BAB 4
IMPLEMENTASI DAN PENGUJIAN SISTEM

4.1 Implementasi Sistem

Tahap implementasi merupakan tahapan penerapan dari tahap perancangan. Pada tahap ini dipaparkan semua elemen-elemen yang terlibat pada pembangunan sistem *data mining* pada data siswa di SMA Pasundan 2 Bandung.

4.1.1 Lingkungan Implementasi

Lingkungan implementasi merupakan kebutuhan lingkungan baik secara perangkat lunak maupun perangkat keras minimal yang direkomendasikan untuk menjalankan atau mengakses sistem

1) Lingkungan Perangkat Keras

Berikut ini adalah spesifikasi perangkat keras minimal untuk dapat menjalankan sistem, yang dapat dilihat pada Tabel 4.1.

Tabel 4.1 Lingkungan Perangkat Keras

No	Jenis	Komponen
1	Processor	Dual Core @ 2.1 Ghz
2	Memory	DDR 3 1 GB
3	Hardisk	250 GB
4	Monitor	Resolusi 1024 x 768 pixel

2) Lingkungan Perangkat Lunak

Berikut ini adalah spesifikasi perangkat lunak minimal untuk menjalanakan sistem, yang dapat dilihat pada Tabel 4.2.

Tabel 4.2 Lingkungan Perangkat Lunak

No	Komponen
1	Sistem Operasi Windows 7
2	Web Browser Firefox 20 keatas Google Chrome 30 keatas

3) Lingkungan *Server*

Berikut ini adalah spesifikasi *server* dimana sistem akan dijalankan yang dapat dilihat pada Tabel 4.3.

Tabel 4.3 Lingkungan Server

No	Jenis	Komponen
1	Sistem Operasi	Linux Based OS
2	Web Server	Apache 2.4
3	PHP	PHP 5.5
4	*Database* Server	Mysql 5.6
5	Processor	4*Core*
6	Memory	4GB

4.1.2 Implementasi Data

Basis data yang digunakan dalam sistem ini menggunkan MySQL server. Berikut ini adalah implementasi yang dilakukan dalam sistem.

1. Query Pembuatan Basis Data

```
CREATE DATABASE kbb;
```

2. Query Pembuatan Tabel kelas

```
CREATE TABLE IF NOT EXISTS `kelas` (
  `id_kelas` int(10) NOT NULL AUTO_INCREMENT PRIMARY KEY,
  `nama_kelas` varchar(50) NOT NULL );
```

3. Query Pembuatan Tabel kelas_ips

```
CREATE TABLE IF NOT EXISTS `kelas_ips` (
  `id` int(11) NOT NULL AUTO_INCREMENT  PRIMARY KEY,
  `id_kelas` int(10) NOT NULL FOREIGN KEY,
  `nomor urut` varchar(20) NOT NULL,
  `nis` varchar(9) NOT NULL,
  `nama_siswa` int(100) NOT NULL,
  `jenis_kelamin` int(2) NOT NULL,
  `semester` int(2) NOT NULL ),
```

```
‘bahasa_indonesia’ decimal(10,2) NOT NULL,
‘bahasa_inggris’ decimal(10,2) NOT NULL,
‘matematika’ decimal(10,2) NOT NULL,
‘geografi’ decimal(10,2) NOT NULL,
‘sejarah’ decimal(10,2) NOT NULL,
‘sosiologi’ decimal(10,2) NOT NULL,
‘rata - rata’ decimal(10,2) NOT NULL,
```

4. Query Pembuatan Tabel kelas_ipa

```
CREATE TABLE IF NOT EXISTS `kelas_ipa` (
`id` int(11) NOT NULL AUTO_INCREMENT  PRIMARY KEY,
`id_kelas` int(10) NOT NULL FOREIGN KEY,
`nomor urut` varchar(20) NOT NULL,
`nis` varchar(9) NOT NULL,
`nama_siswa` int(100) NOT NULL,
`jenis_kelamin` int(2) NOT NULL,
`semester` int(2) NOT NULL ),
‘bahasa_indonesia’ decimal(10,2) NOT NULL,
‘bahasa_inggris’ decimal(10,2) NOT NULL,
‘matematika’ decimal(10,2) NOT NULL,
‘biologi’ decimal(10,2) NOT NULL,
‘fisika’ decimal(10,2) NOT NULL,
‘kimia’ decimal(10,2) NOT NULL,
‘rata - rata’ decimal(10,2) NOT NULL,
```

4.1.3 Implementasi Antarmuka

Implementasi antarmuka dilakukan untuk mengetahui setiap tampilan yang dibangun dan pengkodeannya dalam bentuk file program. Implementasi antarmuka dijelaskan pada tabel Tabel 4.4

Tabel 4.4 Implementasi Antarmuka

No	Nama Antarmuka	Nama File	Keterangan
1	Menu Utama	frmPilihanJurusan.cs	Tampilan beranda sistem
2	*Import* Data	ucrDataSet.cs	Digunakakan untuk menginput data yang dilakukan oleh user
3	Atribut *Selection*	ucr*Selection*.cs	Antarmuka pemilihan atribut
4	*Cleaning*	ucr*Preprocessing*.cs	Antarmuka proses *cleaning*
5	Transformasi Data	ucrTransform.cs	Antarmuka proses transformasi data pada setiap atribut
6	Pembagian Kelas	ucrPembagianKelas.cs	Antarmuka hasil / *output* dari *data mining*

4.2 Pengujian Sistem

Pengujian sistem merupakan serangkaian tahapan untuk menguji ketangguhan dari sistem yang telah dibangun dengan tujuan untuk menemukan kesalahan atau kekurangan pada sistem dan apakah sistem yang dibangun telah sesuai dengan perancangan di awal.

4.2.1 Rencana Pengujian

Rencana pengujian sistem merupakan strategi dokumentasi yang digunakan untuk memverifikasi dan memastikan bahwa sistem atau produk yang telah dibangun menemui desain dan persyaratan yang telah dirancang sebelumnya. Pada rencana pengujian sistem terdapat dua tahapan rencana pengujian yaitu rencana pengujian *alpha*, dan pengujian *beta*.

1. Rencana Pengujian *Alpha*

Rencana pengujian fungsionalitas merupakan suatu proses pengujian fungsionalitas sistem yang menggunakan metode *alpha*, dimana pengujian tersebut berfokus pada *output* yang dihasilkan dalam menanggapi *input* dan kondisi eksekusi yang dipilih. Rencana pengujian *Alpha* itu sendiri dapat dilihat pada Tabel 4.5

Tabel 4.5 Rencana Pengujian *Alpha*

Item Uji	Detail Pengujian	Jenis Ujian
Pemilihan File	1. Pilih file yang akan digunakan	*Black box*
Penyimpanan Data	1. Simpan data ke *database*	*Black box*
Pemilihan Atribut	1. Pilih atribut yang akan digunakan 2. Pilih atribut tujuan	*Black box*
Output Kelas	1. Pembentukan daftar siswa yang sudah dikelompokan berdasarkan kelasnya	*Black box*

2. Rencana Pengujian *Beta*

Rencana pengujian *beta* yang akan dilakukan adalah dengan melakukan wawancara kepada pengguna sistem yaitu manajer pemasaran. Wawancara ini terdiri dari beberapa pertanyaan yang mendukung tujuan akhir aplikasi yang dibangun.

3. Rencana Pengujian Validitas Algoritma

Memeriksa apakah implementasi akgoritma *ID3* yang dipakai pada *data mining* klasifikasi ini berfungsi dengan benar dan valid dengan membandingankan hasil keluaran algoritma *ID3* pada aplikasi WEKA.

4.2.2 Pengujian *Alpha*

Pengujian alpha dilakukan menggunakan metode *black box(equivalen partioning).* Untuk menentukan pengujian alpha. Seperti yang terlihat dibawah ini.

1. Pemilihan File

Tabel 4.6 pengujian pemilihan file transaksi merupakan *scenario* pengujian fungsionalitas yang dilakukan pada bagian pemilihan file transaksi.

Tabel 4.6 Pengujian Pemilihan File Transaksi

Kasus dan hasil uji (data normal)			
Data masukan	**Yang diharapkan**	**Pengamatan**	Kesimpulan
File : Kelas IPA.xlsx	Dapat menampilkan file yang sudah dibuka untuk disimpan ke database.	Dapat menampilkan file yang sudah dibuka untuk disimpan ke database	[√] Diterima [ ] Ditolak
Kasus dan hasil uji (data salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	Kesimpulan
File : Data-mining-klasifikasi.pdf	Muncul pesan kesalahan agar memilih file berformat excel	Menampilkan pesan kesalahan pilih *file* berformat excel.	[√] Diterima [ ] Ditolak
File : Kelas.xls	Data yang dimasukan tampil tapi ditidak dapat disimpan	Data yang dimasukan tampil tapi ditidak dapat disimpan	[√] Diterima [ ] Ditolak

2. Pemilihan File Penyimpanan data ke database

Tabel 4.7 pengujian penyimpanan kedalam database merupakan *scenario* pengujian yang dilakukan pada bagian penyimpanan data ke dalam *database*

Tabel 4.7 Pengujian Penyimpanan Data Kedalam *Database*

Kasus dan hasil uji (data normal)			
Data masukan	**Yang diharapkan**	**Pengamatan**	Kesimpulan
File : Kelas IPA.xlsx	Dapat menyimpan data f*ile* excel ke dalam database dan menampilkan data excel yang telah disimpan.	Penyimpanan ke dalam database berhasil dan dapat menampilkan data excel yang telah disimpan	[√] Diterima [ ] Ditolak
Kasus dan hasil uji (data salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	Kesimpulan
File : Data-mining-klasifikasi.pdf	Muncul pesan kesalahan agar memilih file	Menampilkan pesan kesalahan pilih *file* berformat excel dan	[√] Diterima [ ] Ditolak

	berformat excel dan tidak dapat disimpan	tidak dapat disimpan	
File : Kelas.xls	Muncul pesan kesalahan bahwa data yang di masukan tidak dapat disimpan ke database.	Muncul pesan kesalahan bahwa data yang di masukan tidak dapat disimpan ke database.	[√] Diterima [ ] Ditolak

3. Proses Atribut *Selection*

Scenario pengujian untuk menentukan atribut dapat dilihat pada Tabel 4.8 berikut.

Tabel 4.8 Pengujian Penyimpanan Data Kedalam *Database*

Kasus dan hasil uji (data normal)			
Data masukan	**Yang diharapkan**	**Pengamatan**	Kesimpulan
Atribut data: `id`, `id_kelas`, `nomor urut`, `nis`, `nama_siswa`, 'jenis_kelamin', `jenis_kelamin`, `semester`, 'pendidikan_agama', 'PKNS', ‘bahasa_indonesia’, 'penjaskes', ‘bahasa_inggris’, ‘matematika’, 'prakarya',	Menampilkan atribut hasil seleksi `id`, `id_kelas`, `nomor urut`, `nis`, `nama_siswa`, `jenis_kelamin`, `semester`, ‘bahasa_indonesia’, ‘bahasa_inggris’, ‘matematika’, ‘geografi’, ‘sejarah’, ‘sosiologi’, ‘rata - rata’	Menampilkan atribut hasil seleksi `id`, `id_kelas`, `nomor urut`, `nis`, `nama_siswa`, `jenis_kelamin`, `semester`, ‘bahasa_indonesia’, ‘bahasa_inggris’, ‘matematika’, ‘geografi’, ‘sejarah’, ‘sosiologi’, ‘rata - rata’	[√] Diterima [ ] Ditolak

'kewirausahaan', 'etika_profesi', ‘geografi’, ‘sejarah’, ‘sosiologi’, 'multimedia', ‘rata - rata’			
Kasus dan hasil uji (data salah)			
Data masukan	**Yang diharapkan**	**Pengamatan**	Kesimpulan
Id= kosong Id_kelas= kosong Nama_siswa= kosong	Menanpilkan hasil seleksi atribut dengan hasil kosong	Menanpilkan hasil seleksi atribut dengan hasil kosong	[ ] Diterima [√] Ditolak

4. Output Kelas

Scenario pengujian untuk menentukan atribut dapat dilihat pada Tabel 4.9 berikut.

Tabel 4.9 Pengujian Output Kelas

Kasus dan hasil uji (data normal)			
Data masukan	**Yang diharapkan**	**Pengamatan**	Kesimpulan
Data masukan berasal dari hasil proses decision tree	Menampilkan daftar pola aturan mengenai klasifikasi siswa untuk masuk kelas “Intensif” atau “Tidak Intensif”	Dapat menampilkan daftar pola aturan klasifikasi, sesuai yang diharapkan.	[√] Diterima [ ] Ditolak

Hasil pengujian *black box* menyatakan bahwa aplikasi yang dibangun belum bebas dari kesalahan prosedur internal program dan secara fungsional mengeluarkan hasil yang sesuai dengan yang diharapkan.

4.2.3 Kesimpulan Pengujian *Alpha*

Berdasarkan hasil pengujian sistem yang telah dilakukan secara keseluruhan, dapat ditarik kesimpulan bahwa proses yang terjadi dalam sistem telah mengalami tahap perbaikan, dan sudah memaksimalkan proses–proses tersebut. Secara fungsional sistem sudah dapat digunakan, dan menghasilkan keluaran yang sesuai dengan yang diharapkan.

4.2.4 Pengujian *Beta*

Pengujian beta dilakukan pada satu *user* yang merupakan pemakai akhir perangkat lunak yang dibangun. Dalam pengujian beta, dilakukan penelitian secara langsung terhadap pengguna sistem dengan menggunakan wawancara kepada bagian kesiswaan di sekolah terhadap sistem yang telah dibangun.

Wawancara dilakukan kepada bagian kesiswaan di SMA Pasundan 2 Bandung. Berikut adalah hasil wawancara dapat dilihat sebagai berikut :

a Apakah tampilan (Antarmuka) aplikasi terlihat menarik?
  Jawaban: “Interface menarik, pemilihan warna cukup baik. Penggunaan icon juga cukup proporsional denga ukuran huruf.”

b Apakah informasi pengklasifikasian yang dihasilkan data mining mudah dimengerti?
  Jawaban: “Informasi pengklasifikasian data mudah dipahami”

c Apakah fungsionalitas aplikasi ini memenuhi kebutuhan?
  Jawaban: “sudah cukup memenuhi kebutuhan”

d Apakah sistem yang dibangun dapat membantu anda untuk pengelompokan kelas siswa ?
  Jawaban : “ya, sistem ini dapat membantu saya untuk pengelompokan siswa”.

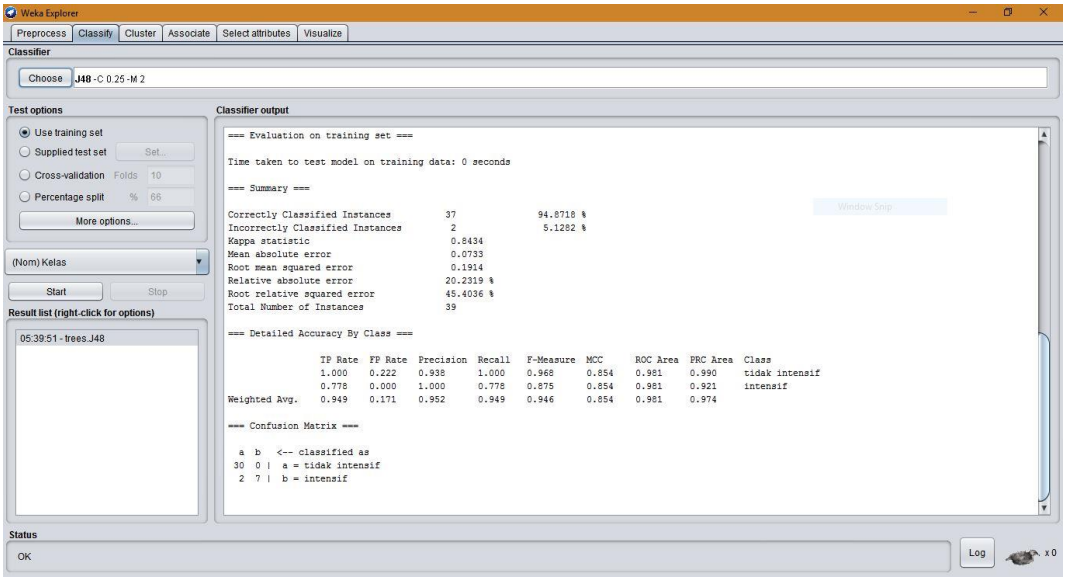
e Apakah sistem yang dibangun mudah digunakan ?
  Jawaban : “ya, aplikasinya mudah digunakan, tidak memerlukan keahlian khusus untuk menggunakan aplikasi ini”.

4.2.5 Kesimpulan Pengujian *Beta*

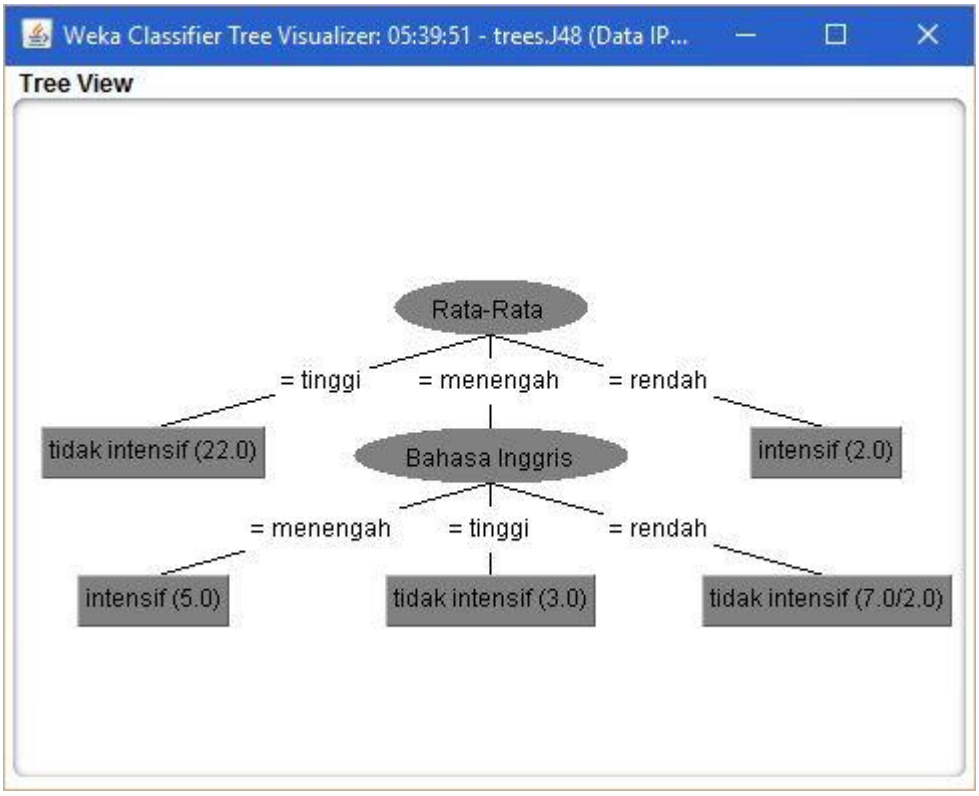
Berdasarkan hasil dari pengujian beta, dapat disimpulkan bahwa sistem yang dibangun ini dapat membantu bagian kesiswaan di SMA Pasundan 2 Bandung untuk mengolah data siswa pada pengelompokan kelas bimbingan belajar.

4.2.6 Pengujian Validitas Algoritma

Dalam pengujian validitas algoritma pada aplikasi WEKA digunakan data training siswa yang terdapat pada **Error! Reference source not found.**. Berikut pengujian dan *trees* yang dihasilkan WEKA dapat dilihat pada gambar



Gambar 4.1 Hasil Pengujian Aplikasi WEKA



Gambar 4.2 Visualisasi *Tree* Pada Apliaksi Weka

4.2.7 Kesimpulan Pengujian Validitas Algoritma

Dari Gambar 4.1 Hasil Pengujian Aplikasi WEKA menampilkan hasil dari metode klasifikasi dengan algoritma algoritma *ID3* dengan tingkat keakuratan sebesar 94,818 %. Dengan detail sebagai berikut: data yang tidak terklasifikasi sebanyak 2 data dari 39 data (5,1282% *error*). Dan untuk visualisasi *trees* tanpa menunjukan kesalahan atau *error*.