

APLIKASI SISTEM PAKAR UNTUK DIAGNOSIS KERUSAKAN *HARDWARE* KOMPUTER *DESKTOP* PADA LABORATORIUM KOMPUTER DI SEKOLAH

Alcianno G. Gani
localghost2000@yahoo.com

Abstract

In this paper will describe the design of an application program using the programming language Visual Basic.Net, to assist in diagnosing damage to the desktop computer and find a solution to repair the hardware. By applying the diagnosis expert system on a desktop computer hardware damage is expected to ease in finding damage to computer hardware and get the right solution to repair the damage, so as to facilitate and accelerate time to repair a troubled computer. Through this expert system, we can create reports on the diagnosis of a computer, so we can print the report to be given to the technician.

Keywords: expert system, computer, hardware, troubleshoot, VB.Net

PENDAHULUAN

Pada era teknologi dan Informasi saat ini komputer memiliki tingkat penggunaan yang tinggi karena telah menjadi bagian dari kehidupan sehari-hari. Tingginya tingkat pemanfaatan komputer berbanding terbalik dengan pengetahuan pengguna komputer itu sendiri mengenai masalah teknis komputer.

Umumnya bagi pengguna komputer, perbaikan kerusakan komputer biasanya diserahkan pada teknisi sehingga membutuhkan waktu, tenaga dan biaya ekstra karena permasalahan jarak maupun berat dan ukuran komputer saat akan dibawa ke tempat teknisi.

Berdasarkan permasalahan diatas, dibutuhkan suatu aplikasi sistem pakar yang dapat mendeteksi atau mendiagnosis masalah yang timbul dari kerusakan komponen komputer. Untuk itu sistem pakar diagnosis ini berfungsi sebagai solusi dari permasalahan kerusakan pada komputer, sistem pakar ini akan membantu :

1. mengidentifikasi kerusakan pada komputer
2. Menemukan Penyebab Kerusakan pada komputer
3. Memberikan solusi dan Informasi mengenai Biaya Servis komputer.

PENGERTIAN SISTEM PAKAR

Sistem pakar adalah aplikasi berbasis komputer yang digunakan untuk menyelesaikan masalah sebagaimana yang dipikirkan oleh pakar yang mempunyai keahlian khusus pada bidang tertentu, seperti: melakukan diagnosis kerusakan perangkat keras komputer. Pengetahuan seorang pakar disimpan pada basis pengetahuan, dan diolah dengan mekanisme penalaran tertentu

Struktur Bagan Program Sistem Pakar

Untuk membangun sebuah aplikasi pakar berbasis pengetahuan, maka digunakan sistem pakar berbasis kaidah produksi, pada sistem pakar berbasis kaidah produksi terdapat 5 komponen penting membangun sistem tersebut yaitu basis pengetahuan, basis data, mesin penalaran, fasilitas penjelasan, interaksi pengguna.

- **Basis Pengetahuan**
Pengetahuan adalah pemahaman teoritis atau praktis mengenai suatu subjek atau domain. Agar pengetahuan dapat digunakan dalam sistem, pengetahuan harus direpresentasikan dalam format tertentu yang kemudian dihimpun dalam suatu basis pengetahuan, perrepresentasian dimaksudkan untuk menangkap sifat-sifat penting problema dan mem-

buat informasi itu dapat diakses oleh prosedur pemecahan problema. Kaidah produksi adalah cara formal untuk mempresentasikan rekomendasi, arahan, strategi, bentuk penulisan kaidah produksi dalam bentuk jika-maka (*if-then*). Kaidah jika-maka menghubungkan anteseden dengan konsekuesnsi yang diakibatkannya. Berikut berbagai bentuk kaidah jika-maka:

JIKA (antecedant) MAKA (konsekuen)

JIKA (kondisi) MAKA (aksi)

JIKA (premis) MAKA (konklusi)

- **Basis Data**
Fakta, informasi dikumpulkan didalam kesatuan basis data, yang akan digunakan pada aplikasi sistem pakar. Untuk menetapkan kesamaan kondisi pada aturan jika-maka pada basis pengetahuan dan mencari kesimpulan atas kondisi yang diberikan.
- **Metode Penalaran**
Penalaran merupakan proses untuk menghasilkan informasi dari fakta yang diketahui atau diasumsikan. Ada 2 metode penalaran yang penting dalam sistem pakar, yaitu: runut maju dan runut balik.
 1. **Runut Maju**
Runut maju berarti menggunakan himpunan aturan kondisi-aksi. Dalam metode ini, data digunakan untuk menentukan aturan mana yang akan dijalankan, kemudian aturan tersebut dijalankan.
 2. **Runut Mundur**
Runut mundur berarti menggunakan himpunan aturan *goal-driven*. Dalam metode ini, tujuan/goal digunakan untuk menentukan kemungkinan mana yang akan dijalankan, kemudian aturan tersebut dijalankan.
- **Fasilitas Penjelasan**
Fasilitas penjelasan memungkinkan pengguna untuk mengetahui pada apli-

kasi sistem pakar: bagaimana kesimpulan tertentu tercapai dan mengapa fakta diperlukan, maka sistem pakar harus dapat memberikan penjelasan alasan dan membenarkan saran, analisis, atau kesimpulan.

- **Interaksi Pengguna**
Interaksi pengguna adalah bentuk komunikasi pengguna antara aplikasi sistem pakar untuk mencari dan menemukan sebuah solusi.

Perancangan Sistem Pakar

Perancangan sistem pakar berbasis kaidah produksi dimulai dengan melakukan pengumpulan pengetahuan pakar pada suatu domain tertentu, kemudian membentuk *database* fakta untuk digunakan sistem pakar menyesuaikan fakta yang di masukkan *user*, mempresentasikan pengetahuan pada basis pengetahuan melalui akuisisi pengetahuan, membentuk program interaksi pengguna, serta fasilitas penjelasan dan pengembangan database pengetahuan.

Kelebihan dan Kekurangan Sistem Pakar

Sistem pakar sebagai aplikasi komputer juga memiliki berbagai kelebihan dan kekurangan dalam fungsinya:

- a. **Kelebihan Sistem Pakar**
 1. Pengetahuan seorang pakar yang sudah diadaptasi kebentuk *software* dapat diperbanyak dan disebarluaskan dalam jumlah yang tidak terbatas.
 2. Biaya memperbanyak *software* lebih murah dibanding menghadirkan seorang pakar..
 3. *Software* sistem pakar dapat digunakan kapan saja tanpa ada batas waktu
 4. Sifat komputer yang pasti dan selalu benar selama masukan dan algoritma yang diberikan benar sehingga keluaran sistem pakar dapat menjadi penunjang keputusan yang dapat diterima.

- b. Kelemahan Sistem Pakar
Pengembangan perangkat lunak sistem pakar lebih sulit dibandingkan perangkat lunak konvensional.

Faktor Kepastian

Ketidakpastian yang disebabkan oleh aturan yang tidak pasti dan jawaban pengguna yang tidak pasti, dapat menyebabkan kesulitan dalam mencapai kesimpulan yang sempurna. Aplikasi sistem pakar membutuhkan nilai parameter, untuk menunjukkan besarnya suatu kepercayaan atas hasil pencapaian suatu kesimpulan pada suatu aturan.

KERUSAKAN PERANGKAT KERAS KOMPUTER

Komputer sebagai alat elektronik dapat mengalami gangguan kinerja ketika digunakan, hal tersebut dapat menyebabkan pekerjaan seseorang menjadi tidak selesai tepat waktu dan efektif. Untuk dapat melakukan perbaikan kerusakan *hardware* komputer tersebut, maka perlu mengetahui apa penyebab dari kerusakan tersebut. Berikut bagian komponen *hardware* komputer yang mengalami kerusakan:

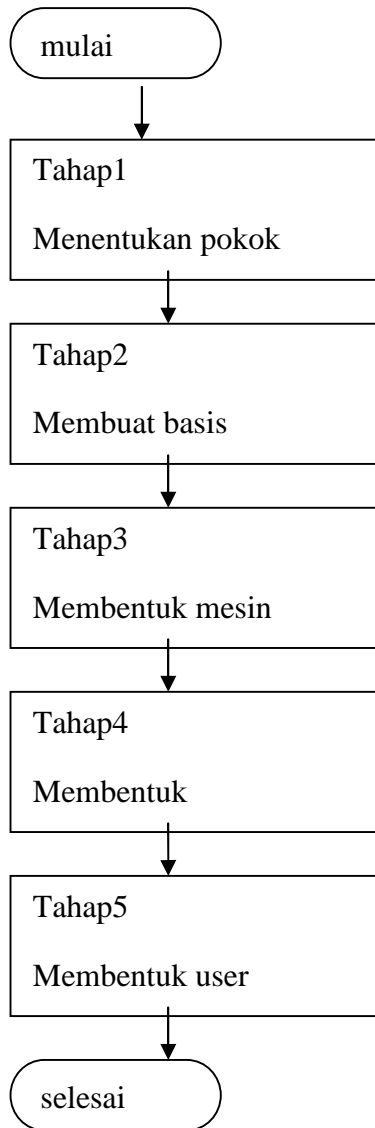
- A. *Keyboard*
Keyboard adalah *input device* primer pada *PC*, bentuk fisiknya berupa sebuah papan ketik dengan tombol abjad dan tombol numerik lengkap serta tambahan tombol fungsi lainnya. Pada komponen tersebut terdapat berbagai jenis kerusakan seperti: muncul pesan tampilan *keyboard error*, tombol *keyboard* tidak berfungsi, beberapa tombol tidak berfungsi.
- B. *Motherboard*
Motherboard adalah papan besar pada komponen komputer yang berfungsi sebagai induk atau pusat penghubung seluruh komponen sekaligus mengatur kerja komponen tersebut. Pada komponen tersebut terdapat berbagai jenis kerusakan seperti: sering terjadi *hang* pada *motherboard*, tidak dapat *booting*, suara “bib” terus menerus.

- C. *Monitor*
Monitor adalah *output* utama pada *PC*, perangkat tersebut bertugas untuk memberikan tampilan dari apa yang telah dikerjakan oleh pengguna *PC*. Pada komponen tersebut terdapat berbagai jenis kerusakan seperti: *monitor* menampilkan *RGB* rusak.
- D. *Printer Inkjet*
Printer adalah suatu alat untuk mencetak semua tulisan, dokumen yang terlihat pada layar *monitor*, dan lain-lain yang terkandung dalam komputer. Pada komponen tersebut terdapat berbagai jenis kerusakan seperti: hasil cetakan hanya 1 warna, warna cetakan tidak bersih, bagian belakang kertas terkena tinta, *printer* tidak dapat mencetak.
- E. *Harddisk*
Harddisk adalah sebuah subsistem perangkat keras komputer yang digunakan untuk menyimpan data dalam jumlah besar. Pada komponen tersebut terdapat berbagai jenis kerusakan seperti: tidak dapat di partisi.
- F. *Power Supply*
Power supply adalah perangkat keras komputer yang mendapatkan aliran listrik dan mengirimkan arus listrik yang dibutuhkan komponen komputer lain. Pada komponen tersebut terdapat berbagai jenis kerusakan seperti: kipas berisik dan tidak berputar.
- G. *CD-ROM*
Cd-rom adalah perangkat keras komputer bertugas membaca dan menulis pada media *disk*. Pada komponen tersebut terdapat berbagai jenis kerusakan seperti: sulit membaca media *disk*, pintu *cd-rom* sulit dibuka/ditutup.

PERANCANGAN DAN IMPLEMENTASI SISTEM

- A. Tahapan Perancangan Sistem Pakar
Dalam merancang suatu sistem pakar, perlu menentukan tahapan-tahapan peranc-

cangan sistem pakar tersebut, tahapan-tahapan perancangan di mulai dari menentukan pokok permasalahan, membuat basis pengetahuan, membentuk mesin penalaran, membentuk pengembangan sistem, membangun user interface. Berikut adalah gambar 4.1 menggambarkan *flowchart* dari tahapan perancangan:



Gambar 1 *flowchart* tahapan perancangan

Tahap 1: Menentukan Pokok Permasalahan

Pada tahap analisis sebelumnya telah mengidentifikasi adanya permasalahan pada sistem berjalan diagnosis kerusakan perang-

kat keras komputer *desktop*. Berikut permasalahan yang telah diidentifikasi :

1. Belum adanya aplikasi bantu yang dapat mendiagnosis kerusakan perangkat keras komputer.
2. Karyawan kesulitan dalam mendiagnosis kerusakan perangkat keras komputer *desktop*.
3. Karyawan kesulitan menemukan solusi perbaikan perangkat keras

Permasalahan tersebut membutuhkan sebuah aplikasi yang dapat membantu karyawan mendiagnosis dan melakukan perbaikan kerusakan pada setiap komponen perangkat keras komputer *desktop*, maka perlu membentuk aplikasi sistem pakar yang dapat memberikan solusi pada permasalahan tersebut.

Tahap 2: Membuat Basis Pengetahuan

Basis data dibentuk dan digunakan di dalam sistem pakar untuk menyesuaikan fakta yang diberikan oleh user kepada sistem pakar sesuai dengan basis pengetahuan aturan produksi, dan menyimpan hasil/kesimpulan solusi yang dihasilkan sistem pakar.

- Aturan 1 : mengalami masalah kipas *power supply* tidak berputar, cf: 0,7
Jika penyebab1: sebuah *circuit* elektronik dalam *fan* terbakar,
Maka solusi: ganti kipas dengan yang baru.
Atau penyebab2: klaker telah macet oleh debu,
Maka solusi: beri pelumas
- Aturan 2 : mengalami masalah kipas *power supply* berisik, cf: 0,5
Jika penyebab1: klaker telah macet oleh debu,
Maka solusi: beri pelumas
- Aturan 3 : mengalami masalah *PC* kadang nyala, kadang tidak, cf: 0,5
Jika penyebab1: tegangan listrik dalam rumah tidak stabil,
Maka solusi: gunakan *stabilizer*, agar menstabilkan litrik

- Aturan 4 : mengalami masalah *PC* mengalami *hang*, cf: 0,8
Jika penyebab1: *memory* utama tidak kompatibel dengan *motherboard* yang digunakan,
Maka solusi: coba ganti dengan *memory* lain.
Atau penyebab2: terdapat aplikasi *virus* di dalam *file harddisk*,
Maka solusi: gunakan *anti virus* yang terpercaya
Atau penyebab3: *PC* di *overclock* melebihi batasan kecepatan yang tertera pada *processor*,
Maka solusi: kembalikan kondisi *PC* pada kecepatan aslinya
- Aturan 5 : mengalami masalah *PC* tidak dapat *booting*, cf: 0,8
Jika penyebab1: *memory* cadangan rusak,
Maka solusi: non-aktifkan *memory* cadangan.
Atau penyebab2: *memory* utama tidak kompatibel dengan *motherboard* yang digunakan,
Maka solusi: coba ganti dengan *memory* lain.
Atau penyebab3: terdapat *bad sector* yang parah,
Maka solusi: gunakan *disk manager* untuk mengatasi *bad sector*.
- Aturan 6 : mengalami masalah suara “bib” panjang, terus-menerus dan tidak ada tampilan, cf: 0,6
Jika penyebab1: *memory* utama tidak kompatibel dengan *motherboard* yang digunakan,
Maka solusi: coba ganti dengan *memory* lain.
Atau penyebab2: pemasangan *memory ram* tidak tepat pada tempatnya,
Maka solusi: lakukan pemasangan dengan benar.
- Aturan 7 : mengalami masalah suara “bib” bagus 1 atau 2 kali, tapi tidak ada tampilan, cf: 0,8
Jika penyebab1: pemasangan *vga card* tidak tepat pada tempatnya,
Maka solusi: cek pemasangan rangkaian *video*
Atau penyebab2: kerusakan pada kabel *power monitor*,
Maka solusi: coba ganti dengan *monitor* lain.
- Aturan 8 : mengalami masalah *monitor* warna *RGB* rusak, cf: 0,6
Jika penyebab1: pemasangan *vga card* tidak tepat pada tempatnya,
Maka solusi: cek pemasangan rangkaian *video*
- Aturan 9 : mengalami masalah sulit membaca data pada *cd-rom*, cf: 0,7
Jika penyebab1: media *disk* tidak bagus,
Maka solusi: bersihkan dengan tisu diberi *alcohol* pada permukaan *disk*
Atau penyebab2: melemahnya kerja *optic cd-rom*,
Maka solusi: kalibrasi *trimpot*
Atau penyebab3: lensa *optic* kotor,
Maka solusi:bersihkan dengan *disk cleaner*.
- Aturan 10 : mengalami masalah *disk* tidak berputar, cf: 0,8
Jika penyebab1: kerusakan pada motor *disk*,
Maka solusi: ganti motor *disk* yang baru
- Aturan 11: mengalami masalah *CD-rom* susah buka tutup pintu, cf: 0,7
Jika penyebab1: motor pintu tidak bekerja baik,
Maka solusi: mengganti motor baki yang baru
Atau penyebab2: karet *tape* dan roda gigi tidak bagus,
Maka solusi: ganti karet *tape* baki dan bersihkan roda gigi poros baki
- Aturan 12 : mengalami masalah *optic* tidak dapat maju mundur, cf: 0,5
Jika penyebab1: terdapat debu menempel pada roda gigi perputaran poros,

Maka solusi: bersihkan dengan kuas halus

- Aturan 13 : mengalami masalah *hard-disk* tidak dapat di partisi, cf: 0,7
Jika penyebab1: *master boot record* rusak,
Maka solusi: gunakan *utility restore MBR*.
- Aturan 14 : mengalami masalah *hard-disk* dapat dipartisi, tetapi tidak dapat diformat, cf:0,7
Jika penyebab1: tidak pernah melakukan perawatan pada *hardisk*,
Maka solusi: lakukan *low level format*.
- Aturan 15 : mengalami masalah muncul tampilan *keyboard error* pada layer *monitor*, cf: 0,8
Jika penyebab1: konektor *keyboard* renggang, bila di goyang,
Maka solusi: ganti konektor *keyboard* yang baru
Atau penyebab2: kabel data tidak bagus,
Maka solusi: ganti dengan kabel data baru
Atau penyebab3: sekring pada *mother-board* dekat konektor *keyboard female*,
Maka solusi: ganti sekring yang baru
- Aturan 16 : mengalami masalah beberapa tombol *keyboard* tidak berfungsi, cf: 0,7
Jika penyebab1: jalur *pcb* kotor ,
Maka solusi: bersihkan dengan tisu yang diberi *alcohol* pada seluruh permukaan *pcb*.
Atau penyebab2: ada jalur yang terputus,
Maka solusi: sambungkan kembali dengan tinta emas
- Aturan 17 : mengalami masalah *printer inkjet* mencetak dokumen berwarna, tetapi yang tercetak hanya 1 warna, cf: 0,8
Jika penyebab1: hanya menggunakan *cartridge* hitam,

Maka solusi: gunakan *cartridge* berwarna, untuk mencetak dokumen berwarna

Atau penyebab2: pilihan *print* hanya pada skala *grayscale*,

Maka solusi: *unchecked* pada kotak pilihan *grayscale*.

Atau penyebab3: *driver printer* dengan *printer* yang digunakan tidak sesuai,

Maka solusi: sesuaikan *driver* dengan *printer* yang digunakan

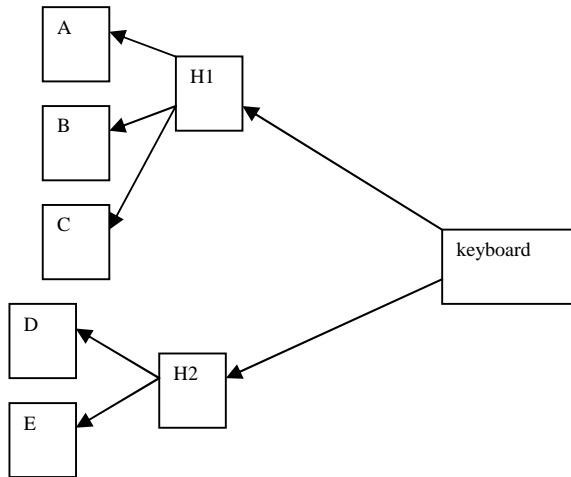
- Aturan 18 : mengalami masalah warna cetakan tidak bersih, cf: 0,8
Jika penyebab1: tidak menggunakan kertas cetakan dengan kebutuhan percetakan ,
Maka solusi: gunakan kertas cetakan yang sesuai kebutuhan percetakan.
Atau penyebab2: tidak menggunakan *cartridge* dengan benar,
Maka solusi: gunakan *cartridge* warna *photo*.
Atau penyebab3: tinta *cartridge* dalam kondisi tidak bagus, seperti bocor/tersumbat,
Maka solusi: lakukan pengecekan *head printer* pada “*nozzle check printer*”
- Aturan 19 : mengalami masalah bagian belakang kertas terkena tinta, cf: 0,6
Jika penyebab1: bagian dalam *printer* tidak berfungsi,
Maka solusi: bersihkan bagian dalam dan luar *printer* secara berkala
- Aturan 20 : mengalami masalah *printer* tidak dapat mencetak, cf:0,8
Jika penyebab1: kabel *power* belum terpasang ke pusat listrik,
Maka solusi: sambungkan kabel *power* pada pusat listrik
Atau penyebab2: *printer* dan komputer belum benar terkoneksi,
Maka solusi: sambungkan kabel *printer* pada komputer untuk terkoneksi

Tahap 3: Membentuk Mesin Penalaran

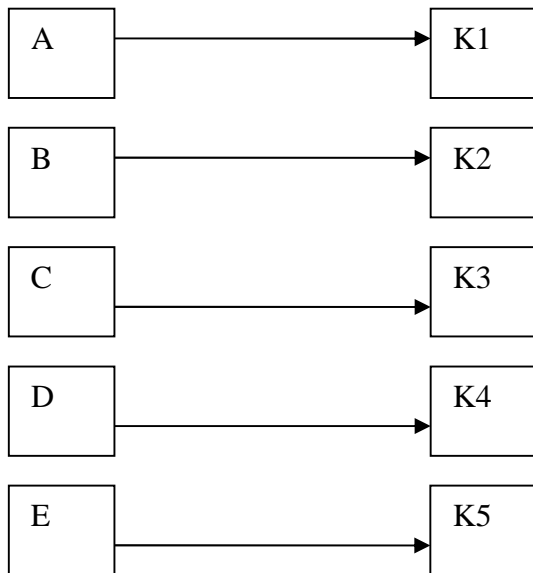
Metode penalaran adalah sebuah proses untuk menghasilkan informasi dari fakta

yang diketahui atau diasumsikan. Metode penalaran yang digunakan pada sistem pakar ini merupakan runut maju dan runut balik. Berikut bentuk penalaran runut maju dan runut mundur:

- Runut Mundur dan Maju Pada Kerusakan Keyboard



Gambar 2 Proses Runut Mundur Penyebab Kerusakan keyboard

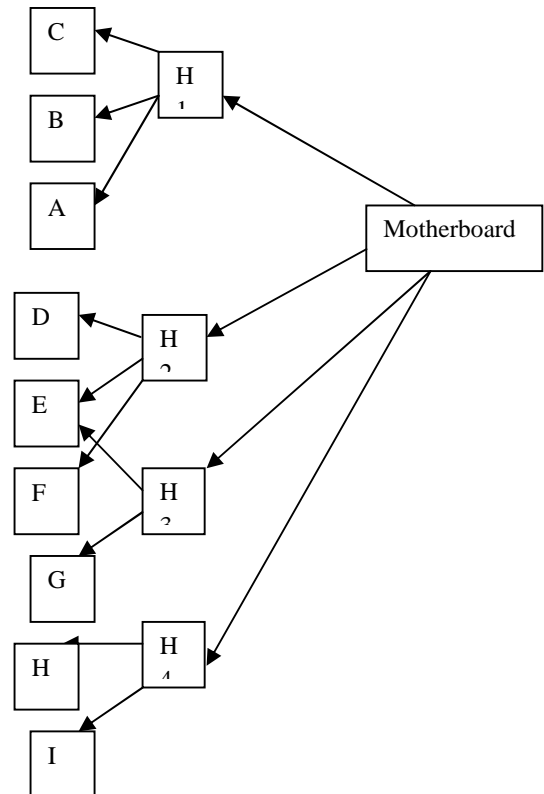


Gambar 3 Proses Runut Maju Solusi Perbaikan Kerusakan Keyboard

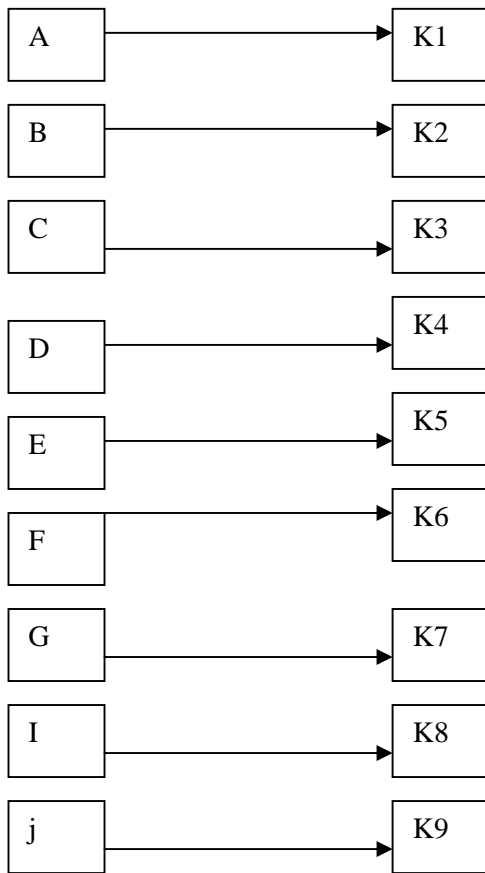
Keterangan

- A: konektor keyboard renggang, bila di goyang
- B: kabel data tidak bagus,
- C: sekring pada motherboard dekat konektor keyboard female
- D: jalur pcb kotor
- E: jalur penghubung putus
- H1: muncul tampilan keyboard error pada layar monitor
- H2: mengalami masalah beberapa tombol keyboard tidak berfungsi
- K1: ganti konektor keyboard yang baru
- K2: ganti dengan kabel data baru
- K3: ganti sekring yang baru
- K4: bersihkan dengan tisu yang diberi alcohol pada seluruh permukaan pcb
- K5: sambungkan kembali dengan tinta emas

- Runut Mundur dan Maju Pada Kerusakan Motherboard



Gambar 4 Proses Runut Mundur Penyebab Kerusakan Motherboard



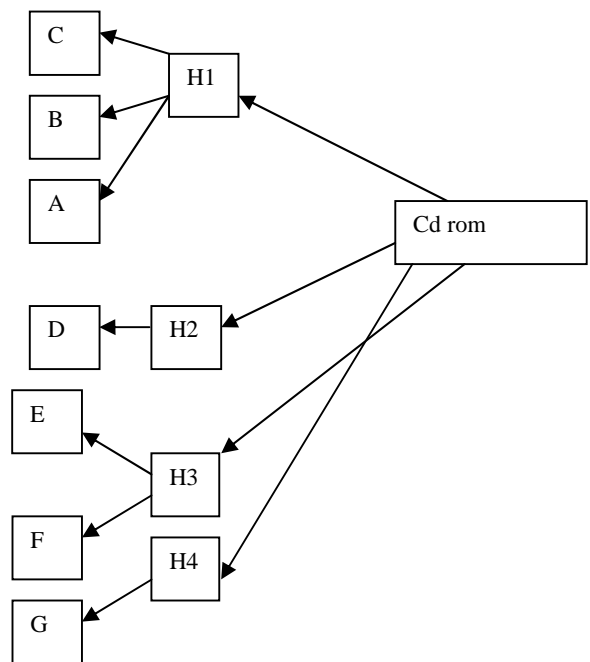
Gambar 5 Proses Runut Maju Solusi Perbaikan Kerusakan *Motherboard*

Keterangan

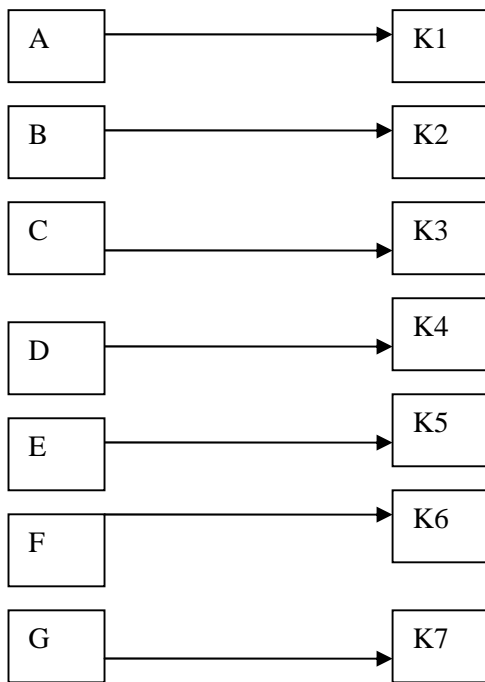
- A: *memory* utama tidak kompatibel dengan *motherboard* yang digunakan
- B: terdapat aplikasi *virus* di dalam *file harddisk*
- C: *pc* di *overclock* melebihi batasan kecepatan yang tertera pada *processor*
- D: *cache memory* rusak
- E: *memory* utama tidak kompatibel dengan *motherboard* yang digunakan
- F: terdapat *bad sector* yang parah
- G: pemasangan *memory ram* tidak tepat pada tempatnya
- H: pemasangan *vga card* tidak tepat pada tempatnya
- I: kerusakan pada kabel *power monitor*
- H1: *pc* mengalami *hang*
- H2: *pc* tidak dapat *booting*

- H3: suara “bib” panjang, terus-menerus dan tidak ada tampilan
- H4: suara “bib” bagus 1 atau 2 kali, tapi tidak ada tampilan
- K1: coba ganti dengan *memory* lain
- K2: gunakan anti *virus* yang terpercaya
- K3: kembalikan kondisi *pc* pada kecepatan aslinya
- K4: non-aktifkan *memory* cadangan
- K5: coba ganti dengan *memory* lain
- K6: gunakan *disk manager* untuk mengatasi *bad sector*
- K7: lakukan pemasangan dengan benar
- K8: cek pemasangan rangkaian *video*
- K9: coba ganti dengan monitor lain

- Runut Mundur dan Maju Pada Kerusakan *CD ROM*



Gambar 6 Proses Runut Mundur Penyebab Kerusakan *CD ROM*

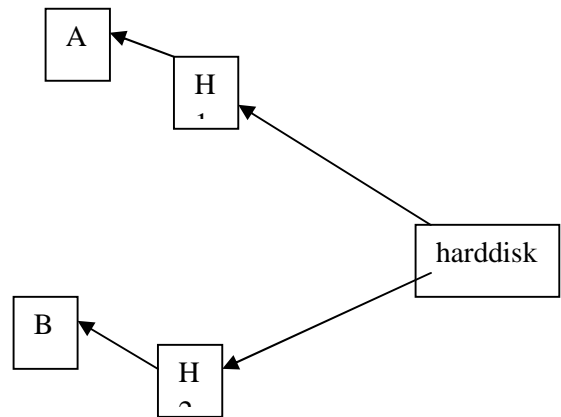


Gambar 7 Proses Runut Maju Solusi Perbaikan Kerusakan CD ROM

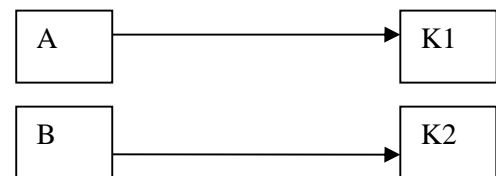
Keterangan

- A: media *disk* tidak bagus,
- B : melemahnya kerja *optic cd-rom*,
- C : lensa *optic* kotor,
- D: kerusakan pada motor *disk*,
- E: motor pintu tidak bekerja baik,
- F : karet *tape* dan roda gigi tidak bagus,
- G: terdapat debu menempel pada roda gigi perputaran poros,
- H1: sulit membaca data pada *cd-rom*
- H2: *disk* tidak berputar
- H3: *CD-rom* susah buka tutup pintu
- H4: *optic* tidak dapat maju mundur
- K1: bersihkan dengan tisu diberi *alcohol* pada permukaan *disk*
- K2 : kalibrasi *trimpot*
- K3: bersihkan dengan *disk cleaner*
- K4: ganti motor *disk* yang baru
- K5: mengganti motor baki yang baru
- K6: ganti karet *tape* baki dan bersihkan roda gigi poros baki
- K7: bersihkan dengan kuas halus

- Runut Mundur dan Maju Pada Kerusakan *Harddisk*



Gambar 8 Proses Runut Mundur Penyebab Kerusakan *Harddisk*

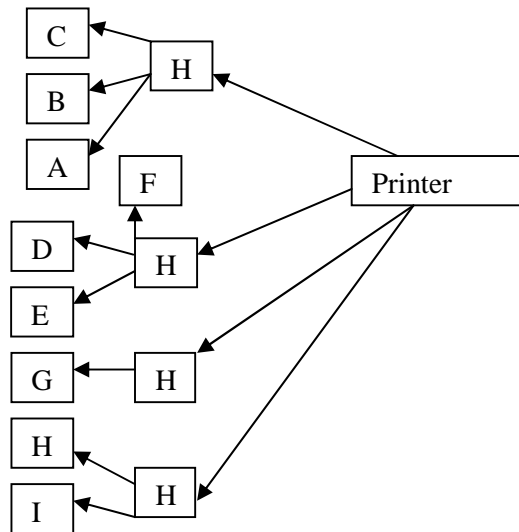


Gambar 9 Proses Runut Maju Solusi Perbaikan Kerusakan *Harddisk*

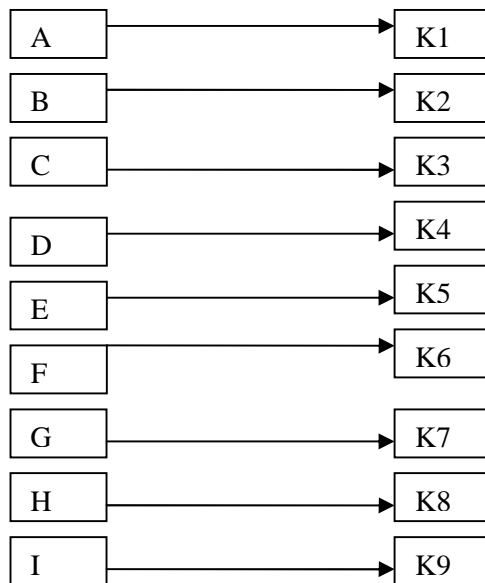
Keterangan

- H1: *harddisk* tidak dapat di partisi
- H2: *harddisk* dapat dipartisi, tetapi tidak dapat diformat
- A: *master boot record* rusak,
- B: tidak pernah melakukan perawatan pada *harddisk*,
- K1: gunakan *utility restore MBR*
- K2: lakukan *low level format*

- Runut Mundur dan Maju Pada Kerusakan *Printer*



Gambar 10 Proses Runut Mundur Penyebab Kerusakan *Printer*



Gambar 11 Proses Runut Maju Solusi Perbaikan Kerusakan *printer*

Keterangan

H1: *printer inkjet* mencetak dokumen berwarna, tetapi yang tercetak hanya 1 warna

A: hanya menggunakan *cartridge* hitam,

B: pilihan print hanya pada skala *grayscale*,

C: *driver printer* dengan *printer* yang digunakan tidak sesuai,

D: tidak menggunakan kertas cetakan dengan kebutuhan percetakan ,

E: tidak menggunakan *cartridge* dengan benar,

F: tinta *cartridge* dalam kondisi tidak bagus, seperti bocor/tersumbat,

G: bagian dalam *printer* tidak berfungsi,

H: kabel *power* belum terpasang ke pusat listrik,

I: *printer* dan komputer belum benar terkoneksi,

H2: warna cetakan tidak bersih

H3: bagian belakang kertas terkena tinta

H4: *printer* tidak dapat mencetak

K1: gunakan *cartridge* berwarna, untuk mencetak dokumen berwarna

K2: *unchecked* pada kotak pilihan *grayscale*

K3: sesuaikan *driver* dengan *printer* yang digunakan

K4: gunakan kertas cetakan yang sesuai kebutuhan percetakan

K5: gunakan *cartridge* warna *photo*

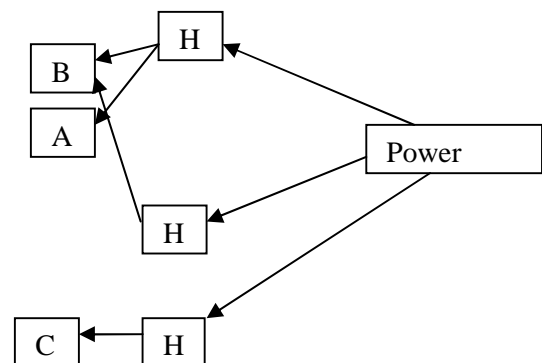
K6: lakukan pengecekan *head printer* pada “*nozzle check printer*”

K7: bersihkan bagian dalam dan luar *printer* secara berkala

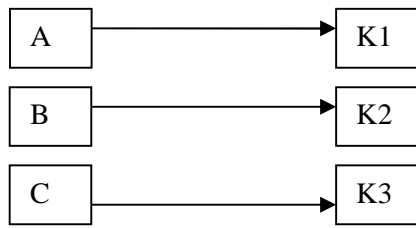
K8: sambungkan kabel *power* pada pusat listrik

K9: sambungkan kabel *printer* pada komputer untuk terkoneksi

- Runut Mundur dan Maju Pada Kerusakan *Power Supply*



Gambar 12 Proses Runut Mundur Penyebab Kerusakan *Power Supply*

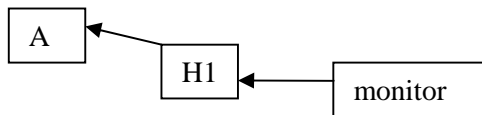


Gambar 13 Proses Runut Maju Solusi Perbaikan Kerusakan *Power Supply*

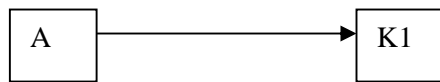
Keterangan

- A: sebuah *circuit* elektronik dalam *fan* terbakar,
- B: klaker telah macet oleh debu,
- C: tegangan listrik dalam rumah tidak stabil,
- H1: kipas *power supply* tidak berputar
- K1: ganti kipas dengan yang baru
- K2: beri pelumas
- H2: kipas *power supply* berisik
- K3: beri pelumas
- H3: *pc* kadang nyala, kadang tidak
- K4: gunakan *stabilizer*, agar menstabilkan listrik

- Runut Mundur dan Maju Pada Kerusakan *Monitor*



Gambar 14 Proses Runut Mundur Penyebab Kerusakan *Monitor*



Gambar 15 Proses Runut Maju Solusi Perbaikan Kerusakan *monitor*

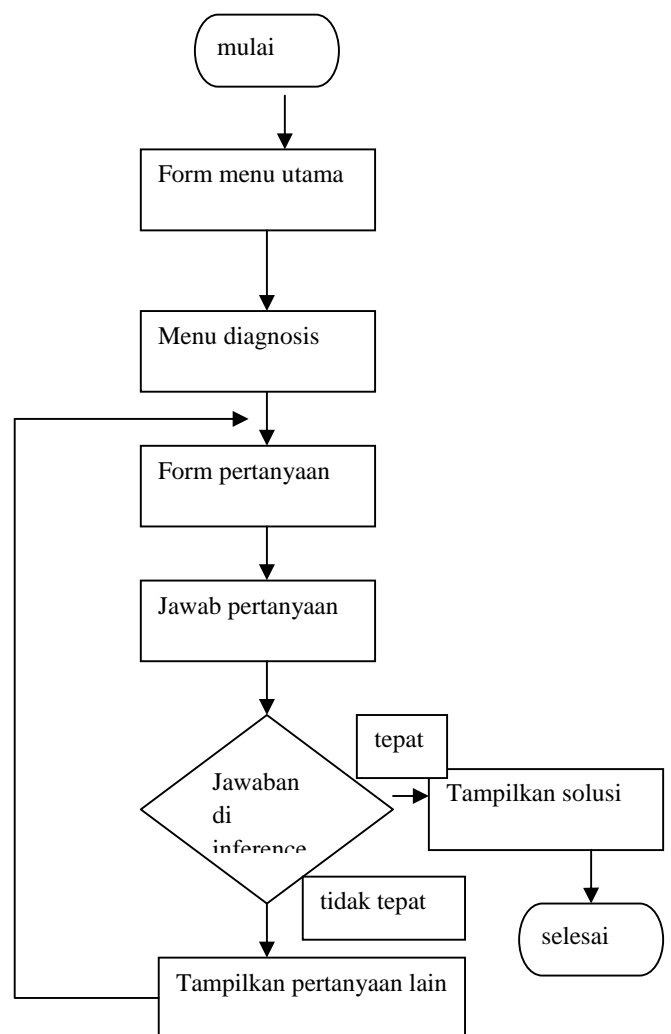
Keterangan

- A: tidak ada warna
- H1: kerusakan *rgb*
- K1: pasang *vga card* dengan tepat dan benar

Tahap 4: Membentuk Pengembangan Sistem

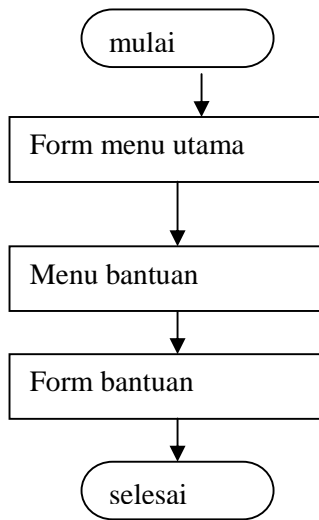
Pengembangan sistem pakar dimulai dengan membentuk alur program, membuat perangkat aturan yang mampu menggunakan mesin penalaran dan basis pengetahuan menjadi kesatuan sistem pakar. Perangkat aturan tersebut merupakan bahasa pemrograman yang digunakan aplikasi sistem pakar melakukan tugas tertentu yang didefinisikan.

- Bagan alir form diagnosis



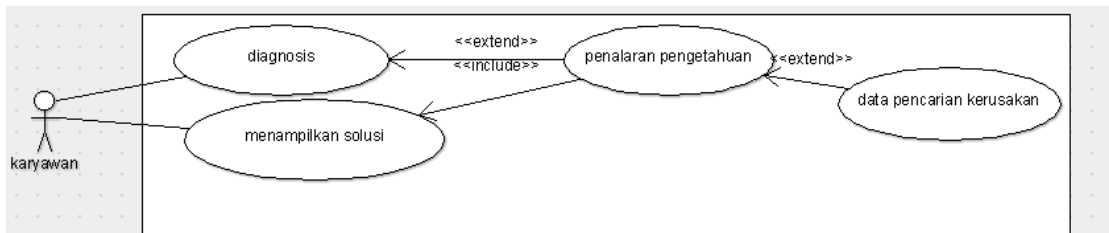
Gambar 16 Bagan Alir Form Menu diagnosis

- Bagan alir form bantuan



Gambar 17 Bagan Alir Form Menu Bantuan

- Diagram *Usecase* Sistem Pakar
Usecase merupakan gambaran fungsionalitas dari suatu sistem, sehingga user paham dan mengerti mengenai kegunaan sistem yang akan dibangun. Setelah mengidentifikasi *actor* yang memainkan peran dalam menjalankan sistem ini, tahap selanjutnya adalah mengidentifikasi *usecase* dalam sistem.



Gambar 18 *Usecase* Diagram Sistem Pakar

- Skenario *Use Case Diagram*
 Skenario *use case* digunakan untuk memudahkan dalam menganalisa skenario yang akan digunakan pada fase-fase selanjutnya dengan melakukan penilaian terhadap skenario tersebut. Berikut tahapan-tahapan skenario *use case* sistem pakar diagnosis kerusakan perangkat keras pada laboratorium komputer di sekolah adalah sebagai berikut:

1. Nama *Use Case* :Diagnosis
 Aktor : karyawan
 Tujuan : karyawan memulai diagnosis

Tabel 1 Skenario *Use Case* Diagnosis

Karyawan	Sistem
1.memulai menjalankan aplikasi diagnosis kerusakan perangkat keras komputer <i>desktop</i>	2. sistem menampilkan form diagnosis tanya jawab, untuk memulai diagnosis kerusakan komputer

2. Nama *Use Case* : Penalaran Pengetahuan

Aktor : karyawan

Tujuan : sistem melakukan penalaran pengetahuan pada data pencarian kerusakan

Tabel 2 Skenario *Use Case* Penalaran Pengetahuan

Karyawan	Sistem
	1. menampilkan form diagnosis
	2. mengambil data pencarian kerusakan
	3. menampilkan data tersebut pada form diagnosis
4. karyawan mempelajari setiap pertanyaan yang ditampilkan	
5. memasukkan jawaban ke aplikasi	

3. Nama *Use Case* : Menampilkan Solusi Perbaikan

Aktor : karyawan

Tujuan : karyawan mendapatkan solusi perbaikan atas hasil jawab pertanyaan diagnosis

Tabel 3 Skenario *Use Case* Menampilkan Solusi Perbaikan

Karyawan	Sistem
	1. sistem menampilkan solusi perbaikan, atas input jawab pertanyaan yang sesuai
2. karyawan mendapatkan solusi perbaikan, untuk melakukan perbaikan kerusakan perangkat keras komputer	

4. Nama *Use Case* : Data Pencarian Kerusakan

Aktor : sistem

Tujuan : sistem mengambil data pencarian kerusakan untuk menampilkan diagnosis pada aplikasi

Tabel 4 Skenario *Use Case* Data Pencarian Kerusakan

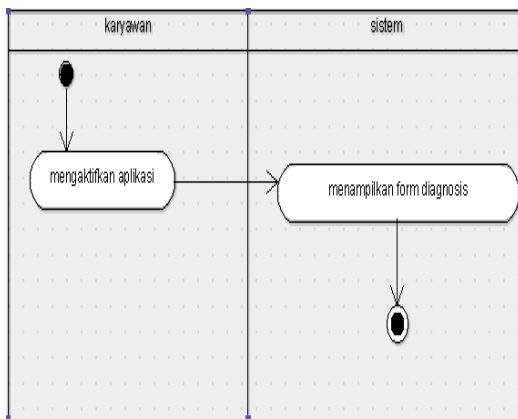
Sistem	
1. sistem menampilkan data komponen, data gejala pada form diagnosis dari data pencarian kerusakan	

- *Activity Diagram*

Activity Diagram digunakan untuk menggambarkan kegiatan-kegiatan yang ada dalam suatu sistem. Agar dapat lebih memahami tentang sistem yang akan dibuat, maka perlu dibuatkan *activity diagram* tentang sistem pakar diagnosis kerusakan perangkat keras komputer desktop, yaitu:

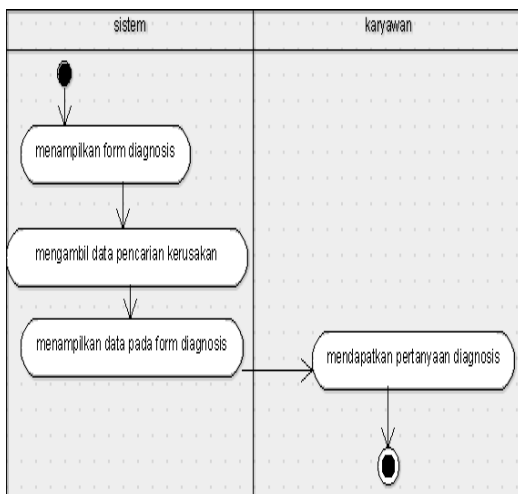
1. Diagram Aktifitas Pada *Usecase* Diagnosis

Karyawan memulai menjalankan aplikasi diagnosis kerusakan perangkat keras komputer *desktop*, sistem menampilkan form diagnosis tanya jawab, untuk memulai diagnosis kerusakan komputer.



Gambar 19 Activity Diagram Diagnosis

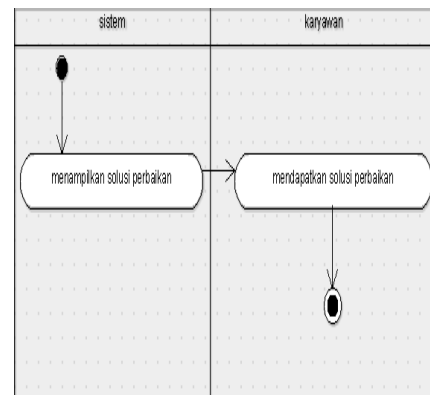
2. Diagram Aktifitas Pada Usecase Penalaran Pengetahuan
Sistem menampilkan form diagnosis, mengambil data pencarian kerusakan, menampilkan data pada form diagnosis, dan karyawan mempelajari setiap pertanyaan diagnosis yang ditampilkan.



Gambar 20 Activity Diagram Penalaran Pengetahuan

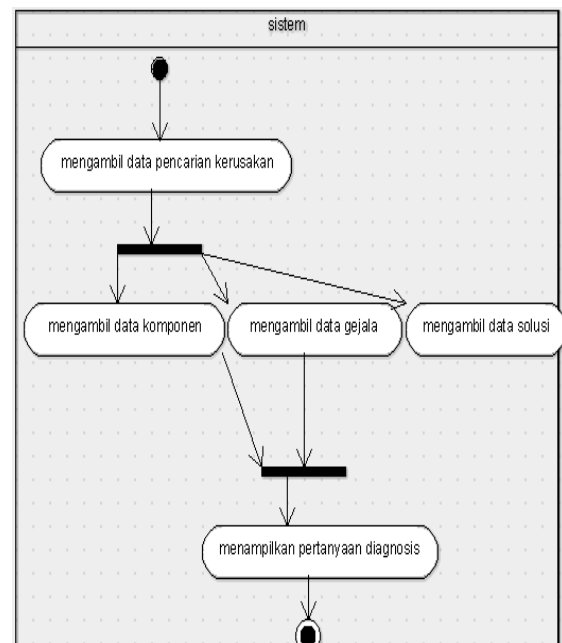
3. Diagram Aktifitas Pada Usecase Menampilkan Solusi Perbaikan
Sistem menampilkan solusi perbaikan, atas input jawab pertanyaan yang sesuai, karyawan mendapatkan solusi perbaikan, untuk me-

lakukan perbaikan kerusakan perangkat keras komputer.



Gambar 21 Activity Diagram Menampilkan Solusi Perbaikan

4. Diagram Aktifitas Pada Usecase Data Pencarian Kerusakan
Sistem menampilkan data komponen, data gejala kerusakan, data solusi perbaikan pada form diagnosis dari data pencarian kerusakan.

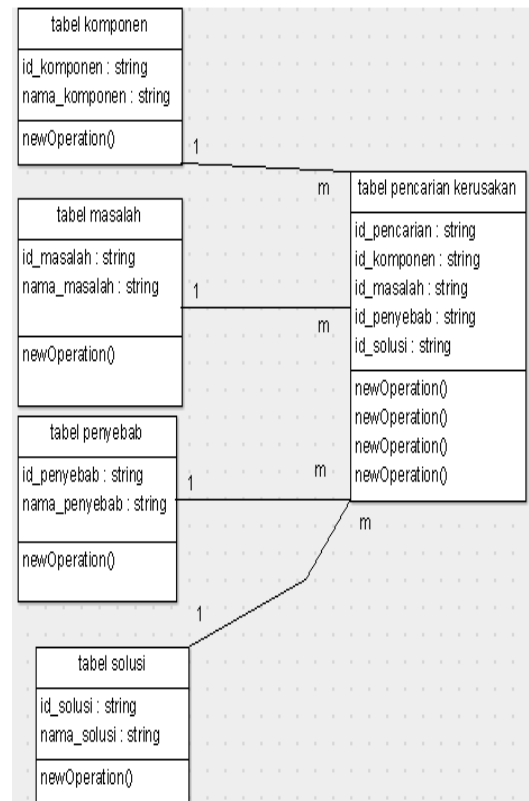
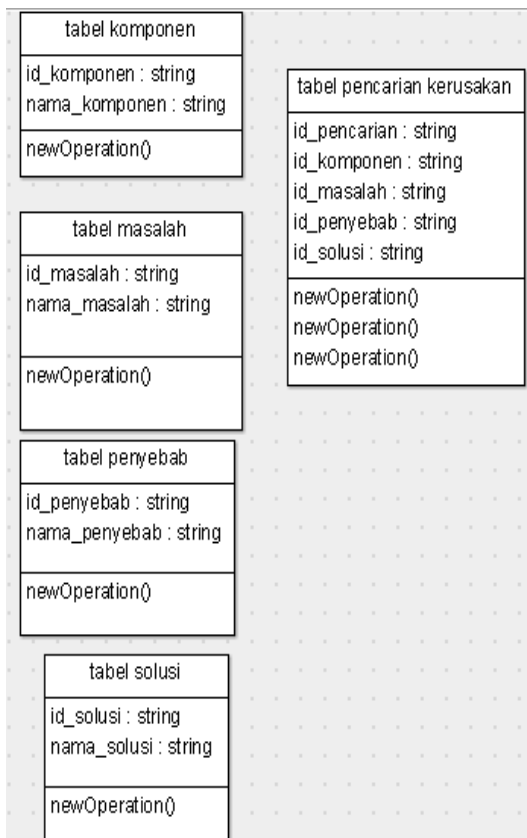


Gambar 22 Activity Diagram Data Pencarian Kerusakan

- **Pemodelan Objek**
Pemodelan objek adalah suatu metode untuk menggambarkan struktur sistem yang memperlihatkan semua objek yang ada berdasarkan hubungannya terhadap objek lainnya, menampilkan atribut serta operasi yang menjadi ciri suatu kelas tertentu.

Class diagram pada *diagram aktifitas* pencarian kerusakan

Pada *diagram aktifitas* memeriksa komputer terdapat aktifitas memeriksa kerusakan komponen komputer, maka di dapatkan sebuah tabel komponen komputer, dan terdapat pula aktifitas menemukan kerusakan pada komputer, dan menampilkan solusi perbaikan maka di dapatkan sebuah tabel masalah kerusakan, tabel gejala kerusakan, tabel pencarian kerusakan dan table solusi.



Gambar 23 Diagram ERD Sistem Pakar Diagnosis Kerusakan Perangkat Keras

Dari ERD diagram di atas dapat disimpulkan bahwa table-table yang berelasi pada implementasinya berperan sebagai *storage* basis data dan merupakan bagian paling penting dalam aktifitas sistem pakar diagnosis kerusakan perangkat keras, untuk mendapatkan informasi yang disediakan oleh layanan sistem.

- **Perancangan Basis Data**
Setelah mendefinisikan fungsi-fungsi, pada kebanyakan aplikasi-aplikasi diperlukan mendefinisikan berkas (*file*) basis data yang digunakan. Berkas-berkas tersebut disimpan dalam table-table dalam satu *database*, berikut berkas-berkas yang termasuk dalam *file library* adalah sebagai berikut:

Table 5 Pencarian Kerusakan

Nama table: pencarian kerusakan	
Primary key: id_kerusakan	
Nama field	Type
Id_pencarian	String(50)
Id_komponen	String(100)
Id_masalah	String(100)
Id_penyebab	String(100)
Id_solusi	String(100)

Table 6 Komponen

Nama table: komponen	
Primary key: id_komponen	
Nama field	Type
Id_komponen	String(50)
Nama_komponen	String(100)

Table 7 Penyebab

Nama table: Penyebab	
Primary key: id_penyebab	
Nama field	Type
Id_penyebab	String(50)
Nama_penyebab	String(100)

Table 8 Solusi

Nama table: solusi	
Primary key: id_solusi	
Nama field	Type
Id_solusi	String(50)
Nama_solusi	String(100)

Table 4.9 Masalah Kerusakan

Nama table: masalah	
Primary key: id_masalah	
Nama field	Type
Id_masalah	String(50)
Nama_masalah	String(100)

1. Halaman Menu Utama

Menu diagnosis	Menu bantuan
	Gambar
	Fasilitas penjelasan

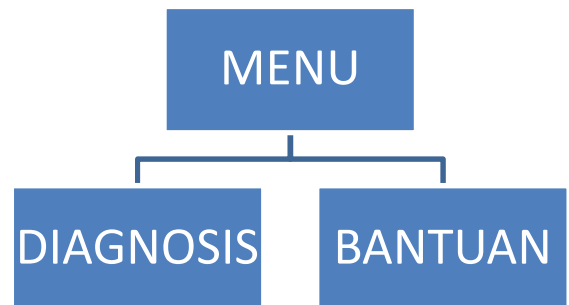
Gambar 25 Halaman Menu Utama

Tahap 5: Membentuk User Interface

Interaksi pengguna adalah bentuk komunikasi pengguna antara aplikasi sistem pakar untuk mendiagnosis kerusakan perangkat keras komputer *desktop*, dan menemukan sebuah solusi perbaikan yang tepat. Penggunaan aplikasi berbasis objek memudahkan pengguna dalam berinteraksi dengan aplikasi.

- Perancangan Menu Aplikasi
Rancangan struktur menu pada aplikasi sistem pakar digunakan untuk menggambarkan hak akses actor

Menu Utama



Gambar 24 Struktur Menu Utama

- Perancangan Halaman Aplikasi
Rancangan tampilan masukan berupa tampilan layar yang dapat digunakan oleh *user* untuk membuka atau menutup antarmuka yang lain atau mengirim message pada aplikasi untuk melakukan proses tertentu.

2. Halaman Menu Diagnosis

Diagnosis Kerusakan		
Form pertanyaan:		
<input type="radio"/>	Ya	<input type="radio"/>
	Tidak	
Button xxx	Button xxx	Button xxx

Gambar 26 Halaman Menu Diagnosis

3. Halaman Menu Bantuan

Bantuan	
Fasilitas penjelasan:	xxx

Gambar 27 Halaman menu bantuan

B. Faktor Kepastian Aturan

Ketidakpastian yang disebabkan oleh aturan yang tidak pasti dan jawaban pengguna yang tidak pasti, dapat menyebabkan kesulitan dalam mencapai kesimpulan yang sempurna. Aplikasi sistem pakar membutuh-

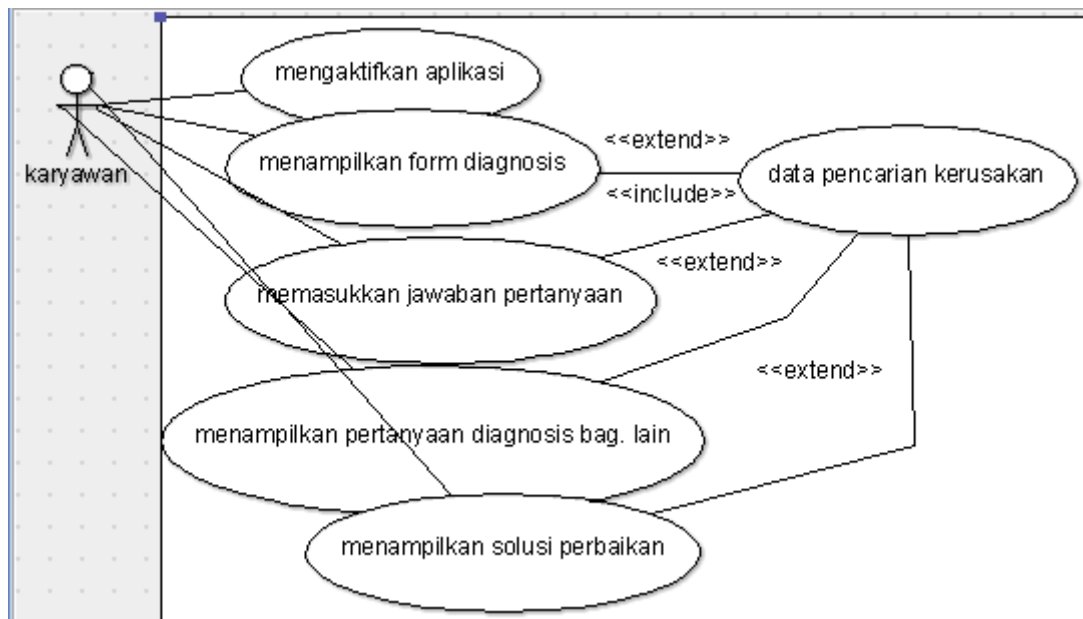
kan nilai parameter, untuk menunjukkan besarnya suatu kepercayaan atas hasil pencapaian suatu kesimpulan pada suatu aturan. Berikut bentuk perhitungan parameter faktor kepastian:

$$\begin{aligned}CF_{keyboard rusak1} &= CF_{masalah} * \min(CF_{gejala1}) \\ &= 0,8 * 0,3 \\ &= 0,21\end{aligned}$$

C. Diagram UseCase Sistem Usulan

Usecase merupakan gambaran fungsionalitas dari suatu sistem, sehingga user paham dan mengerti mengenai kegunaan sistem

yang akan dibangun. Setelah mengidentifikasi *actor* yang memainkan peran dalam menjalankan sistem ini, tahap selanjutnya adalah mengidentifikasi *usecase* dalam sistem.



Gambar 28 Usecase Diagram Aplikasi Sistem Pakar

1. **Skenario UseCase Diagram**

- a. Nama *Use Case* : Mengaktifkan Aplikasi
 Aktor : karyawan
 Tujuan : karyawan mengaktifkan aplikasi untuk memulai diagnosis

Tabel 10 Skenario Use Case Mengaktifkan Aplikasi

Karyawan	Sistem
1.memulai menjalankan aplikasi diagnosis kerusakan perangkat keras komputer desktop	2. sistem menampilkan form diagnosis tanya jawab, untuk memulai diagnosis kerusakan komputer

- b. Nama *Use Case* : Menampilkan Form Diagnosis
 Aktor : karyawan
 Tujuan : karyawan mendapatkan tampilan form diagnosis untuk memulai diagnosis kerusakan

Tabel 11 Skenario Use Case Menampilkan Form Diagnosis

Karyawan	Sistem
	1. menampilkan form diagnosis
	2. mengambil data pencarian kerusakan
	3.menampilkan data tersebut pada form diagnosis
4.karyawan mempelajari setiap pertanyaan yang ditampilkan	

- c. Nama *Use Case* : Memasukkan Jawaban Pertanyaan
 Aktor : karyawan
 Tujuan : karyawan menjawab pertanyaan dan menginput jawab tersebut ke sistem

Tabel 12 Skenario *Use Case* Memasukkan Jawaban Pertanyaan

Karyawan	Sistem
1.memasukkan jawaban ke aplikasi	2.Memproses jawaban tersebut, jika jawaban sesuai sistem akan menampilkan solusi, jika jawaban tidak sesuai, sistem akan menampilkan tampilan pertanyaan diagnosis lain

- d. Nama *Use Case* : Menampilkan Pertanyaan Diagnosis Bagian Lain
 Aktor : karyawan
 Tujuan : karyawan mempelajari pertanyaan diagnosis bagian lain

Tabel 4.13 Skenario *Use Case* Menampilkan Pertanyaan Diagnosis Bagian Lain

Karyawan	Sistem
	1. sistem menampilkan data komponen, data masalah, data penyebab pada form diagnosis lain, jika hasil input jawab pertanyaan karyawan tidak sesuai fakta dengan data pencarian kerusakan
2.Mempelajari setiap pertanyaan diagnosis yang ada	

- e. Nama *Use Case* : Menampilkan Solusi Perbaikan
 Aktor : karyawan
 Tujuan : karyawan mendapatkan solusi perbaikan atas hasil jawab pertanyaan diagnosis

Tabel 14 Skenario *Use Case* Menampilkan Solusi Perbaikan

Karyawan	Sistem
	1.sistem menampilkan solusi perbaikan, atas input jawab pertanyaan yang sesuai
2. karyawan mendapatkan solusi perbaikan, untuk melakukan perbaikan kerusakan perangkat keras komputer	

- f. Nama *Use Case* : Data Pencarian Kerusakan
 Aktor : sistem
 Tujuan : sistem mengambil data pencarian kerusakan untuk menampilkan diagnosis pada aplikasi

Tabel 15 Skenario *Use Case* Data Pencarian Kerusakan

Sistem	
1.sistem menampilkan data komponen, data masalah, data penyebab, data solusi pada form diagnosis dari data pencarian kerusakan	

2. Activity Diagram

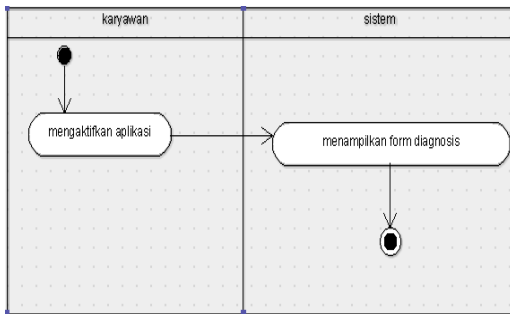
Activity Diagram digunakan untuk menggambarkan kegiatan-kegiatan yang

ada dalam suatu sistem. Agar dapat lebih memahami tentang sistem yang akan dibuat, maka perlu dibuatkan *activity*

diagram tentang sistem pakar diagnosis kerusakan perangkat keras komputer *desktop*, yaitu:

1. Diagram Aktifitas Pada *UseCase* Mengaktifkan Aplikasi

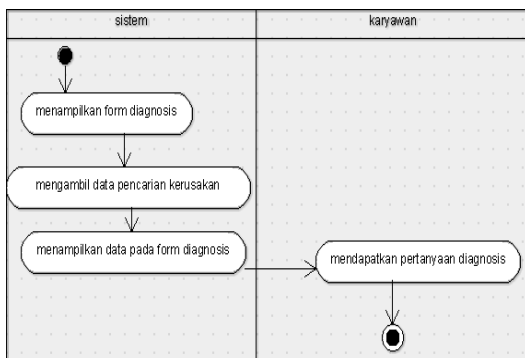
Karyawan memulai menjalankan aplikasi diagnosis kerusakan perangkat keras komputer *desktop*, sistem menampilkan form diagnosis tanya jawab, untuk memulai diagnosis kerusakan komputer.



Gambar 29 Activity Diagram Mengaktifkan Aplikasi

2. Diagram Aktifitas Pada *UseCase* Menampilkan Form Diagnosis

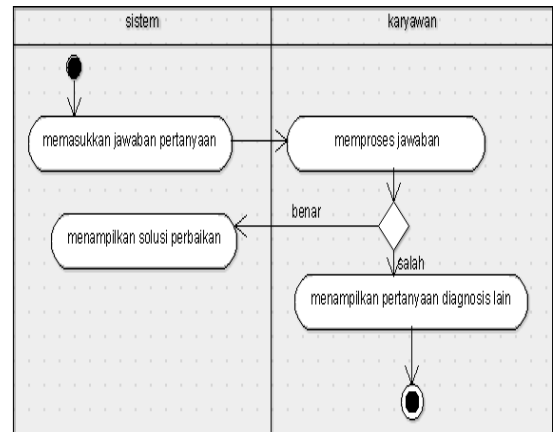
Sistem menampilkan form diagnosis, mengambil data pencarian kerusakan, menampilkan data pada form diagnosis, dan karyawan mempelajari setiap pertanyaan diagnosis yang ditampilkan.



Gambar 30 Activity Diagram Menampilkan Form Diagnosis

3. Diagram Aktifitas Pada *UseCase* Memasukkan Jawaban Pertanyaan

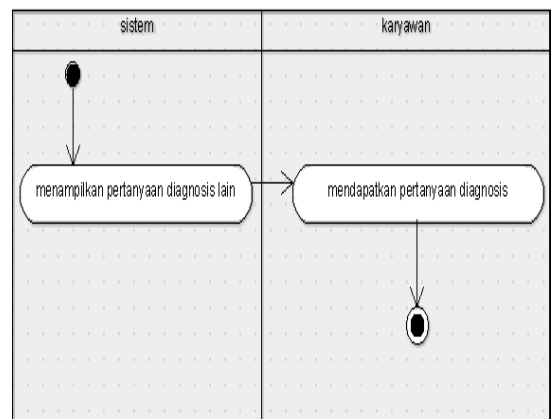
Karyawan memasukkan jawaban pertanyaan ke sistem, dan sistem memproses jika jawaban bernilai benar maka sistem menampilkan solusi perbaikan, tetapi jika jawaban bernilai salah maka sistem menampilkan pertanyaan diagnosis lain.



Gambar 31 Activity Diagram Memasukkan Jawaban Pertanyaan

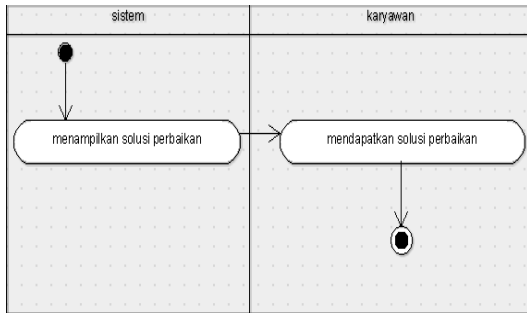
4. Diagram Aktifitas Pada *UseCase* Menampilkan Diagnosis Bagian Lain

Sistem menampilkan pertanyaan diagnosis lain, karyawan mendapatkan pertanyaan diagnosis.



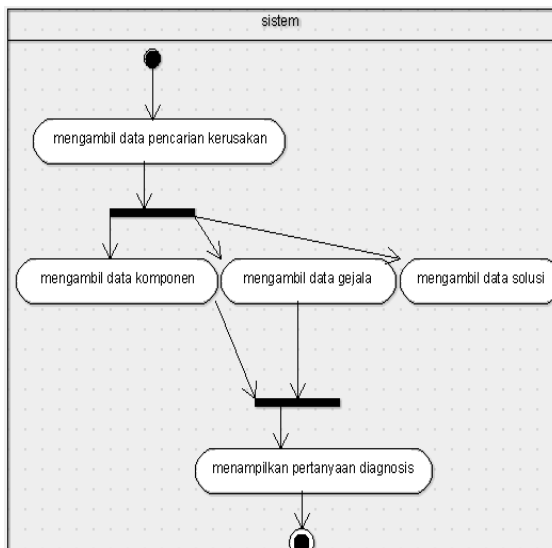
Gambar 32 Activity Diagram Menampilkan Diagnosis Bagian Lain

5. Diagram Aktifitas Pada *UseCase* Menampilkan Solusi Perbaikan
Sistem menampilkan solusi perbaikan, atas input jawab pertanyaan yang sesuai, karyawan mendapatkan solusi perbaikan, untuk melakukan perbaikan kerusakan perangkat keras komputer.



Gambar 33 *Activity Diagram* Menampilkan Solusi Perbaikan

6. Diagram Aktifitas Pada *UseCase* Data Pencarian Kerusakan
Sistem menampilkan data komponen, data gejala kerusakan, data solusi perbaikan pada form diagnosis dari data pencarian kerusakan.



Gambar 34 *Activity Diagram* Data Pencarian Kerusakan.

D. Implementasi Program Aplikasi

Sistem pakar diagnosis kerusakan perangkat keras *desktop* diimplementasikan dengan menggunakan bahasa pemrograman *visual basic* dan sistem manajemen *database Microsoft access*, *visual basic* merupakan bahasa tingkat tinggi yang berfungsi membuat aplikasi berbasis berorientasi objek. *Microsoft access* adalah *software* untuk membangun *database* yang berisi tabel-tabel yang digunakan sebagai data di aplikasi.

1. Spesifikasi Sistem Komputer

Sebelum membangun sebuah sistem, pengidentifikasian dari kebutuhan sebuah sistem yang akan dibangun harus dilakukan. Hal ini dilakukan agar diketahui kebutuhan standar yang diperlukan oleh sistem. Tujuannya, agar sistem yang dibangun dapat bekerja sesuai dengan tujuan dan kebutuhan pengguna.

- a. Kebutuhan Perangkat Keras
Pada bagian ini dijelaskan kebutuhan minimum perangkat keras yang diperlukan untuk berjalannya aplikasi sistem yang dibuat.
Komputer notebook
 - Prosesor :Intel Celeron
 - Ram : 512 mb
 - Hardisk :80 Gb free
 - Monitor :15" vga
- b. Kebutuhan Perangkat Lunak
Kebutuhan minimum perangkat lunak pada sistem ini adalah:
Komputer notebook
 - Windows Xp
 - Microsoft Access
 - Visual studio 2005

2. Struktur Program Aplikasi

Sistem pakar diagnosis kerusakan perangkat keras *desktop* menampilkan halaman diagnosis dengan karyawan memilih menu diagnosis untuk memulai diagnosis kerusakan.

- a. Menu Utama
Halaman muka yang digunakan untuk akses masuk ke dalam sistem bagi *user*. Terdapat menu *diagnosis kerusakan* dan *bantuan*.

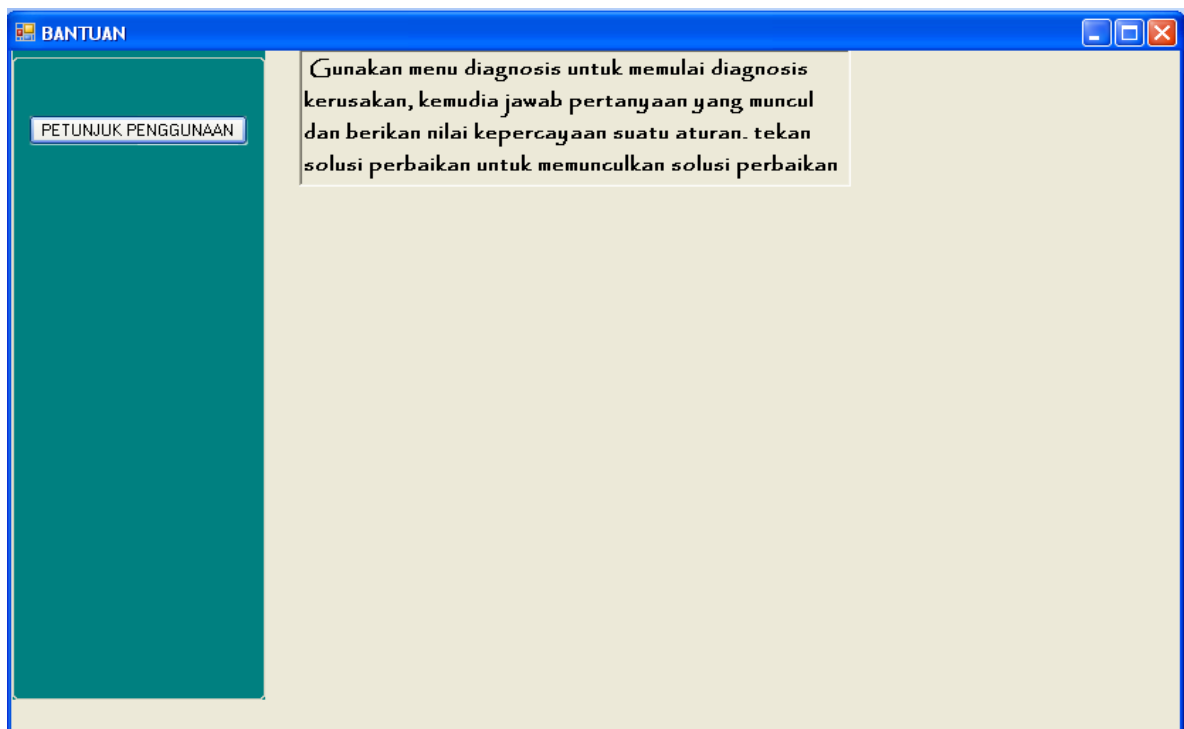


Gambar 35 Menu Utama

- b. Form Diagnosis Kerusakan
Halaman ini ditampilkan setelah *user* memilih menu *mulai diagnosis kerusakan*, untuk memulai melakukan diagnosis kerusakan perangkat keras.

Gambar 36 Form Diagnosis Kerusakan

- c. Form Bantuan
Halaman ini ditampilkan setelah *user* memilih menu *bantuan*, untuk mempelajari lebih rinci melakukan diagnosis kerusakan perangkat keras.



Gambar 37 Form Bantuan

3. Pengujian Program

Program yang telah dibuat harus diuji terlebih dahulu untuk memastikan fungsinya secara benar sebelum menghasilkan informasi secara benar. Langkah pengujian tersebut adalah dengan kompilasi program guna mengidentifikasi kesalahan sintak yang disebabkan oleh kesalahan pengetikan, inkonsistensi program atau kesalahan tata bahasa pemrograman.

Kesalahan yang mungkin terjadi pada saat program tersebut diuji dapat berupa kesalahan sintak bahasa pemrograman *Visual Basic.Net* atau kesalahan sintak yang berfungsi menghubungkan *Visual Basic.Net* dengan *Microsoft Access*. Kesalahan-kesalahan juga dapat disebabkan karena kesalahan *logical* yang berasal dari *human error* atau kekurangan komponen pendukung pemrograman yang akan sangat mempengaruhi program aplikasi yang dibuat. Apabila ditemukan kesalahan-kesalahan tersebut maka harus diperbaiki sampai

program tersebut dapat digunakan dan bersih dari kesalahan-kesalahan.

PENUTUP

Kesimpulan

Berdasarkan uraian dan pembahasan sebelumnya ada beberapa hal yang dapat disimpulkan dari masalah yang telah dianalisis sebelumnya. Beberapa hal yang dapat disimpulkan adalah sebagai berikut:

1. Sistem pakar diagnosis kerusakan perangkat keras komputer desktop ini dibuat sebagai desktop application sehingga dapat digunakan setiap komputer desktop dan memudahkan dalam melakukan diagnosis kerusakan komputer.
2. Dengan adanya sistem pakar yang telah dibangun ini dapat membantu karyawan laboratorium komputer di sekolah dalam melakukan diagnosis kerusakan komputer desktop di laboratorium komputer, menganalisis penyebab kerusakan pada setiap komponen komputer, menggunakan solusi perbaikan kerusakan dengan hasil tepat guna

Saran-Saran

Saran yang perlu disampaikan pada jurnal ini menyangkut sistem yang ada pada sistem pakar diagnosis kerusakan perangkat keras komputer *desktop* adalah sebagai berikut:

1. Melakukan pengembangan sistem secara berlanjut guna memenuhi kebutuhan sistem dimasa sekarang dan yang akan datang karena akan selalu terjadi perubahan dinamis.
2. Melakukan pengembangan sistem pakar ini, agar dapat memudahkan membuat laporan setiap hasil diagnosis kerusakan perangkat keras komputer *desktop*.

REFERENSI

- Yuswanto, *Pemrograman Database*, Prestasi Pustaka, Jakarta, 2007
- Verdi Yasin, *Rekayasa Perangkat Lunak Berorientasi Objek*, Mitra Wacana Media, Jakarta, 2007
- Kusrini, *Sistem Pakar*, Prestasi Pustaka, Jakarta, 2007
- Lim Rusyamsi, *Menjadi Dokter Spesialis Komputer*, Kawan Pustaka, 2009
- Tan Amelia, *Pemrograman Database Menggunakan Ado.Net*, Graha Ilmu, Yogyakarta, 2007
- Ying Bai, *Practical Database Programming with Visual_Basic.NET*, 2008
- Jim McKeown, *Programming in Visual Basic 2010 The Very Beginners Guide*, 2010