BAB III

ANALISIS DAN PERANCANGAN SISTEM

3.1. Gambaran Umum Sistem

Sistem yang akan dikembangkan ditujukan untuk dua pengguna, yaitu administrator, dan pengunjung. Masing-masing pengguna memiliki hak akses yang berbeda-beda. Untuk itu diperlukan suatu mekanisme keamanan, yaitu dengan menyediakan fasilitas *login* untuk mengetahui hak akses masing-masing pengguna. Bagi administrator agar memasukkan data yang diperlukan untuk login seperti *username* dan *password*. Bagi pengunjung perpustakaan, tidak diberi *username* atau *password,* karena siapa saja pengunjung dapat mengakses data perpustakaan. Admin untuk administrator adalah seseorang yang ditunjuk untuk mengelola sistem ini.

Sistem Automasi Perpustakaan yang akan diimplementasikan meliputi subsistem peminjaman, subsistem pengembalian, dan subsistem klasifikasi bahan bacaan.

Subsistem peminjaman bertujuan mempermudah petugas perpustakaan dalam proses peminjaman. Sama halnya dengan subsistem pengembalian. Sedangkan subsistem klasifikasi akan mempermudah petugas dalam mengklasifikasi bahan bacaan apa saja yang ditempatkan pada koleksi perpustakaan. Untuk subsistem klasifikasi dan pengadaan bahan bacaan hanya dapat diakses oleh administrator, demi keamanan data yang tersimpan.

3.2. Analisis Sistem

Untuk analisis sitem akan ada beberapa point antara lain:

3.2.1. Analisis Masalah dari Sistem yang berjalan

Dilihat dari sistem yang berjalan saat ini, proses sirkulasi informasi pada perustakaan Salman masih belum efektif dalam pembuatan laporan secara berkala, peminjaman yang melalui proses yang rumit dan belum ada keputusan dalam pengadaan bahan bacaan. Semua proses yang dilakukan masih manual, sedangkan komputer hanya digunakan untuk menyimpan data koleksi perpustakaan. Proses manual seperti itu dirasa belum efektif dan akurat karena proses peminjaman yang dilakukan dan koleksi buku perpustakaan sangat banyak, sehingga diperlukan sebuah sistem untuk membantu semua proses manual tersebut. Dan salah satu solusinya adalah Sistem Automasi Perpustakaan.

Sehingga muncul beberapa masalah, yang kemudian bisa diselesaikan dari pembangunan sistem ini. Diantaranya :

1. Bagaimana agar pembuatan laporan secara berkala dapat lebih efektif.
   Sistem yang sedang berjalan sekarang di Perpustakaan Salman masih menggunakan sistem manual. Akan sulit untuk membuat laporan peminjaman per-bulan bahkan per-tahun bagi petugas dengan tenaga yang sedikit. Dengan sistem ini diharapkan dapat lebih mengoptimalkan hasil laporan yang diperlukan.
2. Bagaimana agar peminjaman pada perpustakaan dapat dilakukan secara digital.

Tidak ada masalah sebenarnya dalam proses peminjaman secara manual, tetapi akan repot nantinya dalam pembuatan laporan dengan peminjaman yang banyak. Untuk itu diharapkan dari sistem ini agar dapat memberikan kinerja yang optimal dalam proses peminjaman maupun dalam pembuatan laporan.

3. Bagaimana agar klasifikasi bahan bacaan dapat terorganisir dengan baik.

   Dalam pelaksanaannya, pengklasifikasian buku masih dilakukan dengan cara melihat kategori buku pada acuan buku Dewey. Waktu yang lama akan diperlukan untuk memberikan kode klasifikasi pada buku baru atau buku yang didapat dari sumbangan. Ditambah dengan tidak adanya acuan dalam bentuk digital. Untuk mengatasi hal itu diperlukan adanya digitalisasi kode Dewey agar memudahkan petugas dalam pengklasifikasian buku.

3.2.2. Analisis Kebutuhan Sistem Non Fungsional

Analisis kebutuhan sistem non fungsional adalah sebuah langkah dimana seorang pembangun perangkat lunak menganalisis sumber daya manusia yang akan menggunakan perangkat lunak yang dibangun.

Analisis kebutuhan non fungsional tidak hanya menganalisis siapa saja yang akan menggunakan aplikasi tetapi juga menganalisis perangkat keras dan perangkat lunak yang dimiliki oleh instansi, sehingga dapat ditentukan kompabilitas aplikasi yang dibangun terhadap sumber daya yang ada. Setelah melakukan analisis kebutuhan non fungsional, maka dilanjutkan ke langkah

berikutnya yaitu menentukan kebutuhan non fungsional sistem yang akan dibangun untuk disesuaikan dengan fakta yang ada.

Analisis non fungsional dan kebutuhan non fungsional yang dilakukan dibagi dalam empat tahap, yaitu:

1. Analisis perangkat keras,
2. Analisis perangkat lunak,
3. Analisis pengguna sistem atau *user*,
4. Analisis pengkodean.

3.2.2.1. Analisis Perangkat Keras

Dengan melihat berbagai permasalahan di atas diperlukan diperlukan alat spesifikasi perangkat keras (*hardware*) untuk mendukung sistem yang akan dibangun.

1. Kebutuhan perangkat keras

   Kebutuhan perangkat keras untuk *server* yang dimiliki oleh perpustakaan Salman adalah sebagai berikut:

   a. *Processor* dengan kecepatan 1 Ghz
   b. RAM 512 MB
   c. *Hard Disk* 80 GB
   d. Monitor dengan resolusi 1024 x 768
   e. Keyboard dan mouse standar

Setelah dilakukan analisis terhadap perangkat keras di Perpustakaan Salman ITB, ternyata perangkat keras yang akan digunakan untuk

mengimplementasikan aplikasi Sistem Automasi Perpustakaan telah memenuhi spesifikasi minimum perangkat keras.

3.2.2.2 Analisis Perangkat Lunak

Di Perpustakaan Salman ITB terdapat beberapa proses dengan menggunakan perangkat lunak komputer, yaitu pemasukan data buku baru, pemasukan data sumbangan buku dengan menggunakan perangkat lunak sebagai berikut:

1. Processore dengan kecepatan 2,00 GHz
2. RAM 256 Mb
3. Monitor 14” beresolusi 1024x768
4. Keyboard dan mouse standar

Agar dapat memenuhi kebutuhan perangkat lunak pada system yang dibangun, maka diperlukan tambahan perangkat lunak lainnya, antara lain:

1. *Macromedia Dreamweaver 8*
2. Wamp server
3. *Web Browser*

3.2.2.3. Analisis Pengguna Sistem atau User

Analisis *user* dimaksudkan untuk mengetahui siapa saja *user* yang terlibat dalam Sistem Automasi Perpustakaan beserta karakteristiknya sehingga dapat diketahui tingkat pemahaman *user* terhadap komputer. Sistem yang sedang berjalan melibatkan satu user, yaitu :

1. Staff IT/Admin

Pendidikan : S1

Sistem Operasi yag digunakan : *Windows XP/ 7/ Vista*

Software yang digunakan : *MySql, Adobe Reader 8.0*

Job Description : Admin dapat melakukan semua fungsi yang ada dalam aplikasi ini.

3.2.2.3.1. Karakteristik Pengguna Sistem

Sistem Automasi Perpustakaan ini akan digunakan oleh dua pengguna, yaitu:

1. Staff IT/Admin

   Pendidikan : S1

   Sistem Operasi yag digunakan : *Windows XP/7*

   Software yang digunakan : *MySql, Adobe Reader 8.0*

   Job Description : Admin dapat melakukan semua fungsi yang ada dalam aplikasi ini.

2. Pengunjung

   Pengunjung dapat mencari koleksi perpustakaan tanpa harus memiliki akun, karena laman untuk pengunjung ini hanya untuk mencari dan melihat data koleksi perpustakaan, data peminjaman, data angota dan data pengembalian tanpa dapat melakukan perubahan di dalamnya.

Pengguna	Tanggung Jawab	Hak akses	Tingkat Pendidikan	Tingkat Keterampilan	Pengalaman	Jenis Pelatihan
Admin	Dapat menambah, mengubah dan mencari data-data yang ada pada aplikasi	Berinteraksi dengan layar komputer	Minimal lulusan D3	Mengerti teknik *maintenance* aplikasi yang berbasis web	Pelatihan Admin	Cara *maintenance* aplikasi yang berbasis web
Pengunjung	Hanya dapat melihat tanpa dapat menambah atau mengubah data-data yang ada pada aplikasi	Berinteraksi dengan layar komputer	SMP, SMA, Perguruan Tinggi, Umum	Bisa mengikuti petunjuk	-	-

Tabel 3.1 Karakteristik Pengguna

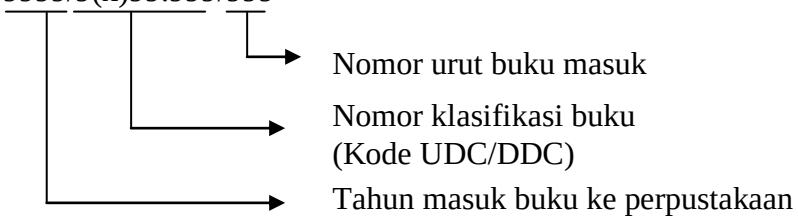
3.2.2.4. Karakteristik Pengkodean

Pada proses pengkodean di Perpustakaan Salman ITB, terdapat beberapa jenis pengkodean, antara lain:

a. Pengkodean nomor buku

Pada pengkodean buku format yag digunakan adalah sebagai berikut:

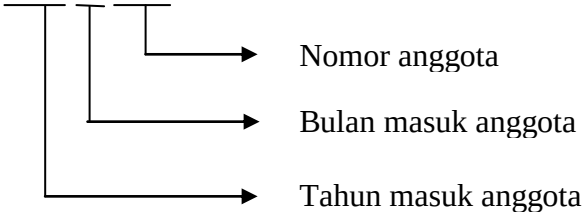
Format nomor buku : 9999/9(x)99.999/999



Contoh : 2010/175.101/516 → buku tersebut masuk pada tahun 2010 dengan nomor klasifikasi 175.101 dan nomor urut 516

b. Pengkodean nomor kartu

Format nomor kartu : 9999 99 999

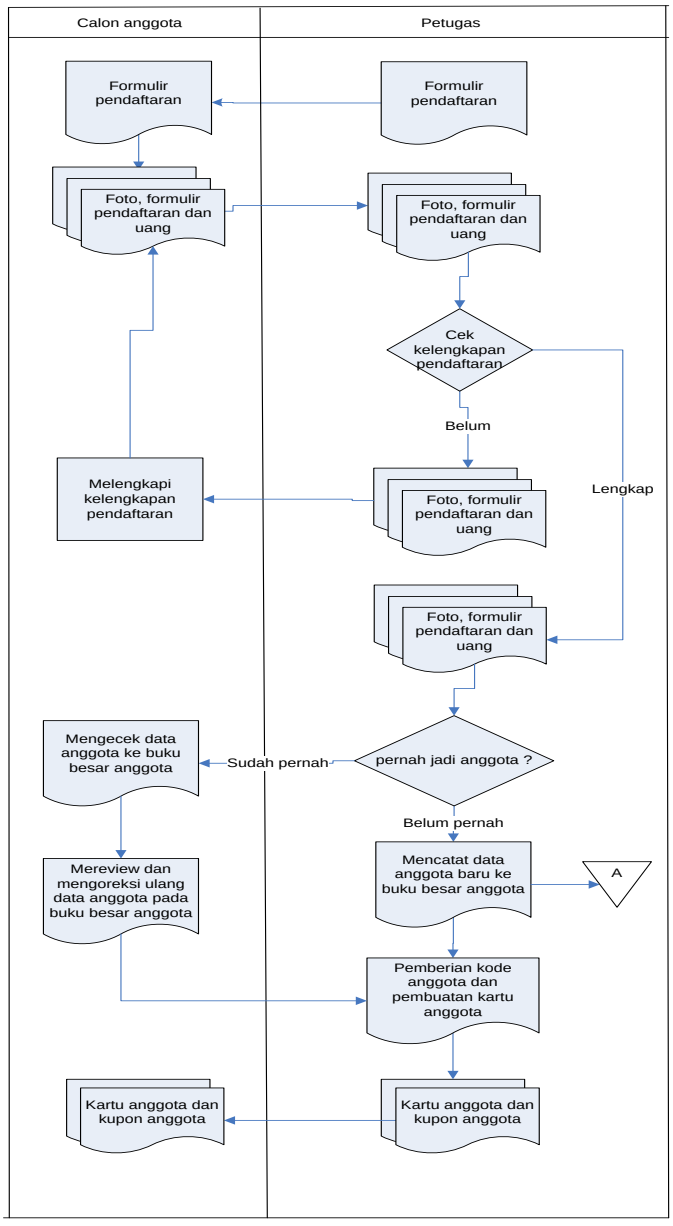


Contoh : 2010/10/112 → anggota tersebut masuk pada tahun 2010, bulan Oktober dan nomor urutnya 112.

Format nomor pendaftaran anggota, dicatat berdasarkan urutan orang yang mendaftar pada bulan tersebut karena setiap bulan nomor akan diulang dari awal lagi. Disediakan 3 digit karena pernah terjadi dalam satu bulan orang yang mendaftar menjadi anggota perpustakaan hingga mencapai lebih dari seratus orang.

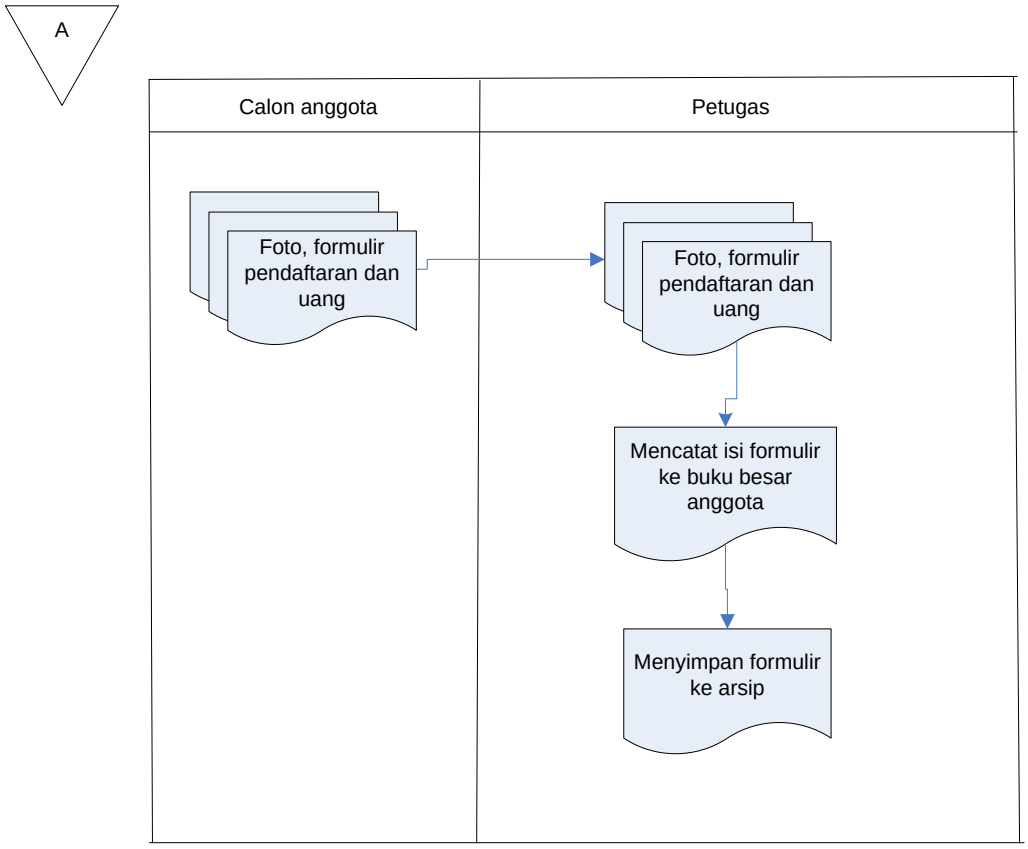
3.2.3. Prosedur Pendaftaran pada proses yang sedang berjalan DFD

Pada proses prosedur pendaftaran ini akan ada dokumen yang masuk ke arsip, yaitu pada proses pengecekan apakah calon anggota sudah pernah menjadi anggota sebelumnya atau tidak. Jika belum pernah maka data calon anggota akan langsung dimasukkan ke arsip buku besar.



Gambar 3.1 *Flow Map* Prosedur Pendaftaran

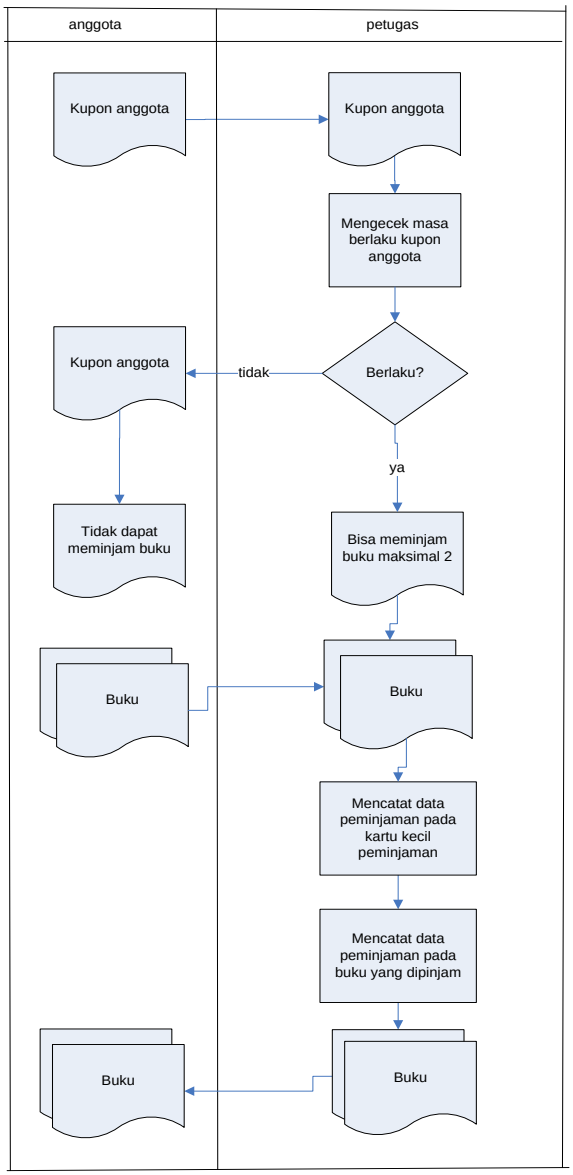
3.2.4. Arsip



Gambar 3.2 *Flow Map* Arsip pendaftaran

3.2.5. Prosedur Peminjaman dengan 1 kupon

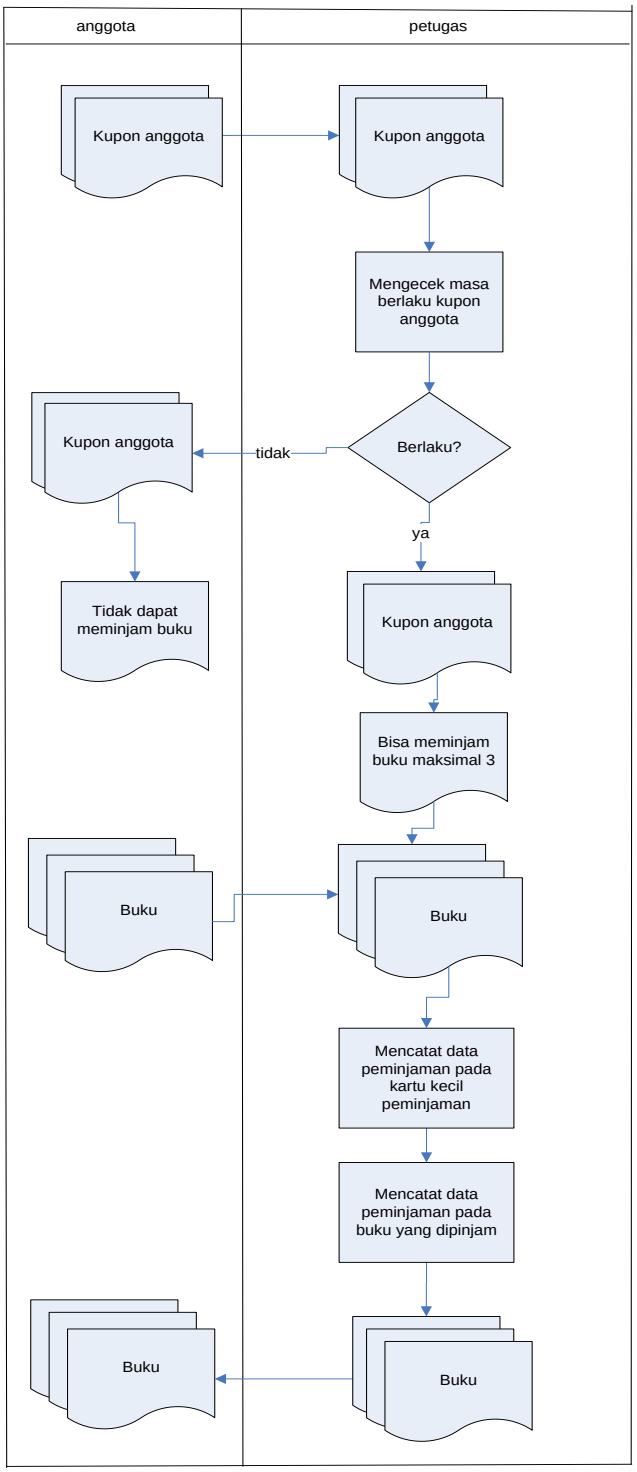
Pada perpustakaan Salman ITB, setelah mendapat kartu, anggota juga mendapat kupon, yaitu sejenis amplop kecil berjumlah dua buah. Kegunaan kupon tersebut adalah dapat digunakan untuk meminjam buku. Jika kupon tersebut hilang, maka anggota tidak dapat meminjam buku walaupun ia mempunyai kartu anggota dan statusnya masih aktif sebagai anggota.



Gambar 3.3 *Flow Map* Prosedur Peminjaman dengan 1 kupon

3.2.6. Prosedur Peminjaman dengan 2 kupon

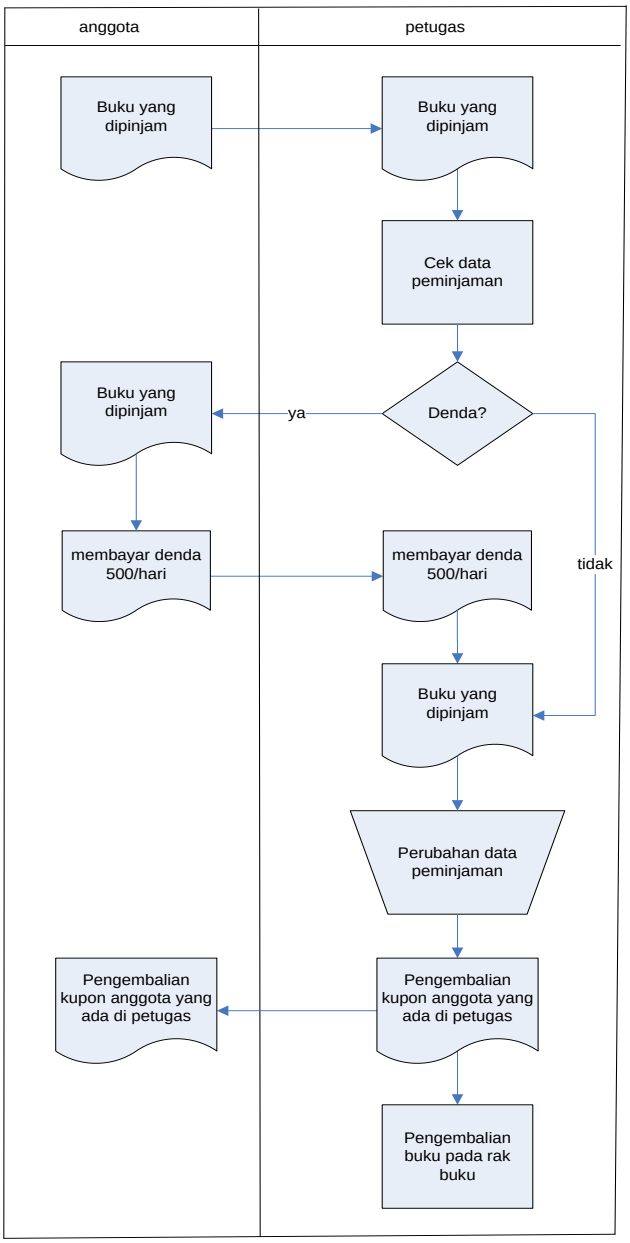
Berikut adalah *flow map* dari prosedur peminjaman dengan dua kupon.



Gambar 3.4 *Flow Map* Prosedur Peminjaman dengan 2 kupon

3.2.7. Prosedur Pengembalian

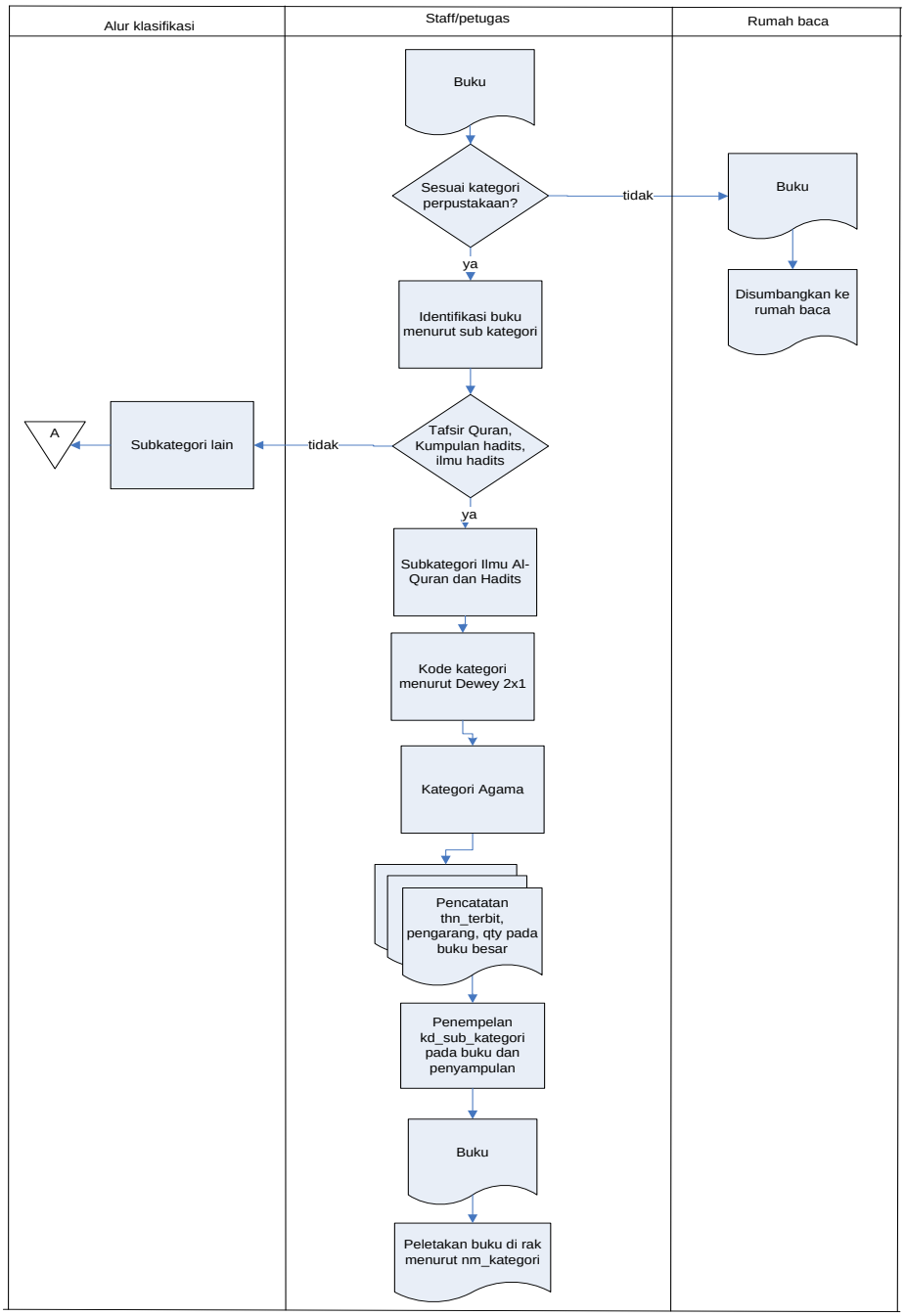
Untuk *fow map* pada prosedur pengembalian, aka nada proses hitung denda. Dengan ketentuan, syarat dan peraturan perpustakaan maka denda yang dikenakan untuk buku yang terlambat dikembalikan adalah Rp 500/hari. Denda ini berlaku untuk semua jenis buku yang dipinjam.



Gambar 3.5 *Flow Map* Prosedur Pengembalian

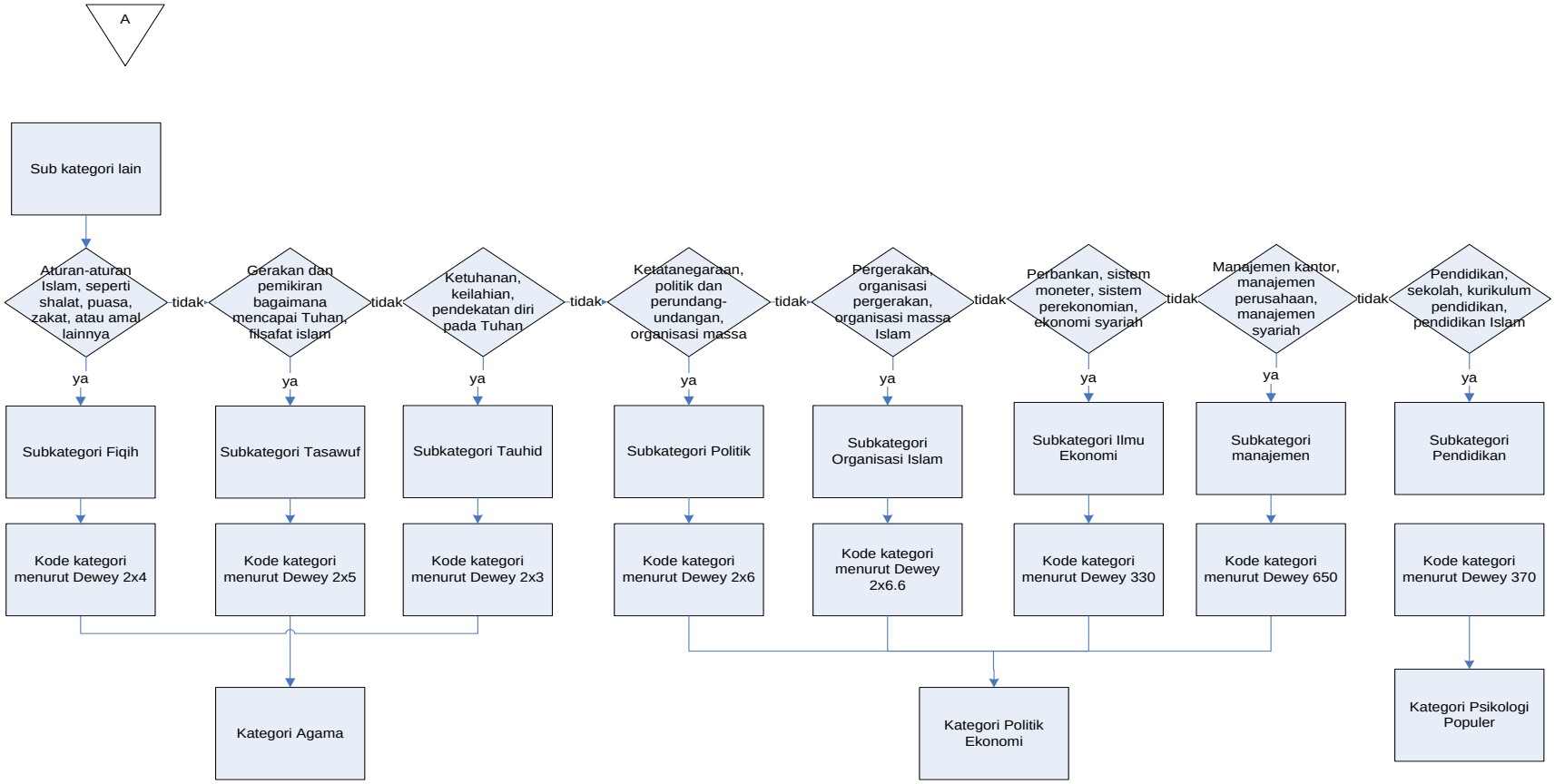
3.2.8. Prosedur Klasifikasi

Pada *flow map* prosedur klasifikasi, akan ada proses yang masuk arsip, yaitu proses identifikasi buku menurut subkategori. Hal ini dikarenakan subkategori yang ada pada perpustakaan Salman yang cukup banyak.



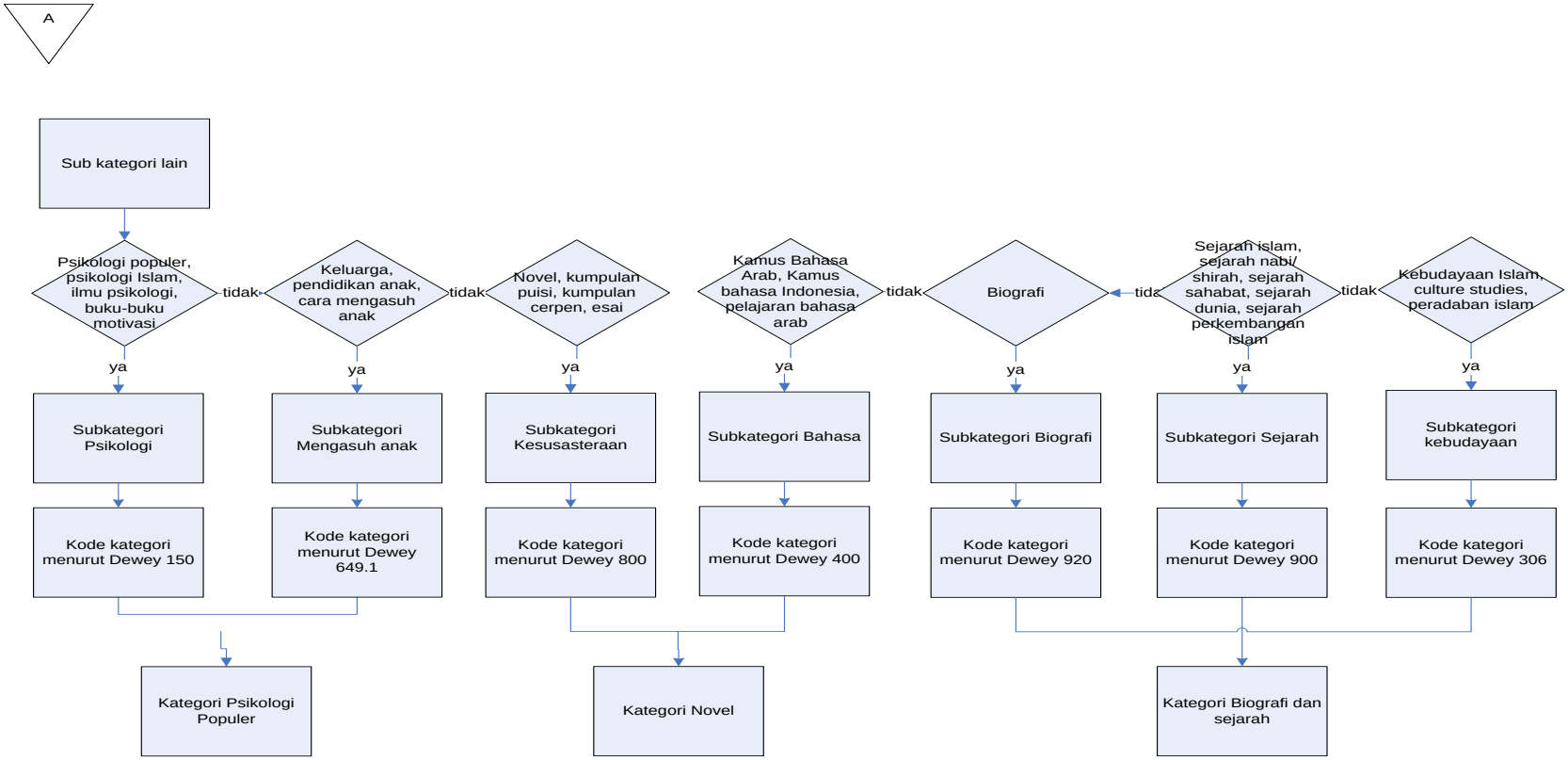
Gambar 3.6 *Flow Map* Prosedur Klasifikasi

3.2.9. Arsip Klasifikasi 1



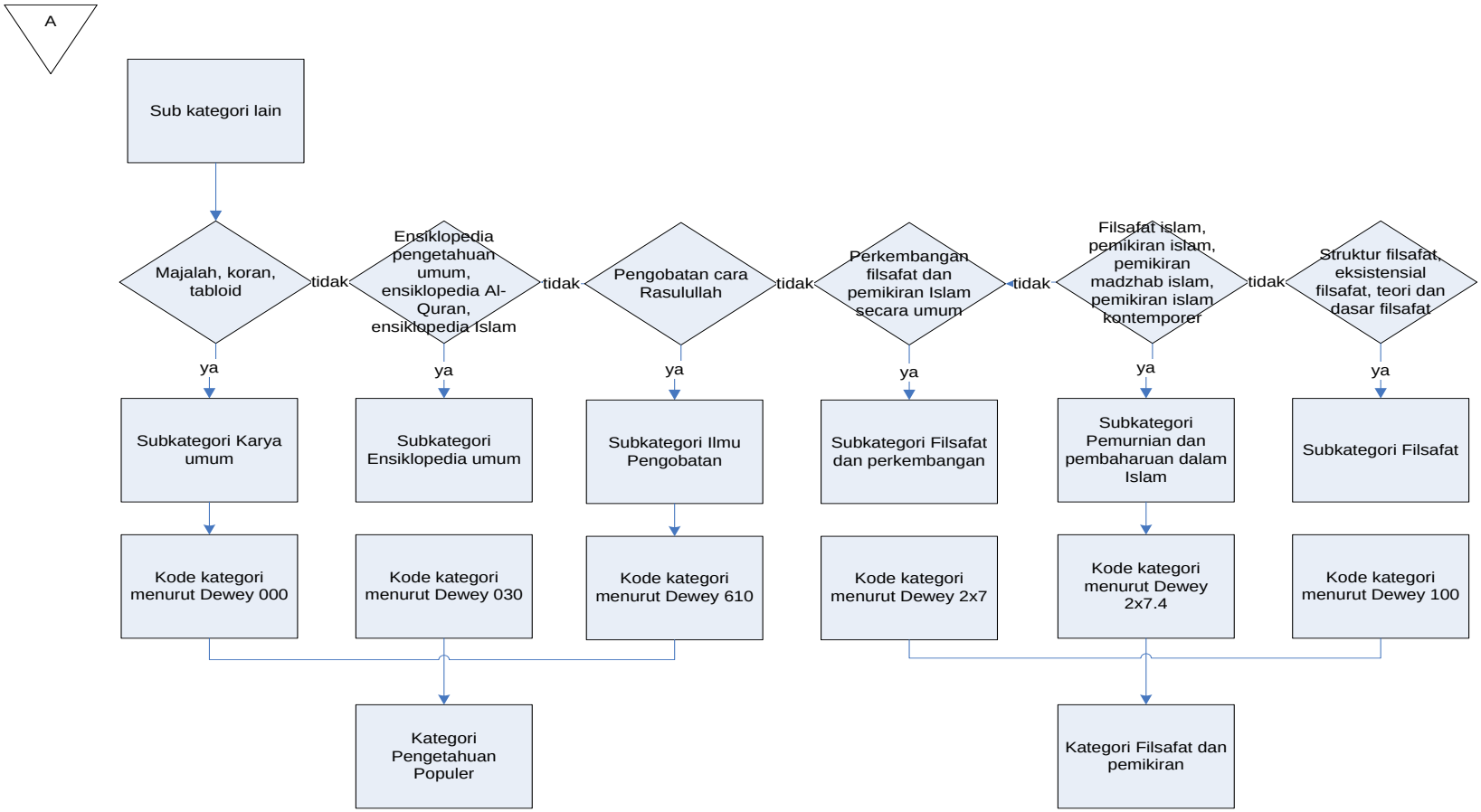
Gambar 3.7 Arsip Klasifikasi 1

3.2.10. Arsip Klasifikasi 2



Gambar 3.8 Arsip Klasifikasi 2

3.2.11. Arsip Klasifikasi 3



Gambar 3.9 Arsip Klasifikasi 3

3.2.12. Analisis metode Text Mining dalam kategorisasi buku

Contoh 1:

Pada metode Text Mining ada 5 tahap yang harus dilalui, antara lain *Tokenizing, Filtering, Stemming, Tagging*, dan *Analyzing*. Berikut diagram alir dari tahap *text mining* pada aplikasi:



Gambar 3.10 Flowchart algoritma *text mining*

Berikut algoritma text mining yang digunakan:

Get sinopsis

Token (sinopsis) // memisahkan sesuatu kumpulan kata menjadi array berupa kata-kata

if (kata(n) = penting // proses filtering yaitu pengambilan kata penting dengan cara membandingkannya dengan tabel aturan produksi.

then

return kata (n)

else

kata (n) = null; // kata tidak penting dibuang

end if

if (kata(n)! = kata dasar) // jika kata tersebut bukan kata dasar

then

stemming (kata(n)) // membuang imbuhan yang terdapat pada kata

if (stemming(kata(n))! = kata dasar) // proses tagging

then

tag (kata(n))

return (kata(n))

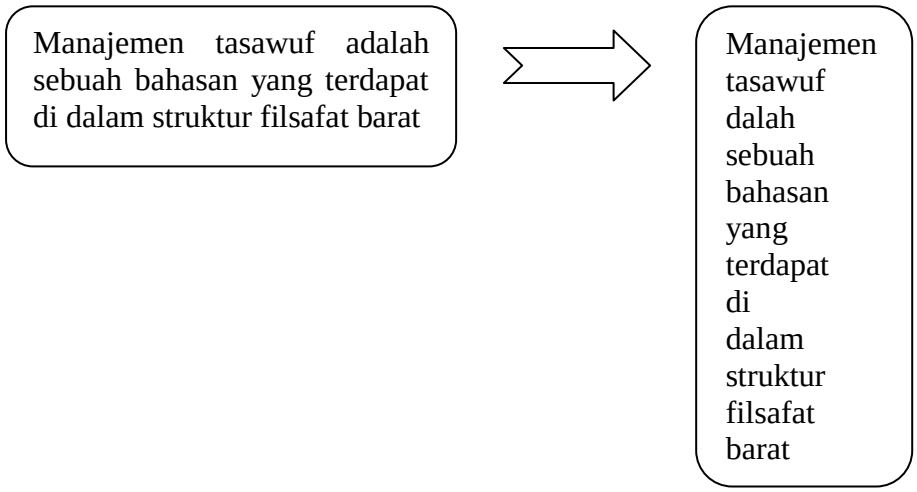
else

return(kata(n))

endif

analyzing (kata(n)) //penentuan nilai kata penting berdasarkan banyak kata yang keluar dari sinopsis kemudian dikategorikan.

Contoh dari tahap tokenizing adalah sebagai berikut:



Token (sinopsis) // memisahkan sesuatu kumpulan kata menjadi array berupa kata-kata

if (kata(n) = penting // proses filtering yaitu pengambilan kata penting dengan cara membandingkannya dengan tabel aturan produksi.

then

return kata (n)

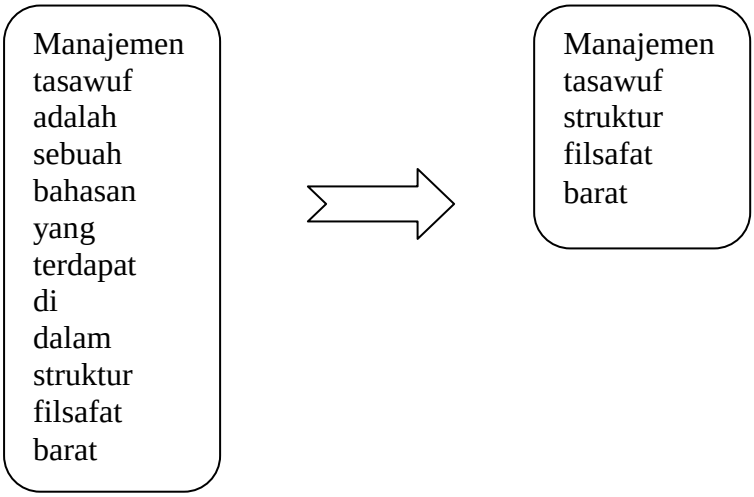
else

kata (n) = null; // kata tidak penting dibuang

end if

Untuk kata penting yang digunakan adalah acuan yang digunakan untuk mengklasifikasi buku baru seperti untuk subkategori Filsafat dan perkembangan maka acuan kata pentingnya adalah perkembangan filsafat, pemikiran Islam secara umum. Selain kata penting tersebut maka kata tidak penting seperti ‘adalah’, ‘sebuah’, dan ‘yang’ tidak akan digunakan.

Contoh dari tahap filtering adalah sebagai berikut:



if (kata(n) = penting // proses filtering yaitu pengambilan kata penting dengan cara membandingkannya dengan tabel aturan produksi.

then

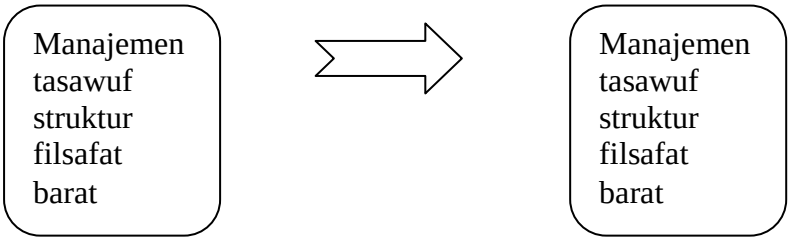
return kata (n)

else

kata (n) = null; // kata tidak penting dibuang

end if

Contoh dari tahap stemming adalah sebagai berikut:




```
if (kata(n)! = kata dasar) // jika kata tersebut bukan kata dasar
then
        stemming (kata(n)) // membuang imbuhan yang terdapat pada kata
        if (stemming(kata(n))! = kata dasar) // proses tagging
        then
                tag (kata(n))
                return (kata(n))
        else
                return(kata(n))
        endif
```

Untuk tahapan selanjutnya adalah tahap *Tagging* tetapi tidak dipakai dala sistem ini. Hal ini dikarenakan bahasa Indonesia tidak memiliki bentuk lampau.

Tahap terakhir adalah tahap *analyzing*, yaitu tahap penentuan seberapa jauh keterhubungan antar kata-kata pada dokumen yang ada.

analyzing (kata(n)) //penentua n nilai kata penting berdasarkan banyak kata yang keluar dari sinopsis kemudian dikategorikan.

Pada tahap *analyzing* ini digunakan sebuah rumus. Formula yang digunakan untuk menghitung bobot (w) masing-masing dokumen terhadap kata kunci adalah:

$$W_{d,t} = tf_{d,t} * IDF_t$$

Dengan :

d = dokumen ke-d

t = kata ke-t dari kata kunci

$W_{d,t}$ = bobot dokumen ke-d terhadap kata ke-t

Setelah bobot (w) masing-masing dokumen diketahui maka dilakukan proses *sorting*/pengurutan dimana semakin besar nilai w, semakin besar tingkat similaritas dokumen tersebut terhadap kata yang dicari, demikian sebaliknya. Contoh implementasi sederhana dari *TF-IDF* adalah sebagai berikut:

Diketahui:

1. Dokumen 1 (D1) = kategori Politik Ekonomi dengan subkategori Manajemen
2. Dokumen 2 (D2) = kategori Filsafat dan Pemikiran dengan subkategori Filsafat
3. Dokumen 3 (D3) = kategori Filsafat dan Pemikiran dengan subkategori Filsafat dan Perkembangan Islam
4. Kata kunci (kk) = kata kunci yang dimasukkan ke dalam sistem

Kata Kunci (kk) = Struktur Filsafat

Dokumen 1 (D1) = Manajemen Struktur

Jadi jumlah dokumen (D) = 3

Dokumen 2 (D2) =Filsafat Barat

Pada perhitungan *TF/IDF* ini menggunakan metode *word list*, sehingga proses

Dokumen 3 (D3) = Dalam manajemen tasawuf terdapat struktur filsafat

atau langkah yang dilakukan hanya sampai tahap *Tokenizing*, karena itu hasil

Token akan dipakai pada tabel perhitungan.

Di bawah ini adalah contoh tabel perhitungan TF/IDF:

token	**Tf**				**df**	**D/df**	**IDF** **log(D/f)**	**W**			
	kk	**D1**	**D2**	**D3**				**kk**	**D1**	**D2**	**D3**
Manajemen	0	1	0	1	2	1.5	0.176	0	0.176	0	0.176
Struktur	1	1	0	1	2	1.5	0.176	0.176	0.176	0	0.176
filsafat	1	0	1	1	2	1.5	0.176	0.176	0	0.176	0.176
Barat	0	0	1	0	1	3	0.477	0	0	0.477	0
tasawuf	0	0	0	1	1	3	0.477	0	0	0	0.477
Total								0.352	0.352	0.653	1.005

Tabel 3.2 Perhitungan *TF/IDF* Contoh 1

tf = banyak kata yang dicari pada sebuah dokumen

D = Dokumen

df = banyak dokumen yang mengandung kata yang dicari

w = bobot dokumen ke-d terhadap kata ke-t

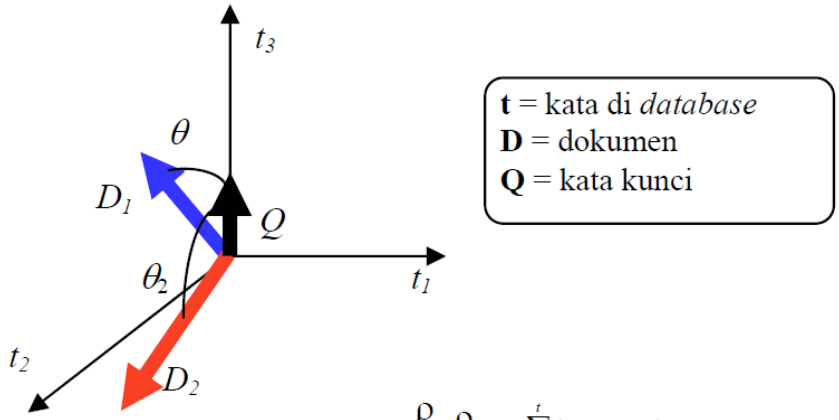
Bobot (w) untuk D1 = 0.176 + 0 = 0.176

Bobot (w) untuk D2 = 0 + 0.176 = 0.176

Bobot (w) untuk D3 = 0.176 + 0.176 = 0.352

Apabila diurutkan maka proses *sorting* juga tidak akan dapat mengurutkan secara tepat, karena nilai w keduanya sama. Untuk mengatasi hal ini, algoritma dari *vector-space model* adalah jawabannya

Ide dari metode ini adalah dengan menghitung nilai cosinus sudut dari dua vektor, yaitu W dari tiap dokumen dan W dari kata kunci.



Rumus: $\text{CosSim}(\vec{d_j}, \vec{q}) = \frac{\vec{d_j} \cdot \vec{q}}{|\vec{d_j}| \cdot |\vec{q}|} = \frac{\sum_{i=1}^{t}(w_{ij} \cdot w_{iq})}{\sqrt{\sum_{i=1}^{t} w_{ij}^2} \cdot \sqrt{\sum_{i=1}^{t} w_{iq}^2}}$

Apabila studi kasus di atas dicari nilai cosinus sudut antara vektor masing-masing dokumen dengan kata kunci, maka hasil yang didapatkan akan lebih presisi.

Berikut adalah contoh tabel perhitungan *Vektor Space Model:*

token	**Kk^2**	**D1^2**	**D2^2**	**D3^2**	**Kk*D1**	**Kk*D2**	**Kk*D3**
Manajemen	0	0.031	0	0.031	0	0	0
Struktur	0.031	0.031	0	0.031	0.031	0	0.031
filsafat	0.031	0	0.031	0.031	0	0.031	0.031
Barat	0	0	0.228	0	0	0	0
tasawuf	0	0	0	0.228	0	0	0

	Sqrt(kk)	Sqrt(Di)			Sum(kk dot Di)		
	0.249	0.249	0.509	0.567	0.031	0.031	0.062

Tabel 3.3 Perhitungan *Vektor-Space Model* Contoh 1

Kk^2 = kata kunci pada tabel *TF/IDF* dikuadratkan

D = dokumen

Sqrt = *Squareroot* atau akar dari hail penjumlahan

Selanjutnya menghitung nilai Cosinus sudut antara vektor kata kunci dengan tiap dokumen dengan rumus:

$$\text{Cosine } (D_i) = \text{sum } (kk \text{ dot } D_i) / [\text{sqrt}(kk) * \text{sqrt}(D_i))]$$

Cosine (Di) = sum (kk dot Di) / [sqrt (kk) * sqrt (Di))]

Cosine (D1) = 0.031 / [0.249 * 0.249]

= 0.031 / 0.062

= 0.500

Cosine (D2) = 0.031 / [0.249 * 0.509]

=0.031 / 0.127

= 0.244

Cosine (D3) = 0.062 / [0.249 * 0.567]

= 0.062 / 0.141

= 0.440

	D1	D2	D3
Cosine	0.500	0.244	0.440

	Rank 1	Rank 3	Rank 2

Tabel 3.4 Hasil Akhir Cosine Contoh 1

Dari hasil akhir (Cosine) maka dapat diketahui bahwa dokumen 1 (D_1) memiliki tingkat similaritas tertinggi kemudian disusul dengan D_3 lalu D_2. Berarti dapat diketahui bahwa dokumen dengan *rank* tertinggi-lah yang cocok dengan kata kunci yang dimasukkan yaitu Dokumen 1 dengan *rank* (0.500).

Mengingat :

1. Dokumen 1 (D1) = kategori Politik Ekonomi dengan subkategori Manajemen
2. Dokumen 2 (D2) = kategori Filsafat dan Pemikiran dengan subkategori Filsafat
3. Dokumen 3 (D3) = kategori Filsafat dan Pemikiran dengan subkategori Filsafat dan Perkembangan Islam

Maka kategori yang cocok untuk kata kunci Struktur Filsafat adalah kategori Filsafat dan Pemikiran dengan subkategori Filsafat dan Perkembangan Islam.

Contoh 2:

Pada metode Text Mining ada 5 tahap yang harus dilalui, antara lain *Tokenizing*, *Filtering*, *Stemming*, *Tagging*, dan *Analyzing*.

Berikut diagram alir dari tahap *text mining* pada aplikasi:



Gambar 3.15 Flowchart algoritma *text mining* 2

Berikut algoritma text mining yang digunakan:

Get sinopsis

Token (sinopsis) // memisahkan sesuatu kumpulan kata menjadi array berupa kata-kata

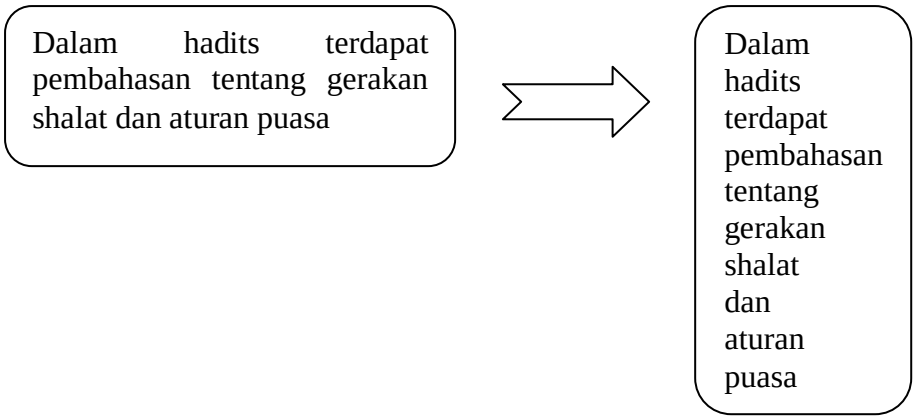
if (kata(n) = penting // proses filtering yaitu pengambilan kata penting dengan cara membandingkannya dengan tabel aturan produksi.

then

return kata (n)

```
else
        kata (n) = null; // kata tidak penting dibuang
end if
if (kata(n)! = kata dasar) // jika kata tersebut bukan kata dasar
then
        stemming (kata(n)) // membuang imbuhan yang terdapat pada kata
        if (stemming(kata(n))! = kata dasar) // proses tagging
        then
                tag (kata(n))
                return (kata(n))
        else
                return(kata(n))
        endif
analyzing (kata(n)) //penentua n nilai kata penting berdasarkan banyak kata yang
keluar dari sinopsis kemudian dikategorikan.
```

Contoh dari tahap tokenizing adalah sebagai berikut:



Token (sinopsis) // memisahkan sesuatu kumpulan kata menjadi array berupa kata-kata

if (kata(n) = penting // proses filtering yaitu pengambilan kata penting dengan cara membandingkannya dengan tabel aturan produksi.

then

return kata (n)

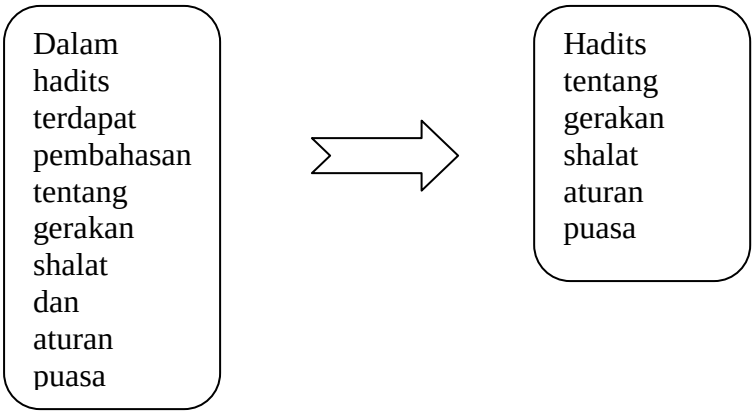
else

kata (n) = null; // kata tidak penting dibuang

end if

Untuk kata penting yang digunakan adalah acuan yang digunakan untuk mengklasifikasi buku baru seperti untuk subkategori Filsafat dan perkembangan maka acuan kata pentingnya adalah perkembangan filsafat, pemikiran Islam secara umum. Selain kata penting tersebut maka kata tidak penting seperti ‘adalah’, ‘sebuah’, dan ‘yang’ tidak akan digunakan.

Contoh dari tahap filtering adalah sebagai berikut:



if (kata(n) = penting // proses filtering yaitu pengambilan kata penting dengan cara membandingkannya dengan tabel aturan produksi.

then

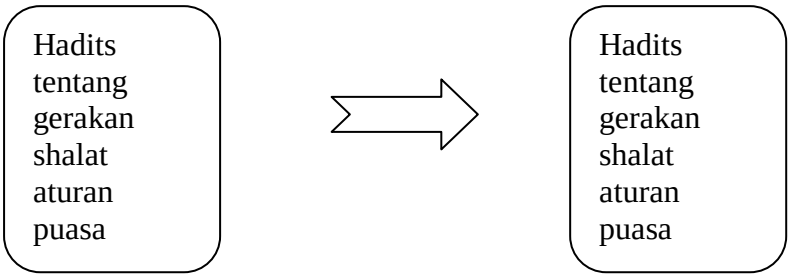
return kata (n)

else

kata (n) = null; // kata tidak penting dibuang

end if

Contoh dari tahap stemming adalah sebagai berikut:



if (kata(n)! = kata dasar) // jika kata tersebut bukan kata dasar

then

stemming (kata(n)) // membuang imbuhan yang terdapat pada kata

if (stemming(kata(n))! = kata dasar) // proses tagging

then

tag (kata(n))

return (kata(n))

else

return(kata(n))

endif

Untuk tahapan selanjutnya adalah tahap *Tagging* tetapi tidak digunakan dalam kasus sistem Automasi Perpustakaan ini. Hal ini dikarenakan bahasa Indonesia tidak memiliki bentuk lampau.

Tahap terakhir adalah tahap *analyzing*, yaitu tahap penentuan seberapa jauh keterhubungan antar kata-kata pada dokumen yang ada.

analyzing (kata(n)) //penentua n nilai kata penting berdasarkan banyak kata yang keluar dari sinopsis kemudian dikategorikan.

Pada tahap *analyzing* ini digunakan sebuah rumus. Formula yang digunakan untuk menghitung bobot (w) masing-masing dokumen terhadap kata kunci adalah:

$$W_{d,t} = tf_{d,t} * IDF_t$$

Dengan :

d = dokumen ke-d

t = kata ke-t dari kata kunci

$W_{d,t}$ = bobot dokumen ke-d terhadap kata ke-t

Setelah bobot (w) masing-masing dokumen diketahui maka dilakukan proses *sorting*/pengurutan dimana semakin besar nilai w, semakin besar tingkat similaritas dokumen tersebut terhadap kata yang dicari, demikian sebaliknya. Contoh implementasi sederhana dari *TF-IDF* adalah sebagai berikut:

Diketahui:

1. Dokumen 1 (D1) = kategori Agama dengan subkategori Lain-lain
2. Dokumen 2 (D2) = kategori Agama dengan subkategori Fiqih
3. Dokumen 3 (D3) = kategori Agama dengan subkategori Ilmu Al-Quran dan Al-Hadits
4. Kata kunci (kk) = kata kunci yang dimasukkan ke dalam sistem

Kata Kunci (kk) = Aturan Puasa

Dokumen 1 (D1) = Hadits tentang puasa

Dokumen 2 (D2) = Aturan gerakan shalat

Dokumen 3 (D3) = Pada hadits terdapat aturan puasa

Jadi jumlah dokumen (D) = 3

Pada perhitungan *TF/IDF* ini menggunakan metode *word list*, sehingga proses atau langkah yang dilakukan hanya sampai tahap *Tokenizing*, karena itu hasil Token akan dipakai pada tabel perhitungan.

token	tf				df	D/f	IDF log(D/f)	W			
	kk	D1	D2	D3				kk	D1	D2	D3
Hadits	0	1	0	1	2	1.5	0.176	0	0.176	0	0.176
tentang	0	1	0	0	1	3	0.477	0	0.477	0	0
puasa	1	1	0	1	2	1.5	0.176	0.176	0.176	0	0.176
aturan	1	0	1	1	2	1.5	0.176	0.176	0	0.176	0.176
gerakan	0	0	1	0	1	3	0.477	0	0	0.477	0
shalat	0	0	1	0	1	3	0.477	0	0	0.477	0
total								0.352	0.829	1.13	0.528

Tabel 3.5 Perhitungan *TF/IDF* Contoh 2

tf = banyak kata yang dicari pada sebuah dokumen

D = Dokumen

df = banyak dokumen yang mengandung kata yang dicari

w = bobot dokumen ke-d terhadap kata ke-t

Bobot (w) untuk D1 = 0.176 + 0 = 0.176

Bobot (w) untuk D2 = 0 + 0.176 = 0.176

Bobot (w) untuk D3 = 0.176 + 0.176 = 0.352

Apabila diurutkan maka proses *sorting* juga tidak akan dapat mengurutkan secara tepat, karena nilai w keduanya sama. Untuk mengatasi hal ini, algoritma dari *vector-space model* adalah jawabannya

token	Kk^2	$D1^2$	$D2^2$	$D3^2$	Kk*D1	Kk*D2	Kk*D3
Hadits	0	0.031	0	0.031	0	0	0
tentang	0	0.228	0	0	0	0	0

puasa	0.031	0.031	0	0.031	0.031	0	0.031
aturan	0.031	0	0.031	0.031	0	0.031	0.031
gerakan	0	0	0.228	0	0	0	0
shalat	0	0	0.228	0	0	0	0
	Sqrt(kk)	Sqrt(Di)			Sum(kk dot Di)		
	0.249	0.290	0.487	0.305	0.031	0.031	0.062

Tabel 3.6 Perhitungan *Vektor-Space Model* Contoh 2

Kk^2 = kata kunci pada tabel *TF/IDF* dikuadratkan

D = dokumen

Sqrt = *Squareroot* atau akar dari hail penjumlahan

Selanjutnya menghitung nilai Cosinus sudut antara vektor kata kunci dengan tiap dokumen dengan rumus:

$$\text{Cosine } (D_i) = \text{sum } (kk \text{ dot } D_i) / [\text{sqrt}(kk) * \text{sqrt}(D_i))]$$

Cosine (Di) = sum (kk dot Di) / [sqrt (kk) * sqrt (Di))]

Cosine (D1) = 0.031 / [0.249 * 0.290]

= 0.031 / 0.072

= 0.431

Cosine (D2) = 0.031 / [0.249 * 0.487]

=0.031 / 0.121

= 0.256

Cosine (D3) = 0.062 / [0.249 * 0.305]

= 0.062 / 0.076

= 0.816

	D1	D2	D3
Cosine	0.431	0.256	0.816
	Rank 2	Rank 3	Rank 1

Tabel 3.6 Hasil Akhir Cosine Contoh 2

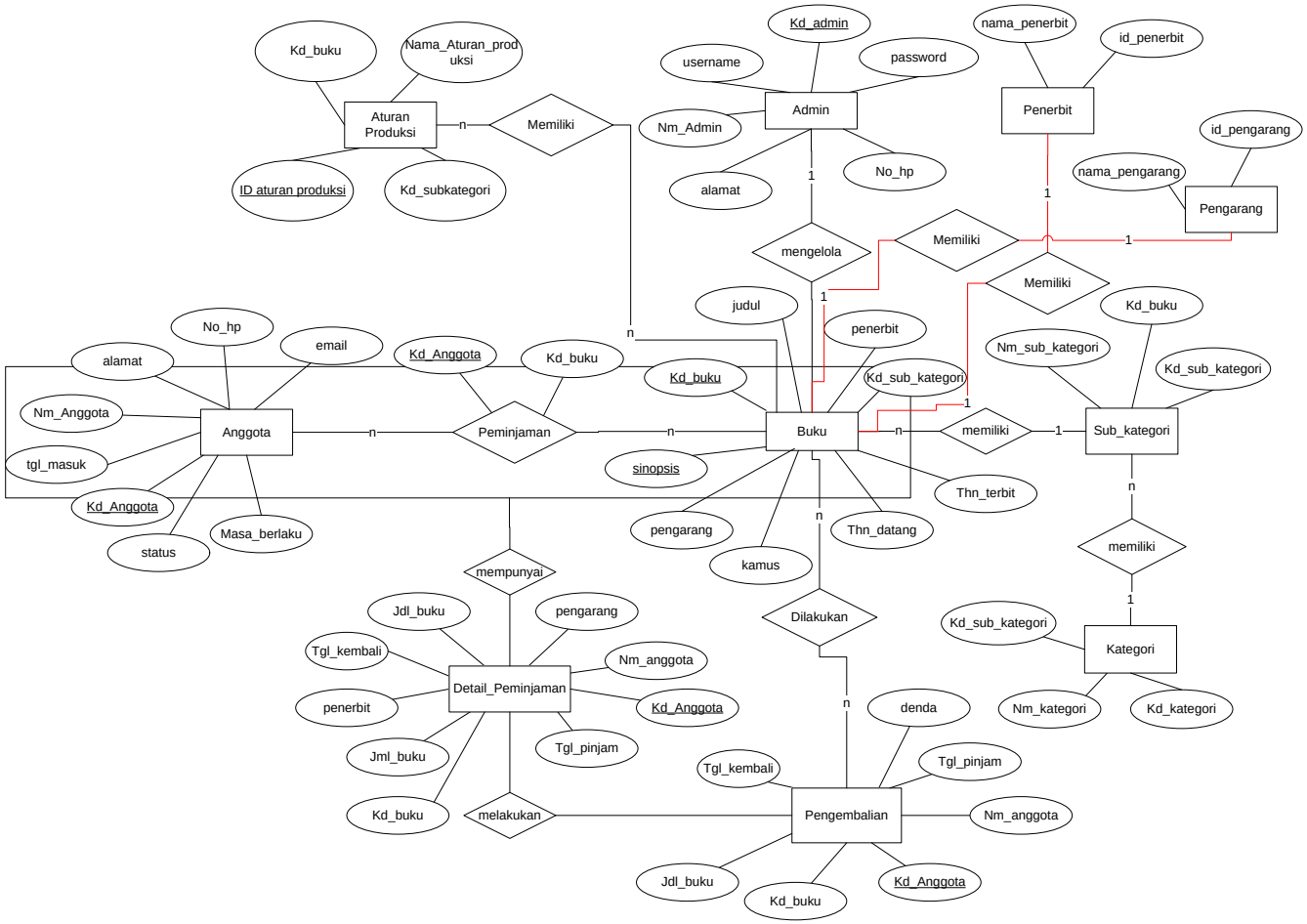
Dari hasil akhir (Cosine) maka dapat diketahui bahwa dokumen 1 (D_1) memiliki tingkat similaritas tertinggi kemudian disusul dengan D_3 lalu D_2. Berarti dapat diketahui bahwa dokumen dengan *rank* tertinggi-lah yang cocok dengan kata kunci yang dimasukkan yaitu Dokumen 1 dengan *rank* (0.500).

Mengingat :

1. Dokumen 1 (D1) = kategori Agama dengan subkategori Lain-lain
2. Dokumen 2 (D2) = kategori Agama dengan subkategori Fiqih
3. Dokumen 3 (D3) = kategori Agama dengan subkategori Ilmu Al-Quran dan Al-Hadits
4. Kata kunci (kk) = kata kunci yang dimasukkan ke dalam sistem

Maka kategori yang cocok untuk kata kunci Aturan Puasa adalah kategori Agama dengan subkategori Ilmu Al-Quran dan Al-Hadits.

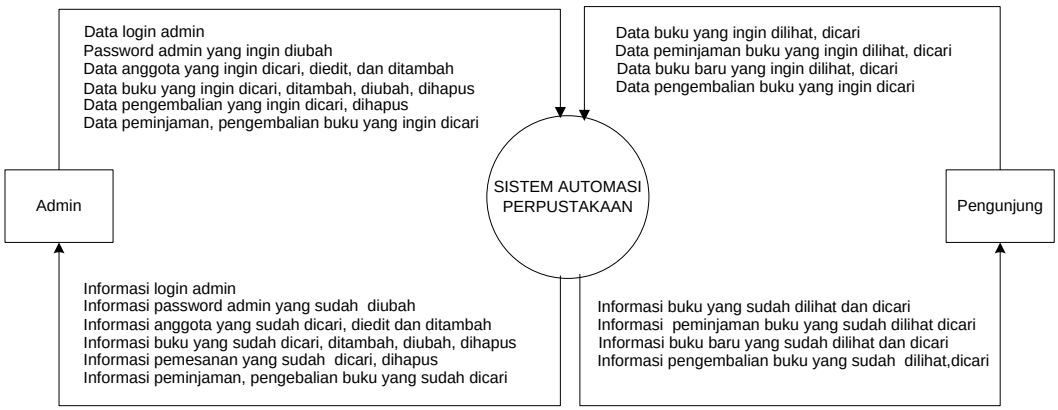
3.3 Entity Relationship Diagram



Gambar 3.20 ERD (Entity Relationship Diagram)

3.4 Analisis Kebutuhan Sistem Fungsional

Pada kebutuhan sistem fungsional ini terdapat Diagram Konteks juga *Data Flow Diagram* atau biasa disebut dengan DFD. Dalam kasus ini *Data Flow Diagram* yang dibuat terdapat 6 proses yang mana dari aliran data inilah akan terlihat gambaran umum sistem yang akan dibangun.



Gambar 3.21 Diagram Konteks

Terdapat 6 proses utama, yaitu Login Admin, Pengolahan Data Anggota, Pengolahan Data Kategori dan Subkategori, Pengadaan Koleksi buku/Buku baru, Pengolahan Data Peminjaman dan Pengolahan Data Pengembalian.

1. Login Admin

   Proses ini dilakukan oleh admin untuk memulai semua proses yang ada dalam sistem. Proses yang bisa dilakukan untuk admin yaitu verifikasi id_admin dan password.

2. Pengolahan Data Anggota

   Dalam proses ini, mencari data anggota, mengubah dan menambahnya, hanya bisa dilakukan oleh admin. Sedangkan pengunjung perpustakaan atau anggota hanya dapat melihat dan mencari data anggota saja.

3. Pengolahan Data Kategori dan Subkategori

   Sama halnya dengan proses ini, admin bisa melakukan kegiatan pencarian, penambahan, dan perubahan kategori dan subkategori. Untuk pengunjung tidak bisa melihat, mencari data, juga mencari data kategori dan subkategori.

4. Pengadaan Koleksi/Buku baru

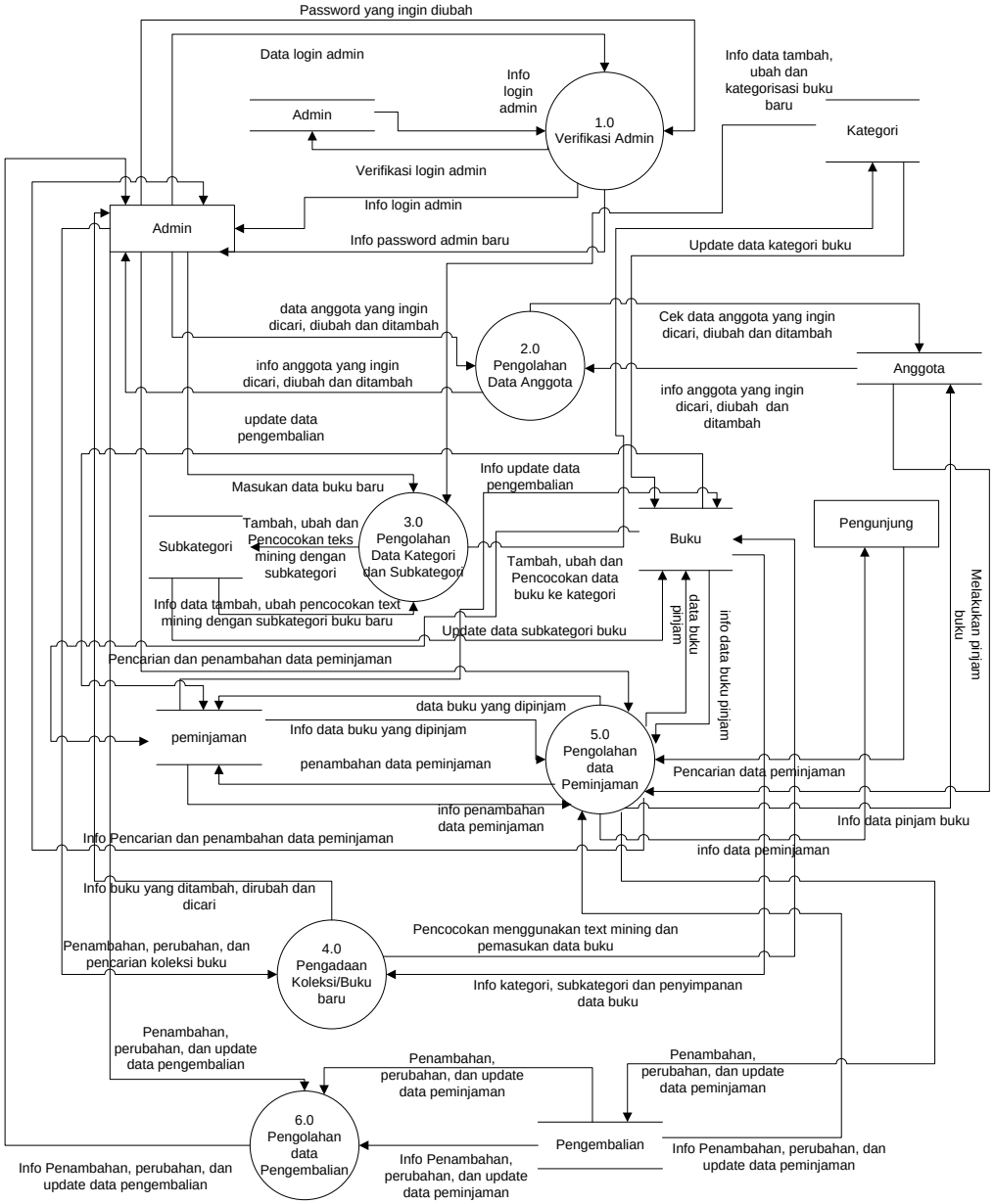
   Pengunjung perpustakaan atau anggota, dapat melihat dan mencari data peminjaman, tetapi hanya admin yang dapat melakukan penambahan atau perubahan data peminjaman.

5. Pengolahan Data Peminjaman

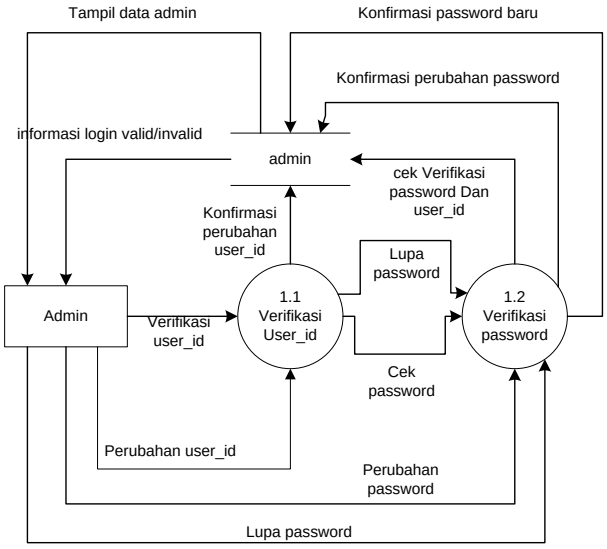
   Dalam proses ini, admin dapat menambah, merubah, dan meng-*update* informasi data pengembalian dari data peminjaman

6. Pengolahan Data Pengembalian

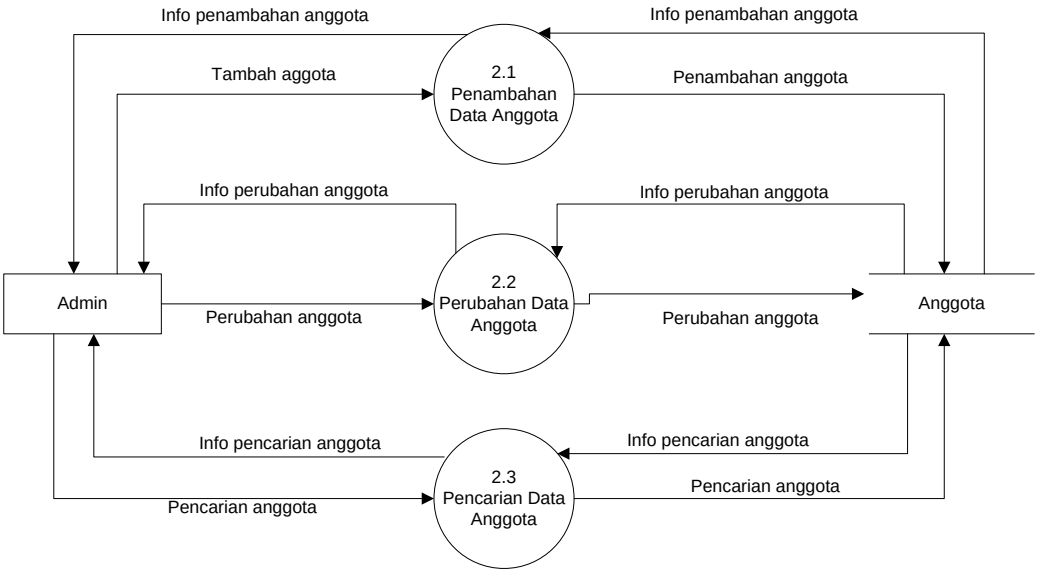
   Proses ini dilakukan oleh admin. Proses ini akan didapat melalui jenis buku-buku yang banyak diminati pengunjung.



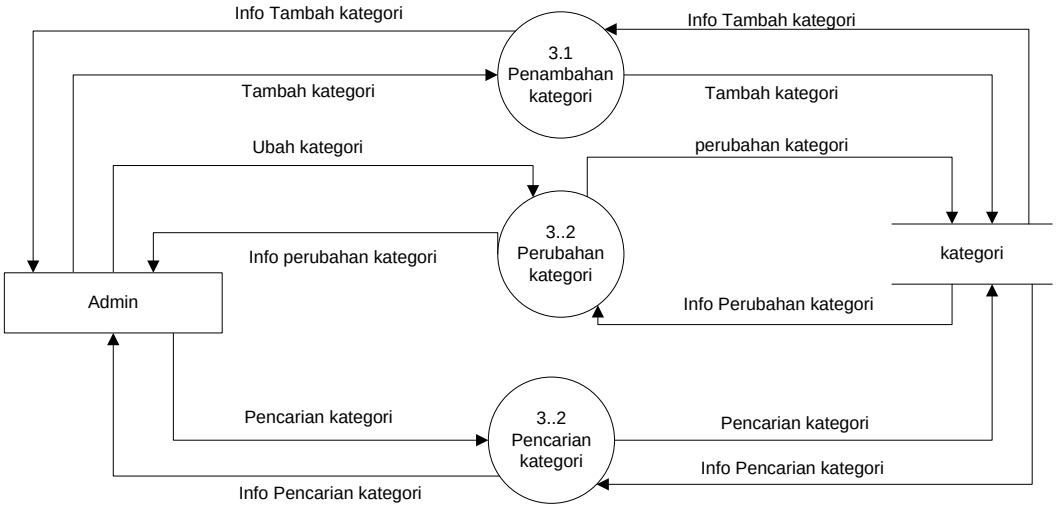
Gambar 3.22 DFD Level 0



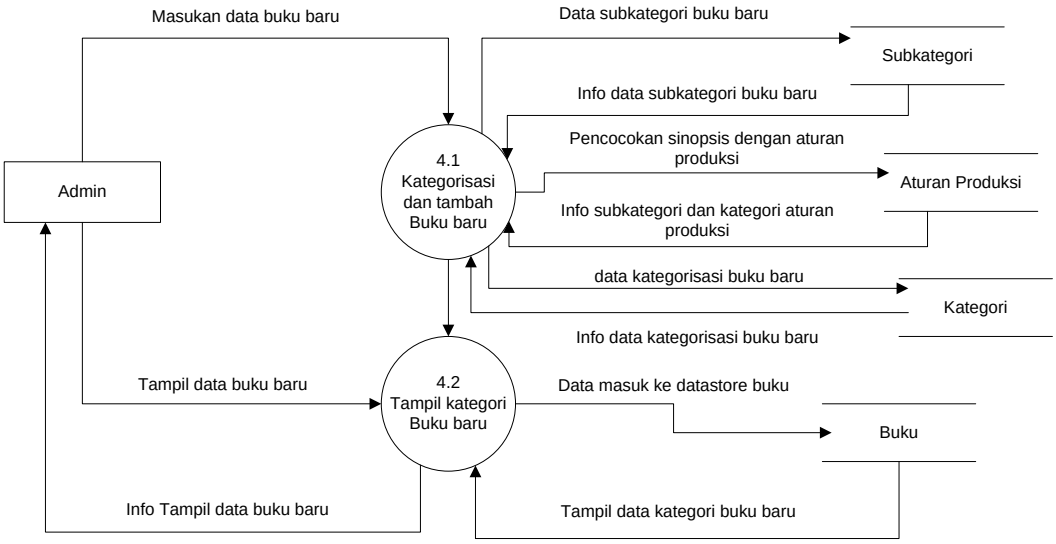
Gambar 3.23 DFD Level 1 Proses 1



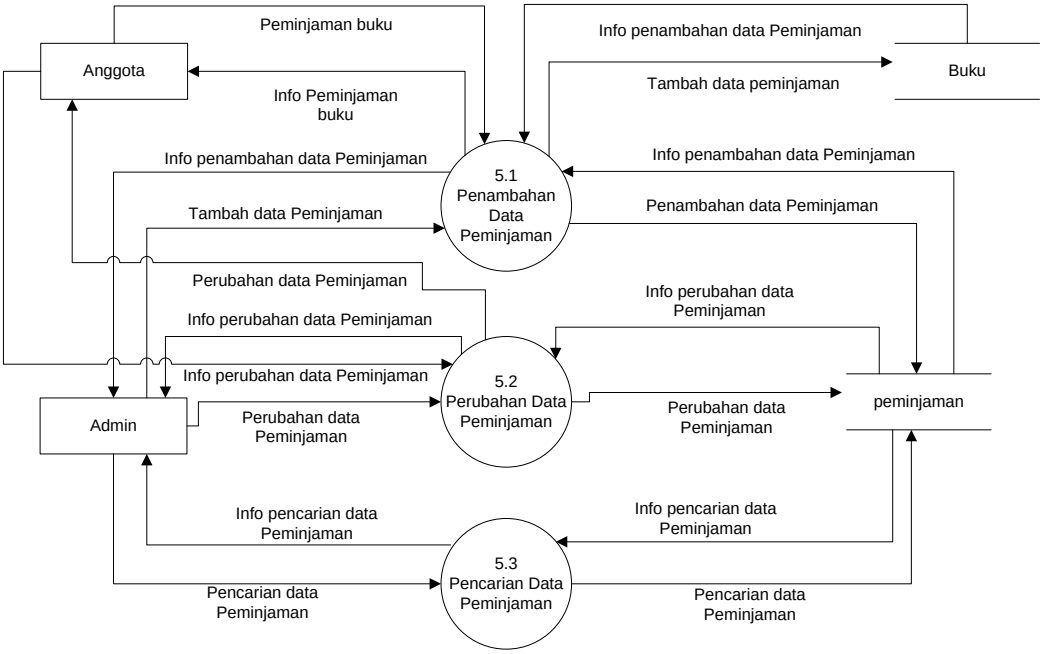
Gambar 3.24 DFD Level 1 Proses 2



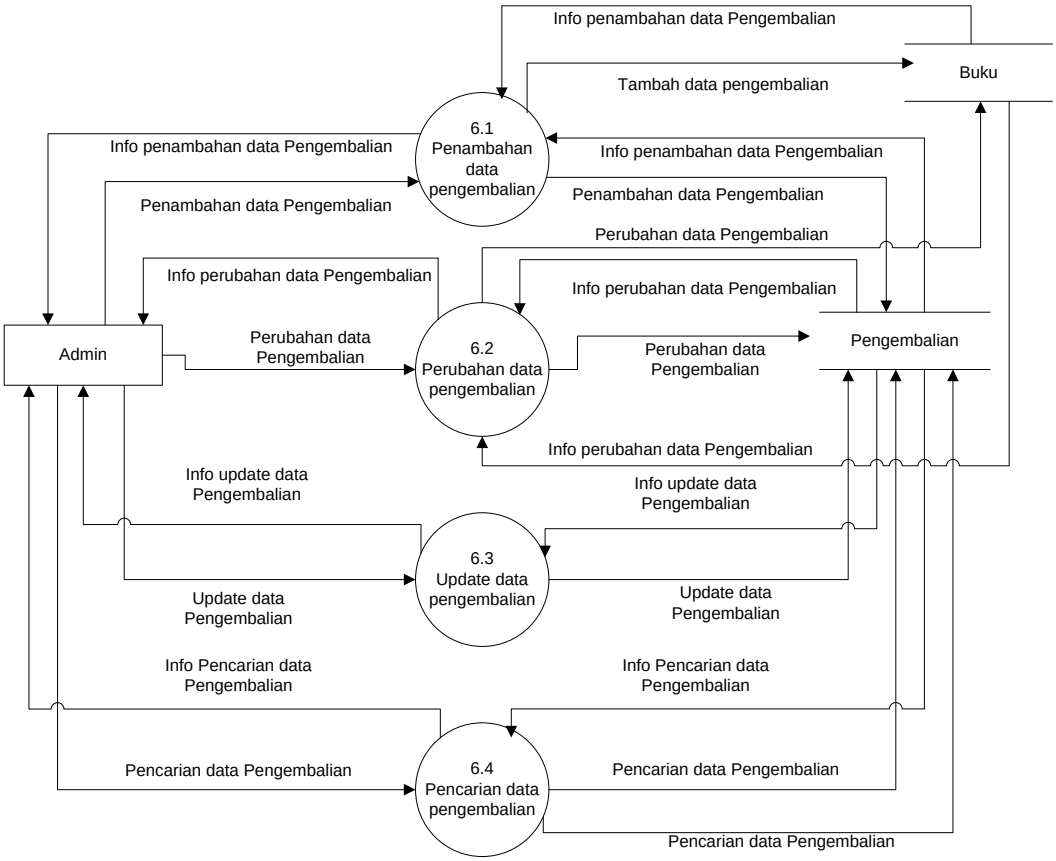
Gambar 3.25 DFD Level 1 Proses 3



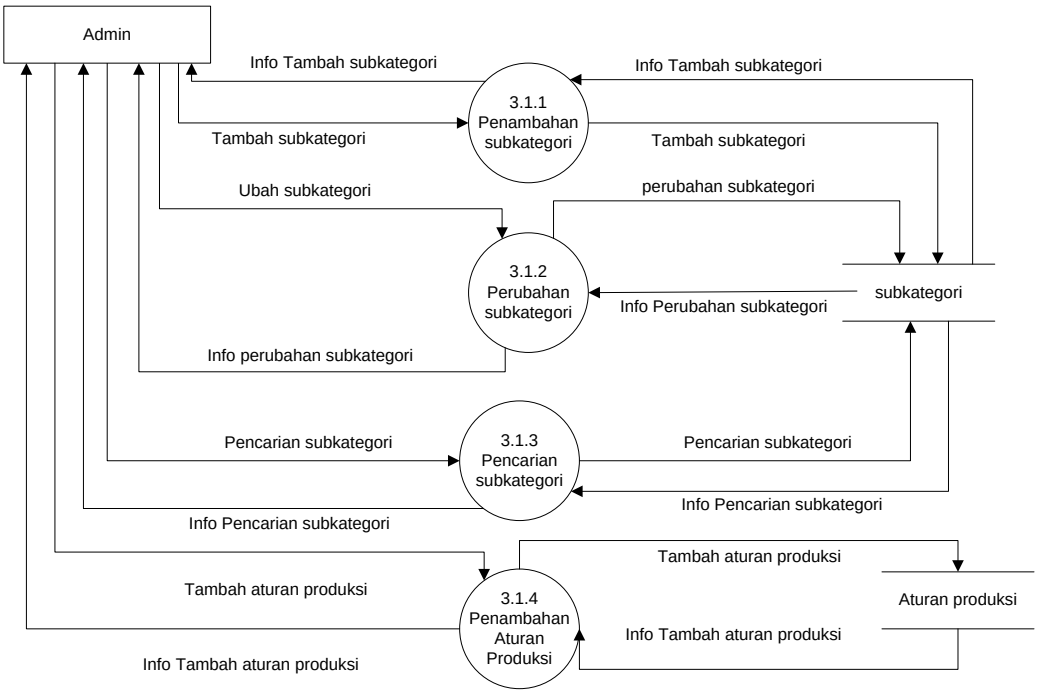
Gambar 3.26 DFD Level 1 Proses 4



Gambar 3.27 DFD Level 1 Proses 5



Gambar 3.28 DFD Level 1 Proses 6



Gambar 3.29 DFD Level 2 Proses 3

3.5 Spesifikasi Proses

Spesifikasi proses digunakan untuk menggambarkan proses model aliran yang terdapat pada Diagram alir data. Spesifikasi proses dari gambaran Diagram alir data diatas akan dijelaskan pada tabel dibawah ini:

Tabel 3.8 Spesifikasi Proses

No.	Proses	Keterangan
1	No. Proses	1.
	Nama Proses	Login
	Source (Sumber)	Admin
	Input	-data Login invalid
	Output	-info Login invalid
	Destination (Tujuan)	Admin
	Logika Proses	Begin {Admin memasukan *data login* ke database} if nama pengguna and sandi ada then login valid else tampil login invalid End
2	No. Proses	2.1
	Nama Proses	Tambah Anggota Baru
	Source (Sumber)	Admin
	Input	-data anggota baru
	Output	-info data anggota baru
	Destination (Tujuan)	Admin
	Logika Proses	Begin {Admin menambah anggota baru ke *database*} if data anggota baru lengkap then data anggota baru disimpan di *database* else data anggota baru gagal disimpan End
3	No. Proses	2.2
	Nama Proses	Ubah Data Anggota Baru
	Source (Sumber)	Admin
	Input	-ubah data anggota baru
	Output	-info data anggota baru
	Destination (Tujuan)	Admin
	Logika Proses	Begin {Admin mengubah data anggota baru yang ada dalam tabel anggota} if data anggota baru lengkap then anggota baru disimpan di *database* else data anggota baru gagal disimpan End

Tabel 3.8 Spesifikasi Proses (Lanjutan)

No.	Proses	Keterangan
4	No. Proses	2.3
	Nama Proses	Cari Data Anggota Baru
	Source (Sumber)	Admin
	Input	-data anggota baru
	Output	-info cari data anggota baru
	Destination (Tujuan)	Admin
	Destination (Tujuan)	Admin
	Logika Proses	Begin {Admin mencari data anggota baru ke d*atabase*} if data anggota baru ketemu then data anggota baru ditampilkan else data anggota baru gagal ditampilkan End
5	No. Proses	3.1
	Nama Proses	Tambah Kategorisasi buku baru
	Source (Sumber)	Admin
	Input	- Tambah kategorisasi buku baru
	Output	-info Tambah kategorisasi buku baru
	Destination (Tujuan)	Admin
	Logika Proses	Begin {Admin menambah kategorisasi buku baru yang ada dalam tabel buku} if data Tambah kategorisasi lengkap then data Tambah kategorisasi disimpan di *database* else data Tambah kategorisasi gagal disimpan End
6	No. Proses	3.2
	Nama Proses	Ubah Data kategorisasi buku
	Source (Sumber)	Admin
	Input	-ubah data kategorisasi buku
	Output	-info data kategorisasi buku
	Destination (Tujuan)	Admin
	Logika Proses	Begin {Admin mengubah kategorisasi buku ke d*atabase*} if data kategorisasi buku lengkap then data kategorisasi buku disimpan di *database* else data kategorisasi buku gagal disimpan End
7	No. Proses	3.3
	Nama Proses	Cari Data kategorisasi buku
	Source (Sumber)	Admin
	Input	-cari data kategorisasi buku
	Output	-info cari data kategorisasi buku
	Destination (Tujuan)	

		Admin

Tabel 3.8 Spesifikasi Proses (Lanjutan)

	Proses	Keterangan
	Logika Proses	Begin {Admin mencari data kategorisasi buku yang ada dalam tabel buku} if data kategorisasi buku ketemu then data kategorisasi buku disimpan ditampilkan else data kategorisasi buku gagal ditampilkan End
8	No. Proses	3.1.1
	Nama Proses	Tambah Subkategori buku baru
	Source (Sumber)	Admin
	Input	- Tambah Subkategori buku baru
	Output	-info Tambah Subkategori buku baru
	Destination (Tujuan)	Admin
	Logika Proses	Begin {Admin menambah Tambah Subkategori buku baru yang ada dalam tabel buku} if data Tambah Subkategori lengkap then data Tambah Subkategori disimpan di *database* else data Tambah Subkategori gagal disimpan End
9	No. Proses	3.1.2
	Nama Proses	Ubah Data Subkategori buku
	Source (Sumber)	Admin
	Input	-ubah data Subkategori buku
	Output	-info data Subkategori buku
	Destination (Tujuan)	Admin
	Logika Proses	Begin {Admin mengubah Subkategori buku ke d*atabase*} if data Subkategori buku lengkap then data Subkategori buku disimpan di *database* else data Subkategori buku gagal disimpan End
10	No. Proses	3.1.3
	Nama Proses	Cari Data Subkategori buku
	Source (Sumber)	Admin
	Input	-cari data Subkategori buku
	Output	-info cari data Subkategori buku
	Destination (Tujuan)	Admin
	Logika Proses	Begin {Admin mengubah data cari Subkategori buku ke d*atabase*} if data cari Subkategori buku lengkap then data cari Subkategori buku disimpan di *database* else data cari Subkategori buku gagal disimpan End
11	No. Proses	4.1
	Nama Proses	Kategorisasi buku baru

	Source (Sumber)	Admin
	Input	- kategorisasi buku baru
	Output	-info kategorisasi buku baru

Tabel 3.8 Spesifikasi Proses (Lanjutan)

	Proses	**Keterangan**
	Destination (Tujuan)	Admin
	Logika Proses	Begin {Admin menambah kategorisasi buku baru yang ada dalam tabel buku} if data kategorisasi buku lengkap then data kategorisasi disimpan di *database* else data kategorisasi gagal disimpan End
12	No. Proses	4.2
	Nama Proses	Tampil kategori buku baru
	Source (Sumber)	Admin
	Input	-tampil kategorisasi buku baru
	Output	-info kategorisasi buku baru
	Destination (Tujuan)	Admin
	Logika Proses	Begin { kategori buku baru yang ada dalam tabel buku ditampilkan kepada admin} End
13	No. Proses	5.1
	Nama Proses	tambah data peminjaman
	Source (Sumber)	Admin
	Input	-data peminjaman
	Output	-info data peminjaman
	Destination (Tujuan)	Admin
	Logika Proses	Begin {Admin menambah data peminjaman ke d*atabase*} if data peminjaman lengkap then data peminjaman ketemu disimpan di *database* else if data peminjaman tidak lengkap then data peminjaman gagal disimpan End
14	No. Proses	5.2
	Nama Proses	Ubah Data Peminjaman
	Source (Sumber)	Admin
	Input	-ubah data peminjaman
	Output	-info data peminjaman
	Destination (Tujuan)	Admin
	Logika Proses	Begin {Admin megubah data peminjaman ke d*atabase*} if data peminjaman lengkap then data peminjaman disimpan di *database* else data peminjaman gagal disimpan End
15	No. Proses	5.3
	Nama Proses	Cari Data Peminjaman

	Source (Sumber)	Admin
	Input	-cari data peminjaman
	Output	-info cari data peminjaman

Tabel 3.8 Spesifikasi Proses (Lanjutan)

	Proses	**Keterangan**
	Destination (Tujuan)	Admin
	Logika Proses	Begin {Admin mencari data peminjaman yang ada dalam tabel peminjaman} if data peminjaman ketemu then data peminjaman ditampilkan else data peminjaman gagal ditammpilkan End
16	No. Proses	6.1
	Nama Proses	Tambah Data Pengembalian
	Source (Sumber)	Admin
	Input	-tambah data pengembalian
	Output	-info tambah data pengembalian
	Destination (Tujuan)	Admin
	Logika Proses	Begin {Admin menambah data pengembalian ke d*atabase*} if data pengembalian lengkap then data pengembalian disimpan di *database* else data pengembalian gagal disimpan End
17	No. Proses	6.2
	Nama Proses	Ubah Data Pengembalian
	Source (Sumber)	Admin
	Input	-data pengembalian
	Output	-info ubah data pengembalian
	Destination (Tujuan)	Admin
	Logika Proses	Begin {Admin mengubah data pengembalian ke d*atabase*} if data pengembalian lengkap then data pengembalian disimpan di *database* else data pengembalian gagal disimpan End
18	No. Proses	6.3
	Nama Proses	Update data pengembalian
	Source (Sumber)	Admin
	Input	-data pengembalian
	Output	-info update data pengembalian
	Destination (Tujuan)	Admin
	Logika Proses	Begin {Admin meng-update data pengembalian yang ada dalam tabel pengembalian} if data pengembalian selesai di-update then data pengembalian disimpan di *database* else data pengembalian gagal disimpan End
19	No. Proses	6.4

	Nama Proses	Cari data pengembalian
	Source (Sumber)	Admin
	Input	-cari data pengembalian

Tabel 3.8 Spesifikasi Proses (Lanjutan)

Proses	Keterangan
Output	-info cari data pengembalian
Destination (Tujuan)	Admin
Logika Proses	Begin {Admin mencari data pengembalian ke d*atabase*} if data pengembalian yang dicari ketemu then data pengembalian ditampilkan else data pengembalian gagal ditampilkan End

3.6 Perancangan Sistem

Perancangan merupakan bagian dari metodologi pembangunan suatu perangkat lunak yang harus dilakukan setelah melalui tahapan analisis. Pada bagian ini akan dijelaskan perancangan sistem yang dimaksudkan untuk menggambarkan perbedaan antara sistem yang sedang berjalan dengan sistem yang diusulkan. Perancangan sistem ini menggunakan pendekatan sistem terkomputerisasi.

Langkah-langkah yang dilakukan dalam tahapan perancangan sistem ini adalah sebagai berikut:

1. Perancangan Data
    a. Struktur *File*
2. Arsitektur
3. Perancangan antar muka
    a. Perancangan Struktur Menu Admin
    b. Perancangan Struktur Menu Pengunjung
    c. Perancangan Form

d. Perancangan Pesan

e. Perancangan Prosedural

3.7 Perancangan Data

Perancangan data yang dimaksud akan dijabarkan dalam subbab berikut ini

3.7.1 Struktur File

Pada tahap perancangan struktur tabel ini akan dijelaskan mengenai perancangan database yang akan digunakan, melakukan penyusunan file data yang sesuai dengan kelas datanya. Penyusunan struktur tabel ini akan mempermudah dalam pemasukan data sesuai dengan pengelompokan dari data atau informasi tersebut. Adapun struktur tabel masing-masing database dapat dilihat pada tabel dibawah ini:

1. Tabel Admin

Nama tabel : admin

Tempat Penyimpanan : harddisk

No	Nama *Field*	*Type* dan *size*	Keterangan
1	id_admin	varchar (2)	Primary key
2	Username	varchar (8)	
3	Userpass	varchar (41)	
4	nama	varchar(50)	
5	Alamat	varchar(50)	
6	No.hp	varchar(15)	

Tabel 3.9 Struktur tabel admin

2. Tabel Anggota

Nama Tabel : Anggota

Tempat Penyimpanan : harddisk

No	Nama *Field*	*Type* dan *size*	Keterangan
1	Tanggal_masuk	varchar (2)	

2	Kode_anggota	varchar (9)	Primary key
3	Nama	varchar (41)	
4	Alamat	varchar(50)	
5	Email	varchar(50)	
6	No.hp	varchar(14)	
7	Masa berlaku	date	
8	Status	Varchar (15)	

Tabel 3.10 Struktur tabel anggota

3. Tabel Peminjaman

Nama Tabel : Peminjaman

Tempat Penyimpanan : harddisk

No	**Nama *Field***	***Type* dan *size***	**Keterangan**
1	Kode_anggota	varchar (8)	
2	Tanggal_masuk	varchar (8)	Primary key
3	Nama	date	
4	Alamat	date	
5	Email	Year (4)	
6	No Hp	varchar(50)	
7	Status	varchar(50)	

Tabel 3.11 Struktur tabel peminjaman

4. Tabel Pengembalian

Nama Tabel : Pengembalian

Tempat Penyimpanan : harddisk

No	**Nama *Field***	***Type* dan *size***	**Keterangan**
1	Tanggal_pengembalian	date (2)	Primary key
2	Kode_anggota	varchar (8)	
3	Nama	varchar (41)	
4	Judul_buku	varchar(50)	

Tabel 3.12 Struktur tabel pengembalian

5. Tabel Buku

Nama Tabel : Buku

Tempat Penyimpanan : harddisk

No	Nama *Field*	*Type* dan *size*	Keterangan
1	Kode_buku	varchar (7)	Primary key
2	Judul_Buku	varchar (50)	
3	Pengarang	varchar (50)	
4	Penerbit	varchar(50)	
5	Thn_terbit	Year(4)	
6	Qty	Varchar(1)	

Tabel 3.13 Struktur tabel buku

6. Tabel Kamus

Nama Tabel : Kamus

Tempat Penyimpanan : harddisk

No	Nama *Field*	*Type* dan *size*	Keterangan
1	Id_subkategori	int (11)	Primary key
2	Nama_subkategori	varchar (70)	
3	Kamus_kata	text	
4	Nama_kategori	varchar(50)	

Tabel 3.14 Struktur tabel kamus

7. Tabel Kategori

Nama Tabel : Kategori

Tempat Penyimpanan : harddisk

No	Nama *Field*	*Type* dan *size*	Keterangan
1	Id_Kategori	int (11)	Primary key
2	Nama_kategori	varchar(50)	

Tabel 3.15 Struktur tabel kategori

8. Tabel Subkategori

Nama Tabel : Subkategori

Tempat Penyimpanan : harddisk

No	Nama *Field*	*Type* dan *size*	Keterangan
1	Id_subkategori	int (11)	Primary key
2	Nama_subkategori	varchar (70)	
3	Id_kategori	int (11)	
4	Nama_kategori	varchar(50)	

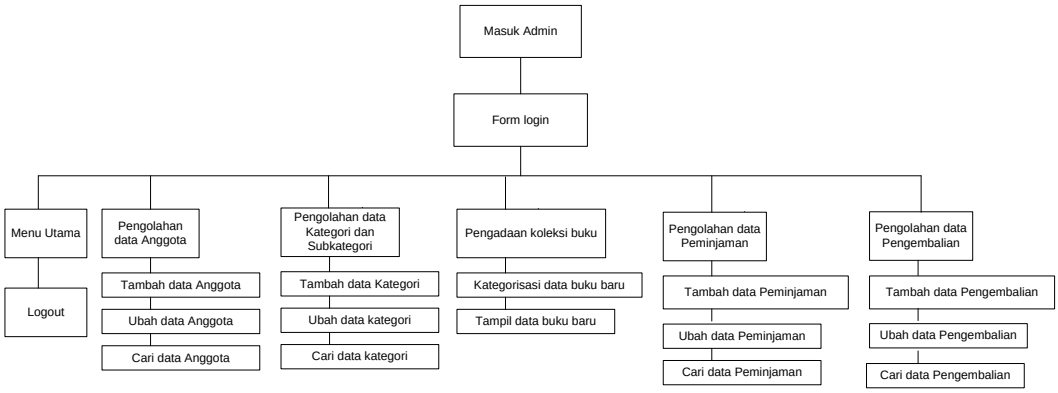
Tabel 3.16 Struktur tabel subkategori

3.8 Perancangan antar muka

Untuk perancangan antar muka akan terbagi menjadi beberapa bagian, antara lain:

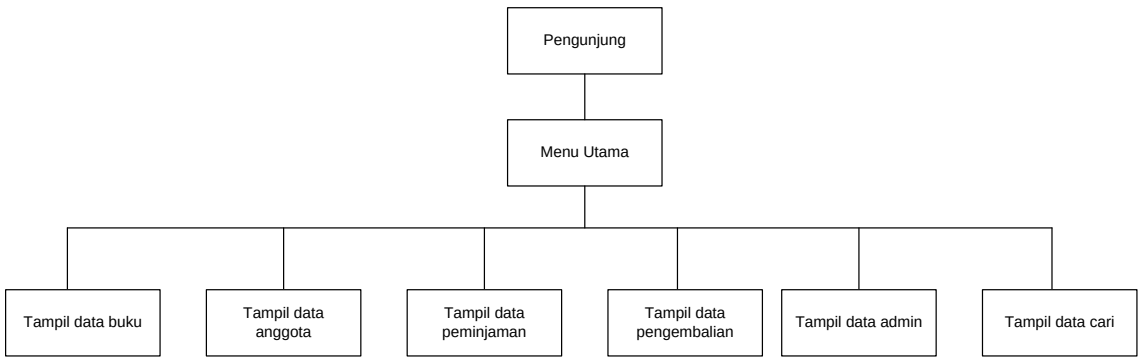
3.8.1 Perancangan Struktur Menu Admin

Struktur menu adalah bentuk umum dari suatu rancangan program untuk memudahkan pemakai dalam menjalankan program komputer. Sehingga saat menjalankan program, admin tidak mengalami kesulitan dalam memilih menu-menu yang diinginkan. Pada perancangan struktur menu admin ini dibuat menu seperti gambar dibawah ini:



Gambar 3.30 Perancangan struktur menu admin

3.8.2 Perancangan Struktur Menu Pengunjung



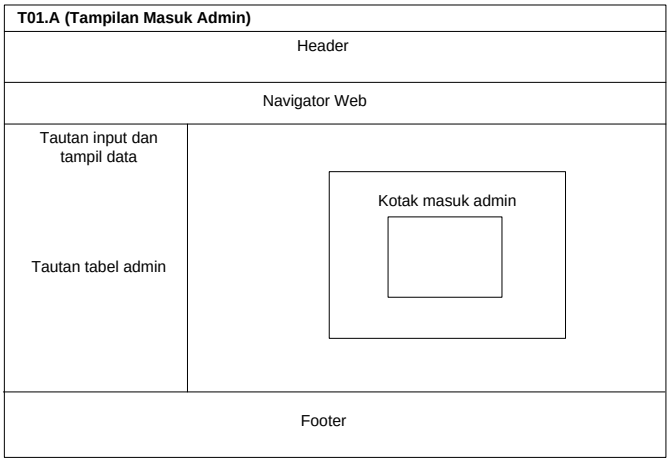
Gambar 3.31 Perancangan struktur menu pengunjung

Setelah melakukan perancangan data, maka dibuatlah perancangan antar muka sebagai tempat tampilan data. Perancangan antar muka dilakukan untuk mempermudah untuk mengimplementasikan sistem yang akan dibangun.

3.8.3 Perancangan Form

Dalam perancangan *form* disini dilakukan perancangan baik terhadap *form input* dan *form output*. Adapun *form-form* yang terdapat dalam sistem automasi perpustakaan di Perpustakaan Salman berbasis *Web* adalah sebagai berikut:

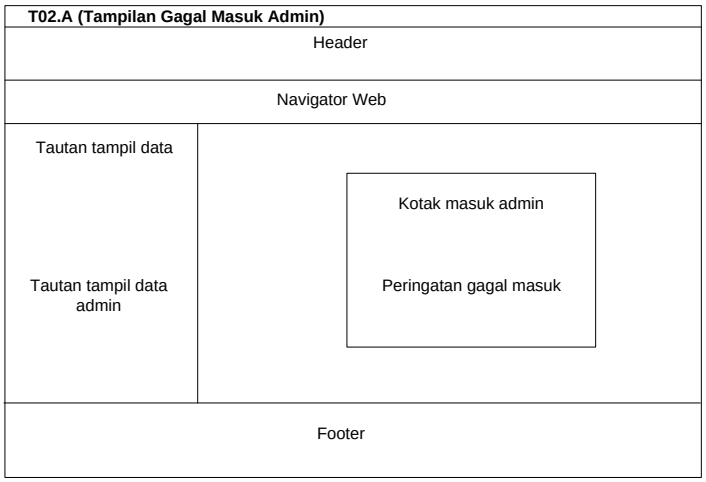
1. Perancangan tampilan masuk Admin



Gambar 3.32 Tampilan masuk admin

```
Input           : nama dan sandi
Proses          : jika nama atau sandi salah, maka akan menuju ke
                  halaman info nama atau sandi salah
Output          : tampilan utama admin
Keterangan      : T01.A (Tampilan masuk Admin)
```

2. Perancangan tampilan gagal masuk admin

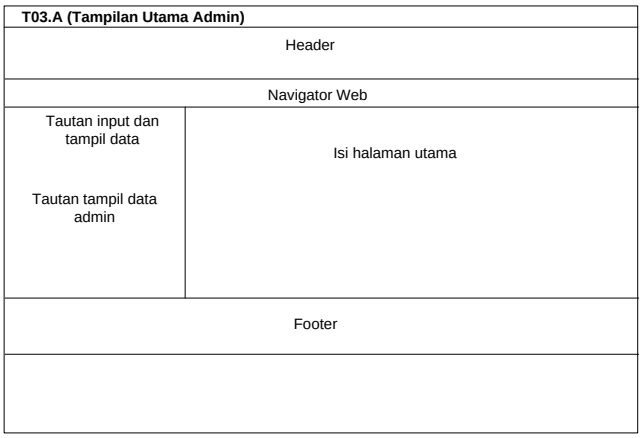


Gambar 3.33 Tampilan gagal masuk admin

```
Input           : -
Proses          : jika nama atau sandi benar, maka akan menuju ke
                  halaman utama admin
Output          : tampilan info gagal masuk admin
Keterangan      : T02.A (Tampilan gagal masuk admin)
```

3. Perancangan tampilan utama Admin

Tampilan utama ini adalah antarmuka pembuka aplikasi sekaligus menjadi antarmuka standar yang akan selalu ada pada antarmuka selanjutnya. Tampilan ini dihubungkan dari tampilan satu ke tampilan lain.

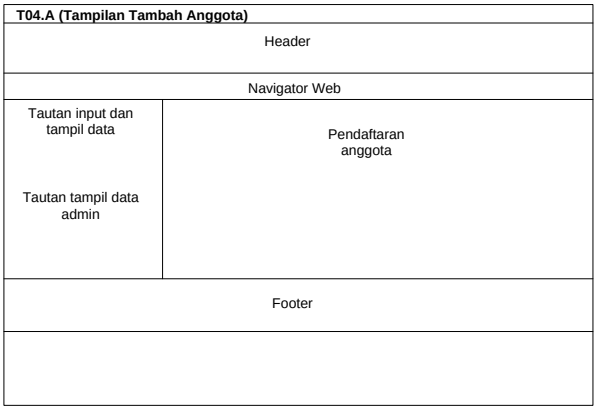


Gambar 3.34 Tampilan utama admin

```
Input           : -
```

Proses : jika masuk admin valid maka keabsahan nama dan sandi admin di cek, jika tidak maka akan menuju ke halaman info nama atau sandi salah
Output : pilihan admin
Keterangan : T03.A (Tampilan utama Admin)

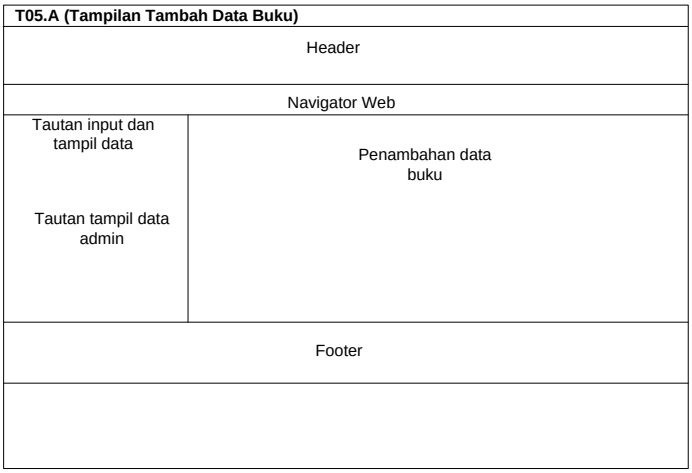
4. Perancangan tampilan pendaftaran anggota



Gambar 3.35 Tampilan pendaftaran anggota

Input : link data anggota di klik
Proses : jika data anggota berhasil disimpan maka akan menuju ke halaman info penambahan data anggota
Output : pilihan admin
Keterangan : T04.A (Tampilan tambah anggota)

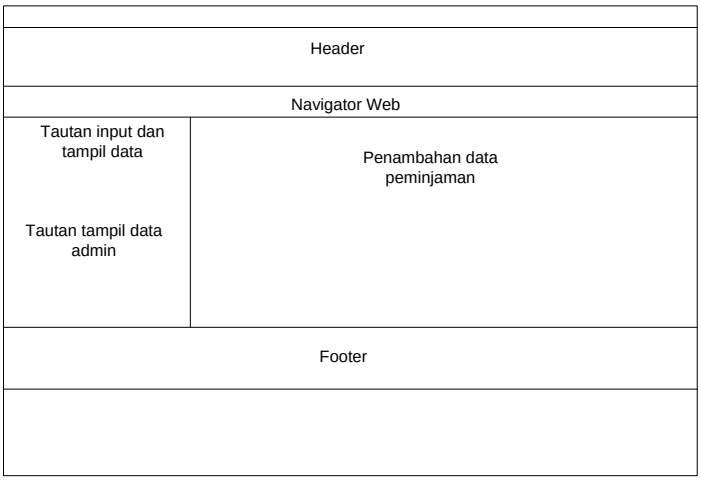
5. Perancangan tampilan penambahan data buku



Gambar 3.36 Tampilan penambahan data buku

Input : tambah data buku
Proses : jika data buku berhasil disimpan maka akan menuju ke halaman info penambahan data buku
Output : pilihan admin
KKeterangan : T05.A (Tampilan tambah data buku)

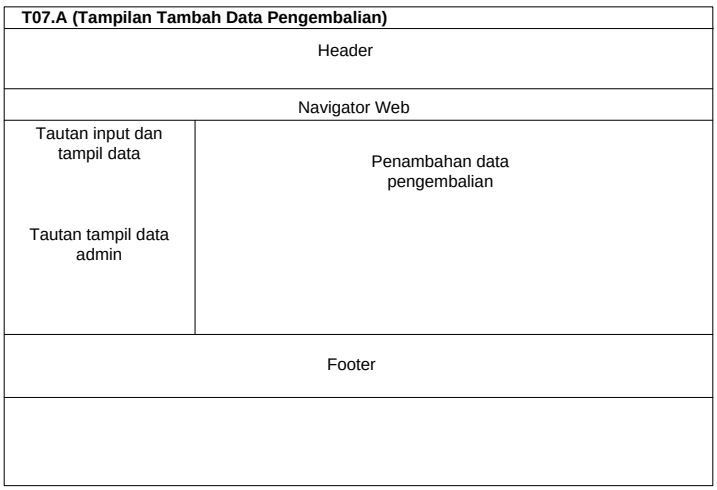
6. Perancangan tampilan penambahan data peminjaman



Gambar 3.37 Tampilan penambahan data peminjaman

```
Input           : tambah data peminjaman
Proses          : jika data peminjaman berhasil disimpan maka akan
                  menuju ke halaman info penambahan data peminjaman
Output          : pilihan admin
Keterangan      : T06.A (Tampilan tammbah data peminjaman)
```

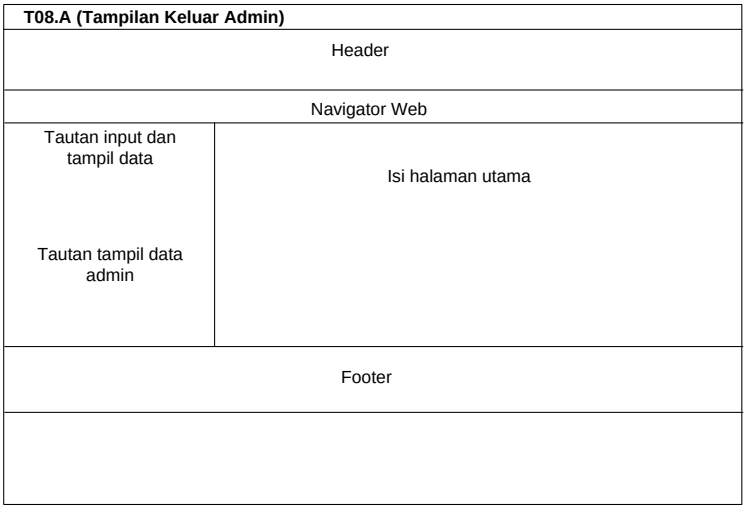
7. Perancangan tampilan penambahan data pengembalian



Gambar 3.38 Tampilan penambahan data pengembalian

```
Input           : tambah data pengembalian
Proses          : jika data pengembalian berhasil disimpan maka akan
                  menuju ke halaman info penambahan data pengembalian
Output          : pilihan admin
Keterangan      : T07.A (Tampilan tambah data pengembalian)
```

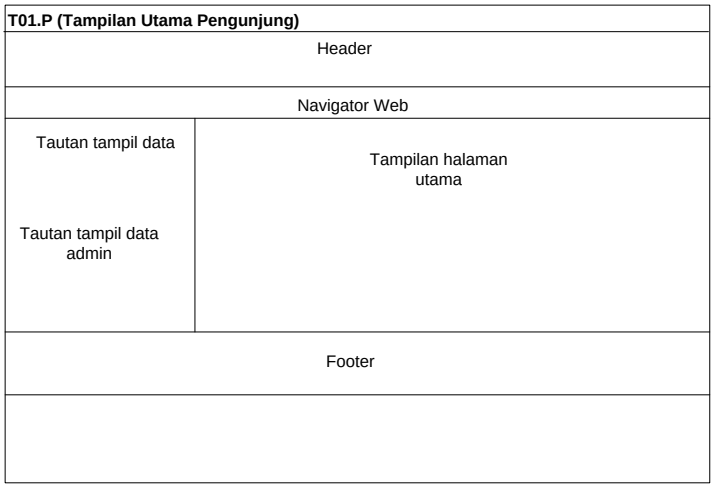
8. Perancangan tampilan keluar admin



Gambar 3.39 Tampilan keluar admin

```
Input           : jika tautan keluar admin di klik
Proses          : keluar
Output          : info keluar system
Keterangan      : T08.A (Tampilan keluar sistem)
```

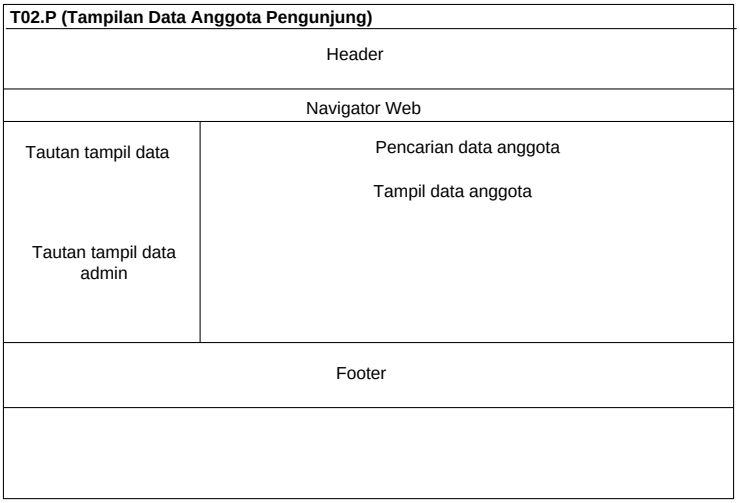
9. Perancangan tampilan utama pengunjung



Gambar 3.40 Tampilan utama pengunjung

```
Input           : -
Proses          : jika user adalah pengunjung maka akan langsung masuk
                  ke halaman utama pengunjung
Output          : pilihan pengunjung
Keterangan      : T01.P (Tampilan utama pengunjung)
```

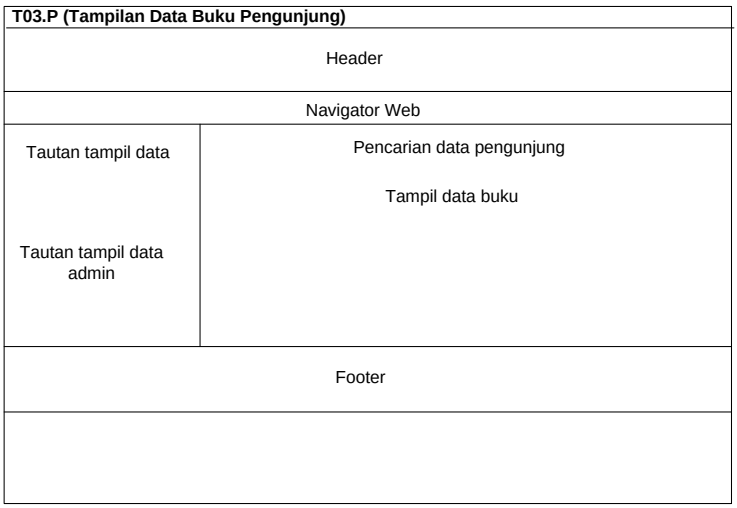
10. Perancangan tampilan data anggota



Gambar 3.41 Tampilan data anggota

```
Input           : -
Proses          : -
Output          : tampilan data anggota
Keterangan      : T02.P (Tampilan data anggota pengunjung)
```

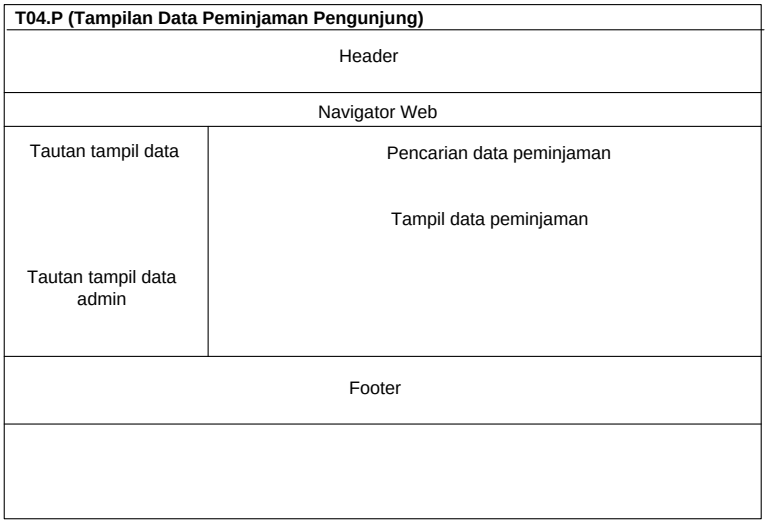
11. Perancangan tampilan data buku



Gambar 3.42 Tampilan data buku

```
Input           : -
Proses          : -
Output          : tampilan data buku
Keterangan      : T03.P (Tampilan data buku penngunjung)
```

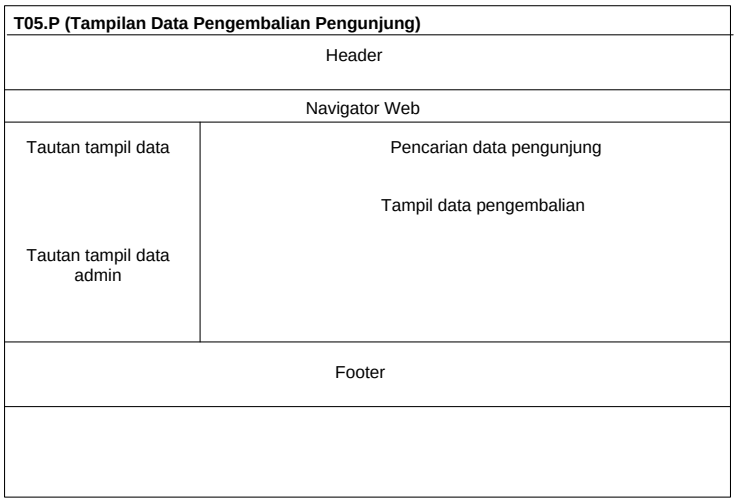
12. Perancangan tampilan data peminjaman



Gambar 3.43 Tampilan data peminjaman

```
Input           : -
Proses          : -
Output          : tampilan data peminjaman
Keterangan      : T04.P (Tampilan data peminjaman pengunjung)
```

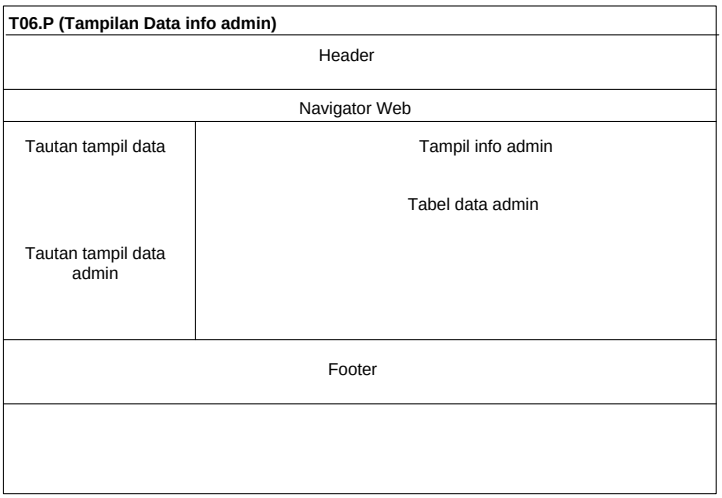
13. Perancangan tampilan data pengembalian



Gambar 3.44 Tampilan data pengembalian

```
Input           : -
Proses          : -
Output          : tampilan data pengembalian
Keterangan      : T05.P (Tampilan data pengembalian pengunjung)
```

14. Perancangan tampilan info data admin



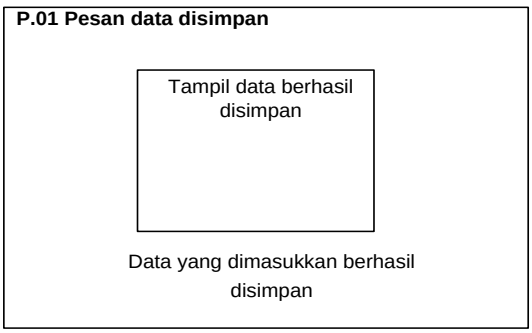
Gambar 3.45 Tampilan info data admin

```
Input          : -
Proses         : -
Output         : tampilan info data admin
Keterangan     : T06.P (Tampilan info data admin)
```

3.8.4 Perancangan Pesan

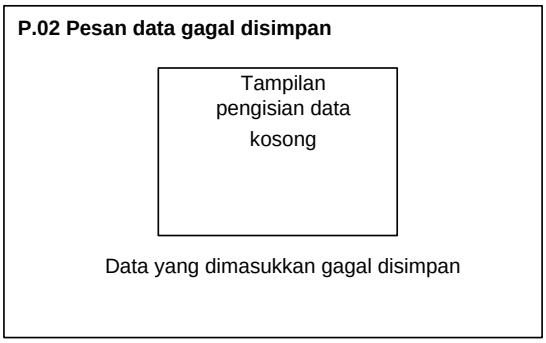
Dalam perancangan pesan disini dilakukan perancangan terhadap *form output*. Adapun *form-form* yang terdapat dalam sistem automasi perpustakaan di Perpustakaan Salman berbasis *Web* adalah sebagai berikut:

1. Perancangan pesan data disimpan



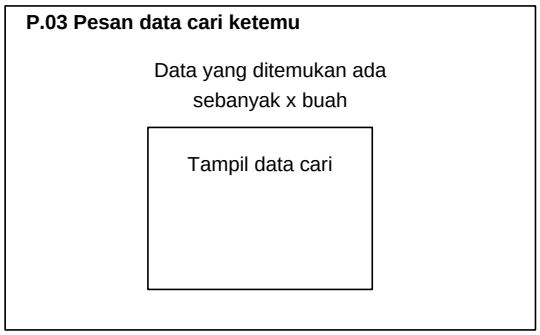
Gambar 3.46 Tampilan pesan data disimpan

2. Perancangan pesan data gagal disimpan



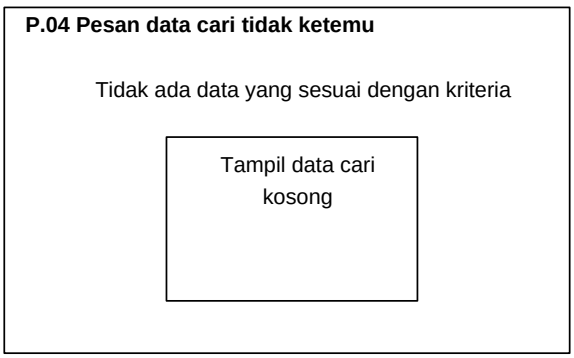
Gambar 3.47 Tampilan pesan data gagal disimpan

3. Perancangan pesan data cari ketemu



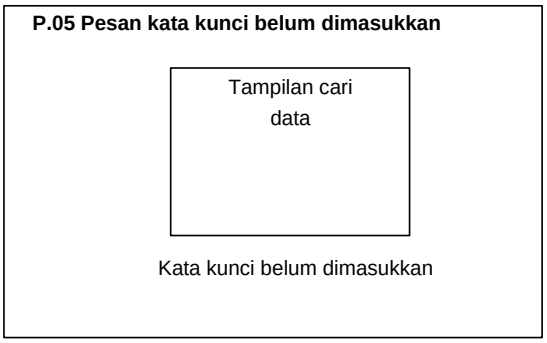
Gambar 3.48 Tampilan pesan data cari ketemu

4. Perancangan pesan data cari tidak ketemu



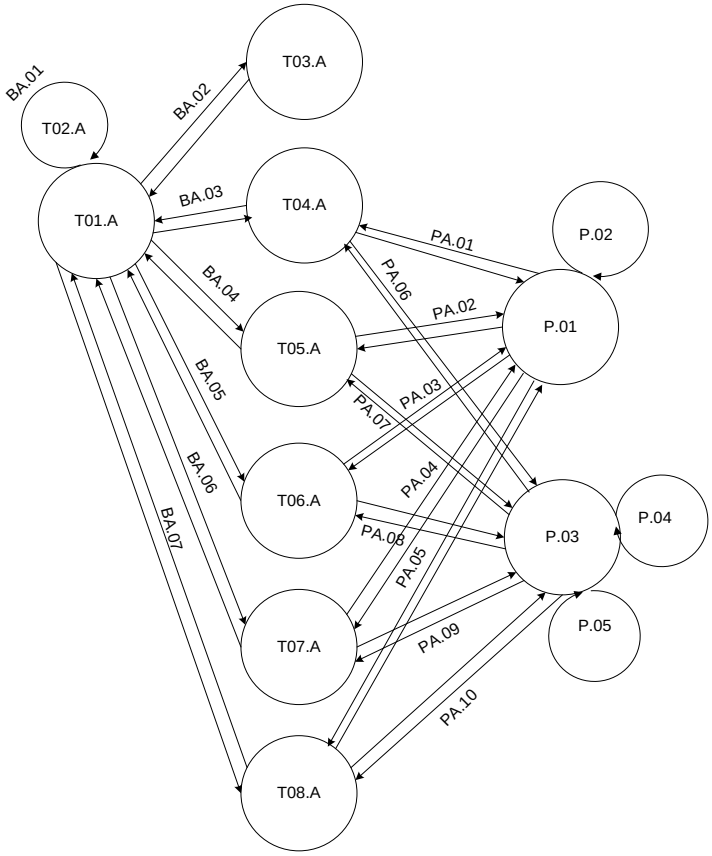
Gambar 3.49 Tampilan pesan data cari tidak ketemu

5. Perancangan pesan kata kunci belum dimasukkan



Gambar 3.50 Tampilan pesan kata kunci belum dimasukkan

3.9 Jaringan Semantik Admin



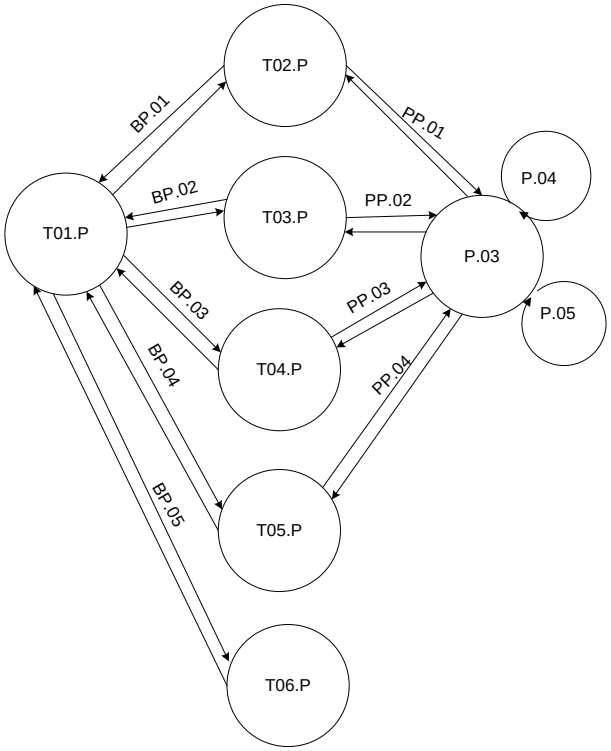
Gambar 3.51 Jaringan semantik Admin

3.10 Tabel Jaringan Semantik Admin

Simbol	Keterangan
T01.A	Tampilan Masuk Admin
T02.A	Tampilan Gagal Masuk Admin
T03.A	Tampilan Utama Admin
T04.A	Tampilan Tambah Anggota
T05.A	Tampilan Tambah Data Buku
T06.A	Tampilan Tambah Data Peminjaman
T07.A	Tampilan Tambah Data Pengembalian
T08.A	Tampilan Keluar Admin
P.01	Pesan Data Disimpan
P.02	Pesan Data Gagal Disimpan
P.03	Pesan Data Cari Ketemu
P.04	Pesan Data Cari Tidak Ketemu
P.05	Pesan Kata Kunci Belum Dimasukkan

Tabel 3.17 Tabel Jaringan Semantik Admin

3.11 Jaringan Semantik Pengunjung



Gambar 3.53 Jaringan semantik Pengunjung

3.12 Tabel Jaringan Semantik Pengunjung

Simbol	Keterangan
T01.P	Tampilan Utama Pengunjung
T02.P	Tampilan Data Anggota Pengunjung
T03.P	Tampilan Data Buku Pengunjung
T04.P	Tampilan Data Peminjaman Pengunjung
T05.P	Tampilan Data Pengembalian Pengunjung
T06.P	Tampilan Info Data Admin
P.03	Pesan Data Cari Ketemu
P.04	Pesan Data Cari Tidak Ketemu
P.05	Pesan Kata Kunci Belum Dimasukkan

Tabel 3.18 Tabel Jaringan Semantik Pengunjung