

BAB III

METODOLOGI PENELITIAN

2.1. Tujuan Operasional Penelitian

Tujuan dari penelitian ini adalah untuk mengembangkan aplikasi sistem pendukung keputusan dalam memilih peminatan di Program Studi Pendidikan Teknik Informatika dan Komputer yang dapat membantu mahasiswa Program Studi Pendidikan Teknik Informatika dan Komputer mengambil keputusan terkait pemilihan peminatan yang akan mereka ambil.

2.2. Tempat dan Waktu Penelitian

Penelitian ini dilakukan di Jurusan Teknik Elektro Universitas Negeri Jakarta sejak bulan Maret 2015 hingga Juli 2015.

2.3. Metode Penelitian

Untuk menyelesaikan permasalahan yang ada, diperlukan strategi pengembangan yang terdiri dari proses, metode dan alat perancangan. Oleh karena itu dalam pembuatan aplikasi sistem pendukung keputusan ini menggunakan metode pengembangan sistem *waterfall* untuk pengembangan aplikasi dan metode eksperimen untuk penelitian.

2.4. Instrumen Penelitian

Perangkat yang digunakan selama penelitian adalah sebagai berikut:

- a. Perangkat Keras
 1. *Netbook*
 2. Processor Intel(R) Atom(TM) N550 1.5 GHz.
 3. Memori 2GB

4. *Harddisk*160 GB
 5. Monitor 10"
- b. Perangkat Lunak
1. Sistem Operasi Windows 7 (Ultimate)
 2. Xampp for Windows x86 3.2.1
 3. Weka Environment for Knowledge Analysis V. 3.7.12
 4. Microsoft Windows Excel 2007
 5. Sublime Text Editor
 6. PHP 5.4.7
 7. MySQL 5.5.27
- c. Data Penelitian
1. Data Nilai Mahasiswa Program Studi Pendidikan Teknik Informatika dan Komputer dari 2010 – 2012 yang berhubungan dengan pilihan peminatan.

2.5. Pelaksanaan Penelitian

2.5.1. Proses Pengumpulan Data Penelitian

3.5.1.1. Analisis Data

Tidak semua data nilai mahasiswa PTIK 2010-2012 yang akan digunakan dalam proses data mining ini. Kriteria – kriteria yang digunakan untuk memilih mata kuliah yang akan dijadikan sebagai atribut penelitian adalah sebagai berikut:

- Mata kuliah terdaftar antara semester 1 sampai dengan semester 4 pada perkuliahan Program Studi Pendidikan Teknik Informatika dan Komputer Universitas Negeri Jakarta di angkatan 2010-2012

- Mata kuliah yang digunakan adalah mata kuliah yang ada pada setiap angkatan, antara angkatan 2010-2012.
- Mata kuliah memiliki hubungan keilmuan dengan mata kuliah peminatan sesuai dengan peminatan yang ada di Program Studi Pendidikan Teknik Informatika dan Komputer Universitas Negeri Jakarta.
- Mata kuliah yang di pilih sebagai atribut adalah mata kuliah wajib.

.Berdasarkan kriteria, data yang akan digunakan dalam proses mining antara lain :

1. Algoritma Pemrograman 1
2. Algoritma Pemrograman 2
3. Sistem Multimedia
4. Desain Web
5. Jaringan Komputer
6. Komunikasi Data

Data training yang digunakan pada proses data mining di dapatkan dari PUSKOM Universitas Negeri Jakarta.

3.5.1.2. Analisis Model

Pembuatan model klasifikasi algoritma C4.5 membutuhkan data atribut dan data nilai dari atribut tersebut. Data – data tersebut nantinya akan dimasukkan dalam tabel data. Untuk melakukan klasifikasi, umumnya data atribut yang akan dicari klasifikasinya di letakkan di kolom akhir dari tabel. Proses klasifikasi algoritma C4.5 ini menggunakan aplikasi WEKA dibawah lisensi GPL (General Public License).

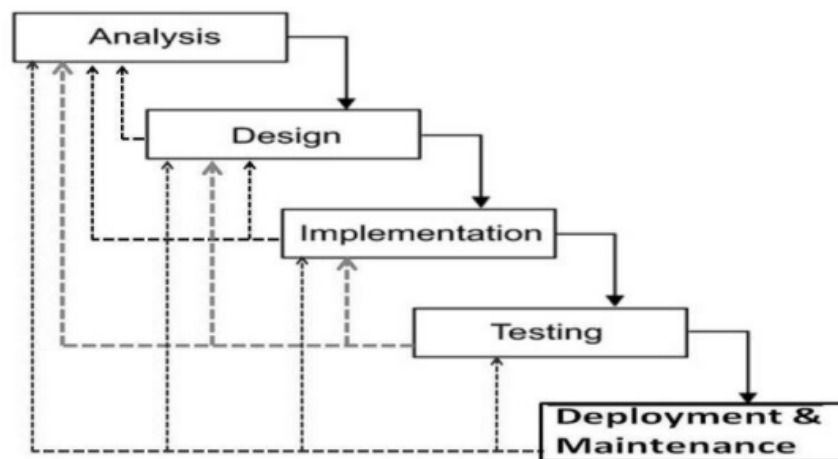
WEKA merupakan API Java yang menyediakan API untuk pengolahan data dalam data mining yang ber- *engine* JAVA. WEKA dikembangkan oleh

Universitas Waikato di Selandia Baru. WEKA menyediakan penggunaan teknik klasifikasi menggunakan pohon keputusan dengan algoritma J48. Teknik yang digunakan WEKA adalah classifier.

Data akademik yang telah di kumpulkan selanjutnya akan dibuat tabel 7 kolom dengan data nilai yang sesuai dengan atributnya masing-masing. Data atribut dan data nilai yang telah dibuat tabel harus disimpan dalam file berformat .arff atau .csv agar dapat di kenali oleh sistem pada WEKA. Data penelitian tersebut dilampirkan pada bagian lampiran.

2.5.2. Proses Pengembangan Aplikasi

Tahapan – tahapan yang dilakukan dalam pengembangan aplikasi menggunakan metode waterfall yang dapat di ilustrasikan pada Gambar 3.1.



Gambar 3.1 Model Waterfall

Pada Tabel 3.1 adalah penjelasan dari masing – masing tahapan *waterfall* yang terdapat pada Gambar 3.1.

Tabel 3.1 Tahap Pengembangan Model Waterfall

No	Tahapan Waterfall	Proses yang dilakukan
1	<i>Analysis</i>	Menganalisis masalah yang akan dihadapi dan mencari solusi dengan cara menguraikan gambaran sistem dan membuat analisis kebutuhan dari sistem tersebut
2	<i>Design</i>	Pembuatan <i>Data Flow Diagram</i> (DFD), dan pembuatan desain tampilan sistem atau <i>User Interface</i>
3	<i>Implementation</i>	Menerapkan Tahapan <i>design</i> kedalam kode – kode program yang akan di eksekusi dan menghasilkan sebuah sistem atau aplikasi baru
4	<i>Testing</i>	Menguji sistem yang telah dibuat pada tahapan <i>implementation</i> .
5	<i>Deployment & Maintenance</i>	Mengoperasikan sistem dan melakukan <i>maintenance</i> .

2.5.3. Analisis Kebutuhan

Pada tahap analisis kebutuhan aplikasi sistem pendukung keputusan bertujuan untuk menjawab kebutuhan aplikasi yang dapat membantu *user* atau dalam hal ini mahasiswa untuk memilih peminatan yang sesuai dengan nilai akademik mahasiswa tersebut.

Kebutuhan fungsional harus mendefinisikan aksi dasar yang diambil oleh sistem untuk menerima dan memproses *input* yang berupa informasi yang berhubungan dengan data – data nilai mahasiswa yang akan di olah sistem. Keluaran yang dihasilkan adalah hasil pengolahan data yang dilakukan sistem

berupa prediksi dan saran peminatan yang hendaknya di ambil oleh mahasiswa tersebut.

Tabel 3.2 Daftar Kebutuhan Fungsional

Aspek	Kebutuhan
<i>User</i>	<i>User</i> dapat melakukan prediksi untuk pemilihan peminatan dengan memasukkan data – data nilai akademik miliknya yang sesuai dengan form pengisian
Content	Suatu sistem yang dapat mengolah data – data yang di input oleh <i>user</i> sehingga menghasilkan sebuah hasil prediksi yang dapat menjadi saran bagi <i>user</i> dalam menentukan pilihan peminatan mereka
Admin	Admin yang dapat menginput data dan mengelola record yang di input oleh <i>user</i>

2.5.4. Desain dan Analisis Sistem

Pada tahap desain dan analisis sistem bertujuan untuk menentukan langkah – langkah detail dalam pembuatan aplikasi sistem pendukung keputusan, untuk itu langkah – langkah yang diperlukan adalah sebagai berikut :

1. Mempelajari aturan, fitur dan format yang diperlukan dalam penyusunan dan pembuatan sistem yang akan dibuat.
2. Menganalisa hambatan yang akan muncul dalam proses pembuatan sistem.
3. Menentukan desain input dan output yang akan dihasilkan dalam sistem secara keseluruhan agar lebih mudah mengevaluasi komponen – komponen yang bermasalah dalam sistem.

Berdasarkan poin – poin tersebut berguna untuk proses penerapan agar tujuan dari penerapan sistem tercapai yang dapat terangkum dalam prosedur sistem.

2.5.4.1. Prosedur Sistem

Prosedur sistem dari aplikasi sistem pendukung keputusan ini adalah sebagai berikut :

1. Mahasiswa (*User*) akan mendapatkan info tentang peminatan yang ada di Pendidikan Teknik Informatika dan Komputer, Universitas Negeri Jakarta dan mendapatkan informasi tentang panduan penggunaan aplikasi
2. Mahasiswa (*User*) dapat memasukkan data yang dibutuhkan sesuai dengan form yang disediakan untuk melakukan prediksi pemilihan peminatan yang kemungkinan dapat dia ambil berdasarkan data nilai mereka.
3. Semua data informasi yang terdapat dalam aplikasi dimasukkan oleh seorang admin.
4. Selain memasukkan data yang berkaitan dengan kegiatan prediksi pemilihan peminatan, *user* tidak dapat mengubah data apapun.
5. Admin dapat mengubah beberapa data seperti informasi yang ditampilkan pada aplikasi dan tidak bisa mengubah data hasil prediksi pemilihan peminatan yang dilakukan *user* kecuali menghapus data.

2.5.4.2. Kriteria Sistem

Kriteria yang diajukan dalam pembuatan aplikasi sistem pendukung keputusan berdasarkan analisis yaitu:

1. *User* dapat menerima informasi berkaitan dengan peminatan yang akan mereka ambil, *user* juga dapat menerima informasi tentang cara penggunaan aplikasi sistem pendukung keputusan.

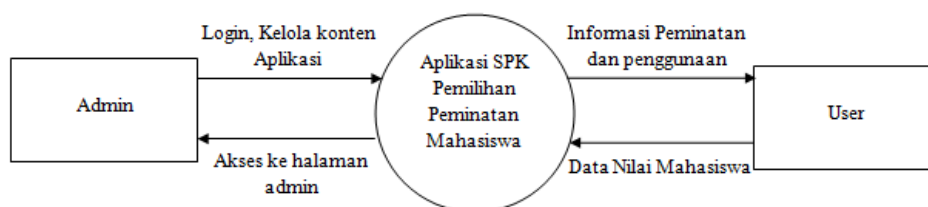
2. *User* dapat melakukan kegiatan prediksi pemilihan peminatan pada aplikasi ini tidak dibatasi limit penggunaan.
3. Admin dapat melihat data hasil prediksi yang dilakukan *user* dan diberikan akses untuk menghapus yang bertujuan untuk pemeliharaan data, dan tidak dapat menambahkan data maupun mengubah isi dari data tersebut.
4. Admin dapat menambahkan, mengubah, menghapus data informasi yang akan ditampilkan pada halaman aplikasi sistem pendukung keputusan.

2.5.4.3. DFD

Data Flow Diagram (DFD) adalah proses model yang populer untuk menggambarkan suatu sistem yang akan dikembangkan.

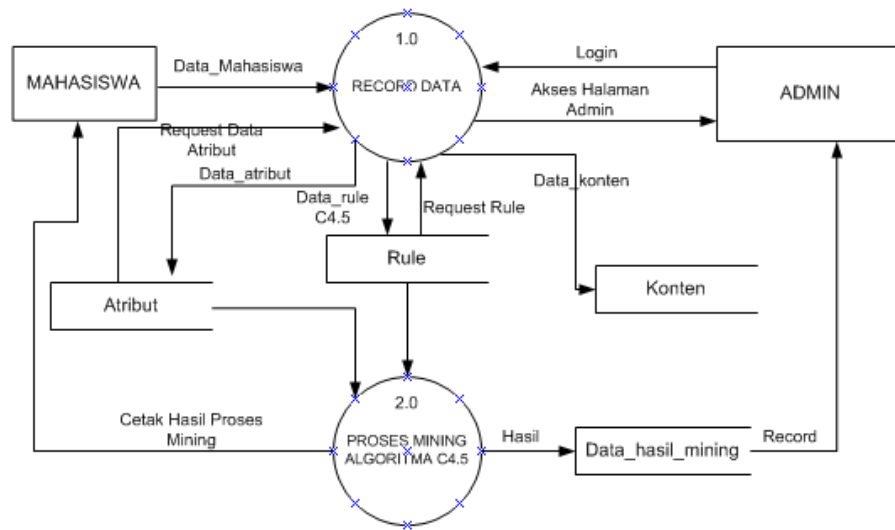
2.5.4.3.1. Diagram Konteks

Diagram konteks dibuat untuk menggambarkan secara umum dari keseluruhan sistem yang ada. Dengan kata lain diagram konteks ini sesuai dengan namanya level 0 dirancang untuk menggambarkan sumber dan tujuan data yang akan di proses. Berikut ini adalah diagram konteks dari Aplikasi Sistem Pendukung Keputusan Pemilihan Peminatan Mahasiswa Pendidikan Teknik Informatika dan Komputer Universitas Negeri Jakarta dapat digambarkan sebagai berikut :



Gambar 3.2 Diagram Konteks Aplikasi SPK

2.5.4.3.2. Diagram Nol Sistem



Gambar 3.3 Diagram Nol Sistem

Keterangan :

1. Proses 1.0

Pada proses 1.0 pengolahan data meliputi data nilai mahasiswa yang di masukkan *user* dan memasukkan data dari sisi admin seperti konten, *rule* Algoritma C4.5 dan disimpan di dalam *database*.

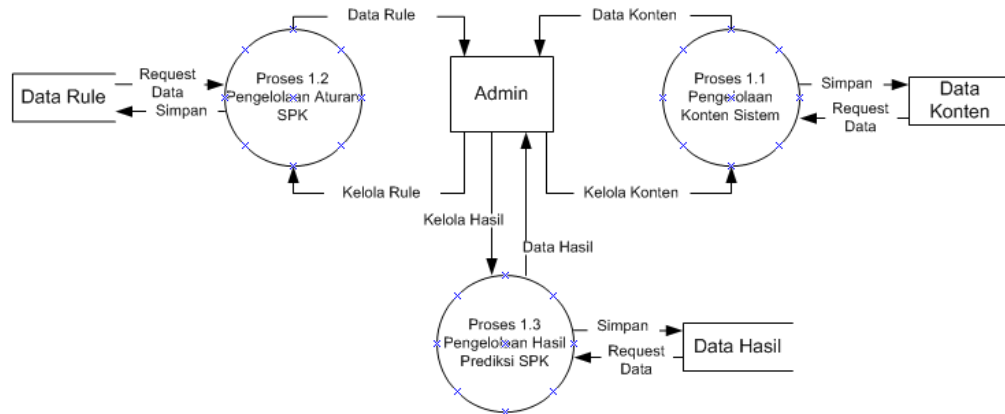
2. Proses 2.0

Pada proses 2.0 adalah pengolahan data untuk prediksi SPK. Data nilai yang dimasukkan *user* akan diolah dengan data *rule data mining* yang di masukkan oleh *admin* dan menghasilkan sebuah prediksi yang akan di simpan ke *database* dan dapat di cetak oleh *user*,

2.5.4.3.3. Diagram Proses 1 Level 1

Pada diagram proses 1 level 1 menggambarkan bahwa *admin* dapat melakukan entri data mengenai konten sistem dan dapat mengentri aturan yang telah di dapat

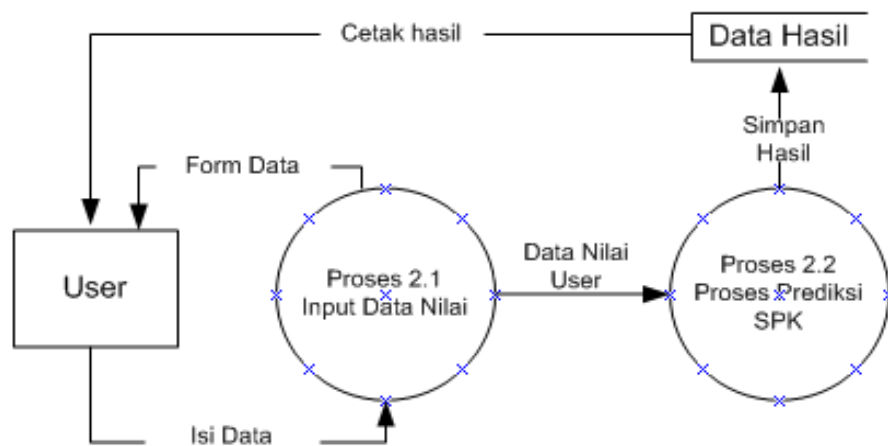
dari pengolahan *data mining*. Diagram proses 1 level 1 sistem ditujukan pada gambar sebagai berikut:



Gambar 3.4 Diagram Proses 1 Level 1

2.5.4.3.4. Diagram Proses 2 Level 1

Pada diagram proses 2 level 1 menggambarkan alur *user* melakukan proses prediksi SPK. Diagram proses 2 level 1 sistem ditujukan pada gambar berikut:



Gambar 3.5 Diagram Proses 2 Level 1

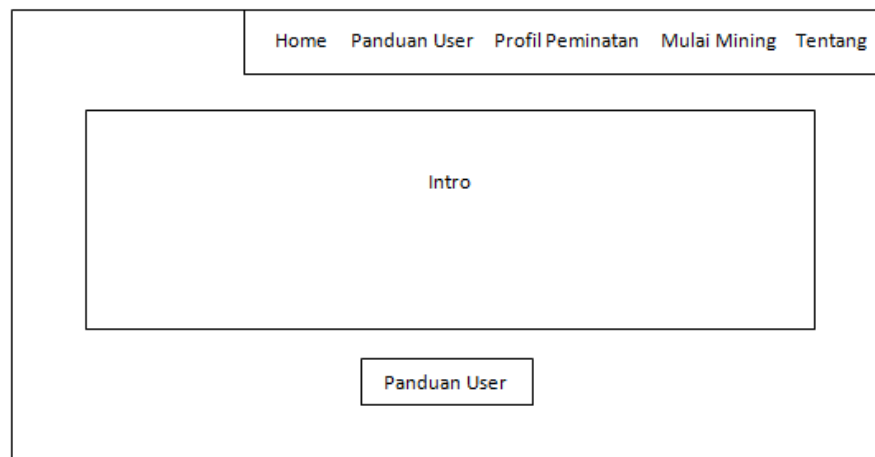
2.5.4.4. Pembuatan Desain Tampilan Halaman Sistem

Hal yang dibutuhkan dalam tampilan adalah desain yang sederhana dan nyaman bagi *user*. Sehingga *user* tidak kesulitan pada saat menggunakannya. Berikut ini adalah rancangan tampilan:

1. Rancang tampilan *home user*

Pada tampilan *home*, *user* mendapatkan tampilan sederhana dengan beberapa teks untuk memperkenalkan aplikasi yang akan *user* gunakan.

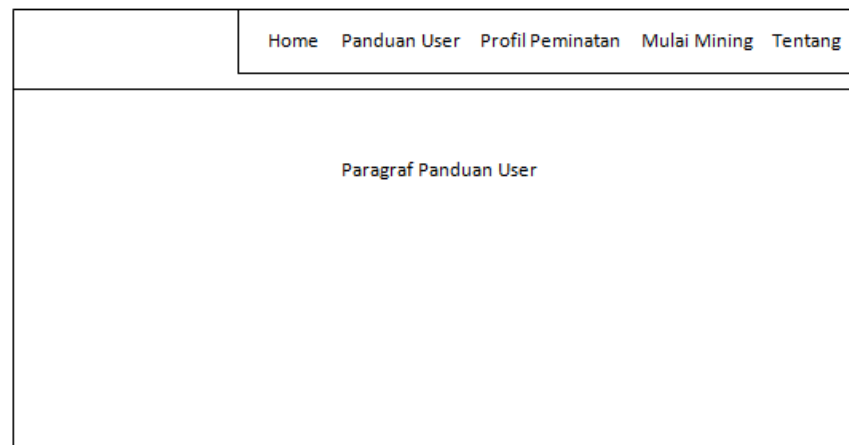
Hasil rancangan tampilan beranda di gambarkan pada gambar berikut:



Gambar 3.6 Rancang Tampilan *Home*

2. Rancang tampilan panduan *user*

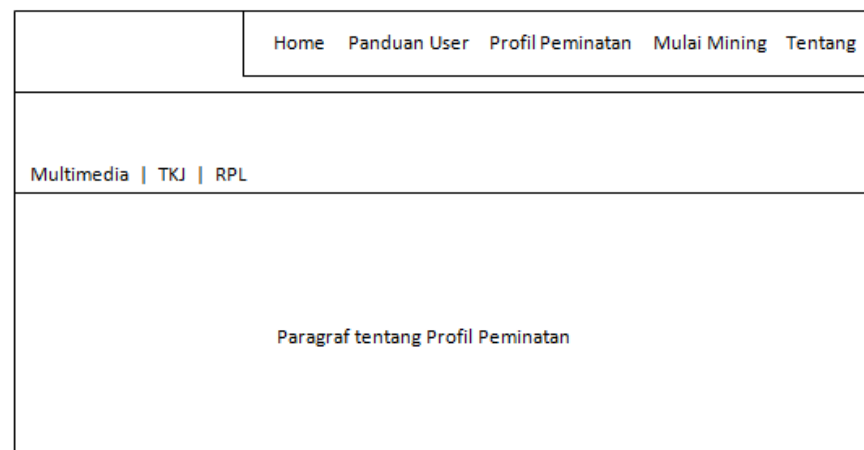
Pada tampilan panduan *user*, *user* bisa mendapatkan informasi tentang cara menggunakan aplikasi sistem pendukung keputusan, data apa saja yang akan dimasukkan dalam proses prediksi data. Hasil rancangan tampilan panduan *user* ditujukan pada gambar berikut:



Gambar 3.7 Rancang Tampilan Panduan *User*

3. Rancang tampilan profil peminatan

Pada tampilan profil peminatan, *user* dapat melihat informasi mengenai profil peminatan yang ada di Program Studi Pendidikan Teknik Informatika dan Komputer Universitas Negeri Jakarta. Hasil rancangan tampilan profil peminatan ditunjukkan pada gambar berikut:



Gambar 3.8 Rancang Tampilan Profil Peminatan

4. Rancang tampilan mulai mining (*Step 1*)

Pada tampilan mulai *mining* (*Step 2*), *user* diberikan *form* data mahasiswa yang harus diisi oleh *user* untuk melakukan proses SPK. *Form*

yang terdiri dari nama mahasiswa, NIM dan angkatan mahasiswa. Tekan lanjut untuk masuk tahap selanjutnya. Hasil rancangan tampilan mulai mining ditunjukkan pada gambar berikut:

The image shows a web form layout for the 'Mulai Mining' step. At the top, there is a navigation bar with the following links: Home, Panduan User, Profil Peminatan, Mulai Mining, and Tentang. Below the navigation bar, the form consists of several input fields and a button. The first field is a single-line text input labeled 'Form Nama'. Below it are two side-by-side input fields: a small one on the left and a larger one labeled 'Form NIM'. Underneath these is a single-line text input labeled 'Form Angkatan'. At the bottom of the form is a button labeled 'Lanjutkan'.

Gambar 3.9 Rancang Tampilan Mulai Mining (Step 1)

5. Rancang tampilan tentang

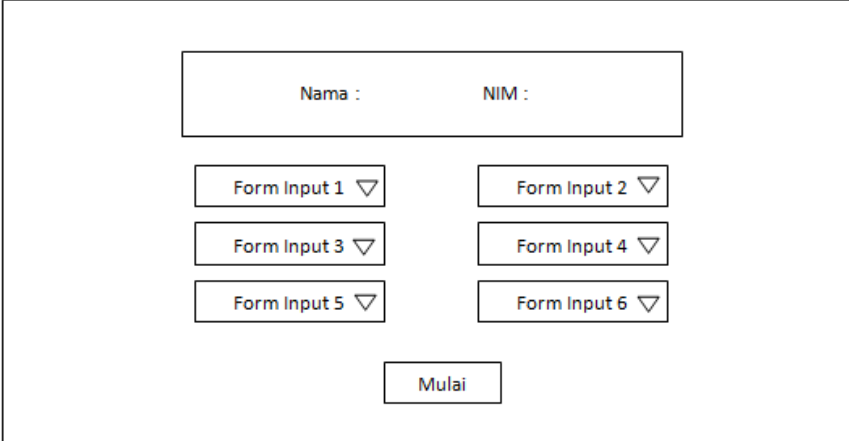
Pada rancangan tampilan tentang, *user* dapat melihat informasi tentang aplikasi

The image shows a web page layout for the 'Tentang' section. At the top, there is a navigation bar with the following links: Home, Panduan User, Profil Peminatan, Mulai Mining, and Tentang. Below the navigation bar, the main content area contains a single paragraph labeled 'Paragraf Tentang'.

Gambar 3.10 Rancang Tampilan Tentang

6. Rancang tampilan mulai *mining* (Step 2)

Pada rancangan tampilan mulai *mining* (Step 2), setelah mengisi data mahasiswa pada *form* mulai mining pertama *user* diharuskan mengisi *form* di tampilan mulai *mining* langkah 2 ini. Data yang di masukkan berupa nilai akademik mahasiswa dengan skala A-E. Data tersebut akan diproses untuk memprediksi pilihan peminatan yang mungkin dapat di ambil oleh *user* berdasarkan nilai yang mereka masukkan. Hasil rancangan tampilan mulai *mining* langkah 2 ditujukan pada gambar berikut:



The image shows a user interface for the 'Start Mining' step. It consists of a rectangular frame containing several elements: a top section with two input fields labeled 'Nama :' and 'NIM :'; a middle section with six dropdown menus arranged in two columns, labeled 'Form Input 1' through 'Form Input 6'; and a bottom section with a single button labeled 'Mulai'.

Gambar 3.11 Rancang Tampilan Mulai Mining (Step 2)

7. Rancang tampilan hasil akhir

Pada tampilan hasil akhir, *user* akan mendapat hasil proses prediksi SPK yang telah dilakukan berdasarkan data yang di masukkan pada tahap mulai *mining* 1 dan 2. *User* dapat mencetak hasil prediksi dengan mengunduh *file* pdf atau kembali ke menu utama. Hasil rancangan tampilan hasil akhir ditujukan pada gambar berikut:

Diagram yang menunjukkan rancangan tampilan akhir. Terdapat form input dengan label 'Nama :' dan 'NIM :'. Di bawahnya terdapat enam input field yang disusun dalam dua kolom: 'Input 1' dan 'Input 2' di baris pertama, 'Input 3' dan 'Input 4' di baris kedua, serta 'Input 5' dan 'Input 6' di baris ketiga. Setelah input field, terdapat dua baris area untuk 'Teks' dan 'Hasil'. Di bagian bawah terdapat dua tombol: 'Kembali Ke Menu' dan 'Cetak Hasil'.

Gambar 3.12 Rancang Tampilan Hasil Akhir

8. Rancang tampilan halaman *login* admin

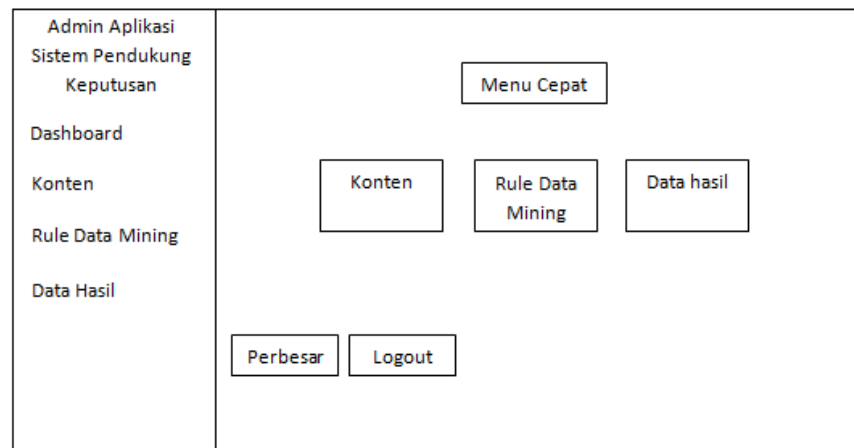
Pada tampilan halaman *login*, admin dapat memasukkan *username* dan *password* yang sudah disediakan pada *form* untuk masuk ke halaman admin. Hasil rancangan tampilan halaman *login* ditujukan pada gambar berikut:

Diagram yang menunjukkan rancangan tampilan login admin. Terdapat tombol 'Akses Admin' di bagian atas. Di bawahnya terdapat form input yang terdiri dari dua baris: 'Username' dan 'Password'. Di sisi kanan form input terdapat ikon lingkaran dengan panah yang menunjukkan ke arah kanan.

Gambar 3.13 Rancang Tampilan *Login* Admin

9. Rancang tampilan halaman *dashboard* admin

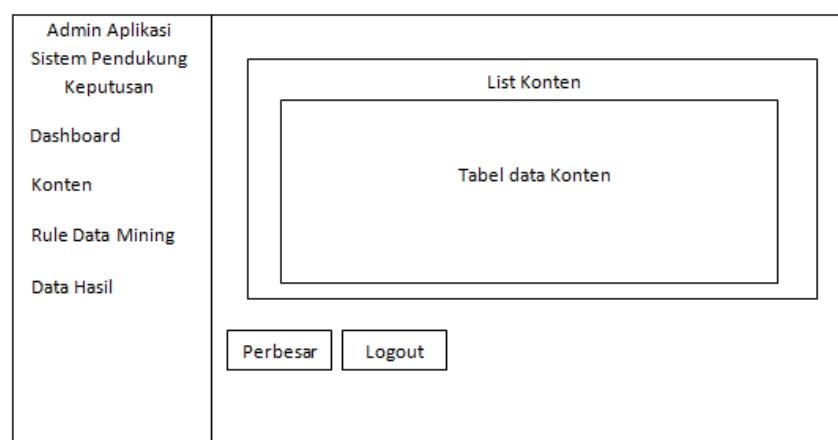
Pada tampilan *dashboard* admin, admin dapat melihat menu sederhana menuju menu lain untuk memasukkan dan melihat data. Hasil rancangan tampilan halaman *dashboard* admin ditujukan pada gambar berikut:



Gambar 3.14 Rancang Tampilan *Dashboard*

10. Rancang tampilan halaman konten

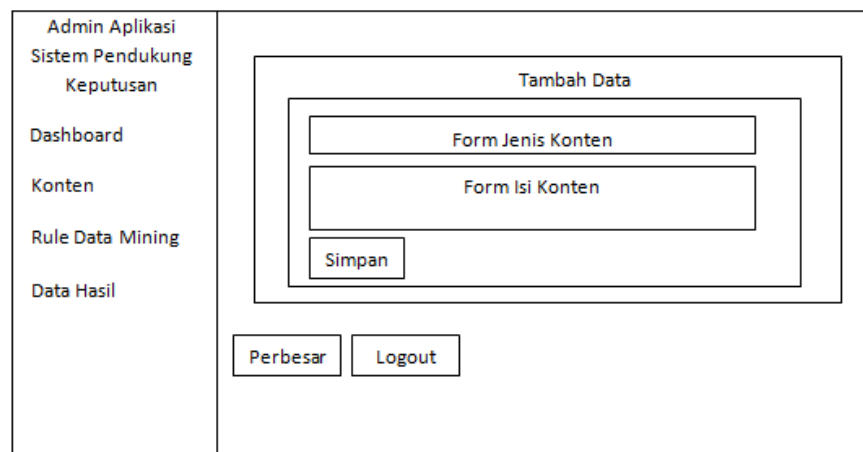
Pada tampilan halaman konten, *admin* dapat melihat data – data yang telah di masukkan pada *list* konten. Hasil rancangan tampilan halaman konten ditujukan pada gambar berikut:



Gambar 3.15 Rancang Tampilan Konten

11. Rancang tampilan halaman tambah konten

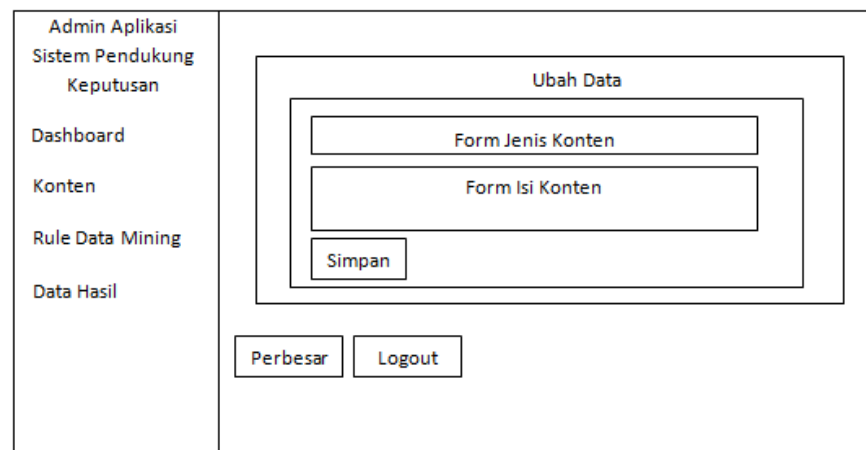
Pada tampilan halaman tambah konten, *admin* dapat menambah data pada halaman ini. Ada 2 *form* yang harus diisi untuk menambahkan data. Hasil rancangan tampilan halaman tambah konten ditujukan pada gambar berikut:



Gambar 3.16 Rancang Tampilan Tambah Konten

12. Rancang tampilan halaman ubah konten

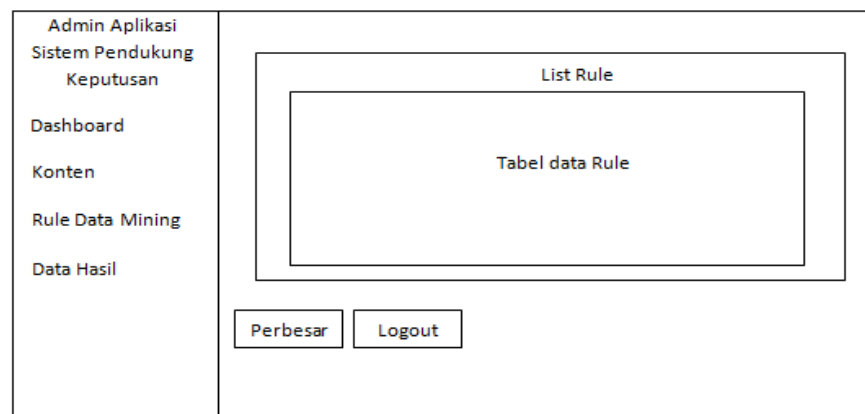
Pada tampilan ubah konten, *admin* dapat mengubah data pada halaman ini. Sama seperti halaman tambah konten, ubah konten juga memiliki 2 *form* yang akan diisi oleh nilai yang akan diubah. Hasil rancangan tampilan halaman ubah konten ditujukan pada gambar berikut:



Gambar 3.17 Rancang Tampilan Ubah Konten

13. Rancang tampilan halaman *rule data mining*

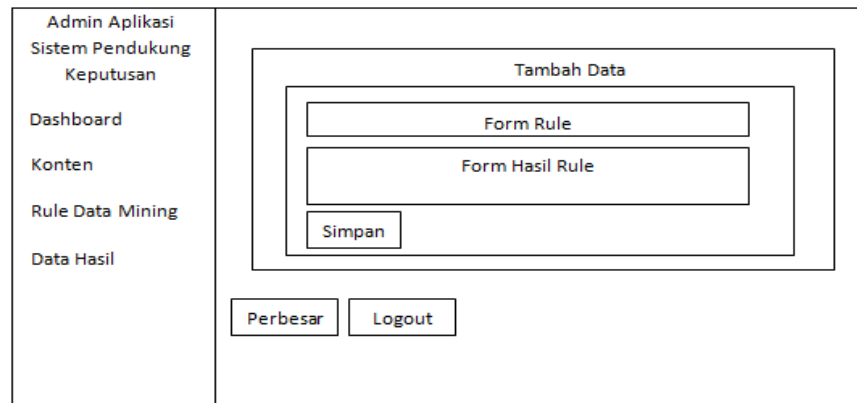
Sama seperti halaman konten, halaman *rule data mining* juga menyajikan data – data yang pernah dimasukkan oleh *admin*, data tersebut dapat di ubah maupun dihapus. Hasil rancangan tampilan halaman *rule data mining* ditunjukkan pada gambar berikut:



Gambar 3.18 Rancang Tampilan *Rule Data Mining*

14. Rancang tampilan halaman tambah *rule data mining*

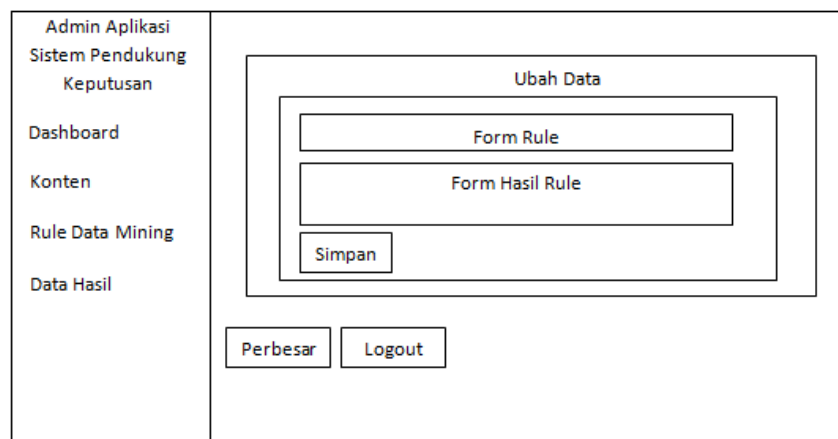
Tampilan halaman tambah data *rule*, admin dapat menambah data *rule* dengan mengisi *form* yang telah disediakan. Hasil rancangan tampilan halaman tambah *rule data mining* ini ditunjukkan pada gambar berikut:



Gambar 3.19 Rancang Tampilan Tambah Rule Data Mining

15. Rancang tampilan halaman ubah *rule data mining*

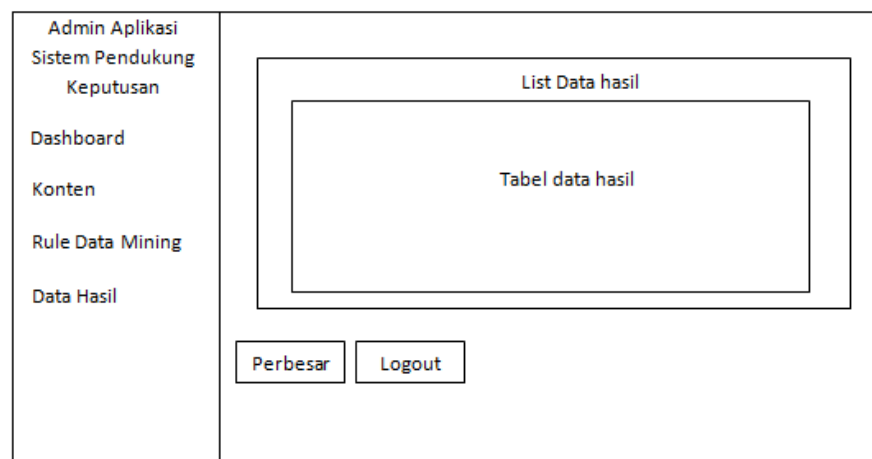
Pada halaman ubah *rule*, admin dapat mengubah data yang ada dengan mengubah nilai yang ada pada *form* ubah *rule data mining*. Hasil rancangan tampilan halaman ubah *rule data mining* ini ditunjukkan pada gambar berikut:



Gambar 3.20 Rancang Tampilan Ubah Rule Data Mining

16. Rancang tampilan halaman data hasil

Sama seperti menu sebelumnya, tampilan data hasil juga menampilkan data. Tetapi bukan data yang di masukkan oleh *admin*, melainkan data yang telah di masukkan oleh *user*. Hasil rancangan tampilan halaman data hasil ditujukan pada gambar berikut:



Gambar 3.21 Rancang Tampilan Data hasil

2.5.5. Implementasi dan Testing Sistem

Pada tahap *testing*/pengujian aplikasi ini menggunakan metode Black Box. Black Box *Testing* merupakan pengujian yang berfokus pada spesifikasi fungsional dari perangkat lunak, tester dapat mendefinisikan kumpulan kondisi input dan melakukan pengetesan pada spesifikasi fungsional program.

Pada black box testing terdapat jenis teknik disain tes yang dapat dipilih berdasarkan pada tipe testing yang akan digunakan, yang diantaranya :

- 1) *Equivalence Class Testing*
- 2) *Boundary Value Testing*
- 3) *Use Case Testing*
- 4) *Comparison Testing*

5) *Performance Testing*

6) *Endurance Testing*

Pada aplikasi ini digunakan pengujian blackbox dengan tipe *Equivalence Class Testing*. *Equivalence Class Testing* merupakan metode black box testing yang membagi domain masukan dari suatu program ke dalam kelas-kelas data, dimana test cases dapat diturunkan [BCS97a]. Equivalence partitioning berdasarkan pada premis masukan dan keluaran dari suatu komponen yang dipartisi ke dalam kelas-kelas, menurut spesifikasi dari komponen tersebut, yang akan diperlakukan sama (ekuivalen) oleh komponen tersebut. Dapat juga diasumsikan bahwa masukan yang sama akan menghasilkan respon yang sama pula. Nilai tunggal pada suatu partisi ekuivalensi diasumsikan sebagai representasi dari semua nilai dalam partisi.

Analisa partisi pada *Equivalence Class Testing* Black Box

- 1) Tester menyediakan suatu model komponen yang dites yang merupakan partisi dari nilai masukan dan keluaran komponen.
- 2) Masukan dan keluaran dibuat dari spesifikasi dari tingkah laku komponen.
- 3) Partisi adalah sekumpulan nilai, yang dipilih dengan suatu cara dimana semua nilai di dalam partisi, diharapkan untuk diperlakukan dengan cara yang sama oleh komponen (seperti mempunyai proses yang sama).

Dengan menggunakan tipe Black Box ini kebenaran sistem yang diuji dilihat berdasarkan *output* yang ada dari data atau kondisi masukkan yang diberikan untuk fungsi yang ada. Tanpa melihat proses, dari *output* tersebut dapat diukur dan diketahui kesalahan dari program dalam memenuhi kebutuhan *user*.

Pengujian dilakukan menggunakan skenario proses yang telah ditentukan pada tabel 3.3 untuk *user* dan tabel 3.4 untuk admin.

Tabel 3.3 Skenario Proses *User*

No	Fungsi	Skenario Proses	Sistem Bekerja	Keterangan
1.	Menu	<i>User</i> mengklik menu		
2.	Masukkan input data pada mulai mining 1	<i>User</i> mengosongkan form dan menekan tombol lanjutkan		
3.	Masukkan input data pada mulai mining 1	<i>User</i> mengisi seluruh form sesuai dengan yang ada di label form		
4.	Masukkan input pada data mulai mining 1	<i>User</i> mengisi form nim dengan huruf		
5.	Masukkan input pada data mulai mining 1	<i>User</i> mengisi form nim dengan angka lebih dari 6 digit		
6.	Masukkan input pada data mulai mining 2	<i>User</i> mengosongkan form dan mengklik mulai		
7.	Masukkan input pada data mulai mining 2	<i>User</i> melengkapi form dan mengklik mulai		
8.	Masukkan input pada data mulai mining 2	<i>User</i> hanya mengisi sebagian form dan sebagian lainnya di kosongkan lalu mengklik mulai		
9.	Cetak PDF	<i>User</i> mengklik cetak PDF setelah hasil SPK keluar		

Tabel 3.4 Skenario Proses Admin

No	Fungsi	Skenario Proses	Sistem Bekerja	Keterangan
1.	Admin login	<i>User</i> melakukan login dengan <i>username</i> dan password yang benar		
2.	Admin login	<i>User</i> melakukan login dengan <i>username</i> salah dan password benar		
3.	Admin login	<i>User</i> melakukan login dengan <i>username</i> benar dan password salah		

	Admin login	<i>User</i> melakukan login dengan <i>username</i> salah dan <i>password</i> salah		
4.	Menu	<i>User</i> mengklik pada menu cepat dashboard		
5.	Menu	<i>User</i> mengklik menu sisi samping		
6.	Tambah Konten	Masukkan jenis konten, dan kategori konten lalu mengklik “Simpan”		
7.	Tambah Konten	Masukkan jenis konten lalu mengklik “Simpan”		
8.	Tambah Konten	Masukkan kategori konten lalu mengklik “Simpan”		
9.	Data Konten	Klik ikon pensil untuk mengubah isi data		
10.	Data Konten	Klik ikon tempat sampah untuk menghapus data		
11.	Update Konten	Masukkan form yang akan diubah (jenis konten dan kategori konten) lalu klik “Ubah”		
12.	Update Konten	Mengosongkan seluruh form atau mengisi sebagian form yang akan diubah (jenis konten dan kategori konten) lalu klik “Ubah”		
13.	Tambah Data Rule	Masukkan rule, dan pilihan lalu mengklik “Simpan”		
14.	Tambah Data Rule	Masukkan rule lalu mengklik “Simpan”		
15.	Tambah Data Rule	Masukkan pilihan lalu mengklik “Simpan”		
16.	Data Rule	Klik ikon pensil untuk mengubah isi data		
17.	Data Rule	Klik ikon tempat sampah untuk menghapus data		
18.	Ubah data rule	Masukkan form yang akan diubah (rule dan pilihan) lalu klik “Ubah”		

19.	Ubah data rule	Mengosongkan seluruh form atau mengisi sebagian form yang akan diubah (rule dan pilihan) lalu klik “Ubah”		
20.	Data hasil	Klik ikon tempat sampah untuk menghapus data		
21.	Logout	<i>User</i> mengklik Logout		

2.5.6. Operasi dan Maintenance

Pemeliharaan dianggap perlu dilakukan untuk memperbaiki jika ada kesalahan sistem yang tidak berjalan sesuai dengan fungsinya yang mungkin disebabkan karena perubahan tampilan atau perubahan kode program. Walaupun sudah diuji sebelumnya.