

Terbit online pada laman web jurnal: <http://jurnal.iaii.or.id>

JURNAL RESTI

(Rekayasa Sistem dan Teknologi Informasi)

Vol. 4 No. 3 (2020) 558 - 568

ISSN Media Elektronik: 2580-0760

Sistem Pakar Untuk Mengidentifikasi Kerusakan Perangkat PABX Panasonic NS1000 Dengan A* Pathfinding

Siswanto¹, Helmy Ligaputra², M. Anif³, Windu Gata⁴, Basuki Hari Prasetyo⁵

^{1,2,3,5}Teknik Informatika, Fakultas Teknologi Informasi, Universitas Budi Luhur, Jakarta, Indonesia

⁴Postgraduate of Computer Science, STMIK Nusa Mandiri, Jakarta, Indonesia

¹siswantodppiaii@gmail.com, ²muhammad.anif@budiluhur.ac.id, ⁴windu.gata@gmail.com, ⁵haribhopal01@yahoo.com

Abstract

Someone can be said to have the ability to handle damage to the Panasonic NS1000 PABX device and perform a configuration of at least 3 years or already have a certificate of competence and training. The problem to be investigated is the number of experts there is only one and can only identify 5 damage per day and 1 day there are 25 damage. PABX Panasonic NS1000 has 30 rules and 130 data symptoms of damage as its knowledge base. This Expert System is designed to identify damage to PABX Panasonic NS1000 mobile application-based applications using the A* (A star) pathfinding algorithm and the forward chaining method with the PHP (Jquery Mobile) programming language and the database using MySQL, as an application program used by PT. Mediatama Anugrah Citra to facilitate the process of troubleshooting and effective problem solving on target. The features in this expert system include the identification of problems, the identification process, up to the achievement of the goal state and solutions effectively, quickly and precisely, the user as the user or admin has its own login and access rights in identifying, entry and editing. In the programming case testing process it can be seen that the application of the A* Pathfinding algorithm with the heuristic function has proven ineffectively implemented in the expert system and the results of the UAT testing process, the respondents agree (above 91.23%) that overall the expert system helps the expert and can deduce damage to PABX devices correctly.

Keywords: Expert System, PABX Panasonic NS1000, A* Pathfinding Algorithm, UAT, JQuery Mobile

Abstrak

Seseorang dapat dikatakan mempunyai kemampuan menangani kerusakan perangkat PABX Panasonic NS1000 dan melakukan konfigurasi minimal 3 tahun atau telah mempunyai sertifikat kompetensi serta pelatihan. Masalah yang akan diteliti adalah jumlah pakarnya hanya ada satu serta hanya bisa mengidentifikasi 5 kerusakan per hari dan 1 hari ada 25 kerusakan. Kerusakan perangkat PABX Panasonic NS1000 ada 30 aturan (*rule*) dan 130 data gejala kerusakan sebagai *knowledge basenya*. Sistem Pakar ini dirancang untuk mengidentifikasi kerusakan pada perangkat PABX Panasonic NS1000 berbasis *mobile application* menggunakan algoritma A* (A star) pathfinding dan metode *forward chaining* dengan bahasa pemrograman PHP (Jquery Mobile) serta basis data menggunakan MySQL, sebagai program aplikasi yang dipakai oleh PT. Mediatama Anugrah Citra untuk mempermudah dalam melakukan proses *troubleshooting* dan *problem solving* yang efektif dan tepat sasaran. Fitur yang ada dalam sistem pakar ini diantaranya adalah identifikasi masalah, proses identifikasi, hingga pada pencapaian *goal state* serta solusi dengan efektif, cepat dan tepat, pengguna sebagai *user* ataupun *admin* mempunyai login dan hak akses tersendiri dalam melakukan identifikasi, *entry* dan *edit*. Pada proses uji kasus pemrograman dapat diketahui bahwa penerapan algoritma A* Pathfinding dengan fungsi *heuristic* terbukti tidak efektif diimplementasikan dalam sistem pakar dan hasil proses pengujian dengan UAT, para responden setuju (di atas 91,23%) bahwa secara keseluruhan sistem pakar membantu pakar serta dapat menyimpulkan kerusakan perangkat PABX dengan benar.

Kata kunci: Sistem Pakar, PABX Panasonic NS1000, Algoritma A* Pathfinding, UAT, JQuery Mobile

1. Pendahuluan

Dengan berkembangnya teknologi informasi dan komunikasi saat ini, berbagai perangkat penunjang dan medianya telah berhasil menjadi pilar utama dalam

setiap inovasinya, maka teknologi informasi sangat memegang peranan yang sangat penting. Seseorang dapat dikatakan mempunyai kemampuan dalam melakukan konfigurasi minimal telah berkecimpung

Diterima Redaksi : 13-04-2020 | Selesai Revisi : 16-06-2020 | Diterbitkan Online : 20-05-2020

selama 3 tahun dalam bidangnya atau telah mempunyai sertifikat kompetensi ataupun pelatihan menangani kerusakan perangkat PABX Panasonic NS1000. Oleh karena itu menjadi sebuah kebutuhan dengan adanya sebuah sistem pakar yang dapat membantu semua kebutuhan operasional sistem tersebut.

Masalah yang akan diteliti adalah jumlah pakarnya hanya ada satu serta hanya bisa mengidentifikasi 5 kerusakan per hari dan 1 hari ada 25 kerusakan. Kerusakan perangkat PABX Panasonic NS1000 ada 30 aturan (*rule*) dan 130 data gejala kerusakan sebagai *knowledge basenya*.

Sistem Pakar ini dirancang untuk mengidentifikasi kerusakan pada perangkat PABX Panasonic NS1000 sebagai program aplikasi yang dipakai oleh PT. Mediatama Anugrah Citra untuk mempermudah dalam melakukan proses *troubleshooting* dan *problem solving* yang efektif dan tepat sasaran. Metode algoritma sistem pakar ini menggunakan algoritme A* (*A star*) *pathfinding* dan metode *forward chaining* berbasis *mobile application* dengan bahasa pemrograman PHP (*Jquery Mobile*) dan basis data menggunakan MySQL.

PT. Mediatama Anugrah Citra bergerak dalam bidang jasa layanan TV berbayar berbasis *digital terrestrial* hadir dengan produknya bernama *Nexmedia*, salah satu teknologi yang menyajikan tayangan hiburan keluarga. Seiring dengan peningkatan jumlah pelanggan maka sudah menjadi kewajiban penyedia layanan untuk selalu meningkatkan kualitas layanan demi kepuasan para pelanggan. Kualitas layanan memerlukan koordinasi tim yang handal dan bagian dari kualitas koordinasi ditentukan oleh komunikasi yang baik, oleh karena itu PT. Mediatama Anugrah Citra mempercayakan kepada Panasonic IT Comm sebagai vendor tunggal dalam penyediaan perangkat komunikasi PABX dengan produknya IP-PBX NS1000 [1].

Dalam kegiatan operasionalnya perangkat PABX ini berada di bawah infrastruktur departemen IT dan support dari Panasonic IT Comm sebagai vendor. Setiap user PT. Mediatama Anugrah Citra yang melakukan komunikasi baik di kantor pusat maupun cabang akan terhubung dengan jaringan IP-PBX ini sehingga dengan teknologi VoIP, maka biaya komunikasi menjadi lebih hemat [2].

PABX dan *VoIP* adalah salah satu bentuk dari inovasi perangkat komunikasi dan telah dipakai di berbagai perusahaan sebagai sarana untuk menunjang komunikasi dalam menjalankan bisnisnya. Berbagai jenis dan *type* telah diproduksi oleh beberapa produsen seperti Panasonic yang setiap produknya memiliki ciri khas dan *feature* yang berbeda-beda sebagai bentuk dari persaingan inovasi secara global. Komunikasi server yang digunakan untuk bisnis harus memberikan solusi untuk kebutuhan bisnis nyata dunia saat ini seperti: menyederhanakan dan meningkatkan komunikasi, menurunkan biaya, meningkatkan produktivitas,

mendukung SOHOs dan diversifikasi dari tempat kerja. [3].

IP-PBX Panasonic KX NS1000 merupakan salah satu produk unggulan dari Panasonic dengan *type* yang paling baru. *Server* komunikasi jaringan IP dan SIP. Olah pesan terpadu hingga 24 *port* per unit. Bangun sistem untuk hingga 8.000 pengguna. Sistem cadangan yang handal untuk mengatasi kegagalan sistem [4].

Sistem pakar (*expert system*) yaitu program-program yang bertingkah laku seperti manusia pakar/ahli (*human expert*). Sistem pakar atau sistem berbasis pengetahuan paling banyak aplikasinya dalam membantu menyelesaikan masalah-masalah dalam dunia nyata. Komputer berbasis sistem pakar adalah program komputer yang mempunyai pengetahuan yang berasal dari manusia yang berpengetahuan luas (pakar) dalam *domain* tertentu, dimana pengetahuan disini adalah pengetahuan manusia yang sangat minim penyebarannya, mahal serta susah didapat. Disini keahlian manusia dimasukan ke pengetahuan tersebut untuk menyelesaikan masalah, seperti yang dikatakan manusia [5].

Algoritma A* adalah algoritma pencarian yang merupakan pengembangan dari kelas algoritma *Greedy*. Seperti halnya pada *Greedy*, untuk menemukan solusi, A* juga dilakukan oleh fungsi heuristik, yang menentukan urutan titik mana yang akan dikunjungi terlebih dahulu. Heuristik merupakan penilai yang memberi harga pada tiap verteks yang memandu A* mendapatkan solusi yang diinginkan. Algoritma A* menyelesaikan masalah yang menggunakan *graph* untuk perluasan ruang statusnya. Dengan kata lain digunakan untuk menyelesaikan permasalahan yang bisa direpresentasikan dengan *graph*. Algoritma A* adalah sebuah algoritma yang telah dikembangkan. Dengan menerapkan fungsi heuristik, algoritma ini membuang langkah-langkah yang tidak perlu dengan pertimbangan bahwa langkah-langkah yang dibuang sudah pasti merupakan langkah yang tidak akan mencapai solusi yang diinginkan [6].

AI yang digunakan dalam pembangunan *game* petualangan labirin ini adalah algoritma A* (*A Star*) dengan *euclidean distance*. Algoritma ini merupakan algoritma pencarian untuk menemukan *route* terpendek dengan *cost* paling minimum, algoritma A* mencari *route* terpendek dengan menjumlahkan jarak sebenarnya dengan jarak perkiraan, sehingga membuatnya optimum dan *complete*. *Genre* dari *game* ini adalah *adventure* dan *puzzle*, dibangun dengan bahasa pemrograman java dengan *tools android studio*, AI yang digunakan adalah algoritma A* dengan *euclidean distance* yang digunakan pada bantuan untuk melakukan pencarian jalur guna menemukan makanan kelinci. Hasil uji dari *game* petualangan labirin ini adalah jika pemain dalam kesusahan menemukan jalur menuju makanan kelinci, maka pemain dapat menggunakan tombol bantuan yang

akan dicarikan jalur terpendek oleh algoritma A* (*A Star*) dengan *euclidean distance* untuk menuju lokasi tempat makanan kelinci berada [7].

Di Bandar Lampung setidaknya terdapat 194 rumah makan yang dicatat oleh Dinas Pariwisata dan Ekonomi Kreatif Provinsi Lampung pada tahun 2016. Kurangnya informasi yang akurat mengenai informasi jalur terdekat menuju wisata kuliner dan informasi mengenai kuliner yang terdapat di sekitar menjadikan masyarakat terkadang kesulitan dalam mencari dan menentukan tempat makan. Dalam menyelesaikan permasalahan penentuan jalur terdekat dapat diselesaikan dengan beberapa algoritma salah satunya adalah A* (*A Star*). Hasil dari penelitian ini nantinya adalah membuat sebuah aplikasi *mobile android* untuk memberikan informasi mengenai jalur terdekat menuju wisata kuliner dan juga memberikan informasi mengenai kuliner yang terdapat di sekitar pengguna [8].

Optimasi pencarian jalur dengan metode *A-Star* studi kasus: area Gading Serpong, Tangerang, algoritma pencarian jalur menggunakan metode A* (*A-star*) dengan proses *backtrack*. Metode A* untuk mencari jalur terpendek dari tempat awal ke tempat tujuan, dan jika terjadi macet, *agent* menggunakan proses *backtrack* untuk mencari jalur terpendek kedua, dan seterusnya. Penggunaan metode A* tidak menjamin bahwa rute yang terpilih adalah rute terbaik atau paling optimal (dilihat dari $t = 0$), karena *agent* tidak bersifat *full observable* setelah proses *backtrack* dilakukan (menutup *node* yang macet), sehingga hanya menghitung jarak dari *node* yang dilaluinya saja [9].

Pencarian *route* terdekat pada labirin menggunakan metode A*, proses *backtrack* yang digunakan oleh module *agent* tidak hanya dilakukan dalam melakukan pencarian jalur terpendek, tetapi juga dilakukan ketika menemui kemacetan pada jalur yang dilewati dan menemui jalan buntu, sehingga *agent* harus bergerak mundur dan memilih jalur alternatif lain [10].

*Investigation of the * (Star) Search Algorithms Characteristics, Methods, and Approaches*, algoritma A* tersebut mengalami perubahan, khususnya saat perluasan *node* atau *Node Expansion*, yaitu saat memindai jalur atau *link*. Karena A* merupakan *best-first search*, semua *node* memenuhi pertidaksamaan $cost(i) + estimated\ Cost(i,d)$, $cost(d)$ *Node* yang memenuhi pertidaksamaan di atas diperiksa sebelum algoritma diterminasi atau sebelum *node* tujuan diperiksa [11].

Penerapan *JQuery Mobile* dan PHP data *object* pada aplikasi pencarian lokasi tempat ibadah di Yogyakarta. Dalam metode aplikasinya menggunakan *JQuery Mobile* yang mempunyai sifat memudahkan pengembangan *user-interface* untuk *mobile web apps*, Konfigurasi antarmukanya bersifat *markup-driven*, dalam kode HTML, tanpa perlu menulis satu baris

JavaScript dan menggunakan theme untuk memudahkan penyesuaian tampilan aplikasi [12].

Algoritma A* (*A Star*) sebagai salah satu contoh metode pemrograman *branch and bound*. Beberapa terminologi dasar yang terdapat pada algoritma ini adalah *starting point*, simpul (*nodes*), *A*, *open list*, *closed list*, harga (*cost*), halangan (*unwalkable*) [13].

Jurnal ini berjudul *A Optimization of A* Algorithm to Make it Close To Human Pathfinding Behavior*. Dalam pembahasannya disebutkan bahwa terdapat beberapa strategi dalam optimalisasi Algoritma *A star* yang didasarkan pada pendekatan dengan kebiasaan manusia. Pada prosesnya tidak semua *Neighbor Node* selalu diakses, hanya *node-node* yang menjadi *Key Position* antara *Start* dan *Goal* saja yang harus diakses [14].

Rancang bangun aplikasi sistem pakar untuk kerusakan komputer dengan metode *backward chaining*, yang di mulai dari sekumpulan hipotesis gejala kerusakan menuju fakta-fakta yang mendukung hipotesa tersebut, hasilnya akan memberikan informasi kepada pemakai komputer bagaimana mengenali dan menangani kerusakan komputer [15].

Aplikasi *expert system* pengembangan karir menggunakan *inventory* kepribadian *entrepreneurship*, ditemukannya 4 pengelompokan tipe kepribadian berdasarkan korelasi indikator-indikator yang diperoleh dari hasil analisis faktor menggunakan *confirmatory factor analysis* adapun 4 tipe kepribadian yang dibentuk dari hasil analisis faktor ini adalah *Extrovert*, *Leader*, *Moderate Risk Taker* dan *Ambitious*. Aplikasi *Expert system* Pengembangan Karir menggunakan Kepribadian *Entrepreneurship* dinyatakan *Valid* (0.887), Sangat Praktis (91.11) dan efektif (82,47) [16].

Sistem pakar identifikasi modalitas belajar siswa menggunakan metode *forward chaining*, sistem pakar yang diolah dalam penelitian ini diambil dari kepakaran guru bimbingan konseling Sekolah Menengah Atas Negeri 1 Tilatang Kamang serta dapat direkomendasikan untuk membantu guru dan siswa dalam meningkatkan cara belajar siswa yang tepat [17].

User Acceptance Test (UAT) adalah suatu proses pengujian yang dilakukan oleh pengguna dengan hasil output sebuah dokumen hasil uji yang dapat dijadikan bukti bahwa *software* atau *sistem* sudah diterima dan sudah memenuhi kebutuhan yang diminta dan mewakili keinginan *user* serta memecahkan masalah yang dihadapi *user*. UAT tidak jauh beda dengan kuesioner pada tahap awal pembuatan aplikasi [18].

User acceptance testing (UAT) merupakan pengujian yang ditujukan di luar sistem yaitu *user*. Tujuan dari *user acceptance testing* adalah untuk mengetahui kelayakan dari perangkat lunak [19].

Pada penelitian sebelumnya, UAT dilakukan dengan metode *survey* yaitu dengan menyebar kuesioner kepada

pengguna (petugas TPHD) yang sebelumnya sudah diberikan tutorial penggunaan sistem layanan haji. Model kuesioner menggunakan *likert scale* dengan skala 5 yaitu *strongly agree; agree; neutral/undecided; disagree; strongly disagree*. UAT digunakan untuk menjawab permasalahan perangkat lunak seputar *system metric; usability; satisfaction* dan beberapa *setting* pada masing – masing fungsi/fitur [20].

2. Metode Penelitian

Adapun metode perancangan yang digunakan adalah model *waterfall* dengan langkah - langkah berikut:

2.1 Identifikasi Masalah

Masalah yang akan diteliti adalah jumlah pakarnya hanya ada satu serta hanya bisa mengidentifikasi 5 *kerusakan* per hari dan 1 hari ada 25 kerusakan PABX Panasonic NS1000.

2.2 Analisa Kerusakan Pada Sebuah Jaringan IP-PBX

Kerusakan perangkat PABX Panasonic NS1000 ada 30 aturan (*rule*) dan 130 data gejala kerusakan sebagai *knowledge basenya*, selanjutnya adalah melakukan pengumpulan data mengenai 130 fakta-fakta/data gejala kerusakan yang terjadi pada sistem IP-PBX yang nantinya akan dimasukkan ke dalam *knowledge base*.

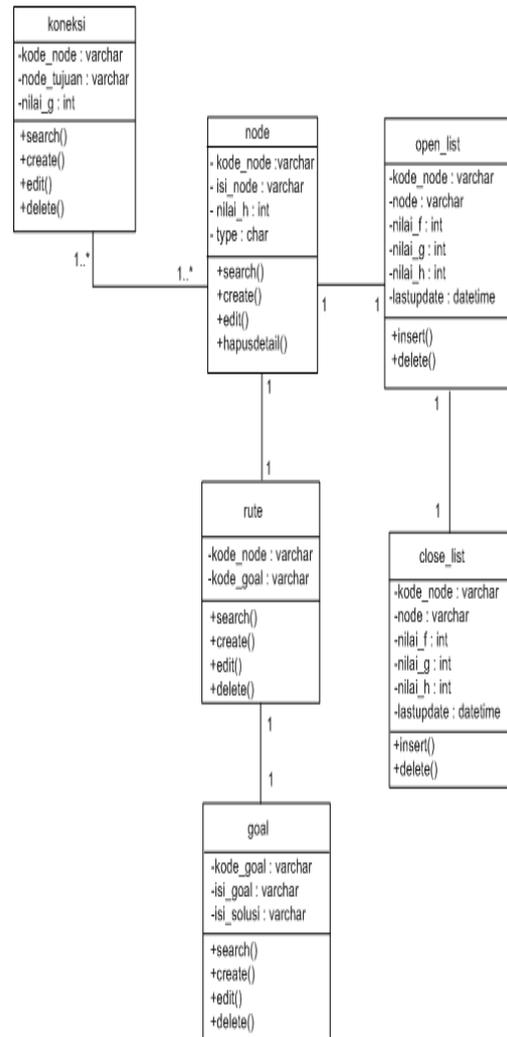
Setelah memahami bidang yang akan dibuat sistem pakar berbasis *mobile application*, selanjutnya menentukan metode representasi *knowledge* yaitu: kaidah produksi untuk memperoleh *knowledge base* sistem pakar ini. class *diagram* yang digunakan dalam sistem pakar ini dapat dilihat pada Gambar 1.

Fitur yang ada dalam sistem pakar ini diantaranya adalah identifikasi masalah, proses identifikasi, hingga pada pencapaian *goal state* serta solusi dengan efektif, cepat dan tepat, pengguna sebagai *user* ataupun *admin* mempunyai login dan hak akses tersendiri dalam melakukan identifikasi, *entry* dan *edit*.

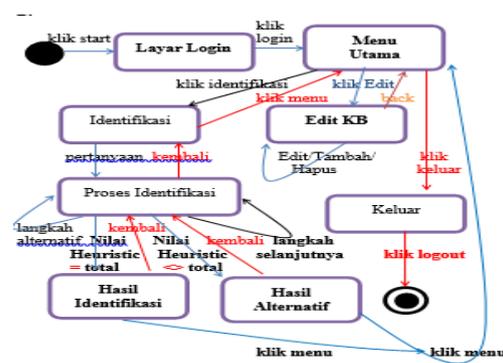
Statechart diagram sistem pakar ini dilihat pada Gambar 2.

2.3 Memilih Teknik Inferensi Pengetahuan

Setelah mengetahui dan memahami masalah dan konsep pengetahuan yang akan dikembangkan ke dalam sistem pakar maka tahapan selanjutnya menentukan teknik inferensinya. Teknik inferensi yang akan dipakai dalam pembuatan sistem pakar ini adalah teknik pelacakan ke belakang (*Backward Channing*) dengan menggunakan teknik penelusuran, yaitu Algoritme A * (*A Star*) *pathfinding*.



Gambar 1. Class Diagram

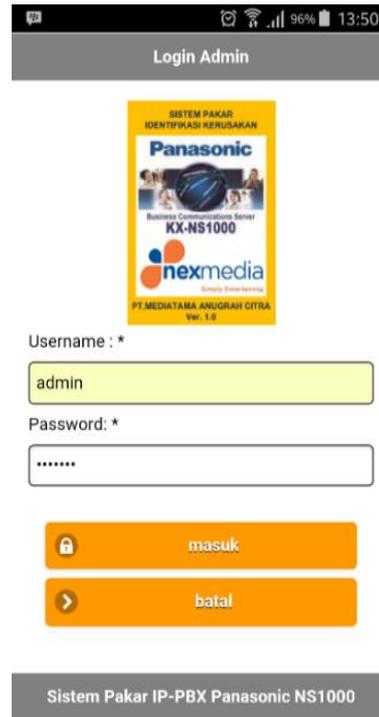


Gambar 2. Statechart Diagram Sistem Pakar

Algoritme A * (*A Star*) *pathfinding*:

1. Ambil parameter *kode_node*
2. **If** parameter = "langkah1" **Then**
3. Kosongkan tabel *open_list*
4. **If** *kode_node* berawalan "A" **Then**
5. Simpan *kode_node*, dengan nilai *f*=0, nilai *g*=0, nilai *h*=0, *lastupdate* = *now()* ke dalam tabel *close_list*
6. Cari data dari tabel *koneksi.node_tujuan* dimana *koneksi.node_tujuan* sama dengan sama dengan *node.kode_node*

```
7.      If kode_node = "ada" Then
8.          Ambil kode_node dari tabel koneksi kemudian
          cari kode_node, nilai_h pada tabel node
9.          Cari nilai_g dari tabel close_list diurutkan
          secara descending dengan Limit 1
10.         If nilai_g = "ada" then
11.             Nilai_g = nilai_g sebelumnya
12.         Else
13.             Nilai_g_sebelumnya = 0
14.         Endif
15.         Lakukan perhitungan Nilai_f =
          nilai_g sebelumnya + nilai_g + nilai_h
16.         Simpan data kode_node, nilai_f, nilai_g,
          nilai_h, lastupdate ke dalam tabel Open_list
17.         Cari kode_node, nilai_f, nilai_g, nilai_h
          lastupdate dari tabel open_list berdasarkan
          nilai_f secara ascending limit 1
18.         Simpan kode_node, nilai_f, nilai_g, nilai_h,
          lastupdate ke dalam tabel close_list
19.         Jalankan layar proses proses identifikasi
20.     Endif
21.     Endif
22.     Endif
23.     If Parameter = "Langkah2" Then
24.         Hapus kode_node dari tabel close_list
25.         Ambil parameter kode_node dari tabel open_list
26.     Endif
        Kembali ke baris 15
```



Gambar 3. Tampilan Layar Login

2.4 Pembuatan Dan Uji Coba Sistem Pakar

Setelah membuat struktur *knowledge base* dan menentukan teknik mesin inferensi selanjutnya adalah pembuatan sistem pakar.

Sistem pakar untuk mengidentifikasi kerusakan perangkat PABX *Panasonic NS1000* dibuat dengan menggunakan bahasa pemrograman *PHP Framework* dengan *Framework JQuery Mobile*, sedangkan untuk basis datanya menggunakan *MySQL*. Adapun hasil implementasi dari sistem pakar ini adalah dalam bentuk *Mobile Application* dengan format *.apk*, sehingga dapat diinstall pada perangkat *smartphone*.

Selanjutnya menguji sistem pakar yang telah dibuat dengan metode *User Acceptance Test (UAT)*, agar dapat diketahui apakah sistem pakar layak atau sesuai dengan menggunakan tatap muka pemakai berbasis *mobile application*. Tampilan layar *Login Pakar* seperti Gambar 3 terdapat *form login* untuk *admin* dapat masuk ke menu *admin*. Berikut ini adalah keterangan tampilan layar login pakar yang ada pada *file login.php*.

Layar hasil dari proses identifikasi yang telah dilakukan. Terdapat beberapa informasi yang ditampilkan yaitu :

- 1) Informasi permasalahan yang dihadapi sesuai yang dipilih pada layar identifikasi awal.
- 2) Informasi serangkaian pengecekan gejala yang timbul dan telah dipastikan oleh *user*.
- 3) Informasi hasil kesimpulan.
- 4) Informasi solusi yang dapat dilakukan.
- 5) Informasi nilai *heuristic* yang merupakan total nilai pelacakan optimal dari sistem pakar dalam penelusuran simpul-simpul identifikasi dalam ruang pencarian (*State space*) dari mulai simpul *Start State* sampai kepada simpul *Goal State*.

3. Hasil dan Pembahasan

3.1 Pengujian Algoritma dan Rule Program

Logika pemrograman sistem pakar ini menggunakan metode algoritma *A*(A star)* dengan pelacakan *heuristic backward channing* pada *graph* berarah di mana setiap *graph* mempunyai masing-masing ruang status atau disebut juga dengan *state space* dengan bobotnya sebagai nilai *heuristic*. Ruang status seperti Gambar 5 tersebut terdiri dari *Start State*, *Goal State*, *Node*, nilai *hypotesis* $h(n)$, nilai *geographical* $g(n)$ dan lapisan (*layer*) yang menyatakan kategori setiap *node*.

Setiap *Graph* akan membentuk perluasan ruang statusnya berdasarkan data dari *matrix* yang telah dibuat. Berikut adalah salah satu *graph* yang kita ambil sebagai contoh dalam uji kasus, yaitu *graph 1* sebagaimana ditunjukkan pada Gambar 6 dengan referensi Tabel *graph 1* seperti Tabel 1.

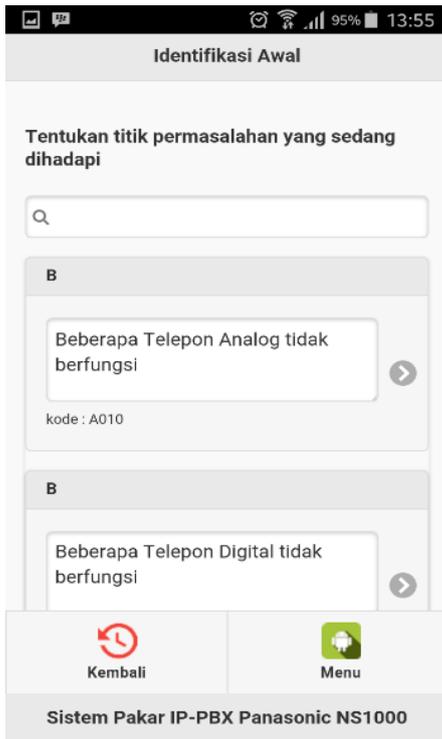
Proses pelacakan algoritma *A Star* diuji dalam program Sistem Pakar dengan cara menjalankan program meliputi 3 hal yaitu :

1) *Start State*

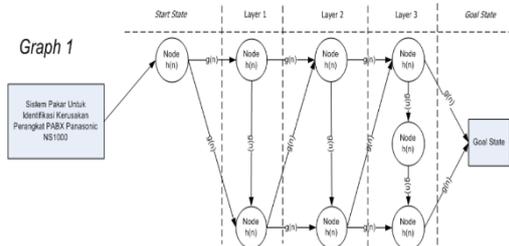
Sistem akan menelusuri *node* atau simpul yang merupakan *start state* dimana simpul tersebut berisikan kondisi awal dari permasalahan, semua *node* yang berkategori *Start State* mempunyai nilai *hypotesisnya* adalah 0 (nol) dan mempunyai kode awal A nilai $h(A010)=0$, nilai $h(A020)=0$, nilai $h(A30)=0$, ..., nilai $h(A(n))=0$, maka hasilnya adalah $[A010, A020, A030, \dots, A(n)]$, ditampilkan pada layar identifikasi awal

secara *list*, *user* tinggal memilih *list* data sesuai dengan permasalahan yang dihadapi.

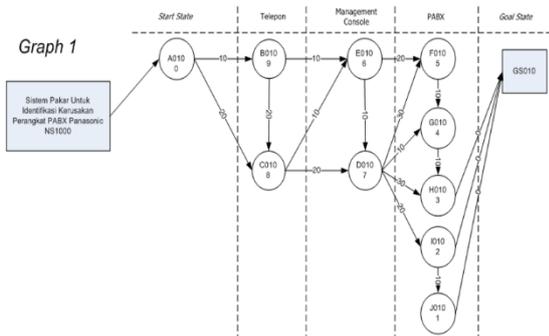
CLOSE_LIST : [A010]



Gambar 4. Tampilan Layar Identifikasi Awal



Gambar 5. Graph Ruang Status (State Space Graph)



Gambar 6. Graph 1

Jika yang dipilih adalah [A010] maka, node [A010] akan dipindahkan ke OPEN_LIST, dikarenakan hanya ada satu *node* maka *node* [A010] dengan nilai $h(A010)=0$, terpilih menjadi *node* terbaik dan dipindahkan ke CLOSE_LIST.

OPEN_LIST : []

Tabel 1. Tabel *graph* 1

Kode Node	Nilai h(n)	Kategori	Isi Node
A010	0	Start State	Beberapa Telepon Analog tidak berfungsi
B010	9	Telepon	Tidak terdengar nada tone pada pesawat telepon
C010	8	Telepon	Tidak terdengar nada panggilan saat menghubungi nomor extension
D010	7	Web Console	License ip-trunk masih dalam kapasitas
E010	6	Web Console	Card slc "not available" dalam virtual slot legacy
F010	5	PABX	PABX dalam konsidi ON
G010	4	PABX	Minor Alarm menyala
H010	3	PABX	Card SLC Alarm
I010	2	PABX	Cabut pasang card alarm masih menyala
J020	1	PABX	Interkoneksi Legacy dengan controller normal
GS010	-	Goal State	Card SLC Bermasalah

2) *Movement Cost* Proses Identifikasi

a) Langkah Pertama

Node A010 sudah berada di dalam CLOSE_LIST, kemudian *node* suksesor A010 dibangkitkan dan menghasilkan *node* [B010] dan *node* C010 (terlampir pada *Graph* 1), kemudian kedua *node* ini dimasukkan di OPEN_LIST.

OPEN_LIST : [B010, C010]
 CLOSE_LIST : [A010]

Setelah kedua *node* suksesor berada di OPEN_LIST, maka dilakukan perhitungan nilai heuristik dengan Rumus yang digunakan pada algoritme A* seperti rumus (1).

$$f(n) = g(n) + h(n) \dots\dots\dots(1)$$

dimana $f(n)$ = biaya(*cost*) yang dibutuhkan,
 $g(n)$ =biaya yang ditempuh dari *node* asal/nilai geographical,
 $h(n)$ =nilai perkiraan dari *node* saat ini ke tujuan/nilai hypotesis,
 n =*node* yang dikunjungi.

Nilai *geographical awal* $g(A010) = 0$, untuk nilai *geographical* berikutnya: $g(B010)=0+10$, untuk $g(C010)=0 +20$, untuk $g(D010)=0 +30$, dan seterusnya.

$$f[B010] = g[B010] + h[B010]$$

$$f[B010] = 0 + 10 + 9$$

$$f[B010] = 10 + 9$$

$$f[B010] = 19$$

$$f[C010] = g[C010] + h[C010]$$

$$f[C010] = 0 + 20 + 8$$

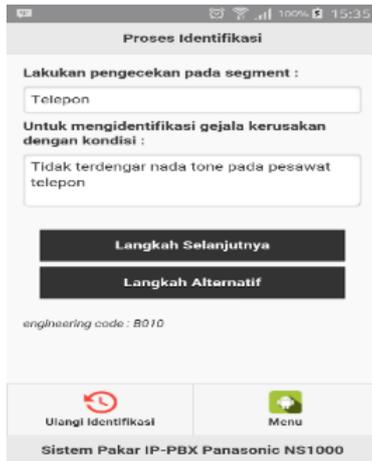
$$f[C010] = 20 + 8$$

$$f[C010] = 28$$

Node [B010] terpilih menjadi node terbaik dengan nilai f terkecil *dibandingkan* dengan node [C010] oleh karena itu node [B010] dengan parent [A010] dipindahkan ke CLOSE_LIST

OPEN_LIST : [C010]
 CLOSE_LIST : [A010, B010]

Node [B010] akan dijadikan parameter untuk dilihat isi node tersebut dan dimunculkan layar proses identifikasi seperti Gambar 7.



Gambar 7. Tampilan Layar Langkah Pertama

Tampilan layar proses identifikasi memungkinkan user untuk melihat sebuah instruksi dari sistem pakar untuk melakukan *langkah* identifikasi dan memastikan sebuah kondisi atau gejala yang terjadi.

Jika user telah memastikan kondisi tersebut maka user dapat memilih tombol “Langkah Selanjutnya” untuk mengetahui langkah terbaik apa yang harus diambil, atau jika user tidak *dapat* memastikan kondisi atau gejala sesuai dengan yang diinstruksikan sistem pakar, user dapat memilih “Langkah Alternatif”.

Jika user memilih tombol “Langkah Selanjutnya”, maka node di dalam tabel OPEN_LIST akan dieliminasi (*Prunning*)

OPEN_LIST : []
 CLOSE_LIST : [A010, B010]

b) Langkah Kedua

Node B010 sudah berada di dalam CLOSE_LIST, kemudian node suksesor B010 dibangkitkan dan menghasilkan node C010 dan node E010 (tertera pada *Graph 1*) kedua node ini dimasukkan di OPEN_LIST.

OPEN_LIST : [C010, E010]
 CLOSE_LIST : [A010, B010]

Setelah kedua node suksesor berada di OPEN_LIST, maka dilakukan perhitungan nilai heuristik dengan rincian sebagai berikut :

$$f[C010] = g[C010] + h[C010]$$

$$\begin{aligned} f[C010] &= 10 + 20 + 8 \\ f[C010] &= 30 + 8 \\ f[C010] &= 38 \\ f[E010] &= g[E010] + h[E010] \\ f[E010] &= 10 + 20 + 6 \\ f[E010] &= 20 + 6 \\ f[E010] &= 26 \end{aligned}$$

Hasil perhitungan menunjukkan node E010 terpilih menjadi node terbaik dengan nilai f terkecil dibandingkan dengan node C010 oleh karena itu node E010 dengan parent B010 dipindahkan ke CLOSE_LIST

OPEN_LIST : [C010]
 CLOSE_LIST : [A010, B010, E010]

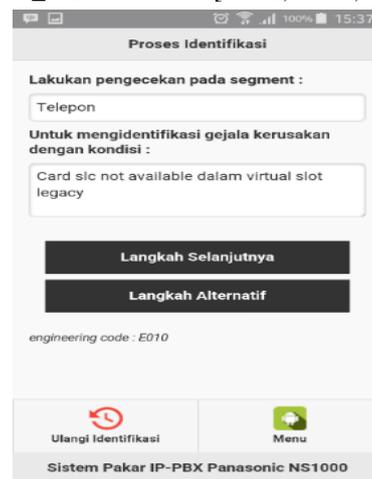
Node E010 akan dijadikan parameter untuk dilihat isi node tersebut dan dimunculkan layar proses identifikasi seperti Gambar 8.

Tampilan layar proses identifikasi kedua user dapat melihat sebuah instruksi dari sistem pakar untuk melakukan langkah identifikasi dan memastikan sebuah kondisi atau gejala yang terjadi.

Jika user telah memastikan kondisi tersebut maka user dapat memilih tombol “Langkah Selanjutnya” untuk mengetahui langkah terbaik apa yang harus diambil, atau jika user tidak dapat memastikan kondisi atau gejala sesuai dengan yang diinstruksikan sistem pakar, user dapat memilih “Langkah Alternatif”.

Jika user memilih tombol “Langkah Selanjutnya”, maka node di dalam tabel OPEN_LIST akan dieliminasi (*Prunning*)

OPEN_LIST : []
 CLOSE_LIST : [A010, B010, E010]



Gambar 8. Tampilan Layar Langkah Kedua

Jika user memilih tombol “Langkah Selanjutnya”, maka node di dalam tabel OPEN_LIST akan dieliminasi (*Prunning*)

OPEN_LIST : []
 CLOSE_LIST : [A010, B010, E010]

c) Langkah Ketiga

Node D010 sudah berada di dalam CLOSE_LIST, kemudian node suksesor D010 dibangkitkan dan menghasilkan node F010 dan node D010 (tertera pada Graph 1) kedua node ini dimasukkan di OPEN_LIST.

OPEN_LIST : [F010, D010]
 CLOSE_LIST : [A010, B010, E010]

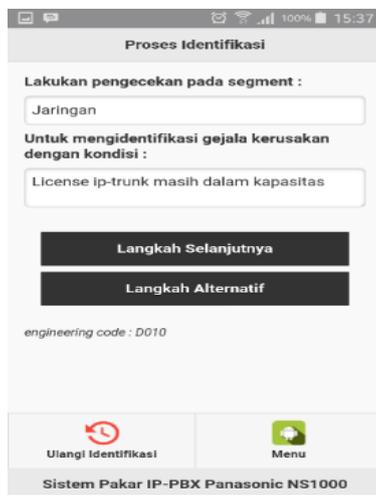
Setelah kelima node suksesor berada di OPEN_LIST, maka dilakukan perhitungan nilai heuristik dengan rincian sebagai berikut :

$$\begin{aligned}
 f[F010] &= g[F010] &+& & h[F010] \\
 f[F010] &= 10 + 10 + 20 &+& & 5 \\
 f[F010] &= 40 &+& & 5 \\
 f[F010] &= 45 \\
 \\
 f[D010] &= g[D010] &+& & h[D010] \\
 f[D010] &= 10 + 10 + 10 &+& & 7 \\
 f[D010] &= 30 &+& & 7 \\
 f[D010] &= 37
 \end{aligned}$$

Hasil perhitungan menunjukkan node D010 terpilih menjadi node terbaik dengan nilai f terkecil dibandingkan dengan node F010 oleh karena itu node E010 dengan parent B010 dipindahkan ke CLOSE_LIST

OPEN_LIST : [F010]
 CLOSE_LIST : [A010, B010, E010, D010]

Node D010 akan dijadikan parameter untuk dilihat isi node tersebut dan dimunculkan layar proses identifikasi seperti gambar 9.



Gambar 9. Tampilan Layar Langkah Ketiga

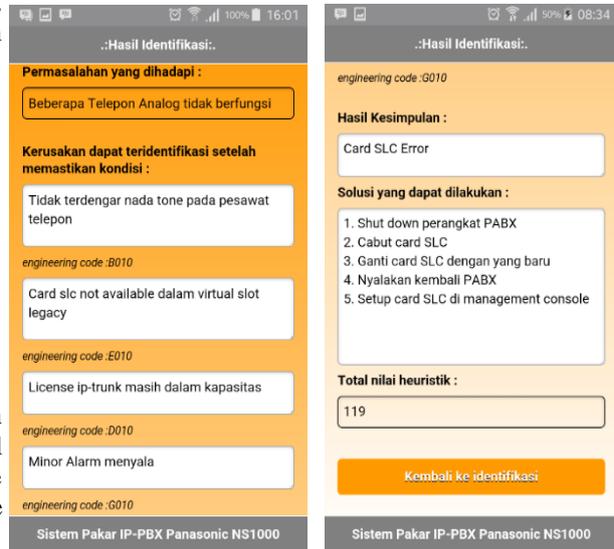
Jika user memilih tombol “Langkah Selanjutnya”, maka node di dalam tabel OPEN_LIST akan dieliminasi (*Pruning*)

OPEN_LIST : []
 CLOSE_LIST : [A010, B010, E010, D010]

Langkah Keempat dan Langkah Kelima dilakukan sama seperti tiga Langkah sebelumnya.

3) Goal State

Jika node sudah mengarah ke goal state maka sistem secara otomatis menampilkan layar Hasil Identifikasi dengan melakukan algoritma runut balik node terakhir yang dikunjungi sampai ke start state node atau simpul akar, seperti pada gambar 10.



Gambar 10. Tampilan Layar Hasil identifikasi

3.2 Pengujian User Acceptance Test (UAT)

Pengujian UAT melibatkan 25 responden pengguna Sistem pakar untuk mengidentifikasi kerusakan perangkat PABX Panasonic NS1000. Para responden menjawab kuesioner setelah menggunakan Sistem pakar untuk mengidentifikasi kerusakan perangkat PABX Panasonic NS1000 berbasis mobile application. Pada Tabel 2 mempresentasikan daftar pertanyaan survei kuesioner yang terdiri dari 4 bagian: *setting* fungsi; *system metric*; *user satisfaction*; dan *usability*.

Tabel 2. Daftar Pertanyaan Survei Kuesioner

No	Daftar Pertanyaan
1.	Apakah Tampilan Sistem Pakar berbasis <i>mobile application</i> menarik?
2.	Apakah Menu-menu Sistem pakar berbasis <i>mobile application</i> ini mudah dipahami?
3.	Apakah Sistem pakar berbasis <i>mobile application</i> ini mudah dioperasikan?
4.	Apakah Sistem pakar berbasis <i>mobile application</i> ini Responsive?
5.	Apakah Performa Sistem pakar berbasis <i>mobile application</i> ini baik?
6.	Apakah Sistem pakar berbasis <i>mobile application</i> dapat mengidentifikasi kerusakan perangkat PABX Panasonic NS1000?
7.	Apakah Fitur-fitur Sistem pakar berbasis <i>mobile application</i> ini sudah cukup baik?
8.	Apakah keluaran dari Sistem pakar berbasis <i>mobile application</i> sudah sesuai hasil identifikasi dengan pakar kerusakan perangkat PABX Panasonic NS1000 yang melakukannya?

Pertanyaan 1 dan 2 merupakan fokus *setting* fungsi dari sistem pakar berbasis *mobile application*. Pertanyaan 3 sampai dengan 5 merupakan fokus *system metric* dari sistem pakar berbasis *mobile application*. Pertanyaan 6 merupakan fokus *user satisfaction terhadap* sistem pakar berbasis *mobile application*. Pertanyaan 7 dan 8 merupakan fokus *usability terhadap* Sistem pakar berbasis *mobile application*.

Sistem pakar untuk mengidentifikasi kerusakan perangkat PABX *Panasonic NS1000* berbasis *mobile application* yang akan diimplementasikan untuk mengetahui tanggapan responden (*user*), maka dilakukan pengujian dengan memberikan 8 pertanyaan kepada 25 responden di mana jawaban dari pertanyaan tersebut terdiri dari tingkatan yang dapat dipilih disertai bobotnya, seperti Tabel 3.

Tabel 3. Tabel Bobot Nilai Jawaban UAT

Jawaban UAT	Bobot
A Sangat: Mudah/Baik/Sesuai/Jelas/Menarik/Paham	5
B Mudah/Baik/Sesuai/Jelas/Menarik/Paham	4
C Netral	3
D Cukup: Sulit/cukup Baik/Tidak Sesuai/Tidak Jelas/Tidak Menarik/Tidak Paham/Tidak Setuju	2
E Sangat: Sulit/Jelek/Tidak Sesuai/Tidak Jelas/Tidak Menarik/Tidak Paham/Tidak Setuju	1

Tabel 4 merupakan hasil UAT yang melibatkan pengguna sistem pakar untuk mengidentifikasi kerusakan perangkat PABX *Panasonic NS1000* berbasis *mobile application*. Sebanyak 25 responden melakukan evaluasi dalam pengisian kuesioner. Diperoleh hasil kuesioner dalam bentuk *likert scale* yang akan dianalisis.

Tabel 4. Hasil UAT

Pertanyaan	Pilihan Jawaban				
<i>Setting Fungsi</i>	A	B	C	D	E
Apakah Tampilan Sistem Pakar berbasis <i>mobile application</i> menarik?	10	13	1	1	0
Apakah Menu-menu Sistem pakar berbasis <i>mobile application</i> ini mudah dipahami?	14	8	1	2	0
<i>System Metric</i>	A	B	C	D	E
Apakah Sistem pakar berbasis <i>mobile application</i> ini mudah dioperasikan?	13	9	2	1	0
Apakah Sistem pakar berbasis <i>mobile application</i> ini Responsive?	18	6	1	1	0
Apakah Performa Sistem pakar berbasis <i>mobile application</i> ini baik?	19	5	1	0	0
<i>User Satisfaction</i>	A	B	C	D	E
Apakah Sistem pakar berbasis <i>mobile application</i> dapat mengidentifikasi kerusakan perangkat PABX <i>Panasonic NS1000</i> ?	20	2	3	0	0
<i>Usability</i>	A	B	C	D	E
Apakah Fitur-fitur Sistem pakar berbasis <i>mobile application</i> ini sudah cukup baik?	19	4	1	1	0
Apakah keluaran dari Sistem pakar berbasis <i>mobile application</i> sudah	18	6	1	0	0

sesuai hasil identifikasi dengan pakar kerusakan perangkat PABX *Panasonic NS1000*.yang melakukannya?

Tabel 5 merupakan hasil perkalian masing-masing jawaban UAT dikalikan dengan masing-masing bobot nilai jawaban UAT

Tabel 5. Hasil UAT x Bobot Nilai

Pertanyaan	Pilihan Jawaban					Jumlah
	A	B	C	D	E	
<i>Setting Fungsi</i>						
Apakah Tampilan Sistem Pakar berbasis <i>mobile application</i> menarik?	50	52	3	2	0	107
Apakah Menu-menu Sistem pakar berbasis <i>mobile application</i> ini mudah dipahami?	70	32	3	4	0	109
<i>System Metric</i>	A	B	C	D	E	
Apakah Sistem pakar berbasis <i>mobile application</i> ini mudah dioperasikan?	65	36	6	2	0	109
Apakah Sistem pakar berbasis <i>mobile application</i> ini Responsive?	90	24	3	0	0	117
Apakah Performa Sistem pakar berbasis <i>mobile application</i> ini baik?	95	20	3	0	0	118
<i>User Satisfaction</i>	A	B	C	D	E	
Apakah Sistem pakar berbasis <i>mobile application</i> dapat mengidentifikasi kerusakan perangkat PABX <i>Panasonic NS1000</i> ?	10	8	9	0	0	117
<i>Usability</i>	A	B	C	D	E	
Apakah Fitur-fitur Sistem pakar berbasis <i>mobile application</i> ini sudah cukup baik?	95	16	3	2	0	116
Apakah keluaran dari Sistem pakar berbasis <i>mobile application</i> sudah sesuai hasil identifikasi dengan pakar kerusakan perangkat PABX <i>Panasonic NS1000</i> .yang melakukannya?	90	24	3	0	0	117

- Analisa pertanyaan pertama Dari tabel 5 dapat dilihat bahwa jumlah nilai dari 25 responden untuk pertanyaan pertama adalah 107. Nilai rata-ratanya adalah $107/25 = 4,28$. Prosentase nilainya adalah $4,28/5 \times 100\% = 85,6\%$.
- Analisa pertanyaan kedua Dari tabel 5 dapat dilihat bahwa jumlah nilai dari 25 responden untuk pertanyaan kedua adalah 109. Nilai rata-ratanya adalah $109/25 = 4,36$. Prosentase nilainya adalah $4,36/5 \times 100\% = 87,2\%$.
- Analisa pertanyaan ketiga Dari tabel 5 dapat dilihat bahwa jumlah nilai dari 25 responden untuk pertanyaan ketiga adalah 109. Nilai rata-ratanya adalah $109/25 = 4,36$ Prosentase nilainya adalah $4,36/5 \times 100\% = 87,2\%$.

4. Analisa pertanyaan keempat
Dari tabel 5 dapat dilihat bahwa jumlah nilai dari 25 responden untuk pertanyaan keempat adalah 117. Nilai rata-ratanya adalah $117/25 = 4,68$. *Prosentase* nilainya adalah $4,68/5 \times 100\% = 93,6\%$.
 5. Analisa pertanyaan kelima
Dari tabel 5 dapat dilihat bahwa jumlah nilai dari 25 responden untuk pertanyaan kelima adalah 118. Nilai rata-ratanya adalah $118/25 = 4,72$. *Prosentase* nilainya adalah $4,72/5 \times 100\% = 94,4\%$.
 6. Analisa pertanyaan keenam
Dari tabel 5 dapat dilihat bahwa jumlah nilai dari 25 responden untuk pertanyaan keenam adalah 117. Nilai rata-ratanya adalah $117/25 = 4,68$. *Prosentase* nilainya adalah $4,68/5 \times 100\% = 93,6\%$.
 7. Analisa pertanyaan ketujuh
Dari tabel 5 dapat dilihat bahwa jumlah nilai dari 25 responden untuk pertanyaan ketujuh adalah 116. Nilai rata-ratanya adalah $116/25 = 4,64$. *Prosentase* nilainya adalah $4,64/5 \times 100\% = 92,8\%$.
 8. Analisa pertanyaan kedelapan
Dari tabel 5 dapat dilihat bahwa jumlah nilai dari 25 responden untuk pertanyaan kedelapan adalah 117. Nilai rata-ratanya adalah $117/25 = 4,68$. *Prosentase* nilainya adalah $4,68/5 \times 100\% = 93,6\%$.
- 1) Berbasis *Mobile Application*, sehingga program sistem pakar ini dapat *diinstall* pada perangkat *handphone* dengan berbagai *platform*
 - 2) Menjadi tools utama dalam analisa masalah, dengan adanya sistem pakar ini, maka teknisi dapat melakukan pekerjaan *troubleshooting* beserta *problem solving*.
 - 3) Komunikatif dan informatif, dengan adanya menu kamus maka pemahaman yang awam terhadap istilah PABX dapat dihindari.
 - 4) *Basic knowledge base* resource bagi seorang pakar, mereka dapat dengan mudah menyimpan dokumentasi pengetahuannya sebagai pengayaan pustaka ilmu.

3.4 Kekurangan Program

Kekurangan yang dimiliki program sistem pakar ini, antara lain:

Dari data di atas dapat disimpulkan bahwa *prosentase* dari *setting* fungsi sebesar 86,4% setuju sistem pakar untuk mengidentifikasi kerusakan perangkat PABX *Panasonic* NS1000 tersebut mempunyai tampilan sistem pakar berbasis *mobile application* menarik dan menu-menu sistem pakar berbasis *mobile application* ini mudah dipahami, dan *prosentase* dari *system metric* sebesar 91,73% setuju sistem pakar berbasis *mobile application* ini mudah dioperasikan, sistem pakar berbasis *mobile application* ini *responsive* dan performa sistem pakar berbasis *mobile application* ini baik serta *prosentase* dari *User Satisfaction* sebesar 93,6% setuju sistem pakar berbasis *mobile application* dapat mengidentifikasi kerusakan perangkat PABX *Panasonic* NS1000 dan *prosentase* dari *Usability* sebesar 93,2% setuju fitur-fitur sistem pakar berbasis *mobile application* ini sudah cukup baik dan keluaran dari sistem pakar berbasis *mobile application* sudah sesuai hasil identifikasi dengan pakar kerusakan perangkat PABX *Panasonic* NS1000 yang melakukannya.

Nilai rata-rata dari *prosentase setting* fungsi, *prosentase system metric*, *prosentase User Satisfaction* dan *prosentase Usability* yang setuju = $(86,4\% + 91,73\% + 93,6\% + 93,2\%) / 4 = 364,93/4 = 91,23\%$, sehingga hasil proses pengujian dengan UAT, para responden setuju (di atas 91,23%) bahwa secara keseluruhan sistem pakar membantu pakar serta dapat menyimpulkan kerusakan perangkat PABX dengan benar.

3.3 Kelebihan Program

Kelebihan yang dimiliki program sistem pakar ini, antara lain :

- 1) Penerapan Algoritma A* *Pathfinding* dengan fungsi *heuristic* terbukti tidak efektif dalam implementasi sistem pakar, karena sangat efektif untuk aplikasi pencarian jarak terdekat dan aplikasi permainan dengan menghitung nilai *heuristic*. Berdasarkan perbandingan waktu proses algoritma A* *Pathfinding* dengan fungsi *heuristic* dijalankan di sistem pakar untuk menemukan fakta-fakta yang sesuai untuk mendukung kesimpulan kerusakan perangkat PABX *Panasonic* NS1000 rata-rata membutuhkan waktu proses 300,30 detik, dijalankan di aplikasi pencarian jarak terdekat rata-rata memerlukan waktu proses 0,00324 detik dan di aplikasi permainan rata-rata memerlukan waktu proses 0,00344 detik.
- 2) Banyak sekali permasalahan yang dihadapi dalam sistem PABX ini, bukan hanya identifikasi kerusakan, namun prosedur instalasi dan pemograman perangkat juga masih menjadi hal yang sangat penting oleh karena itu pengembangan data sangat perlu dilakukan.
- 3) Basis data masih menggunakan *server-based* dan membutuhkan koneksi internet untuk mendapatkan dan melakukan perubahan data, walaupun demikian seiring dengan proses riset telah disiapkan *local-server-based* untuk menunjang kapasitas dan performa.
- 4) Perangkat NS1000 ini masih terus mengalami pengembangan oleh karena itu masih banyak hal-hal yang perlu diketahui berdasarkan pengalaman di lapangan.

4. Kesimpulan

Kesimpulan dapat diambil Algoritma A* (*A star*) *pathfinding* mempunyai karakteristik pencarian secara menyeluruh terhadap ruang statusnya. Algoritma A* (*A star*) *pathfinding* dengan fungsi *heuristic* terbukti tidak efektif dalam implementasi sistem pakar. Algoritme A Star hanya dapat digunakan dalam metode pencarian rute terpendek dalam penerapan sebuah peta dengan

menggunakan *handphone* sebagai sarana pendukung diharapkan. Sistem pakar dapat menjadi *database* pengetahuan dan dokumentasi mengenai ciri-ciri kerusakan pada perangkat PABX Panasonic NS1000 di perusahaan. Hasil proses pengujian dengan UAT, para responden setuju (di atas 91,23%) bahwa secara keseluruhan sistem pakar membantu pakar serta dapat menyimpulkan kerusakan perangkat PABX dengan benar.

Saran yang perlu diajukan untuk penelitian selanjutnya perlu adanya pengembangan algoritma dalam teknik-teknik *search*, sehingga ilmu pengetahuan tentang penerapan metode pelacakan untuk sistem pakar secara dinamis dapat berkembang dan lebih variatif, data gambar, diharapkan dapat ditampilkan berupa bagian-bagian yang mengalami kerusakan, sehingga akan lebih memperjelas proses identifikasi kerusakan jika dilihat pada computer dan data berupa video juga diharapkan dapat ditampilkan untuk lebih memperjelas proses identifikasi kerusakan pada perangkat PABX Panasonic NS1000.

Daftar Rujukan

- [1] PT. Mediatama Anugrah Citra, 2018, Detail Bisnis PT. Mediatama Anugrah Citra, Jakarta, Tersedia di: <https://yellowpages.co.id/bisnis/mediatama-anugrah-citra-pt>. [Accessed 10 November 2019].
- [2] PT. Mediatama Anugrah Citra, 2018, Profil PT. Mediatama Anugrah Citra, Jakarta, Tersedia di: <https://www.qerja.com/company/view/mediatama-anugrah-citra-pt>. [Accessed 10 November 2019].
- [3] PABX Panasonic Centra Telekomunikasi, 2018, PABX PANASONIC KX-NS1000, Bekasi: PABX Panasonic Centra Telekomunikasi, Tersedia di: <https://www.distributorpabxpanasonic.com/product/pabx-panasonic-kx-ns1000/>. [Accessed 15 November 2019].
- [4] Panasonic UK & Ireland. 2020, IP-PBX Panasonic KX-NS1000, United Kingdom: Panasonic UK & Ireland, Tersedia di: https://panasonic.net/cns/office/products/pbx/products/pbx/kx_ns1000/, [Accessed 15 November 2019].
- [5] Siswanto, 2010, Kecerdasan Tiruan. Edisi Kedua, Cetakan Pertama, Yogyakarta : Graha Ilmu.
- [6] Dedy H., Sepri D., 2018, Penerapan Algoritma A-Star Sebagai Pencari Rute Terpendek pada Robot Hexapod, Jurnal Nasional Teknik Elektro, Vol 7, No 2: July 2018, p-ISSN: 2302-2949, eISSN: 2407-7267, pp.122-129. DOI: <https://doi.org/10.25077/jnte.v7n2.545.2018>.
- [7] Wahyu W., Imam A., Desember 2017, Penerapan Algoritma A Star (A*) pada Game Petualangan Labirin Berbasis Android. KHAZANAH INFORMATIKA, Vol. 3 No. 2 |Online ISSN: 2477-698X, pp.57-63. DOI: <https://doi.org/10.23917/khif.v3i2.5221>.
- [8] Sandy P., Dyah A. M., Yusra F., 2018, Penerapan Algoritma A Star Untuk Penentuan Jarak Terdekat Wisata Kuliner di Kota Bandar Lampung, Jurnal TEKNOINFO, Vol. 12, No. 1, 2018, 28-32. ISSN: 1693-0010(print). DOI: <https://doi.org/10.33365/jti.v12i1.37>.
- [9] Veronica M., Fitria A., Noviana M., Desember 2013, Optimasi Pencarian Jalur dengan Metode A-Star Studi Kasus: Area Gading Serpong, Tangerang, Jurnal ULTIMATICS, Vol. V, No. 2, ISSN 2085-4552, pp. 42-47.
- [10] Rengga D.P., Muhammad A., dan Waru D., Desember 2012, "Pencarian Rute Terdekat Pada Labirin Menggunakan Metode A*," Jurnal EECIS, vol. 6, no. 2, Desember 2012, ISSN (online): 2460-8122. pp.1-4.
- [11] Masoud Nosrati, Ronak Karimi, dan Hojat Allah Hasanvand, April 2012 "Investigation of the * (Star) Search Algorithms Characteristics, Methods, and Approaches," World Applied Programming, vol. 2, no. 4, hal. 251-256.
- [12] Eka S., Wahyuni, Sholeh M., 2013, "Penerapan Jquery Mobile dan PHP Data Object Pada Aplikasi Pencarian Lokasi Tempat Ibadah Di Yogyakarta". Jurnal SCRIPT Vol. 1 No. 1 Desember 2013 ISSN:2338-6304, pp. 69-77. Tersedia di: <http://journal.akprind.ac.id/index.php/script/article/view/50>. [Accessed 10 November 2019].
- [13] Victor, Hamidi Ivan, Aldilah Diko. 2014, *Algoritma A* (A Star) Sebagai Salah Satu Contoh Metode Pemrograman Branch and Bound*. Tersedia di: <http://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/Makalah/MakalahStmik24.pdf>, [Accessed 10 November 2019].
- [14] Zhao, ZhenGuo, Riu, LunTao, 2015. "A Optimization of A* Algorithm to Make it Close to Human Pathfinding Behavior". 2nd International Conference on Electrical, Computer Engineering and Electronics (ICECEE 2015) Published by Atlantis Press pp. 708- 714, Tersedia di: http://www.atlantispress.com/php/download_paper.php?id=24625, [Accessed 10 November 2019]
- [15] Minami, Hidayat R., April 2013, *Rancang Bangun Aplikasi Sistem Pakar Untuk Kerusakan Komputer Dengan Metode Backward Chaining*, jurnal Teknoif, Vol. 1 No. 1, e-ISSN: 2598-9197, pp. 26-35, Tersedia di: <http://www.ejournal.itp.ac.id/index.php/tinformatika/article/viewFile/79/76>, . [Accessed 2 Februari 2020]
- [16] Resmi Darni, Dony Novaliendry, Ika Parma Dewi, Februari 2020, Aplikasi *Expert system* Pengembangan Karir Menggunakan *Inventory* Kepribadian *Entrepreneurship*, Jurnal Resti, Vol. 4 No. 1, ISSN Media Elektronik: 2580-0760, pp. 163 – 171.
- [17] Asep Kurniawan, Sumijan, Jufriadif Na'am, Desember 2019, Sistem Pakar Identifikasi Modalitas Belajar Siswa Menggunakan Metode *Forward Chaining*, Jurnal Resti, Vol. 3 No. 3, ISSN Media Elektronik: 2580-0760, pp. 518 – 523.
- [18] Endang C. P., 2017. Pengujian UAT (*User Acceptance Test*). Tersedia di: <https://endangcahyapermana.wordpress.com/2017/03/14/pengujian-uat-user-acceptance-test/>. [Accessed 10 Desember 2019].
- [19] C. S. Theng, 2017. "Leisure Technology for the Elderly: A Survey , User Acceptance Testing and Conceptual Design," *Int. J. Adv. Comput. Sci. Appl.*, vol. 8, no. 12, pp. 100–115, 2017.
- [20] Danang W. U., Defri K. dan Yani P. A.. 2018. Teknik Pengujian Perangkat Lunak Dalam Evaluasi Sistem Layanan Mandiri Pemantauan Haji Pada Kementerian Agama Provinsi Jawa Tengah. Jurnal SIMETRIS, Vol. 9 No. 2 November 2018, P-ISSN: 2252-4983, E-ISSN: 2549-3108, pp.731–746.