

## BAB V PENGEMBANGAN APLIKASI

Bab ini menjelaskan mengenai tahapan yang dilakukan dalam mengembangkan aplikasi *case base reasoning* untuk membantu bagian *customer service* dalam memberikan solusi pada komplain pelanggan. Tahapan yang akan dilakukan meliputi desain, implementasi, dan *testing*.

### 5.1 Desain

Desain adalah rancangan awal dari suatu sistem yang dikehendaki. Adapun sistem yang dimaksud pada penelitian ini adalah aplikasi yang dikembangkan untuk menanggapi komplain pelanggan dengan pendekatan *case base reasoning*.

#### 5.1.1 Perancangan Struktur Kasus

Pada tahapan desain ini akan dirancang struktur kasus sebagai dasar pembuatan database kasus. Perancangan struktur kasus dilakukan dengan menentukan *feature* dan atribut yang melekat pada *feature* tersebut. Atribut pada *feature* antara lain *feature name*, *feature class*, *data type*, domain dan *similarity matching*.

Perancangan struktur kasus didasarkan pada data sekunder yang telah ada. Pada tahapan perancangan struktur kasus ini data sekunder dikembangkan dan dipilah agar lebih terperinci. Adapun *feature name* pada perancangan struktur kasus ini adalah sebagai berikut :

1. Nama *Customer* adalah nama pelanggan yang memberikan komplainnya pada perusahaan.
2. Alamat adalah daerah tempat pelanggan. Lokasi pelanggan dalam hal ini adalah provinsi dimana pelanggan berada.
3. Jenis pelumas adalah penggolongan pelumas berdasarkan fungsi atau kegunaannya pada *engine* tertentu.
4. Status pelumas adalah keadaan pelumas terkomplain, yang menyatakan pelumas dalam keadaan belum digunakan (*new oil*) atau pelumas sedang digunakan pada *engine* (*in service*).
5. Fisik pelumas adalah ciri-ciri dari pelumas terkomplain yang dapat diamati dan dirasakan secara fisik dengan indera manusia.

6. Kinerja mesin adalah kondisi mesin sebagai sistem pelumas yang bisa diamati dan dirasakan dengan indera manusia.
7. *Trouble* adalah gangguan, permasalahan atau temuan pendukung terkait kerja sistem pelumas dan bisa juga menggambarkan keadaan kerja mesin
8. Solusi adalah pemecahan masalah terhadap komplain pelanggan.
9. Status komplain adalah keadaan dari suatu komplain apakah menjadi tanggung jawab perusahaan untuk menindaklanjuti dan mengganti kerugian pelanggan (*justified*) atau komplain terjadi karena kesalahan dari pihak pelanggan sehingga perusahaan tidak bertanggung jawab terkait komplain tersebut (*unjustified*).

Adapun atribut *feature* pada perancangan struktur kasus dapat diperlihatkan pada Tabel 5.1 sebagai berikut :

Tabel 5.1 Struktur Kasus

No	Feature Name	Feature Class	Data Type	Domain
1	NamaCustomer	ID	Text (20 karakter)	Any Valid Name
2	Alamat	ID	Text (15 karakter)	Any Valid Name
3	Jenis Pelumas	Adjustment	Text	2 Stroke engine, 4 Stroke engine, Marine, Sistem hidrolik
4	Status Pelumas	Adjustment	Text	New oil In service
5	Fisik Pelumas	Adjustment	Text	Ok, Pelumas terlihat <i>cloudy</i> , Pelumas terlihat <i>milky</i> , Pelumas terlihat keruh, Perubahan warna, Berbau terbakar,
6	Kinerja Mesin	Adjustment	Text	Ok, Mesin mengalami gagal starter, Mesin mengalami macet, Mesin cepat panas, Bunyi mesin kasar, Suhu tinggi performa turun, Hidrolik tidak normal, Pompa hidrolik bergetar.
7	<i>Trouble</i>	Adjustment	Text	Terdapat pengotor pada filter oli, Terdapat endapan pada mesin, Ditemukan kontaminan air, Ditemukan kontaminan produk lain, Ditemukan kontaminan bbm, Top up pelumas, Suplai pelumas kurang, Ditemukan ring piston longgar, Indikator menunjukkan penurunan tekanan
8	Solusi	Solution	Text	Any Valid Name
9	Status Komplain	Solution	Text	<i>Justified, Unjustified</i>



### 5.1.2 Pendefinisian Nilai Kemiripan (*Similarity*) dan Pembobotan

Kemiripan atau *similarity* merupakan dasar dari pendekatan *case base reasoning*. *Case base reasoning* dalam menghasilkan output yang diinginkan user, dengan cara mencari (*retrieve*) kemiripan antara kasus baru (*new case*) terhadap kasus-kasus lama (*past cases*). *Nearest neighbor* merupakan salah satu teknik *retrieval* yang digunakan dalam mencari nilai kemiripan. Cara kerja *nearest neighbor* adalah dengan mencari nilai kemiripan antar domain dalam satu *feature* (kemiripan lokal) selanjutnya kemiripan pada masing-masing *feature* diakumulasikan menjadi nilai kemiripan total (kemiripan global). Dalam *tool esteem*, metode *nearest neighbor* lebih dispesifikan sesuai dengan tipe data yang digunakan. Misalnya untuk tipe data *text*, mencari nilai kemiripan bisa dilakukan dengan menggunakan pencocokan domain *feature* (*Type of feature matching*) jenis *exact* atau *partial*.

Pendefinisian nilai kemiripan untuk pengembangan aplikasi pada penelitian ini dilakukan dengan melihat tipe data yang digunakan dan hubungan antar domain dalam satu *feature*. Berdasarkan *feature-feature* yang telah dijelaskan pada perancangan struktur kasus, maka tipe data yang digunakan adalah jenis *text*. Sedangkan pada penelitian ini nilai kemiripan antar domain dalam satu *feature* dikelompokkan kedalam tiga kemiripan yaitu kemiripan mutlak dengan nilai 1, kemiripan rentang dengan nilai antara lebih dari 0 dan kurang dari 1 ( $0 < x < 1$ ), dan tidak ada kemiripan dengan nilai 0. Kemiripan mutlak didefinisikan ketika suatu domain kasus baru sama dengan domain kasus lama, misalkan pada *feature* status pelumas, antara kasus baru dengan kasus lama memiliki domain yang sama yaitu *new oil*. Kemiripan rentang didefinisikan ketika domain pada kasus baru dan kasus lama tidak sama mutlak, akan tetapi masih memiliki kemiripan. Misalkan domain kasus baru adalah *cloudy* dan domain kasus lama adalah *milky*, kedua domain tersebut tidak sama mutlak tetapi masih memiliki kemiripan yaitu sama-sama menjelaskan tentang sifat keruh dari suatu zat atau pelumas. Tidak adanya kemiripan didefinisikan ketika domain dalam satu *feature* antara kasus baru dengan kasus lama tidak memiliki kemiripan. Misalnya *feature* status pelumas untuk kasus baru memiliki domain *in service* sedangkan kasus lama memiliki domain *new oil*. Kedua domain tersebut menjelaskan dua keadaan yang berbeda sehingga nilai kemiripannya adalah 0. Penentuan *Type of feature matching* untuk setiap *feature* dapat dijelaskan sebagai berikut

1. *Feature* status pelumas menggunakan *Type of feature matching exact*. Karena status komplain hanya terdiri dari dua domain dan keduanya memiliki pengertian yang berbeda.

2. *Feature* jenis pelumas, fisik pelumas, kinerja mesin, dan *trouble* menggunakan *Type of feature matching partial word (case indifferent)*. Tipe partial memungkinkan pengelompokan kemiripan menjadi tiga kelompok sesuai penjelasan diatas. *Partial word* dikhususkan untuk tipe data teks yang berbentuk kalimat (terdiri dari beberapa kata), dan *case indifferent* dimaksudkan untuk menghilangkan perbedaan penggunaan antara huruf biasa dan kapital sehingga kesalahan dalam meng-input-kan domain bisa diminimalkan.

Adapun pendefinisian nilai kemiripan pada masing-masing *feature* dijelaskan pada Tabel 5.2. Pendefinisian nilai kemiripan secara lengkap dapat dilihat pada lampiran 2.

Tabel 5.2 Nilai kemiripan *feature* Jenis Pelumas

Parameter	2 stroke engine	4 stroke engine	Hidrolik	Marine
2 stroke engine	1	0,67	0	0
4 stroke engine	0,67	1	0	0
Hidrolik	0	0	1	0
Marine	0	0	0	1

Pada teknik *retrieval nearest neighbor* dilakukan pembobotan pada masing-masing *feature*. Pembobotan didasarkan pada seberapa penting suatu *feature* dibandingkan dengan *feature* yang lain dalam menghasilkan *solution feature*, dimana dalam hal ini yang termasuk *solution feature* adalah *feature* solusi dan status komplain. Penentuan nilai pembobotan disajikan pada Tabel 5.3.

Tabel 5.3 Pembobotan pada *adjustment feature*

Feature Name	Type of Feature Matching	Bobot (Weight)	Pertimbangan
Jenis Pelumas	<i>Partial (case indifferent)</i>	1	Jenis pelumas adalah salah satu atribut dari produk pelumas sehingga perlu dipertimbangkan dalam aplikasi <i>prototype</i> ini. Beberapa kasus yang ada mengindikasikan bahwa jenis pelumas yang berbeda bisa memiliki ciri-ciri kasus yang hampir sama. Sehingga <i>feature</i> jenis pelumas dirasa tidak lebih penting dari <i>feature</i> yang lainnya dalam mempengaruhi munculnya solusi kasus.



<i>Feature Name</i>	<i>Type of Feature Matching</i>	<i>Bobot (Weight)</i>	<i>Pertimbangan</i>
Status Pelumas	<i>Exact</i>	2	Terdapat dua status komplain yaitu <i>new oli</i> dan <i>in service</i> . Beberapa kasus mengindikasikan bahwa status komplain memberi pengaruh pada <i>feature</i> tertentu. Misalkan untuk status pelumas <i>new oil</i> maka variabel kinerja mesin akan bernilai Ok, karena pelumas belum digunakan pada mesin. <i>Feature</i> status pelumas sangat penting dalam menentukan <i>solution feature</i> .
Fisik Pelumas	<i>Partial word (case indifferent)</i>	2	Pelumas adalah subyek utama pada kasus komplain pelanggan ini. Sebagai produk yang dikomplain pelanggan yang secara fisik dapat dilihat oleh pelanggan dengan ciri-ciri umum seperti yang diberikan pada perancangan struktur kasus. Deskripsi tentang keadaan fisik pelumas sangat penting dalam menghasilkan <i>solution feature</i> .
Kinerja Mesin	<i>Partial word (case indifferent)</i>	2	Kinerja mesin merupakan kondisi dari mesin yang bisa dirasakan dan diamati oleh user atau pelanggan. Kondisi kerja mesin sering dipengaruhi oleh kondisi fisik dari pelumas. <i>Feature</i> kinerja mesin sangat penting dan saling melengkapi <i>feature</i> fisik pelumas dan <i>trouble</i> dalam menghasilkan <i>solution feature</i> .
<i>Trouble</i>	<i>Partial word (case indifferent)</i>	2	<i>Trouble</i> adalah gangguan, permasalahan atau temuan pendukung terkait kerja sistem pelumas. <i>Feature trouble</i> sangat penting karena saling melengkapi <i>feature</i> fisik pelumas, dan kinerja mesin dalam menghasilkan <i>solution feature</i> .

### 5.1.3 Pendefinisian Rule Adaptasi

Adaptasi merupakan proses penyesuaian *feature* antara kasus baru dengan kasus lama. Adaptasi juga berfungsi untuk membatasi keadaan tertentu dan mengubah *solution feature* pada kasus lama agar sesuai dengan kondisi kasus baru. Didalam *case base reasoning*, proses adaptasi terdapat pada tahapan *revise*. Pada pengembangan aplikasi *prototype* ini digunakan *rule* untuk proses adaptasi, ketika kasus baru dengan nilai kemiripan yang tinggi akan disimpan kedalam database kasus. *AANamaCustomer* adalah salah satu pendefinisian *rule* yang dibuat untuk mengadaptasikan domain kasus baru dengan domain kasus lama. Jadi *rule AANamaCustomer* berfungsi mengubah domain *feature* Nama *Customer* pada kasus lama yang terpilih, dengan domain Nama *Customer*

pada kasus baru. Pendefinisian *rule* adaptasi seluruhnya disajikan pada lampiran 3. Adapun *rule* *AANamaCustomer* adalah sebagai berikut :

```

If
  TRUE
Then
  KOMPLAINSC:>NamaCustomer=KOMPLAINTC:>NamaCustomer

```

Dimana jika domain *feature* bernilai benar (*true*) maka domain *feature* Nama *Customer* pada database KOMPLAIN (nama database) kasus lama adalah sama dengan domain *feature* Nama *Customer* pada kasus baru. SC adalah *Selected Case* atau kasus lama terpilih dan TC adalah *Target Case* atau kasus baru.

AAAlamat, AAJenisPelumas, AAStatusPelumas, AAFisikPelumas, AAKinerjaMesin, AATrouble, AASolusi adalah *rule* dimana pendefinisian sama dengan *rule* *AANamaCustomer*. Yang membedakan adalah penulisan nama *feature* pada pendefinisian *rule*-nya. Secara umum pendefinisian *rule* adaptasi adalah sebagai berikut:

```

If
  TRUE
Then
  KOMPLAINSC:X=KOMPLAINTC:X

```

Dimana :

SC = Selected Case

TC = Target Case

X = nama feature

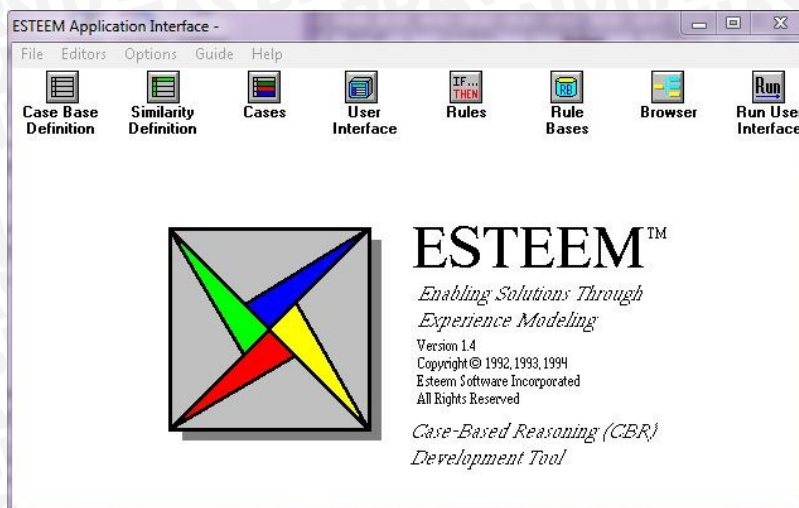
## 5.2 Implementasi

Implementasi merupakan tahapan dimana teori dan desain perancangan yang telah dibuat diterapkan dalam pembuatan aplikasi berbasis komputer. Pada tahapan ini diharapkan aplikasi yang dibuat sesuai dengan yang direncanakan sebelumnya. Adapun aplikasi *prototype* yang dikembangkan pada penelitian ini diberi nama Aplikasi Penanganan Komplain Pelumas (APKPelumas).

Pada tahap implementasi ini dilakukan pembuatan Aplikasi Penanganan Komplain Pelumas dengan menggunakan *tool* esteem 1.4. Langkah dalam menjalankan *tool* esteem 1.4 adalah sebagai berikut:

1. Buka folder file aplikasi esteem disimpan
2. Cari file ESTEEM14 dengan tipe file *windows batch file*
3. Double klik File ESTEEM14 tersebut maka akan keluar tampilan *interface* dari *tool* esteem seperti Gambar 5.1.





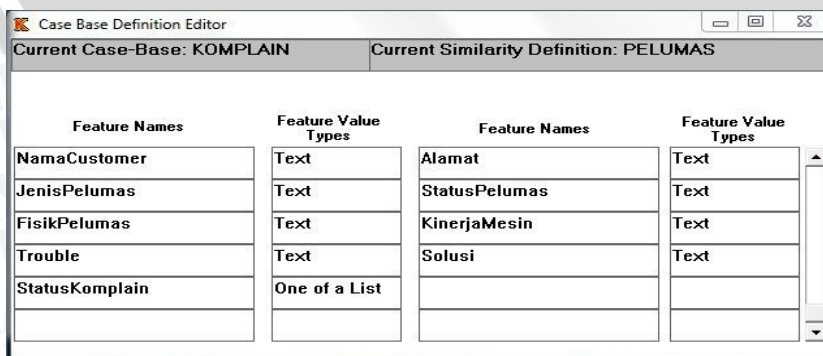
Gambar 5.1 Interface tool esteem 1.4

Selanjutnya tahapan dalam membuat aplikasi *case base reasoning* dengan *tool* esteem dapat dijelaskan sebagai berikut :

1. Pembuatan database kasus

Pembuatan database kasus dengan *tool* esteem dimulai dengan meng-input-kan *feature name* dan *feature type value* pada tampilan *case base definition editor*. Penentuan *feature name* dan *feature type value* didasarkan pada struktur kasus yang telah dirancang sebelumnya. Adapun proses pembuatan database kasus pada aplikasi esteem dapat dijelaskan sebagai berikut:

- a. Pilih icon *case base definition*, pilih *New*, dan ketik KOMPLAIN pada *Case Base Name*. Selanjutnya muncul tampilan *Case Base Definition Editor* seperti Gambar 5.2.
- b. Ketik *feature name* sesuai yang ada pada perancangan struktur kasus.
- c. Tentukan tipe data atau *feature value type*
- d. Exit



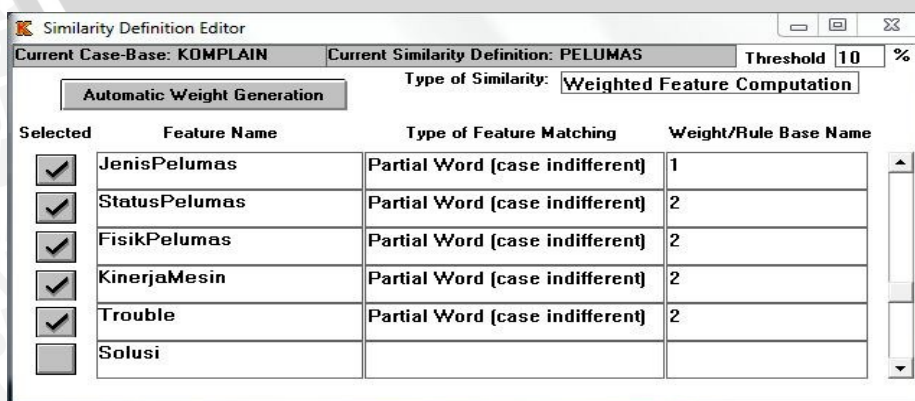
Gambar 5.2 Case Base Definition Editor

## 2. Menentukan nilai kemiripan dan pembobotan

Penentuan nilai kemiripan pada *tool esteem* dilakukan dengan menentukan *type of feature matching*. *Type of feature matching* diberikan pada *feature* yang dijadikan sebagai *adjustment feature*. *Type of feature matching* yang digunakan pada kasus ini adalah *partial word (case indifferent)*. Adapun pengertian *partial word (case indifferent)* adalah memberikan nilai pencocokan pada susunan kata dengan tidak mempertimbangkan penggunaan huruf kapital yang terkandung dalam susunan kata tersebut sehingga penggunaan *type of feature matching* ini akan memberikan fleksibilitas dari susunan kata yang di-input-kan oleh *user*.

Penandaan *adjustment feature* dilakukan dengan memberi tanda *check* (✓) pada kolom *Selected*. Setelah ditentukan *Type of feature matching* maka langkah selanjutnya adalah menentukan bobot nilai pada masing-masing *adjustment feature*. Penentuan bobot ini didasarkan pada seberapa penting suatu atribut dibandingkan dengan atribut yang lain dalam menghasilkan *output* yang diharapkan, dimana dalam hal ini *output* yang diharapkan adalah status komplain. Alasan dan pertimbangan dalam memberikan bobot nilai pada masing-masing *feature* telah dijelaskan pada tahapan desain Tabel 5.4. Adapun langkah dalam mendefinisikan nilai kemiripan dan pembobotan adalah sebagai berikut:

- Klik ikon *Similarity Definition*, pilih *New*, dan ketik PELUMAS pada *Similarity Definition Name*. Selanjutnya muncul tampilan *Similarity Definition Editor* seperti pada Gambar 5.3.
- Pilih *Weighted Feature Computation* pada *Type of Similarity*.
- Tentukan nilai *threshold* (batas terendah nilai kemiripan total yang ditampilkan)
- Beri tanda *check* (✓) pada *adjustment feature* dan tentukan jenis pencocokan pada kolom *Type of feature matching*.
- Tentukan nilai pembobotan *feature* pada kolom *Weight/Rule Base Name*.



Gambar 5.3 *Similarity Definition Editor*



### 3. Input data komplain pelanggan dalam database kasus

Input data komplain pelanggan dilakukan dengan memasukkan data kasus dari departement *customer service* yang telah dipilah dan dikembangkan. Terdapat 22 kasus komplain pelanggan yang akan dimasukkan pada database kasus. Langkah-langkah yang dilakukan untuk input data pada *tool* esteem adalah sebagai berikut:

- Klik ikon *Cases*, pilih *New*, dan ketik case1 pada *Case Name*.
- Inputkan data kasus pada tampilan *Case Editor*.

Adapun tampilan input data kasus komplain pelanggan diperlihatkan pada Gambar 5.4 dan Gambar 5.5.

Feature Names	Feature Values	Case Saved	Clear Case
NamaCustomer	Udaya		
Alamat	Jawa Timur		
JenisPelumas	4 stroke engine		
StatusPelumas	In service		
FisikPelumas	Ok		
KinerjaMesin	Mesin mengalami gagal start		
Trouble	Terdapat pengotor pada filter oli		
Solusi	Ganti pelumas dan bersihkan filter oli		

Gambar 5.4 Input case 1

Feature Names	Feature Values	Case Saved	Clear Case
NamaCustomer	Wigati Motor		
Alamat	Jawa Timur		
JenisPelumas	4 stroke engine		
StatusPelumas	In service		
FisikPelumas	Pelumas terlihat keruh		
KinerjaMesin	Mesin mengalami macet		
Trouble	Terdapat pengotor pada filter oli		
Solusi	Hasil analsis lab, pelumas on spesifikasi. Bersihkan		

Gambar 5.5 Input case 22

### 4. Implementasi *Rule* Adaptasi

Langkah pendefinisian *rule* untuk proses adaptasi adalah sebagai berikut:

- Klik ikon *Rules*, pilih *New*
- Tentukan *rule name*, AANamaCustomer

- c. Ketik *rule* pada tampilan *Rule Editor*.

Adapun *rule* adaptasi *AANamaCustomer* yang di-input-kan adalah sebagai berikut :

If

TRUE

Then

KOMPLAINSC : *NamaCustomer* = KOMPLAINTC : *NamaCustomer*

Selanjutnya *rule* *AANamaCustomer* di-input-kan pada database *rule* (*Rule Bases*)

- d. Klik ikon *Rule Bases*, pilih *New*
- e. Ketik nama *rule base*, misalkan *AutoAdapt*.
- f. Masukkan nama *rule* yang telah dibuat sebelumnya pada tampilan *Rule Base Editor*, Kemudian Exit
- g. Klik ikon *User interface* dan pilih *Additional End User Fuctionality*. Maka muncul tampilan *Additional End User Fuctionality* seperti Gambar 5.6.
- h. Pilih *Yes* pada kolom *allow user to add cases to case base*. Hal ini berfungsi untuk menyimpan hasil adaptasi kedalam database kasus (tahapan *retain*).
- i. Pilih *Yes* pada kolom *allow auto-adaption of retrieved case*.
- j. Pilih *rule base* *AutoAdapt* pada kolom *name of rule base for auto-adaption*.
- k. Klik OK.

**Additional End User Functionality**

Allow user to add cases to case-base?:

Allow user to modify the similarity definition?:

Allow auto-adaption of retrieved cases?:

Name of rule base for auto-adaption:

Name of bitmap file with application information:

Application Startup Function:

Selected Case Editor Startup Function:

Down Button Function:

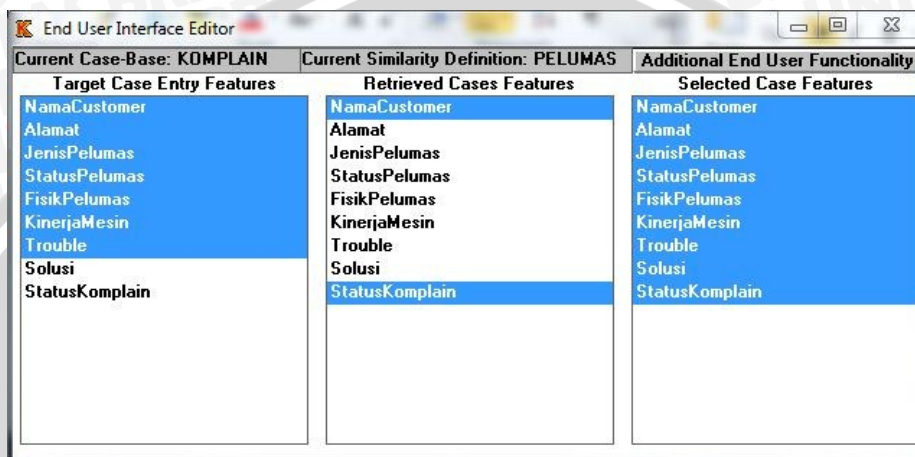
Gambar 5.6 Tampilan *Additional End User Fuctionality*

*Rule* adaptasi yang digunakan pada Aplikasi Penanganan Komplain Pelumas disajikan secara lengkap pada lampiran 3.



## 5. Pemilihan Tampilan *User interface*

Desain *interface* dimaksudkan sebagai tampilan awal yang muncul ketika user menggunakan *prototype* Aplikasi Penanganan Komplain Pelumas. Desain *interface* akan memberikan kemudahan pada user dalam meng-input-kan domain pada masing-masing *feature* untuk menghasilkan output yang diharapkan. Pemilihan desain *interface* dilakukan dengan memilih menu *user interface* yang selanjutnya akan muncul tampilan *End User Interface Editor* seperti Gambar 5.7.



Gambar 5.7 Pengaturan *interface*

Hal-hal yang perlu diperhatikan dalam pemilihan desain *interface* ini adalah:

- Target case entry features* adalah *feature name* yang akan dimunculkan untuk memasukkan data pada kasus baru. Jenis *feature* yang dijadikan sebagai *target case entry features* adalah *adjustment feature*, adapun *feature* yang dijadikan sebagai *target case entry features* antara lain : jenis pelumas, status pelumas, fisik pelumas, kinerja mesin, dan *trouble*.
- Retrieve case features* adalah *feature name* yang akan dimunculkan *tool esteem* pada proses pencarian kasus pada database kasus yang memiliki kemiripan dengan kasus baru. Pada kasus ini *feature name* yang dijadikan sebagai *retrieve case features* adalah nama *customer* dan status komplain.
- Selected case feature* adalah *feature name* yang ingin ditampilkan dari kasus lama yang memiliki kemiripan dengan kasus baru. Tujuan dari pemilihan *feature name* pada *selected case feature* adalah untuk mengetahui lebih detail nilai dari setiap *feature* kasus lama untuk dapat dimanfaatkan sebagai informasi dan pengetahuan. Pada aplikasi ini semua *feature* akan dimunculkan sehingga user

tidak hanya tahu status komplain tetapi juga solusi dari kasus lama yang bisa diterapkan pada kasus baru yang bersangkutan.

### 5.3 Pengujian Aplikasi (*Testing*)

*Testing* merupakan tahapan terakhir dalam mengembangkan aplikasi setelah tahap implementasi selesai dilakukan. Tahapan *testing* bertujuan untuk mengetahui apakah Aplikasi Penanganan Komplain Pelumas dapat dijalankan dengan benar dan sesuai dengan desain perancangan. *Testing* atau pengujian aplikasi ini dilakukan dengan pengujian verifikasi, validasi dan *prototype*.

#### 5.3.1 Uji Verifikasi

Uji verifikasi bertujuan untuk mengetahui apakah implementasi Aplikasi Penanganan Komplain Pelumas telah sesuai dengan tahap desain. Hal ini untuk menyesuaikan aplikasi hasil implementasi agar tidak menyimpang dari tahapan desain yang dilakukan sebelumnya. Uji verifikasi Aplikasi Penanganan Komplain Pelumas dilakukan dengan melihat tampilan untuk proses input kasus (*case*), apakah telah sesuai dengan struktur kasus yang telah dibuat pada tahapan desain, apakah semua *feature* telah tersedia sesuai dengan desain struktur kasus dan dapat digunakan untuk input kasus kedalam database KOMPLAIN. Uji verifikasi juga dilakukan dengan menjalankan adaptasi (*adaptation*) apakah berjalan sesuai dengan yang direncanakan.

Berdasarkan Gambar 5.8 yaitu tampilan *Case Editor* maka dapat dikatakan bahwa tampilan untuk proses input data kasus telah sesuai dengan perancangan struktur kasus pada tahapan desain, semua *feature name* tersedia dan dapat digunakan untuk input data kasus.

Feature Names	Feature Values	Case Saved	Clear Case
NamaCustomer			
Alamat			
JenisPelumas			
StatusPelumas			
FisikPelumas			
KinerjaMesin			
Trouble			
Solusi			

Gambar 5.8 Tampilan *Case Editor* untuk input data



Gambar 5.9 adalah tampilan kasus lama sedangkan Gambar 5.10 adalah kasus hasil proses adaptasi kasus lama dengan kasus baru. Berdasarkan Gambar 5.9 dan Gambar 5.10 maka dapat dikatakan bahwa proses adaptasi dapat berjalan sesuai dengan yang diharapkan pada tahapan desain. *Feature* jenis pelumas pada kasus lama memiliki domain 4 *stroke engine* setelah terjadi proses adaptasi maka *feature* jenis pelumas kasus lama telah berubah sesuai domain kasus baru yaitu 2 *stroke engine*.

NamaCustomer	Udaya
Alamat	Jawa Timur
JenisPelumas	4 stroke engine
StatusPelumas	In service
FisikPelumas	Ok
KinerjaMesin	Mesin mengalami gagal start

Gambar 5.9 *Selected case 1* sebelum adaptasi

NamaCustomer	Hikam
Alamat	Jawa Timur
JenisPelumas	2 stroke engine
StatusPelumas	In service
FisikPelumas	Ok
KinerjaMesin	Bunyi mesin kasar

Gambar 5.10 *Selected case 1* setelah adaptasi

### 5.3.2 Uji Validasi

Uji validasi merupakan pengujian terhadap keabsahan aplikasi yang telah dibuat yaitu apakah keputusan akhir sebagai solusi yang diberikan dapat merepresentasikan tujuan perancangan. Tujuan dari perancangan aplikasi ini adalah untuk dapat mencari informasi dan pengetahuan yang terkandung pada kasus-kasus komplain yang ada untuk digunakan sebagai solusi pada kasus baru dengan cara mencari kemiripan kasus baru dengan kasus lama. Untuk membuktikan bahwa proses pencarian berdasarkan kemiripan ini berjalan dengan benar maka dilakukan pengujian validasi. Dalam uji validasi Aplikasi Penanganan Komplain Pelumas dilakukan dengan membandingkan nilai kemiripan secara perhitungan manual dengan nilai kemiripan yang dihasilkan aplikasi.

Pada tahapan uji validasi ini akan dilakukan pengujian terhadap 5 kasus baru untuk mencari kasus lama yang paling mirip. Selanjutnya nilai kemiripan total yang dihasilkan Aplikasi Penanganan Komplain Pelumas dibandingkan dengan perhitungan nilai kemiripan sesuai rumus *similarity*. Adapun kasus-kasus baru disajikan seperti pada Tabel 5.4.

Tabel 5.4 *New Case*

New case	Jenis Pelumas	Status Pelumas	Fisik Pelumas	Kinerja Mesin	Trouble
1	2 stroke engine	<i>In service</i>	Ok	Bunyi mesin kasar	Terdapat pengotor pada filter oli
2	4 stroke engine	<i>New oil</i>	Pelumas terlihat <i>cloudy</i>	Ok	Ditemukan kontaminan produk lain
3	4 stroke engine	<i>In service</i>	Berbau terbakar	Mesin cepat panas	Terdapat endapan pada mesin
4	Sistem hidrolik	<i>In service</i>	Perubahan warna	Mesin mengalami macet	Ditemukan kontaminan produk lain
5	Marine	<i>New oil</i>	Pelumas terlihat keruh	Ok	Ditemukan kontaminan air

Langkah uji validasi dengan cara menjalankan Aplikasi Penanganan Komplain Pelumas sehingga muncul kasus lama beserta nilai kemiripan terhadap kasus baru.

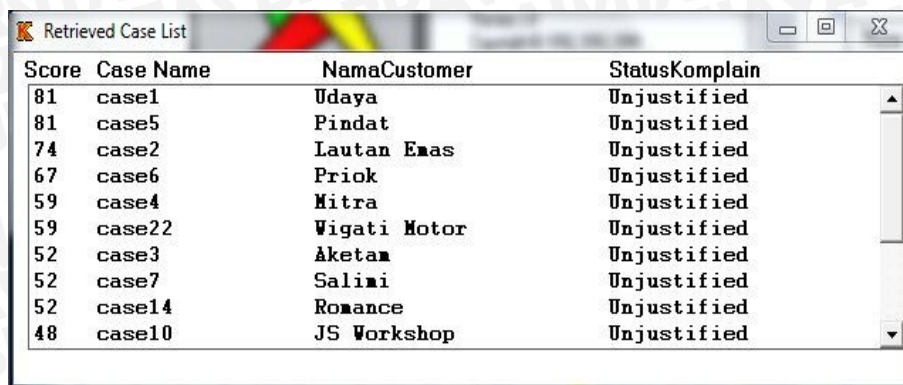
- Jalankan Aplikasi Penanganan Komplain Pelumas.
- Klik menu *Run User interface*.
- Inputkan data *new case* 1 seperti pada Tabel 5.5, pada tampilan *Enter Target Case*.

Enter Target Case	
Alamat	Jawa Timur
JenisPelumas	2 stroke engine
StatusPelumas	In service
FisikPelumas	Ok
KinerjaMesin	Bunyi mesin kasar
Trouble	Terdapat pengotor pada filter

Gambar 5.11 *Tampilan enter target case*

- Klik menu *retrieve*, akan muncul tampilan *Retrieve Case List* seperti pada Gambar 5.12 yang menampilkan daftar kasus lama dengan nilai kemiripan total tertinggi pada urutan teratas.

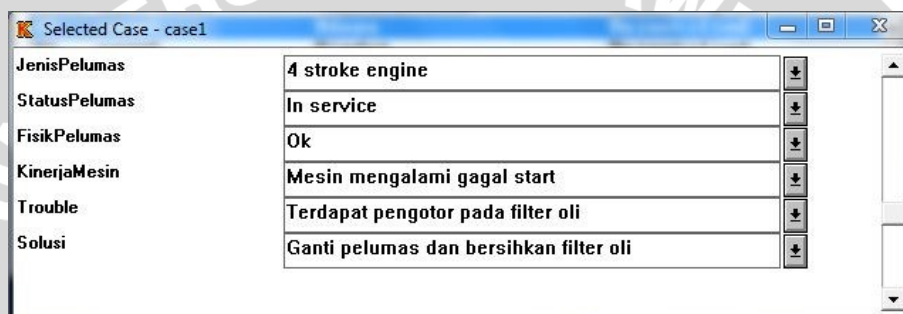




Score	Case Name	NamaCustomer	StatusKomplain
81	case1	Udaya	Unjustified
81	case5	Pindat	Unjustified
74	case2	Lautan Emas	Unjustified
67	case6	Priok	Unjustified
59	case4	Mitra	Unjustified
59	case22	Wigati Motor	Unjustified
52	case3	Aketan	Unjustified
52	case7	Salimi	Unjustified
52	case14	Romance	Unjustified
48	case10	JS Workshop	Unjustified

Gambar 5.12 Tampilan *retrieve case list*

- e. Klik kasus terpilih untuk melihat informasi dan pengetahuan yang bisa digunakan untuk kasus baru. Kasus terpilih disajikan pada Gambar 5.13.



JenisPelumas	4 stroke engine
StatusPelumas	In service
FisikPelumas	Ok
KinerjaMesin	Mesin mengalami gagal start
Trouble	Terdapat pengotor pada filter oli
Solusi	Ganti pelumas dan bersihkan filter oli

Gambar 5.13 Tampilan *selected case*

- f. Klik menu *Adaption* bila proses adaptasi dibutuhkan.  
 g. Klik menu *Incorporate New Case* jika hasil adaptasi ingin disimpan (retain) pada database KOMPLAIN.

Selanjutnya untuk mengetahui apakah nilai kemiripan yang dihasilkan valid sesuai rumus similarity 2-1, yang ada pada landasan teori maka dilakukan perhitungan manual sebagai berikut:

$$\begin{aligned}
 \text{similarity (T,S)} &= \sum_{i=1}^n \frac{f(T_i, S_i) \times W_i}{W_i} \\
 &= \frac{(0,67 \times 1) + (1 \times 2) + (1 \times 2) + (0,33 \times 2) + (1 \times 2)}{(1+2+2+2+2)} \\
 &= \frac{7,34}{9} \\
 &= 0,815 \\
 &= 81 \%
 \end{aligned}$$

Pada *type of feature matching partial word (case indifferent)* perhitungan nilai kemiripan dilakukan dengan membagi jumlah kata yang sama (antara dua domain yang

dicocokkan) dengan jumlah keseluruhan kata (domain dengan keseluruhan kata yang lebih kecil dari pada domain yang dicocokkan). Nilai 0,67 merupakan nilai kemiripan domain *feature* jenis pelumas antara *new case* dengan *past case* dimana memiliki dua kata yang sama dari tiga jumlah kata. Nilai 1 merupakan nilai kemiripan domain *feature* status pelumas, fisik pelumas, dan trouble dimana keseluruhan kata adalah sama. Nilai 0,333 merupakan nilai kemiripan domain *feature* kinerja mesin dimana terdapat satu kata yang sama dari tiga susunan kata. Perhitungan manual selengkapnya dapat dilihat pada lampiran 4. Pencocokan domain *feature* antara kasus lama dengan kasus baru disajikan pada Table 5.5.

Tabel 5.5 Pencocokan domain *feature*

<i>Target case</i>	<i>New case 1</i>	<i>Past case 1</i>	Kemiripan domain <i>feature</i> ( $f(T_i, S_i)$ )	Pembobotan ( $W_i$ )	<i>Similarity value</i> ( $f(T_i, S_i) \times W_i$ )
<b>Jenis Pelumas</b>	2 Stroke engine	4 Stroke engine	0,67	1	0,67
<b>Status Pelumas</b>	<i>In service</i>	<i>In service</i>	1	2	2
<b>Fisik Pelumas</b>	Ok	Ok	1	2	2
<b>Kinerja Mesin</b>	Bunyi mesin kasar	Mesin mengalami gagal starter	0,333	2	0,67
<b>Trouble</b>	Terdapat pengotor pada filter oli	Terdapat pengotor pada filter oli	1	2	2
<b>Total value</b> ( $f(T_i, S_i) \times W_i / \sum W_i$ )(%)					81,5 $\approx$ 81

Berdasarkan perhitungan nilai kemiripan secara manual dapat disimpulkan bahwa nilai kemiripan total yang dihasilkan Aplikasi Penanganan Komplain Pelumas sama dengan nilai kemiripan total perhitungan manual sehingga aplikasi Aplikasi Penanganan Komplain Pelumas dinyatakan valid

### 5.3.3 Uji *Prototype*

Uji *prototype* bertujuan untuk menguji *prototype* yang dibuat apakah sudah sesuai dengan kebutuhan pengguna. Kebutuhan Pengujian *prototype* Aplikasi Penanganan Komplain Pelumas dilakukan dengan melihat karakteristik aplikasi yang *user friendly* atau tidak. *User friendly* dapat diartikan bahwa aplikasi dapat dengan mudah digunakan oleh user. Gambar 5.14 menjelaskan tampilan *Case Editor* untuk kebutuhan proses input data kedalam database kasus.



The screenshot shows a window titled "Case Editor" with a menu bar containing "Current Case-Base: KOMPLAIN", "Case Name: Newcase", "Save Case", and "Get Case". Below the menu bar are buttons for "Case Modified" and "Clear Case". The main area is a table with two columns: "Feature Names" and "Feature Values".

Feature Names	Feature Values
NamaCustomer	
Alamat	
JenisPelumas	
StatusPelumas	
FisikPelumas	
KinerjaMesin	
Trouble	
Solusi	

Gambar 5.14 Tampilan untuk input kasus ke dalam database KOMPLAIN

Tampilan untuk input kasus baru diperlihatkan pada Gambar 5.15. Tampilan hasil penelusuran (*retrieve*) kasus lama beserta nilai kemiripannya oleh Aplikasi Penanganan Komplain Pelumas disajikan pada Gambar 5.16

The screenshot shows a window titled "Enter Target Case" with a form for entering case details. The form has the following fields:

- NamaCustomer
- Alamat
- JenisPelumas
- StatusPelumas
- FisikPelumas
- KinerjaMesin

Gambar 5.15 Tampilan *interface* untuk input kasus baru

The screenshot shows a window titled "Retrieved Case List" with a table of retrieved cases. The table has the following columns: Score, Case Name, NamaCustomer, and StatusKomplain.

Score	Case Name	NamaCustomer	StatusKomplain
93	case8	Sarang Mas Sejahtera	Justified
93	case12	Kemilau NS	Justified
93	case19	Sumber Jaya	Justified
93	case20	PT. Sinar Mas Abadi	Unjustified
74	case21	PT. Armada Segara Perka	Justified
59	case16	Maju	Justified
56	case13	Sinar	Unjustified
48	case15	Mitra	Unjustified
37	case9	Multiplast	Unjustified
33	case3	Aketan	Unjustified

Gambar 5.16 Tampilan hasil *retrieval* yang informatif

Pada tahapan *testing* juga akan dibahas mengenai keuntungan dari penggunaan Aplikasi Penanganan Komplain Pelumas dibandingkan dengan sistem yang digunakan oleh PT. ALP Petro Industry saat ini dalam menanggapi komplain pelanggan saat ini.

Bisa dikatakan dalam menanggapi komplain pelanggan, PT. ALP Petro Industry belum memanfaatkan aplikasi berbasis komputer yang bisa memberikan kemudahan. Keberadaan dokumentasi kasus komplain pelanggan terkait produk pelumas juga belum dimanfaatkan dengan baik. Pada kasus yang bersifat umum dimana

identifikasi fisik pelumas, kinerja mesin, dan *trouble* bisa dilakukan oleh pelanggan tanpa bantuan dari bagian *technical support* dan laboratorium maka adanya *Prototype* Aplikasi Penanganan Komplain Pelumas ini mampu memangkas lama waktu pemberian solusi dan status komplain, karena manager *customer* dapat secara langsung memasukkan variabel pada *prototype* aplikasi dan bisa langsung memunculkan solusi serta status komplain untuk mendukung pengambilan keputusan terkait status komplain baru. Namun untuk kasus-kasus yang bersifat khusus dimana bagian *technical support* dan laboratorium harus melakukan analisis lapangan dan sampel pelumas seperti identifikasi adanya kandungan kontaminan produk lain, maka fungsi dari *prototype* aplikasi ini kurang bisa diandalkan sebagai alat pendukung keputusan untuk memberikan solusi yang cepat. Adapun beberapa perbandingan *prototype* Aplikasi Penanganan Komplain Pelumas dibandingkan sistem metode lama disajikan pada Tabel 5.6.

Tabel 5.6 Perbandingan aplikasi yang dikembangkan dengan sistem saat ini

No	Unsur Pembeding	Metode Lama	Metode baru dengan CBR
1	<b>Kecepatan</b>	Penarikan kesimpulan dilakukan melalui urutan alur berikut: manager <i>customer service</i> , <i>technical support</i> , laboratorium, dan kembali ke manager untuk memberikan kesimpulan. Sehingga kesimpulan diberikan 2 x 24 jam setelah sample pelumas diterima bahkan bisa lebih dari 2 hari.	Pada kasus yang bersifat umum, dimana analisa bisa dilakukan oleh pelanggan, maka dapat dilakukannya penarikan kesimpulan dengan cepat oleh manager atau user dengan memasukkan variabel yang harus diinputkan sesuai dengan keterangan yang didapat dari komplain pelanggan atau laporan <i>technical support</i> dilapangan. Namun pada identifikasi kasus-kasus tertentu khususnya identifikasi unjuk kerja dan <i>trouble</i> , penggunaan aplikasi ini tidak sepenuhnya terlepas dari peran laboratorium dan <i>technical support</i> .
2	<b>Fleksibilitas</b>	Dibutuhkan tenaga ahli dibidang kimia terkait kerja laboratorium dan tenaga ahli yang paham akan keterkaitan antara pelumas dengan kerja mesin yaitu seorang <i>technical support</i> .	Sifat dari kecerdasan buatan yang bisa digandakan dan digunakan oleh siapa saja tanpa harus adanya orang yang ahli dibidang tersebut. Hal ini memungkinkan Aplikasi Penanganan Komplain Pelumas digunakan oleh orang yang bukan ahli di bidang pelumas. Namun pada identifikasi kasus-kasus tertentu khususnya identifikasi unjuk kerja dan <i>trouble</i> , penggunaan aplikasi ini tidak sepenuhnya terlepas dari peran laboratorium dan <i>technical support</i> .



No	Unsur Pemanding	Metode Lama	Metode baru dengan CBR
3	<b>Keajekan</b>	Laboran dan seorang <i>technical support</i> sebagai manusia tidak lepas dari sifat lupa atau kurang ketelitian karena faktor lelah dan yang lainnya, sehingga hasil inkonsisten bisa terjadi.	Mampu memberikan hasil yang konsisten, karena pencarian kesimpulan dilakukan oleh mesin komputer terhadap kasus-kasus lama yang dibakukan dalam database kasus.
4	<b>Kepuasan Customer</b>	Manusia dalam memberi penilaian tidak lepas dari unsur subjektivitas, sehingga sangat dimungkinkan mengurangi nilai kepuasan <i>customer</i> terhadap kesimpulan yang dihasilkan.	Hasil dan kesimpulan yang dihasilkan oleh mesin komputer bersifat objektif. Sehingga kepuasan <i>customer</i> terjamin.

