

SISTEM PAKAR PENDETEKSIAN PERMASALAHAN KOMPUTER PADA PT. PASIFIK SATELIT NUSANTARA CIKARANG

Joko Dwi Hartanto¹, Herlawati²

Abstract — Almost every people in the business world use computers to do their work. The users usually only know how to operate a computer, while it is difficult for them to fix it every time the computer got a problem, although minor problems on the computer. Thus the need for a system that can help the user so for some minor issues that could be resolved with the assistance of the system. The simple idea based on the background is about how to create a system that can provide solutions to users in fixing the computer when the computer is in trouble. Model the system based on the results of the analysis. This modeling form of ERD (Entity Relationship Diagram), UML (United Modeling Language). At this stage search method is needed to draw conclusions from the data that has been entered by the user of the symptoms that occur, the method used is a forward chaining is a method in which the search starts from taking the facts first and then used to draw conclusions. The results of this study can be concluded that the expert system that made this can allow a user to resolve minor problems that often occur without the help of the helpdesk. Minor problems experienced user can be resolved over having to wait addressed by the helpdesk when it is busy.

Intisari — Hampir setiap orang dalam dunia bisnis menggunakan komputer untuk melakukan pekerjaan mereka. Para pengguna biasanya hanya mengetahui bagaimana cara mengoperasikan komputer, sementara itu sulit bagi mereka untuk memperbaikinya setiap kali komputer mendapat masalah, meskipun permasalahan yang ringan pada komputer. Sehingga perlunya sebuah sistem yang dapat membantu user sehingga untuk beberapa masalah yang ringan maka bisa diselesaikan sendiri dengan bantuan sistem tersebut. Ide sederhana berdasarkan latar belakang tersebut adalah tentang bagaimana caranya membuat suatu sistem yang dapat memberikan solusi kepada pengguna komputer dalam memperbaiki komputer saat sedang bermasalah. Memodelkan sistem berdasarkan hasil analisa. Pemodelan ini berupa ERD (Entity Relationship Diagram), UML (United Modelling Language). Pada tahapan ini metode penelusuran diperlukan untuk menarik kesimpulan dari data-data yang telah dimasukkan oleh pengguna mengenai gejala-gejala yang terjadi, metode yang digunakan adalah forward chaining yaitu metode dimana penelusuran dimulai dari mengambil fakta-fakta terlebih dahulu baru kemudian digunakan untuk menarik kesimpulan. Hasil dari penelitian ini dapat disimpulkan bahwa sistem pakar yang dibuat ini dapat memudahkan user dalam menyelesaikan permasalahan ringan yang seringkali terjadi tanpa bantuan helpdesk. Permasalahan ringan yang dialami user dapat segera diselesaikan dibandingkan harus menunggu ditangani oleh helpdesk ketika sedang sibuk.

Kata kunci : Sistem Pakar, Forward Chaining, Helpdesk, Troubleshoot Komputer.

^{1,2} Program Studi Sistem Informasi STMIK Nusa Mandiri Jakarta, Jln. Damai No. 8 Warung Jati Barat (Margasatwa) Jakarta Selatan Telp. (021) 78839513 Fax. (021) 78839421; e-mail: dwi.tp1@gmail.com; herlawati@nusamandiri.ac.id

I. PENDAHULUAN

Teknologi informasi (TI) merupakan komponen yang sangat penting di era informasi saat ini sebagai kunci sukses suatu perusahaan dalam menjalankan bisnisnya. Para pengguna komputer biasanya hanya mengetahui bagaimana cara mengoperasikan komputer, sementara itu sulit bagi mereka untuk memperbaikinya setiap kali komputer mendapat masalah, meskipun permasalahan yang ringan pada komputer. Keberadaan Help Desk dalam sebuah perusahaan adalah hal yang penting untuk membantu pengguna dalam memecahkan masalah pada komputer. Mereka sangat diperlukan dalam membantu dan memastikan proses bisnis dapat berjalan dengan baik dalam hal sisi IT. Pengguna akan sangat membutuhkan bantuan staf Help Desk setiap kali mereka mengalami permasalahan pada komputer mereka.

Ide sederhana berdasarkan uraian tersebut adalah tentang bagaimana caranya membuat suatu sistem yang dapat memberikan solusi kepada pengguna komputer dalam memperbaiki komputer saat sedang bermasalah.

Penulis ingin membuat sebuah aplikasi berbasis web menggunakan konsep sistem pakar yang dapat membantu pengguna dalam menyelesaikan permasalahan pada komputer secara cepat tanpa bantuan Help Desk.

Sistem ini akan menghasilkan kemampuan dari seorang pakar untuk mendiagnosa suatu kemungkinan penyebab masalah dan bagaimana untuk mengatasi permasalahan tersebut. Untuk mengetahui penyebab kemungkinan suatu permasalahan tersebut, dibutuhkan sekumpulan informasi yang lengkap dimana informasi tersebut akan dimasukkan ke sistem pakar dan diolah untuk mendapatkan rekomendasi solusi bagaimana memecahkan permasalahan pada komputer.

Berdasarkan permasalahan yang diuraikan, maka maksud dari penelitian ini adalah :

1. Membuat sebuah sistem yang dapat merangkumkan pengetahuan seorang pakar dan berguna untuk mengetahui suatu permasalahan pada komputer dan cara mengatasinya. Sistem yang dibuat hanya akan digunakan dalam lingkungan internal perusahaan PT. PSN yang menggunakan jaringan intranet.
2. Membuat suatu aplikasi berbasis web menggunakan konsep sistem pakar yang dapat membantu pengguna dalam melakukan perbaikan pada komputer secara mandiri.

II. KAJIAN LITERATUR

Beberapa literatur yang dipergunakan penulis dalam penelitian ini yaitu:

- a. Kecerdasan Buatan
“Kecerdasan buatan adalah suatu ilmu yang mempelajari cara membuat komputer melakukan sesuatu seperti yang dilakukan oleh manusia” [7].
- b. Sistem Pakar
“Sistem pakar adalah aplikasi berbasis komputer yang digunakan untuk menyelesaikan masalah sebagaimana yang dipikirkan oleh pakar” [7].
- c. Metode Inferensi
Inferensi merupakan proses untuk menghasilkan informasi dari fakta yang diketahui atau diasumsikan. Inferensi adalah konklusi logis (*logical conclusion*) atau implikasi berdasarkan informasi yang tersedia [7].
Ada dua metode inferensi yang penting dalam sistem pakar yaitu : runut maju (*forward chaining*) dan runut balik (*backward chaining*).
 - 1) Runut Maju (*Forward Chaining*)
Menurut Wilson dalam [7] menyimpulkan bahwa :
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 Balik (*Backward Chaining*)
Menurut Giaratano dan Riley dalam [7] “Runut balik merupakan metode penalaran kebalikan dari runut maju. Dalam runut balik penalaran dimulai dengan tujuan kemudian merunut balik ke jalur yang akan mengarahkan ke tujuan tersebut”.
- d. Helpdesk
“*Help desk* adalah sebuah organisasi yang menyediakan satu titik kontak untuk pengguna yang membutuhkan dukungan teknis, apakah mereka pekerja internal maupun klien eksternal” [2].
- e. Program
Program adalah algoritma yang ditulis dalam bahasa komputer sedangkan bahasa komputer yang digunakan untuk menulis program dinamakan bahasa pemrograman, orang yang membuat program disebut pemrogram, kegiatan merancang dan menulis program disebut pemrograman dan aktivitas menulis kode program disebut coding [10].
- f. HTML
HTML (*Hyper Text Markup Language*) adalah sekumpulan simbol-simbol atau *tag-tag* yang dituliskan dalam sebuah file yang digunakan untuk menampilkan halaman pada web *browser* [1].
- g. PHP
PHP (*PHP Hypertext Preprocessor*) [1] yaitu bahasa pemrograman web *server-side* yang bersifat *open source*. PHP merupakan *script* yang terintegrasi dengan HTML dan berada pada server (*server side HTML embedded scripting*).

- h. Oracle
“Saat ini versi database Oracle, Oracle 11g, adalah sistem manajemen database yang memungkinkan pengguna untuk membuat, memanipulasi, dan mengambil data” [3].
- i. CSS
Cascading Style Sheet (CSS) adalah suatu kumpulan kode-kode untuk memformat, yang mengendalikan tampilan isi dalam suatu halaman web [8].
- j. Javascript
“Javascript adalah suatu bahasa pemrograman yang dikembangkan untuk dapat berjalan pada web *browser*” [13].
- k. Adobe Dreamweaver CS5
“Adobe dreamweaver CS5 adalah sebuah HTML editor profesional untuk mendesain sebuah web secara visual dan mengelola situs atau halaman web” [8].
- l. Adobe Photoshop CS5
“Adobe photoshop CS5 adalah perangkat lunak editor citra buatan Adobe Systems yang dikhususkan untuk pengolahan foto atau gambar dan pembuatan efek” [4].

III. METODE PENELITIAN

Metode pengumpulan data yang digunakan yaitu: metode observasi, metode wawancara dan metode studi pustaka. Beberapa tahap yang dilakukan dalam pengembangan *software* aplikasi *helpdesk* yang dibuat antara lain :

- a. Analisa Kebutuhan *Software*
Dalam tahapan ini, penulis melakukan analisa kebutuhan sistem, desain sistem untuk menentukan solusi perangkat lunak yang akan digunakan sebagai proses otorisasi sistem, penyebab kemungkinan yang terjadi sebagai diagnosa terhadap kerusakan yang terjadi pada komputer. Sistem yang dibangun ini dianalisis agar penerapan teori ke dalam praktik program dapat sejalan. Sehingga jika dicek baik secara manual dengan programnya menghasilkan diagnosa yang tidak jauh berbeda.
- b. Desain
Memodelkan sistem berdasarkan hasil analisa. Pemodelan ini berupa ERD (*Entity Relationship Diagram*), UML (*United Modelling Language*). Pada tahapan ini metode penelusuran diperlukan untuk menarik kesimpulan dari data-data yang telah dimasukkan oleh pengguna mengenai gejala-gejala yang terjadi, metode yang digunakan adalah *forward chaining* yaitu metode dimana penelusuran dimulai dari mengambil fakta-fakta terlebih dahulu baru kemudian digunakan untuk menarik kesimpulan. Dalam hal ini gejala digunakan sebagai fakta, setelah semua data gejala terpenuhi dapat digunakan untuk menarik kesimpulan untuk menemukan permasalahan dan solusinya.
- c. *Code Generation*
Desain yang telah dibuat kemudian diubah bentuknya menjadi bentuk yang dapat dimengerti oleh mesin, yaitu ke dalam bahasa pemrograman terstruktur melalui proses *coding*. Tahap ini merupakan implementasi dari tahap

design yang secara teknis nantinya dikerjakan oleh programmer.

d. *Testing*

Di tahap ini dilakukan pengujian menggunakan *white box testing* untuk mengetahui apakah aplikasi *helpdesk* yang dibuat masih terdapat kesalahan atau tidak, dan hasilnya harus benar-benar sesuai dengan kebutuhan yang sudah didefinisikan sebelumnya.

e. *Support*

Pada tahap *support* ini bertujuan untuk menjaga agar sistem yang telah dibuat tetap berjalan dengan produktif, dilakukan pemeliharaan dan pengembangan kemampuan sistem.

G22 : Tidak bisa buka youtube

G23 : Tidak bisa menggunakan fitur chat di browser

G24 : Tidak bisa putar video *streaming*

Tabel 1. Jenis Permasalahan dan Gejala

G/P	P1	P2	P3	P4	P5	P6	P7	P8
G1	x							
G2	x							
G3		x						
G4		x						
G5		x						
G6			x					
G7				x				
G8				x				
G9				x				
G10				x				
G11				x				
G12					x			
G13					x			
G14					x			
G15					x			
G16						x		
G17						x		
G18						x		
G19							x	
G20							x	
G21								x
G22								x
G23								x
G24								x

Sumber: Hasil Penelitian (2014)

Keterangan dari Jenis Kerusakan :

P1 : *Paper Jam*

P2 : Hasil print kurang bagus yang diakibatkan oleh toner tinta

P3 : *Driver* belum terinstall

P4 : Komputer Hang

P5 : *System Error*

P6 : Kuota Email Penuh

P7 : *Browser Corrupt*

P8 : Aplikasi *Addins* Perlu Diinstall

IV. HASIL DAN PEMBAHASAN

A. Rule-rule pada Pakar

Kaidah produksi biasanya dituliskan dalam bentuk jika maka (IF-THEN). Kaidah dapat dikatakan sebagai hubungan implikasi dua bagian yaitu premis (jika) dan bagian konklusi (maka). Apabila bagian premis dipenuhi maka bagian konklusi juga akan bernilai benar. Sebuah kaidah terdiri dari klausa-klausa sebuah klausa mirip sebuah kalimat subjek, kata kerja dan objek yang menyatakan suatu fakta. Ada sebuah klausa premis dan klausa konklusi pada sebuah kaidah. Suatu kaidah juga dapat terdiri dari beberapa premis dan lebih dari satu konklusi. Aturan premis dan konklusi dapat berhubungan dengan "OR" atau "AND".

Berdasarkan setiap permasalahan tersebut dijelaskan gejala-gejala yang biasanya terjadi pada komputer kemudian dirangkumkan oleh para pakar untuk ditemukan solusi penyelesaiannya ke dalam sebuah tabel pakar sebagai berikut.

Keterangan dari Gejala Kerusakan :

G1 : Tidak bisa print

G2 : *Paper jam*

G3 : Hasil print tidak sempurna

G4 : Hasil print pudar

G5 : Hasil print kurang bagus

G6 : Printer tidak menemukan driver

G7 : Komputer hang

G8 : Aplikasi hang

G9 : Tidak dapat membuka aplikasi

G10 : Mouse tidak berfungsi

G11 : Keyboard tidak berfungsi

G12 : Sistem operasi crash atau hang

G13 : PC hang terkena virus

G14 : Aplikasi berjalan lambat

G15 : Tidak bisa masuk ke *system*

G16 : *Inbox* penuh

G17 : Tidak bisa terima email

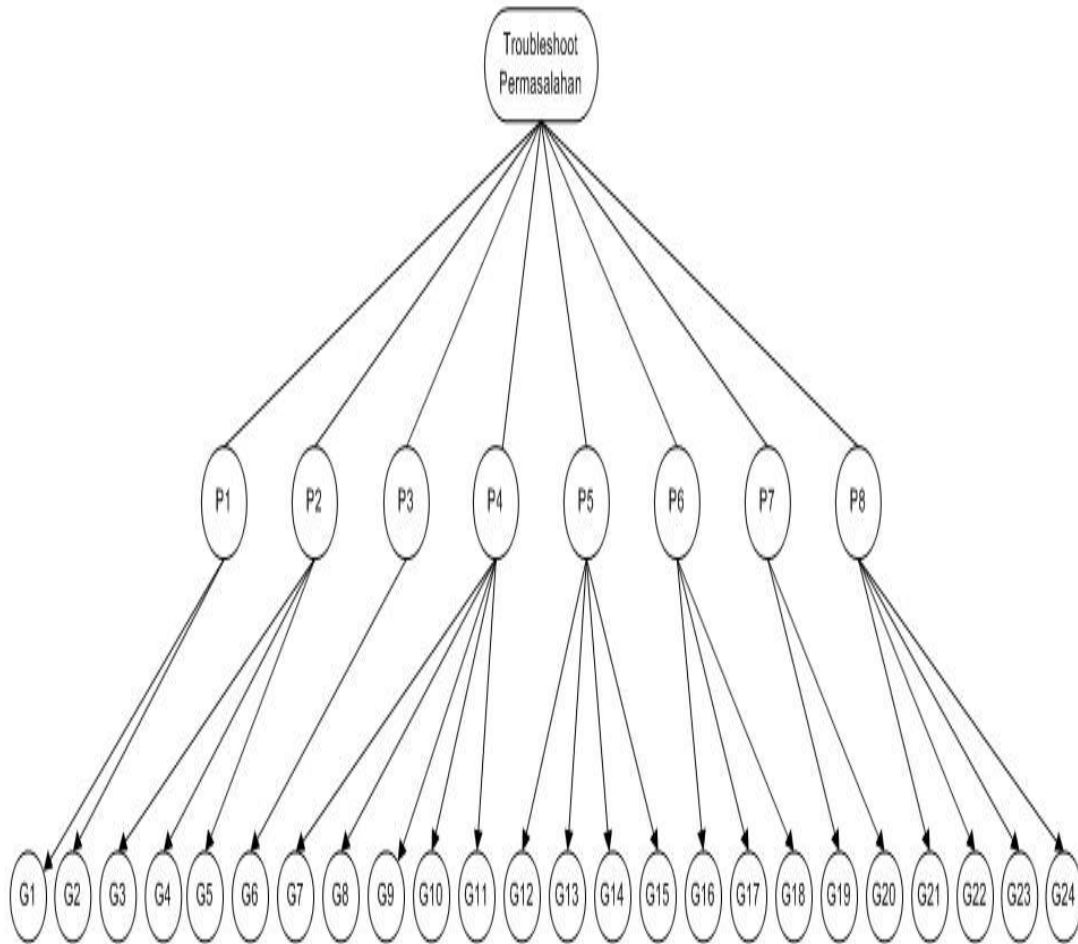
G18 : Tidak bisa kirim email

G19 : Aplikasi Mozilla tidak bisa dibuka

G20 : Aplikasi Mozilla Firefox lambat

G21 : Tidak bisa buka file di *browser*

Dari tabel 1 akan digunakan untuk menentukan rule, sehingga terbentuk pohon pakar sebagai berikut.



Sumber: Hasil Penelitian (2014)

Gambar. 1 Pohon Keputusan (Decision Tree)

Dari pohon keputusan akan dapat menghasilkan rule – rule pada pakar :

1. Keterangan GEJALA :

G1 : Tidak bisa print

G2 : Paper Jam

Keterangan RULE :

P1 : Paper jam, kertas tersangkut di dalam printer, sehingga hasil print tidak keluar

Keterangan SOLUSI :

Buka printer, kemudian keluarkan kertas tersebut dan tutup kembali printer

Dan seterusnya sampai gejala dan rule 8.

B. Tahapan Analisis

Sistem ini menyimpan basis pengetahuan seorang pakar dan pengguna dapat mengaksesnya dengan menginputkan gejala-gejala yang timbul pada komputernya yang sedang bermasalah. Halaman Front-page:

A1. *User* terlebih dahulu harus melakukan login.

A2. *User* dapat melihat petunjuk bagaimana menggunakan sistem pakar

A3. *User* dapat menginputkan kategori dan gejala permasalahan komputer

A4. *User* dapat menginputkan gejala permasalahan baru yang belum terdapat dalam sistem

A5. *User* dapat melihat permasalahan yang terjadi dan penjelasan solusi yang dilakukan

A6. *User* dapat memberikan feed back berupa laporan status dan komentar apakah solusi yang diberikan dapat menyelesaikan permasalahan

Halaman Administrasi:

B1. Admin dapat mengelola data kategori

B2. Admin dapat mengelola data subkategori

B3. Admin dapat mengelola data tim *helpdesk*

B4. Admin dapat mengelola data permasalahan dan solusinya

B5. Admin dapat mengelola data gejala permasalahan

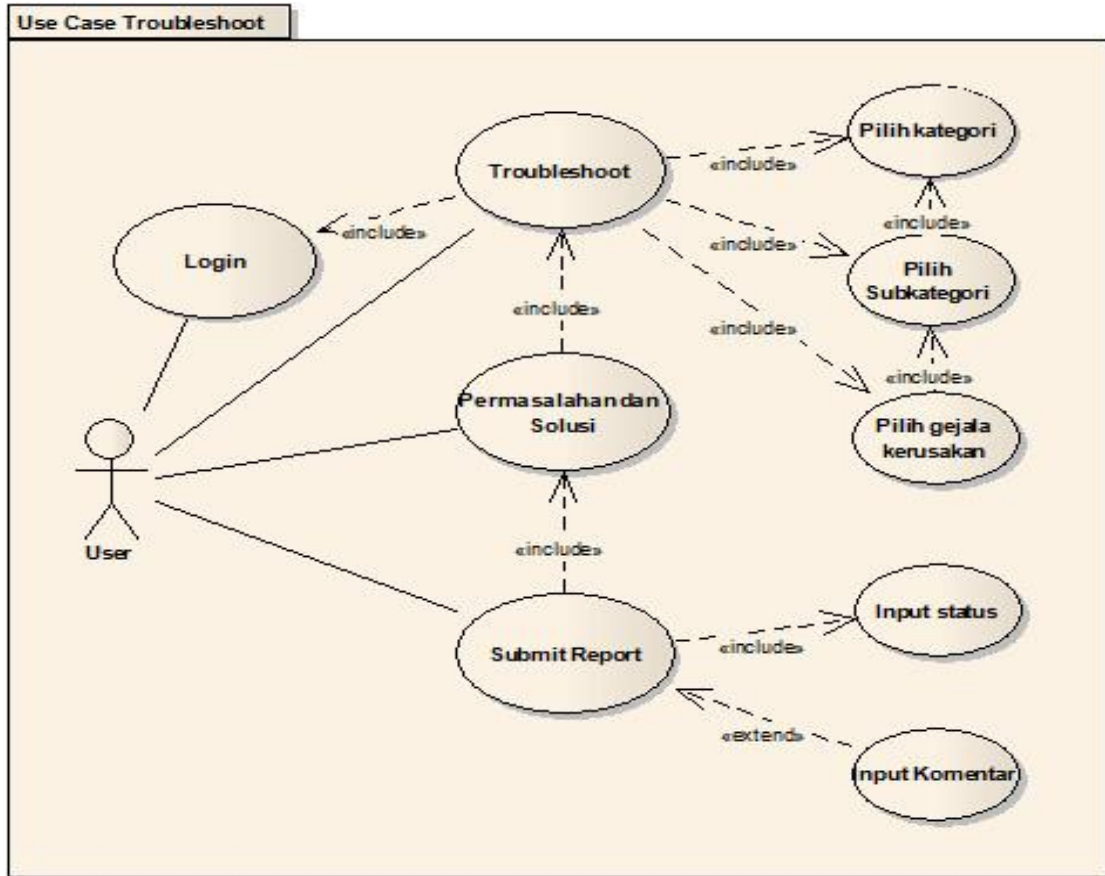
B6. Admin dapat melihat *log* dan laporan permasalahan berdasarkan *input* user.

C. Use Case

“ use case diagram bersifat statis dan memperlihatkan himpunan use case dan aktor-aktor (suatu jenis khusus dari kelas)” [15].

1. Use Case Diagram Troubleshoot

Use Case Diagram Troubleshoot pada penelitian ini adalah:



Sumber: Hasil Penelitian (2014)

Gambar 2. Usecase Diagram Troubleshoot

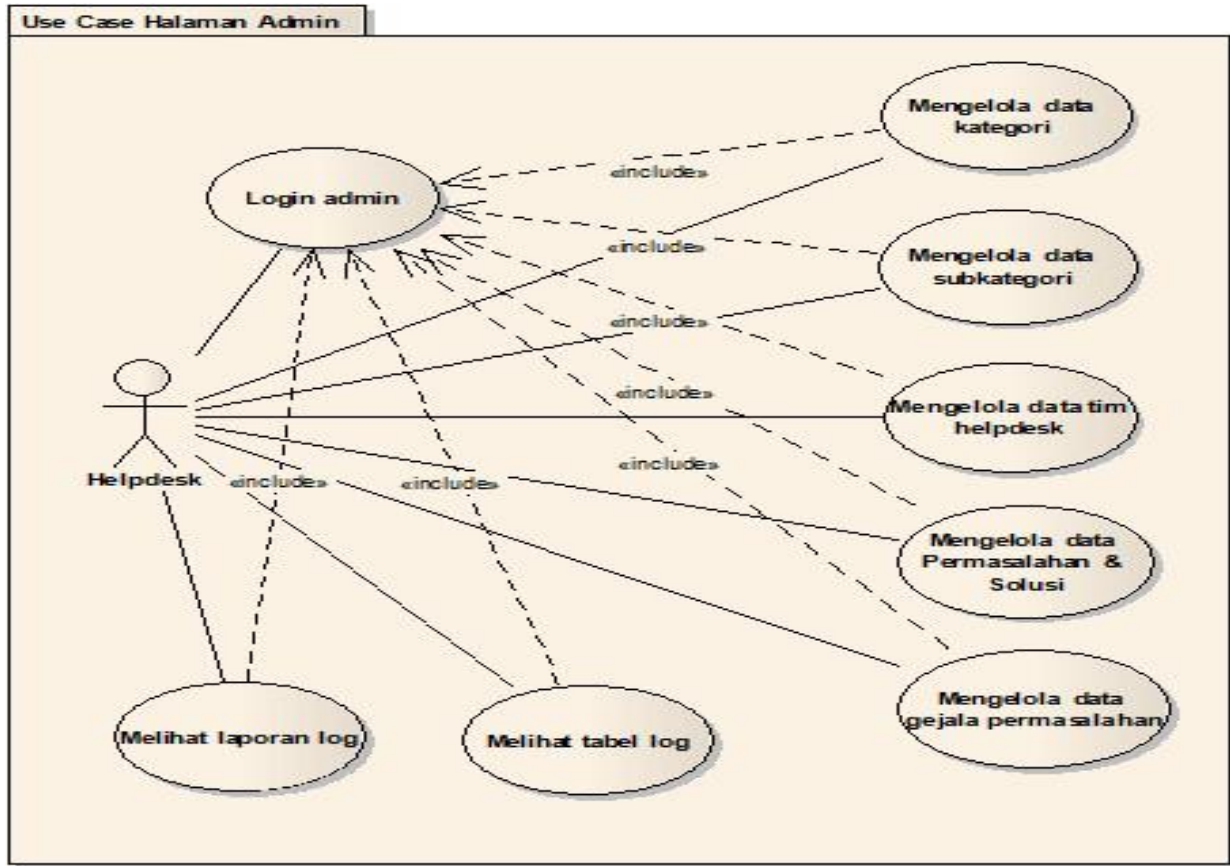
Tabel 2. Deskripsi Usecase Diagram Troubleshoot

Use Case Name	Troubleshoot
Requirements	A3, A5
Goal	User dapat troubleshoot permasalahan pada komputernya
Pre-conditions	User input kategori dan gejala
Post-conditions	Sistem memberikan output solusi
Failed end condition	User membatalkan troubleshoot. Kemungkinan user ingin menambahkan gejala yang terjadi
Primary Actors	User

Main Flow / Basic Path	<ol style="list-style-type: none"> 1. User melakukan troubleshoot dengan memilih kategori, subkategori dan gejala 2. System menampilkan permasalahan dan solusi yang disarankan 3. User dapat melihat solusi yang diberikan kemudian mencobanya dan klik Next 4. System menampilkan form <i>report</i> untuk <i>user</i> apakah solusi dapat dijalankan 5. User menginputkan status apakah status permasalahan sudah <i>solved</i> dan menginputkan komentar.
------------------------	--

Sumber: Hasil Penelitian (2014)

2. Use Case Diagram Halaman Admin



Sumber: Hasil Penelitian (2014)

Gambar 3. Usecase Diagram Halaman Admin

Tabel 3. Deskripsi Use Case Halaman Admin

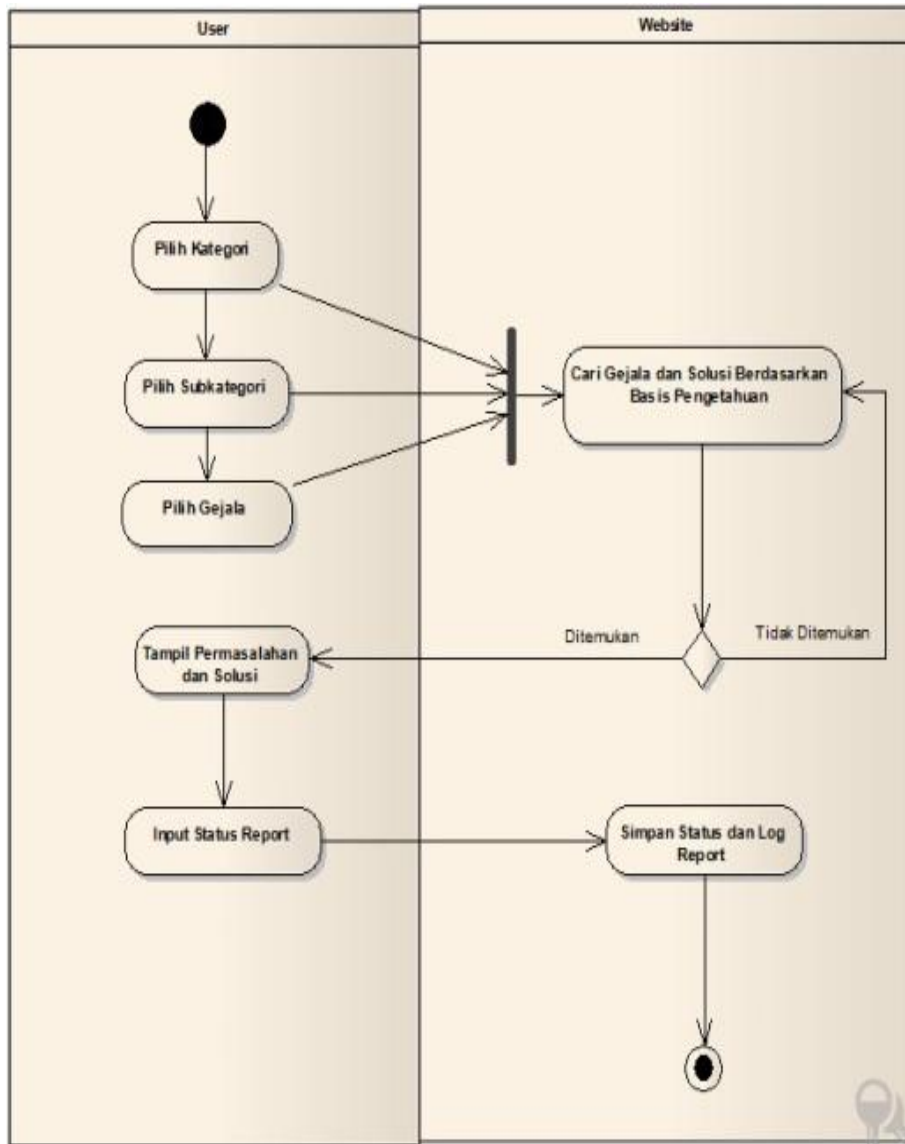
Use Case Name	Halman Admin
Requirements	B1, B2, B3, B4, B5, B6
Goal	Admin dapat mengelola data kategori, mengelola data subkategori, mengelola data tim helpdesk, mengelola data permasalahan dan solusi, dan mengelola data gejala.
Pre-conditions	Admin login hak akses terlebih dahulu untuk membuka halaman administrator.
Post-conditions	Admin mengelola semua data yang ada dihalaman administrator.
Failed end condition	Admin gagal melakukan login, kemungkinan pengisian data tidak lengkap atau pengisian data salah sehingga sistem meminta pengisian data

	kembali.
Primary Actors	Admin
Main Flow / Basic Path	<ol style="list-style-type: none"> Admin masuk ke dalam menu Admin. Admin menginput <i>user name</i> dan <i>password</i>. Admin mengklik tombol “Login” Sistem memvalidasi <i>user name</i> dan <i>password</i>.

Sumber: Hasil Penelitian (2014)

D. Activity Diagram

Diagram aktivitas bersifat dinamis dan tipe khusus dari diagram status yang memperlihatkan aliran dari suatu aktivitas ke aktivitas lainnya dalam suatu sistem”. Diagram ini terutama penting dalam pemodelan fungsi-fungsi suatu sistem dan memberi tekanan pada aliran kendali antar objek [15].

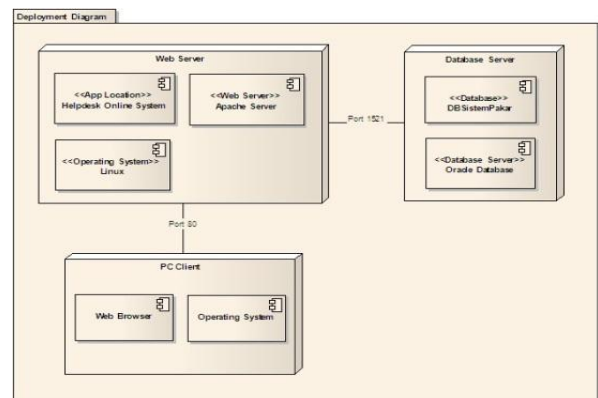


Sumber: Hasil Penelitian (2014)

Gambar 4. Activity Diagram Troubleshoot

E. Deployment Diagram

Diagram deployment bersifat statis dan memperlihatkan konfigurasi saat aplikasi dijalankan (run-time)”. Memuat simpul-simpul beserta komponen-komponen yang ada didalamnya. Diagram deployment berhubungan erat dengan diagram komponen dimana diagram ini memuat satu atau lebih komponen-komponen. Diagram ini sangat berguna saat aplikasi kita berlaku sebagai aplikasi yang dijalankan pada banyak mesin (distributed computing) [15].

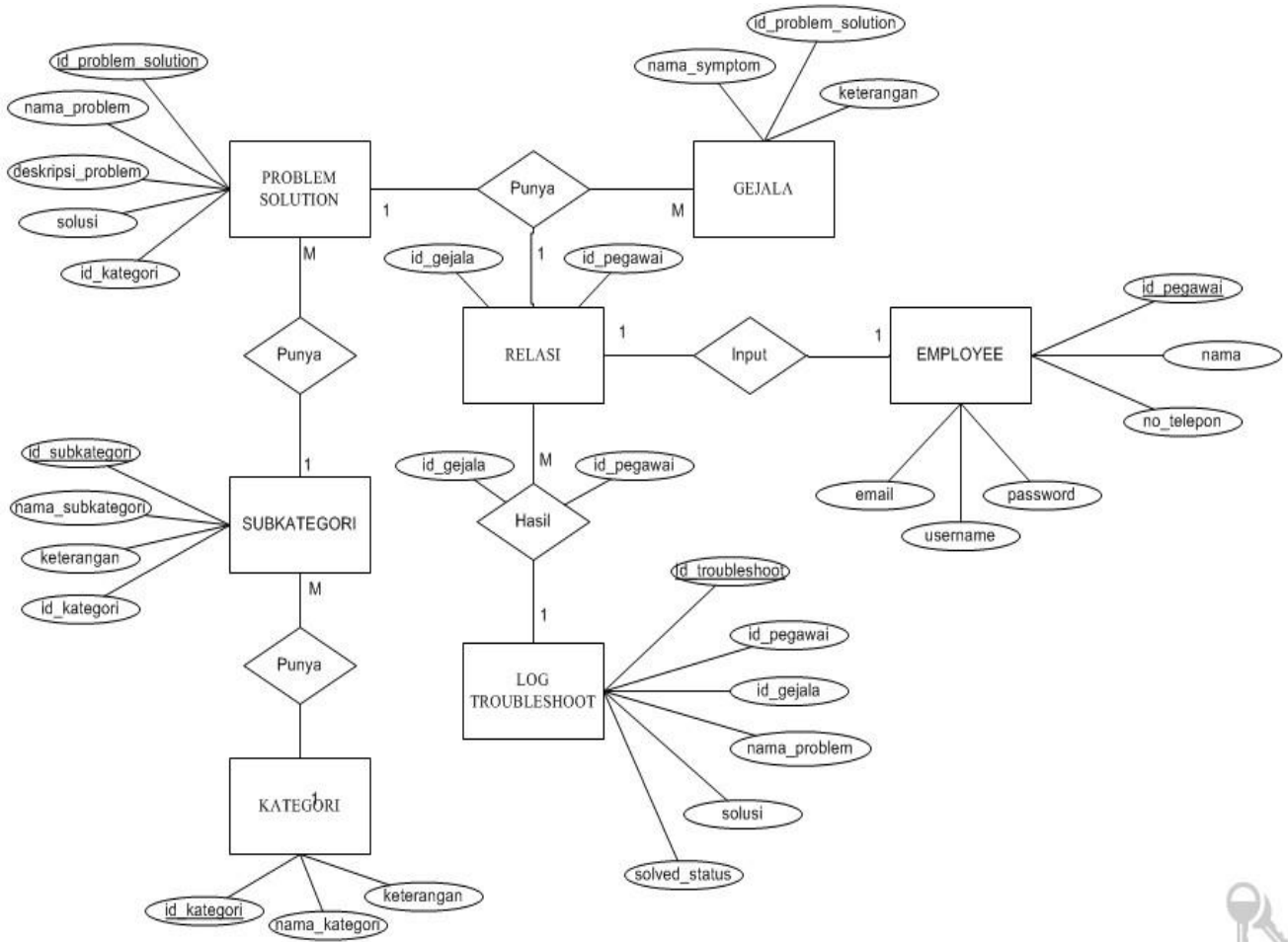


Sumber: Hasil Penelitian (2014)

Gambar 5. Deployment Diagram

F. Desain Database

Desain database dalam penelitian ini adalah:



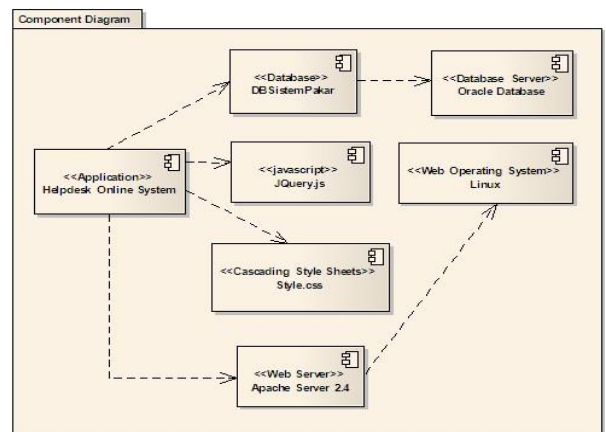
Sumber: Hasil Penelitian (2014)

Gambar 6. Entity Relation Diagram (ERD)

“Entity Relationship Diagram adalah alat pemodelan data utama dan akan membantu mengorganisasi data dalam suatu proyek ke dalam entitas-entitas dan menentukan hubungan antar entitas” [14]

G. Component Diagram

Diagram komponen (*component diagram*) bersifat statis dan memperlihatkan organisasi serta kebergantungan sistem/perangkat lunak pada komponen-komponen yang telah ada sebelumnya”. Diagram ini berhubungan dengan diagram kelas dimana komponen secara tipikal dipetakan kedalam satu atau lebih kelas-kelas, antarmuka-antarmuka serta kolaborasi-kolaborasi [15].



Sumber: Hasil Penelitian (2014)

Gambar 7. Component Diagram

H. Spesifikasi File

File yang ada di dalam *database* aplikasi *troubleshoot online* ini terdiri dari :

Nama Database : SP
 Nama File : Tabel *Category*
 Akronim : *categorytbl.frm*
 Type File : File Master
 Akses File : *Random*
 Panjang Record : 300 Byte
 Primary Key : id

Tabel 4. Spesifikasi Kategori

No	Nama Field	Akronim	Type	Size	Keterangan
1	<i>Id</i>	<i>Id</i>	<i>number</i>	10	<i>Primary Key</i>
2	<i>Category Name</i>	<i>category</i>	<i>Varchar2</i>	30	
3	<i>Remarks</i>	<i>remarks</i>	<i>Varchar2</i>	250	
4	<i>Valid Status</i>	<i>is_valid</i>	<i>Varchar2</i>	10	

Sumber: Hasil Penelitian (2014)

Nama Database : SP
 Nama File : Tabel *Subcategory*
 Akronim : *subcategorytbl.frm*
 Type File : File Master
 Akses File : *Random*
 Panjang Record : 380 Byte
 Primary Key : id

Tabel 5. Spesifikasi Subkategori

No	Nama Field	Akronim	Type	Size	Keterangan
1	<i>Id</i>	<i>Id</i>	<i>number</i>	10	<i>Primary Key</i>
2	<i>Sub Category</i>	<i>subcategory</i>	<i>Varchar2</i>	100	
3	<i>Remarks</i>	<i>remarks</i>	<i>Varchar2</i>	250	
4	<i>Category Id</i>	<i>category_id</i>	<i>number</i>	10	
4	<i>Valid Status</i>	<i>is_valid</i>	<i>Varchar2</i>	10	

Sumber: Hasil Penelitian (2014)

Nama Database : SP
 Nama File : Tabel *Problem Solution*
 Akronim : *problemsolutiontbl.frm*
 Type File : File Master

Akses File : *Random*
 Panjang Record : 870 Byte
 Primary Key : id

Tabel 6. Spesifikasi Problem Solution

No	Nama Field	Akronim	Type	Size	Keterangan
1	<i>Id</i>	<i>Id</i>	<i>number</i>	10	<i>Primary Key</i>
2	<i>Problem Name</i>	<i>problem_name</i>	<i>Varchar2</i>	100	
3	<i>Description</i>	<i>problem_desc</i>	<i>Varchar2</i>	250	
4	<i>Solution</i>	<i>Solution</i>	<i>Varchar2</i>	250	
5	<i>Subcategory Id</i>	<i>subcategory_id</i>	<i>number</i>	10	
6	<i>Remarks</i>	<i>remarks</i>	<i>Varchar2</i>	250	

Sumber: Hasil Penelitian (2014)

Nama Database : SP
 Nama File : Tabel *Symptoms*
 Akronim : *symptomstbl.frm*
 Type File : File Master
 Akses File : *Random*
 Panjang Record : 370 Byte
 Primary Key : id

Tabel 7. Spesifikasi Symptoms

No	Nama Field	Akronim	Type	Size	Keterangan
1	<i>Id</i>	<i>Id</i>	<i>number</i>	10	<i>Primary Key</i>
2	<i>Symptoms Name</i>	<i>symptoms</i>	<i>Varchar2</i>	100	
3	<i>Problem Id</i>	<i>ps_id</i>	<i>number</i>	10	
4	<i>Remarks</i>	<i>remarks</i>	<i>Varchar2</i>	250	

Sumber: Hasil Penelitian (2014)

I. User Interface

Suatu aplikasi harus mempunyai desain tampilan yang menarik, interaktif dan mudah dimengerti oleh *user*. Sehingga perlu didesain dengan baik. Pada aplikasi ini terdapat dua aktor yang dapat mengakses sistem pakar pendeteksian permasalahan computer yaitu *user* dan *admin*.



Sumber: Hasil Penelitian (2014)

Gambar 8. Tampilan halaman utama user



Sumber: Hasil Penelitian (2014)

Gambar 9. Tampilan halaman troubleshoot



Sumber: Hasil Penelitian (2014)

Gambar 10. Tampilan halaman hasil troubleshoot

V. KESIMPULAN

Berdasarkan apa yang telah penulis uraikan pada bab-bab sebelumnya maka dapat ditarik kesimpulan sebagai berikut :

1. Sistem pakar yang dibuat ini dapat memudahkan *user* dalam menyelesaikan permasalahan ringan yang seringkali terjadi tanpa bantuan *helpdesk*.
2. Permasalahan ringan yang dialami user dapat segera diselesaikan dibandingkan harus menunggu ditangani oleh helpdesk ketika sedang sibuk.

Aplikasi yang penulis buat tentunya masih belum sempurna, masih banyak kekurangan yang masih di rasakan oleh penulis. Agar aplikasi troubleshoot permasalahan komputer ini dapat digunakan dengan lebih optimal, maka penulis memberikan beberapa saran sebagai berikut :

Aspek Manajerial.

- a. Sebaiknya diadakannya pelatihan kepada karyawan untuk meningkatkan skill (keahlian atau ketrampilan) dalam melakukan troubleshoot ringan karena hal ini merupakan tugas dari divisi IT.
- b. Mengingat pentingnya data-data pada database maka diperlukan sistem backup untuk mencegah kerusakan dan hilangnya data.

Aspek Sistem.

- a. User interface dibuat dengan lebih menarik dan mudah digunakan bagi user dalam melakukan troubleshoot permasalahan.
- b. Menampilkan solusi dengan cara yang lebih interaktif dengan menyisipkan gambar langkah-langkah untuk troubleshoot komputer agar memudahkan dalam mengikuti langkah-langkah troubleshoot yang diberikan.

Aspek untuk penelitian selanjutnya.

- a. Sistem pakar yang telah dibuat dapat lebih dikembangkan dengan menambahkan kategori misalnya hardware dan subkategori lainnya sehingga sistem ini menampung banyak pengetahuan sehingga pengguna dapat dengan mudah menemukan permasalahan berdasarkan gejala yang dialami.
- b. Untuk pengembangan selanjutnya selanjutnya agar lebih banyak lagi gejala berikut permasalahan dan solusinya untuk menambahkan basis pengetahuan pada database agar user semakin mudah dalam menyelesaikan permasalahannya.

REFERENSI

- [1] Anhar. Panduan Menguasai PHP & MySQL Secara Otodidak. Jakarta: Mediakita. 2010.
- [2] Beisse, Fred. *A Guide to Computer User Support for Help Desk & Support Specialist*, Fifth Edition . Boston: Course Technology. 2013.
- [3] Casteel, Joan. Oracle 11g:SQL. Boston: Course Technology. 2010.

- [4] Effendhy, Asep. Otodidak Photoshop dari Basic Hingga Mahir. Bogor: Kubus Media. 2012.
- [5] Irawan. Panduan Berinternet Untuk Orang Awam. Maxikom. 2011.
- [6] Kusri. Sistem Pakar Teori dan Aplikasi. Yogyakarta: Andi Offset. 2006.
- [7] Kusri. Aplikasi Sistem Pakar. Yogyakarta: Andi Offset. 2008.
- [8] Madcoms. Kitab Suci *Web Programming*. Yogyakarta: Andi Offset. 2011.
- [9] Minami dan Rahmat Hidayar. Rancang Bangun Aplikasi Sistem Pakar Untuk Mendeteksi Kerusakan Komputer Dengan Metode Backward Chaining. Padang, Jurnal TEKNOIF Vol.1 No.1 Edisi April 2013. 2013.
- [10] Munir, Rinaldi. Algoritma dan Pemrograman. Bandung: Informatika. 2011.
- [11] Leung, Nelson K. Y. dan Sim Kim Lau. *Relieving the Overloaded Help Desk: A Knowledge Management Approach*. New South Wales, Communications of the IIMA 2006 Volume 6 Issue 2. 2006.
- [12] Purwanto, Iwan. Desain Sistem *Help Desk Troubleshooting Hardware dan Software Online*. ISSN : 1907-5022. Seminar Nasional Aplikasi Teknologi Informasi 2011 (SNATI 2011). Yogyakarta, 17-18 Juni 2011. 2011.
- [13] Sibero, Alexander. Kitab Suci *Web Programming*. Yogyakarta: Mediakom. 2011.
- [14] Simarmata, Janner dan Iman Paryudi. Basis Data Yogyakarta: Andi Offset. 2010.
- [15] Widodo, Prabowo Pudjo dan Herlawati. Menggunakan UML. Bandung: Informatika. 2011.



Joko Dwi Hartanto, S.Kom. Tahun 2014 lulus dari Program Strata Satu (S1) Program Studi Sistem Informasi STMIK Nusa Mandiri Jakarta.



Herlawati, S.Si, MM, M.Kom. Tahun 1998 lulus dari Program Strata Satu (S1) Program Studi Matematika Universitas Islan As'Syafi'iyah Jakarta. Tahun 2007 lulus dari Program Strata Dua (S2) Program Studi Magister Manajemen Universitas Budi Luhur Jakarta. Tahun 2010 lulus dari Program Strata Dua (S2) Program Studi Magister Ilmu Komputer STMIK Nusa Mandiri Jakarta. Tahun 2009 sudah tersertifikasi dosen dengan Jabatan Fungsional Akademik Lektor di STMIK Nusa Mandiri Jakarta. Tahun 2011 menerbitkan buku dengan judul "Menggunakan UML" penerbit Informatika Bandung sebagai penulis kedua bersama dengan Dr.Ir. Prabowo Pudjo Widodo. Tahun 2013 menerbitkan buku dengan judul "Penerapan Data Mining dengan Matlab" penerbit Informatika Bandung sebagai penulis ketiga bersama dengan Dr. Ir. Prabowo Pudjo Widodo dan Rahmadya Trias Handyanto, ST, M.Kom. Aktif mengikuti seminar dan menulis paper di beberapa jurnal diantaranya Jurnal Pilar STMIK Nusa Mandiri, Jurnal Paradigma AMIK BSI Jakarta dan Jurnal PIKSEL Universitas Islam 45 Bekasi. Aktif sebagai anggota Asosiasi Dosen Indonesia (ADI) dan anggota Asosiasi Pendidikan Tinggi Informatika dan Komputer (APTIKOM).