

BAB II

ANALISIS DAN PERANCANGAN SISTEM

2.1 Analisis Sistem

Analisis sistem merupakan suatu tindakan untuk mengetahui lebih jauh tentang objek yang akan diteliti. Dengan demikian, akan diperoleh gambaran analisis kebutuhan berupa informasi serta perangkat yang digunakan untuk membuat sistem.

2.1.1 Analisis Sistem Informasi Perpustakaan Digital

Aplikasi yang akan dibuat menggunakan kebutuhan data yang meliputi *input*, proses, dan *output*. Penjelasan sebagai berikut.

- a. *Input* berupa data anggota perpustakaan, berkas buku digital berekstensi .swf.
- b. Proses berupa validasi login anggota perpustakaan, log tentang buku yang telah dibaca oleh anggota, dan log buku yang paling sering dibaca.
- c. *Output* meliputi hasil tampilan buku digital yang dapat dibaca melalui web browser, laporan anggota perpustakaan, laporan buku yang sering dibaca dan laporan buku yang tersedia

2.1.2 Kebutuhan Perangkat Lunak

Perangkat lunak yang digunakan dalam pembuatan aplikasi ini adalah Sistem Operasi Windows 7, SQLYog database manager, Dreamweaver, Sublime Text 3.

2.1.3 Kebutuhan Perangkat Keras

Perangkat keras yang digunakan untuk membuat aplikasi ini adalah Laptop dengan *Processor Intel Core i3*, 2,40 GHz, Ram 2 GB, HDD 500 GB , LCD 14 inch.

2.2 Perancangan Sistem

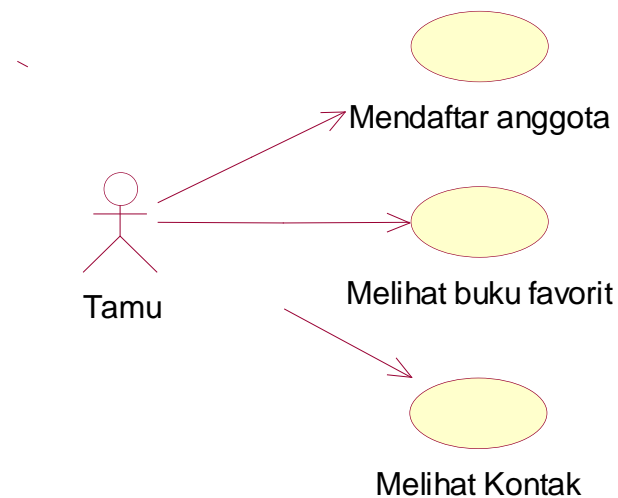
Sistem akan dibangun menggunakan bahasa pemrograman CodeIgniter yang berbasis MVC (*Model View Controller*) oleh karena itu untuk pemodelan sistem digunakan *Unified Modeling Language* (UML) yaitu bahasa spesifikasi standar untuk mendokumentasikan, menspesifikasikan dan membangun sistem perangkat lunak. UML menggambarkan himpunan yang terstruktur dan teknik untuk pemodelan desain program berorientasi objek (OOP) serta aplikasinya. Diagram yang akan digunakan sebagai pemodelan sistem adalah *use case diagram*, *class diagram* dan *activity diagram*.

2.2.1 . Use Case Diagram

Use case diagram diperlukan untuk menggambarkan fungsionalitas yang diharapkan dari perspektif pengguna. Hal yang ditekankan dalam *use case diagram* adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”.

Use case mempresentasikan sebuah interaksi antara aktor dengan sistem.

Pada gambar 3.1 digambarkan kegiatan yang dapat dilakukan oleh aktor tamu, dalam sistem merupakan pengguna umum yang belum melakukan pendaftaran sebagai anggota. Tamu dapat melakukan kegiatan antara lain: mendaftar anggota melihat buku favorit dan melihat kontak.

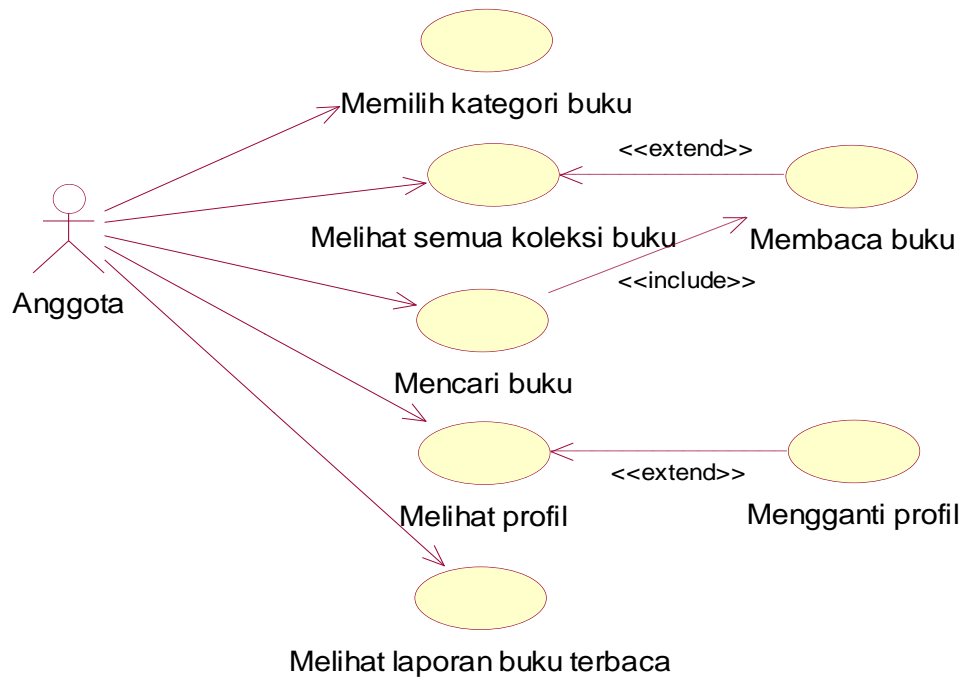


Gambar 2.1 *Use Case Diagram* Tamu

Sedangkan pada gambar 2.2 menggambarkan kegiatan yang dapat dilakukan oleh aktor anggota. Anggota memiliki relasi agregasi terhadap aktor tamu yang berarti semua kegiatan yang dilakukan oleh tamu dapat dilakukan oleh anggota, sedang kegiatan yang hanya dapat dilakukan oleh

anggota yakni kegiatan memilih kategori buku, melihat semua koleksi buku, melihat profil, mencari buku dan melihat laporan buku terbaca. Relasi *include* terjadi saat anggota mencari buku terhadap *use case* membaca buku.

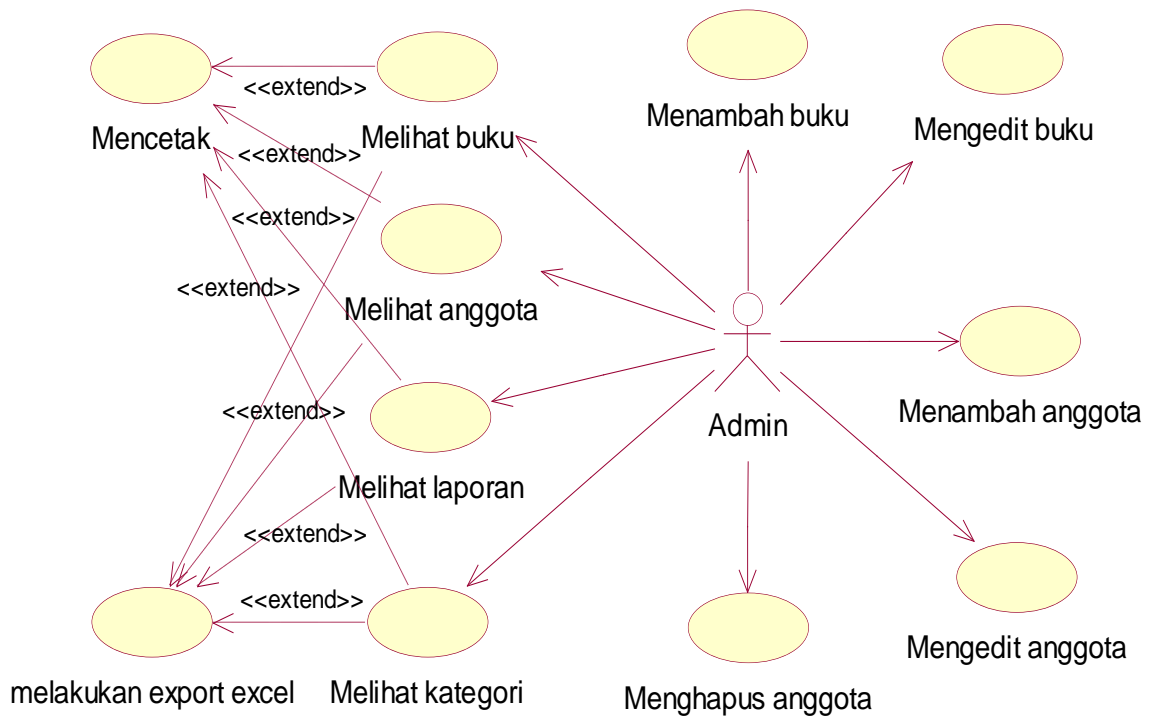
Relasi *extend* terjadi setelah anggota melihat profil, anggota dapat memilih untuk mengganti profil atau hanya melihat profil saja dan ketika melihat koleksi buku, anggota dapat memilih untuk membaca buku atau tidak.



Gambar 2.2 *Use Case Diagram Anggota*

Pada gambar 2.3 digambarkan kegiatan yang dilakukan oleh aktor admin. Admin dapat melakukan kegiatan antara lain: melakukan login, melihat buku, melihat member, melihat laporan, melihat kategori, menambah member, mengedit member, menghapus member, menambah buku, mengedit buku dan logout. Admin juga dapat memilih kegiatan mencetak dan

melakukan export excel, yang ditunjukkan dengan relasi extend pada kegiatan melihat buku, melihat member, melihat laporan dan melihat kategori.



Gambar 2.3 Use Case Diagram Admin

2.2.2 Class Diagram

Class diagram digunakan untuk menampilkan kelas-kelas dan paket-paket di dalam sistem. *Class diagram* menggambarkan relasi antar kelas dalam aplikasi serta atribut-atribut dan fungsi-fungsi yang melekat pada kelas. *Class diagram* pada lampiran 1 dapat juga menggambarkan rancangan MVC (*model view controller*) yang digunakan dalam sistem. Diagram tersebut membantu pengembang mendapatkan struktur sistem sebelum kode ditulis, dan membantu untuk memastikan bahwa sistem adalah desain terbaik.

Relasi asosiasi terjadi pada kelas Controller : Home dan kelas-kelas View : Daftar, FAQ, dan Login serta Model : Account. relasi tersebut terjadi karena fungsi yang ada pada kelas Home akan diteruskan ke kelas Account untuk koneksi database, sedangkan hasilnya akan ditampilkan pada kelas-kelas View : Daftar, FAQ dan Login.

Relasi asosiasi selanjutnya terjadi pada kelas login ke Controller : Member dan Admin. Jika relasi asosiasi menuju ke kelas Member maka Controller : Member akan berelasi dengan kelas View : Profil, Beranda, Baca, List, Laporan dan Model : RecordBaca. Sedangkan jika relasi pada kelas Login menuju ke kelas Admin maka Controller : Admin akan berelasi dengan kelas View : Grocery Crud dan Model Grocery Crud. Grocery Crud merupakan library dari pihak ketiga (*third party library*) yang berfungsi sebagai *template* untuk fungsi administrasi database.

2.2.3 Activity Diagram

Activity diagram memperlihatkan aliran kendali dari suatu aktivitas ke aktivitas lainnya. *Activity diagram* berfungsi untuk memvisualisasikan, menspesifikasi, mengkonstruksi, serta mendokumentasi sifat dari sekumpulan objek, selain itu dapat digunakan memodelkan aliran kendali dari suatu operasi. *Activity diagram* juga dapat disebut sebagai *flowchart* dari aplikasi, perbedaan dengan *flowchart* biasa dengan *activity diagram* yaitu *activity diagram* dapat bekerja secara paralel dengan adanya fungsi *synchronization*. *Activity diagram* pada aplikasi ini dapat dilihat pada lampiran 2 tentang

activity diagram tamu dan anggota dan lampiran 3 tentang *activity diagram* admin.

Kegiatan admin pada *activity diagram* admin dimulai dari mengisi username dan password untuk login, kemudian aplikasi akan mencari di dalam database data username dan password yang dimasukkan. Setelah melakukan validasi data, jika ditemukan maka kegiatan akan dilanjutkan dengan manajemen buku dan manajemen member. Kegiatan manajemen yang dimaksud adalah kegiatan untuk menghapus, mengedit, menambah dan/atau menyimpan data member atau buku.

Sedangkan kegiatan tamu dan anggota dimulai dari tamu menginputkan data diri (email, nama dan password) yang kemudian akan disimpan ke database. Kemudian tamu akan berubah menjadi anggota yang dapat melakukan login. Kegiatan selanjutnya yakni validasi login seperti yang terjadi pada *activity diagram* admin dan jika validasi gagal maka anggota akan kembali melakukan kegiatan mengisi username dan password untuk login. Jika validasi login berhasil, maka anggota dapat melakukan kegiatan mengganti profil, melihat semua koleksi buku, membaca buku dan melihat laporan buku yang terbaca.

2.3. Desain Struktur Basis Data

2.3.1 Struktur Tabel

Pada tahap ini dilakukan pendefinisian basis data yang akan disimpan, meliputi struktur penyimpanan data, format data, dan jalur akses. Dalam perancangannya juga dilakukan transformasi struktur data yang akan disimpan dengan membuat spesifikasi struktur tiap berkas data.

1. Desain tabel buku.

Desain tabel buku terdiri dari 11 field yaitu id_buku, judul, pengarang, tahun, edisi, isbn, penerbit, tgl_masuk, id_kategori, alamat, foto dan hit, berfungsi menginput dan menyimpan data buku.

Tabel 2.4 Desain tabelbuku

	Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?
<input type="checkbox"/>	id_buku	int	10		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input type="checkbox"/>	judul	varchar	25		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	pengarang	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	tahun	int	4		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	edisi	int	2		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	isbn	varchar	50		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	penerbit	varchar	150		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	tgl_masuk	date			<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	id_kategori	char	3		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	foto	text			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	alamat	varchar	250		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	hit	int	11	0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>					<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

2. Desain tabel kategori.

Desain tabel kategoriberfungsi menginput dan menyimpan data kategori, yang digunakan untuk mengelompokkan buku, field yang dilibatkan pada tabel ini meliputi id_kategori dan nama_kategori.

Tabel 2.5Desain tabel kategori.

	Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	A
<input type="checkbox"/>	id_kategori	char	3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
<input type="checkbox"/>	nama_kategori	varchar	100		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
<input type="checkbox"/>					<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

3. Desain tabel person.

Pada desain tabel personterdiri dari 4field (lihat tabel 2.6), berfungsi untuk menginput dan menyimpan biodata anggota dan petugasyang menggunakan sistem ini.

Tabel 2.6 Desain tabel person.

	Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?
<input type="checkbox"/>	id_person	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input type="checkbox"/>	nama_person	varchar	50		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	email	varchar	50		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	password	varchar	10		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>					<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

4. Tabel record baca

Tabel record bacapenyimpan data-data yang berasal dari proses baca buku oleh anggota, atribut dalam tabel record baca berupa id_record, id_person, id_buku, dan tgl_baca.

Tabel 2.7Tabel record baca

	Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?
<input type="checkbox"/>	id_record	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input type="checkbox"/>	id_person	int	11		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	id_buku	int	11		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	tgl_baca	date			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>					<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

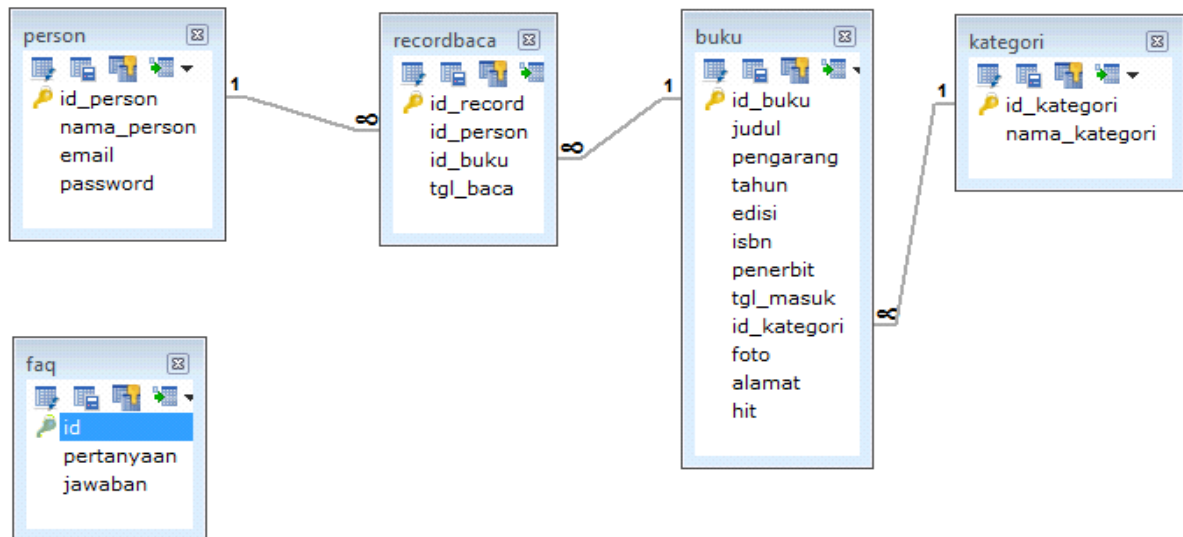
5. Tabel FAQ

Desain tabel faqberfungsi menampilkan pertanyaan dan jawaban yang sering muncul, fungsi FAQ adalah untuk anggota yang berisi petunjuk tentang sistem, atribut-atribut didalamnya berupa penempatan id, pertanyaan dan jawaban.

Tabel 2.8Tabel faq

	Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?
<input type="checkbox"/>	id	int	11		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input type="checkbox"/>	pertanyaan	text			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	jawaban	text			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>					<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

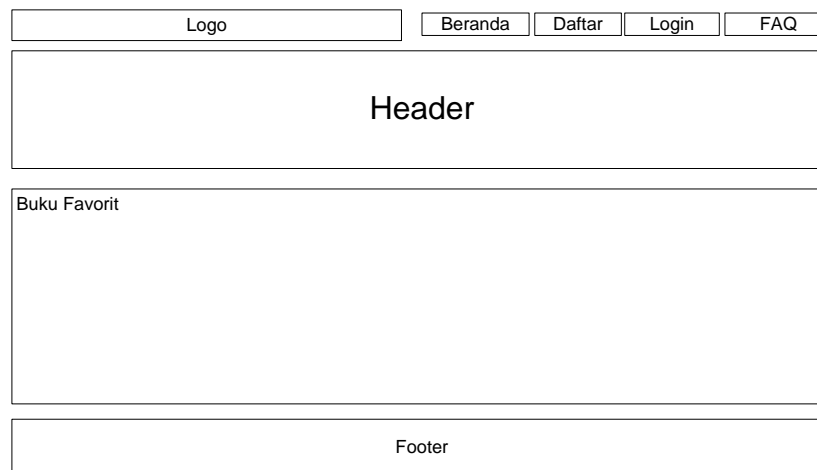
2.3.2. Relasi Antar Tabel



Gambar 2.9 Relasi Tabel

2.4. Perancangan Antarmuka

2.4.1 Tampilan Menu Beranda



Gambar 2.10 Tampilan Menu Beranda

2.4.2 Tampilan Menu Pendaftaran

Logo	Beranda	Daftar	Login	FAQ
------	---------	--------	-------	-----

Header

Daftar anggota

Nama

Email

Password

Konfirmasi

Footer

Gambar 2.11 Tampilan Menu Pendaftaran

2.4.3 Tampilan Menu Login

Logo	Beranda	Daftar	Login	FAQ
------	---------	--------	-------	-----

Header

Login

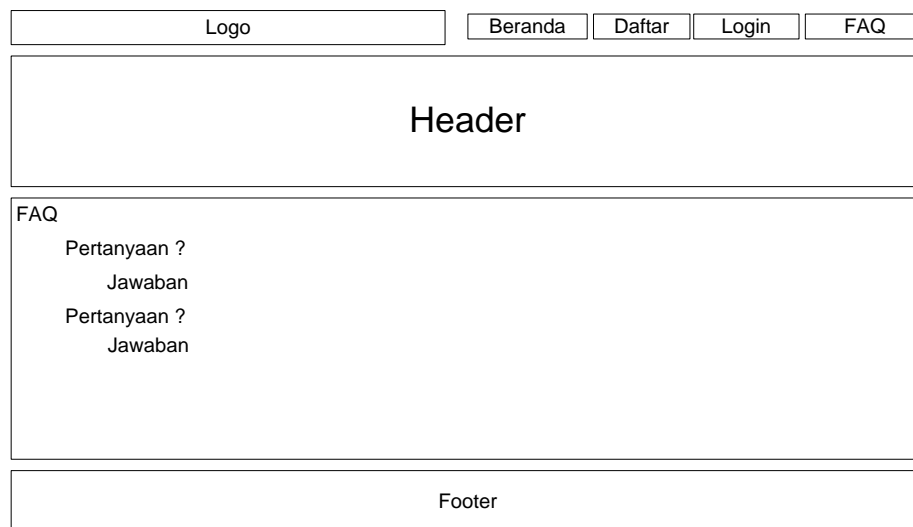
Username

Password

Footer

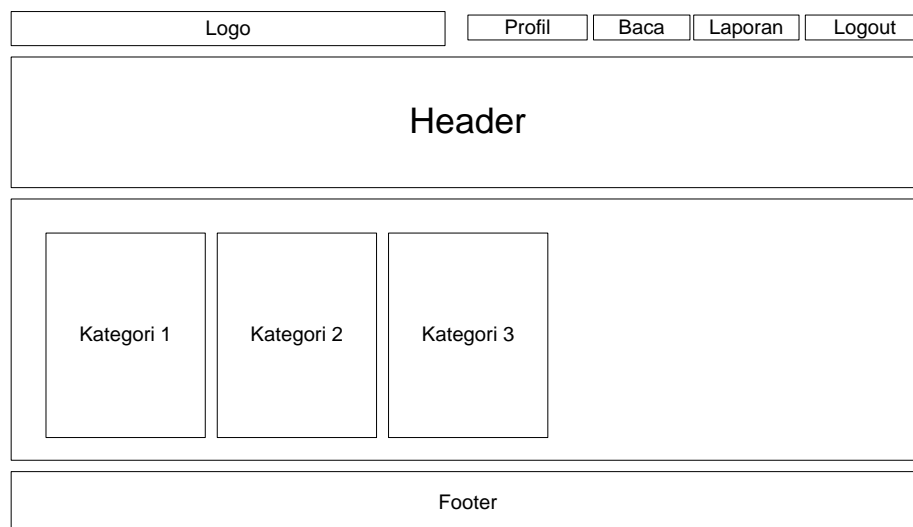
Gambar 2.12 Tampilan Menu Login

2.4.4 Tampilan Menu FAQ



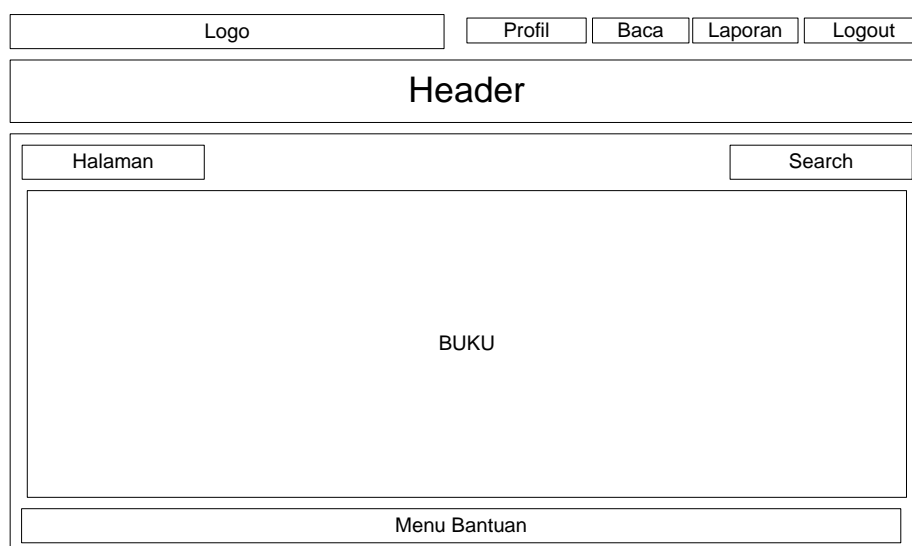
Gambar 2.13. Tampilan Menu FAQ

2.4.5 Tampilan Menu Kategori



Gambar 2.14. Tampilan Menu Kategori

2.4.6 Tampilan Menu Baca Buku



Gambar 2.15. Tampilan Menu Baca Buku