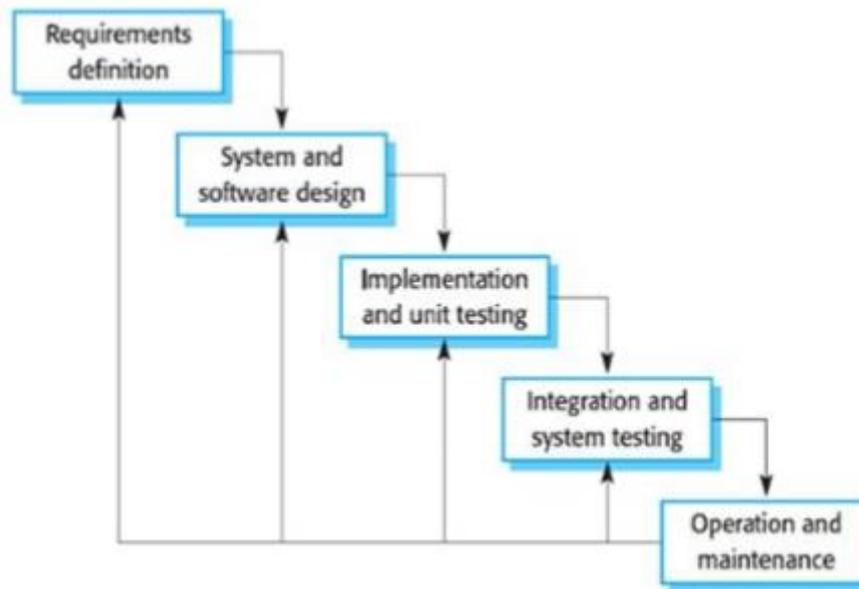


BAB III METODOLOGI PENELITIAN

Dalam bab ini penulis akan menjelaskan tentang metode penelitian yang digunakan yaitu metode *waterfall*, dengan metode ini dapat diketahui lebih lanjut masalah yang akan dijadikan objek pada penelitian.



Gambar 3.1 Metode *Waterfall*

Gambar 3.1 adalah bagan metode *waterfall* yang merupakan metode pengembangan sistem yang digunakan pada penelitian ini. Berikut adalah tahap-tahap pada metode *waterfall*.

1. Tahap *requirements analysis and definition*
2. Tahap *system and software design*
3. Tahap *implementation and unit testing*
4. Tahap *integration and system testing*
5. Tahap *operation and maintenance*

3.1 Requirements analysis and definition

Tahapan ini dilakukan dengan wawancara terbuka kepada subjek penelitian. Subjek penelitian ini adalah 36 siswa SMA kelas 11. Selain siswa, guru juga diwawancarai untuk mendapatkan kebutuhan sistem. Selain wawancara, studi literature juga dilakukan untuk memperdalam pemahaman peneliti terkait teori-teori yang ada dalam penelitian ini. Adapun prosedur pengumpulan data dapat dilihat pada Tabel 3.1.

Tabel 3.1 Prosedur Pengumpulan Data

Teknik	Data	Sumber
Studi Literatur	<ul style="list-style-type: none"> - Teori Sistem Pembelajaran Adaptif dalam penggunaannya di dunia pendidikan 	Jurnal dan artikel
Wawancara	<ul style="list-style-type: none"> - Mata pelajaran yang akan dikaji - Materi dan soal yang terdapat pada mata pelajaran tersebut 	Guru SMA dan siswa

Setelah dilakukan wawancara, didapatkan kebutuhan pengguna atau *user requirement*. Kebutuhan sistem pada aplikasi ini terletak pada *user requirement*, kebutuhan dari sudut pandang pengguna, apa saja yang pengguna inginkan dan bagaimana skenario untuk memenuhi kebutuhan. Pengguna hanya terdiri dari satu jenis yaitu siswa. Berikut akan dijabarkan pada Tabel 3.2.

Tabel 3.2 *User Requirement*

No	Kebutuhan	Skenario
1.	<i>User</i> ingin dapat <i>login</i> dan masuk ke dalam sistem.	<ul style="list-style-type: none"> - Masukkan <i>email</i> dan <i>password</i> - Tekan tombol <i>Login</i> untuk masuk ke sistem menuju ke <i>Dashboard user</i>
2.	<i>User</i> ingin dapat mengunduh materi	- Sistem memiliki halaman materi yang memungkinkan <i>user</i> untuk mengunduh materi sesuai dengan bab yang <i>user</i> pilih
3.	<i>User</i> ingin dapat mengerjakan soal	<ul style="list-style-type: none"> - Sistem memiliki halaman ujian yang memungkinkan <i>user</i> untuk memilih bab yang akan dikerjakan - <i>User</i> bisa mengerjakan soal pada halaman soal
4.	<i>User</i> ingin dapat melihat nilai hasil ujian	- Sistem bisa menampilkan nilai hasil ujian tiap bab pada halaman <i>dashboard</i>

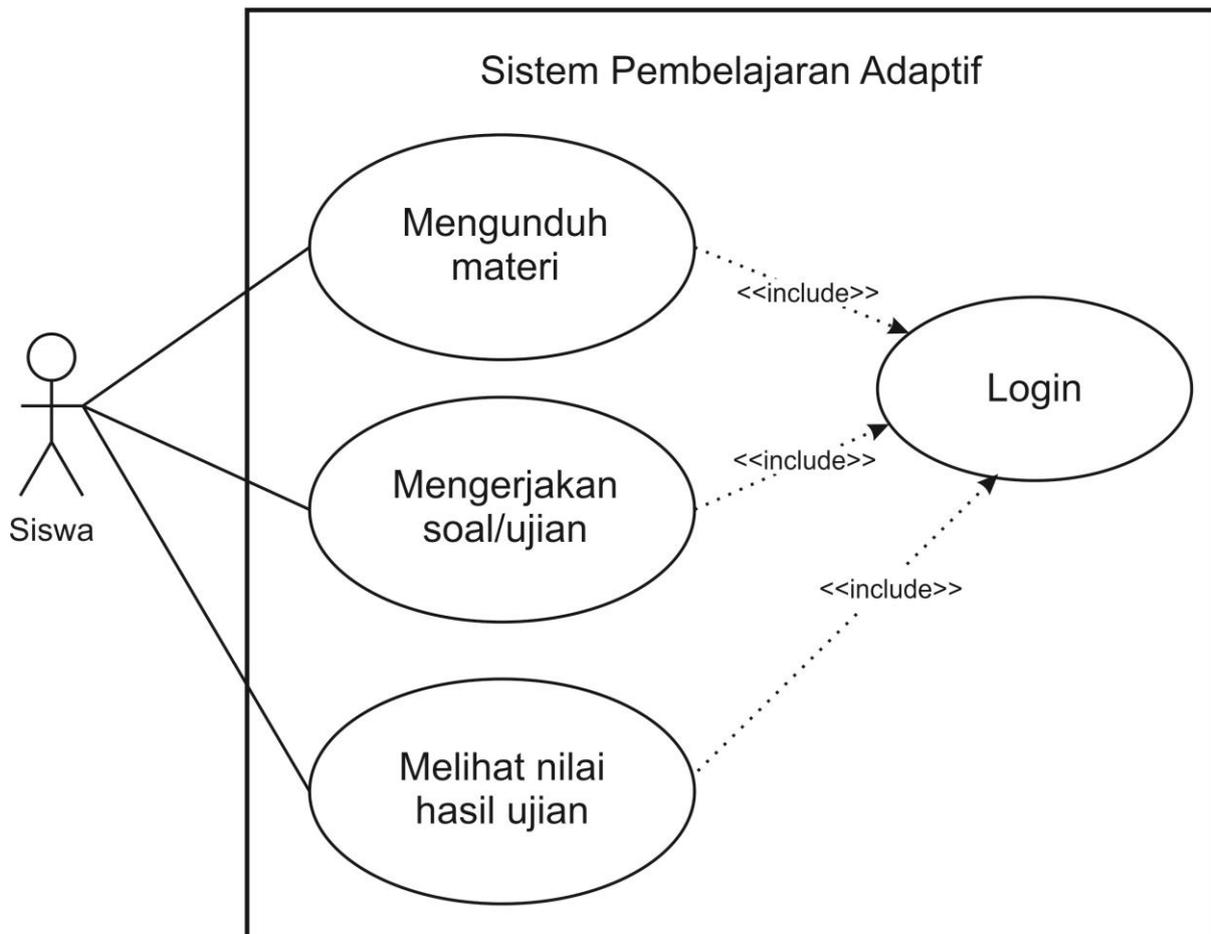
Pada Tabel 3.2, didapatkan 4 *user requirement* pada sistem ini yaitu *user* ingin dapat *login* dan masuk ke dalam sistem, *user* ingin dapat mengunduh materi, *user* ingin dapat mengerjakan soal, dan *user* ingin dapat melihat nilai hasil ujian. Keadaptifan dari sistem ini terletak pada saat *user* mengerjakan soal. Sistem akan menyesuaikan nilai yang diperoleh siswa dengan batasan KKM (Kriteria Ketuntasan Minimal). KKM (Kriteria Ketuntasan Minimal) menyesuaikan tempat dilakukannya penelitian yaitu SMA Negeri 1 Muntilan dengan KKM (Kriteria Ketuntasan Minimal) 75. Pada saat mengerjakan soal, apabila siswa mendapatkan nilai lebih dari sama dengan 75, maka soal yang akan ditampilkan pada bab selanjutnya adalah soal dengan tingkat kesulitan tinggi. Sedangkan apabila siswa mendapatkan nilai kurang dari 75 dan sudah mengerjakan soal sebanyak 3x percobaan, maka soal yang akan ditampilkan pada bab selanjutnya adalah soal dengan tingkat kesulitan mudah.

3.2 System and software design

Setelah mendapatkan data-data yang dibutuhkan, pada tahap ini akan dilakukan rancangan perangkat lunak sesuai dengan kebutuhan yang telah didapatkan pada tahap sebelumnya.

3.2.1 Use Case Diagram

Rancangan pertama yaitu *use case* yang bertujuan untuk mengetahui fungsi pada sebuah sistem dengan interaksinya, pada Gambar 3.2 dibuat rancangan *use case diagram* seperti ini:



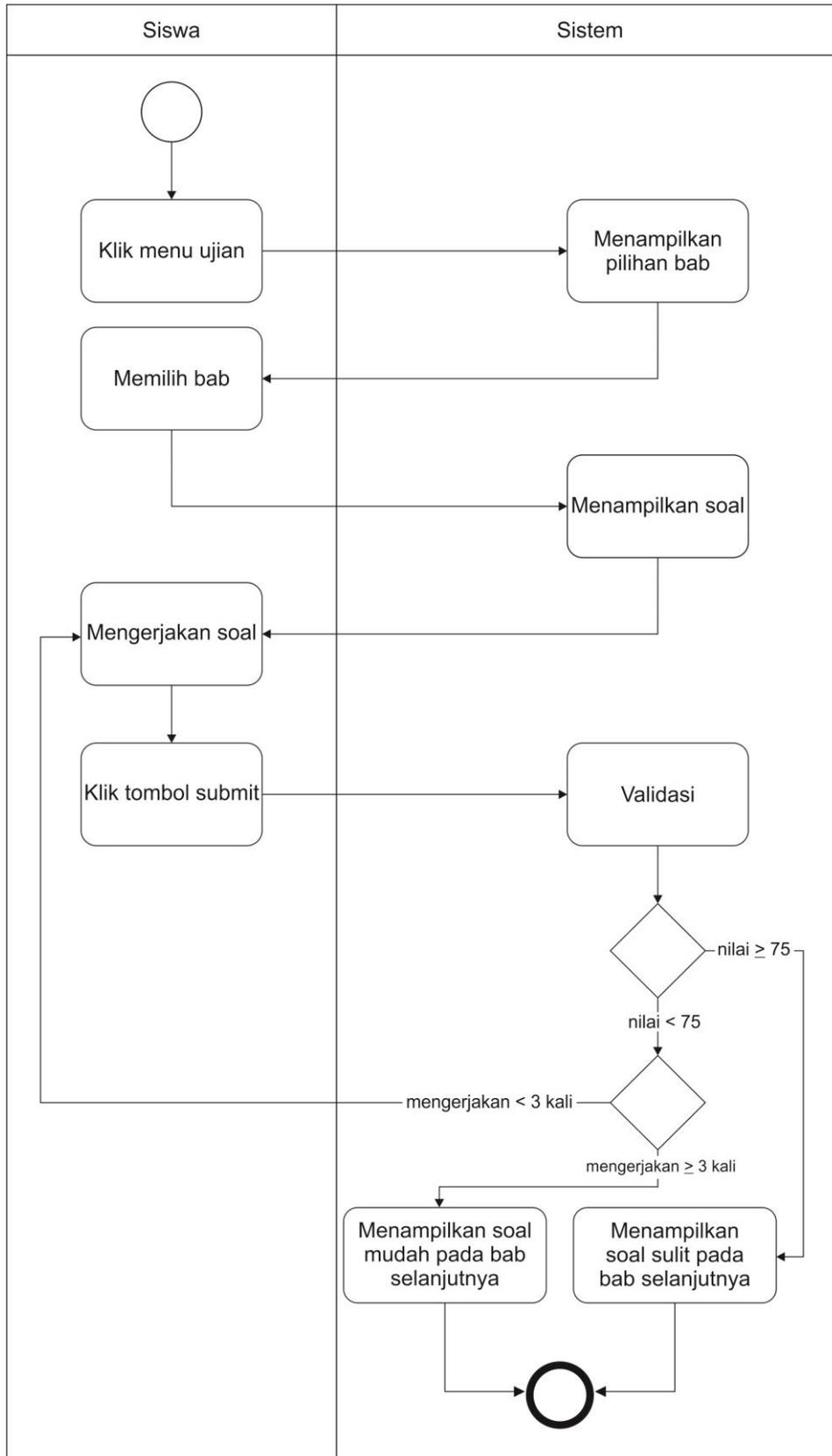
Gambar 3.2 Use Case Diagram

Pada Gambar 3.2 menjelaskan rancangan fungsi sebuah sistem yang akan diimplementasikan pada bab berikutnya. Dari gambar diatas dapat diketahui hanya ada 1 aktor yang berperan penting di sistem ini yaitu siswa. Dapat dilihat juga adanya 4 proses yang terdapat pada sistem, yaitu siswa dapat *login*, dapat mengunduh materi, dan mengerjakan soal.

3.2.2 Activity Diagram

Untuk memahami lebih lanjut dari fungsi yang telah dibuat sebelumnya pada *use case*, dibuatlah rancangan *activity diagram* sebagai berikut:

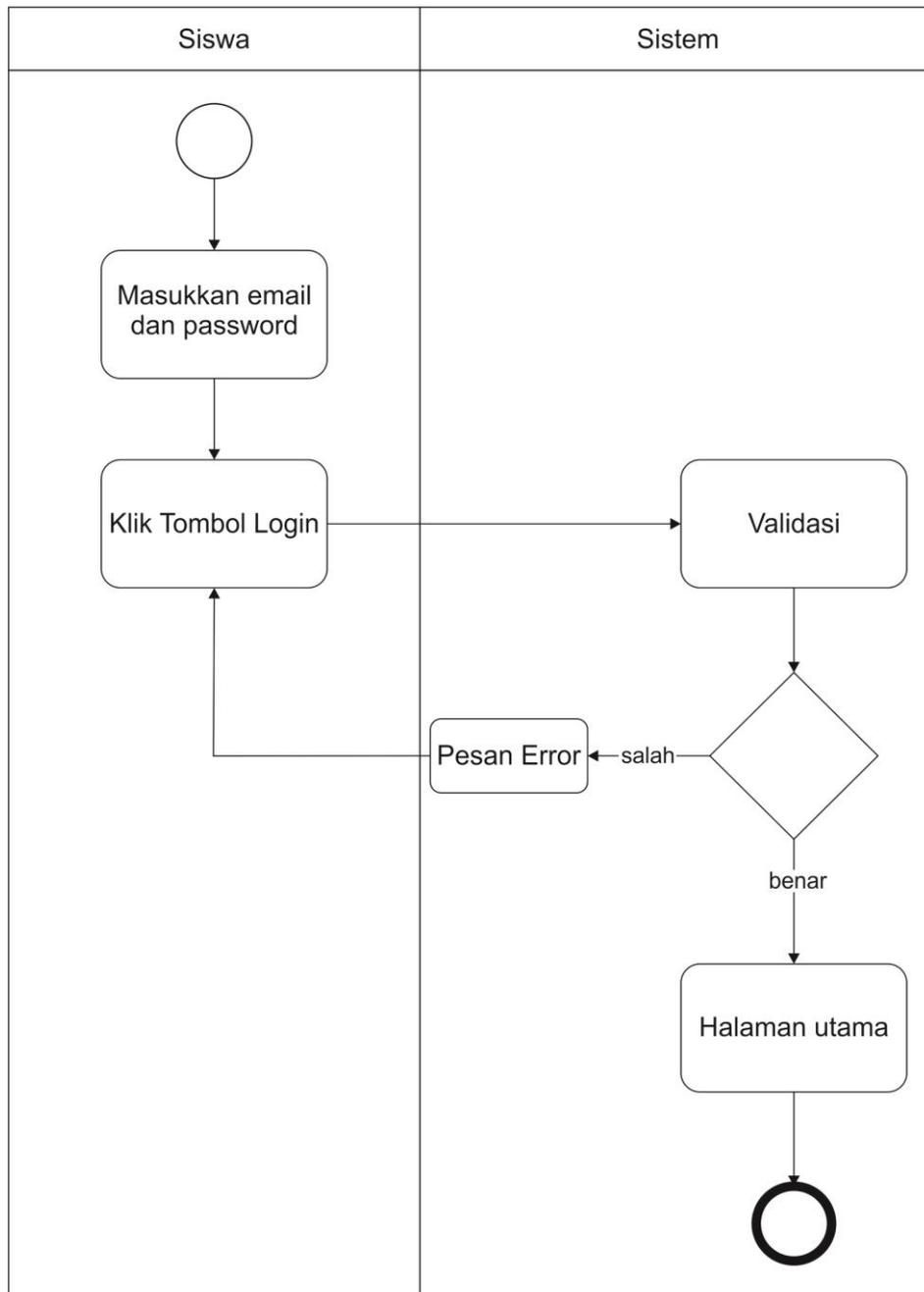
1. Fungsi Mengerjakan Soal/Ujian



Gambar 3.3 Activity Diagram Keadaptifan Sistem

Dari Gambar 3.3 diatas dapat disimpulkan bahwa keadaptifan dari sistem ini terletak pada saat pengerjaan soal. Dimulai dari saat pengguna meng-klik tombol ujian, lalu sistem merespon dengan menampilkan pilihan bab yang tersedia. Pengguna bisa memilih bab yang tersedia dan selanjutnya pengguna langsung bisa mengerjakan soal. Apabila sudah selesai, pengguna bisa meng-klik tombol *submit*. Sistem akan memvalidasi hasil pekerjaan pengguna. Apabila pengguna mendapatkan nilai lebih dari sama dengan 75, maka sistem akan menampilkan soal dengan tingkat kesulitan tinggi pada bab selanjutnya. Sedangkan apabila nilai yang didapatkan kurang dari 75, maka sistem akan mengecek terlebih dahulu apakah pengguna tersebut sudah mengerjakan soal sebanyak 3 kali. Apabila sudah mengerjakan soal sebanyak 3 kali, maka sistem akan menampilkan soal pada bab selanjutnya dengan tingkat kesulitan rendah dan apabila pengguna belum mengerjakan soal sebanyak 3 kali, maka pengguna bisa mengerjakan kembali soal yang tersedia dan diberi batas sampai 3 kali mengerjakan agar bisa mencapai KKM (Kriteria Ketuntasan Minimal). Jika ditinjau dari *user model*, maka sistem ini menggunakan *user model* berupa *goals and tasks* (tujuan dan tugas) dan *knowledge* (pengetahuan). *Goals and tasks* (tujuan dan tugas) diambil dari nilai KKM (Kriteria Ketuntasan Minimal) yaitu 75 dan *knowledge* (pengetahuan) diambil dari nilai yang didapatkan pengguna pada bab sebelumnya.

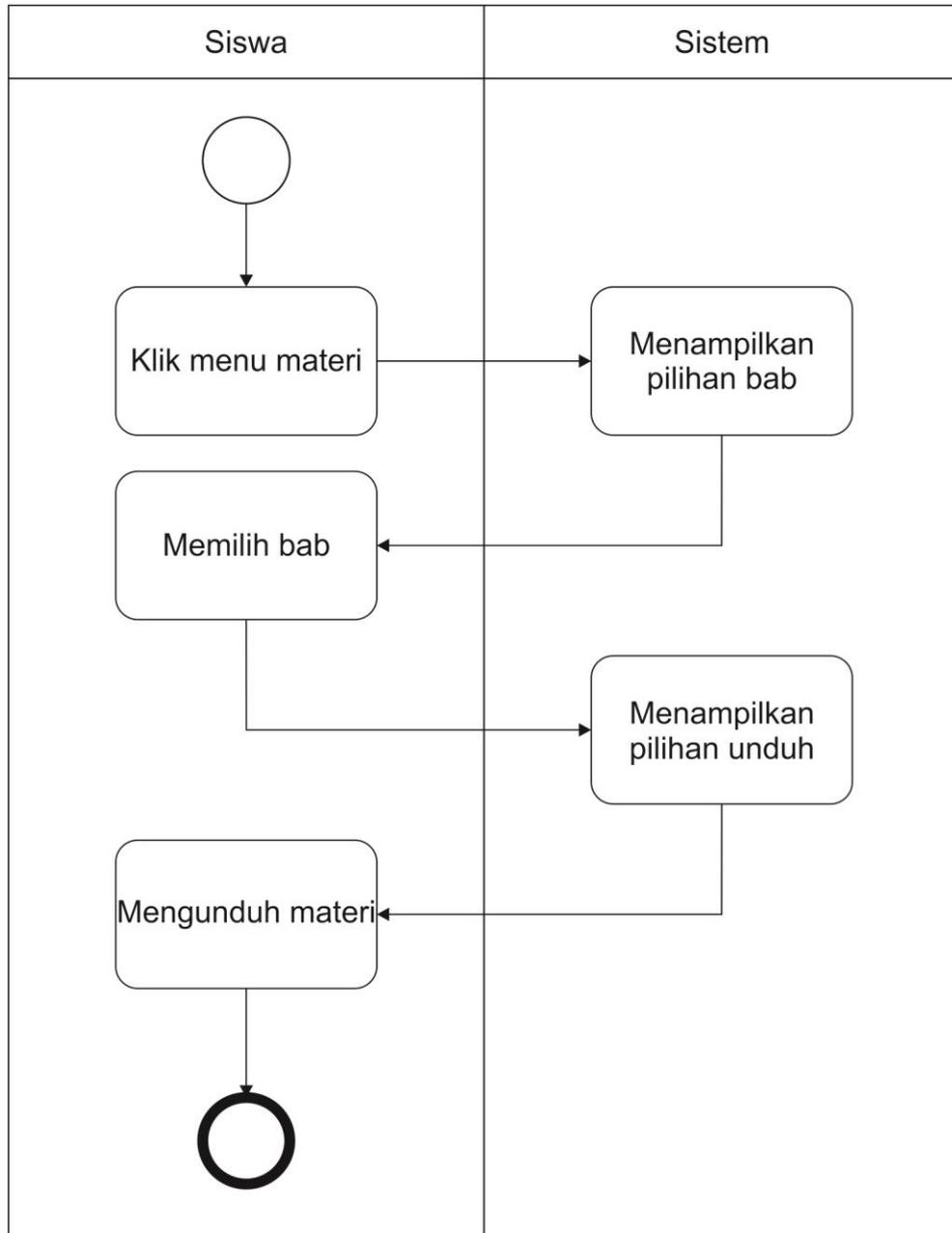
2. Fungsi Login



Gambar 3.4 Activity Diagram Login

Dari Gambar 3.4 diatas dapat disimpulkan bahwa semua aktor dapat melakukan *login* ke dalam sistem, dimulai dengan memasukkan *email* dan *password* kemudian klik tombol *login* dan akan melewati tahap validasi oleh sistem. Jika data *username* dan *password* tidak sesuai maka akan keluar pesan *error* dan jika sesuai maka akan langsung masuk ke halaman utama sistem.

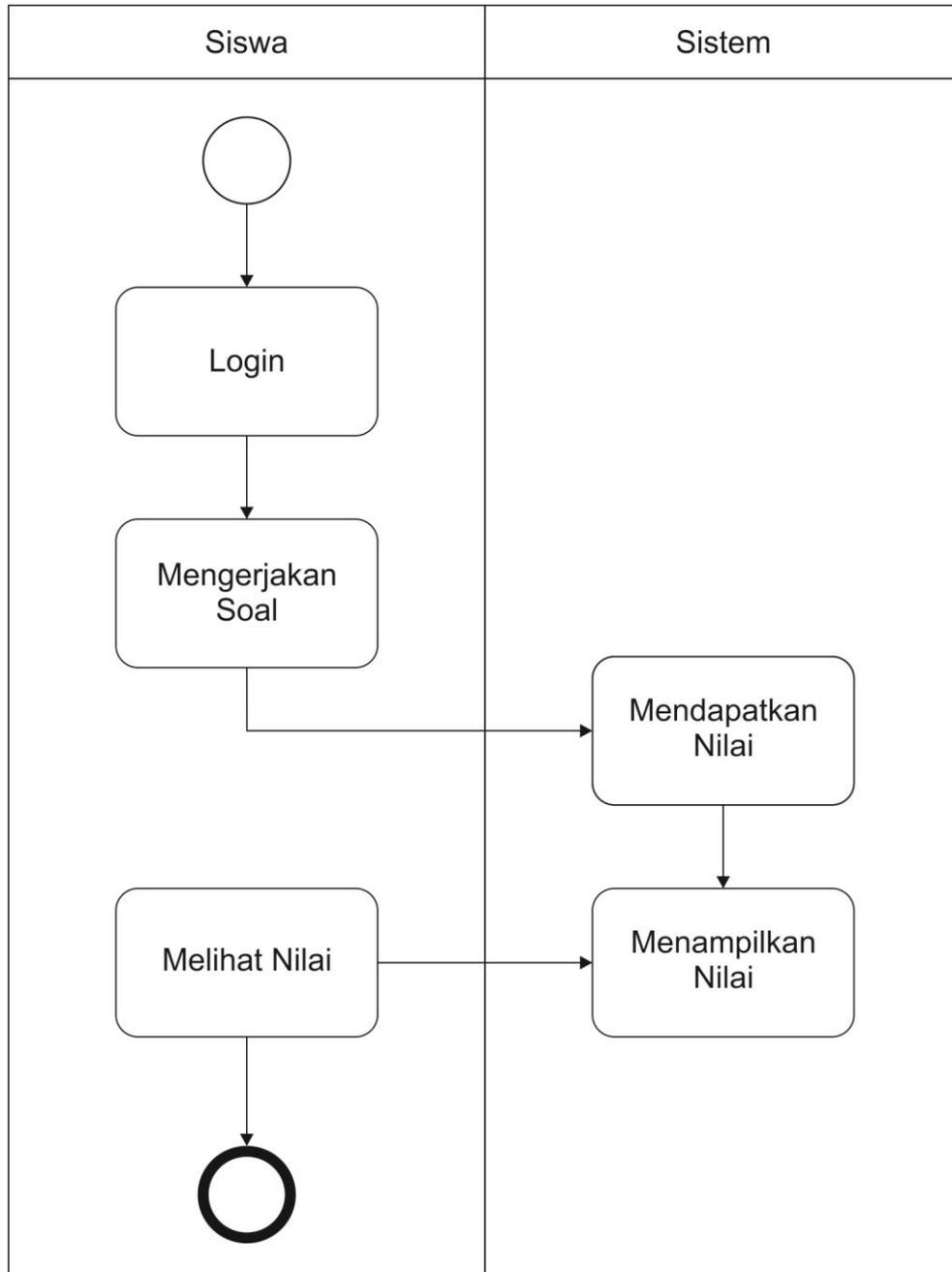
3. Fungsi Mengunduh Materi



Gambar 3.5 *Activity Diagram* Mengunduh Materi

Dari Gambar 3.5 diatas memperlihatkan fungsi mengunduh materi yang dilakukan oleh siswa. Ketika sudah *login* dan siswa berada pada halaman utama, siswa bisa memilih menu materi dan kemudian akan dialihkan ke halaman yang menampilkan daftar pilihan bab. Pada halaman ini, siswa memilih bab yang akan diunduh materinya. Setelah itu, sistem akan menampilkan pilihan unduh dan siswa bisa mengunduh materi.

4. Fungsi Melihat Nilai

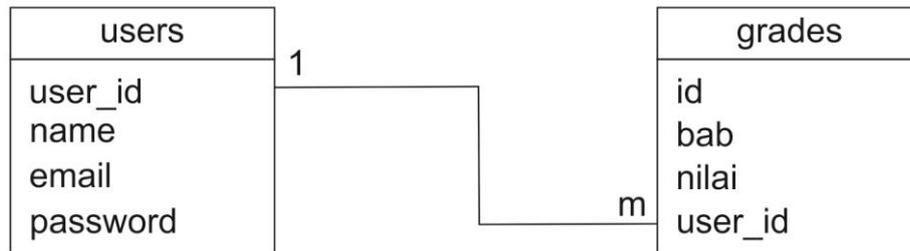


Gambar 3.6 Activity Diagram Melihat Nilai

Pada Gambar 3.6 menunjukkan fungsi siswa untuk melihat nilai yang didapatkan dari ujian yang telah dikerjakan. Nilai akan otomatis muncul pada halaman memilih bab ujian ketika pengguna telah selesai mengerjakan soal pada bab tertentu.

3.2.3 Relasi Antar Tabel

Perancangan *database* yang akan dibuat pada sistem ini memiliki 2 tabel yang mempunyai relasi satu sama lain, yaitu tabel *users* dan *grades* yang digambarkan seperti pada Gambar 3.7. Tabel *users* dan tabel *grades* mempunyai relasi *one to many*.



Gambar 3.7 Relasi Antar Tabel

a. Struktur Tabel

Untuk memahami lebih lanjut dari fungsi yang telah dibuat sebelumnya pada *use case*, dibuatlah rancangan *activity diagram* sebagai berikut:

Tabel 3.3 Tabel *users*

No.	Atribut	Tipe Data	Contrain	Keterangan
1.	user_id	BigInteger (20)	<i>Primary_key</i>	<i>Not Null</i>
2.	name	Varchar (191)		<i>Not Null</i>
3.	email	Varchar (191)		<i>Not Null</i>
4.	password	Varchar (191)		<i>Not Null</i>

Tabel *users* digunakan untuk menyimpan data pengguna seperti pada Tabel 3.3.

Terdiri dari 4 atribut yaitu:

user_id : Tipe data big integer, batasan karakter 20, *Primary Key*, *Auto Increment*

name : Tipe data varchar, batasan karakter 191

email : Tipe data varchar, batasan karakter 191

password : Tipe data varchar, batasan karakter 191

Tabel 3.4 Tabel *grades*

No.	Atribut	Tipe Data	Contrain	Keterangan
1.	id	BigInteger (20)	<i>Primary_key</i>	<i>Not Null</i>
2.	bab	Varchar (191)		<i>Not Null</i>
3.	nilai	Integer (11)		<i>Not Null</i>
4.	user_id	BigInteger (20)	<i>Foreign Key</i>	<i>Not Null</i>

Tabel *grades* digunakan untuk menyimpan data nilai pengguna dan memiliki relasi dengan tabel *users*. Tabel 3.4 terdiri dari 4 atribut yaitu:

id : Tipe data big integer, batasan karakter 5, *Primary Key*, *Auto Increment*
 bab : Tipe data varchar, batasan karakter 191
 nilai : Tipe data integer, batasan karakter 11
 user_id : Tipe data big integer, batasan karakter 20, *Foreign Key* dari tabel *users*, digunakan untuk menyimpan data *pengguna* dalam tabel *grades*.

3.3 Implementation and unit testing

Setelah melalui tahap *system and software design*, langkah selanjutnya adalah menuliskan rancangan tersebut ke dalam kode-kode program. Dalam implementasi ini menggunakan *framework laravel* dan *database mysql*.

3.3.1 Implementasi

a. Halaman Awal

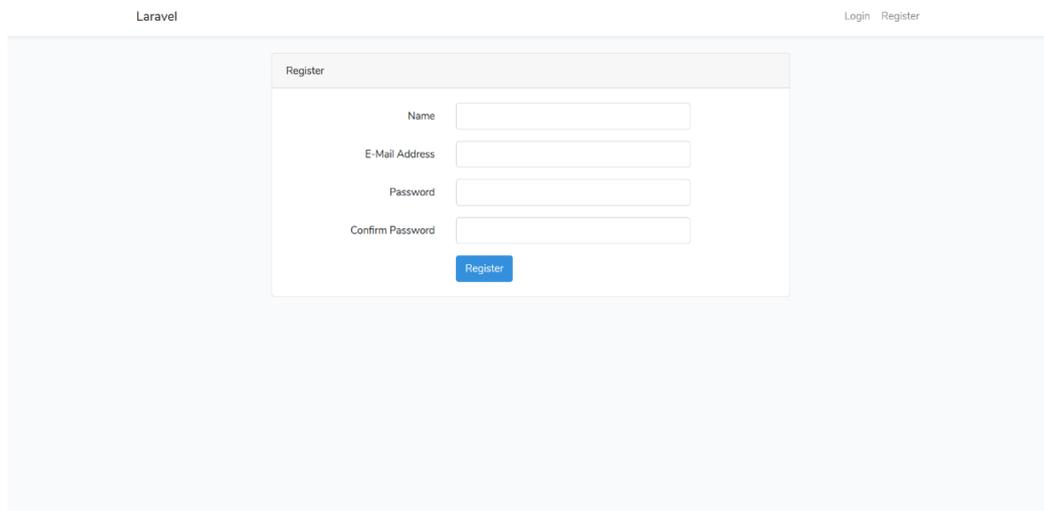
Halaman awal adalah halaman yang ditujukan untuk pengguna saat pertama kali mengakses sistem. Ada 2 tombol yang bisa dipilih untuk masuk ke dalam sistem yaitu tombol *sign up* dan tombol *log in*. Tombol *sign up* ditujukan untuk pengguna yang belum pernah mendaftarkan diri ke dalam sistem. Sedangkan tombol *login* ditujukan untuk pengguna yang sudah pernah mendaftarkan diri ke dalam sistem. Halaman awal bisa dilihat pada Gambar 3.8.



Gambar 3.8 Halaman Awal

b. Halaman *Sign Up*

Halaman *sign up* adalah salah satu lanjutan dari halaman awal. Setelah pengguna berhasil mengakses sistem, yang dilakukan selanjutnya adalah masuk ke dalam sistem. Halaman ini ditujukan bagi pengguna baru yang belum pernah mendaftarkan diri ke dalam sistem dengan mengisi *form* yang tersedia. Pengguna cukup mengisikan nama, *e-mail*, dan kata sandi untuk mendaftarkan diri ke dalam sistem. Setelah dirasa sudah benar dalam pengisian *form*, pengguna bisa meng-klik tombol *register* agar data-data pengguna tersebut tersimpan ke dalam sistem dan bisa langsung digunakan. Setelah meng-klik tombol *register*, pengguna akan langsung masuk ke dalam sistem menggunakan data diri yang pengguna isikan pada *form*. Halaman *sign up* bisa dilihat pada Gambar 3.9.



The image shows a web browser window displaying a registration form. The page title is 'Laravel' and there are links for 'Login' and 'Register' in the top right corner. The form itself is titled 'Register' and contains the following fields:

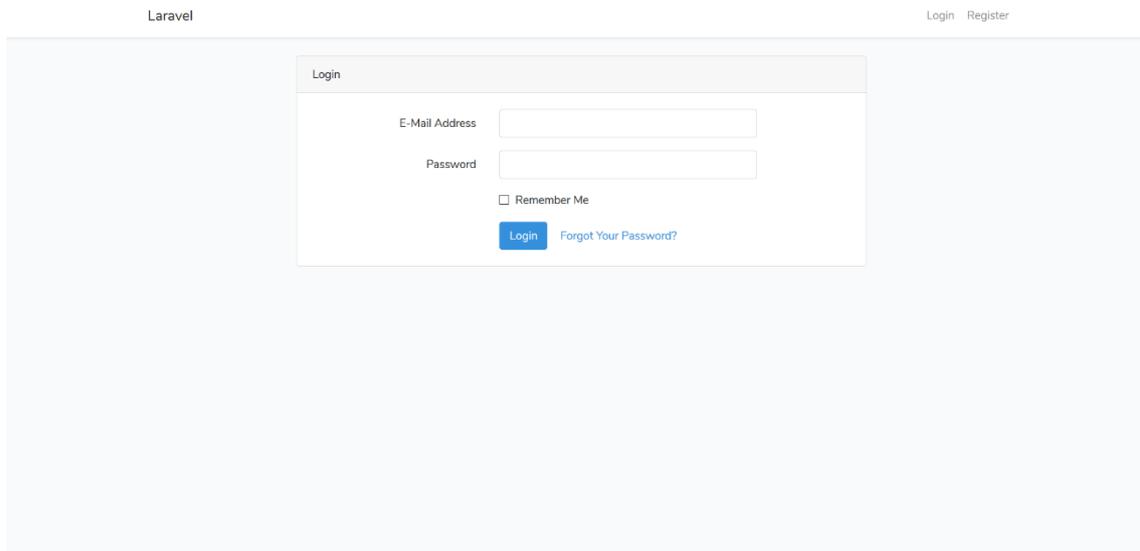
- Name:
- E-Mail Address:
- Password:
- Confirm Password:

Below the Confirm Password field is a blue button labeled 'Register'.

Gambar 3.9 Halaman *Sign Up*

c. Halaman *Log In*

Halaman *log in* adalah lanjutan dari halaman awal. Halaman ini ditujukan bagi pengguna yang sudah pernah mendaftarkan diri ke dalam sistem. Pengguna hanya perlu memasukkan *e-mail* dan kata sandi yang sudah terdaftar ke dalam sistem untuk masuk. Setelah dirasa benar dalam mengisi *e-mail* dan kata sandi, pengguna bisa meng-klik tombol *log in* yang tersedia. Halaman *log in* bisa dilihat pada Gambar 3.10.



Gambar 3.10 Halaman *Log In*

d. Halaman Utama

Halaman ini adalah halaman utama dari sistem pembelajaran adaptif untuk siswa. Terdapat *dashboard* yang menunjukkan nilai-nilai pengguna. *Dashboard* ini terdiri dari:

- Kriteria Ketuntasan Minimal (KKM) didapatkan dari hasil wawancara di tempat penelitian. KKM merupakan salah satu *user model* pada sistem ini, yaitu tujuan dan tugas (*goals and tasks*). KKM adalah batas nilai minimum yang didapatkan oleh pengguna (siswa) untuk mencapai kelulusan.

- Nilai Ideal

Nilai ideal adalah nilai minimal yang harus didapatkan oleh pengguna. Nilai ini akan berubah menyesuaikan dengan jumlah bab yang telah dikerjakan. Rumus dari penghitungan ini adalah :

$$\text{Nilai Ideal} = \frac{KKM * n}{8}$$

KKM = 75.

n = jumlah bab yang telah dikerjakan.

- Nilai Aktual

Nilai Aktual adalah nilai pengguna yang sebenarnya, Nilai ini akan berubah menyesuaikan dengan jumlah nilai yang didapatkan tiap bab. Rumus dari penghitungan ini adalah:

$$\text{Nilai Aktual} = \sum_{i=1}^n \frac{n}{8}$$

n didapatkan dari jumlah keseluruhan nilai yang didapatkan pada masing-masing bab.

- Jarak

Jarak didapatkan dari pengurangan antara nilai aktual dengan nilai ideal.

- Lingkaran abu-abu

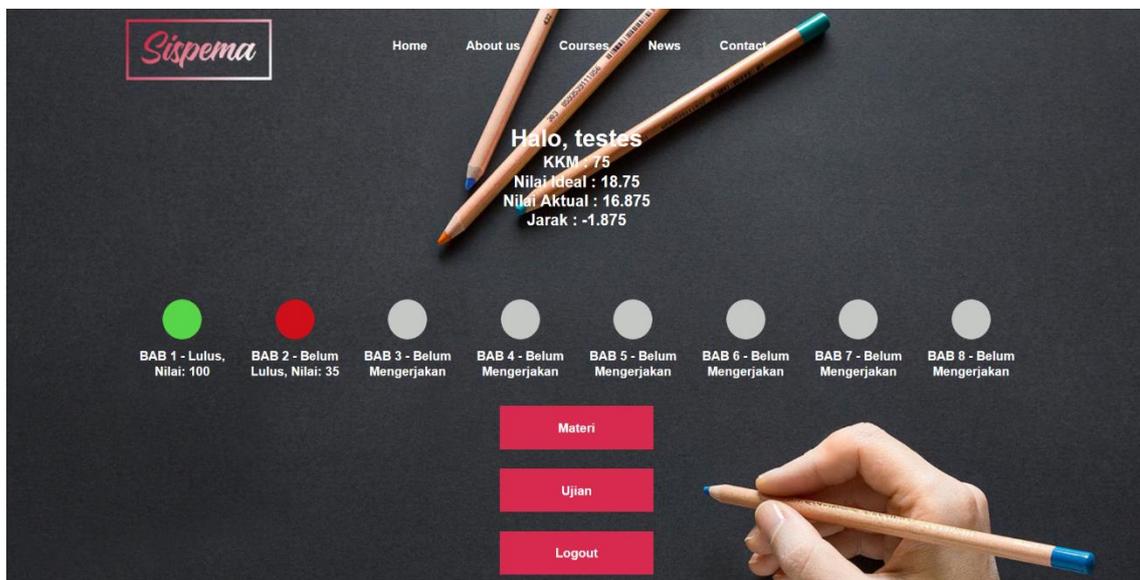
Lingkaran abu-abu akan muncul jika belum ada nilai yang masuk ke dalam sistem. Hal ini berarti pengguna belum pernah mengerjakan soal pada bab tersebut.

- Lingkaran merah

Lingkaran merah akan muncul jika nilai sudah masuk ke dalam sistem tetapi nilai belum memenuhi KKM. Hal ini menandakan bahwa pengguna sudah mengerjakan soal pada bab tersebut, tetapi nilai yang didapatkan tidak melebihi KKM (kurang dari 75).

- Lingkaran hijau

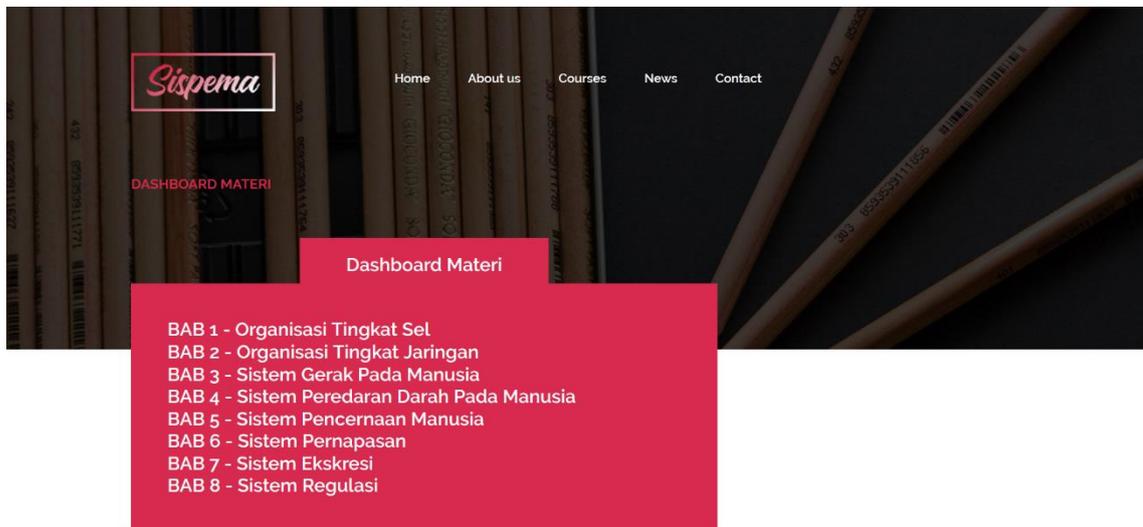
Lingkaran hijau akan muncul jika nilai sudah masuk ke dalam sistem dan nilai sudah memenuhi KKM. Hal ini menandakan bahwa pengguna sudah mengerjakan soal pada bab tersebut dan nilai yang didapatkan telah melebihi KKM (lebih dari 75).



Gambar 3.11 Halaman Utama

e. Halaman Materi

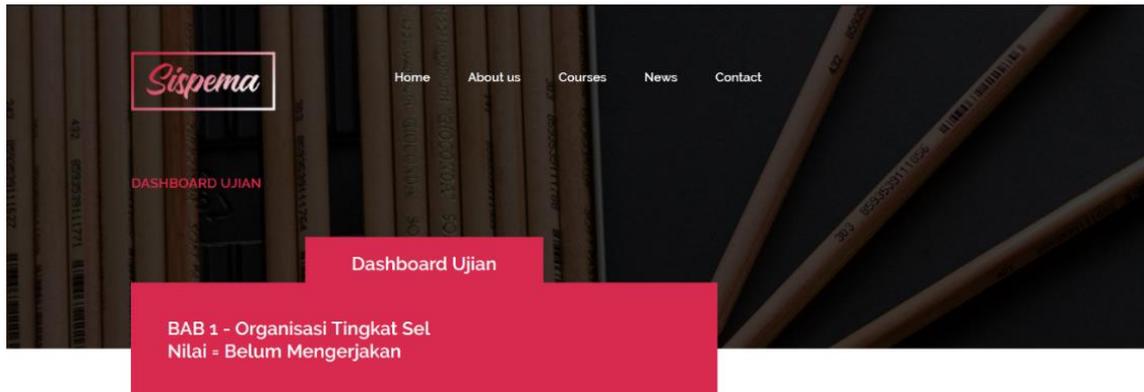
Pengguna akan diarahkan ke halaman materi jika pengguna meng-klik tombol materi yang terdapat pada halaman utama (Gambar 3.11). Halaman ini berisi keseluruhan materi yang sudah dibagi tiap bab. Halaman materi ditunjukkan oleh Gambar 3.12.



Gambar 3.12 Halaman Materi

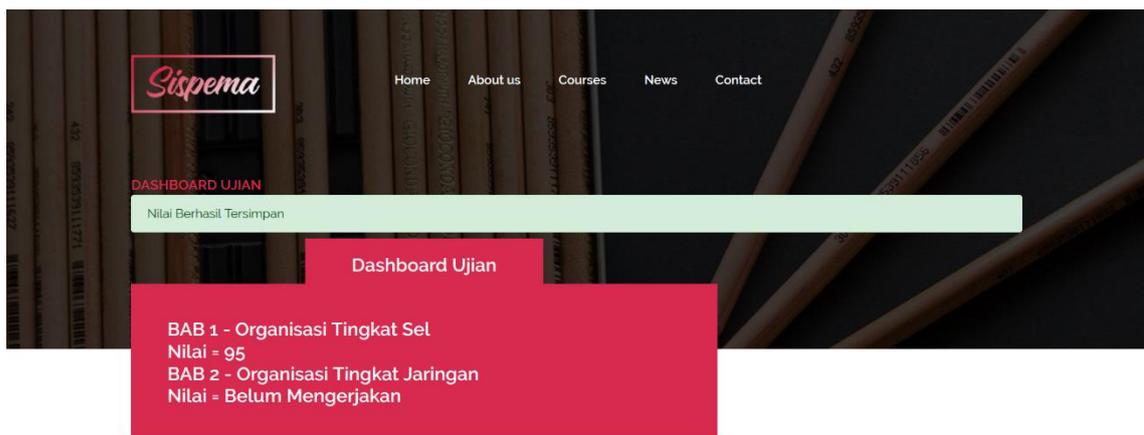
f. Halaman Ujian

Halaman ujian akan muncul apabila pengguna telah *log in* ke dalam sistem dan meng-klik tombol ujian. Halaman ini berisi daftar ujian pada masing-masing bab. Ketika pengguna baru yang belum pernah mengerjakan ujian, ujian yang akan muncul hanya ujian pada bab 1 seperti yang ditunjukkan pada Gambar 3.13.



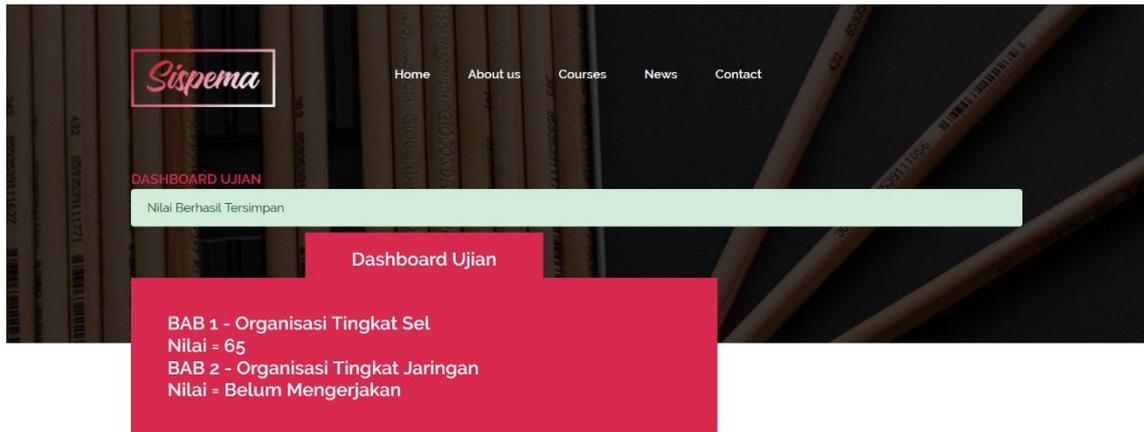
Gambar 3.13 Halaman Ujian 1

Ujian bab 2 akan muncul apabila pengguna telah mengerjakan ujian satu kali dan meraih nilai lebih dari 75 seperti yang ditunjukkan pada Gambar 3.14.



Gambar 3.14 Halaman Ujian 2

Bab 2 juga akan muncul bagi pengguna yang telah mengerjakan ujian sebanyak tiga kali dan meraih nilai kurang dari 75 seperti yang ditunjukkan pada Gambar 3.15.

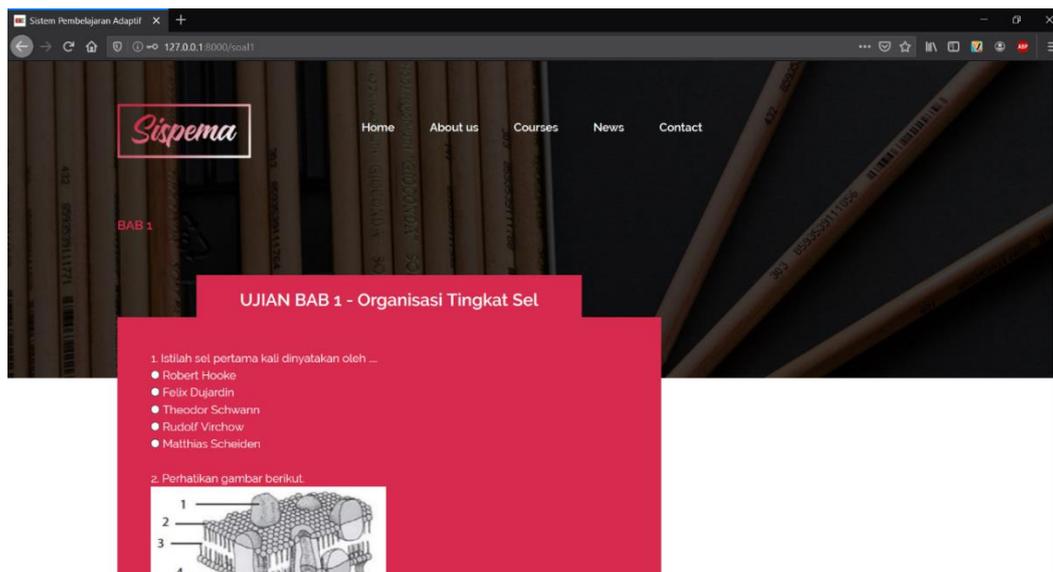


Gambar 3.15 Halaman Ujian 3

Hal ini berlaku juga untuk ujian-ujian pada bab selanjutnya. Nilai yang didapatkan dari hasil ujian ini akan langsung terkoneksi dengan lingkaran yang terdapat pada halaman utama. Apabila mendapatkan nilai lebih dari 75, lingkaran akan berubah warna menjadi hijau. Apabila mendapatkan nilai kurang dari 75, lingkaran akan berubah warna menjadi merah.

g. Halaman Soal

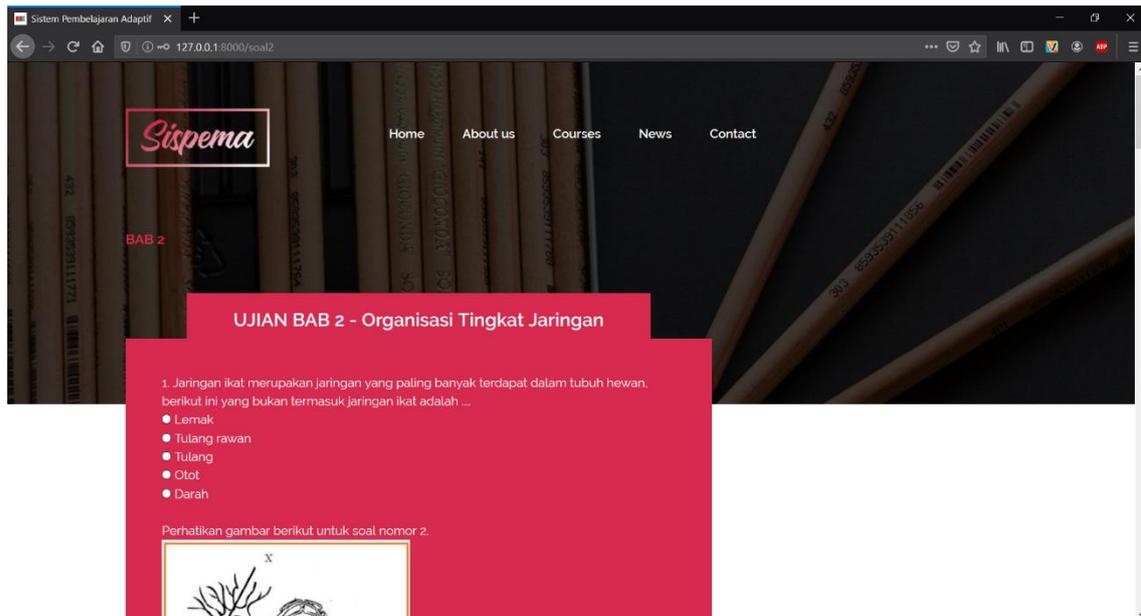
Halaman soal akan muncul apabila pengguna meng-klik salah satu bab pada halaman ujian. Soal ini terdiri dari 20 soal pada tiap bab. Contoh halaman soal ditunjukkan pada Gambar 3.16.



Gambar 3.16 Halaman Soal

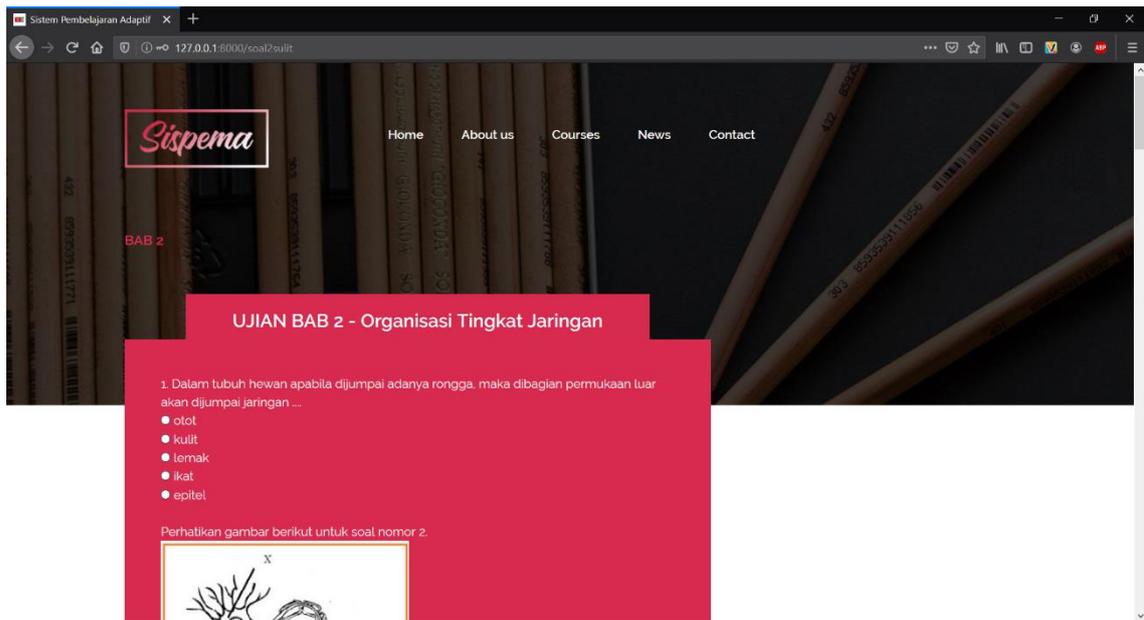
Apabila pengguna telah selesai mengerjakan soal pada halaman tersebut, pengguna bisa meng-klik tombol *submit* untuk mengirimkan jawaban pengguna ke dalam sistem.

Pengguna bisa diarahkan kepada soal dengan tingkat kesulitan tinggi maupun rendah. Hal ini didasari oleh nilai yang didapatkan pengguna pada bab sebelumnya. Apabila nilai yang didapatkan pada bab sebelumnya kurang dari 75, maka akan muncul soal dengan tingkat kesulitan rendah seperti yang ditunjukkan oleh Gambar 3.17.



Gambar 3.17 Soal Tingkat Kesulitan Rendah

Apabila nilai yang didapatkan pada bab sebelumnya lebih dari 75, maka akan muncul soal dengan tingkat kesulitan tinggi seperti yang ditunjukkan pada Gambar 3.18.



Gambar 3.18 Soal Tingkat Kesulitan Tinggi

Hal ini dengan pertimbangan bahwa nilai 75 adalah batas nilai agar siswa/pengguna tersebut dinyatakan lulus pada ujian tersebut. Pada Gambar 3.17 dan 3.18 ini adalah inti dari fungsi adaptif pada sistem ini. Hal ini didasari oleh pengetahuan (*knowledge*) pengguna. Pengetahuan (*knowledge*) ini diambil dari nilai pengguna pada bab sebelumnya. Apabila pengguna mendapatkan nilai lebih dari sama dengan 75, maka pengguna akan diberikan soal dengan tingkat kesulitan tinggi pada bab selanjutnya. Sedangkan apabila pengguna mendapatkan nilai kurang dari 75 dan telah mengerjakan soal tersebut sebanyak 3x, maka pengguna akan diberikan soal dengan tingkat kesulitan rendah.