BAB IV

IMPLEMENTASI DAN PENGUJIAN

Pada bab ini akan dilakukan implementasi dan pengujian terhadap aplikasi data mining yang telah dibangun. Tahapan ini dilakukan setelah perancangan selesai dilakukan dan selanjutnya akan diimplementasikan pada bahasa pemrograman. Setelah implementasi dilakukan, maka dilakukan pengujian terhadap aplikasi data mining yang telah dibangun dan akan dilihat kekurangan-kekurangan pada aplikasi tersebut untuk pengembangan selanjutnya.

4.1. Implementasi Sistem

Implementasi sistem merupakan tahap meletakkan sistem sehingga siap untuk dioperasikan. Setelah sistem dianalisis dan didesain secara rinci, maka akan menuju tahap implementasi. Tujuan implementasi sistem adalah untuk mengkonfirmasikan modul program perancangan pada para pelaku sistem sehingga *user* dapat memberi masukan kepada pembangun sistem.

4.1.1. Implementasi Perangkat Keras

Spesifikasi perangkat keras yang digunakan untuk membangun aplikasi *data mining* ini adalah sebagai berikut:

1. *Motherboard* : *ASUS P4 S533X*
2. *Processor* : *Intel Pentium 4 (R)* CPU 1700 Mhz
3. RAM : DDR1 384 Mb

4. *Harddisk* : 40 GB
5. Monitor : 15”
6. *Keyboard* : *Standard*
7. *Mouse* : *Standard*

4.1.2. Implementasi Perangkat Lunak

Perangkat lunak yang digunakan untuk membangun aplikasi *data mining* ini adalah sebagai berikut :

1. *Operating System* : *Microsoft Windows XP Professional Service Pack 3*
2. *Developer tool* : *Borland Delphi 7*
3. *DBMS : MySQL* 5.0.45 -community-nt
4. *MySQL manager and admin tool : SQLyog*
5. *UML modeling tool : Visual Paradigm for UML 7*

4.1.3. Implementasi Basis Data

Pembuatan *database* dilakukan dengan menggunakan bantuan aplikasi SQLyog sebagai *GUI tool* dalam merancang dan membangun *database* aplikasi *data mining* di MySQL. Berikut implementasi pembuatan *database* dalam bahasa SQL berserta tabel-tabel yang mendukung aplikasi data mining.

1. Pembuatan d*atabase*

```
CREATE DATABASE `dmd3-labs`;
```

2. Pembuatan tabel kasus

```
CREATE TABLE `kasus` (
`NASABAH_ID` int(9) unsigned NOT NULL,
`NAMA_NASABAH` varchar(20) NOT NULL,
`ALAMAT` varchar(50) NOT NULL default '',
`JENIS_KELAMIN` char(1) NOT NULL,
`TEMPATLAHIR` varchar(9) NOT NULL,
`TGLLAHIR` date NOT NULL,
`UMUR` tinyint(2) unsigned NOT NULL,
`JENIS_ID` char(3) NOT NULL,
`KODE_AGAMA` char(5) NOT NULL,
`DESA` varchar(20) default NULL,
`KECAMATAN` varchar(14) NOT NULL,
`KOTA_KAB` char(7) NOT NULL,
`PROPINSI` char(9) NOT NULL,
`VERIFIKASI` tinyint(1) unsigned NOT NULL,
`TGL_REGISTER` date NOT NULL,
`kodepos` smallint(5) unsigned NOT NULL,
`KODE_KANTOR` tinyint(2) unsigned NOT NULL,
`STATUS_MARITAL` char(7) NOT NULL,
`BI_JENIS_PENGGUNAAN` tinyint(2) unsigned NOT NULL,
`BI_SEKTOR_EKONOMI` smallint(4) unsigned NOT NULL,
`JML_PINJAMAN` mediumint(7) unsigned NOT NULL,
`NILAI_PINJAMAN` char(6) NOT NULL,
`JML_ANGSURAN` tinyint(2) unsigned NOT NULL,
`KODE_INTEGRASI` tinyint(2) unsigned NOT NULL,
`STATUS` tinyint(1) unsigned NOT NULL,
`JENIS_AGUNAN` tinyint(1) unsigned NOT NULL,
`CEK` varchar(11) NOT NULL,
`id_kasus` char(3) default '',
KEY `FK_kasus` (`id_kasus`),
CONSTRAINT `FK_kasus` FOREIGN KEY (`id_kasus`)
REFERENCES `detailkasus` (`Id_kasus`)
);
```

3. Pembuatan tabel parameter

```
 CREATE TABLE `parameter` (
`nama_parameter` varchar(30) NOT NULL,
`aktif` char(1) default 'Y',
`target` char(1) default 'T',
 PRIMARY KEY  (`nama_parameter`)
 );
```

4. Pembuatan tabel tree

```
 CREATE TABLE `tree` (
`Node` int(11) default NULL,
`Parameter_Node` varchar(30) default NULL,
`Induk` varchar(30) default NULL,
```

```
`parameter_Induk` varchar(30) default NULL,
`Ujung` char(1) default 'Y',
`id_kasus` char(3) default NULL,
KEY `FK_tree` (`id_kasus`),
CONSTRAINT `FK_tree` FOREIGN KEY (`id_kasus`)
REFERENCES `detailkasus` (`Id_kasus`)
);
```

5. Pembuatan tabel kerja

```
 CREATE TABLE `kerja` (
`nama_parameter` varchar(30) default NULL,
`gain` decimal(15,2) default NULL,
KEY `FK_kerja` (`nama_parameter`),
CONSTRAINT `FK_kerja` FOREIGN KEY (`nama_parameter`)
REFERENCES `parameter` (`nama_parameter`)
);
```

6. Pembuatan tabel subkerja

```
  CREATE TABLE `sub_kerja` (
`nama_parameter` varchar(30) default NULL,
`nilai` varchar(255) default NULL,
`entropy` decimal(15,2) default NULL,
`result_1` varchar(30) default NULL,
`result_2` varchar(30) default NULL,
`jml_kasus` int(11) default NULL,
KEY `FK_sub_kerja` (`nama_parameter`),
CONSTRAINT `FK_sub_kerja` FOREIGN KEY (`nama_parameter`)
REFERENCES `parameter` (`nama_parameter`)
);
```

7. Pembuatan tabel detailkasus

```
 CREATE TABLE `detailkasus` (
`Id_kasus` char(3) NOT NULL default '',
`nama_parameter` varchar(255) default NULL,
PRIMARY KEY  (`Id_kasus`),
KEY `FK_detailkasus` (`nama_parameter`),
CONSTRAINT `FK_detailkasus` FOREIGN KEY (`nama_parameter`)
REFERENCES `parameter` (`nama_parameter`)
);
```

8. Pembuatan tabel *user*

```
  CREATE TABLE `user` (
```

```
`password` varchar(255) default '',
`checked` varchar(255) NOT NULL default 'T'
);
```

4.1.4. Implementasi Kelas

Implementasi kelas dilakukan berdasarkan hasil perancangan kelas yang telah dirancang. Berikut implementasi kelas dari aplikasi yang dibangun,

1. TLoginForm

```
type
  TLoginForm = class(TForm)
  private
    { Private declarations }
    procedure login;
    procedure batal;
  public
    { Public declarations }
  end;
```

2. TLogin

```
Type TLogin = class
  private
  isipassword:string;
  public
  procedure get_password(n:string);
  procedure verifikasi;
  function validasi:string;
end;
```

3. TMainForm

```
type
  TMainForm = class(TForm)
  private
    { Private declarations }
    procedure show_importdataform;
    procedure show_parameterform;
    procedure show_pohonkeputusan;
    procedure show_aturanprediksi;
    procedure show_prediksikriteria;
    procedure show_prediksinasabah;
    procedure show_laporanform;
  public
    { Public declarations }
  end;
```

4. TImportDataForm

```
type
  TImportDataForm = class(TForm)
  private
    { Private declarations }
    procedure caridata;
    procedure call_import;
    procedure hapusdata;
  public
    { Public declarations }
  end;
```

5. TImportData

```
Type TImportData = class(TSMImportFromXLS)
  private
  lokasidata:string;
  public
  procedure get_lokasidata(n:string);
  procedure import;
end;
```

6. TParameterForm

```
type
  TParameterForm = class(TForm)
  private
    procedure simpan_pengaturan;
    { Private declarations }
  public
    { Public declarations }
  end;
```

7. TParameter

```
Type TParameter = class
  private
  public
  procedure set_parameter1;
  procedure unset_parameter1;

  procedure set_parameter2;
  procedure unset_parameter2;

  procedure set_parameter3;
  procedure unset_parameter3;

  procedure set_parameter4;
  procedure unset_parameter4;
  procedure set_parameter5;
  procedure unset_parameter5;


  procedure set_parameter6;
```

```
  procedure unset_parameter6;

  procedure set_parameter7;
  procedure unset_parameter7;

  procedure set_parameter8;
  procedure unset_parameter8;

end;
```

8. TPohonKeputusan

```
type
  TPohonKeputusan = class(TForm)
  private
    { Private declarations }
    procedure buat_node();
    procedure buat_tree();
  public
    { Public declarations }
  end;

var
  PohonKeputusan: TPohonKeputusan;
  level : integer;
   hasil,tampil,node,parameter_terpakai,
   nilai_parameter_terpakai, NoCabang : TStringList;
  x, y, parameter_hasil : string;
  induk : TTreeNode;
  jml_hasil, v_jml_kasus, v_sub_jml_kasus, jml_node : integer;
  entropi1:Tentropi;
```

9. TEntropi

```
Type Tentropi = class
  private
  jml_all_kasus, jml_kasus : integer;
  public
  procedure set_all_kasus(n:integer);
  procedure set_kasus(n:integer);
  function get_all_kasus:integer;
  function get_kasus:integer;
  function hitung_entropi:real;
end;
```

10. TPrediksi

```
 type
   TPrediksi = class(TForm)
   private
     { Private declarations }
     procedure buat_prediksi_berpotensi();
     procedure  tampilkan_prediksi_berpotensi(node  :  integer;
 parameter_induk : string);
     procedure buat_prediksi_tidakberpotensi();
```

```
    procedure    tampilkan_prediksi_tidakberpotensi(node    :
integer; parameter_induk : string);
  public
    { Public declarations }
  end;

var
  Prediksi: TPrediksi;
  no_prediksi:integer;
  daftar : TListItem;
  level:integer;
  Prediksi2: TPrediksi;
  no_prediksi2:integer;
  daftar2 : TListItem;
  level2:integer;
```

11. TPrediksiKriteriaForm

```
type
  TPrediksiKriteriaForm = class(TForm)
  private
    { Private declarations }
    // procedure cari_node(node:integer);
     procedure cari_node2(node:integer);
     procedure cari_node3(node:integer);
     procedure cari_node4(node:integer);
     procedure cari_node5(node:integer);
     procedure cari_node6(node:integer);
     procedure cari_node7(node:integer);
     procedure cari_node8(node:integer);
  public
    { Public declarations }
  end;
```

12. TPrediksiNasabahForm

```
type
  TPrediksiNasabahForm = class(TForm)
  private
    { Private declarations }
    procedure prediksi_nasabahpotensi;
    procedure prediksi_nasabah_tdkpotensi;
  public
    { Public declarations }
  end;
```

13. TLaporanForm

```
type
  TLaporanForm = class(TForm)
  private
```

```
    { Private declarations }
    procedure buat_laporan_tidakberpotensi;
    procedure buat_laporan_berpotensi;
  public
    { Public declarations }
    procedure buat_laporan_nasabahtidakberpotensi;
    procedure buat_laporan_nasabahberpotensi;
  end;
```

4.1.5. Implementasi *Form*

Implementasi *form* dilakukan untuk mengetahui setiap halaman program yang dibuat dan pengkodeannya dalam bentuk *file program*. Berikut ini adalah implementasi antarmuka yang dibuat.

Tabel 4.1 Implementasi *form*

Menu	Deskripsi	Nama File
Login User	Digunakan untuk proses autentikasi user	USplashScreen.pas ULoginForm.pas ULogin.pas
Main	Menampilkan halaman utama aplikasi	UMain.pas
ImportData	Digunakan untuk meng-*import* data ke dalam *database* aplikasi.	UImportDataForm.pas UImportData.pas
Parameter	Digunakan untuk	UParameterForm.pas

Tabel 4.1 Implementasi *Form* (Lanjutan)

Pohon Keputusan	Memproses pohon keputusan berdasarkan parameter yang telah ditentukan *user*	UPohonKeputusan.pas UEntropi.pas
Aturan Prediksi	Menampilkan hasil penelusuran dari pohon	UPrediksi.pas

Bersambung,

	keputusan.	
Prediksi Kriteria	Memprediksi kriteria seorang nasabah dengan parameter-parameter tertentu.	UPrediksiKriteriaForm.pas
Prediksi Nasabah	Memprediksi nasabah mana saja yang berpotensi dan tidak berpotensi.	UNasabahForm.pas
Laporan	Menampilkan halaman laporan dimana *user* dapat mencetak laporan	ULaporan.pas
Data Modul	File program yang digunakan untuk mengitegrasikan *component connection, tabel, query, data source,* ke dalam satu *form* yang *invisible.*	UDM.pas
Perihal	Menampilkan perihal mengenai aplikasi	UPerihal.pas

4.2 . Pengujian

Pengujian merupakan metode yang dilakukan untuk menjelaskan mengenai pengoperasian perangkat lunak yang terdiri dari perangkat pengujian, metode pengujian

dan pelaksanaan pengujian. Pengujian yang digunakan untuk menguji sistem ini adalah menggunakan metode pengujian *black box*. Pengujian *black box* berfokus pada persyaratan fungsional perangkat lunak.

4.2.1 Rencana Pengujian

Rencana pengujian yang akan dilakukan adalah dengan menguji sistem secara *alpha* dan *beta*. Pengujian sistem ini menggunakan data angsuran nasabah kredit PD BPR Batujajar selama periode juni 2009. Jumlah data yang akan digunakan sebanyak 250 *records*. Rencana pengujian selengkapnya bisa dilihat pada tabel 4.2

Tabel 4.2 Rencana pengujian aplikasi *data mining*

Item Uji	Detail Pengujian	Jenis Uji
Login	Verifikasi *Password*	Black Box
Import Data	*Import Data*	Black Box
	Hapus Data	Black Box
Parameter	Memilih parameter	Black Box
	Menyimpan pengaturan	Black Box
Pohon Keputusan	Memproses pohon	Black Box

Tabel 4.2 Rencana pengujian aplikasi *data mining* (Lanjutan)

Aturan Prediksi	Membuat daftar aturan prediksi	Black Box
Prediksi Kriteria	Memilih parameter	Black Box

Bersambung,

	Mengulangi pengujian	Black Box
Prediksi Nasabah	Membuat daftar prediksi nasabah	Black Box
Laporan	Memilih jenis laporan	Black Box
Ganti Password	Merubah password	Black Box

4.2.2 Kasus dan Hasil Pengujian *Alpha*

Berikut akan dijelaskan secara rinci mengenai pengujian *alpha* terhadap aplikasi *data mining* yang telah dibangun, adapun untuk setiap pengujiannya dapat dilihat pada tabel 4.3 hingga 4.9

Tabel 4.3 Pengujian *login.*

Kasus dan Hasil Uji (Data Normal)	
Data Masukan	*Password : 123*
Kegiatan	Klik tombol *login*, jika *valid* maka masuk ke halaman utama aplikasi *data mining*
Yang Diharapkan	*Password login* yang diisikan valid.
Hasil Pengamatan	*Password login* yang dimasukkan diterima, masuk ke halaman utama aplikasi *data mining*

Tabel 4.3 Pengujian *login* (Lanjutan)

Kasus dan Hasil Uji (Data Salah)	Bersambung,
Data Masukan	*Password* : administator

Kegiatan	Setelah klik tombol *login* maka muncul pesan kesalahan *login*
Yang Diharapkan	*Password login* salah dan muncul pesan kesalahan
Hasil Pengamatan	Muncul pesan bahwa *password login* salah
Kesimpulan	Diterima

Tabel 4.4 Pengujian *Import Data*

Kasus dan Hasil Uji (Data Normal)	
Data Masukan	Path : H:\10 SAMPLE DATA.xls
Kegiatan	Klik tombol *browse* untuk mencari data, klik *Import Data*
Yang Diharapkan	Data yang dimasukan *user* benar dan ketika *user* mengklik *Import Data* maka data tersimpan dalam *database*
Hasil Pengamatan	Data berhasil tersimpan dalam *database*
Kesimpulan	Diterima
Kasus dan Hasil Uji (Data Kosong)	
Data Masukan	Path : -
Kegiatan	Klik tombol *Import Data*
Yang Diharapkan	User tidak bisa mengklik tombol *Import Data*
Hasil Pengamatan	Tombol *Import Data* dalam keadaan *disabled*
Kesimpulan	Diterima

Ha **Tabel 4.4** Pengujian *Import Data* (Lanjutan)

Kasus dan Hasil Uji (Data Normal)	Bersambung,
Data Masukan	Path : H:\10 SAMPLE DATA.xls

Kegiatan	Klik tombol hapus data
Yang Diharapkan	Data terhapus
Hasil Pengamatan	Data berhasil terhapus
Kesimpulan	Diterima
Kasus dan Hasil Uji (Data Kosong)	
Data Masukan	Path : -
Kegiatan	Klik tombol Hapus Data
Yang Diharapkan	*User* tidak bisa mengklik tombol Hapus Data
Hasil Pengamatan	Tombol Hapus Data dalam keadaan *disabled*
Kesimpulan	Diterima

Tabel 4.5 Pengujian parameter

Kasus dan Hasil Uji (Data Normal)	
Data Masukan	Parameter : - Jenis Kelamin - Umur - Desa - Kecamatan - Status Marital - Nilai Pinjaman - Jumlah Angsuran
Kegiatan	Memilih parameter dengan cara mencentangn

Tabel 4.5 Pengujian parameter (Lanjutan)

Bersambung,

Tabel 4.8 Pengujian mencari parameter (Lanjutan)

Yang Diharapkan	Parameter bisa dipilih
Hasil Pengamatan	Parameter berhasil dipilih
Kesimpulan	Diterima
Kasus dan Hasil Uji (Data Salah)	
Data Masukan	Parameter : - Nama Nasabah - Alamat - Propinsi
Kegiatan	Memilih parameter dengan cara mencentangnya
Yang Diharapkan	Parameter tidak bisa dipilih
Hasil Pengamatan	Muncul pesan bahwa parameter tidak bisa digunakan
Kesimpulan	Diterima
Menyimpan Pengaturan	
Kasus dan Hasil Uji (Data Normal)	
Data Masukan	Parameter : - Jenis Kelamin - Umur - Desa - Kecamatan - Status Marital - Nilai Pinjaman - Jumlah Angsuran
Kegiatan	Klik tombol simpan
Yang Diharapkan	Muncul pesan konfirmasi simpan

Tabel 4.5 Pengujian parameter (Lanjutan)

Bersambung,

Hasil Pengamatan	Muncul pesan apakah pengaturan akan disimpan?
Kesimpulan	Diterima
Kasus dan Hasil Uji (Data Salah)	
Data Masukan	Parameter : -
Kegiatan	Klik tombol simpan
Yang Diharapkan	User tidak bisa mengklik tombol simpan
Hasil Pengamatan	Tombol simpan dalan keadaan *disabled*
Kesimpulan	Diterima

Tabel 4.6 Pengujian pohon keputusan

Kasus dan Hasil Uji (Data Normal)	
Data Masukan	-
Kegiatan	Klik tombol proses untuk memproses pohon keputusan
Yang Diharapkan	Aplikasi membentuk pohon keputusan sampai terbentuk
Hasil Pengamatan	Pohon keputusan terbentuk
Kesimpulan	Diterima

Tabel 4.7 Pengujian aturan prediksi

Kasus dan Hasil Uji (Data Normal)	
I	
Kegiatan	Klik tombol proses untuk memulai membuat daftar aturan prediksi berdasarkan pohon keputusan yang tela[illegible]
Yang Diharapkan	Aplikasi menampilkan daftar aturan prediksi

Tabel 4.7 Pengujian aturan prediksi (Lanjutan)

Bersambung,

Hasil Pengamatan	Daftar aturan prediksi ditampilkan
Kesimpulan	Diterima

Tabel 4.8 Pengujian prediksi (mencari) kriteria

Memilih parameter	
Kasus dan Hasil Uji (Data Normal)	
Data Masukan	Parameter Pekerjaan : Di bidang perdagangan Parameter Desa : Batujajar Barat Jenis Kelamin : Laki-laki
Kegiatan	Klik parameter pertama, kedua, dan seterusnya sampai muncul hasil prediksi
Yang Diharapkan	Hasil prediksi terlihat
Hasil Pengamatan	Hasil prediksi terlihat.
Kesimpulan	Diterima
Mengulangi pengujian	
Kasus dan Hasil Uji (Data Normal)	
Data Masukan	Parameter Pekerjaan : Di bidang jasa Parameter Desa : Galanggang
Kegiatan	Klik tombol ulangi
Yang Diharapkan	Proses pemprediksian diulangi
H	
Kesimpulan	Diterima

Tabel 4.8 Pengujian prediksi kriteria (Lanjutan)

Bersambung,

Tabel 4.9 Pengujian prediksi nasabah

Kasus dan Hasil Uji (Data Normal)	
Data Masukan	-
Kegiatan	Klik tombol Nasabah Berpotensi atau Nasabah Tidak Berpotensi untuk memulai memprediksi nasabah.
Yang Diharapkan	Aplikasi menampilkan daftar prediksi nasabah, baik nasabah yang berpotensi ataupun yang tidak berpotensi
Hasil Pengamatan	Daftar prediksi nasabah ditampilkan
Kesimpulan	Diterima

Tabel 4.10 Pengujian laporan

Memilih jenis laporan	
Kasus dan Hasil Uji (Data Normal)	
Data Masukan	-
Kegiatan	Klik *radio button* Laporan Prediksi Kriteria Nasabah ataupun Laporan Prediksi Nasabah, baik yang berpotensi ataupun yang tidak berpotensi, kemudian klik tombol Pilih.
Yang Diharapkan	Laporan prediksi ditampilkan
Hasil	Laporan ditampilkan
Kesimpulan	Diterima

4.2.3 Kesimpulan Hasil Pengujian *Alpha*

Berdasarkan hasil pengujian dengan kasus uji sample diatas dapat ditarik kesimpulan bahwa perangkat lunak telah bebas dari kesalahan sintaks dan secara fungsional menghasilkan hasil yang sesuai dengan yang diharapkan.

4.2.4 Kasus dan Hasil Pengujian *Beta*

Pengujian *beta* merupakan pengujian langsung kepada *user* atau pengguna untuk mencoba aplikasi yang baru dengan mengisi kuesioner mengenai kepuasan *user*. Dari kuesioner tersebut akan dilakukan perhitungan untuk dapat mengambil kesimpulan terhadap penilaian dari aplikasi *data mining* yang telah dibangun. (Lembaran kuesioner dilampirkan). Penulis melakukan pengujian beta terhadap seorang karyawan PD BPR Kabupaten Bandung Cabang Batujajar yang menjabat sebagai Bagian Dana, dimana aplikasi *data mining* ini dirancang untuk digunakan oleh *user* atau pengguna di Bagian Dana tersebut. Berikut pertanyaan-pertanyaan yang disampaikan kepada para responden,

1. Apakah anda setuju bahwa tampilan aplikasi yang dibangun menarik?
2. Apakah anda setuju bahwa aplikasi yang dibangun mudah digunakan?
3. Apakah anda setuju petunjuk yang disediakan aplikasi yang dibangun mudah dipelajari?
4. Apakah anda setuju aplikasi yang dibangun dapat membantu dalam analisis data nasabah?
5. Apakah anda setuju informasi yang dihasilkan aplikasi yang dibangun mudah dipahami?

6. Apakah anda setuju hasil prediksi dari aplikasi yang dibangun akurat dalam memprediksi kriteria nasabah?
7. Apakah anda setuju aplikasi yang dibangun dapat membantu dalam pengambilan keputusan di bidang marketing?

Tabel 4.11 Hasil kuesioner.

Pertanyaan	Jumlah Responden	SS	S	CS	BS	KS	TS	STS
1	1	1	0	0	0	0	0	0
2	1	0	1	0	0	0	0	0
3	1	0	1	0	0	0	0	0
4	1	0	1	0	0	0	0	0
5	1	0	1	0	0	0	0	0
6	1	1	0	0	0	0	0	0
7	1	0	1	0	0	0	0	0

Ket :

SS = Sangat Setuju
S = Setuju
CS = Cukup Setuju
BS = Biasa Saja
KS = Kurang Setuju
TS = Tidak Setuju
STS = Sangat Tidak Setuju

Berdasarkan data hasil kuesioner, dapat dicari persentase masing-masing jawaban dengan menggunakan rumus: Y = P/Q *100%

Keterangan :

P = Banyaknya jawaban responden tiap soal

Q = Jumlah responden

Y = Nilai Persentase

Tabel 4.12 Persentase hasil kuesioner.

Pertanyaan	Jumlah Responden	SS	S	CS	BS	KS	TS	STS
1	1	100%	0%	0%	0%	0%	0%	0%
2	1	0%	100%	0%	0%	0%	0%	0%
3	1	0%	100%	0%	0%	0%	0%	0%
4	1	0%	100%	0%	0%	0%	0%	0%
5	1	0%	100%	0%	0%	0%	0%	0%
6	1	100%	0%	0%	0%	0%	0%	0%
7	1	0%	100%	0%	0%	0%	0%	0%

Ket :

SS = Sangat Setuju

S = Setuju

CS = Cukup Setuju

BS = Biasa Saja

KS = Kurang Setuju

TS = Tidak Setuju

STS = Sangat Tidak Setuju

4.2.5 Kesimpulan Hasil Pengujian *Beta*

Dari pengujian *beta* yang telah dilakukan yaitu dengan pengujian perhitungan pilihan kategori jawaban dari kuesioner yang telah dibagikan dapat disimpulkan sebagai berikut,

1. Bagian Dana mengatakan sangat setuju tampilan aplikasi ini menarik.
2. Bagian Dana mengatakan setuju bahwa aplikasi ini mudah untuk digunakan
3. Bagian Dana mengatakan setuju bahwa petunjuk yang disediakan aplikasi ini mudah untuk dipelajari
4. Bagian Dana mengatakan setuju aplikasi ini dapat membantu dalam analisis data nasabah,
5. Bagian Dana mengatakan setuju bahwa informasi yang dihasilkan aplikasi ini mudah untuk dipahami
6. Bagian Dana mengatakan sangat setuju bahwa hasil prediksi dari aplikasi yang dibangun akurat dalam pemprediksian kriteria nasabah,
7. Bagian Dana mengatakan setuju bahwa aplikasi yang dibangun dapat membantu dalam pengambilan keputusan di bidang marketing