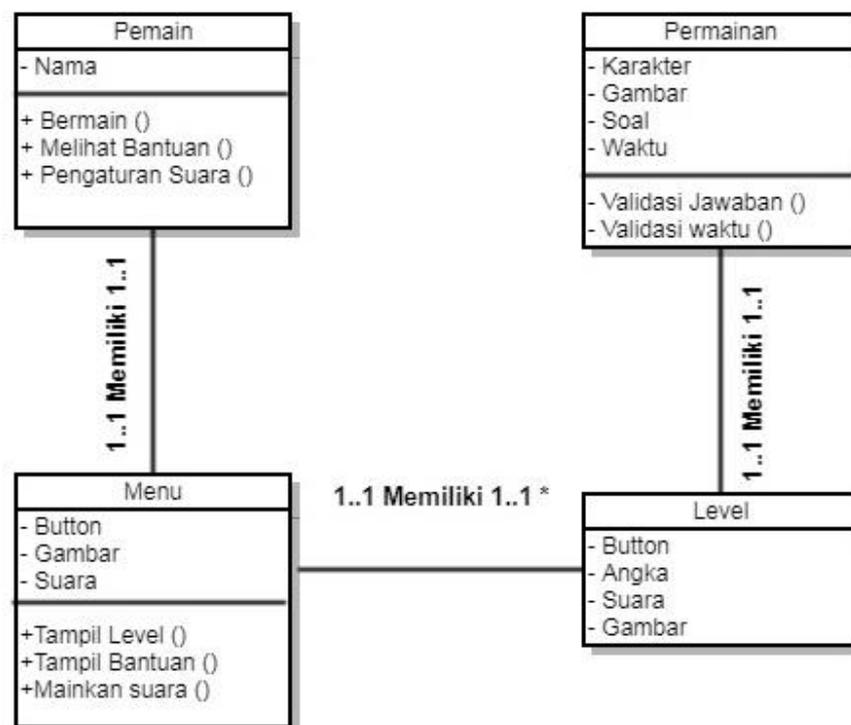


Gambar 4.4 Diagram Aktivitas PengaturanSuara

Gambar diatas merupakan Diagram aktivitas dari deskripsi *usecase* Pengaturan Suara pada tabel 4.3. Pada diagram ini menjelaskan bagaimana user menjalankan sistem atau program pada saat memilih pengaturan suara.

4.3 CLASS DIAGRAM

Untuk menggambarkan *class-class* yang ada di dalam aplikasi nantinya berupa *class diagram*. Berikut ini *class diagram* dari sistem yang sedang dirancang :



Gambar 4.5 Class Diagram dari Game Edukasi

4.4 PERANCANGAN ANTARMUKA

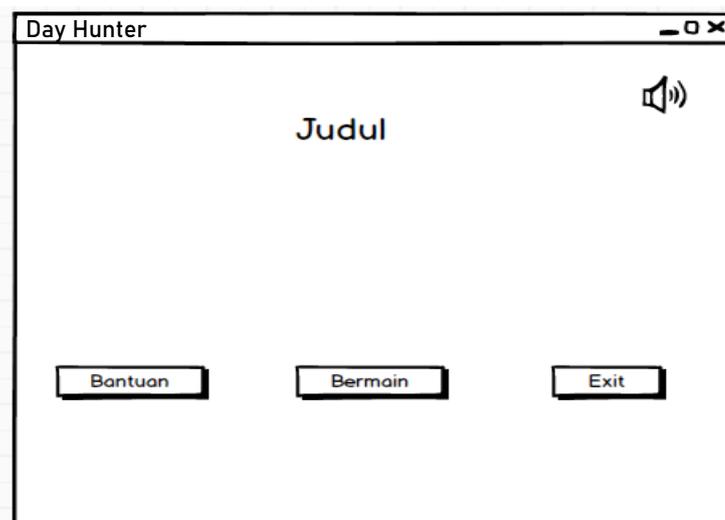
Perancangan Antar Muka (*interface*) merupakan guna untuk membentuk tampilan dari perangkat lunak yang berinteraksi dengan pemakai. Perancangan ini

menggunakan aturan-aturan tertentu untuk menghasilkan antar muka yang baik dan benar.

4.4.1 Perancangan Menu Utama

Desain menu utama adalah awal dimulainya proses *game* tersebut, desain ini dirancang untuk memenuhi pemakai agar proses dalam *game* tersebut dapat berjalan dengan sebagaimana mestinya. Didalam menu utama terdapat beberapa pilihan antara lain *Start*, *Bantuan*, *Suara*, *Keluar*. Pilihan-pilihan tersebut memiliki fungsi yang berbeda diantaranya.

Fungsi *start* berfungsi untuk memulai *game* itu sendiri dimana *user* ditugaskan untuk mengetik kata-kata yang disediakan dengan batas waktu, fungsi *Bantuan* berfungsi untuk membantu pengguna untuk mengetahui tentang fitur fitur yang terdapat pada *game* itu, sedangkan fungsi *Suara* berfungsi untuk menghidupkan atau mematikan suara permainan dan *Keluar* adalah fungsi untuk keluar dari aplikasi tersebut.

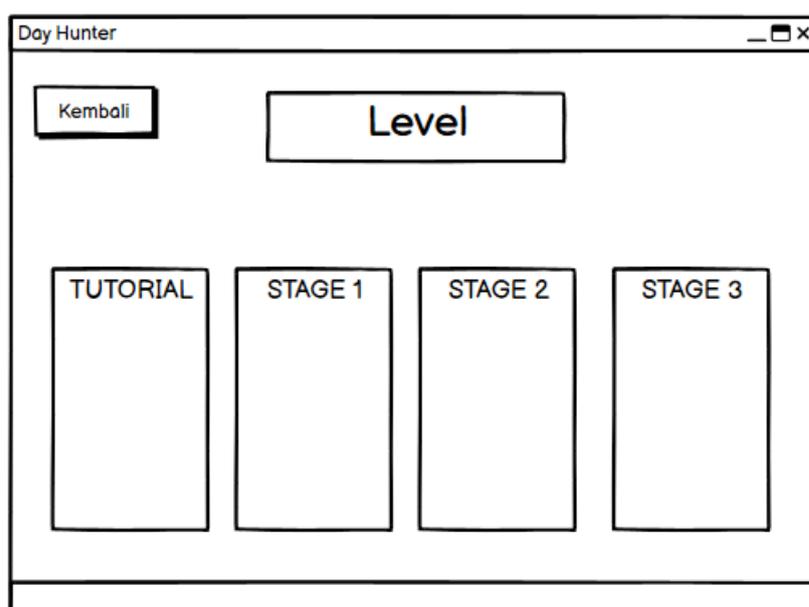


Gambar 4.6 Rancangan Menu Awal

Dari gambar 4.6 dapat dilihat terdapat tiga menu, yaitu bantuan, bermain, dan *exit*. Menu bantuan akan menunjukkan bagaimana cara bermain, Menu bermain akan menunjukkan ke jendela level yang terdiri dari 3 *stage*, dan Menu *exit* akan mengeluarkan dari game yang sedang dimulai.

4.4.2 Perancangan Menu Level

Dalam game edukasi yang dirancang, terdapat sebuah menu level yang dimana pemain dapat memilih level yang ingin dimainkan, dan tentunya setiap level memiliki tingkat kesulahan yang berbeda.



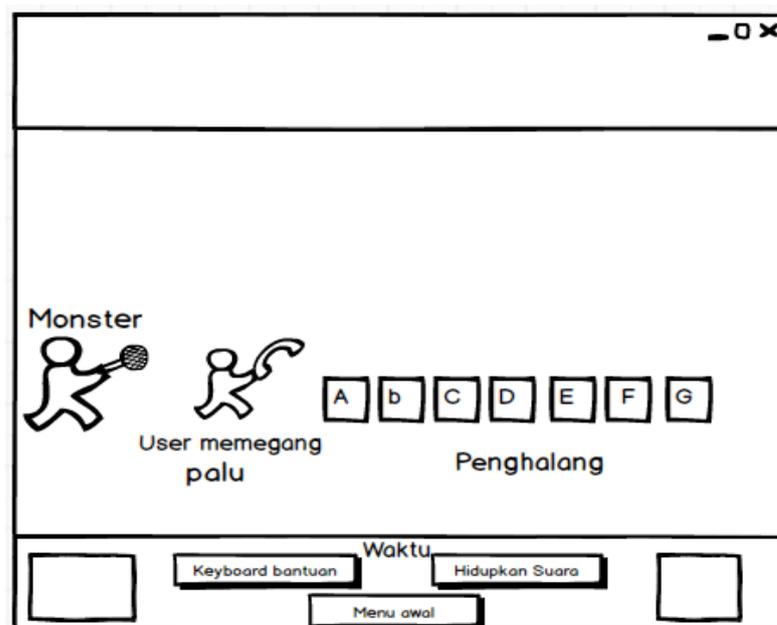
Gambar 4.7 Rancangan Menu Level

Terdapat 3 *stage* berbeda pada game ini. *Stage 1* memiliki tingkat kesulitan kata yang tidak cukup sulit namun memiliki teks yang panjang-panjang yang harus diselesaikan dalam waktu 10 menit. Pada *stage 2*, tingkat kesulitan kata menjadi menengah karena diikuti dengan istilah-istilah dan masih harus diselesaikan selama 10 menit. Pada *stage 3*, tingkat kesulitan kata menjadi sulit

karena sedikit diikuti istilah asing dan simbol-simbol, dan harus diselesaikan dalam 10 menit juga.

4.4.3 Perancangan Permainan

Dalam Game Edukasi ini dibuatlah sebuah game mengetik yang didalam permainan tersebut, sistem akan menampilkan kata-kata atau kalimat sesuai level, stage dan waktu yang harus di ketik agar *user* selamat dari kejaran *monster*, rancangan permainan ini juga menampilkan karakter karakter yang dapat bergerak.

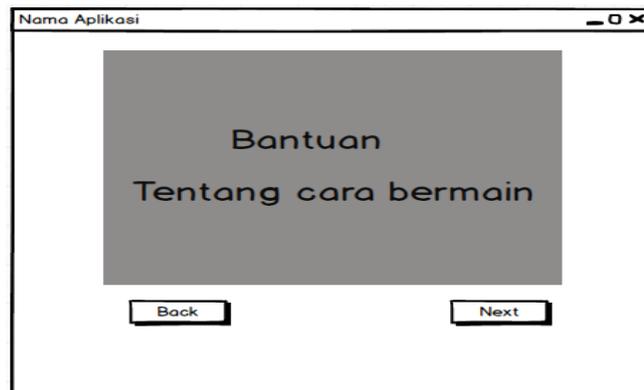


Gambar 4.8 Rancangan Permainan Stage 1

Dari gambar 4.8 dapat dilihat bahwa game dimulai dengan adanya monster di ujung belakang yang mengejar user memegang palu untuk menghancurkan kotak-kotak yang ada. Disebelah kiri ada text box yang akan memunculkan kalimat untuk diketik dan disebelah kanan ada text box untuk tempat mengetik kalimatnya.

4.4.4 Perancangan Bantuan

Dalam game edukasi yang dirancang, ditampilkan tentang rancangan bantuan yang berguna untuk *user* dalam memainkan *game*. Rancangan Bantuan ini berisi tentang bagaimana cara permainan ini berlangsung.

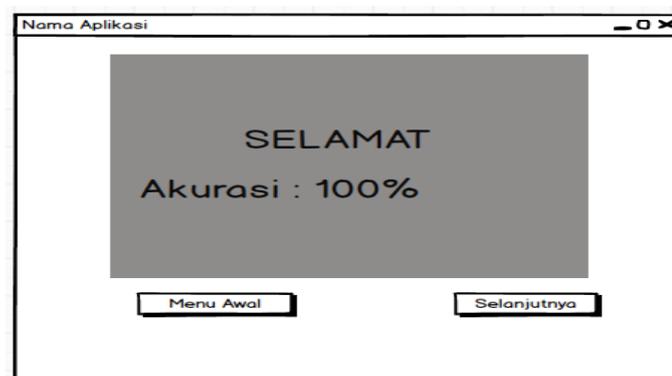


Gambar 4.9 Rancangan Bantuan

Pada menu bantuan terdapat cara-cara ataupun petunjuk bagaimana memainkan game tersebut. Yang terdiri dari letak-letak dan fungsi menu pada satge yang akan dimainkan, sehingga tidak sulit untuk mengetahui bagaimana cara bermain.

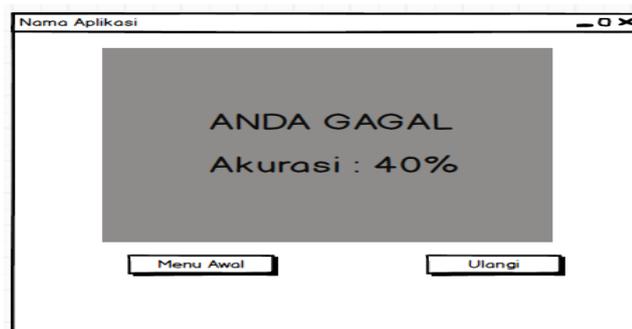
4.4.5 Perancangan Score

Dalam game edukasi yang dirancang ini, ditampilkan tentang rancangan *score* yang berguna untuk mengetahui nilai atau *score* dalam *game*.



Gambar 4.10 Rancangan Score Menang

Dalam game ini, nilai akan diberikan sesuai dengan berapa banyak yang dapat diketikkan dalam waktu 10 menit. Seperti contoh pada gambar 4.11, nilai tersebut dapat diberikan ketika pemain dapat mencapai kalimat terakhir pada game tersebut tanpa membuat kesalahan. Jikapun membuat kesalahan, maka teks tersebut tidak dapat berpindah pada teks selanjutnya jika tidak diperbaiki lagi.



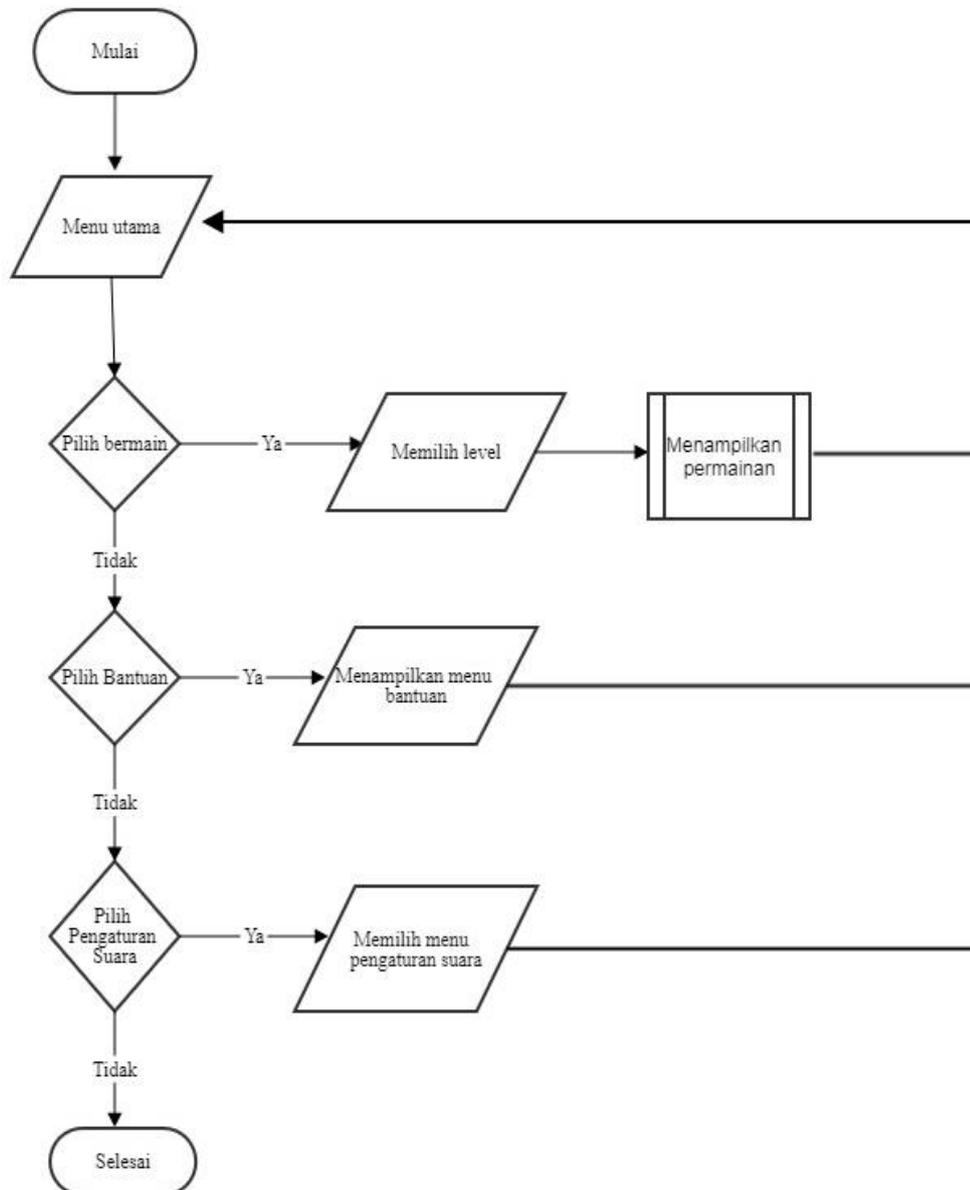
Gambar 4.11 Rancangan Score Kalah

Pada gambar 4.12, nilai tersebut dapat keluar sesuai dengan jumlah baris yang diketikkan, yaitu 4 baris sampai waktu berakhir. Waktu berakhir dan nilai yang dikeluarkan adalah 40% yang berarti hanya itulah yang dapat diketikkan dalam waktu 10 menit.

4.5 RANCANGAN ALGORITMA PROGRAM

Untuk menggambarkan algoritma program, dilakukan dengan menggunakan alat bantu berupa *flowchart*. *Flowchart* adalah suatu bagian yang menggambarkan arus logika dari data yang akan diproses ke dalam suatu program dari awal sampai akhir. Berikut ini *flowchart-flowchart* program yang dibuat :

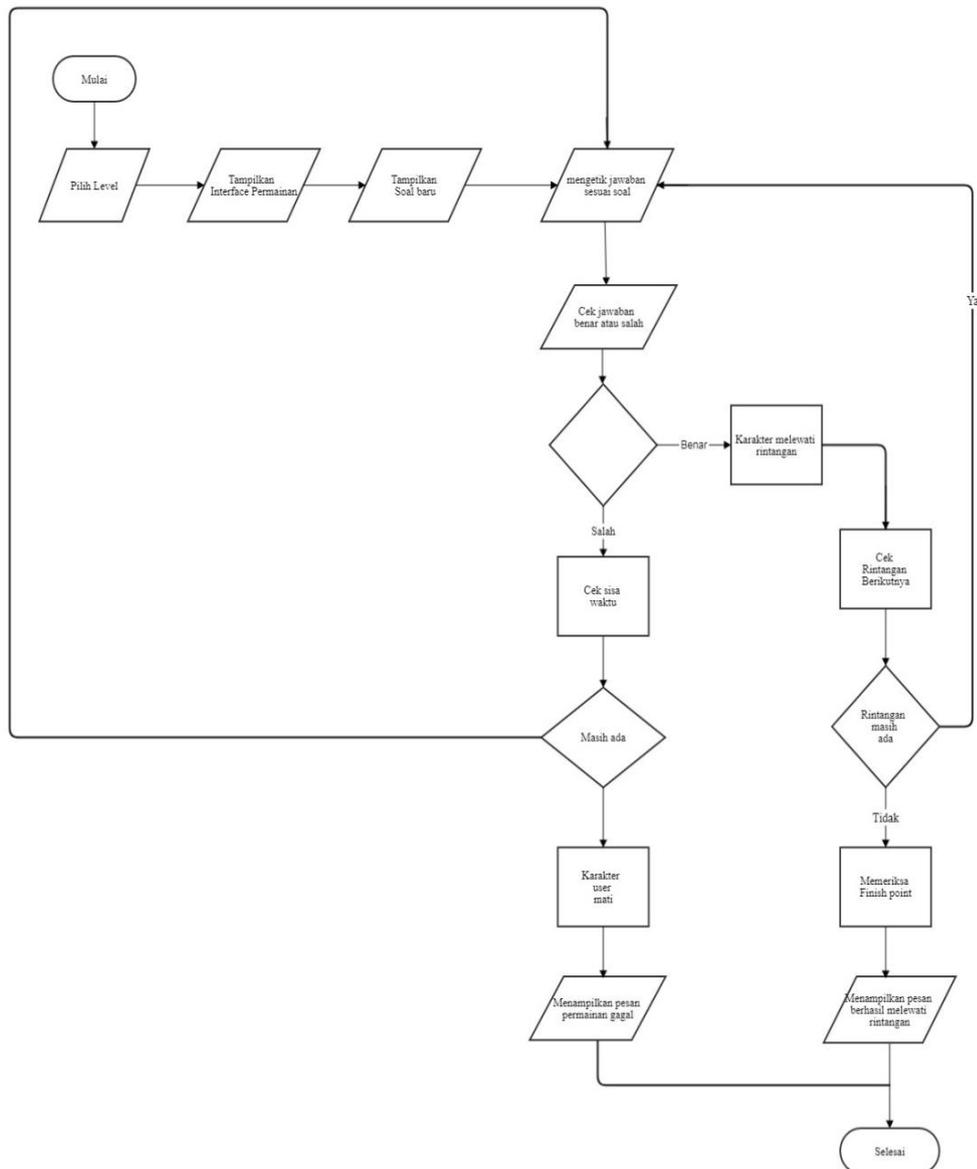
1. Flowchart Diagram keseluruhan program



Gambar 4.12 Flowchart keseluruhan program

User dapat memilih beberapa pilihan yang terdapat pada menu utama, pilihan tersebut yaitu pilihan bermain, bantuan dan pengaturan suara. Pada saat user memilih bermain, user akan di perlihatkan lagi menu level yang dimana dapat dipilih untuk di mainkan.

2. Flowchart Main



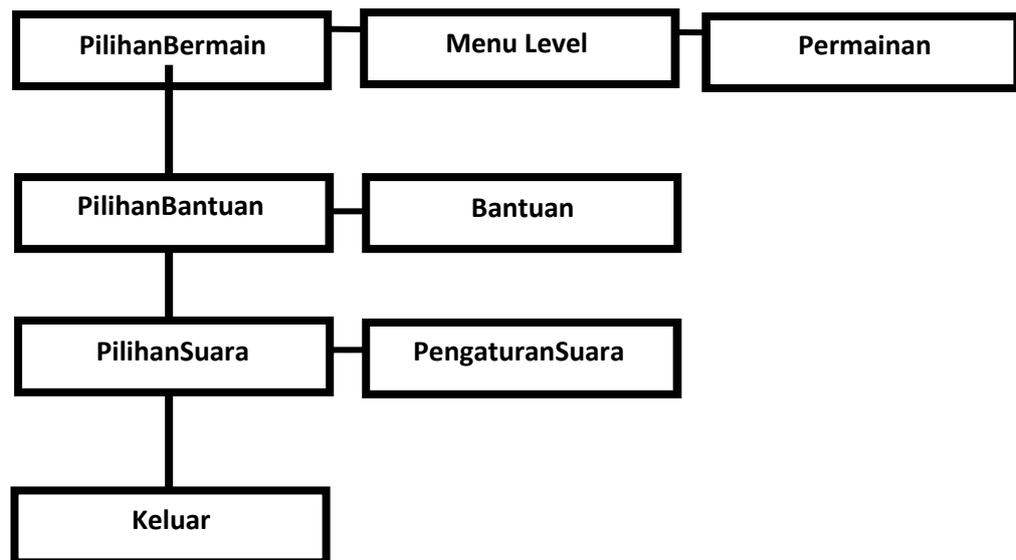
Gambar 4.13 Flowchart main

User dapat memilih level yang Dimana setiap level memiliki Tingkat kesusahan yang berbeda. Pada saat user memilih level maka *system* akan menampilkan *interface* permainan beserta tampilan soal. *User* diharuskan untuk menyesuaikan jawaban dengan soal yang di beri. Jika jawaban dan soal sesuai

maka karakter akan melewati sebuah rintangan. User akan dinyatakan menang jika karakter mencapai garis finish. Di dalam permainan terdapat musuh atau waktu yang digunakan sebagai parameter yang Dimana jika karakter user tersentuh oleh musuh atau waktu habis dalam menjawab soal maka user dinyatakan kalah.

4.6 PERANCANGAN STRUKTUR PROGRAM KESELURUHAN

Berikut Struktur Keseluruhan dari program yang telah dibuat :



Gambar 4.14 Struktur Keseluruhan Program