BAB 4

IMPLEMENTASI DAN PENGUJIAN SISTEM

4.1 Implementasi Sistem

4.1.1 Implementasi Perangkat Keras

Perangkat keras yang digunakan pada saat mengimplementasikan perangkat lunak dari Sistem Pencarian Informasi Buku di Perpustakaan Daerah Provinsi Jawa Barat adalah sebagai berikut :

1. Processor Intel(R) Core i3 2.1 GHz,
2. RAM 3GB,
3. VGA ATI Radeon HD 530v,
4. Hard Disk 320GB,
5. Monitor 14”, dan
6. Modem berkecepatan 153Kbps.

4.1.2 Implementasi Perangkat Lunak

Perangkat keras yang digunakan pada saat mengimplementasikan perangkat lunak dari Sistem Pencarian Informasi Buku di Perpustakaan Daerah Provinsi Jawa Barat adalah sebagai berikut :

1. Sistem operasi Windows 7 Professional,
2. Bahasa pemrograman PHP 5.3.0,
3. Database MySQL Server 5.0.37,
4. Browser Maxthon 3.4.2.2000,
5. Macromedia Dreamweaver CS5, dan
6. UML Modeler Power Designer 12.

4.1.3 Implementasi Basis Data

Tabel 4.1 Implementasi Basis Data

No.	Nama Tabel	SQL
1	tb_bobot	CREATE TABLE tb_bobot(id int(11) NOT NULL auto_increment, dok_id int(11) NOT NULL, idx bigint(20) NOT NULL, term varchar(30) NOT NULL, bobot float NOT NULL, PRIMARY KEY (`id`), KEY `term` (`term`), FOREIGN KEY (dok_id) REFERENCES tb_buku (id))ENGINE=InnoDB
2	tb_buku	CREATE TABLE tb_buku(id mediumint(9) NOT NULL auto_increment, judul varchar(100 NOT NULL, pengarang varchar(100) NOT NULL, id_golongan tinyint(4) NOT NULL, urlgambar varchar(210 default 'gambar-buku/default.jpg', deskripsi longtext NOT NULL, keyword NOT NULL, terakhir_ubah datetime NOT NULL, PRIMARY KEY (`id`), FOREIGN KEY (id_golongan) REFERENCES tb_golongan (id_golongan))ENGINE= InnoDB
3	tb_golongan	CREATE TABLE tb_golongan (id_golongan tinyint(4) NOT NULL auto_increment, golongan varchar(100) NOT NULL, terakhir_ubah datetime NOT NULL, PRIMARY KEY (`id_golongan`), UNIQUE KEY (golongan))ENGINE= InnoDB
4	tb_katadasar	CREATE TABLE tb_katadasar (id_ktdasar int(10) NOT NULL auto_increment, katadasar varchar(20) NOT NULL, tipe_katadasar varchar(20) NOT NULL, PRIMARY KEY (`id_ktdasar`))ENGINE= MyISAM
5	tb_stoplist	CREATE TABLE tb_stoplist (id_stoplist int(10) NOT NULL auto_increment, stoplist varchar(50) NOT NULL,

		PRIMARY KEY (`id_stoplist`))ENGINE= MyISAM
6	tb_user	CREATE TABLE tb_user (namauser varchar(30) NOT NULL, katakunci varchar(30) NOT NULL, PRIMARY KEY (namauser))ENGINE= MyISAM

4.1.4 Implementasi Kelas

Tabel 4.2 Implementasi Kelas

No.	Nama Kelas	Nama File Fisik	Nama File Executable
1	Database	database.php	database.php
2	PreProcessing	preprocessing.php	preprocessing.php
3	Stemming	stemming.php	stemming.php
4	Weighting	weighting.php	weighting.php
5	Svm	svm.php	svm.php
6	Menu	Menu.php	Menu.php
7	PencarianBuku	index.php	index.php
8	SvmSearch	SvmSearch.php	SvmSearch.php
9	DeskripsiBuku	deskripsi.php	deskripsi.php
10	Login	login.php	login.php
11	MainOperator	index.php	index.php
12	DaftarGolongan	DaftarGolongan.php	DaftarGolongan.php
13	DaftarBuku	DaftarBuku.php	DaftarBuku.php
14	OlahBuku	OlahBuku.php	OlahBuku.php
15	DetailBuku	DetailBuku.php	DetailBuku.php
16	HapusBuku	HapusBuku.php	HapusBuku.php
17	KonfirmasiHapusBuku	KonfirmasiHapusBuku.php	KonfirmasiHapusBuku.php
18	TambahBuku	TambahBuku.php	TambahBuku.php
19	UbahBuku	UbahBuku.php	UbahBuku.php
20	Optimisasi	optimisasi.php	optimisasi.php

4.1.5 Implementasi Antarmuka

Tabel 4.3 Implementasi Antarmuka

No.	Nama Kelas	Nama File Fisik	Nama File Executable
1	PencarianBuku	index.php	index.php
2	SvmSearch	SvmSearch.php	SvmSearch.php
3	DeskripsiBuku	deskripsi.php	deskripsi.php
4	Login	login.php	login.php
5	MainOperator	index.php	index.php

6	DaftarGolongan	DaftarGolongan.php	DaftarGolongan.php
7	DaftarBuku	DaftarBuku.php	DaftarBuku.php
8	OlahBuku	OlahBuku.php	OlahBuku.php
9	DetailBuku	DetailBuku.php	DetailBuku.php
10	HapusBuku	HapusBuku.php	HapusBuku.php
11	KonfirmasiHapusBuku	KonfirmasiHapusBuku.php	KonfirmasiHapusBuku.php
12	TambahBuku	TambahBuku.php	TambahBuku.php
13	UbahBuku	UbahBuku.php	UbahBuku.php
14	Optimisasi	optimisasi.php	optimisasi.php

4.2 Pengujian Sistem

4.2.1 Rencana Pengujian Aplikasi

Pengujian perangkat lunak Sistem Pencarian Informasi Buku di Perpustakaan ini menggunakan dua data uji yaitu data buku dan data golongan. Rencana pengujian selengkapnya dapat dilihat pada tabel 4.1:

Tabel 4.4 Rencana Pengujian Aplikasi Sistem Pencarian Informasi Buku

No	Use Case	Detail Pengujian	Jenis Pengujian
1	Pencarian Data Buku	Mengisi *query* dan menekan tombol, cari.	*Black Box*
2	Pencarian Data Buku Lebih Lanjut	Proses pelatihan.	*White Box*
		Proses klasifikasi.	*White Box*
3	Penyajian Deskripsi	Menampilkan identitas dan deskripsi buku.	*Black Box*
4	Login	Verifikasi data login (nama user dan kata kunci)..	*Black Box*
		Percobaan masuk ke salah satu halaman operator tanpa melakukan proses login.	*Black Box*
5	Tambah Data Buku	Melakukan proses penambahan data buku lalu melakukan proses *tokenizing*, *filtering*, dan *stemming*.	*Black Box*
6	Ubah Data Buku	Melakukan proses pengubahan data buku lalu melakukan proses *tokenizing*, *filtering*, dan *stemming*.	*Black Box*

7	Hapus Data Buku	Hapus data buku serta data tesnya(jika ada)	*Black Box*
8	Penyajian Detail Data Buku	Menampilkan informasi secara detail dari salah satu buku.	*Black Box*
9	Pencarian Data Buku Berdasarkan Judul	Melakukan proses pencarian data buku berdasarkan judul.	*Black Box*
10	Tambah Data Golongan	Melakukan proses penambahan data golongan.	*Black Box*
11	Ubah Data Golongan	Melakukan proses pengubahan data golongan.	*Black Box*
12	Optimisasi Sistem Pencarian	Pembobotan kata.	*Black Box*
		Pembuatan data tes.	*Black Box*

4.2.1.1 Kasus dan Hasil Pengujian

1. Pengujian Pencarian Data Buku

Tabel 4.5 Pengujian Pencarian Data Buku

Kasus dan Hasil Uji (Data Normal)				
Masukan	**Prosedur Pengujian**	**Keluaran yang Diharapkan**	**Hasil yang Didapat**	**Kesimpulan**
Deskripsi kebutuhan pengunjug (*query*)	Periksa data buku yang tampil, apakah dalam data buku terdapat *query*.	Menampilkan maksimal 10 informasi buku yang memiliki tingkat ukuran kemiripan tertinggi dengan *query*	Sistem melakukan proses yang diharapkan	Diterima
Kasus Dan Hasil Uji (Data Salah)				
Kosong atau tidak diisi	Amati hasil pencarian, pastikan tidak ada data buku yang tampil.	Menampilkan pesan kesalahan pencarian	Sistem melakukan proses yang diharapkan	Diterima

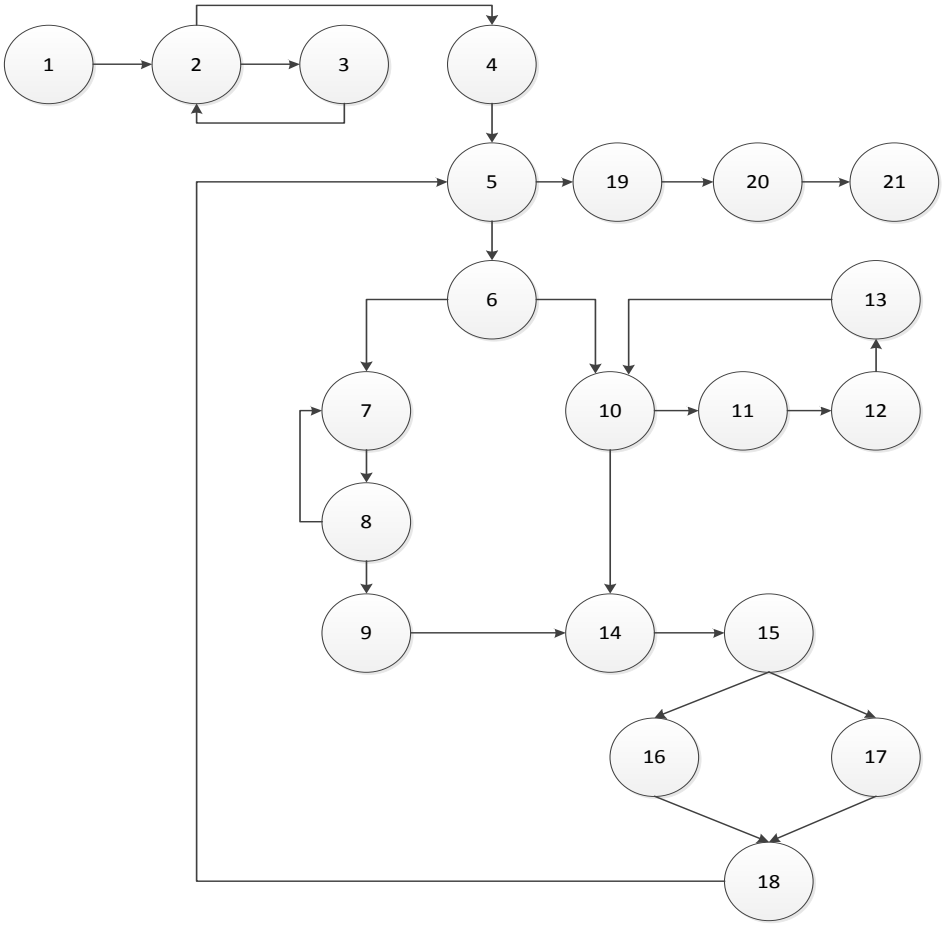
2. Pengujian Pencarian Data Buku Lebih Lanjut

2.1 Pengujain Proses Pelatihan

Pada proses pelatihan SVM bertujuan untuk menemukan vektor α dan konstanta b. Dalam proses pelatihan dibutuhkan satu set *input-output* data atau dalam kasus ini dibutuhkan dokumen relevan dan dokumen tidak relevan dimana penilaian relevan atau tidaknya suatu dokumen ditentutkan oleh pengguna. Dokumen-dokumen relevan diberi label 1 dan dokumen-dokumen tidak relevan deberi label -1.

Tujuan pengujian ini yaitu untuk mengetahui apakah proses pelatihan sudah berjalan sesuai dengan yang diharapkan. Berikut adalah pseudocode prosedur *train* yang merupakan proses komputasi untuk menemukan vektor α dan konstanta b.

```
Procedure train(dataFile, modelFile, testDataFile<-- null)
1:   recordCount ← loadDataFile(dataFile)
2:   FOR i=0 to recordCount
3:   lagrangeMults[i] ← 0
4:   ENDFOR
4:   numChanged ← 0
4:   examined ← 0
5:   WHILE(numChanged > 0 || examined = 0)
6:   numChanged ← 0
6:   IF(examined = 0) THEN
7    FOR i=0 to recordCount
8:   numChanged ← numChanged + examineExample(i)
9:   ENDFOR
10:  ELSE
10:  FOR each lagrangrMults as id => val
11:  IF(val ≠ 0 && val ≠ UPPER_BOUND) THEN
12:  numChanged ← numChanged(id)
12:  ENDIF
13:  ENDFOR
14:  ENDIF
15:  IF(examined = 0) THEN
16:  examined ← 1
17:  ELSE
17:  Examined ← 0
18:  ENDIF
19:  ENDWHILE
19:  IF(modelFile ≠ null && is_string(modelFile)) THEN
20:  writeSVM(modelFile)
21:  ENDIF
```



Gambar 4.1 *Flow Graph* Prosedur *train*

Jumlah dari *cyclomatics complexity* atau V(G) dari diagram alir pada gambar 4.1 adalah sebagai berikut :

V(G) = E – N + 2

V(G) = 35 – 24 + 2 = 13

Keterangan :

◯ = Menggambarkan Kondisi

→ = Menggambarkan Aksi

E = jumlah aksi, N = jumlah kondisi

Dari gambar 4.1 didapat *independent path* sebagai berikut :

Tabel 4.6 Pengujian Path pada Proses Pelatihan

Kasus dan Hasil Uji				
No. *Path*	**Node(n)**	**Kondisi**	**Hasil Uji**	**Kesimpulan**
1	1-2-3-2-4-5-19-20-21	Nilai examined = 1, Nilai numchanged=1	Path berjalan sesuai jalur	*Path* dikerjakan
2	1-2-3-2-4-5-6-7-8-9-14-15-16-18-5-19-20-21	Nilai examined=0, jumlah recordCount lebih dari 0	Path berjalan sesuai jalur	*Path* dikerjakan
3	1-2-3-2-4-5-6-7-8-9-14-15-17-18-5-19-20-21	Nilai examined=1, jumlah recordCount lebih dari 0	Path berjalan sesuai jalur	*Path* dikerjakan
4	1-2-3-2-4-5-6-10-14-15-16-18-5-19-20-21	Nilai examined=0, nilai lagrangeMults = 0 atau melebihi nilai konstanta cost	Path berjalan sesuai jalur	*Path* dikerjakan
5	1-2-3-2-4-5-6-10-14-15-17-18-5-19-20-21	Nilai examined=1, nilai lagrangeMults = 0 atau melebihi nilai konstanta cost	Path berjalan sesuai jalur	*Path* dikerjakan
6	1-2-3-2-4-5-6-10-11-12-13-10-14-15-16-18-5-19-20-21	Nilai examined=0, nilai lagrangeMults ≠ 0 dan tidak melebihi nilai konstanta cost	Path berjalan sesuai jalur	*Path* dikerjakan
7	1-2-3-2-4-5-6-10-11-12-13-10-14-15-17-18-5-19-20-	Nilai examined=1, nilai	Path berjalan sesuai jalur	*Path* dikerjakan

	21	lagrangeMults ≠ 0 dan tidak melebihi nilai konstanta cost		

Graph Matrix Proses Pelatihan

Tabel 4.7 *Graph Matrix* Proses Pelatihan

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1		1																			
2			1																		
3				1																	
4					1																
5						1													1		
6							1			1											
7								1													
8									1												
9														1							
10											1			1							
11												1									
12													1								
13														1							
14															1						
15																1	1				
16																		1			
17																		1			
18					1																
19																				1	
20																					1
21																					

Keterangan :

1. Baris dan kolom merepresentasikan simpul
2. Nilai 1 merepresentasikan adanya keterhubungan antar simpul

Kesimpulan Hasil Pengujian Proses Pelatihan

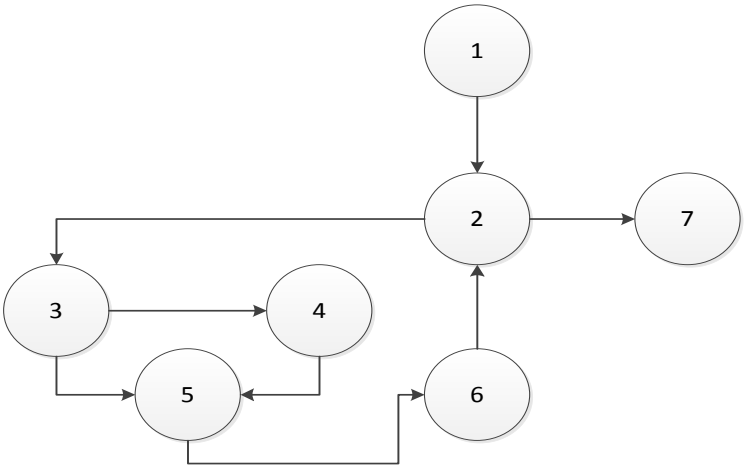
Dari hasil pengujian proses pelatihan yang dilakukan, maka didapat kesimpulan bahwa seluruh *node* yang ada pada setiap *independent path* telah dikerjakan , serta tidak terjadi pengulangan tak terhingga.

2.2 Pengujian Proses Klasifikasi

Proses klasifikasi bertujuan untuk mengklasifikasikan data buku kedalam kategori relavan dan tidak relevan terhadap data buku hasil *checklist* pengunjung. Data buku yang ditampilkan kepada pengunjung adalah data buku yang masuk dalam kategori relevan.

Tujuan pengujian ini yaitu untuk mengetahui apakah proses klasifikasi sudah berjalan sesuai dengan yang diharapkan. Berikut adalah pseudocode function *classify* yang merupakan proses komputasi untuk menilai data buku apakah masuk dalam kategori relevan atau tidak.

```
Function classify(input rowID)
1:  score ← 0
2:  FOR each value lagrangeMults as key => val
3:  IF(val > 0) THEN
4:  score = score + val * targets[key] * kernel(rowID, key)
5:  ENDIF
6:  ENDFOR
6:  return score - bias
```



Gambar 4.2 Flow Graph Function Classify

Jumlah dari *cyclomatics complexity* atau V(G) dari diagram alir pada gambar 4.2 adalah sebagai berikut :

V(G) = E – N + 2

V(G) = 8 – 7 + 2 = 3

Keterangan :

 = Menggambarkan Kondisi

 = Menggambarkan Aksi

E = jumlah aksi, N = jumlah kondisi

Dari gambar 4.2 didapat *independent path* sebagai berkut :

1: 1-2-6

2: 1-2-3-4-5-2-6

3: 1-2-3-5-2-6

Tabel 4.8 Pengujian Path pada Proses Pelatihan

Kasus dan Hasil Uji				
No. *Path*	**Node(n)**	**Kondisi**	**Hasil Uji**	**Kesimpulan**
1	1-2-6	Jumlah lagrangeMults=0	Path berjalan sesuai jalur	*Path* dikerjakan
2	1-2-3-4-5-2-6	Jumlah lagrangeMults lebih dari 0 dan nilai lagrangeMults lebih dari 0	Path berjalan sesuai jalur	*Path* dikerjakan
3	1-2-3-5-2-6	Jumlah lagrangeMults lebih dari 0 dan nilai lagrangeMults lebih sama dengan 0	Path berjalan sesuai jalur	*Path* dikerjakan

***Graph Matrix* Proses Klasifikasi**

Tabel 4.9 *Graph Matrix* Proses Klasifikasi

	1	2	3	4	5	6
1		1				
2			1			1
3				1	1	
4					1	
5		1				
6						

Keterangan :

1. Baris dan kolom merepresentasikan simpul
2. Nilai 1 merepresentasikan adanya keterhubungan antar simpul

Kesimpulan Hasil Pengujian Proses Klasifikasi

Dari hasil pengujian proses klasifikasi yang dilakukan, maka didapat kesimpulan bahwa seluruh *node* yang ada pada setiap *independent path* telah dikerjakan , serta tidak terjadi pengulangan tak terhingga.

3. Pengujian Penyajian Deskripsi

Tabel 4.10 Pengujian Penyajian Deskripsi

Kasus dan Hasil Uji (Data Normal)				
Masukan	**Prosedur Pengujian**	**Keluaran yang Diharapkan**	**Hasil yang Didapat**	**Kesimpulan**
Id buku	Amati deskripsi buku yang tampil, apakah id buku tersebut sesuai.	Menampilkan identitas buku serta deskripsi dari data buku yang dipilih	Sistem melakukan proses yang diharapkan	Diterima

4. Pengujian Login

Tabel 4.11 Pengujian Verifikasi Data Login

Kasus dan Hasil Uji (Data Normal)				
Masukan	**Prosedur Pengujian**	**Keluaran yang Diharapkan**	**Hasil yang Didapat**	**Kesimpulan**
Nama user dan kata kunci yang terdapat dalam *database*	Memastikan bisa mengakses halaman operator.	Menerima data nama user dan kata kunci serta menampilkan halaman utama operator	Sistem melakukan proses yang diharapkan	Diterima
Kasus Dan Hasil Uji (Data Salah)				
Nama user dan kata kunci yang tidak terdapat	Memastikan tidak bisa mengakses halaman opeator	Menolak data nama user dan kata kunci serta menampilkan pesan salah	Sistem melakukan proses yang diharapkan	Diterima

dalam *database*		login		

Tabel 4.12 Pengujian Percobaan Masuk dalam Lingkungan Operator Tanpa Login

Kasus dan Hasil Uji (Data Normal)				
Masukan	**Prosedur Pengujian**	**Keluaran yang Diharapkan**	**Hasil yang Didapat**	**Kesimpulan**
-	mengetik salah satu alamat halaman operator, memastikan menampilkan halaman login.	Menampilkan halaman login	Sistem melakukan proses yang diharapkan	Diterima

5. Pengujian Tambah Data Buku

Tabel 4.13 Pengujian Tambah Data Buku

Kasus dan Hasil Uji (Data Normal)				
Masukan	**Prosedur Pengujian**	**Keluaran yang Diharapkan**	**Hasil yang Didapat**	**Kesimpulan**
Data buku (Judul, pengarang, deskripsi buku, dan golongan) dan *file* gambar (opsional)	Memastikan data buku, file gambar, dan *keyword-nya* sudah tersimpan di *database*	Menyimpan data buku dan *file* gambar ke *database*, serta melakukan proses *tokenizing*, *filtering*, dan *stemming* pada data buku dan disimpan dalam *field keyword*	Sistem melakukan proses yang diharapkan	Diterima
Kasus Dan Hasil Uji (Data Salah)				
Data buku tidak diisi	Memastikan data buku, file gambar, dan *keyword-nya* tidak tersimpan di *database*	Menolak menyimpan data buku dan *file* gambar ke *database*	Sistem melakukan proses yang diharapkan	Diterima

6. Pengujian Ubah Data Buku

Tabel 4.14 Pengujian Ubah Data Buku

Kasus dan Hasil Uji (Data Normal)				
Masukan	**Prosedur Pengujian**	**Keluaran yang Diharapkan**	**Hasil yang Didapat**	**Kesimpulan**
Data buku (Judul, pengarang, deskripsi buku, dan golongan) dan *file* gambar (opsional)	Memastikan data buku, file gambar, dan *keyword-nya* sudah tersimpan di *database*	Data buku dan *file* gambar dapat diubah dan disimpan kembali ke *database*, serta melakukan proses *tokenizing*, *filtering*, dan *stemming* pada data buku dan disimpan dalam *field keyword*	Sistem melakukan proses yang diharapkan	Diterima
Kasus Dan Hasil Uji (Data Salah)				
Data buku tidak diisi	Memastikan data buku, file gambar, dan *keyword-nya* tidak tersimpan di *database*	Menolak menyimpan data buku dan *file* gambar ke *database*	Sistem melakukan proses yang diharapkan	Diterima

7. Pengujian Hapus Data Buku

Tabel 4.15 Pengujian Hapus Data Buku

Kasus dan Hasil Uji (Data Normal)				
Masukan	**Prosedur Pengujian**	**Keluaran yang Diharapkan**	**Hasil yang Didapat**	**Kesimpulan**
Id buku	Memastikan data buku, file gambar, dan data tes, terhapus dari database.	1. Menghapus data buku dari *database* 2. Jika ada *file* gambar untuk data buku tersebut maka *file* gambar	Sistem melakukan proses yang diharapkan	Diterima

		dihapus 3. Jika ada data tes dari data buku tersebut maka data tes dihapus		

8. Pengujian Penyajian Detail Buku

Tabel 4.16 Pengujian Pembuatan *Keyword*

Kasus dan Hasil Uji (Data Normal)				
Masukan	**Prosedur Pengujian**	**Keluaran yang Diharapkan**	**Hasil yang Didapat**	**Kesimpulan**
Id buku	Amati detail buku yang tampil, apakah id buku tersebut sesuai.	Menampilkan detail dari data buku yang dipilih	Sistem melakukan proses yang diharapkan	Diterima

9. Pengujian Pencarian Data Buku Berdasarkan Judul

Tabel 4.17 Pengujian Pencarian Data Buku Berdasarkan Judul

Kasus dan Hasil Uji (Data Normal)				
Masukan	**Prosedur Pengujian**	**Keluaran yang Diharapkan**	**Hasil yang Didapat**	**Kesimpulan**
Judul buku	Memastikan data buku yang tampil sesuai dengan judul/masukan.	Menampilkan data buku yang sesuai dengan judul/masukan	Sistem melakukan proses yang diharapkan	Diterima
Kosong atau tidak diisi	Memastikan seluruh data buku ditampilkan.	Menampilkan seluruh data buku	Sistem melakukan proses yang diharapkan	Diterima

10. Pengujian Tambah Data Golongan

Tabel 4.18 Pengujian Tambah Data Golongan

Kasus dan Hasil Uji (Data Normal)				
Masukan	**Prosedur Pengujian**	**Keluaran yang Diharapkan**	**Hasil yang Didapat**	**Kesimpulan**
Data golongan	Memastikan data golongan telah tersimpan di *database.*	Menyimpan data golongan ke *database*	Sistem melakukan proses yang diharapkan	Diterima
Kasus Dan Hasil Uji (Data Salah)				
Data golongan tidak diisi	Memastikan data golongan tidak tersimpan di *database.*	Menolak menyimpan data golongan ke *database*	Sistem melakukan proses yang diharapkan	Diterima
Data golongan yang sudah ada di *database*	Memastikan data golongan tidak tersimpan di *database.*	Menolak menyimpan data golongan ke *database*	Sistem melakukan proses yang diharapkan	Diterima

11. Pengujian Ubah Data Golongan

Tabel 4.19 Pengujian Ubah Data Golongan

Kasus dan Hasil Uji (Data Normal)				
Masukan	**Prosedur Pengujian**	**Keluaran yang Diharapkan**	**Hasil yang Didapat**	**Kesimpulan**
Data golongan	Memastikan data golongan telah diubah dan tersimpan di *database.*	Mengubah dan menyimpan kembali data golongan ke *database*	Sistem melakukan proses yang diharapkan	Diterima
Kasus Dan Hasil Uji (Data Salah)				
Data golongan tidak diisi	Memastikan data golongan tidak terubah dan tersimpan di *database.*	Menolak mengubah dan menyimpan data golongan ke *database*	Sistem melakukan proses yang diharapkan	Diterima
Data golongan yang sudah ada di *database*	Memastikan data golongan tidak terubah dan tersimpan di *database.*	Menolak mengubah dan menyimpan data golongan ke *database*	Sistem melakukan proses yang diharapkan	Diterima

12. Pengujian Optimisasi Sistem Pencarian

Tabel 4.20 Pengujian Pembobotan Kata

Kasus dan Hasil Uji (Data Normal)				
Masukan	**Prosedur Pengujian**	**Keluaran yang Diharapkan**	**Hasil yang Didapat**	**Kesimpulan**
Keyword	Memastikan setiap kata sudah dihitung bobotnya. Memastikan data tes untuk setiap data buku berhasil dibuat. Memastikan isi data tes sesuai dengan bobot dari data buku. Memastikan nama file sudah sesuai dengan yang diharapkan.	Menghitung bobot setiap kata yang terdapat di *keyword*, jika ada kata yang sama maka kata diwakili oleh satu nilai bobot saja kemudian nilai bobot disimpan ke *database.* Membuat data tes untuk setiap data buku dengan format file <idbuku>_<judul buku>	Sistem melakukan proses yang diharapkan	Diterima

4.2.1.2 Kesimpulan Hasil Pengujian

Berdasarkan hasil pengujian dengan kasus uji diatas dapat ditarik kesimpulan sebagai berikut :

1. Jika data yang dimasukan benar maka sistem secara fungsional akan mengeluarkan keluaran sesuai dengan harapan.
2. Jika data yang dimasukan salah maka sistem akan mengeluarkan pesan kesalahan dan data keluaran tidak akan sesuai dengan harapan.

Dengan kata lain bahwa pembangunan Sistem ini bebas kesalahan dan secara fungsional mengeluarkan hasil yang sesuai dengan yang diharapkan.

4.2.2 Pengujian Beta

Pengujian Beta merupakan pengujian yang dilakukan secara objektif dimana dilakukan pengujian secara langsung terhadap pengguna sistem dengan

menggunakan kuisioner mengenai kepuasan pengguna terhadap sistem yang telah dibangun. Adapun metode penilaian pengujian yang digunakan adalah metode kuantitatif berdasarkan data dari pengguna.

Adapun pertanyaan yang diajukan dalam kuisioner ini adalah sebagai berikut :

1. Apakah sistem pencarian informasi buku ini mudah untuk digunakan?

   [ ] Ya [ ] Tidak [ ] Tidak Tahu

2. Apakah sistem pencarian informasi buku ini cepat dalam melakukan proses pencarian?

   [ ] Ya [ ] Tidak [ ] Tidak Tahu

3. Apakah hasil pencarian dan hasil pencarian lebih lanjut dari sistem pencarian informasi buku dapat membantu Anda untuk menemukan buku yang sesuai dengan kebutuhan Anda?

   [ ] Ya [ ] Tidak [ ] Tidak Tahu

4.2.2.1 Hasil Pengujian Beta

Berdasarkan data hasil kuisioner tersebut, dicari prosentase masing-masing jawaban dengan menggunakan rumus R = P/Q * 100%, dimana :

P = Banyaknya jawaban responden tiap soal

Q = Jumlah responden

R = Nilai prosentase

1. Hasil Pertanyaan Nomor 1

Tabel 4.21 Hasil Pertanyaan Nomor 1

No	Keterangan	Responden	Prosentase
1	Ya	8	80%
2	Tidak	1	10%
3	Tidak Tahu	1	10%

2. Hasil Pertanyaan Nomor 2

Tabel 4.22 Hasil Pertanyaan Nomor

No	Keterangan	Responden	Prosentase
1	Ya	9	90%
2	Tidak	0	0
3	Tidak Tahu	1	10%

3. Hasil Pertanyaan Nomor 3

Tabel 4.23 Hasil Pertanyaan Nomor

No	Keterangan	Responden	Prosentase
1	Ya	8	80%
2	Tidak	0	0
3	Tidak Tahu	2	20%

4.2.2.2 Kesimpulan Hasil Pengujian Beta

Hasil dari pengujian beta dapat disimpulkan bahwa sistem pencarian informasi buku yang telah dibangun dapat membantu pengguna khususnya pengunjung perpustakaan untuk menemukan buku yang dibutuhkan, selain itu sistem pencarian informasi buku juga mudah untuk digunakan dan cukup cepat dalam proses pencarian.