BAB IV
IMPLEMENTASI DAN PENGUJIAN SISTEM

IV.1 Implementasi Sistem

Tahap implementasi sistem merupakan tahap untuk mengaplikasikan apa yang telah dirancang pada tahap perancangan sistem berdasarkan hasil analisis sistem. Implementasi sistem ini dilakukan sebagai wujud nyata hasil analisis dan perancangan sistem ke dalam bahasa pemrograman untuk diterapkan pada keadaan yang sesungguhnya. Hasil dari penerapan tersebut diharapkan dapat menjadi sebuah sistem yang siap diuji dan digunakan.

IV.1.1 Implementasi Basis Data

Data-data yang dibutuhkan pada sistem informasi ini dibuat menjadi tabel-tabel yang dimasukkan dalam suatu database. DBMS yang digunakan untuk mengolah data-data tersebut adalah MySQL.

Tabel IV.1 Implementasi Basis Data

No	Nama Tabel	Nama File
1	t_siswa	t_siswa.sql
2	t_nilai	t_nilai.sql
3	t_log_pengelompokan	t_log_pengelompokan.sql

a. Table t_siswa

```
1  CREATE TABLE IF NOT EXISTS `t_siswa` (
2    `nis` varchar(10) NOT NULL,
3    `nama` varchar(50) NOT NULL,
4    `jurusan` varchar(10) NOT NULL,
5    `tahun_ajaran` varchar(10) NOT NULL,
6    PRIMARY KEY (`nis`,`tahun_ajaran`),
7    KEY `tahun_ajaran` (`tahun_ajaran`)
8  ) ENGINE=InnoDB;
```

b. Table t_nilai

```
CREATE TABLE IF NOT EXISTS `t_nilai` (
  `id_nilai` int(11) NOT NULL AUTO_INCREMENT,
  `nis` varchar(10) NOT NULL,
  `mata_pelajaran` varchar(25) NOT NULL,
  `semester` int(11) NOT NULL,
  `nilai` float NOT NULL,
  `tahun_ajaran` varchar(10) NOT NULL,
   PRIMARY KEY (`id_nilai`),
   KEY `nis` (`nis`),
   KEY `tahun_ajaran` (`tahun_ajaran`)
   CONSTRAINT `t_nilai_ibfk_1` FOREIGN KEY (`nis`)
   REFERENCES `t_siswa` (`nis`) ON DELETE RESTRICT ON UPDATE
   CASCADE,
   CONSTRAINT `t_nilai_ibfk_2` FOREIGN KEY (`tahun_ajaran`)
   REFERENCES `t_siswa` (`tahun_ajaran`) ON DELETE RESTRICT
   ON UPDATE CASCADE
) ENGINE=InnoDB;
```

c. Table t_log_pengelompokan

```
CREATE TABLE IF NOT EXISTS `t_log_pengelompokan` (
  `no` int(11) NOT NULL AUTO_INCREMENT,
  `nis` varchar(10) NOT NULL,
  `kelompok` int(11) NOT NULL,
  `mata_pelajaran` varchar(25) NOT NULL,
  `nilai_rata_rata` float NOT NULL,
  `tahun_ajaran` varchar(10) NOT NULL,
  `keterangan` varchar(30) NOT NULL,
   PRIMARY KEY (`no`),
   KEY `nis` (`nis`),
   KEY `tahun_ajaran` (`tahun_ajaran`),
    CONSTRAINT `t_log_pengelompokan_ibfk_1` FOREIGN KEY
   (`nis`) REFERENCES `t_siswa` (`nis`) ON DELETE RESTRICT
   ON UPDATE CASCADE,
    CONSTRAINT `t_log_pengelompokan_ibfk_2` FOREIGN KEY
   (`tahun_ajaran`) REFERENCES `t_siswa` (`tahun_ajaran`) ON
   DELETE RESTRICT ON UPDATE CASCADE
) ENGINE=InnoDB;
```

IV.1.2 Implementasi Antar Muka

Implementasi antar muka menggambarkan tampilan dari perangkat lunak yang dibangun yaitu implementasi antarmuka perangkat lunak Pengelompokan Bimbingan Belajar Menggunakan Metode Clustering di SMA Negeri 1 Cilaku Kabupatan Cianjur.

Tabel IV.2 Implementasi Antar Muka

Antarmuka	Proses Ke-	Nama File
Import Data	1	UFormNilai.pas
Import Data Siswa	2	UFormNilai.pas
Import Data Nilai	3	UFormNilai.pas
Log Pengelompokan	4	UFormLogPengelompokan.pas
Hitung nilai rata-rata permata pelajaran	4.1	UFormLogPengelompokan.pas
Klastering	4.2	UFormLogPengelompokan.pas
Report	5	UFormLogPengelompokan.pas

IV.2 Pengujian Sistem

Pengujian sistem merupakan tahapan untuk melakukan serangkaian tes untuk mencoba sistem yang telah dibangun dengan tujuan mengetahui bagaimana kualitas dari suatu sistem tersebut.

IV.2.1 Rencana Pengujian

Pengujian yang akan dilakukan adalah dengan menggunakan metode *white box* dan *black box* serta pengujian beta dengan mewawancarai *user* yang akan menggunakan perangkat lunak ini.

Tabel IV.3 Rencana Pengujian

No	Item Uji	Detail Pengujian	Jenis Uji
1	Algoritma K-means	Uji algoritma	*White box*
2	Import data Siswa	Pengujian validasi import data siswa	*Black Box*
3	Import data Nilai	Pengujian validasi import data nilai	*Black Box*
4	Klastering	Pengujian proses klastering	*Black Box*
5	Simpan	Pengujian penyimpanan data hasil klastering	*Black Box*
6	Report	Pengujian penampilan laporan hasil klastering	*Black Box*

IV.2.2 Pengujian White Box

Pengujian *white box* bertujuan untuk mengetahui kinerja logika yang dibuat pada sebuah perangkat lunak apakah berjalan dengan baik atau tidak. Pengujian *white box* akan digunakan pada algoritma K-means untuk mengukur kinerja logika berdasarkan *pseudocode* yang telah dibuat pada tahap analisis. Adapun tahapan yang ada pada pengujuan *white box* yaitu:

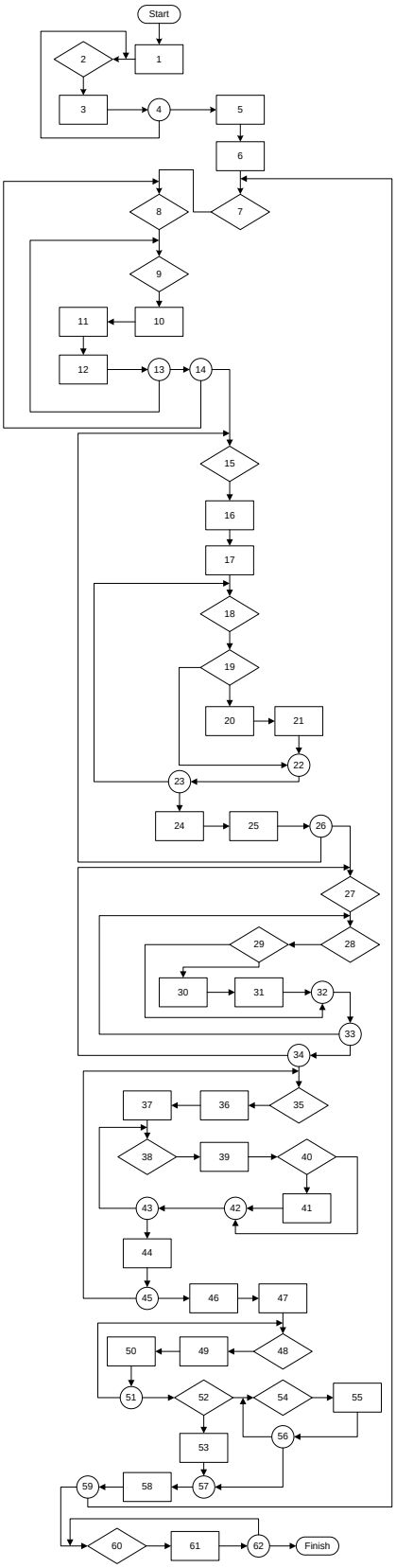
Pengubahan pseucode menjadi *flowchart* kemudianmenjadi *flowgraph.*

Tabel IV.4 Pseudocode Algoritma K-means *Clustering*

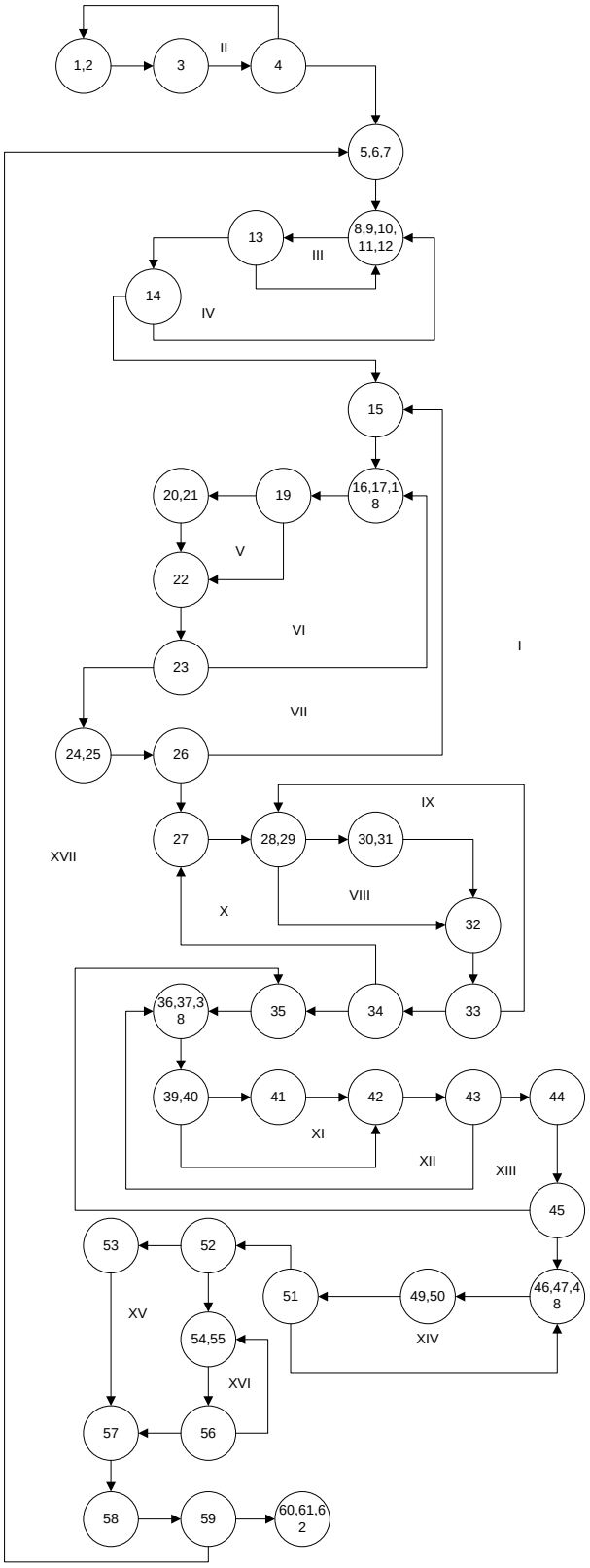
No	Pseudocode
1	**input**(k)
2	**for** i←1 to k do
3	pusat[i]←random(n)
4	**Endfor**
5	beres←**false**
6	maxiter←0
7	**while** (beres = false **or** maxiter <= 50) **do**
8	**for** i←1 **to** n **do**
9	**for** j←1 **to** k **do**
10	min[j]←a[i]-pusat[k]
11	kuadrat[i]←min[j]*min[j]
12	jarak[i,j]←sqrt(kuadrat[i])
13	**Endfor**
14	**Endfor**
15	**for** i←1 **to** N **do**
16	dist←jarak[i,1]
17	cluster←1
18	**for** j←2 **to** k **do**
19	**if** jarak[i,j] < dist **then**
20	dist←jarak[i,j]
21	cluster←j
22	**Endif**
23	**Endfor**
24	anggota[i].nilai←data[i]

No	Pseudocode
25	anggota[i].klaster←cluster
26	**Endfor**
27	**for** i←1 **to** k **do**
28	**for** j←1 **to** n **do**
29	**if** anggota[j].klaster = i **then**
30	c ← anggota[j].nilai
31	hasil[i,j] ← c
32	**endif**
33	**endfor**
34	**Endfor**
35	**for** i←1 **to** k **do**
36	jum ← 0
37	byk ← 0
38	**for** j←1 **to** n **do**
39	jum ← jum + hasil[i,j]
40	**if** hasil[i,j] = 0 **then**
41	byk ← byk + 1
42	**Endif**
43	**Endfor**
44	centroid[i] ← jum / (j-1-byk)
45	**Endfor**
46	temp ← 0
47	temp1 ← 0
48	**for** i←1 **to** k **do**
49	temp ← temp + pusat[i]
50	temp1 ← temp1 + centroid[i]
51	**Endfor**
52	**if** temp = temp1 **then**
53	beres ← true
	Else
54	**for** j←1 **to** k **do**
55	pusat[j] ← centroid[j]
56	**endfor**
57	**Endif**
58	maxiter ← maxiter + 1

No	Pseudocode
59	**Endwhile**
60	**for** i←1 **to** n **do**
61	write(anggota[i].nilai,’ ’, anggota[i].klaster)
62	**endfor**



Gambar IV.1 Flowchart algoritma K-means



Gambar IV.2 Flowgraph algoritma K-means

1. **Cyclometic Complexity**

   V(G) = E - N + 2

   V(G) = 55 – 40 + 2

   V(G) = 17

2. **Hitung Jumlah Region**

   Jumlah Region = Jumlah Kurva Tertutup + 1 (Region Terluar)

   Jumlah Region = 16 + 1

   Jumlah Region = 17

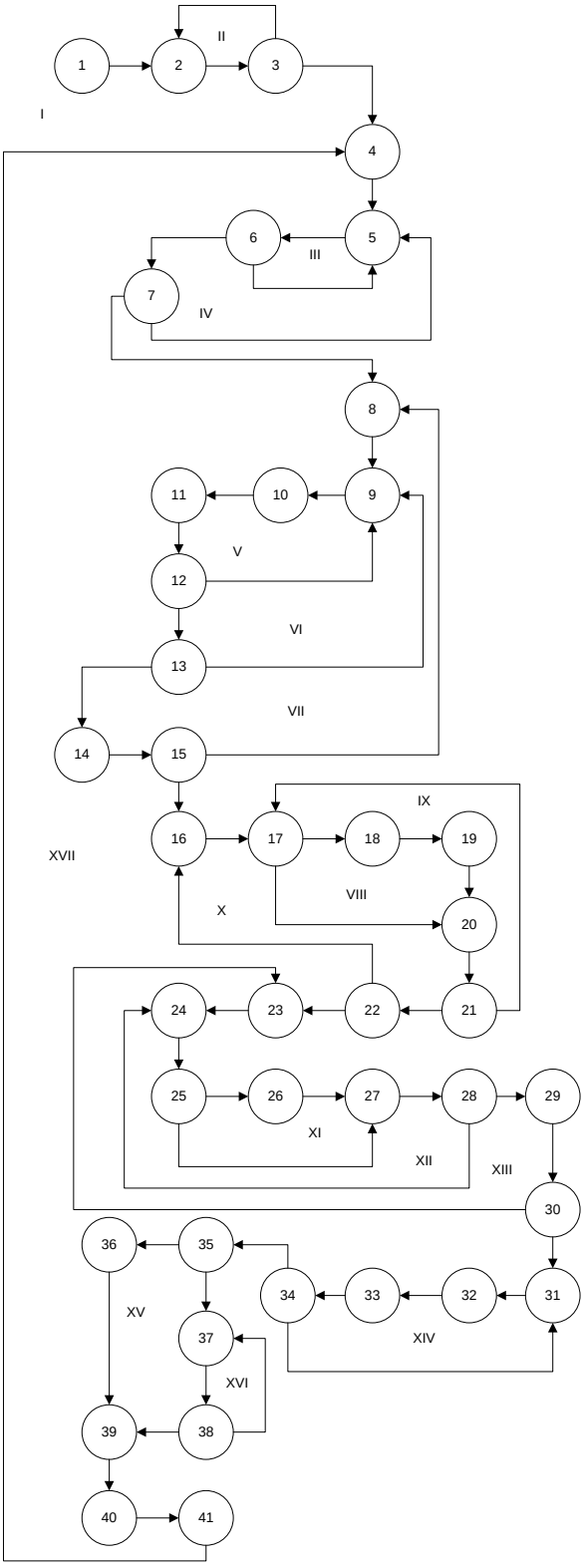
3. **Hitung jumlah predicate Node**

   $V(G) = Jumlahpredicate\ node + 1$

   $V(G) = 16 + 1$

   $V(G) = 17$

4. Penomoran ulang flowgraph



Gambar IV.3 Penggambaran ulang flowgraph

5. Graph matriks

IV.5 Tabel Graph Matriks

Node	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	Jumlah
1		1																																							0
2			1																																						0
3		1		1																																					1
4					1																																				0
5						1																																			0
6					1		1																																		1
7					1			1																																	1
8									1																																0
9										1																															0
10											1	1																													1
11												1																													0
12													1																												0
13								1						1																											1
14															1																										0
15								1								1																										1
16																	1																								0
17																		1	1																						1
18																			1																						0
19																				1																					0
20																	1				1																				1
21																1						1																			1
22																							1																		0
23																								1																	0
24																									1	1															1
25																										1															0
26																											1														0
27																							1					1													1
28																													1												0
29																						1								1											1
30																															1										0
31																																1									0
32																														1			1								1
33																																		1	1						1
34																																			1						0
35																																				1					0
36																																			1		1				1
37																																						1			0
38																																							1		0
39				1																																				1	1
40																																									0
Jumlah																																									16

IV.2.3 Pengujian Black Box

Pengujian *black box* dilakukan dengan tujuan untuk mengetahui apakah data yang dimasukkan sudah benar atau belum serta bagaimana hasil keluarannya apakah sesuai dengan harapan atau tidak. Teknik pengujian *black box* yang digunakan adalah *Equivalence Partitioning* dan *Cause-Effect Relationship*.

a. Import data siswa

Tabel di bawah ini merupakan skenario pengujian menggunakan teknik pengujian *Equivalence Partitioning Testing* yang dilakukan pada bagian import data siswa.

Tabel IV.6 Pengujian import file excel

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	***Yang Diharapkan***	***Pengamatan***	***Kesimpulan***
Alamat file excel data siswa yang akan diimport ke dalam perangkat lunak	Data berhasil diimport	Dapat menampilkan tabel data yang telah diimport	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Kosong)			
Data Masukan	***Yang Diharapkan***	***Pengamatan***	***Kesimpulan***
Alamat file excel data siswa yang akan diimport dikosongkan.	Muncul pesan kesalahan file excel belum dipilih	Menampilkan pesan kesalahan alamat file excel harus di isi.	[ ] Diterima [✓] Ditolak

b. Import data nilai

Tabel di bawah ini merupakan skenario pengujian menggunakan teknik pengujian *Equivalence Partitioning Testing* yang dilakukan pada bagian import data siswa.

Tabel IV.7 Pengujian import file excel

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	***Yang Diharapkan***	***Pengamatan***	***Kesimpulan***
Alamat file excel data nilai yang akan	Data berhasil diimport	Dapat menampilkan tabel data yang telah	[✓] Diterima [ ] Ditolak

diimport ke dalam perangkat lunak		diimport	
Kasus dan Hasil Uji (Data Kosong)			
Data Masukan	***Yang Diharapkan***	***Pengamatan***	***Kesimpulan***
Alamat file excel data nilai yang akan diimport dikosongkan.	Muncul pesan kesalahan file excel belum dipilih	Menampilkan pesan kesalahan alamat file excel harus di isi.	[ ] Diterima [✓] Ditolak

c. Klastering

Tabel di bawah ini merupakan skenario pengujian menggunakan teknik pengujian *Equivalence Partitioning Testing* yang dilakukan pada bagian proses klastering

Tabel IV.8 Pengujian klastering

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	***Yang Diharapkan***	***Pengamatan***	***Kesimpulan***
Data penentuan klastering	Data berhasil diklaster	Dapat menampilkan tabel data hasil klastering berdasarkan data penentuan klastering	[✓] Diterima [ ] Ditolak
Kasus dan Hasil Uji (Data Kosong)			
Data Masukan	***Yang Diharapkan***	***Pengamatan***	***Kesimpulan***
Salahsatu atau lebih data penentuan klastering dikosongkan	Muncul pesan kesalahan data penentuan klastering kurang lengkap	Menampilkan pesan kesalahan data penentuan klastering kurang.	[ ] Diterima [✓] Ditolak

d. Simpan

Tabel di bawah ini merupakan skenario pengujian menggunakan teknik pengujian *Equivalence Partitioning Testing* yang dilakukan pada bagian simpan hasil klastering

Tabel IV.9 Pengujian proses simpan

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	***Yang Diharapkan***	***Pengamatan***	***Kesimpulan***
Data hasil klastering	Data hasil klastering berhasil disimpan	Dapat menyimpan data hasil klastering ke dalam *database*	[✓] Diterima [ ] Ditolak

e. Report

Tabel di bawah ini merupakan skenario pengujian menggunakan teknik pengujian *Equivalence Partitioning Testing* yang dilakukan pada bagian report

Tabel IV.10 pengujian proses report

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	***Yang Diharapkan***	***Pengamatan***	***Kesimpulan***
Pemanggilan report melalui fungsi tombol	Laporan daata hasil klastering berdasarkan mata pelajaran dan taahun ajaran	Dapat menampilkan data hasil klastering dari *database* dalsm bentuk report	[✓] Diterima [ ] Ditolak

IV.2.4 Pengujian Beta

Pengujian beta merupakan pengujian yang dilakukan secara objektif dimana diuji secara langsung ke lapangan, dengan menggunakan wawancara mengenai kepuasan *user* terhadap perangkat lunak yang telah dibangun, dengan kandungan point syarat *user friendly* serta apakah sesuai dengan yang diharapkan.

IV.2.4.1 Skenario Pengujian Beta

Pengujian beta ini dilakukan dengan cara melakukan pengujian langsung di tempat penelitian. Penelitian ini dilakukan di SMA Negeri 1 Cilaku kepada 1 pengguna, yaitu wakil kepala sekolah urusan kurikulum untuk mengetahui sejauh

mana perangkat lunak yang telah dibangun ini membantu pihak sekolah dalam pengelompokan siswa bimbingan belajar pertiap mata pelajaran.

Adapun beberapa poin yang diteliti adalah :

1. Tampilan antar muka yang user friendly.
2. Aplikasi dapat membantu pihak sekolah dalam pengelompokan siswa bimbingan belajar.
3. Ketepatan hasil pengelompokan dengan menggunakan metode *Clustering* .

IV.2.4.2 Skenario Penguujian Beta Pengguna

Pengujian beta yang dilakukan terhadap pengguna adalah dengan wawancara secara langsung kepada *user*. Berikut merupakan daftar pertanyaan yang diajukan :

1. Apakah tombol, tampilan tabel, dan tampilan laporan pada perangkat lunak pengelompokan bimbingan belajar ini mudah dipelajari ?
2. Apakah perangkat lunak pengelompokan bimbingan belajar ini mudah digunakan ?
3. Apakah perangkat lunak pengelompokan bimbingan belajar ini dapat membantu anda dalam melakukan proses pengelompokan siswa yang menjadi peserta bimbingan belajar ?
4. Bagaimana menurut anda mengenai hasil pengelompokan dari perangkat lunak pengelompokan bimbingan belajar ?
5. Bagaimana menurut anda mengenai keseluruhan perangkat lunak pengelompokan bimbingan belajar ?

Brikut merupakan jawaban hasil wawancara :

1. Jawaban pertanyaan pertama

   “ya, mudah dipelajari. Sedikit tombol yan digunakan, tampilantabel dan laporan pun mudah dimengerti dan dipelajari”.
2. Jawaban pertanyaan kedua

   “ya, cukup mudah digunakan karena simple. Tidak banyak ngetik karena cukup klik tombol yang telah disediakan”.

3. Jawaban pertanyaan ketiga
   "membantu sekali. Karen dengan aplikasi ini memudahkan pihak sekolah dalam proses pengelompokan siswa bimbingan belajar".
4. Jawaban pertanyaan keempat
   "dari hasil pengelompokan aplikasi ini, terlihat jelas pengelompokan nilai siswa sudah sesuai dengan kelompoknya masing-masing"
5. Jawaban pertanyaan kelima
   "sudah baik, tidak ada kesalahan di aplikasinya, tujuan yang diminta pihak sekolah sudah tercapai, jadi secara keseluruhan sudah oke."

IV.2.5 Ksimpulan Pengujian

Berdasarkan hasil pengujian *white box, black box,* dan beta yang telah dilakukan, dapat diambil kesimpulan sebagai berikut:

1. Perangkat lunak yang dibangun dapat membantu pihak sekolah dalam pengelompokan siswa calon peserta bimbingan belajar.
2. Perangkat lunak yang dibangun dapt menghasilkan pengelompokan yang *valid* sesuai dengan metode yang dipakai.
3. Berdasarkan pengujian pada setiap metode, dihasilkan nilai Cyclomatic Complexity yang sama yaitu 17. Maka dapat disimpulkan bahwa pengujian *white box* pada algoritma K-Means *clustering* berjalan dengan baik, karena setiap pengujian menghasilkan nilai yang sama.