

LAPORAN PRAKTIKUM KECERDASAN BUATAN

SISTEM PENDUKUNG KEPUTUSAN PADA

KERUSAKAN HANDPHONE



Disusun Oleh :

Pebri Sudibyo	1515015136
Indriawan Niswar	1515015143
Gusti Rezky Ananda	1515015166

Asisten Praktikum :

Anisa Nur Afiah 1415015068	Alvian Nur Wahyudi 1415015066
-------------------------------	----------------------------------

PROGRAM STUDI TEKNIK INFORMATIKA

JURUSAN TEKNOLOGI INFORMASI DAN KOMUNIKASI

FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI

UNIVERSITAS MULAWARMAN

2017

KATA PENGANTAR

Alhamdulillah dengan memanjatkan puji syukur kehadiran Allah swt yang maha pengasih dan penyayang yang telah memberikan rahmat, hidayah dan inayahnya kepada kami, sehingga dapat menyelesaikan penyusunan laporan ini.

Laporan ini merupakan salah satu tugas yang di berikan kepada kami dalam rangka pengembangan dasar ilmu pemograman terutama Pemrograman berbasis Visual Prolog. Selain itu tujuan dari penyusunan laporan ini juga untuk menambah wawasan secara meluas. Sehingga besar harapan kami, laporan yang kami sajikan dapat menjadi kontribusi positif bagi pengembang wawasan pembaca.

Akhirnya kami menyadari dalam penulisan laporan ini jauh dari kesempurnaan. Oleh karena itu, dengan segala kerendahan hati kami menerima kritik dan saran agar penyusunan laporan selanjutnya menjadi lebih baik.

Samarinda, 5 Mei 2017

Kelompok 10

DAFTAR ISI

Halaman Judul	i
Kata Pengantar	ii
Daftar Isi	iii
Daftar Tabel	iv
Daftar Gambar	v
BAB I Pendahuluan	1
1.1. Latar belakang	1
1.2. Rumusan masalah	2
1.3. Batasan masalah.....	2
1.4. Tujuan dan Manfaat	2
BAB II Landasan Teori	3
2.1 Fungsi-Fungsi Visual Prolog yang Digunakan.....	3
2.1.1 Unifikasi dan Lacakbalik	3
2.1.2 Data Objek Sederhana dan Jamak	4
2.1.3 Perulangan Dan Rekursi	6
2.1.4 List	8
2.1.5 Section Facts	9
2.2 Teori Keilmuan yang Diimplementasikan.....	10
2.2.1 Kerusakan LCD	10
2.2.2 Kerusakan Speaker	11
2.2.3 Kerusakan USB Port.....	11
BAB III Metodologi	12

3.1 Alur Pembuatan Sistem	12
BAB IV Hasil dan Pembahasan	14
4.1 Tabel Keputusan	14
4.2 Analisis Aplikasi.....	14
BAB V Penutup	21
5.1 Kesimpulan	21
5.2 Saran	21
Daftar Pustaka	22
Lampiran	23
1 Source Code [sebagian saja]	23
2 Kartu Konsul.....	35

Daftar Table

Table 4.1 Tabel Keputusan	14
---------------------------------	----

Daftar Gambar

Gambar 3.1 Diagram Alur Sistem	13
Gambar 4.2.1 tampilan awal	14
Gambar 4.2.2 keretakan LCD	15
Gambar 4.2.3 beli LCD.....	16
Gambar 4.2.4 speaker tidak bersuara	17
Gambar 4.2.5 speaker bersuara pecah.....	18
Gambar 4.2.6 port USB Patah.....	19
Gambar 4.2.7 port USB tidak berfungsi	20
Gambar 4.2.8 Ulang.....	20

BAB I

PENDAHULUAN

1.1. Latar Belakang

Semakin berkembangnya teknologi yang dibarengi dengan kemajuan zaman, maka semakin banyak teknologi maupun aplikasi-aplikasi yang dibuat dengan fungsi untuk mempermudah segala kegiatan manusia dalam berbagai bidang, sama seperti halnya suatu sistem yang dapat membantu manusia dalam memberikan informasi ataupun suatu pendukung keputusan, sehingga manusia akan terbantu dalam mengetahui apa solusi untuk masalah yang menjadi titik kesulitannya.

Dengan adanya suatu kecerdasan buatan maka segala pertanyaan mengenai suatu sistem yang dapat membantu manusia dalam menentukan suatu hal ini dapat terpenuhi atau terwujud, karena pada dasarnya kecerdasan buatan ini merupakan suatu sistem yang berdasar pada segala sesuatu yang dapat dilakukan oleh manusia, seperti contohnya adalah memberikan suatu informasi diagnosis masalah, dengan kata lain sistem ini beroperasi seperti cara berfikir manusia yang dapat memberikan suatu solusi, biasanya sistem ini bekerja pada sistem pakar, dan juga sistem pendukung keputusan.

Program ini merupakan program yang dapat memberikan suatu informasi dalam menentukan suatu kerusakan pada handphone dengan memberikan pilihan kerusakan apa yang terjadi, yang selanjutnya program akan berjalan dengan memberikan suatu pertanyaan-pertanyaan yang akan mengarahkan pengguna untuk dapat mengetahui masalah apa yang terjadi pada handphonenya serta memberikan solusi untuk masalah tersebut. Dengan adanya sistem program ini diharapkan agar dapat membantu pengguna untuk mengetahui masalah apa yang terjadi pada handphonenya

1.2. Rumusan Masalah

Berdasarkan deskripsi masalah yang dipaparkan diatas, maka rumusan masalah dalam pembuatan program “SPK pada Kerusakan Handphone” adalah :

- a. Bagaimana membuat program sistem pendukung keputusan menggunakan Bahasa program Visual Prolog?
- b. Bagaimana membangun suatu program sistem pendukung keputusan yang dapat membantu dalam penanganan kerusakan pada handphone ?
- c. Bagaimana cara meimplementasikan suatu program dalam kecerdasan buatan?

1.3. Batasan Masalah

Adapun Batasan masalah ini ialah sebagai berikut:

- a. Hanya dapat memberikan informasi kerusakan LCD, Speaker, dan Port USB !

1.4. Tujuan dan Manfaat

Adapun tujuan ini adalah sebagai berikut:

- a. Membuat program untuk mempermudah proses perbaikan beserta biaya pada kerusakan handphone.
- b. Memberikan informasi kerusakan pada suatu handphone.
- c. Menyelesaikan Tugas project akhir Kecerdasan buatan dan mendapat nilai untuk praktikum kecerdasan buatan.

Adapun manfaat ini adalah sebagai berikut:

- a. Membantu konsumen untuk mengetahui kerusakan dan biaya perbaikan pada perangkatnya melalui program Sistem Pendukung Keputusan pada Kerusakan Handphone ini.
- b. Membantu pengguna untuk menentukan apa yang akan dilakukan setelah mengetahui masalah yang terjadi pada perangkat handphonenya.
- c. Dapat menjadi refrensi pembelajaran dalam membangun sebuah sistem pendukung keputusan kecerdasan buatan.

BAB II

LANDASAN TEORI

2.1. Fungsi-fungsi visual prolog Yang Digunakan

2.1.1. Unifikasi Dan Lacak Balik

Unifikasi

Pada waktu Visual Prolog mencoba untuk mencocokkan suatu panggilan (dari sebuah subgoal) ke klausa (pada section **clauses**), maka proses tersebut melibatkan suatu prosedur yang dikenal dengan unifikasi (*unification*), yang mana berusaha untuk mencocokkan antara struktur data yang ada di panggilan (subgoal) dengan klausa yang diberikan. Unifikasi pada Prolog mengimplementasikan beberapa prosedur yang juga dilakukan oleh beberapa bahasa tradisional seperti melewati parameter, menyeleksi tipe data, membangun struktur, mengakses struktur dan pemberian nilai (*assignment*). Pada intinya unifikasi adalah proses untuk mencocokkan dua predikat dan memberikan nilai pada variabel yang bebas untuk membuat kedua predikat tersebut identik. Mekanisme ini diperlukan agar Prolog dapat mengidentifikasi klausa-klausa mana yang dipanggil dan mengikat (*bind*) nilai klausa tersebut ke variabel.

Lacak Balik

Pada waktu menyelesaikan masalah, seringkali, seseorang harus menelusuri suatu jalur untuk mendapatkan konklusi yang logis. Jika konklusi ini tidak memberikan jawaban yang dicari, orang tersebut harus memilih jalur yang lain. Metode balik-ke-atas-dan-coba-lagi (*backing-up-and-trying-again*) ini pada Visual Prolog disebut lacakbalik (*backtracking*). Visual Prolog menggunakan metode ini untuk menemukan suatu solusi dari permasalahan yang diberikan. Visual Prolog dalam memulai mencari solusi suatu permasalahan (atau goal) harus membuat keputusan di antara kemungkinan-kemungkinan yang ada. Ia menandai di setiap percabangan (dikenal dengan titik lacak balik) dan memilih subgoal pertama untuk telusuri. Jika subgoal tersebut gagal (ekivalen dengan menemukan jalan buntu), Visual Prolog akan lacakbalik ke titik lacakbalik (*back-tracking point*) terakhir dan mencoba alternatif subgoal yang lain.

Predikat *fail*

Visual Prolog akan memulai lacakbalik jika ada panggilan yang gagal. Pada situasi tertentu, ada kebutuhan untuk memaksa lacakbalik dalam rangka mencari alternatif solusi. Visual Prolog menyediakan predikat khusus *fail* untuk memaksa kegagalan sehingga memicu terjadinya lacakbalik. Efek dari *fail* sama dengan efek memberikan perbandingan $2=3$ atau subgoal yang tidak mungkin (*impossible*) lainnya.

Predikat *Cut* (!)

Visual Prolog memiliki *cut* yang digunakan untuk mencegah lacakbalik, ditulis berupa sebuah tanda seru (!). Efek dari *cut* adalah sederhana, yaitu tidak akan memungkinkan terjadinya lacakbalik melewati sebuah *cut*.

2.1.2. Data Objek Sederhana Dan Jamak

Data *object* sederhana terdiri dari 2 yaitu variabel atau konstanta.

Variabel

Variable harus dimulai dengan sebuah huruf kapital (A ..Z) atau sebuah *underscore* (_). Sebuah *underscore* tunggal merepresentasikan sebuah variable anonim. Variabel dalam prolog bersifat lokal bukan global, oleh karena itu jika ada dua klausa yang mengandung sebuah variabel X maka X pada kedua klausa tersebut adalah variabel yang berbeda.

Konstanta

Konstanta meliputi karakter, angka, dan atom. Suatu nilai konstanta juga merupakan nama dari konstanta tersebut. Konstanta 2 merepresentasikan angka 2 dan konstanta halo merepresentasikan simbol halo.

Karakter

Karakter bertipe **char**, yaitu karakter-karakter yang bisa tercetak (ASCII 32 – 127), karakter angka (0 – 9), huruf kecil (a – z), huruf kapital (A – Z) dan tanda baca. Konstanta karakter ditulis dengan diapit oleh tanda petik tunggal (‘). Contoh : ‘a’, ‘*’, ‘{’, ‘W’.

Angka

Angka merupakan salah satu dari integer atau real.

Atom

Sebuah atom terdiri dari sebuah simbol atau sebuah string. Perbedaan antara simbol dan string berkaitan dengan representasi dan implementasi mesin. Visual Prolog melakukan konversi tipe otomatis diantara domain string dan domain simbol. Jadi simbol dapat digunakan untuk domain string dan string dapat digunakan untuk domain simbol. Namun string dinyatakan sebagai sesuatu yang berada diantara tanda petik ganda sedang simbol tidak membutuhkan tanda petik ganda.

Simbol dimulai dengan sebuah huruf kecil dan hanya boleh berisikan huruf, angka, dan *underscore*. String adalah sesuatu yang diapit tanda petik ganda dan berisikan kombinasi dari karakter, kecuali ASCII NULL (0), yang dipakai untuk menandai akhir dari string.

Data object jamak

Data object jamak memperlakukan beberapa informasi sebagai sebuah item tunggal. Contohnya : tanggal 1 desember 1999. Tanggal tersebut terdiri dari 3 jenis informasi yaitu hari, bulan dan tahun. Deklarasi suatu domain yang mengandung data object jamak tanggal :

DOMAINS

tanggal_jamak = tanggal(unsigned, string, unsigned)

dan kemudian pada section CLAUSES dapat dituliskan :

T = tanggal(1,"desember",1999).

Deklarasi Domain-Campuran Jamak (*Compound Mix-Domain*)

Deklarasi domain-campuran jamak bermaksud:

- memiliki sebuah argumen dengan kemungkinan lebih dari satu tipe argumen;
- memiliki beberapa macam argumen, masing-masing dengan tipe yang berbeda;
- memiliki beberapa macam argumen, beberapa diantaranya dengan kemungkinan lebih dari satu tipe argumen.

Agar suatu predikat Visual Prolog dapat menerima suatu argumen yang memberikan informasi dengan tipe yang berbeda maka functor tersebut harus dideklarasikan.

Contoh berikut memperlihatkan klausa umur yang dapat menerima suatu argumen usia dengan tipe yang berbeda yaitu **string**, **real** atau **integer**.

DOMAINS

usia = i(integer); r(real); s(string)

PREDICATES

umur(usia)

CLAUSES

umur(i(USIA)):- write(USIA).

umur(r(USIA)):- write(USIA).

umur(s(USIA)):- write(USIA).

2.1.3. Perulangan Dan Rekursi

Proses Perulangan

Prolog menyediakan dua jenis perulangan yaitu lacakbalik (mencari jawaban jamak dari satu pertanyaan) dan rekursi (prosedur pemanggilan dirinya sendiri).

Lacakbalik

Ketika suatu prosedur melakukan lacakbalik, prosedur akan mencari alternatif jawaban dari sebuah goal yang sudah terpenuhi.

Implementasi Lacakbalik dengan Loop

Lacakbalik merupakan cara yang baik untuk mencari alternatif jawaban dari sebuah goal. Namun jika suatu goal tidak memiliki alternatif jawaban, lacakbalik masih dapat digunakan untuk melakukan perulangan. Berikut ini didefinisikan predikat dua-klausa.

Ulang.

Ulang:-ulang.

Ini untuk mengakali struktur kendali Prolog agar berpikir bahwa terdapat sejumlah jawaban berbeda yang tak terbatas (cara kerjanya akan dibahas pada bagian mengenai rekursi ekor / *tail recursion*). Kegunaan ulang adalah agar lacakbalik terjadinya tak terhingga).

Rekursi

Cara lain untuk melakukan perulangan adalah melalui rekursi. Prosedur rekursi adalah prosedur yang di dalamnya ada pemanggilan terhadap dirinya sendiri. Prosedur rekursi dapat merekam perkembangannya karena ia melewatkan (*passing*) pencacah, total, dan hasil sementara sebagai argumen dari satu iterasi ke iterasi berikutnya. Berikut ini merupakan contoh program untuk mencari faktorial dari suatu angka.

Rekursi Ekor (*Tail Recursion*)

Rekursi mempunyai kelemahan yaitu memakan memori. Ketika suatu prosedur memanggil dirinya, keadaan pemanggil prosedur dari eksekusi harus disimpan sehingga prosedur pemanggil dapat meresume keadaan tersebut setelah prosedur pemanggil selesai. Ini berarti jika ada suatu prosedur memanggil dirinya 100 kali, maka ada 100 keadaan dari eksekusi yang harus disimpan. Keadaan (*state*) yang disimpan tersebut dikenal dengan nama *stack frame*. Ukuran *stack* maksimum pada platform 16 bit, seperti IBM-PC dengan sistem operasi DOS, adalah 64KByte yang bisa mengandung sekitar 3000 atau 4000 *stack frame*. Pada platform 32 bit, secara teoritis bisa sampai ukuran Giga Byte.

Untuk mengatasi kelemahan tersebut, maka digunakan optimasi rekursi ekor (*tail recursion optimization*). Diumpamakan, selain memanggil prosedur C, prosedur B memanggil dirinya sendiri pada langkah terakhir. Ketika prosedur B memanggil B, *stack frame* dari pemanggilan B akan ditimpa nilainya oleh sebuah *stack frame* dari pemanggil B, jadi tidak menambah *stack frame* baru. Hanya argumen yang perlu di-*update* nilainya dan kemudian proses akan melompat ke awal prosedur. Dalam perspektif prosedural adalah sama seperti memperbaharui variabel pengendali perulangan.

Syarat dari rekursi ekor adalah:

- Pemanggil merupakan subgoal terakhir dari klausa tersebut.
- Tidak ada titik lacak balik sebelumnya pada klausa.

2.1.4. List

Pada Prolog, yang dimaksud dengan *list* adalah sebuah *object* yang didalamnya mengandung sejumlah *object* yang lain (jumlahnya dapat berubah-ubah). *List* dalam bahasa pemrograman lain bisa disamakan dengan tipe data *pointer* (C dan Pascal). Berikut ini cara penulisan *list* pada Prolog.

[1, 2, 3] /* list yang mengandung integer 1, 2 dan 3 */

[kucing, anjing, tikus] /* list yang terdiri dari 3 buah symbol */

["Syarif Musadek", "Yusida Andriani", "Diana Putri"]

/* list yang terdiri dari 3 buah string */

Untuk mendeklarasikan *list* yang elemennya bertipe integer dapat dilakukan seperti berikut ini.

DOMAINS

integerlist = integer*

Tanda asterik (*) berarti domain tersebut merupakan sebuah *list*. Tanda asterik juga dipakai di bahasa C untuk pendeklarasian tipe data *pointer*.

Head dan Tail (Kepala dan Ekor)

List adalah suatu data *object* jamak rekursif (*recursive compound object*). *List* terdiri dari 2 bagian yaitu *head*, yang merupakan elemen pertama dari *list* dan *tail*, elemen sisanya. *Tail* dari *list* adalah juga merupakan sebuah *list*, sedangkan *head* dari *list* merupakan sebuah elemen. Contoh:

head dari list [a, b, c] adalah a

tail dari list [a, b, c] adalah [b, c]

Bagaimana jika suatu *list* hanya mempunyai satu elemen?

head dari [c] adalah c

tail dari [c] adalah []

Jika kita selalu mengambil elemen pertama dari suatu *list* maka pada akhirnya kita akan mendapat sebuah *list* kosong (*empty list*) yang ditulis dengan tanda [] (kurung siku buka dan kurung siku tutup). *List* kosong tidak bisa dipecah lagi menjadi *head* dan *tail*.

Prolog juga menyediakan cara untuk secara eksplisit memisahkan antara bagian *head* dan *tail* dari suatu *list*. Pemisah tersebut menggunakan tanda *vertical bar* (|).

Contoh:

[a, b, c] ekuivalen dengan [a|[b, c]]

ekivalen juga dengan [a|[b|[c]]]

dan ekuivalen juga dengan [a|[b|[c|[]]]]

Pemisah tersebut bisa terletak pada bagian *list* manapun seperti:

[a, b, c, d] ekuivalen dengan [a, b|[c, d]]

2.1.5. Section Facts

Section facts terdiri dari fakta-fakta yang mana fakta-fakta tersebut dapat ditambah dan dihapus secara langsung dari sebuah program pada saat program sedang berjalan (*at run time*). Kita dapat mendeklarasikan sebuah predikat pada *section facts* dan predikat tersebut dapat digunakan sama halnya seperti kalau dideklarasikan pada *section predicates*.

Visual Prolog menyediakan beberapa predikat *built-in* untuk menangani hal yang berkaitan dengan penggunaan *section facts*, antara lain:

- *assert*, *asserta* dan *assertz* untuk menambah fakta baru pada *section facts*.
- *retract* dan *retractall* untuk menghapus fakta yang ada.
- *consult* untuk membaca fakta dari sebuah file dan menyertakan fakta tersebut ke dalam fakta internal.
- *save* menyimpan isi fakta internal ke dalam sebuah file.

2.2. Teori Keilmuan Yang DiImplementasikan

2.2.1. Kerusakan LCD

Gejala

Sebagai komponen dari ponsel yang berfungsi sebagai interface, tentu peran LCD sangat penting untuk menampilkan gambar, informasi, status dan lain-lainnya. Terbuat dari bahan yang sangat mudah pecah, ternyata LCD sangat rentan rusak. Umumnya kerusakan LCD disebabkan oleh singgungan fisik, jatuh, atau terkena benturan keras. Namun bisa saja kerusakan LCD disebabkan oleh hal lain seperti, korosi (berkarat), korslet (biasanya karena terendam air), atau bisa saja karena

pemakaian yang ekstrim dalam waktu lama (brightness yang diatur maksimal memaksa LCD bekerja berat).

Terjadinya LCD Retak dan LCD tidak menyala. Solusi perbaikan dapat dengan :

1. Menggunakan pelindung layar.
2. Hindari segala bentuk benturan.

2.2.2. Kerusakan Speaker

Semua *handphone* memiliki semacam *built-in* speaker di dalamnya. Speaker memungkinkan Anda untuk mendengar suara dari setiap panggilan telepon.

Speaker juga tidak hanya memungkinkan untuk mendengar audio dari percakapan tetapi juga bisa untuk memutar musik jika *handphone built-in MP3 player*.

Masalah yang sering terjadi pada speaker handphone biasanya tidak terdengarnya suara apa pun

Terjadinya kerusakan speaker pada handphone. Solusi perbaikan dapat dengan :

1. Speaker tidak ada suara.
2. Speaker bersuara pecah.

2.2.3. Kerusakan Port USB

Perlu diketahui bahwa jenis charger HP sangatlah bermacam-macam. Klasifikasi tersebut bisa dilihat berdasarkan bentuknya, port yang digunakan, sampai dengan output tegangannya.

terjadinya kerusakan pada USB Port. Solusi perbaikan dapat dengan :

1. Jangan memutar suara dengan kondisi volume full.
2. Hindari dari segala bentuk benturan yang dapat mengakibatkan kerusakan pada usb port

BAB III

METODOLOGI

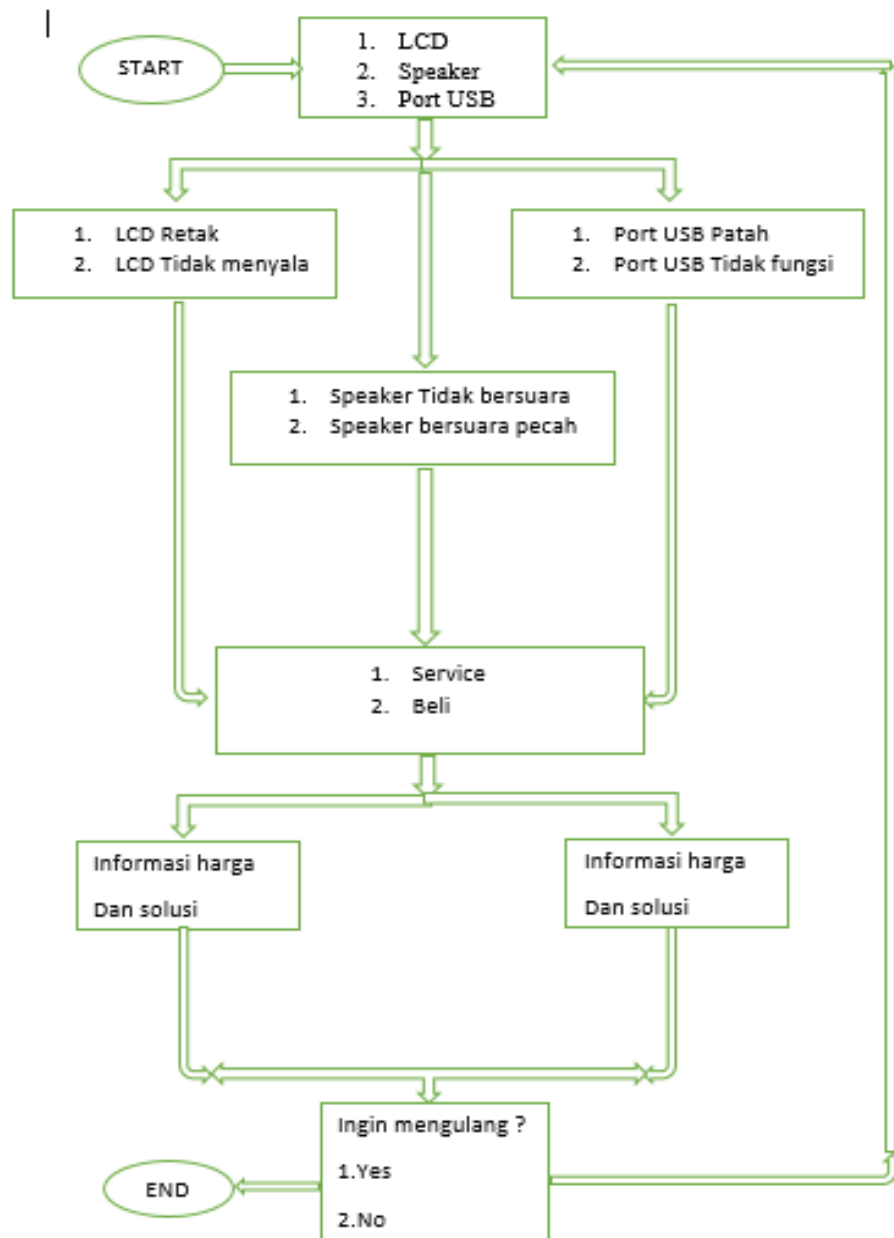
3.1. Alur Pembuatan Sistem

Sistem ini berjalan dengan memberikan suatu opsi inputan kepada pengguna, dimana pengguna hanya melakukan input berupa pilihan angka, ketika program ini dijalankan , akan menampilkan suatu pilihan kerusakan seperti :

1. Kerusakan LCD
2. Kerusakan Speaker
3. Kerusakan Port USB

Pengguna hanya perlu memilih dan program akan memberikan pilihan kerusakan pada LCD , dan selanjutnya program akan mengarahkan pada penanganan yang akan dilakukan berupa Service atau membeli baru. Kemudian program akan memberikan detail harga untuk perbaikan beserta memberikan solusi perbaikan, terakhir program akan memberikan opsi untuk melanjutkan atau mengakhiri.

Berikut diagram Alur pada program ini :



Gambar 3.1 Diagram Alur Sistem

BAB IV

HASIL DAN PEMBAHASAN

4.1. Table Keputusan

Tabel 4.1 Tabel Keputusan

Gejala	KERUSAKAN		
	LCD	Speaker	Port USB
LCD Retak	>		
LCD Tidak Menyala	>		
Speaker Tidak Bersuara		>	
Speaker Bersuara Pecah		>	
Port USB Patah			>
Port USB Tidak Berfungsi			>

4.2. Analisi Aplikasi

PROJECT AKHIR AI
 JUDUL : SISTEM PENDUKUNG KEPUTUSAN KERUSAKAN HP BESERTA KISARAN BIAYA
 KELOMPOK 10
 1.Pebri Sudibyo 1515015136
 2.Indriawan Niswar 1515015143
 3.Gusti Rezky Ananda 1515015166

SISTEM PENDUKUNG KEPUTUSAN KERUSAKAN HP BESERTA KISARAN BIAYA
 KELOMPOK 10

1. LCD
 2. Speaker
 3. Port USB

Silahkan Masukan Pilihan Kerusakan HP Anda:

Gambar 4.2.1 tampilan awal

Pada tampilan awal ini terdapat pilihan untuk jenis kerusakan untuk mencari tentang kerusakan LCD dapat mengetikkan angka 1, dan begitu juga untuk pilihan yang lainnya.

Kerusakan LCD

- 1. LCD**
- 2. Speaker**
- 3. Port USB**

Silahkan Masukan Pilihan Kerusakan HP Anda: 1

KERUSAKAN LCD

- 1. LCD Retak**
- 2. LCD Tidak Mau Hidup**

Silahkan Masukan Pilihan Kerusakan LCD Anda: 1

LAYAR LCD

- 1. Service LCD**
- 2. Beli LCD.**

Silahkan Masukan Pilihan: 1

Service LCD

Harga : Rp 450.000,-

- 1. Harga LCD Rp.250.000,-**
- 2. Biaya Service Rp.200.000,-**

SOLUSI LCD ANDA

- 1. Gunakan Pelindung Layar**
 - 2. Hindari Dari Segala Bentuk Benturan**
-

Gambar 4.2.2 keretakan LCD

Setelah melakukan pemilihan maka akan tampil sejumlah pilihan untuk mengetahui kerusakan dan solusinya, pada gambar diatas untuk mengetahui harga dan solusi jika LCD mengalami keretakan untuk service LCD.

1. LCD
2. Speaker
3. Port USB

Silahkan Masukan Pilihan Kerusakan HP Anda: 1

KERUSAKAN LCD

1. LCD Retak
2. LCD Tidak Mau Hidup

Silahkan Masukan Pilihan Kerusakan LCD Anda: 1

LAYAR LCD

1. Service LCD
2. Beli LCD.

Silahkan Masukan Pilihan: 2

BELI LCD

Harga : Rp 200.000,-

SOLUSI LCD ANDA

1. Gunakan Pelindung Layar
 2. Hindari Dari Segala Bentuk Benturan
-

Gambar 4.2.3 beli LCD

pada gambar diatas untuk mengetahui harga dan solusi jika LCD mengalami keretakan untuk bagian Beli LCD.

Untuk bagian LCD pada gejala Retak LCD atau LCD tidak menyala akan ditampilkan sama seperti tampilan diatas namun perbedaan terdapat pada solusi dan harga yang diberikan.

Kerusakan Speaker

- 1. LCD**
- 2. Speaker**
- 3. Port USB**

Silahkan Masukan Pilihan Kerusakan HP Anda: 2

Kerusakan Speaker

- 1. Speaker Tidak Ada Suara**
- 2. Speaker Bersuara Pecah**

Silahkan Masukan Pilihan Anda: 1

Kerusakan Speaker 1

- 1. Service Speaker**
- 2. Beli Speaker**

Silahkan Masukkan Pilihan Anda: 1

Service Speaker

Harga : Rp 200.000,-

- 1. Harga Speaker Rp.100.000,-**
- 2. Biaya Service Speaker Rp.100.000,-**

SOLUSI SPEAKER

- 1. Jangan Memutar Suara Dengan Kondisi Volume Full**
- 2. Hindari Dari Segala Bentuk Benturan**

Gambar 4.2.4 speaker tidak bersuara

Pada gambar diatas merupakan tampilan untuk pilihan kerusakan speaker dan pada pilihan selanjutnya speaker tidak bersuara dan memilih service speaker.

1. LCD
2. Speaker
3. Port USB

Silahkan Masukan Pilihan Kerusakan HP Anda: 2

Kerusakan Speaker

1. Speaker Tidak Ada Suara
2. Speaker Bersuara Pecah

Silahkan Masukan Pilihan Anda: 2

Kerusakan Speaker 2

1. Service Speaker
2. Beli Speaker

Silahkan Masukkan Pilihan Anda : 1

Service SpeakerHarga : Rp 200.000,-

1. Harga Speaker Rp.100.000,-
 2. Biaya Service Speaker Rp.100.000,-
-

Gambar 4.2.5 speaker bersuara pecah

Pada gambar diatas merupakan tampilan untuk pilihan kerusakan speaker dan pada pilihan selanjutnya speaker bersuara pecah dan memilih service speaker. Maka akan ditampilkan detail harga untuk service speaker.

Kerusakan Port USB

1. LCD
2. Speaker
3. Port USB

Silahkan Masukan Pilihan Kerusakan HP Anda: 3

Kerusakan PORT USB

1. Port USB Patah
2. Port USB Tidak Berfungsi

Silahkan Masukan Pilihan Anda: 1

-
- Port USB Patah1. Service Port USB**
2. Beli Port USB

Silahkan Masukan Pilihan Anda: 1

Service PORT USBHarga : Rp 120.000,-

1. Harga Port USB Rp.70.000,-
 2. Biaya Service Port USB Rp.50.000,-
-

Gambar 4.2.6 port USB Patah

Pada gambar diatas menampilkan untuk kerusakan port USB dengan pilihan port usb Patah. Dan melakukan pemilihan pada service port USB.

1. LCD
2. Speaker
3. Port USB

Silahkan Masukan Pilihan Kerusakan HP Anda: 3

Kerusakan PORT USB

1. Port USB Patah
2. Port USB Tidak Berfungsi

Silahkan Masukan Pilihan Anda: 2

Port USB Tidak Berfungsi

1. Service Port USB
2. Beli Port USB

Silahkan Masukan Pilihan Anda: 1

Service Port USBHarga : Rp 120.000,-

1. Harga Port USB Rp.70.000,-
 2. Biaya Service Port USB Rp.50.000,-
-

Gambar 4.2.7 port USB tidak berfungsi

Pada gambar diatas menampilkan untuk kerusakan port USB dengan pilihan port usb tidak berfungsi. Dan melakukan pemilihan pada service port USB. Dan akan tampil seperti gambar diatas.

Apakah anda ingin mengulang?

1. Yes

2. No

2

```
=====
=====
=                                     =
=                               TERIMA KASIH                               =
=====
=====
```

yes|

Gambar 4.2.8 Ulang

Pada gambar diatas merupakan tampilan yang akan tampil ketika pengguna telah mendapatkan informasi, jika memilih 1 maka program akan kembali kepada tampilan awal, dan jika pilih 2 maka program akan menyudahi proses.

BAB V

PENUTUP

5.1. Kesimpulan

- Dengan menggunakan program ini akan membantu memberikan informasi dalam mengatasi rusak handphone
- program ini menyediakan pilihan yang akan mengarahkan pengguna untuk mengetahui rusak pada handphone
- pada program ini akan memberikan informasi kepada pengguna berupa solusi serta harga untuk melakukan service atau perbaikan.
- Program ini melayani untuk memberikan informasi rusak beserta solusi dan harga hanya untuk kerusakan LCD, Speaker, dan Port USB.

5.2. Saran

Program ini masih memiliki kekurangan oleh karena itu kritik beserta saran sangat dibutuhkan untuk perbaikan dan menjadikan lebih baik lagi.

Program ini hanya berkonsentrasi pada kerusakan LCD, Speaker, Dan Port USB, serta pada program ini telah disediakan pilihan-pilihan yang telah memiliki jawaban yang pasti dan tidak memiliki jawaban lain.

DAFTAR PUSTAKA

Modul kecerdasan buatan.2017.*modul praktikum kecerdasan buatan*. Samarinda.

V-Tiga. *Ciri Ciri LCD HP Rusak*. <http://vtiga.com/2017/03/ciri-ciri-lcd-hp-rusak/>.

Diakses Jumat,5 mei 2017

Plimbi Editor. *Cara Memperbaiki Speaker Handphone Sendiri*.

<http://www.plimbi.com/article/4885/cara-memperbaiki-speaker>. Diakses Jumat, 5 mei 2017

Elektronika.2016. *Belajar Elektronika*. <http://belajarelekttronika.net/cara-memperbaiki-charger-hp-yang-rusak/>. Diakses jumat, 5 mei 2017

LAMPIRAN

1. Sourch Code

DOMAINS

kondisi = cond*

cond = string

FACTS

kelompok(kondisi)

false(cond)

PREDICATES

nondeterm start

nondeterm lihat(kondisi)

nondeterm pilih(integer)

nondeterm pilih1

nondeterm pilih2

nondeterm pilih3

nondeterm lcd(integer)

nondeterm lcd1

nondeterm lcd2

nondeterm kerusakanlcd1

nondeterm kerusakanlcd2

nondeterm perbaikanlcd1

nondeterm perbaikanlcd2

nondeterm fixlcd1(integer)

nondeterm fixlcd2(integer)

nondeterm speaker(integer)

nondeterm kerusakanspeaker1

nondeterm kerusakanspeaker2

nondeterm speakertidaksuara1

nondeterm speakertidaksuara2

nondeterm speakerpecah1

nondeterm speakerpecah2

nondeterm fixspeaker1(integer)

nondeterm fixspeaker2(integer)

CLAUSES

ulang(Y,Y):- !.

ulang(_,_):- fail.

kelompok(["PROJECT AKHIR AI\nJUDUL : SISTEM PENDUKUNG
KEPUTUSAN KERUSAKAN HP BESERTA KISARAN BIAYA\nKELOMPOK
10\n1.", "Pebri Sudibyo 1515015136\n2.", "Indriawan Niswar 1515015143\n3.", "Gusti Rezky
Ananda 1515015166\n"]).

start:-

kelompok(Kelompok),

```

        lihat(Kelompok),

        nl,nl,

write("\n"),

write("-----"),

write("SISTEM PENDUKUNG KEPUTUSAN KERUSAKAN HP BESERTA
KISARAN BIAYA\n"),

write("\t\t\t\t\tKELOMPOK 10"),

write("-----\n\n"),

write("\n"),

write("\n"),

write("1. LCD\n"),

write("2. Speaker\n"),

write("3. Port USB\n"),

write("\n"),

write("Silahkan Masukan Pilihan Kerusakan HP Anda: "),

readint(P),pilih(P).

pilih(P):- ulang(P,1),pilih1.

pilih(P):- ulang(P,2),pilih2.

pilih(P):- ulang(P,3),pilih3.

pilih(_):- start.

lihat([H|T]):-

        not(false(H)),

```

```
write(H),
```

```
    lihat(T).
```

```
    lihat([H_]):-
```

```
        assertz(false(H)).
```

```
pilih1:-
```

```
write("\n"),
```

```
write("\n"),
```

```
write("-----\n"),
```

```
write("KERUSAKAN LCD"),
```

```
write("\n"),
```

```
write("\n"),
```

```
write("1. LCD Retak \n"),
```

```
write("2. LCD Tidak Mau Hidup\n"),
```

```
write("\n"),
```

```
write("Silahkan Masukan Pilihan Kerusakan LCD Anda: "),
```

```
readint(M),lcd(M).
```

```
lcd(M):- ulang(M,1),lcd1.
```

```
lcd(M):- ulang(M,2),lcd2.
```

```
lcd(_):- lcd1.
```

```
lcd1 :-
```



```

write("\n"),

write("\n"),

write("-----\n"),

write("LAYAR LCD"),

write("\n"),

write("\n"),

write("1. Service LCD \n"),

write("2. Beli LCD.\n"),

write("\n"),

write("Silahkan Masukan Pilihan: "),

readint(P),fixlcd1(P).

```

```

fixlcd1(M):- ulang(M,1),kerusakanlcd1.

```

```

fixlcd1(M):- ulang(M,2),kerusakanlcd2.

```

```

fixlcd1(_):- kerusakanlcd1.

```

```

kerusakanlcd1:-

```

```

write("\n"),

```

```

write("\n"),

```

```

write("-----\n"),

```

```

write("Service LCD\n\n"),

```

```

write ("Harga : Rp 450.000,-\n\n"),

```

```

write("1. Harga LCD Rp.250.000,-"),nl,
write("2. Biaya Service Rp.200.000,-"),nl,
write("\n"),
write("-----\n"),
write("SOLUSI LCD ANDA"),
write("\n"),
write("1. Gunakan Pelindung Layar"),nl,
write("2. Hindari Dari Segala Bentuk Benturan"),nl,
write("\n"),
write("\n"),
write("\n"),

write("\n"),
write("\n"),
write("-----\n"),
write("Apakah anda ingin mengulang? "),
write("\n"),
write(" 1. Yes "),
write("\n"),
write("2. No "),

```

```
write("\n"),  
readint(A),perulangan(A).
```

```
perulangan(A):- ulang(A,1),start.
```

```
perulangan(A):- ulang(A,2),fix.
```

```
perulangan(_):- kerusakanlcd1.
```

```
kerusakanlcd2:-
```

```
write("\n"),
```

```
write("\n"),
```

```
write("-----  
\n"),
```

```
write("BELI LCD\n\n"),
```

```
write ("Harga : Rp 200.000,- \n\n"),
```

```
write("\n"),
```

```
write("-----  
\n"),
```

```
write("SOLUSI LCD ANDA"),
```

```
write("\n"),
```

```
write("1. Gunakan Pelindung Layar"),nl,
```

```
write("2. Hindari Dari Segala Bentuk Benturan"),nl,
```

```
write("\n"),
```

```
write("\n"),
```

```
write("\n"),
```

```

write("-----\n"),
write("Apakah anda ingin mengulang? "),
write("\n"),
write(" 1. Yes "),
write("\n"),
write(" 2. No  \n"),
kerusakanusb2 :-
write("\n"),
write("\n"),
write("-----\n"),
write("Port USB Tidak Berfungsi"),

write("\n"),
write("\n"),
write("1. Service Port USB \n"),
write("2. Beli Port USB\n"),

write("\n"),
write("Silahkan Masukkan Pilihan Anda: "),
readint(P),fixusb2(P).

fixusb2(M):- ulang(M,1),tidakberfungsi1.
fixusb2(M):- ulang(M,2),tidakberfungsi2.

```

fixusb2(_):- tidakberfungsi1.

tidakberfungsi1:-

write("\n"),

write("\n"),

write("-----
\n"),

write("Service Port USB"),

write ("Harga : Rp 120.000,- \n\n"),

write("1. Harga Port USB Rp.70.000,-"),nl,

write("2. Biaya Service Port USB Rp.50.000,-"),nl,

write("\n"),

write("\n"),

write("-----
\n"),

write("Apakah anda ingin mengulang? "),

write("\n"),

write("1. Yes "),

write("\n"),

write("2. No\n"),

write("\n"),

readint(A),perulangan10(A).

perulangan10(A):- ulang(A,1),start.

perulangan10(A):- ulang(A,2),fix.

perulangan10(_):- tidakberfungsi1.

tidakberfungsi2:-

write("\n"),

write("\n"),

write("-----
\n"),

write("Beli Port USB\n\n"),

write ("Harga : Rp 70.000,- \n\n"),

write("\n"),

write("\n"),

write("-----
\n"),

write("Apakah anda ingin mengulang?"),

write("\n"),

write("1. Yes "),

write("\n"),

write("2. No \n"),

write("\n"),

readint(A),perulangan11(A).

perulangan11(A):- ulang(A,1),start.

perulangan11(A):- ulang(A,2),fix.

perulangan11(_):- tidakberfungsi2.

fix:-

```
write("\n"),
```

```
write("\n"),
```

```
write("=====
=\n"),
```

```
write("=====
=\n"),
```

```
write("\t\tTERIMA KASIH\t\t=\n"),
```

```
write("=====
=\n"),
```

```
write("=====
=\n"),
```

```
write("\n"),
```

```
write("\n").
```

GOAL

start.

KARTU KONSUL

PERATURAN

1. Praktikan wajib konsultasi Programming Project dengan Asisten Yang telah ditentukan.
2. Praktikan yang tidak konsultasi **DILARANG** mengikuti ujian.
3. Konsultasi I, membahas tentang konsep PP yang akan dibuat.
4. Konsultasi II, membahas tentang hal yang berkaitan dengan PP yang sedang dikerjakan.

SISTEM PENGAMBILAN KEPUTUSAN PADA

KERUSAKAN HANDPHONE

PRAKTIKUM KECERDASAN BUATAN

KELOMPOK C2/10/2015


PEBRI SUDIRYO	1515015136
INDRIAWAN NISWAR	1515015143
GUSTI REZKY ANANDA	1515015166



LABORATORIUM FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS MULAWARMAN

2017

B. Kartu Konsultasi ini diprint menggunakan kertas Cover berwarna sesuai dengan warna pada cover modul praktikum

Tanggal Konsultasi : 01 Mei 2017	
Uraian : konsul Program Rppg Sistem Pengambilan Keputusan Pada Kerusakan HandPhone.	
Tanda Tangan Asisten : 	

Tanggal Konsultasi : 02 Mei 2017	
Uraian : - Rppgs Program Usman Rruiog - Penjadwalan mengenai format laporan Project Asmit	
Tanda Tangan Asisten : 