BAB III

ANALISIS DAN PERANCANGAN

III.1 Analisis

III.1.1 Analisis Sistem

Pada tahap ini dilakukan analisis terhadap sistem yang akan dibuat. Analisis sistem terdiri dari beberapa tahapan diantaranya, analisis masalah, analisis *aplikasi* sejenis, analisis kebutuhan non-fungsional dan analisis kebutuhan fungsional.

III.1.2 Analisis Masalah

Pada tahap analisis masalah ini akan menjelaskan masalah yang telah dijelaskan pada bab sebelumnya. Masalah yang ada saat ini untuk pemebelajaran *tenses* bahasa Inggris di Sekolah Menengah Atas (SMA) materi yang diajarkan mengenai *tenses* bahasa Inggris di sekolah, tidak mengajarkan seluruh 16 *tenses* yang ada. Hal tersebut dikarenakan waktu pada kegiatan belajar mengajar saat pelajaran bahasa Inggris hanya sekilas padahal, *tenses* merupakan salah satu *grammar* yang sangat penting dipelajari saat seseorang akan berbicara atau menuliskan bahasa Inggris dengan baik dan benar.

Pada analisis masalah ini, akan dibahasa juga mengenai identifikasi terhadap aplikasi sejenis. Aplikasi yang menjadi acuan adalah aplikasi pembelajaran bahasa Inggris.

Aplikasi pembelajaran bahasa Inggris merupakan aplikasi untuk membantu seseorang mendapatkan materi bahasa Inggris. Tujuan dari aplikasi tersebut adalah memberikan pelajaran yang memudahkan *user* untuk memahami materi yang ada dan menyenangkan. Beberapa contoh dari aplikasi pembelajaran bahasa Inggris adalah Rosetta Stone, My Spelling Tests, dan Daily English Exercise.

Aplikasi edukasi yang akan dibangun merupakan aplikasi pembelajaran bahasa Inggris. Aplikasi sejenis yang menjadi acuan adalah English Grammar dan Rosetta Stone pada sesi *speaking practice* dengan inputan suara sebagai jawaban soal. Sesi *speaking practice* dipilih karena pada sesi ini *user* akan dilatih untuk mengucapkan salah satu kosakata bahasa Inggris dengan benar.

III.1.3 Analisis Aplikasi Sejenis

Pada tahap ini akan dilakukan analisis terhadap aplikasi sejenis. Analisis aplikasi sejenis dilakukan dengan metode observasi terhadap aplikasi-aplikasi sejenis.

III.1.3.1 English Grammar

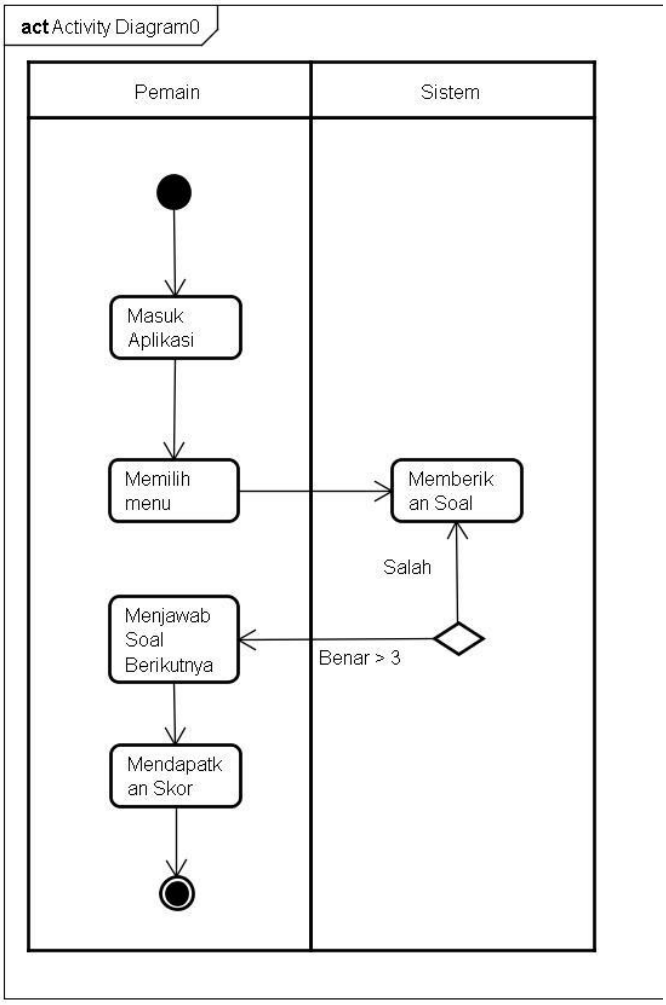
English Grammar merupakan salah satu aplikasi pembelajaran bahasa Inggris. English Grammar berisikan soal-soal materi *grammar* yang terdiri dari beberapa Segmen pembelajaran untuk mengukur kemampuan *user.*

1. Aplikasi English Grammar

Aplikasi English Grammar terdiri dari enam Segmen yaitu, *beginner*, *elementary*, *pre-intermediate*, *intermediate*, *upper-intermediate*, dan *advance*. Setiap Segmen terdiri dari 18 soal dengan tipe soal pilihan ganda. Soal berupa kalimat tidak lengkap yang telah dihilangkan salah satu to be dan kata kerja yang pas berdasarkan kalimatnya. Pemain harus menjawab sesuai pilihan ganda dengan *grammar* yang tepat. Aplikasi ini memiliki kesulitan ditiap Segmennya yaitu Segmen *beginner* pemain harus menjawab penggunaan to be pada kalimat, Segmen *elementary* menentukan *irregular verb* pada kalimat, Segmen *pre-intermediate* pemain harus menjawab kata kerja berdasarkan kalimat, Segmen *upper-intermediate* pemain harus menjawab soal berdasarkan modals, dan Segmen *advance* pemain harus menjawab soal berdasarkan kata penghubung kalimat.

2. Alur Sistem English Grammar

Alur sistem English Grammar merupakan gambaran alur kerja dari aplikasi English Grammar. Alur sistem ini akan dijelaskan bagaimana *user* memainkan aplikasi ini hingga mencapai tujuan seperti terlihat pada Gambar III.1.



Gamabar III.1 Activity Diagram English Grammar

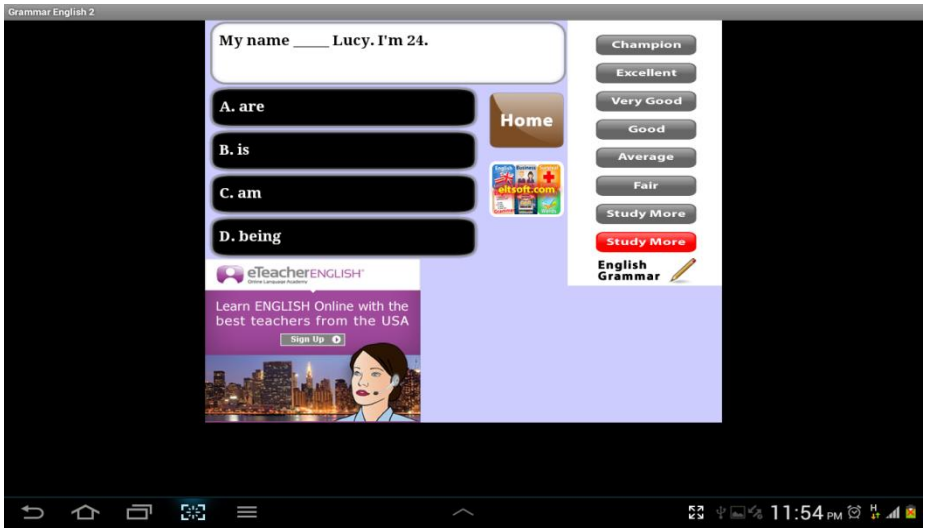
3. Tampilan Aplikasi English Grammar

Pada aplikasi English Grammar terdapat dua tampilan utama yaitu tampilan *home* dan tampilan *quiz.* Aplikasi English Grammar terdiri dari beberapa menu untuk memilih Segmen dan status penilaian kemampuan pemain.



Gambar III.2 Tampilan Menu Utama (*home)* English Grammar

Setelah memilih salah satu Segmen, user akan dihadapkan pada soal-soal pilihan ganda hingga mencapai penilaian/*score* dari Study More hingga Champion seperti pada Gambar III.3.



Gambar III.3 Tampilan Halaman Aplikasi English Grammar

III.1.3.2 Rosetta Stone English

Rosetta Stone adalah sebuah aplikasi untuk mempelajari bahasa Inggris yang memiliki beberapa materi pembelajaran. Materi-materi pembelajaran (*Core*

Lessons) diantaranya adalah *pronunciation*, *vocabulary*, *grammar, listening and reading*, *reading*, *writing*, *listening*, dan *speaking.* Rosetta Stone English memiliki beberapa latihan soal dan Segmen berdasarkan kesulitan pada materi ditiap unit.

1. *Core Lesson Pronunciation*

   Latihan soal pada *pronunciation, user* harus menjawab salah satu kata sesuai dengan gambar dan contoh suara yang diberikan.
2. *Core Lesson Vocabulary*

   Latihan soal pada *vocabulary*, *user* menentukan kata yang sesuai dengan contoh gambar yang diberikan.
3. *Core Lesson Grammar*

   Pada latihan soal ini, *user* mendapatkan sebuah kalimat tidak lengkap yang harus dilengkapi berdasarkan *to be* dan *tenses.*
4. *Core Lesson Reading*

   Pada latihan soal ini, *user* harus bisa membaca dan mengucapaknya berdsarkan contoh suara dan gambar yang diberikan.
5. *Core Lesson Writing*

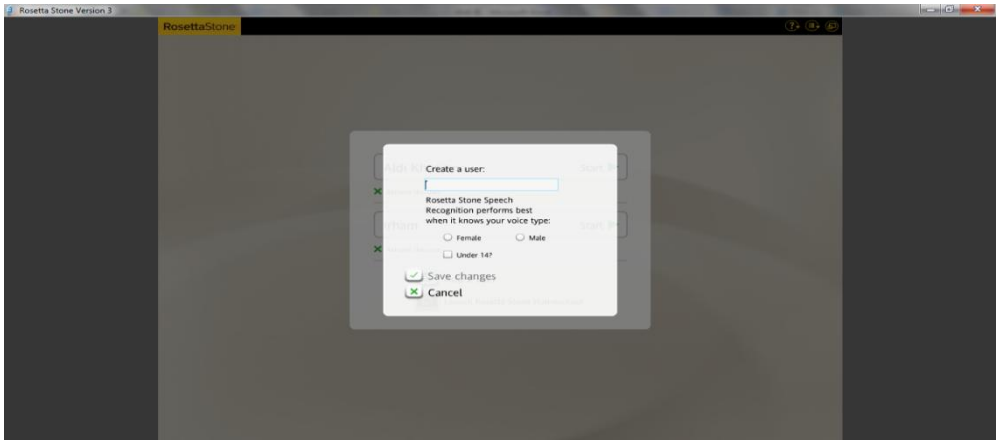
   Pada latihan soal ini, *user* harus menuliskan kata yang telah diucapkan berdasarkan contoh gambarnya.
6. *Core Lesson Listening*

   Pada latihan soal *Core Lesson Listening*, *user* hanya mendapatkan contoh suara dan harus menentukan gambar yang sesuai.
7. *Core Lesson Speaking*

   Pada latihan soal *Core Lesson Speaking,* latihan yang diberikan hampir sama dengan latihan-latihan soal yang menginputkan suara sebagai jawaban. Tetapi pada latihan soal ini, nada suara berdasarkan kondisi sangat diperhatikan.

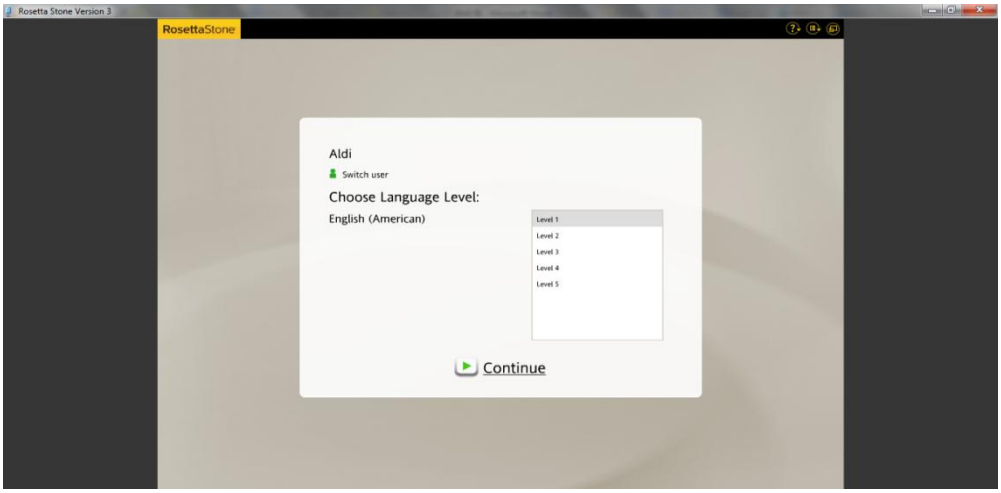
1. Tampilan Aplikasi Rosetta Stone

Aplikasi Rosetta Stone memiliki tampilan menu awal berupa input *user* untuk membuat *user* baru. Tampilan utama lainnya adalah tampilan untuk menentukan Segmen, daftar *Core Lesson*, dan *unit.*



Gambar III.4 Tampilan Awal Rosetta Stone Untuk Input *User* Baru

Setelah masuk dan memasukkan nama seperti pada gambar sebelumnya, user akan langsung memeilih Segmen untuk menentukan tingkat kesulitan seperti pada Gambar III.5.



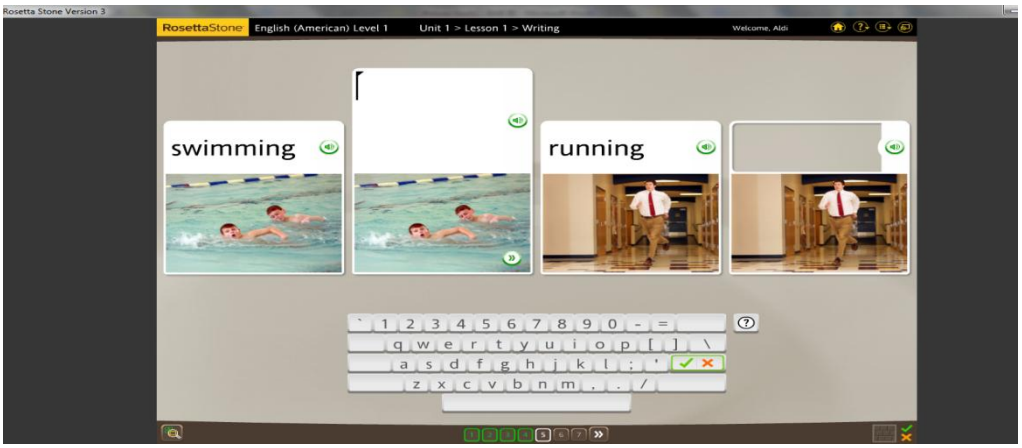
Gambar III.5 Tampilan Pilih Segmen Bahasa di Aplikasi Rosetta Stone

Rosetta Stone pada tampilan awal untuk memulai pelajaran, terdapat beberapa menu. User bisa memilih secara acak agar bisa mendapatkan soal-soal yang lebih variatif seperti pada Gambar III.6.



Gambar III.6 Tampilan Utama Tiap Sesi Unit dan Core Lesson

Salah satu soal *core Lesson* sesi *writing* bisa dilihat pada Gambar III.7 dimana user harus menjawab dengan menuliskan kata seperti pada soal gambar dan contoh penulisannya.



Gambar III.7 Tampilan Latihan Soal Pada Aplikasi Rosetta Stone

III.1.4 Analisis Aplikasi Pembelajaran Berbasis Komputer

1. Pengenalan Jawaban

Pada pengenalan jawaban ini, pemain akan diberikan beberapa soal pada sesi latihan dan ujian. Pertama, pemain akan diberikan bahan materi yang akan dipelajari pada tenses pertama yaitu Present Tenses, kedua yaitu Past tenses, ketiga yaitu Future tenses, dan yang terakhir adalah Past Future tenses. Setelah pemain memahami materi maka akan diberikan latihan soal berdasarkan tenses

yang telah dipelajari. Soal-soal yang telah diisi dengan benar pada latihan soal, maka pemain akan melanjutkan ke sesi ujian dimana pemain akan diberikan soal lebih banyak dengan tingkat kesulitan yang lebih tinggi dari latihan soal.

III.1.4.1 Gameplay

Berikut ini gameplay dari aplikasi pembelajaran tenses bahasa Inggris yang terdiri dari 4 Segmen.

1. Gameplay Segmen 1

Gameplay Segmen 1 pada aplikasi pembelajaran *tenses* bahasa Inggris bisa dilihat pada tabel III.1

Tabel III.1 Gameplay Segmen 1 Aplikasi Pembelajaran Tenses Bahasa Inggris

No	Gameplay
1	Pemain akan diberikan materi Segmen 1 (Present Tense)
2	Pemain membaca dan mempelajari sebelum pemberian latihan soal
3	Ketika siap dimulai, pemain akan diberikan latihan soal dan setelah bisa menjawab semua soal, lalu pemain akan mengikuti ujian
4	Pemain akan diberikan 5 soal pada latihan dan 10 soal pada ujian.

2. Gameplay Segmen 2

Gameplay Segmen 2 pada aplikasi pembeajaran *tenses* bahasa Inggris bisa dilihat pada tabel III.2

Tabel III.2 Gameplay Segmen 2 Aplikasi Pembelajaran Tenses Bahasa Inggris

No	Gameplay
1	Pemain akan diberikan materi Segmen 2 (Past Tense)

2	Pemain membaca dan mempelajari sebelum pemberian latihan soal
3	Ketika siap dimulai, pemain akan diberikan latihan soal dan setelah bisa menjawab semua soal, lalu pemain akan mengikuti ujian
4	Pemain akan diberikan 5 soal pada latihan dan 10 soal pada ujian.

3. Gameplay Segmen 3

Gameplay Segmen 3 pada aplikasi pembelajaran *tenses* bahasa Inggris bisa dilihat pada tabel III.3

Tabel III.3 Gameplay Segmen 3 Aplikasi Pembelajaran Tenses Bahasa Inggris

No	Gameplay
1	Pemain akan diberikan materi Segmen 3 (Future Tense)
2	Pemain membaca dan mempelajari sebelum pemberian latihan soal
3	Ketika siap dimulai, pemain akan diberikan latihan soal dan setelah bisa menjawab semua soal, lalu pemain akan mengikuti ujian
4	Pemain akan diberikan 5 soal pada latihan dan 10 soal pada ujian.

4. Gameplay Segmen 4

Gameplay Segmen 4 pada aplikasi pembelajaran *tenses* bahasa Inggris bisa dilihat pada tabel III.4

Tabel III.4 Gameplay Segmen 4 aplikasi Pembelajaran Tenses Bahasa Inggris

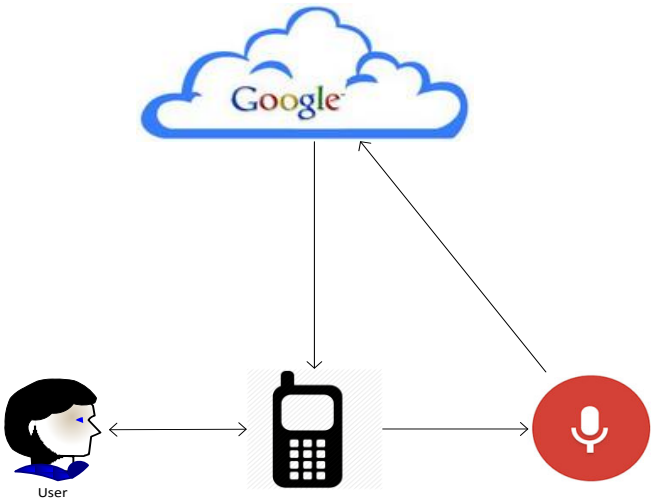
No	Gameplay
1	Pemain akan diberikan materi Segmen 4 (Past Future Tense)
2	Pemain membaca dan mempelajari sebelum pemberian latihan soal

3	Ketika siap dimulai, pemain akan diberikan latihan soal dan setelah bisa menjawab semua soal, lalu pemain akan mengikuti ujian
4	Pemain akan diberikan 5 soal pada latihan dan 10 soal pada ujian.

III.1.5 Google *Speech*

Google *speech* merupakan salah satu fitur yang terdapat pada aplikasi yang dibangun. Google *speech* digunakan saat pemain akan menjawab semua jawaban baik dari sesi latihan maupun ujian. Pemain akan dilatih kemampuan *speaking* agar bisa mengucapakan pronunciation dengan benar. Google *speech* memiliki kemampuan untuk mendeteksi apakah pengucapan kata dalam bahasa Inggris benar atau salah.

Google *speech* sendiri merupakan salah satu aplikasi dari google yang mampu mendeteksi suara yang diucapkan user lalu memprosesnya untuk diolah menjadi sebuah perintah atau keluaran berupa teks. Berikut ini adalah skema infrastruktur dari google *speech*



Gambar III.8 Infrastruktur Google *Speech*

III.1.6. Analisis Algoritma

III.1.6.1. Menentukan Soal Ujian dengan Algoritma Genetika

Pada sub bab ini akan dijelaskan algortima genetika yang diaplikasikan untuk menentukan soal ujian yang akan diterima pemain setelah mendapatkan skor pada latihan soal. Soal pada ujian disesuaikan dengan hasil pada sesi latihan. Pemain mendapatkan 5 soal pada sesi latihan dengan nilai yang bagus, dalam artian pemain bias menjawab benar >=3 maka akan mendapatkan 10 soal, selain itu mendapat 5 soal.. Bersdasarkan ketentuan soal yang akan diterima untuk ujian dari hasil latihan soal, berikut ini adalah kemungkinan soal-soal pada *Present Tense:*

Tabel III.5 Soal-soal *to be Present Tense*

ID	Soal
PT1	I am walking to school
PT2	You are sleeping on the floor
PT3	She is swimming in swimming pool
PT4	Frank has drunk two orange juices
PT5	I have written this letter
PT6	Monita has been living there since 4 years ago
PT7	He is reading three novels everyday
PT8	My mother has cooked for dinner
PT9	Andres is singing on stage
PT10	You have been watering these plants

Tabel III.6 Soal-soal *Verb Present Tense*

ID	Soal
PV1	He ___ to school (Go)
PV2	I have ___ this book (V3 read)
PV3	I have been ___ to go now (Ving think)
PV4	Rio has ___ starfall (V3 see)
PV5	My father is ___ newspaper (Ving read)

Penjelasan mengenai langkah-langkah untuk menentukan soal sesuai ketentuan yang pertama adalah:

1. Pembentukan Kromosom

   Karena yang dicari adalah jumlah kuantitas soal pada ujian maka dibuat persamaan a+b=10 untuk pemain yang dapat menjawab soal latihan >=3 dan a+b=5 untuk pemain yang dapat menjawab soal latihan <=2. a dan b dijadikan gen-gen pembentuk kromosom. Batasan nilai a dan b adalah 10.
2. Inisialisasi

   Proses inisialisasi dilakukan dengan cara memberikan nilai awal gen-gen dengan nilai acak sesuai batasan yang telah ditentukan. Misalkan jumlah populasi adalah 5, maka:

   Chromosome[1]=[a,b]=[6,7]

   Chromosome [2]=[a,b]=[4,8]

   Chromosome [3]=[a,b]=[5,3]

   Chromosome [4]=[a,b]=[7,7]

   Chromosome [5]=[a,b]=[6,1]
3. Evaluasi Kromosom

   Permasalahan yang akan diselesaikan adalah nilai variable a dan b dengan persamaan a+b=10, fungsi objektifnya adalah (Chromosome)= |(a+b)-10|

   Hitung fungsi objektif dari kromosom yang telah dibangkitkan

Chromosome [1]=((6 + 7)-10)

=(13-10)

=3

Chromosome [2]=((4 + 8)-10)

=(12-10)

=2

Chromosome [3]=((5+3)-10)

=(8-10)

= -2

Chromosome [4]=((7 + 7)-10)

=(14-10)

=4

Chromosome [5]=((6+1)-10)

=(7-10)

=3

Rata-rata dari fungsi objektif adalah = (3+2+2+4+3)/5

=2,8

4. Seleksi Kromosom

Proses seleksi dilakukan dengan cara membuat kromosom yang mempunyai fungsi objek terkecil memiliki kemungkinan terpilih yg besar atau memiliki nilai probabilitas yang tinggi, maka digunakan fungsi fitness.

Fungsi fitness: Fitness = 1 / (fungsi_objektif[n]+1)

Fitness[1] = 1/(fungsi objektif[1]+1)

= 1/4

= 0.25

Fitness[2] = 1/(fungsi objektif[2]+1)

= 1/3

= 0.33

Fitness[3] = 1/(fungsi objektif[3]+1)

= 1/3

= 0.33

Fitness[4] = 1/(fungsi objektif[4]+1)

= 1/5

= 0.2

Fitness[5] = 1/(fungsi objektif[5]+1)

= 1/4

= 0.25

Total fitness= 0.25+0.33+0.33+0.2+0.25

= 1.36

Rumus untuk mencari probabilitas P[i]=fitness[i]/total_fitness

P[1] = 0.25/1.36 = 0.18

P[2] = 0.33/1.36 = 0.24

P[3] = 0.33/1.36 = 0.24

P[4] = 0.2/1.36 = 0.14

P[5] = 0.25/1.36 = 0.18

Berdasarkan probabilitas yang telah dihitung, kromosom ke 2 mempunyai nilai fitness paling besar, maka kromosom tersebut memiliki probabilitas untuk terpilih pada generasi selanjutnya yang lebih besar dari kromosom lainnya.

Selanjutnya proses seleksi menggunakan *Roulette Wheel*, sebelum itu nilai nilai kumulatif probabilitasnya harus dicari terlebih dahulu.

C[1] = 0.18

C[2] = 0.18 + 0.24 = 0.42

C[3] = 0.18 + 0.24 + 0.24 = 0.66

C[4] = 0.18 + 0.24 + 0.24 + 0.14 = 0.8

C[5] = 0.18 + 0.24 + 0.24 + 0.14 + 0.18 = 1

Setelah dihitung probabilitas kumulatifnya, maka seleksi menggunakan *Roulette Wheel* dapat dilakukan. Prosesnya adalah dengan membangkitkan bilangan acak R dalam Range 0 - 1. Jika R[k] < C[1] maka pilik kromosom 1 sebagai induk, selain itu pilih kromosom ke-k dengan syarat C[k-1] < R < C[k]. Putar *roulette wheel* sebanyak jumlah populasi yaitu 5 kali dan pada setiap putaran, pilih satu kromosom untuk populasi baru, missal:

R[1] = 0.321

R[2] = 0.251

R[3] = 0.235

R[4] = 0.132

R[5] = 0.422

Angka acak pertama R[3] lebih kecil dari C[1] dan C[2]. Maka pilih kromosom[3] sebagai kromosom untuk populasi baru. Dari bilangan acak yang telah dibangkitkan tadi maka populasi kromosom baru hasil proses seleksi adalah:

Kromosom[1] = Kromosom [3]

Kromosom[2] = Kromosom [3]

Kromosom[3] = Kromosom [3]

Kromosom[4] = Kromosom [1]

Kromosom[5] = Kromosom [3]

Kromosom baru hasil proses seleksi:

Kromosom [1] = [a,b] = [5,3]

Kromosom [2] = [a,b] = [5,3]

Kromosom [3] = [a,b] = [5,3]

Kromosom [4] = [a,b] = [6,7]

Kromosom [5] = [a,b] = [5,3]

5. Crossover

Setelah proses seleksi maka proses selanjutnya adalah proses crossover. Metode yang digunakan adala one-cut point, yaitu memilih secara acak satu posisi dalam chromosome induk kemudian saling menukar gen. chromosome yang dijadikan induk dipilih secara acak dan jumlah chromosome yang mengalami crossover dipengaruhi oleh parameter crossover_rate (pc).

Pseudocode untuk proses crossover adalah sebagai berikut:

```
Begin
k ← 0;
while (k<populasi) do
R[k] ← random (0-1);
if (R[k]< pc) then
select Chromosome[k] as parent;
end;
k=k+1;
end;
end;
```

Misalkan ditentukan crossover probability adalah sebesar 25%, maka diharapkan dalam satu generasi ada 50% Chromosome (+/- 2 chromosome) dari satu generasi mengalami crossover. Prosesnya adalah sebagai berikut:
Pertama dibangkitkan bilangan acak R sebanyak jumlah populasi

R[1] = 0.324
R[2] = 0.035
R[3] = 0.522
R[4] = 0.223
R[5] = 0.411
Maka Chromosome ke k akan dipilih sebagai induk jika R[k] < pc, dari bilangan acak R diatas maka yang dijadikan induk adalah Chromosome[2] dan Chromosome[4].
Setelah melakukan pemilihan induk proses, proses selanjutnya adalah menentukan posisi crossover. Menetukan posisi crossover dilakukan dengan cara membangkitkan bilangan acak dengan batasan 1 sampai (panjang chromosome-1). Misalkan didapatkan posisi crossover adalah 1 maka chromosome induk akan dipotong mulai gen ke-1 kemudian potongan gen tersebut saling ditukar antar induk.
Chromosome[2] >< Chromosome[4]
Chromosome[4] >< Chromosome[2]
Posisi cut-point crossover dipilih menggunakan bilangan acak 1-2 sebanyak jumlah crossover yang terjadi, missal
C[1] = 1
C[2] = 1

Offspring[1] = chromosome[2] >< chromosome[4]
= [3,6] >< [6,7]

= [3,7]

Offspring[2] = chromosome[4] >< chromosome[2]

= [6,7] >< [3,6]

= [6,6]

Setelah mealui proses crossover, maka populasi chromosome menjadi:

Kromosom [1] = [a,b] = [5,3]

Kromosom [2] = [a,b] = [3,7]

Kromosom [3] = [a,b] = [5,3]

Kromosom [4] = [a,b] = [6,6]

Kromosom [5] = [a,b] = [5,3]

6. Mutasi

Jumlah kromosom yang mengalami mutasi dalam satu populasi ditentukan oleh parameter mutation_rate. Proses mutasi dilakukan dengan cara mengganti satu gen yang terpilih secara acak dengan suatu nilai baru yang didapat secara acak. Prosesnya adalah hitung dulu panjang total gen yang ada pada satu populasi. Dalam kasus panjang total gen adalah:

total_gen = jumlah gen dalam kromosom * jumlah populasi

= 2 x 5 = 10

Untuk memilih posisi gen yang mengalami mutasi dilakukan dengan cara membangkitkan bilangan integer acak antara 1 sampai total_gen, yaitu 1 sampai 10. Jika bilangan acak yang dibangkitkan lebih kecil daripada variabel mutation_rate (ρm) maka pilih posisi tersebut sebagai sub-chromosome yang mengalami mutasi. Misalkan mutation_rate ditentukan 10% maka diharapakan ada 10% dari total gen yang mengalami populasi.

Jumlah mutasi = 0.1 * 10 = 1

Misalkan setelah dibangkitkan bilangan acak terpilih posisi gen 3 dan 7 yang mengalami mutasi. Dengan demikian yang akan mengalami mutasi adalah chromosome ke-3 gen nomor 1 dan Chromosome ke-4 gen nomor 1. Maka nilai gen pada posisi tersebut diganti dengan bilangan acak 0-10.

Kromosom [1] = [a,b] = [5,3]

Kromosom [2] = [a,b] = [4,7]

Kromosom [3] = [a,b] = [5,3]

Kromosom [4] = [a,b] = [4,6]

Kromosom [5] = [a,b] = [5,3]

Setelah proses mutasi maka telah diselesaikan satu iterasi dalam algoritma genetika atau disebut dengan satu generasi. Maka fungsi_objective setelah satu generasi adalah:

Chromosome [1] = [5;3]

Fungsi_objektif[1] = Abs((5+3)-10)

= 8-10 = -2

Chromosome [2] = [5;7]

Fungsi Objektif[2] = Abs((5+7)-10)

= 12-10 = 2

Chromosome [3] = [5,3]

Fungsi_objektif[3] = Abs((5+3)-10)

= 8-10 = -2

Chromosome [4] = [4,6]

Fungsi_objektif[4] = Abs((4+6)-10)

= 10-10 = 0

Chromosome [5] = [5+3]

Fungsi_objektif[5] = Abs((5+3)-10)

= 8-10 = -2

Chromosome-chromosome ini akan mengalami proses yang sama seperti generasi sebelumnya yaitu proses evaluasi, seleksi dan mutasi yang kemudian akan menghasilkan chromosome-chromosome baru untuk generasi yang selanjutnya. Proses ini akan berulang sampai sejumlah generasi yang telah ditetapkan sebelumnya.

Setelah beberapa generasi didapatkan kromosom yang terbaik adalah:

Kromosom 4 [4,6]. Jika dikodekan pada peraturan pada soal-soal ujian yang akan diterima adalah 4 *to be* dan 6 *verb.*

Jika dikodekan maka: a=4; b=6

Jika dihitung terhadap persamaan maka: 4+6=10

III.1.7 Analisis Kebutuhan Non-Fungsional

Analisis kebutuhan non-fungsional merupakan analisis yang dibutuhkan untuk menentukan spesifikasi kebutuhan sistem. Spesifikasi ini juga meliputi elemen atau komponen-komponen apa saja yang dibutuhkan untuk sistem yang akan dibangun sampai dengan sistem tersebut diimplementasikan. Analisis kebutuhan non-fungsional juga merupakan spesifikasi masukan yang diperlukan sistem, keluaran yang akan dihasilkan sistem dan proses yang dibutuhkan untuk mengolah masukan sehingga menghasilkan suatu keluaran yang diinginkan. Pada analisis kebutuhan sistem non fungsional ini dijelaskan analisis kebutuhan perangkat lunak, analisis kebutuhan perangkat keras, dan analisis pengguna.

III.1.8 Analisis Kebutuhan Perangkat Keras (Developer)

Perangkat keras yang digunakan untuk pembuatan aplikasi pembelajaran *tenses* bahasa Inggris menggunakan spesifikasi sebgai berikut :

Tabel III.7 Spesifikasi Perangkat Keras (Developer)

Nama Perangkat	Spesifikasi
Processor	Dual core 2 GHz
Harddisk Drive	20 Gb
RAM	1.00 Gb

III.1.9 Analisis Kebutuhan Perangkat Lunak

Perangkat lunak yang digunakan dalam pembuatan aplikasi pembelajaran bahasa Inggris menggunakan software dan tools sebagai berikut :

Tabel III.8 Spesifikasi Perangkat Lunak (Developer)

Nama Perangkat Lunak	Spesifikasi
Eclipse	v21.0.1-543035
Database Manager	SQLite
Android	ICS 4.0.4

III.1.10 Analisis Kebutuhan Perangkat Lunak (User)

Perangkat lunak yang dipergunakan dalam implementasi aplikasi menggunakan spesifikasi minimum sebagai berikut :

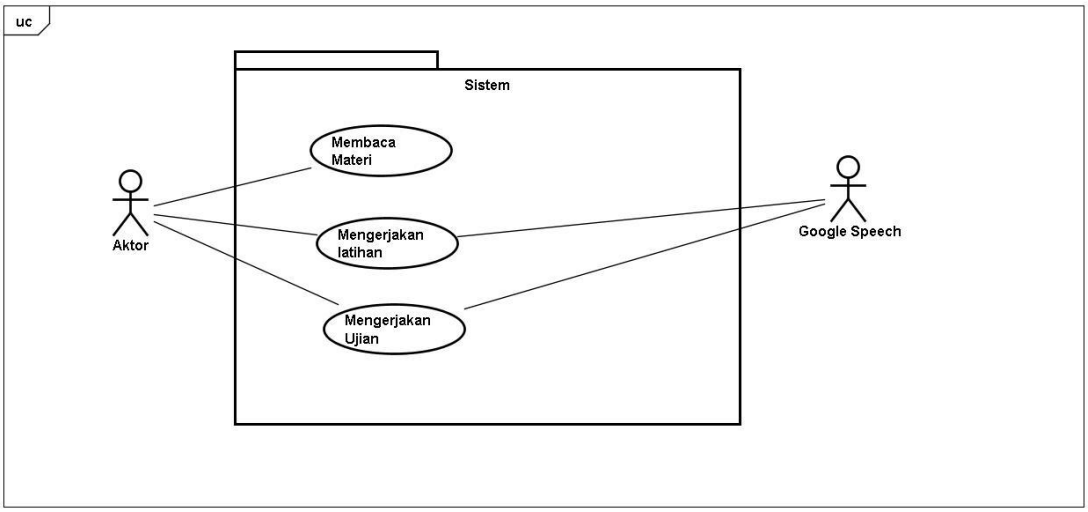
Tabel III.9 Spesifikasi Perangkat Lunak (User)

Nama Perangkat Lunak	Spesifikasi
Android	2.3 (Gingerbread)
Processor	600Mhz
Microphone	Internal / External Device

III.1.11 Analisis Kebutuhan fungsional

III.1.11.1 Diagram *Use Case*

Aplikasi terdiri dari satu aktor yaitu sebagai pemain dan aktor *speech recognition*. Pemain adalah orang yang berperan memainkan dan memilih menu-menu pada permainan.



Gambar III.9 Diagram *Use Case*

III.1.11.2 Skenario *Use Case*

Skenario pada diagram *use case* berisi mengenai alur kejadian "flow of event" use case utama. di dalamnya menjelaskan urutan interaksi antara aktor

dengan *use case* tersebut dari awal sampai akhir. Adapun *use case* yang terdapat pada aplikasi pemelajaran *tenses* bahasa Inggris ini adalah sebagai berikut.

1. Skenario *Use Case* Menerima Materi

Skenario *use case* Menerima Materi ini menjelaskan interaksi antara aktor pemain dengan *use case* saat menerima materi yang dijelaskan pada tabel III.8 berikut.

Tabel III.10 Skenario *Use Case* Menerima Materi

Identifikasi	
Nama	Menerima Materi
Tujuan	Mempelajari Materi
Aktor	Pemain
Skenario Utama	
Kondisi Awal	Menampilkan Rumus dari *Tenses*
Aksi Aktor	**Reaksi Sistem**
a. Masuk Mempelajari Materi	
	b. Menampilkan Materi *Tenses*
Kondisi Akhir	Menampilkan Materi *Tenses*

2. Skenario *Use Case* Mengerjakan Latihan

Skenario *Use Case* Mengerjakan Latihan ini menjelaskan interaksi antara aktor dengan *Use Case* Mengerjakan Latihan seperti terlihat pada Tabel III.9

Tabel III.11 Skenario *Use Case* Mengerjakan Soal Latihan

Identifikasi	
Nama	Mengerjakan Latihan

Tujuan	Menerima Latihan Soal
Aktor	Pemain
Skenario Utama	
Kondisi Awal	Memberikan Latihan Soal
Aksi Aktor	**Reaksi Sistem**
a. Mengerjakan Latihan Soal *tenses* bahasa Inggris mulai dari Segmen 1, Segmen 2, Segmen 3, sampai Segmen 4	
	b. Memberikan Latihan Soal
Kondisi Akhir	Menampilkan Penilaian Benar atau Salah

3. Skenario *Use Case* Mengerjakan Soal Ujian

Skenario *use case* Mengerjakan Soal ini menjelaskan interaksi antara aktor pemain dengan *use case* Mengerjakan Soal yang dijelaskan pada Tabel III.10 berikut: **Tabel III.12 Skenario *Use Case* Mengerjakan Soal Ujian**

Identifikasi	
Nama	Mengerjakan Soal Ujian
Tujuan	Memebrikan Soal Ujian
Aktor	Pemain
Skenario Utama	

Kondisi Awal	Menampilkan Soal Ujian	
Aksi Aktor		**Reaksi Sistem**
a. Menjawab Pertanyaan mulai dari Segmen 1, Segmen 2, Segmen 3, sampai Segmen 4		
		b. Menampilkan Jawaban Benar atau Salah
Kondisi Akhir	Memeberikan Penilaian dari Jawaban Pemain	

4. Skenario *Use Case Speech Recognition*

Skenario *use case* Score ini menjelaskan interaksi antara aktor pemain dengan *use case* Score yang dijelaskan pada Tabel III.11 berikut ini:

Tabel III.13 Skenario *Use Case Speech Recognition*

Identifikasi	
Nama	*Speech Recognition*
Tujuan	Menampilkan form untuk menjawab jawaban menggunakan suara
Aktor	Pemain
Skenario Utama	
Kondisi Awal	Pemain Menjawab Soal Ujian
Aksi Aktor	**Reaksi Sistem**

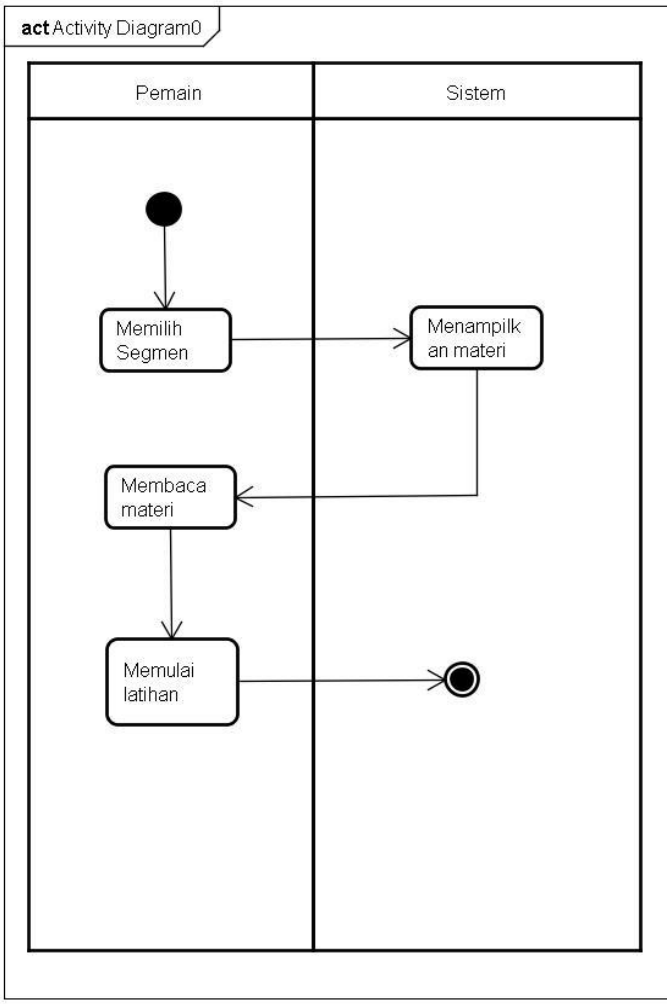
a. Pemain menjawab semua pertanyaan	
	b. Menampilkan form *speech recognition*
Kondisi Akhir	Menampilkan jawaban pemain

III.1.11.3 *Activity Diagram*

Activity Diagram adalah teknik untuk menggambarkan proses bisnis dan alur kerja. Diagram ini memainkan peran mirip sebuah diagram alir, tetapi perbedaan prinsip antara diagram ini dan notasi diagram alir adalah diagram ini mendukung behavior parallel. *Activity Diagram* menggambarkan alur aktifitas pada sistem yang sedang dirancang, bagaimana alur berawal, decision yang mungkin terjadi, dan bagaimana alur berakhir.

1. ***Activity Diagram* Menerima Materi**

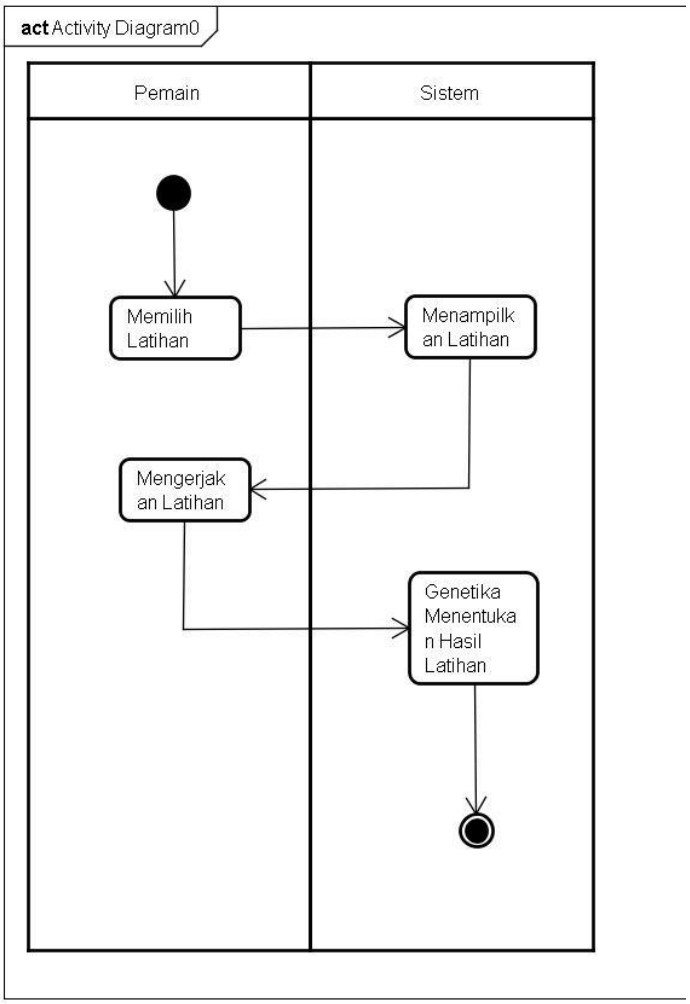
Aktivitas pemilihan Menerima Materi adalah aktivitas aktor saat menerima materi seperti terlihat pada gambar III.13



Gambar III.10 *Activity Diagram* Menerima Materi

2. *Activity Diagram* Mengerjakan Latihan

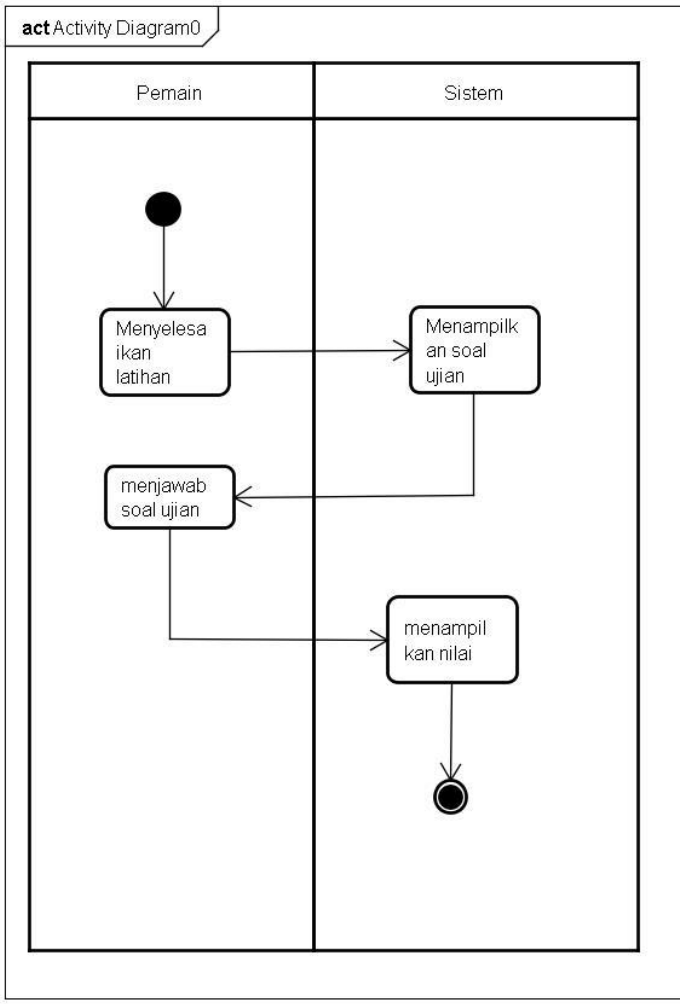
Aktivitas Mengerjakan Latihan akan menunjukkan proses pemain mengerjakan latihan soal yang diberikan. Aktivitas mengerjakan soal seperti terlihat pada gambar III.14



Gambar III.11 *Activity Diagram* Latihan Soal

3. *Activity Diagram* Mengerjakan Ujian

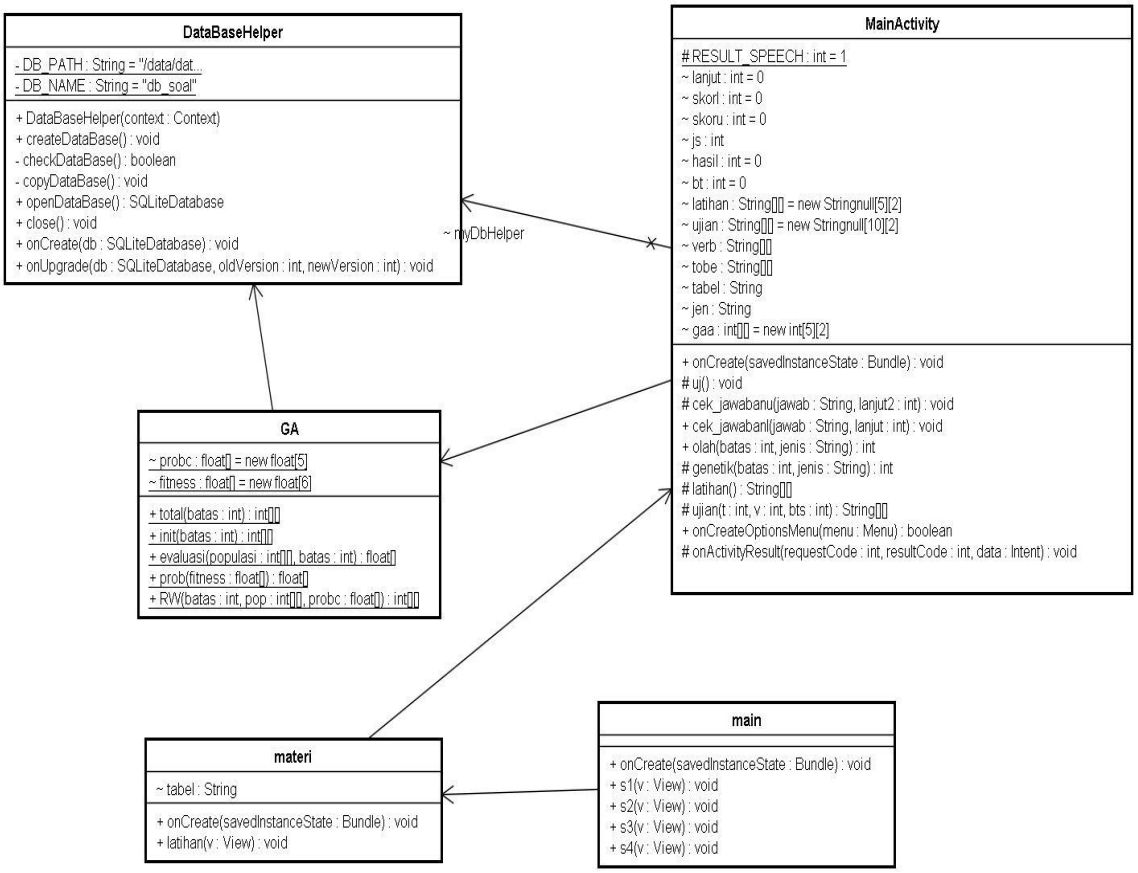
Aktifitas Mengerjakan Ujian adalah aktifitas dimana pemain akan dibaerikan soal ujian untuk mengukur kemampuan *tenses* setelah menjawab latihan soal. Segmen berikutnya, atau pemain telah selesai menyelesaikan semua permainan. Aktifitas ini terlihat seperti pada gambar III.15



Gambar III.12 *Activity Diagram* Soal Ujian

III.1.11.4 Class diagram

Class diagram adalah sebuah class yang menggambarkan struktur dan penjelasan class, paket, serta hubungan satu sama lain seperti containment, pewarisan, asosiasi, dan lain-lain. Class diagram juga menjelaskan hubungan antar class dalam sebuah sistem yang sedang dibuat dan bagaimana caranya agar mereka saling berkolaborasi untuk mencapai sebuah tujuan. Berikut ini adalah class diagram pada aplikasi pembelajaran edukasi *tenses* bahasa Inggris seperti pada Gambar III.16.

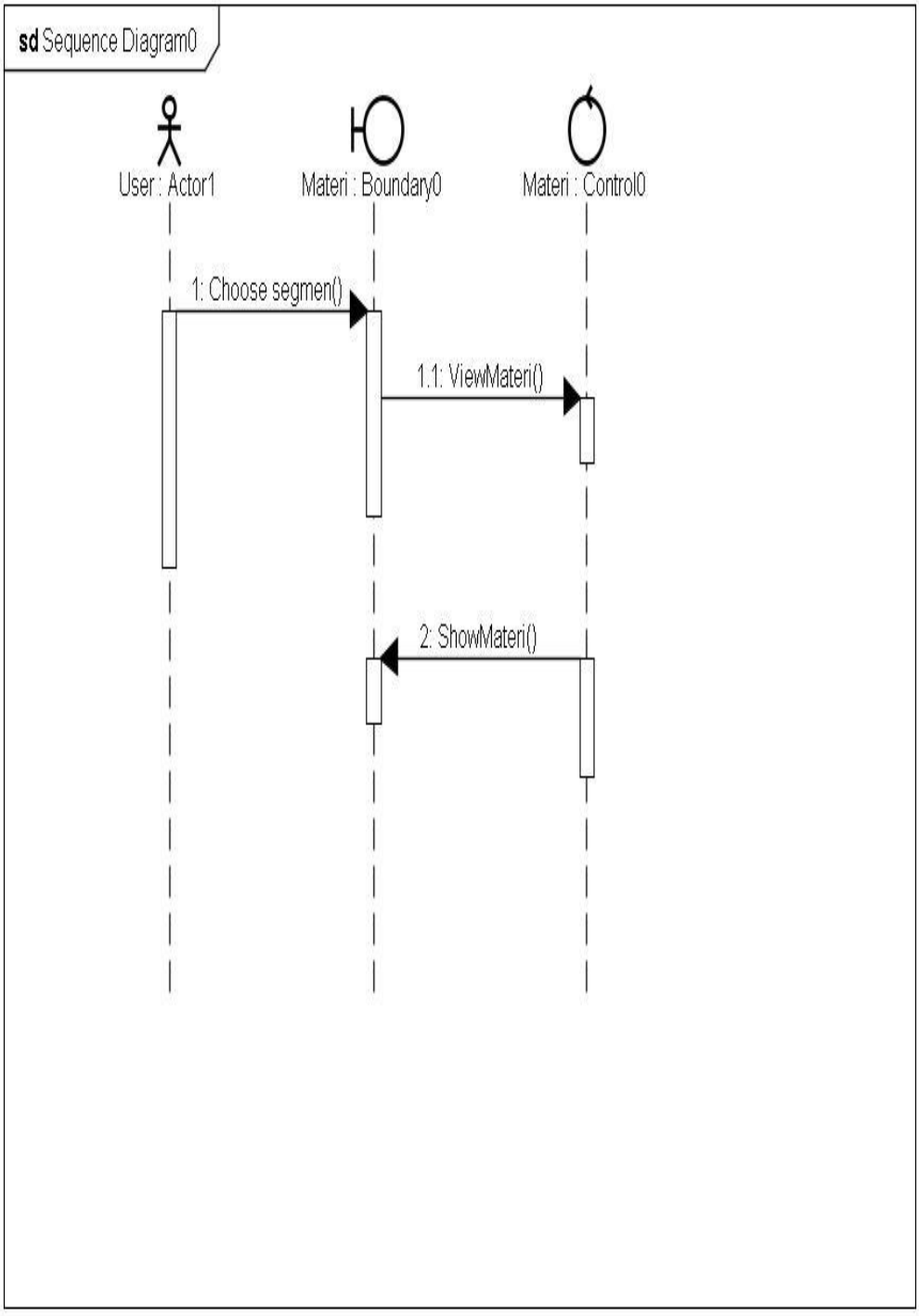


Gambar III.13 Class Diagram

III.1.11.5 Sequence Diagram

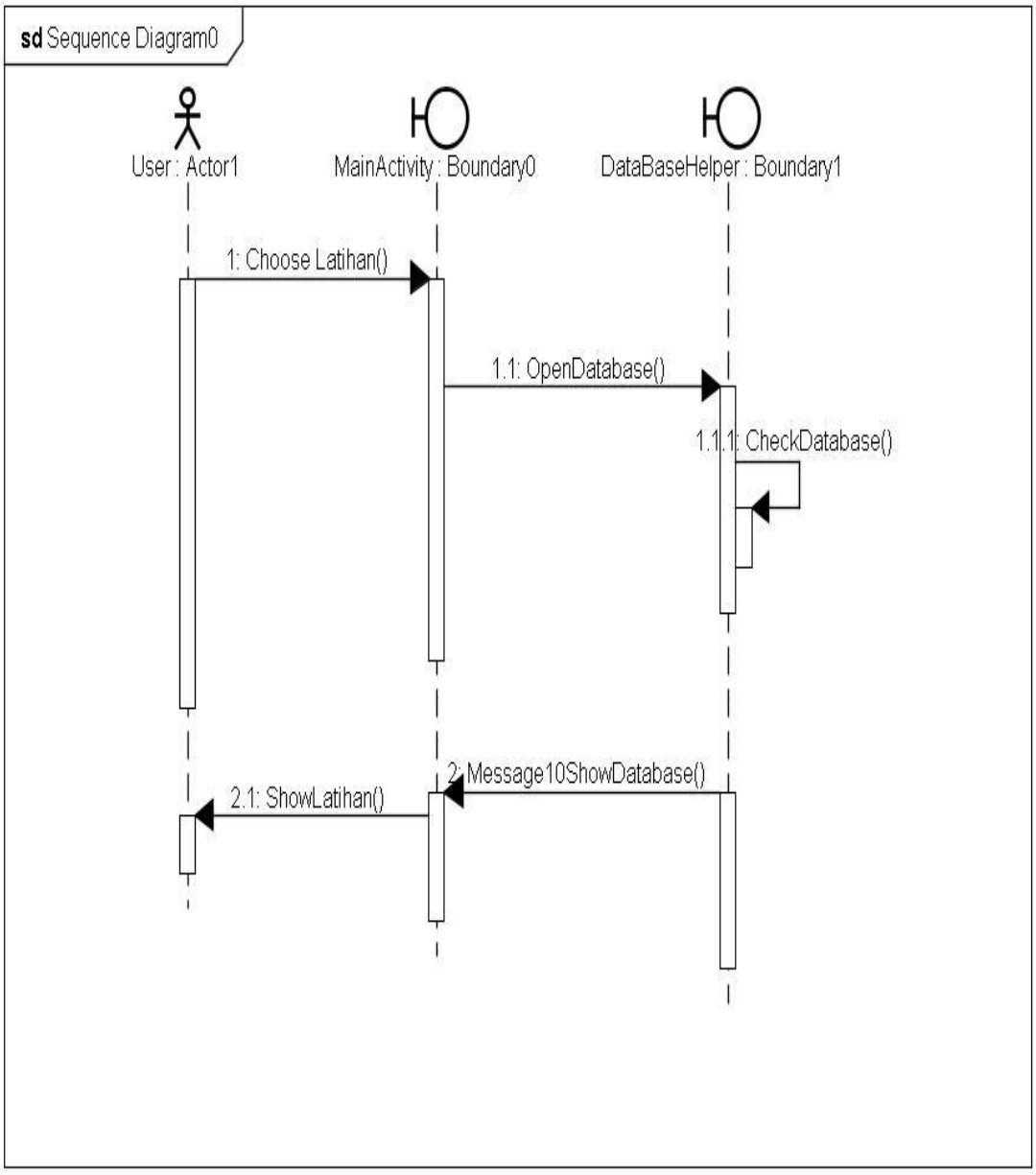
Sequence diagram digunakan untuk menggambarkan maksud dan perilaku suatu scenario. Diagram ini juga menunjukkan serangkaian pesan yang dipertukarkan oleh objek-objek yang melakukan tugas atau aksi.

1. **Sequence Diagram Materi**



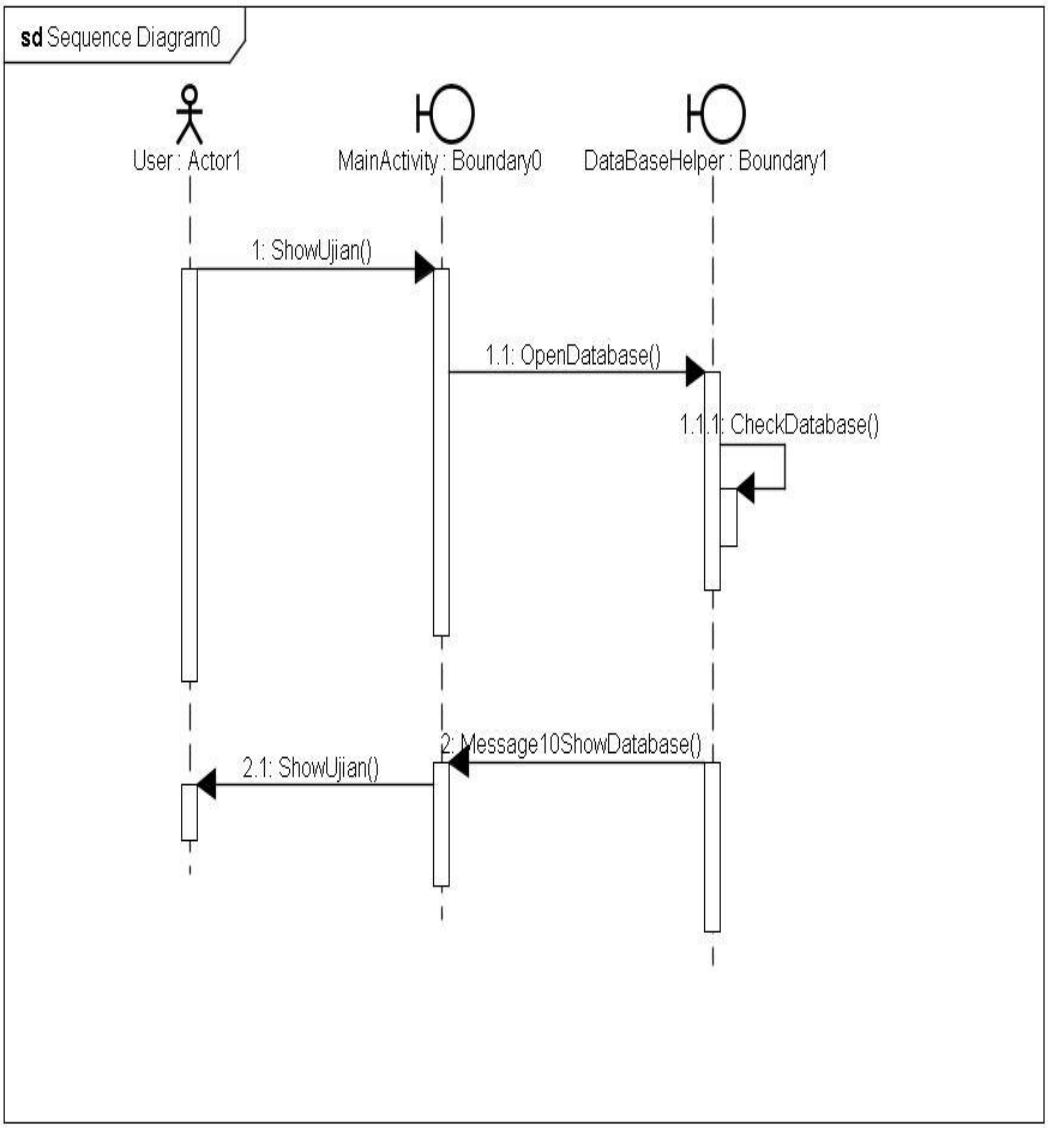
Gambar III.14 Sequence Diagram Materi

2. Sequence Diagram Latihan



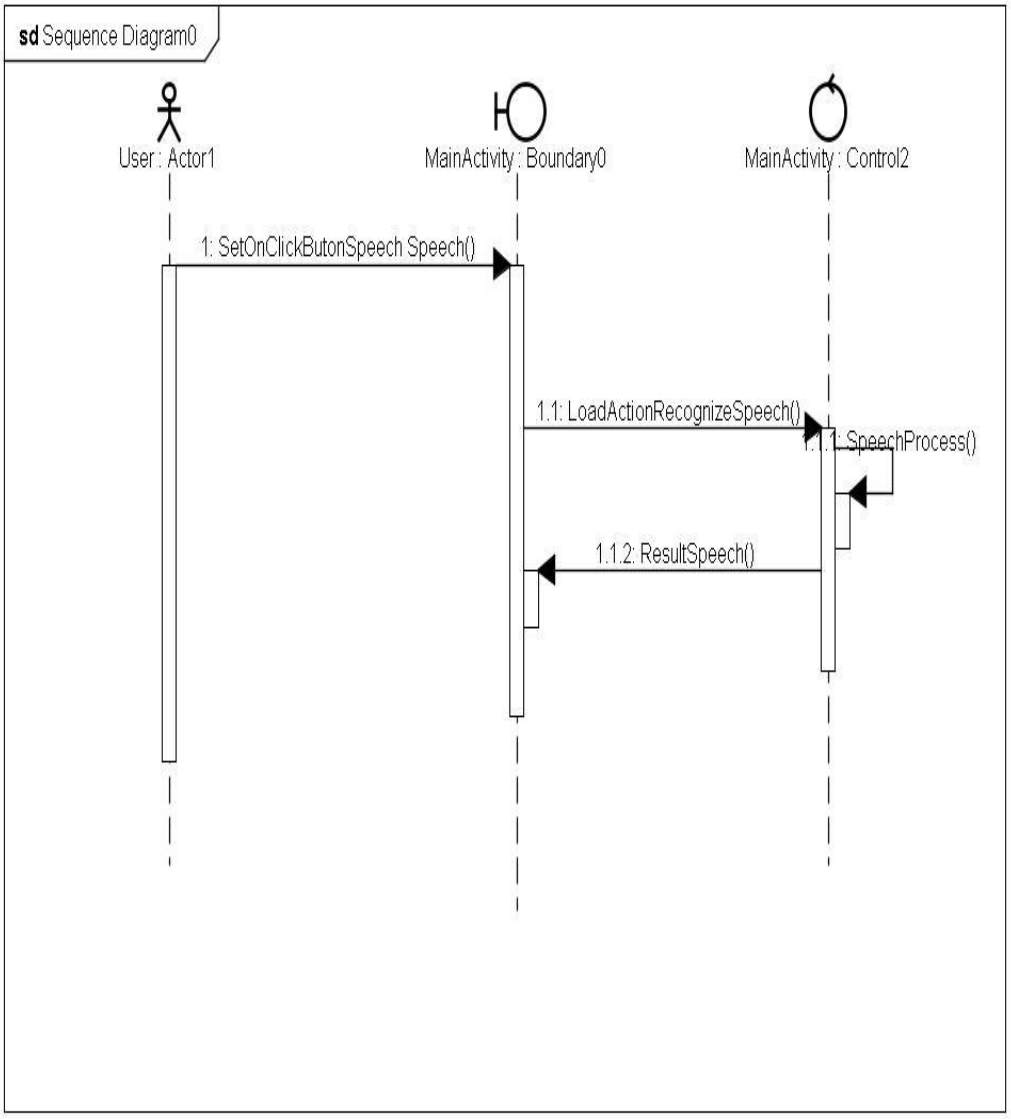
Gambar III.15 Sequence Diagram Latihan

3. Sequence Diagram Ujian



Gambar III.16 Sequence Diagram Ujian

4. Sequence Diagram *Speech*



Gambar III.17 Sequence Diagram *Speech*

III.1.12 Analisis Materi dan Soal

Pada tahap ini akan dianalisis materi dan soal yang akan diberikan kepada pemain yang akan menggunakan aplikasi ini. Materi *tenses* terdiri dari empat segmen materi yang memiliki empat sub materi. Sedangkan pada soal terdiri dari dua jenis, yaitu soal latihan dan soal ujian.

Materi *tenses* yang diberikan ditiap segmen terdiri dari:

1. *Present Tense*
2. *Past Tense*
3. *Future Tense*
4. *Past Future Tense*

Untuk setiap segmen, memiliki empat sub tense berdasarkan pekerjaannya yaitu *Simple, Continuous, Perfect, dan Perfect Continuous.* Materi yang ada pada aplikasi ini terdiri dari empat segmen yang memiliki empat sub materi dengan deskripsi rumus tiap materi seperti berikut ini:

1. Present Tense

Simple Present Tense: S + V1(s/es) + O

Present Continuous Tense: S + Tobe (is/are) + Ving + O

Present Perfect Tense: S + Have/Has + V3 + O

Present Perfect Continuous Tense: S + Have/Has + Been + Ving + O

2. Past Tense

Simple Past Tense: S + V2 + O

Past Continuous Tense: S + Was/Were + Ving + O

Past Perfect Tense: S + Had + V3 + O

Past Perfect Continuous Tense: S + Had + Been + Ving + O

3. Future Tense

Simple Future Tense: S + Will + V1 + O

Future Continuous Tense: S + Will + Be + Ving + O

Future Perfect Tense: S + Will + Have/Has + V3 + O

Future Perfect Continuous Tense: S + Will + Have/Has + Been + Ving + O

4. Past Future Tense

Simpe Past Future Tense: S + Would + V1 + O

Past Future Continuous Tense: S + Would + Be + Ving + O

Past Future Perfect Tense: S + Would + Had + V3 + O

Past Future Perfect Continuous Tense: S + Would + Had + Been + Ving + O

III.1.12.1. Struktur Latihan Soal dan Ujian

Soal pada aplikasi ini terdiri dari dua jenis, yaitu soal untuk latihan soal dan ujian. Soal pada latihan terdiri dari empat soal berdasarkan jumlah materi pada setiap segmen dan lima soal pada ujian.

1. Latihan Soal

Latihan soal berisikan soal-soal dengan materi berututan berdasarkan tenses pada segmen tertentu. Pemain harus melengkapi kata yang hilang pada suatu kalimat. Jika pada soal tertentu ada *Keyword* maka pemain harus menentukan *irregular verb* berdasarkan kata kerjanya. Jawaban yang harus dijawab pada setiap soal adalah kata kerja atau to be.

Contoh soal latihan pada segmen 2:

1. I ___ some fire wings yesterday (Keyword: V2 eat)
2. I ___ sleeping in sofa
3. She had ____ milk (Keyword: V3 drink)
4. He ____ been scoring goal yesterday (Keyword: V2 have)

2. Ujian

Ujian berisikan soal-soal yang merupakan evaluasi dari latihan soal yang sebelumnya telah dikerjakan. Uian memiliki soal yang lebih banyak dari latihan soal. Soal tidak berurutan berdasarkan tenses tiap segmen melainkan diacak untuk setiap soalnya. Jika pada soal tertentu ada *Keyword* maka pemain harus menentukan *irregular verb* berdasarkan kata kerjanya. Jawaban yang harus dijawab pada setiap soal adalah kata kerja atau to be.

Contoh Soal Ujian pada segmen 1

1. We ___ sleeping in barrack
2. I ____ been drowning in that sea
3. She ____ from here (Keyword: kata kerja berimbuhan untuk subjek SHE)
4. I have _____ this table (Keyword: V3 drop)
5. She ___ watering this plants

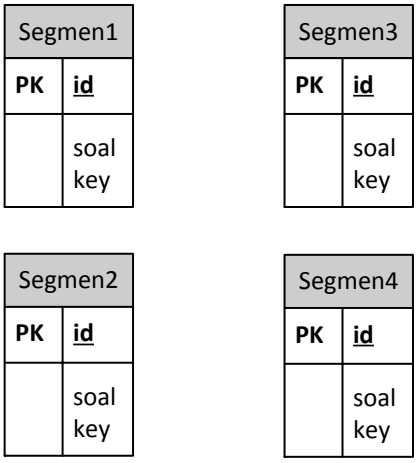
III.2 Perancangan

III.2.1 Perancangan Basis Data

Perancangan basis data bertujuan untuk mengidentifikasi struktur dari tiap-tiap *file* yang digunakan pada basis data. Pada tahap ini akan dibahas Skema basis data dan struktur tabel.

III.2.1.1 Skema Basis Data

Skema basis data menggambarkan bentuk basis data yang terdapat pada system dan tabel apa saja yang ada. Adapun tabel-tabel yang ada dalam sistem ini dapat digambarkan pada gambar III.17 :



Gambar III.18 Skema Basis Data

III.2.1.2 Struktur Tabel

Nama Tabel: segmen1

Keterangan: Penyimpanan Data Soal dan Jawaban

Tabel III.14 Tabel Segmen 1

Nama Field	Tipe	Ukuran	Keterangan
id	Text	10	Primary Key
Soal	Text	50	
key	Text	11	

Nama Tabel: Segmen2

Keterangan: Penyimpanan Data Soal dan Jawaban

Tabel III.15 Tabel Segmen 2

Nama Field	Tipe	Ukuran	Keterangan
Id	Text	10	Primary Key
Soal	Text	50	
key	Text	11	

Nama Tabel: segmen3

Keterangan: Penyimpanan Data Soal dan Jawaban

Tabel III.16 Tabel Segmen 3

Nama Field	Tipe	Ukuran	Keterangan
id	Text	10	Primary Key
soal	Text	50	
key	Text	11	

Nama Tabel: segmen4

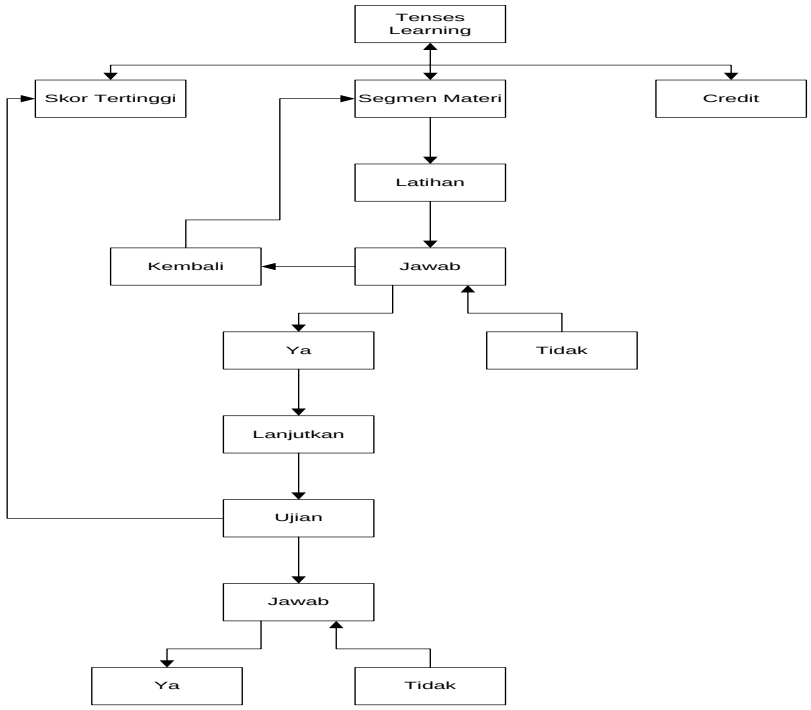
Keterangan: Penyimpanan Data Soal dan Jawaban

Tabel III.17 Tabel Segmen 4

Nama Field	Tipe	Ukuran	Keterangan
id	Text	10	Primary Key
Soal	Text	50	
key	Text	11	

III.2.2 Perancangan Struktur Menu

Perancangan struktur menu bertujuan untuk mempermudah pembuatan sistem yang akan dibangun. Perancangan struktur menu bertujuan untuk merancang bagian-bagian sistem dan mengetahui langkah mana yang terlebih dulu diakses setelah aplikasi yang dibangun selesai. Berikut perancangan arsitektur menu dari sistem yang akan dibangun dapat dilihat pada gambar III.27.

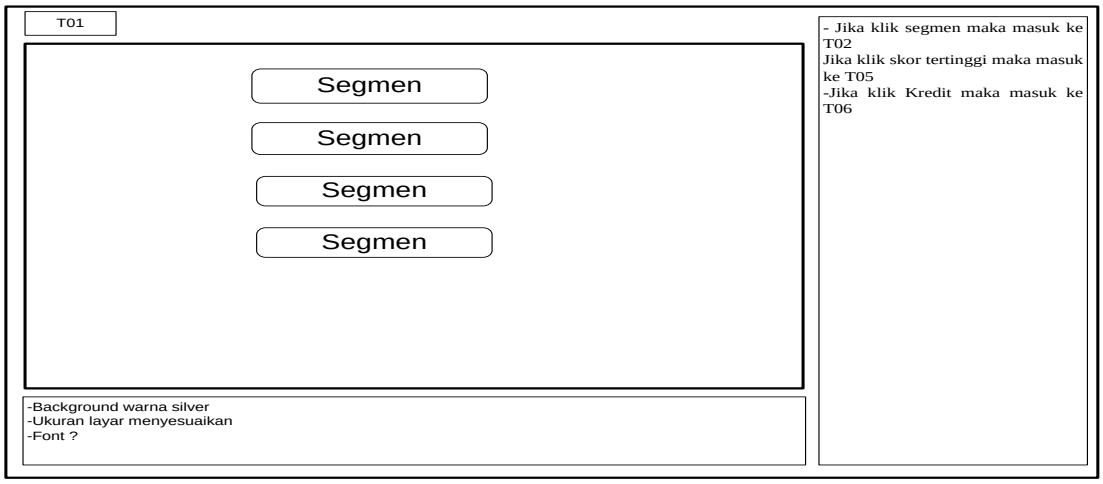


Gambar III.19 Struktur Menu

III.2.3 Perancangan Antar Muka

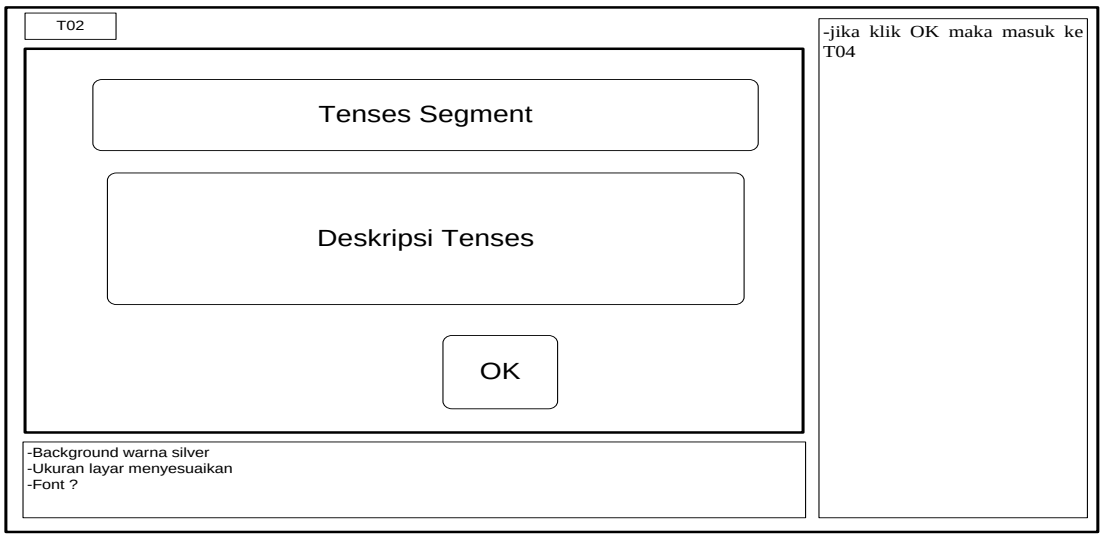
Perancangan antar muka digunakan untuk menggambarkan keadaan yang sebenarnya pada aplikasi yang akan dibangun. Berikut adalah perancangan antar muka yang akan dibangun.

1. Tampilan Awal Aplikasi



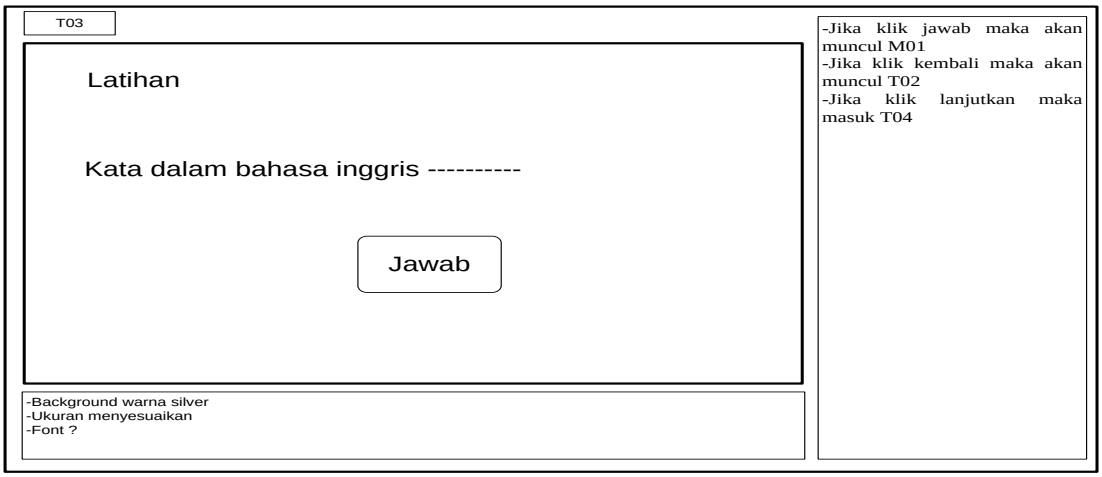
Gambar III.20 Tampilan Awal Aplikasi

2. Tampilan Menu Materi



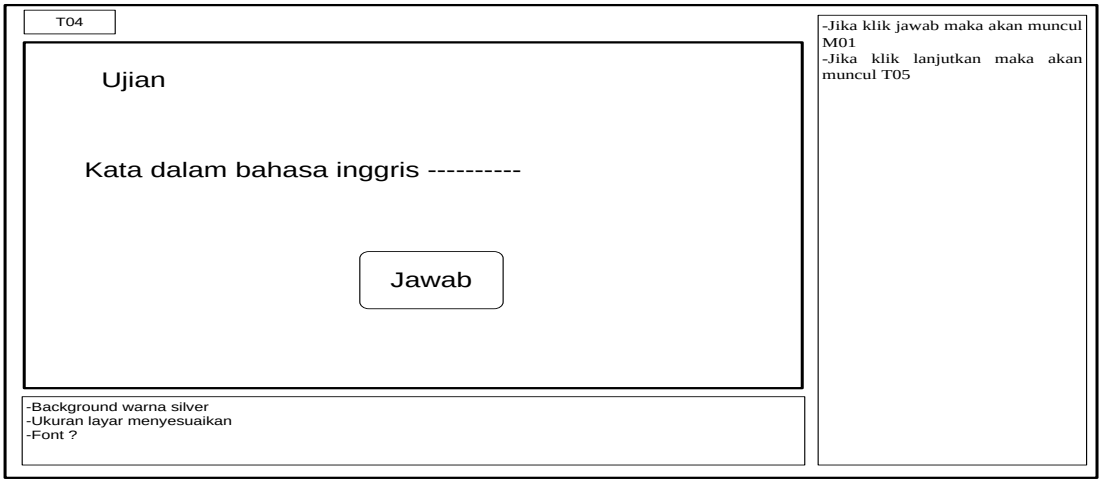
Gambar III.21 Tampilan Materi

3. Tampilan Menu Latihan



Gambar III.22 Tampilan Menu Latihan

4. Tampilan Menu Ujian

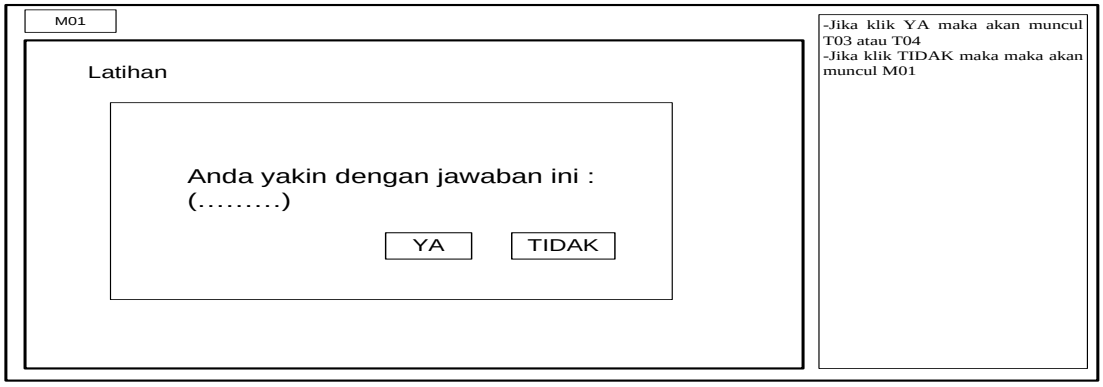


Gambar III.23 tampilan Menu Ujian

III.2.4 Perancangan Pesan

Berikut ini adalah perancangan pesan yang ada pada aplikasi pembelajaran *tenses* bahasa Inggris.

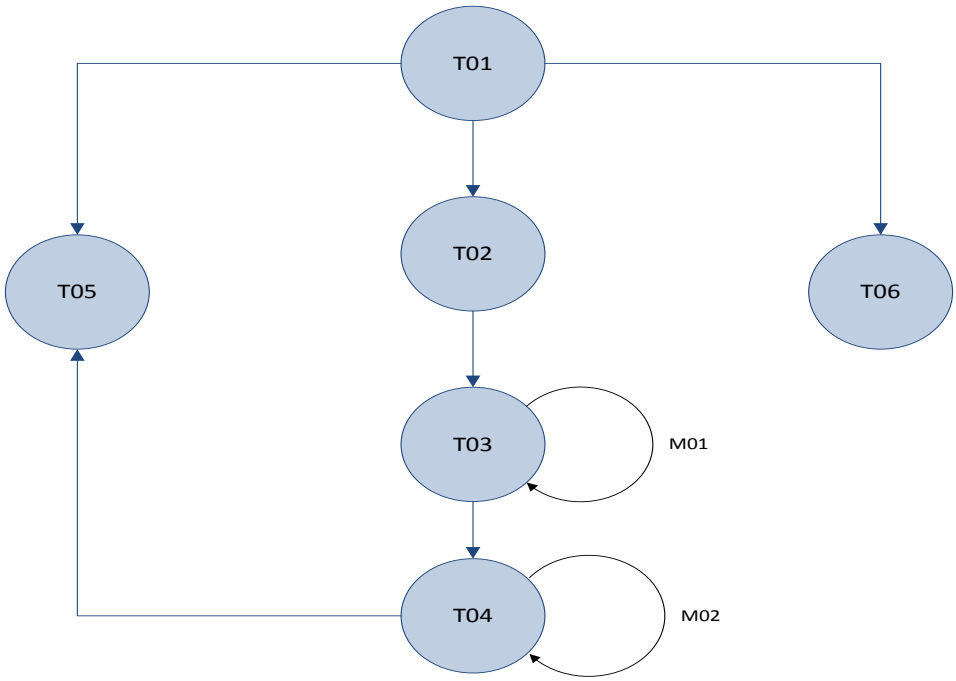
1. Pesan Peringatan Jawaban Latihan Yang Diinputkan



Gambar III.24 Pesan Peringatan Jawaban Latihan Yang Diinputkan

III.2.5 Jaringan Semantik

Tahapan selanjutnya adalah menggambarkan hubungan antar menu pada aplikasi pembelajaran tenses bahasa Inggris. Penggambaran tersebut diwujudkan dalam sebuah jaringan semantik.



Gambar III.25 Jaringan Semantik