BAB V PENGEMBANGAN APLIKASI

Pada bab ini akan dijelaskan mengenai tahap dalam mengembangkan aplikasi *Case Based Reasoning* untuk membantu divisi perawatan pada CV Kajeye Food dalam menghasilkan suatu aplikasi *prototype* yang mendukung pelaksanaan pengumpulan kasuskasus *troubleshooting* mesin *Vacuum Frying* dan dapat digunakan untuk mendukung kegiatan perawatan mesin *Vacuum Frying*. Tahapan pengembangan aplikasi yang akan dilakukan yaitu meliputi desain, implementasi, dan pengujian aplikasi.

5.1 Desain

Desain merupakan rancangan awal dari suatu sistem yang dikehendaki. Adapun sistem yang dimaksud pada penelitian ini adalah aplikasi yang dikembangkan untuk mengumpulkan kasus-kasus *troubleshooting* mesin *Vacuum Frying* dengan pendekatan *Case Based Reasoning* yang nantinya dapat digunakan sebagai solusi terhadap perawatan maupun perbaikan mesin *Vacuum Frying*.

5.1.1 Perancangan Struktur Kasus

Pada tahapan ini akan dirancang struktur kasus sebagai dasar pembuatan *database* kasus. Semua fitur yang terlibat dalam pembentukan kasus-kasus tersebut telah diidentifikasi dan dikelompokan sesuai grup masing-masing. Fitur-fitur tersebut memiliki atribut, yaitu Nama Fitur, Jenis Fitur, Tipe Data, dan Domain. Berikut ini adalah nama fitur yang sesuai dengan hasil pengumpulan kasus-kasus *troubleshooting* mesin *Vacuum Frying* pada divisi perawatan CV Kajeye Food.

- 1. Kode Mesin adalah kode yang diberikan untuk komponen mesin *Vacuum Frying* di lantai produksi.
- 2. Nama Komponen Mesin adalah nama komponen mesin pada mesin Vacuum Frying.
- 3. *Trouble* adalah masalah yang dapat diamati dari komponen mesin yang menjadi salah satu input dari diagnosa kerusakan yang terjadi.
- 4. Bunyi adalah bunyi khas yang dikeluarkan oleh komponen mesin yang menjadi salah satu input dari diagnosa kerusakan yang terjadi.

- 5. Getaran adalah getaran pada komponen mesin yang dihasilkan oleh mesin saat kondisi mesin *ON* yang menjadi salah satu input dari diagnosa kerusakan yang terjadi.
- 6. Hasil Gorengan adalah kondisi fisik pada keripik setelah di goreng yang menjadi salah satu input dari diagnosa kerusakan yang terjadi.
- 7. Jenis Kerusakan adalah nama kerusakan yang terjadi pada mesin.
- 8. Penyebab Kerusakan adalah hal yang menjadi penyebab terjadinya kerusakan atau masalah pada komponen mesin.
- 9. Penanganan Kerusakan adalah tindakan penanganan yang diterapkan pada mesin maupun tindakan lain yang dianggap perlu oleh teknisi saat terjadi kerusakan atau masalah pada komponen mesin.

Adapun atribut *feature* pada perancangan struktur kasus *Troubleshooting* Mesin *Vacuum Frying* dapat dilihat pada Tabel 5.1 sebagai berikut :

No	Nama Fitur	Jenis Fitur	Tipe Data	Domain
1	Kode Komponen Mesin	ID Ç	Text	VF01-VF06
2	Nama Komponen Mesin	ID	Text	Semua nama yang valid
3	Trouble	Adjustment	Text	Semua nama yang valid
4	Bunyi	Adjustment	One of a List	Kletak kletak, Grok grok, Swish swish, Kreek kreek, Suish suish, Ngung ngung, Nyiiing nyiiing, Normal, Suara hilang
5	Getaran	Adjustment	One of a List	Keras, Normal, Getaran hilang
6	Hasil Gorengan	Adjustment	Text	Lembek dan tidak gosong, Renyah dan tidak gosong, Lembek dan gosong, Renyah dan gosong, Normal
7	Jenis Kerusakan	Solution	Text	Semua nama yang valid
8	Penyebab kerusakan	Solution	Text	Semua data yang valid
9	Penanganan Kerusakan	Solution	Text	Semua data yang valid

Tabel 5.1 Struktur Kasus Troubleshooting Mesin Vacuum Frying

Kode Komponen Mesin dan Nama Komponen Mesin dikelompokkan kedalam ID *feature*, karena merupakan pengenal yang dapat membedakan jenis komponen. Untuk acuan atau tanda-tanda yang digunakan teknisi dalam mengamati adanya masalah yang terjadi pada mesin adalah *Trouble*, Bunyi, Getaran, dan Hasil Gorengan sehingga dikelompokkan kedalam jenis fitur *adjustment feature*. Penggunaan empat fitur tersebut sebagai acuan teknisi merupakan pengalaman yang terbentuk dari waktu ke waktu. Untuk Jenis Kerusakan, Penyebab Kerusakan, dan Penanganan Kerusakan dikelompokkan pada *solution feature*, karena merupakan solusi yang diterapakan saat terjadi masalah pada mesin.

5.1.2 Pembentukan Basis Kasus

Setelah merancang struktur kasus, maka hasil pengumpulan kasus-kasus tentang *troubleshooting* mesin *Vacuum Frying* diorganisasikan sesuai struktur kasus masingmasing untuk kemudian dibuatkan *database* kasus. Pada tahap ini, fitur-fitur diklasifikasikan ke dalam tiga *main feature* atau fitur induk. Tiga fitur induk yang dimaksud adalah sebagai berikut:

- 1. Fitur Pengenal sebagai ID Feature.
- 2. Fitur Penyesuai sebagai Adjustment Feature.
- 3. Fitur Solusi sebagai Solution Feature.

Penjabaran detail dari *database* kasus dalam penelitian ini ditampilkan pada Tabel 5.2 dan *database* kasus *Troubleshooting* Mesin *Vacuum Frying* secara lengkap dapat dilihat pada lampiran 1.

ID Fe	eature		Adjustr	nent Feature		Solution Feature			
Kode Kompon en Mesin	Nama Kompon en Mesin	Trouble	Bunyi	Getaran	Hasil Gorengan	Jenis Kerusakan	Penyebab kerusakan	Penanganan Kerusakan	
VF01	Pompa Vacuum	Jarum <i>vacuum</i> tidak naik	Kletak Kletak	Keras	Renyah dan Gosong	Impeller Keropos	Terkikis air	Bisa dilakukan pembubutan pada <i>impeller</i> , namun kinerja mesin tidak maksimal, untuk kinerja yang maksimal ganti dengan <i>impeller</i> yang baru	
VF01	Pompa Vacuum	Jarum vacuum tidak naik dan penghisa pan air pada kondenso r tidak berjalan	Grok grok	Keras	Renyah dan Gosong	Impeller patah	Bahan impeller kw 2, sehingga mudah patah	Bongkar pompa, kemudian lakukan pengelasan pada <i>impeller</i>	
VF01	Pompa Vacuum	Jarum vacuum tidak naik	Kletak Kletak	Normal	Renyah dan Gosong	<i>Bearing</i> pompa macet	<i>Bearing</i> nyangkut dengan dinding pompa	Ganti dengan bearing yang baru, bisa juga dilakukan pengelasan pada bearing	
VF02	Tabung Penggore ngan	Tuas pengaduk longgar	Swish swish	Normal	Lembek dan Tidak gosong	<i>Seal</i> poros tabung longgar	Pemakaian lebih dari 3 bulan menyebabkan aus atau retak pada <i>seal</i> poros tabung	Ganti <i>seal</i> poros dengan yang baru, untuk selanjutnya lakukan penggantian setiap 3 bulan	
VF02	Tabung Penggore ngan	Ada retakan pada sambung an tabung	Normal	Normal	Lembek dan Tidak gosong	Tabung penggorengan bocor	Korosi akibat uap panas berlebih pada tabung penggorengan	Lakukan pengelasan pada kebocoran	

Tabel 5.2 Database Kasus Troubleshooting Mesin Vacuum Frying

5.1.3 Pendefinisian Kemiripan (Similarity) dan Pembobotan

Kemiripan atau disebut juga *similarity* adalah dasar dari pendekatan *Case Based Reasoning* yang merupakan deskripsi mengenai miripnya satu kasus dengan kasus yang lain. Dalam metode CBR, tahap *retrieve* dilakukan dengan melakukan penelusuran kemiripan antara kasus baru (*new case*) dengan kasus lama (*past cases*) yang telah disimpan. *Nearest neighbor* merupakan salah satu teknik *retrieval* yang digunakan untuk mencari nilai kemiripan. *Tool* ESTEEM 1.4 sendiri menerapkan teknik *nearest neighbor* yang lebih dispesifikkan sesuai dengan tipe data yang digunakan. Misalnya untuk tipe data *text*, mencari nilai kemiripan bisa dilakukan dengan menggunakan pencocokan domain atau nilai fitur (*Type of feature matching*) jenis *exact*, *partial match*, dan *partial words*.

Teknik *nearest neighbor* lebih efektif dengan pemberian bobot pada fitur-fitur penyesuai pada setiap kelompok pengetahuan. Pembobotan ini didasarkan pada seberapa penting suatu fitur dibandingkan dengan fitur lainnya untuk memberikan solusi. Pada Tabel 5.3 berisi uraian pertimbangan pemberian bobot pada *adjustment feature*.

		Tabel 5.3 Pembobotan Pada Fitur Penyesuai
Nama Fitur	Bobot	Pertimbangan
Trouble	3	<i>Trouble</i> merupakan fitur yang penting dalam pengetahuan ini. Pada saat mesin mengalami masalah atau gangguan terdapat <i>trouble</i> mesin yang dapat diamati secara langsung oleh teknisi. Hal ini dapat dijadikan acuan penting dalam mendiagnosa masalah atau gangguan yang terjadi.
Bunyi	2	Ketika terjadi masalah atau gangguan pada mesin, seringnya komponen mesin yang bermasalah mengeluarkan bunyi khas yang dapat menjadi acuan penting dalam mendiagnosa masalah atau gangguan yang terjadi.
Getaran	1	Pengukuran getaran dilakukan dengan menggunakan tangan atau pengamatan secara langsung dan tidak ada alat ukur getaran yang digunakan saat ini. Hal ini membuat fitur ini tidak signifikan dalam mendiagnosa masalah atau gangguan yang terjadi.
Hasil Gorengan	1	Pengamatan terhadap hasil gorengan dilakukan dengan mengamati dari kaca yang terdapat pada tabung penggorengan, namun sulit menentukan hasil gorengan secara akurat melalui kaca tabung penggorengan. Hal ini membuat fitur ini tidak signifikan dalam mendiagnosa masalah atau gangguan yang terjadi.

5.1.4 Pendefinisian Rule Adaptasi

Adaptasi pada CBR merupakan proses penyesuaian *feature* antara kasus baru dengan kasus lama. Adaptasi juga dapat berfungsi untuk membatasi keadaan tertentu dan mengubah *solution feature* pada kasus lama agar sesuai dengan kondisi kasus baru. Didalam CBR, proses adaptasi terdapat pada tahapan *revise*. Pengembangan aplikasi

prototype ini menggunakan *rule* untuk proses adaptasi, ketika kasus baru dengan nilai kemiripan yang tinggi akan disimpan kedalam *database* kasus. Dalam melakukan pendefinisian *rule* adaptasi, secara umum dapat dituliskan sebagai berikut:

```
If
   true
Then
   xSC:y=xTC:y
```

Dimana,

```
x = Case Base Name
y = Feature Names
SC = Selected Case
TC = Target Case
```

AATrouble adalah salah satu pendefinisian *rule* yang dibuat untuk mengadaptasi domain kasus baru dengan domain kasus lama. Jadi *rule* AATrouble berfungsi mengubah domain *feature Trouble* pada kasus lama yang terpilih, dengan domain *Trouble* pada kasus baru. Pendefinisian *rule* adaptasi seluruhnya disajikan pada lampiran 2. Adapun *rule* AATrouble adalah sebagai berikut :

asitas BRAM

```
If
true
Then
CB TRBSHTSC:Trouble=CB TRBSHTTC:Trouble
```

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

5.2 Implementasi

Implementasi merupakan tahapan dimana teori dan desain perancangan yang telah dibuat diterapkan dalam pembuatan aplikasi berbasis komputer. Pada tahapan ini penerapan aplikasi memanfaatkan teknologi CBR untuk menerapkan rancangan yang telah dibuat pada tahap sebelumnya. Teknologi CBR yang digunakan yaitu *tool* ESTEEM 1.4 yang dikeluarkan oleh Stottler Henke Associates, Inc. *Tool* ESTEEM 1.4 ini sangat menunjang dalam pembuatan aplikasi *Case Based Reasoning* untuk *troubleshooting* mesin *Vacuum Frying*. Adapun aplikasi *prototype* yang dikembangkan pada penelitian ini diberi nama Aplikasi *Troubleshooting* Mesin *Vacuum Frying* (ATM *Vacuum Frying*).

BRAWIJAYA

5.2.1 Penyimpanan Kasus-Kasus Troubleshooting dengan Tool ESTEEM 1.4

Kasus-kasus yang sudah dikumpulkan dalam bentuk *database* kasus seperti pada tabel 5.2 disimpan ke dalam *tool* ESTEEM 1.4. Tidak hanya berfungsi sebagai memori penyimpan kasus-kasus yang telah dikumpulkan, tetapi juga pemroses aktivitas *retrieval* yang akan dimanfaatkan oleh teknisi dalam menggunakan kasus-kasus yang telah tersimpan.

5.2.2 Pengaktifkan Tool ESTEEM 1.4

Dalam pembuatan aplikasi *prototype* menggunakan *tool* ESTEEM 1.4. Langkahlangkah yang dilakukan untuk mengaktifkan *tool* ESTEEM 1.4 adalah sebagai berikut:

- 1. Membuka *folder* tempat *file tool* ESTEEM 1.4 disimpan.
- 2. Mencari file dengan nama ESTEEM 1.4 dengan tipe file windows batch file.
- 3. Memunculkan tampilan *interface* dari *tool* ESTEEM 1.4 dengan *double* klik *file* ESTEEM 1.4 seperti pada gambar 5.1.



5.2.3 Pembuatan Case Base

Pembuatan *database* kasus pada *tool* ESTEEM 1.4 ini dimulai dengan pembuatan *Case Base* baru pada *Case Base Definition Editor*. Kemudian, editor ini digunakan untuk mendeklarasikan komponen dasar dari sistem yaitu *feature name* dan *feature type value*. *Feature name* merupakan nama-nama fitur yang telah ditetapkan pada struktur kasus, sedangkan *feature type value* merupakan tipe nilai yang akan diberikan pada *feature* sesuai dengan referensi dari struktur kasus. Tipe-tipe *feature* yang terdapat dalam *tool* ESTEEM 1.4 ini adalah sebagai berikut:

• Yes or No

• One of a List

- Text
- Numeric

- Case
- Multimedia

Adapun langkah-langkah yang dilakukan dalam pembuatan *database* kasus pada *tool* ESTEEM 1.4 adalah sebagai berikut:

- Membuka tampilan Case Base Definition Editor dengan memilih ikon Case Base Definition, kemudian mengklik New, dan selanjutnya mengetikkan CB_TRBSHT pada Case Base Name. Tampilan Case Base Definition Editor dapat dilihat pada Gambar 5.2.
- 2. Mengetikkan nama fitur pada kolom *feature name*, sesuai pada perancangan struktur kasus.
- 3. Menentukan tipe data pada kolom *feature value type*, sesuai pada perancangan struktur kasus.
- 4. Kemudian menyimpan *database* yang telah dibuat dengan mengklik tombol *Exit*. Tampilan *Case Base Definition Editor* untuk *troubleshooting* mesin dapat dilihat pada

Gambar 5.2.

Ҟ Case Base Definition Editor				х
Current Case-Base: CB_TF	RBSHT C	urrent Similarity Definition: SD	_TRBSHT	
Feature Names	Feature Value Types	Feature Names	Feature Value Types	
Kode_komponen_mesin	Text	Jenis_kerusakan	Text	-
Nama_komponen_mesin	Text	Penyebab_kerusakan	Text	
Trouble	Text	Penanganan_kerusakan	Text	1
Bunyi	One of a List	· ·		1
Getaran	One of a List			1
Hasil gorengan	Text			

Gambar 5.2 Case Base Definition Editor Untuk Troubleshooting Mesin

5.2.4 Pendefinisian Kemiripan

Pendefinisian Kemiripan merupakan tahap untuk menentukan tipe dari kecocokan fitur dan pembobotan fitur dilakukan pada *Similarity Definition Editor*. Editor ini merupakan editor yang mendeklarasikan bagaimana kasus-kasus akan dicari kembali (*retrieved*) berdasarkan pada *Case-base definition*. *Similarity Definition Editor* ini mendeklarasikan metode dan berbagai macam ukuran untuk menjelaskan keserupaan selama pencarian kembali (*retrieval*).

Adapun langkah-langkah yang dilakukan dalam mendefinisikan nilai kemiripan dan pembobotan adalah sebagai berikut:

1. Membuka tampilan *Similarity Definition Editor* dengan memilih ikon *Similarity Definition*, kemudian mengklik *New*, dan selanjutnya mengetikkan SD_TRBSHT pada

Similarity Definition Name. Tampilan Similarity Definition Editor dapat dilihat pada Gambar 5.3.

- 2. Memunculkan pilihan dengan mengklik kanan pada kolom *Type of Similarity*. Pada penelitian ini menggunakan *Weighted Feature Computation*.
- 3. Mengetikkan nilai *threshold* pada kolom *threshold* (batas terendah nilai kemiripan total yang ditampilkan). Pada penelitian ini menggunakan *threshold* sebesar 60%.
- 4. Memberi tanda *check* ($\sqrt{}$) pada *adjusment feature* dan menentukan jenis pencocokan pada kolom *Type of feature matching*.
- 5. Memasukkan nilai pembobotan *feature* pada kolom *Weight/Rule Base Name* sesuai pada tabel 5.3.
- 6. Kemudian menyimpan data yang telah dibuat dengan mengklik tombol *Exit*.

Tampilan *Similarity Definition Editor* untuk *troubleshooting* mesin dapat dilihat pada Gambar 5.3.

rrent C	ase-Base: CB_TRBSHT Cu	rent Similarity Definition: SD_T	RBSHT Threshold 60
-	utomatic Weight Generation	Type of Similarity: Weig	ghted Feature Computation
elected	Feature Name	Type of Feature Matchin	g Weight/Rule Base Name
	Kode_komponen_mesin		
	Nama_komponen_mesin		
~	Trouble	Partial (case indifferent)	3
~	Bunyi	Exact	2
	Getaran	Exact	1
	Hasil_gorengan	Partial (case indifferent)	1

Gambar 5.3 Similarity Definition Editor Untuk Troubleshooting Mesin

Pada Similarity Definition Editor mempunyai lima daerah penting yaitu:

- 1. Selected Button atau tombol paling kiri dari editor digunakan untuk memberitahu sistem, feature mana saja yang digunakan untuk perhitungan similarity. Dalam hal ini, semua fitur yang tergolong dalam fitur penyesuai (adjustment feature) akan diberi tanda centang ($\sqrt{}$) agar dihitung kemiripannya ketika proses retrieval dilakukan.
- 2. *Type of Feature Matching* digunakan untuk mendeklarasikan persamaan apa yang digunakan pada *feature* yang dipilih, hal ini juga tergantung dari tipe *feature* itu sendiri. Dari tipe kecocokan fitur ini akan diketahui berapa nilai kembalian yang akan disediakan oleh sistem untuk dikalikan dengan nilai bobot yang dideklarasikan sehingga menghasilkan skor kemiripan pada akhir *retrieval*.

Pada penelitian ini menggunakan Partial (case indifferent) dan Exact. Untuk Partial (case indifferent), perhitungan nilai kemiripan antara new case dengan past case dilakukan dengan membagi jumlah kata yang sama dengan jumlah keseluruhan kata. Sedangkan untuk Exact, perhitungan nilai kemiripan dilakukan dengan melihat sama atau tidaknya data antara new case dengan past case. Jika sama bernilai 1 dan jika tidak sama bernilai 0, jika ada sedikit bagian yang berbeda, maka hal tersebut dianggap tidak sama dan bernilai 0.

- 3. Weight/Rule Base Name digunakan untuk mendeklarasikan bobot dimana bobot ini menunjukkan derajat kepentingan suatu *feature* dalam proses *retrieval*. Rule-base name merupakan penunjuk pada kumpulan *rule* yang digunakan untuk mendeklarasikan bagaimana pentingnya suatu *feature* itu pada proses *retrieval*. Nilai bobot bisa langsung dituliskan sesuai dengan nilainya, bisa juga dengan infrensi dari *rules* yang telah ditetapakan.
- 4. *Field Type of Similarity* digunakan untuk mendeklarasikan teknik *similarity* apa yang akan digunakan. Tipe *Similarity* yang digunakan ada tiga, dapat dipilih dengan cara mengklik kanan mouse. Tiga tipe *similarity* yang digunakan adalah :
 - a. Feature Counting
 - b. Weighted Feature Computation
 - c. Inferred Feature Computation

Pada penelitian ini menggunakan Weighted Feature Computation

5. *Threshold* digunakan untuk menginformasikan ESTEEM 1.4 sistem tingkat kecocokan yang diterima untuk ditampilkan pada pemakai. *Threshold* ditampilkan sebagai presentasi dari *similarity*. *Similarity* dideklarasikan pada pemakai sebagai angka 0 sampai 100, dengan 0 menjadi *similarity* terlemah dan 100 yang terkuat. *Threshold* merupakan nilai minimal yang akan membatasi jumlah kasus yang cocok atas dasar skor kemiripan yang telah ditetapkan.

Agar kasus lama yang ditampilkan tidak terlalu banyak dan agar tidak menampilkan kasus lama yang tidak relevan atau tingkat akurasi yang rendah dengan kasus baru, maka pada penelitian ini menggunakan *threshold* 60% karena dianggap nilai minimal *score* yang cukup mewakilkan data kasus lama untuk ditampilkan pada saat proses *retrieval*.

5.2.5 Proses Input Data Kasus Troubleshooting

Untuk mengisikan data dari suatu kasus, atau untuk memindahkan kasus yang ada pada *case base* kedalam aplikasi digunakan *Case Editor*. Setiap pengisian suatu kasus harus diberi nama atau karakter berbeda dengan *case* yang lain.

Adapun langkah-langkah yang dilakukan untuk meng-*input*-kan data ke dalam *Case Editor* adalah sebagai berikut:

- 1. Membuka tampilan *Case Editor* dengan memilih ikon *Cases*, kemudian mengklik *New*, dan selanjutnya mengetikkan nama kasus pada kolom *Case Name*, misalkan pada kasus pertama mengetikkan CASE_01, kasus kedua mengetikkan CASE_02, dan seterusnya. Tampilan *Case Editor* dapat dilihat pada Gambar 5.4.
- 2. Meng-input-kan data kasus-kasus yang telah dikumpulkan pada kolom Feature Value.
- 3. Kemudian mengklik Save Case bila feature value sudah selesai diisikan dengan lengkap.

Tampilan Case Editor untuk troubleshooting mesin dapat dilihat pada Gambar 5.4.

Current Case-Base: CB_TI	RBSHT	Case Name:	CASE	_01	Save Case	Get Case
Feature Names	Feature	Values		Case Saved	Clear	Case
Kode_komponen_mesin	VF01		-			<u>k</u>
Nama_komponen_mesin	Pompa	Vacuum			4	
Trouble	Jarum v	vacuum tidak na	ik		4	
Bunyi	Kletakk	letak				
Getaran	Keras				10	
Hasil_gorengan	Renyah	dan Gosong			4	
Jenis_kerusakan	Impelle	r keropos			4	
Penyebab_kerusakan	Terkikis	air			4	
Penanganan_kerusakan	Bisa dil kinerja maksim	akukan pembut mesin tidak ma al ganti dengan	outan p ksimal n impel	oada impeller, , untuk kinerja ller yang baru	namun 1 yang	

Gambar 5.4. Case Editor Untuk Troubleshooting Mesin

5.2.6 Penggunaan Rules dan Rule-Base

Dalam *tool* ESTEEM 1.4, *rules* mempunyai fungsi, pertama, menghitung kemiripan dengan mendeklarasikan bobot yang digunakan dalam proses *retrieval*. Kedua, untuk melakukan adaptasi pada kasus yang terpilih (*selected case*) untuk lebih memenuhi kebutuhan kasus baru (*target case*). Pada penelitian ini, penggunaan *rule* dibatasi hanya pada fungsi yang kedua dan hanya pada *adjustment feature* saja.

Adapun langkah-langkah yang dilakukan dalam implementasi *rule* untuk adaptasi adalah sebagai berikut:

- Membuka tampilan *Rule Editor* dengan memilih ikon *Rules*, kemudian mengklik *New*, dan selanjutnya mengetikkan nama *rule* pada kolom *Rule Name*, contohnya AATrouble untuk fitur Trouble. Tampilan *Rule Editor* dapat dilihat pada Gambar 5.5.
- 2. Mengetikkan true pada kolom If:, dan selanjutnya mengetikkan CB_TRBSHTSC:Trouble=CB_TRBSHTTC:Trouble pada kolom Then:.
- 3. Kemudian menyimpan data yang telah dibuat dengan mengklik tombol *Exit*.

Detail pendeklarasian *rule* adaptasi AA*Trouble* ditampilkan pada Gambar 5.5 dan untuk *rule* dari *adjustmen feature* yang lain dilampirkan pada lampiran 2.

Kule Editor			×
Current Rule: AATrouble			
lf:			
true			
Then			
CB_TRBSHTSC:Trouble=CB_TRBSHT	TC:Trouble		
Gambar 5 5 Deklara	si Rule Pada Tre	whleshooting N	lesin

Agar kasus dapat dilakukan adaptasi oleh *user* maka dibutuhkan *rule* yang berfungsi mengubah data kasus lama. Misalkan pada *rule* AA*Trouble* yang berfungsi mengubah domain *feature Trouble* pada kasus lama yang terpilih, dengan domain *Trouble* pada kasus baru. Dengan adanya rule ini, *user* hanya mengklik menu *adaptation* maka dapat dilakukan proses adaptasi oleh *prototype*.

Proses selanjutnya memasukkan *rule* yang telah dibuat ke dalam *Rule Base*. Langkahlangkah yang dilakukan untuk memasukkannya adalah sebagai berikut:

- Membuka tampilan *Rule Base Editor* dengan memilih ikon *Rule Base*, kemudian mengklik *New*, dan selanjutnya mengetikkan ADAPTASI pada *Rule Base Name*. Tampilan *Rule Base Editor* dapat dilihat pada Gambar 5.6.
- 2. Mengetikkan *Rule Name* yang telah dibuat sebelumnya pada kolom *Listing of Rules in Rule Base* ADAPTASI.
- 3. Kemudian menyimpan data yang telah dibuat dengan mengklik tombol *Exit*.
- Tampilan *Rule Base Editor* untuk *troubleshooting* mesin dapat dilihat pada Gambar 5.6.

🕱 Rule Base Editor	
Current Rule Base: ADAPTASI	
Listing of Rules in Rule Base ADAPTASI	
AKode_komponen_mesin AANama_komponen_mesin ATrouble ABunyi AAGetaran AAHasil_gorengan AJenis_kerusakan APenyebab_kerusakan APenanganan_kerusakan	

Gambar 5.6 Rule Base Editor Untuk Troubleshooting Mesin

Rule Base sendiri digunakan untuk mengelompokkan *rule-rule* yang dibutuhkan pada proses *auto adaptation*. Kemudian pengelompokkan ini nantinya akan memudahkan dalam peng-*input*-an pada *Additional End User Fuctionality*

5.2.7 Pembuatan User Interface

Tahap ini adalah tahap pembuatan *user interface* untuk *tool* ESTEEM 1.4 dengan memanfaatkan *End User Interface Editor*, dimana *user* dapat meng-*input*-kan data kasus baru lalu mencari solusi atas dasar kemiripan dengan kasus lama. *End User Interface Editor* merupakan editor yang digunakan untuk memilih *feature* apa saja yang ingin ditampilkan pada saat aplikasi dijalankan. Pada saat *execute* akan ada tiga *window* yang muncul, yaitu *Target Case Entry Features*, *Retrieved Case Features*, *dan Selected Case Features*.

Adapun langkah-langkah yang dilakukan untuk memilih *feature* apa saja yang ingin ditampilkan aplikasi dalam *End User Interface Editor* adalah sebagai berikut:

- Membuka tampilan End User Interface Editor dengan memilih ikon End User Interface, maka akan ada tiga window yang muncul pada End User Interface Editor, yaitu Target Case Entry Features, Retrieved Case Features, dan Selected Case Features. Tampilan End User Interface Editor dapat dilihat pada Gambar 5.7.
- 2. Memilih *features* apa yang ingin ditampilkan saaat aplikasi dijalankan dengan mengklik *feature*. *Feature* yang telah dipilih akan tampil pada masing-masing layar saat interaksi dengan *user*.

Tampilan *End User Interface Editor* untuk *troubleshooting* mesin dapat dilihat pada Gambar 5.7.

Current Case-Base: CB_TRBSHT	Current Similarity Definition:	Additional End User Functionality
Target Case Entry Features	Retrieved Cases Features	Selected Case Features
Kode_komponen_mesin	Kode_komponen_mesin	Kode_komponen_mesin
Nama_komponen_mesin	Nama_komponen_mesin	Nama_komponen_mesin
Trouble	Trouble	Trouble
Bunyi	Bunyi	Bunyi
Getaran	Getaran	Getaran
Hasil_gorengan	Hasil_gorengan	Hasil_gorengan
Jenis_kerusakan	Jenis_kerusakan	Jenis_kerusakan
Penyebab_kerusakan	Penyebab_kerusakan	Penyebab_kerusakan
Penanganan_kerusakan	Penanganan_kerusakan	Penanganan kerusakan
	276 255	

Gambar 5.7 End-User Interface Editor Untuk Troubleshooting Mesin

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

- a. *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 : *Trouble*, Bunyi, Getaran, dan Hasil gorengan.
- b. 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 komponen mesin dan Jenis kerusakan.
- c. 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 dimanfaat 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.

Dalam End User Interface Editor ini juga terdapat tombol Additional End User Functionally yang berfungsi untuk mendeklarasikan aspek penting dari End User Interface.

Langkah-langkah penyesuaian yang dilakukan pada *Additional End User Functionally* adalah sebagai berikut:

1. Membuka tampilan Additional End User Fuctionality dengan mengklik tombol Additional End User Fuctionality pada End User Interface Editor. Maka terlihat tampilan Additional End User Fuctionality seperti pada Gambar 5.8.

- 2. Membuka pilihan pada kolom *Allow user to add cases to case-base?* dan memilih *Yes.* Hal ini bertujuan untuk menyimpan hasil adaptasi kedalam *database* kasus (tahapan *retain*).
- 3. Membuka pilihan pada kolom *Allow auto-adaption of retrieved case?* dan memilih *Yes*.
- 4. Membuka pilihan pada kolom *Name of rule base for auto-adaption* dan memilih ADAPTASI yang merupakan *Rule Base Name* yang telah dibuat agar dapat melakukan *auto adaptation*.
- 5. Kemudian menyimpan data yang telah dibuat dengan mengklik tombol *OK*. Tampilan *Additional End User Fuctionality* dapat dilihat pada Gambar 5.7.

Allow user to add cases to case-base?:	Yes	
Allow user to modify the similarity definition?:	No	
Allow auto-adaption of retrieved cases?:	Yes	
lame of rule base for auto-adaption:	ADAPTASI	
Name of bitmap file with application information:		
Application Startup Function:		
Selected Case Editor Startup Function:		
Down Button Function:		

Gambar 5.8 Additional End User Functionality Pada Troubleshooting Mesin

Allow user to add cases to case-base ? digunakan untuk diperbolehkan atau tidaknya user untuk memasukkan kasus baru kedalam case-base, pada penelitian ini user diizinkan menambah kasus pada case-base, karena semakin banyak kasus pada aplikasi ini, maka tingkat kecerdasannya semakin tinggi.

Allow user to modify the similarity definition ? digunakan untuk diperbolehkan atau tidaknya user mengubah atau memodifikasi inputan pada similarity definition, pada penelitian ini user tidak diizinkan mengubah similarity definition, karena nilai similarity membutuhkan akurasi yang dimiliki teknisi ahli.

Allow auto-adaptation of retrieved cases ? digunakan untuk diperbolehkan atau tidaknya sistem melakukan auto adaptation terhadap kasus lama mengalami pembaharuan dari kasus baru.

Name of rule base for auto-adaption digunakan untuk meberitahu nama rule base yang dijadikan sebagai acuan rule dalam melakukan auto adaption.

5.3 Pengujian Aplikasi

Pengujian sistem merupakan tahap akhir dari perancangan sistem *Case Based Reasoning*. Pengujian ini bertujuan untuk mengetahui apakah aplikasi CBR dapat berjalan dengan benar dan sesuai harapan. Proses uji dilakukan dengan menerapkan konsep *four-RE*'s pada siklus CBR.

5.3.1 *Retrieve*

Tahap ini bertujuan untuk mengetahui apakah aplikasi CBR ini sudah berjalan sesuai dengan rancangan atau tidak. Maka dari itu kita perlu melakukan uji coba *retrieve* pada *tool* ESTEEM 1.4. Dibutuhkan suatu kasus baru untuk menguji jalannya fungsi *retrieve* pada *tool* ESTEEM 1.4. Kasus baru yang dibuat dengan rincian tertera pada Tabel 5.4.

			Adjusment	Feature	
No	•	Trouble	Bunyi	Getaran	Hasil Gorengan
1		Tuas pengaduk longgar	Normal	Normal	Lembek dan Tidak gosong

Tabel 5.4 Kasus Baru Pada Troubleshooting Mesin

Langkah berikutnya, yaitu menjalankan *tool* ESTEEM 1.4 dengan penjabaran sebagai berikut:

a. Menampilkan jendela *Target Case Entry* dengan memilih tombol *RUN*, maka akan tampil seperti pada Gambar 5.9.

Change Retrieval Attributes	Retrieve	Adaptation	Incorporate New Case	Help	Print	Exit
K Enter Target Case						X
Trouble Bunyi Getaran Hasil_gorengan					* *	

b. Memasukkan nilai dari *feature* kasus baru seperti yang ditampilkan pada tabel 5.4 ke dalam *Target Case Entry*, maka akan tampil seperti pada Gambar 5.10.

etrieve	Adaptation	Incorporate New Case	Help	Print	
		the second s	TICIP	1 min	EXIC
				_ 6	a X
Tuas pengaduk longgar					
Normal					
Normal					
Lembek dan Tidak gosong					
	as peng ormal ormal mbek da	as pengaduk longga rmal rmal mbek dan Tidak gos	as pengaduk longgar ormal ormal mbek dan Tidak gosong	as pengaduk longgar rmal rmal mbek dan Tidak gosong	as pengaduk longgar 🛃 🛃 rmal rmal mbek dan Tidak gosong 🛃

Gambar 5.10 Target Case Entry Kasus Baru

c. Kemudian memunculkan *Retrieved Case List* dengan mengklik tombol *Retrieve* yang menampilkan daftar kasus lama dengan *score* kemiripan (*similarity*) tertinggi pada urutan teratas, maka akan tampil seperti pada Gambar 5.11.

K Retrie	eved Case List		_ _ ×
Score	Case Name	Nama_komponen_mesin	Jenis_kerusakan
71	CASE_05	Tabung penggorengan	Seal poros tabung longg
71	CASE_09	Tabung penggorengan	Pakram kering
69	CASE_08	Tabung penggorengan	Pakram longgar/aus
65	CASE_06	Tabung penggorengan	Tabung penggorengan boc
64	CASE_03	Pompa vacuum	Mechanic Seal Pompa Mac
64	CASE_07	Tabung penggorengan	Seal penutup tipis
62	CASE_12	Kondensor	Kebocoran⁄retak pada ba
61	CASE_13	Steam	Lubang api stean tersun
60	CASE_15	Steam	Tempat pembakaran steam

Gambar 5.11 Retrived Case List

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 *One of a List* dan *text*. Sedangkan pada penelitian ini nilai kemiripan antar domain dalam satu *feature* dikelompok 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 domian kasus lama. Kemiripan rentang didefinisikan ketika domain pada kasus baru dan kasus lama tidak sama mutlak, akan tetapi masih memiliki kemiripan. Tidak adanya kemiripan didefinisikan ketika domain dalam satu *feature* antara kasus baru dengan kasus lama tidak memiliki kemiripan.

H	iER2LaTA2	Kasus Baru				
	Parameter	Tuas pengaduk longgar	Normal	Normal	Lembek dan Tidak gosong	
la	Tuas pengaduk longgar	1	0	0	0	
Lan	Swish swish	0	0	0	0	
asus	Normal	0	0	1	0	
K	Lembek dan Tidak gosong	0	0	0	1	

Tabel 5.5 Nilai Kemiripan Adjustment Feature

Pada *type of feature matching Partial* (*case indifferent*) perhitungan nilai kemiripan dilakukan dengan membagi jumlah kata yang sama (antara dua domain yang dicocokan) dengan jumlah keseluruhan kata (domain dengan keseluruhan kata yang lebih kecil dari pada domain yang dicocokan). Nilai 1 merupakan nilai kemiripan domain *feature* Trouble dan Hasil Gorengan antara *new case* dengan *past case* dimana memiliki keseluruhan kata yang sama pada masing-masing domain *feature*.

Untuk *type of feature matching Exact* perhitungan nilai kemiripan dilakukan dengan melihat sama atau tidaknya data antara *new case* dengan *past case*. Jika sama bernilai 1 dan jika tidak sama bernilai 0, jika ada sedikit bagian yang berbeda, maka hal tersebut dianggap tidak sama dan bernilai 0. Untuk domain *feature* Bunyi bernilai 0 karena data yang diinputkan pada kasus baru berbeda dengan data pada kasus lama dan untuk domain *feature* Getaran bernilai 1 karena data yang diinputkan pada kasus lama.

Selanjutnya untuk mengetahui proses perhitungan *score* secara manual menggunakan rumus similarity

similarity (T,S) = $\sum_{i=1}^{n} \frac{f(T_i,S_i)x W_i}{W_i}$ = $\frac{(1 \times 3) + (0 \times 2) + (1 \times 1) + (1 \times 1)}{(3 + 2 + 1 + 1)}$ = $\frac{5}{7}$ = 0,714 = 71 %

Keterangan:

- T : Kasus baru
- S : Kasus yang ada dalam penyimpanan
- n : Jumlah atribut dalam setiap kasus
- i : Atribut individu antara 1 s.d n
- f : Fungsi *similarity* atribut *i* antara kasus T dan kasus S
- w: Bobot yang diberikan pada atribut ke-i

Score ini yang nantinya akan dijadikan acuan user untuk membantu dalam memberi informasi seberapa dekat kemiripan kasus baru tersebut dengan kasus lama.

5.3.2 Reuse

Setelah mendapatkan daftar data-data kasus lama yang mirip dengan kasus baru, kemudian memilih salah satu kasus lama untuk digunakan sebagai solusi terhadap permasalahan kasus baru. Dapat dilihat pada Gambar 5.12.

Kode_komponen_mesin	VF02	±
Nama_komponen_mesin	Tabung penggorengan	±
Trouble	Tuas pengaduk longgar	±
Bunyi	Swishswish	
Getaran	Normal	
Hasil_gorengan	Lembek dan Tidak gosong	ŧ
Jenis_kerusakan	Seal poros tabung longgar	ŧ
Penyebab_kerusakan	Pemakaian lebih dari 3 bulan menyebabkan aus atau	ŧ
Penanganan_kerusakan	Ganti seal poros dengan yang baru, untuk selanjutnya lakukan penggantian setiap 3 bulan	ŧ

Gambar 5.12 Selected Case

Dari keterangan gambar diatas diketahui bahwa kasus baru pada *troubleshooting* mesin memiliki permasalahan pada Tabung penggorengan dengan kerusakan yang terjadi adalah *seal* poros tabung longgar. Solusi yang bisa dijadikan refrensi oleh teknisi yaitu, mengganti *seal* poros dengan yang baru dan untuk selanjutnya melakukan penggantian setiap 3 bulan.

5.3.3 Revise

Bila dicermati kasus yang terpilih, terdapat perbedaan gejala (nilai pada *adjustment feature*) dengan kasus baru, yaitu pada *feature* Bunyi. Pada kasus baru (*Target Case*) *feature* Bunyi nilainya normal, sedangkan pada kasus yang terpilih (*Selected Case*) nilainya Swishswish. Bila menurut *user* gejala pada kasus yang terpilih sudah tidak relevan dan perlu diganti maka bisa direvisi atau mengadaptasi gejala kasus baru secara otomatis dengan mengklik tombol Adaptation (lihat Gambar 5.13 dan Gambar 5.14) pada *ESTEEM Aplication Interface*.

	Dahiana	Advertation	In a second a Many Course	11-1-	Deiret	L Luit
hange Hetrieval Attributes	Hetrieve	Adaptation	Incorporate New Lase	Help	Print	Exit
Gambar 5.13 To	ombol Ada	<i>ptation</i> Pad	a ESTEEM Aplicati	ion Int	terface	
K ESTEEM Application Interface						×
File Help				v		
Change Retrieval Attributes	Retrieve	Adaptation	Incorporate New Case	Help	Print	Exit
📕 Enter Target Case					_	X
Trouble	Tuas penç	jaduk longga	r		±	
Bunyi	Normal					
Getaran	ran Normal I_gorengan Lembek dan Tidak gosong					
Hasil gorengan						
nasi_gorengan	Lembek d	an Tidak gos	ong		*	
Selected Case - CASE_05	Lembek d	an Tidak gos	ong		±	ı x
Selected Case - CASE_05	Lembek d	an Tidak gos	ong			x
Selected Case - CASE_05 Kode_komponen_mesin Nama_komponen_mesin	VF02 Tabung po	an Tidak gos enggorengan				×
Selected Case - CASE_05 Kode_komponen_mesin Nama_komponen_mesin Trouble	VF02 Tabung pu Tuas pen	an Tidak gos enggorengan gaduk longga	ong 			×
Selected Case - CASE_05 Kode_komponen_mesin Nama_komponen_mesin Trouble Bunyi	VF02 Tabung pr Tuas pen Normal	an Tidak gos enggorengan gaduk longga	ong 			×
Selected Case - CASE_05 Kode_komponen_mesin Nama_komponen_mesin Trouble Bunyi Getaran	VF02 Tabung pr Tuas pen Normal Normal	an Tidak gos enggorengan gaduk longga	ong			

Gambar 5.14 Hasil Kasus Terpilih Setelah Dilakukan Adaptation

Namun tidak semua proses revisi dapat dilakukan secara otomatis oleh aplikasi ini. Ketika akan melakukan revisi untuk data kasus mengenai Hasil_gorengan, *user* bisa langsung mengganti *feature value* dari *feature* Hasil_gorengan secara manual. Teknik Adaptasi sulit untuk digeneralisir dan diterapkan. Terkadang juga adapatasi tidak perlu dilakukan karena pada saat *user* (teknisi) melakukan *retrieve* kasus, sebenarnya *user* sudah melakukan adapatasi secara internal (Riesbeck, 1996). Memang, terdapat banyak manfaat yang diperoleh dengan menjaga peran manusia dalam proses revisi ini.

5.3.4 Retain

Kasus yang telah direvisi bisa disimpan ke dalam *case base Troubleshooting* Mesin dengan mengklik tombol *Incorporate New Case* seperti pada Gambar 5.15 pada *ESTEEM Aplication Interface*, kemudian nama kasusnya seperti pada gambar 5.16.

nge Retrieval Attributes	Retrieve	Adaptati	on Incorpora	ate Ne	w Case	Help	Print	
Gai	mbar 5.15	Tombol	Incorporate	e New	Case			
K ESTEEM Application Interface							×	
File Help	((
Change Retrieval Attributes	Retrieve	Adaptation	Incorporate Ne	w Case	Help	Print	Exit	
📡 Enter Target Case							×	
Trouble	Tuas pengaduk longgar					*		
Bunyi Normal								
Getaran	Normal				9	1000		
Hasil_gorengan	Lembek dan Tidak gosong Incorporate Ca			ise in Ca	se-Base			
			C	Case Na	me: CAS	SE_22]
Selected Case - CASE_05				ان ا	ОК	Rese	.t	
Kode_komponen_mesin	VF02							
Nama_komponen_mesin	Tabung pe	enggorengan				±		
Trouble	Tuas penç	jaduk longga	r			*		1
Bunyi	Normal							
Calara	Normal					-		
Gergian	Normal							

Gambar 5.16 Penambahan Kasus Baru Pada Case-Base

Menurut Watson (2003), proses *Retain* diapit oleh dua proses lainnya, *Review* dan *Refine*. Pada proses *Review*, *user* (teknisi) melakukan diskusi dalam sebuah tim (divisi perawatan). Hasil diskusi tersebut berupa ulasan yang mengerucut pada keputusan untuk mempertahankan kasus baru pada *troubleshooting* mesin ini sebagai kasus baru dalam *case base*, atau tidak dipertahankan.

Bila kasus ini dianggap relevan dan perlu maka kasus baru akan tetap disimpan dalam *case base. Case base* tidak bersifat statik, dan bukan merupakan unit penyimpanan data yang hanya terus bertambah. Sebaliknya, *case base* mengumpulkan kasus-kasus dalam sebuah *database* kasus dan sekaligus membuang pengetahuan yang sudah lama. Dari waktu ke waktu, bisa saja informasi terhadap kasus-kasus *troubleshooting* mesin berubah, bisa jadi apa yang hari ini berguna, belum tentu besok masih berguna. Oleh karena itu, perlu juga upaya perbaikan (*Refine*) secara berkala dalam menyimpan kasus ini oleh *user* dan tim yang bertanggungjawab dalam proses *troubleshooting* mesin *Vacuum Frying* pada CV Kajeye Food.

5.4 Analisa Sistem

Bisa dikatakan dalam melakukan *troubleshooting* mesin *Vacuum Frying*, CV Kajeye Food belum memanfaatkan aplikasi berbasis komputer yang bisa memberikan kemudahan. Keberadaan pengalaman teknisi pada kasus *troubleshooting* mesin *Vacuum Frying* belum diorganisasikan dengan baik. Dengan pengembangan *prototype* Aplikasi *troubleshooting* mesin *Vacuum Frying* diharapkan dapat diimplementasikan pada CV Kajeye Food dan memberikan kemudahan khususnya pada divisi perawatan dalam melakukan *troubleshooting* mesin *Vacuum Frying* secara cepat. Adapun kelebihan *prototype* Aplikasi *troubleshooting* mesin *Vacuum Frying* dibandingkan penggunaan metode lama disajikan pada Tabel 5.6.

		Tabel 5.6 Perbandingan Sistem La	ma Dengan Sistem Baru
No	Unsur Pembanding	Metode Lama	Metode baru dengan CBR
1	Kecepatan	Pencarian solusi dilakukan dengan mengandalkan daya	Memungkinkan dapat dilakukannya pencarian solusi dengan cepat oleh user atau
		ingat. Bila teknisi ahli atau	pembantu teknisi ahli dengan memasukkan
		pakar berhalangan, maka terjadi	adjustment feature sesuai dengan kondisi
		penundaan perbaikan, sehingga	pada mesin. Kemudian <i>prototype</i> akan
		memakan waktu yang lama.	memiliki kemiripan dengan kasus baru.
2	Fleksibilitas	Dibutuhkan teknisi ahli	Sifat dari kecerdasan buatan yang bisa
		berpengalaman yang sudah	digandakan dan digunakan oleh siapa saja
		lama melakukan	tanpa harus adanya orang yang ahli dibidang
		troubleshooting pada mesin	tersebut. Hal ini memungkinkan aplikasi
		Vacuum Frying.	troubleshooting mesin Vacuum Frying
			digunakan oleh orang yang bukan ahli di
			bidang ini.
3	Konsistensi	Teknisi ahli sebagai manusia	Mampu memberikan hasil yang konsisten,
		tidak lepas dari sifat lupa atau	karena pencarian kesimpulan dilakukan oleh
E D		kurang ketelitian karena faktor	mesin komputer terhadap kasus-kasus lama
43		usia dan yang lainnya, sehingga	yang disimpan dalam <i>database</i> kasus.
		hasil inkonsisten dapat terjadi.	
4	Kepuasan	Manusia dalam memberi	Hasil dan kesimpulan yang dihasilkan oleh
	User	penilaian tidak lepas dari unsur	mesin komputer bersifat objektif. Sehingga
		subjektivitas, sehingga sangat	tingkat akurasi terjamin.
5		dimungkinkan mengurangi nilai	
	VAV	akurasi terhadap kesimpulan	A BRA
	ELIXYA	yang dihasilkan.	A BURNER

Dengan diimplementasikan *prototype* ini pada perusahaan kemudian dilakukan proses *retrieve*, yaitu penelusuran kasus lama untuk dapat digunakan mencari solusi dari kasus baru, setelah ditemukannya kasus lama yang relevan dengan permasalahan kasus baru kemudian dilakukan proses *reuse* yaitu menggunakan kembali solusi kasus lama yang

relevan tersebut untuk digunakan menyelesaikan permasalahan baru. Proses *retrieval* dan *reuse* yang dilakukan perusahaan dapat dikatakan telah melakukan uji verifikasi. Bila menurut perusahaan gejala pada kasus lama yang terpilih sudah tidak relevan dan perlu diganti maka dilakukan proses *revise* atau memperbaiki kasus lama dengan mengadaptasi kasus baru karena dari waktu ke waktu bisa saja terjadi perubahan. Proses *revise* yang dilakukan perusahaan dapat dikatakan telah melakukan uji validasi. Kasus yang telah dilakukan proses *revise* atau terdapat kasus baru yang belum ada pada *case base* kemudian dilakukan proses *retain* atau menyimpan kedalam *case base*. Dengan dilakukannya proses *retain* berarti perusahaan telah melakukan proses secara keseluruhan dan dapat dikatakan telah melakukan uji *prototype*.