

## Penerapan Model *Rapid Application Development* pada Aplikasi *Helpdesk Trouble Ticket* PT. Satkomindo Mediyasa

Maulana Ardhiansyah

Teknik Informatika, Universitas Pamulang, Tangerang Selatan, Banten, Indonesia  
e-mail: maulana1402@gmail.com

### Abstrak

PT. Satkomindo Mediyasa is a limited liability company engaged in providing satellite-based telecommunications infrastructure networks (Very Small Aperture Terminal / VSAT). In an effort to service customer VSAT network connections, PT. Satkomindo Mediyasa often gets various reports of complaints from its customers. However, the complaints report management process that has been running so far is still not effective, causing the handling of customer complaints to run slowly and require a long time. In formulating the solution to the problem, it is proposed a helpdesk trouble ticket information system to help PT. Satkomindo Mediyasa in managing complaints reports is faster and more effective. Information systems were developed using the PHP and MySQL programming languages. The system development model used is Rapid Application Development (RAD) by being modeled through Unified Modeling Language (UML) diagrams. The results of this study produce a trouble ticket helpdesk information system that helps PT. Satkomindo Mediyasa in managing customer complaints reports is faster and more effective.

Keywords: Helpdesk, Trouble Ticket, Information System, Rapid Application Development (RAD), Unified Modeling Language (UML)

### 1. Pendahuluan

PT. Satkomindo Mediyasa adalah sebuah perseroan terbatas yang bergerak di bidang jasa penyediaan jaringan infrastruktur telekomunikasi berbasis satelit (*Very Small Aperture Terminal/VSAT*). PT. Satkomindo Mediyasa mempunyai beberapa pelanggan yang menggunakan jasa layanannya dengan jumlah jaringan VSAT lebih dari 6000 titik yang tersebar di seluruh Indonesia. (Sumber: <http://www.satkomindo.com>)

Dalam upaya pelayanan koneksi jaringan VSAT di pelanggan, PT. Satkomindo Mediyasa juga sering mendapat berbagai laporan keluhan dari pelanggannya. Selama ini, pihak pelanggan akan melaporkan keluhan yang dialami kepada bagian *helpdesk* melalui telepon atau *email*. Namun dalam proses pencatatan laporan keluhan dari pelanggan, tim *helpdesk* masih melakukannya secara manual di *form Excel*. *Helpdesk* akan membuat sebuah *file form excel*, kemudian akan dicatat dengan detail keluhan apa yang dilaporkan oleh pelanggan. Jika kemudian dalam proses penanganan laporan tim *helpdesk* membutuhkan pengecekan lebih lanjut, maka tim *helpdesk* akan

melakukan eskalasi dengan mengirimkan email kepada bagian terkait misalnya koordinator teknisi ataupun NOC. Selama proses penanganan, tim *helpdesk* harus mencatat *history* penanganannya ke dalam *file form excel* tersebut.

Tentu saja dengan proses yang manual menggunakan *input form excel* dan penggunaan *email* sangat tidak praktis. Ketika jumlah laporan keluhan yang masuk cukup banyak maka tim *helpdesk* harus membuat *file form excel* tersebut satu per satu, yang mengakibatkan ketika akan melakukan *update* penanganannya, tim *helpdesk* harus mencari file-file tersebut di komputer yang membutuhkan waktu lebih lama. Kemudian dalam hal eskalasi melalui *email* sering terhambat karena banyaknya *email* yang masuk ke bagian masing-masing, sehingga tindak lanjut kepada penanganan keluhan pelanggan juga akan terhambat. Lalu belum adanya rekap laporan setiap bulan atas laporan-laporan keluhan yang sudah ditangani. Karena rekap laporan ini akan dikirimkan setiap bulannya kepada bagian keuangan perusahaan sebagai dasar perhitungan restitusi atau pengembalian biaya layanan apabila pelanggan tidak mendapatkan layanan sesuai dengan kontrak

yang telah disepakati. Dengan tidak adanya rekap laporan ini akan berdampak pada sulitnya bagian keuangan dalam melakukan perhitungan biaya restitusi atau pengembalian biaya atas kendala layanan yang digunakan oleh pelanggan dalam 1 bulan. Dengan begitu, perlu juga adanya rekap laporan atas setiap keluhan pelanggan yang disertai dengan laporan perhitungan berapa lama waktu penanganan selesai dilakukan sampai layanan yang digunakan pelanggan normal kembali.

Maka dari itu perlu adanya integrasi antar bagian ke dalam suatu sistem yang sangat penting guna menghasilkan kualitas layanan pelanggan yang lebih baik dan efektif, salah satunya yaitu dengan menggunakan sistem informasi *helpdesk trouble tickets* yang sesuai dengan kebutuhan perusahaan. Dengan menerapkan sistem informasi *helpdesk trouble tickets* sebagai sarana pengelolaan laporan keluhan diharapkan mampu mengelola masalah-masalah yang dialami pelanggan untuk secepatnya ditindaklanjuti.

Karena keterbatasan waktu yang ada, maka penulis memilih untuk menggunakan model pengembangan sistem *Rapid Application Development (RAD)*. Metode *Rapid Application Development* adalah sebuah proses pengembangan perangkat lunak sekuensial linear yang menekankan siklus pengembangan dalam waktu yang singkat (Aswati, S., & Siagian, Y, 2016:318). Dalam pengembangan sistem informasi normal, memerlukan waktu minimal 180 hari, namun dengan menggunakan metode RAD, sistem dapat diselesaikan dalam waktu 30-90 hari.

## 2. Landasan Teori

### 2.1. Helpdesk

*HelpDesk* adalah pelaporan masalah yang diatur secara terurut dan terorganisir menggunakan sebuah *center point*. *Helpdesk* juga dikenal sebagai sumber dari pemecahan masalah yang menjadi bagian pelengkap dari sebuah fungsi pelayanan dan pertanggung jawab. (Chintalois, 2015:12)

*HelpDesk* menyediakan dokumen fungsi produk, servis atau teknologi dari perusahaan sehingga *HelpDesk* bagian dari perusahaan tersebut, sehingga *HelpDesk* dikenal juga sebagai sebuah departemen di perusahaan tersebut yang melayani atau menanggapi pertanyaan teknis pengguna atau klien. Pertanyaan dan jawaban dapat disampaikan melalui email, web, telp atau menggunakan *software HelpDesk* yang membuat orang lebih mudah untuk menemukan dengan

cepat jawaban yang bersifat umum. (Chintalois, 2015:12)

#### 2.1.1 Kelebihan Helpdesk:

- HelpDesk* dapat mengurangi biaya serta memberikan pelayanan terbaik pada penggunanya.
- HelpDesk* dalam kurun waktu singkat dapat memberikan solusi atas pertanyaan – pertanyaan.
- HelpDesk* dapat meningkatkan efisiensi perusahaan dalam menangani pertanyaan dan keluhan pelanggan serta dapat mengecek permasalahan yang ada dan mengatur pembagian staff.
- HelpDesk* dapat menangani pertanyaan, keluhan yang sejenis karena pertanyaan dan keluhan tersebut tercatat serta dapat memberi laporan kerja perkembangan kinerja para staff dan pimpinan.



Gambar. 1 Cara Kerja Helpdesk

Sumber : <http://mainhelpdesk.blogspot.com> tanggal akses 30 Agustus 2018

#### 2.1.2 Tipe Helpdesk

Berdasarkan tipe konsumen yang ditanganinya, helpdesk dapat dibagi menjadi dua jenis (Mudhar, 2015:14), antara lain:

- Internal Helpdesk*  
Adalah *helpdesk* yang memberikan pelayanan kepada *internal customer* yaitu para karyawan yang bekerja pada perusahaan tersebut. Tujuan dari *helpdesk* ini adalah agar karyawan dapat memaksimalkan infrastruktur yang ada dan mengoptimalkan kinerja mereka.
- External Helpdesk*  
Adalah *helpdesk* yang ditujukan kepada pihak di luar perusahaan untuk melakukan penjualan produk atau layanan tertentu. Sehingga *helpdesk* disini adalah untuk melayani *customer* pengguna produk/jasa perusahaan sebagai *additional after sales* terhadap penjualan perusahaan tersebut.

## 2.2 Trouble Ticket

*Trouble ticket* (disebut juga *trouble report*) adalah sebuah mekanisme yang digunakan dalam organisasi untuk melacak, melaporkan, dan menyelesaikan beberapa jenis masalah yang dialami. Sistem *trouble ticket* berasal dari sebuah sistem pelaporan masalah berbasis kertas, akan tetapi sekarang kebanyakan berbasis *website* dan terkait dengan lingkungan *Customer Relationship Manajemen* (CRM), seperti *call centre*, *website e-business* atau lingkungan teknologi tingkat tinggi seperti *Network Operations Centers* (NOCs). (Dio Angga Pratama, 2010:5).

## 2.3 Rapid Application Development (RAD)

Salah satu metode pengembangan suatu sistem informasi dengan waktu yang relatif singkat adalah dengan menggunakan *Rapid Application Development* (RAD). Waktu normal untuk pengembangan suatu sistem informasi membutuhkan waktu minimal 180 hari, dengan menggunakan metode RAD maka dalam waktu 30-90 hari suatu sistem dapat diselesaikan. (Noertjahyana, 2002:74)

Tujuan utama dari semua metode sistem *development* adalah harapan dari pemakai atau user akan suatu sistem dapat terpenuhi, akan tetapi dalam melakukan pengembangan suatu sistem para pemakai sistem atau user tidak dilibatkan secara langsung, hal ini menyebabkan sistem informasi yang dibuat tidak sesuai dari harapan pemakai sistem yang berdampak sistem tersebut kurang diterima oleh pemakai sistem untuk menggunakannya atau bahkan para pemakai sistem menolak untuk menggunakannya.

### 2.3.1 Tahapan – Tahapan pada RAD

Metode RAD mempunyai 3 tahapan utama, yaitu sebagai berikut:

#### a. Rencana Kebutuhan (*Requirement Planning*)

Pada tahap ini, mengidentifikasi tujuan dari aplikasi atau sistem dan melakukan identifikasi kebutuhan informasi untuk mencapai tujuan yang diharapkan *user*, maka diperlukan pertemuan antara *user* dan *analyst*. Pada tahap ini adalah adanya keterlibatan kedua belah pihak, dan hal tersebut adalah yang penting untuk dilakukan, maka bukan hanya sekedar persetujuan akan proposal yang telah dibuat. Keterlibatan *user* bukan hanya dari satu tingkatan pada suatu organisasi, melainkan beberapa tingkatan organisasi. Pertemuan semacam ini sering disebut *Joint Application Development*, sehingga informasi

yang dibutuhkan untuk masing-masing *user* dapat terpenuhi dengan baik.

#### b. Proses Design (*Design Workshop*)

Pada tahap ini adalah melakukan proses desain dan melakukan perbaikan-perbaikan apabila masih terdapat ketidaksesuaian desain antara *user* dan *analyst*. Untuk tahap ini maka keaktifan *user* yang terlibat sangat menentukan untuk mencapai tujuan, karena *user* bisa langsung memberikan komentar apabila terdapat ketidaksesuaian pada desain. Biasanya *user* dan *analyst* berkumpul menjadi satu dan duduk bersama.

#### c. Implementasi (*Implementation*)

Setelah desain dari sistem yang akan dibuat sudah disetujui baik itu oleh *user* dan *analyst*, maka pada tahap ini *programmer* mengembangkan desain menjadi suatu program. Setelah program selesai baik itu sebagian maupun secara keseluruhan, maka dilakukan proses pengujian terhadap program tersebut apakah terdapat kesalahan atau tidak sebelum diaplikasikan pada suatu organisasi. Pada saat ini maka *user* bisa memberikan tanggapan akan sistem yang sudah dibuat serta persetujuan mengenai sistem tersebut.

Adapun hal terpenting adalah bahwa keterlibatan *user* sangat diperlukan supaya sistem yang dikembangkan dapat memberikan kepuasan kepada *user*, dan di samping itu, sistem yang lama tidak perlu dijalankan secara paralel dengan sistem yang baru.



Gambar. 2 Tahapan - tahapan RAD

## 3. Analisa dan Perancangan

### 3.1 Perencanaan Kebutuhan

Pada fase ini, penulis melakukan proses identifikasi tujuan dari aplikasi atau sistem dan melakukan identifikasi kebutuhan informasi apa saja yang akan diperlukan dalam pengembangan sistem informasi *helpdesk trouble ticket*.

#### 3.1.1 Analisa Proses Bisnis yang Berjalan

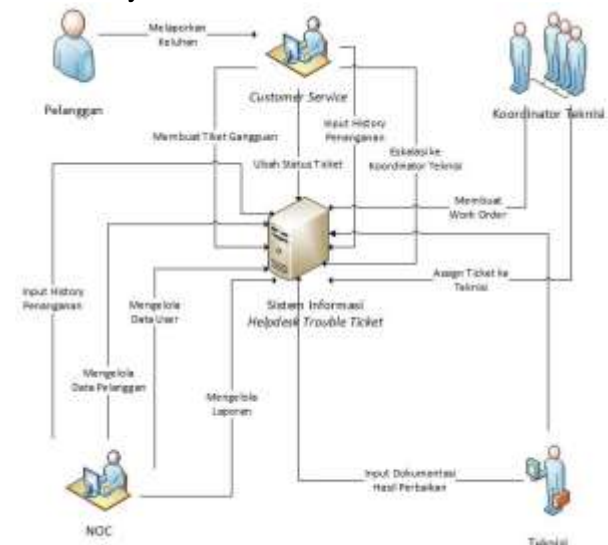
Prosedur pencatatan dan pengelolaan laporan keluhan dari pelanggan pada PT. Satkomindo Mediyasa adalah sebagai berikut:

- 1) Pelanggan melaporkan keluhan atas layanan yang digunakan kepada *helpdesk* melalui telepon ataupun *email*.
- 2) Laporan keluhan diterima oleh *helpdesk*, lalu dibuat sebuah *file form excel* baru untuk mencatat detail keluhan pelanggan.
- 3) *Helpdesk* melakukan analisa atas keluhan yang dialami oleh pelanggan. *Helpdesk* akan memandu pihak pelanggan untuk melakukan pengecekan kondisi perangkat dilokasi seperti modem, kondisi antena parabola, kondisi kelistrikan, kondisi pengkabelan.
- 4) Apabila setelah dipandu layanan yang digunakan pelanggan telah kembali normal, maka pada *file form excel* yang telah dibuat sebelumnya diberi status *Closed* dan simpan pada komputer.
- 5) Namun apabila setelah dipandu layanan yang digunakan pelanggan belum beroperasi dengan normal, maka *helpdesk* akan melakukan eskalasi dengan mengirimkan *email* kepada bagian koordinator teknisi agar bisa ditindaklanjuti dengan mengirimkan teknisi ke lokasi pelanggan langsung untuk perbaikan.
- 6) Koordinator teknisi menerima *email* eskalasi dari *helpdesk*, kemudian mengecek jadwal ketersediaan teknisi sesuai dengan domisili terdekat dengan lokasi pelanggan.
- 7) Koordinator teknisi mengirimkan *email* dan juga dengan telepon ke teknisi yang telah ditentukan dan menugaskan kepada teknisi untuk kunjungan ke lokasi pelanggan untuk perbaikan.
- 8) Teknisi datang ke lokasi pelanggan untuk perbaikan.
- 9) Teknisi selesai perbaikan dan mendokumentasikan hasil perbaikannya.
- 10) Teknisi melaporkan hasil perbaikan kepada NOC melalui telepon atau *whatsapp* untuk dilakukan pengukuran kualitas layanan.
- 11) NOC melakukan pengukuran kualitas link dari hasil perbaikan yang telah dilakukan teknisi.
- 12) Apabila kualitas link belum memenuhi standar, NOC menugaskan kembali teknisi untuk memaksimalkan perbaikan. Apabila kualitas link sudah memenuhi standar maka NOC akan mendokumentasikan hasil pengukuran kualitas link.

- 13) NOC menginformasikan hasil perbaikan teknisi dan hasil pengukuran kualitas link ke *helpdesk*.
- 14) *Helpdesk* mencatat *history* hasil penanganannya sampai beroperasi kembali layanan pelanggan.

### 3.1.2 Analisa Sistem yang Diusulkan

Dari hasil analisa dan identifikasi masalah yang sudah dilakukan sebelumnya, maka penulis merancang sistem yang diusulkan guna memperbaiki kekurangan atau masalah yang dihadapi pada proses bisnis yang berjalan sebelumnya.



Gambar. 3 Analisa Sistem yang Diusulkan

Dalam proses bisnis yang akan diusulkan, pihak pelanggan akan melaporkan keluhan yang dialami ke bagian *helpdesk* melalui telepon atau *email*. Lalu kemudian *helpdesk* akan login ke sistem dan membuat tiket gangguan baru. Setelah tiket gangguan dibuat, maka sistem akan menyimpannya di *database* dan menampilkan nomor tiket gangguan. Nomor tiket gangguan ini yang akan memudahkan *helpdesk* dalam pencarian tiket gangguan apabila akan *update history* penanganannya ke dalam sistem. Tim *helpdesk* akan melakukan analisa keluhan dan pandu pihak pelanggan untuk pengecekan perangkat. Apabila setelah dipandu layanan normal kembali, maka tim *helpdesk* akan mengubah status tiket gangguan di sistem menjadi *Closed*. Apabila membutuhkan penanganan lebih lanjut, maka *helpdesk* akan melakukan eskalasi tiket dengan tujuan kepada koordinator teknisi agar bisa ditangani lebih lanjut oleh teknisi. Setelah itu, koordinator teknisi akan bisa melihat tiket gangguan yang sudah di eskalasi oleh *helpdesk*, sehingga koordinator teknisi mendapatkan informasi indikasi awal masalah dari

hasil analisa dan pandu pengecekan yang sudah dilakukan oleh tim *helpdesk* sebelumnya. Koordinator teknisi akan membuat *work order* di sistem dan memasukkan tujuan kepada teknisi yang telah ditentukan sesuai dengan lokasi terdekat dengan lokasi pelanggan. Sistem akan menyimpannya di *database* dan menampilkan nomor *work order*. Teknisi menerima dan melihat *detail work order* di sistem, kemudian datang ke lokasi sesuai dengan yang ada di *detail work order* untuk perbaikan. Setelah selesai perbaikan, teknisi akan dokumentasi hasil perbaikan, kemudian akan *upload* dokumentasi hasil perbaikannya ke dalam sistem sesuai dengan *work order* lokasi yang sedang dikerjakan. Teknisi kemudian akan melaporkannya kepada NOC via telepon atau *whatsapp* bahwa teknisi sudah selesai perbaikan dengan menyebutkan nomor *work order* yang dikerjakan. NOC akan mencari *work order* sesuai dengan yang diinformasikan oleh teknisi, kemudian akan dilihat dokumentasi hasil perbaikan yang sudah di-upload oleh teknisi. NOC

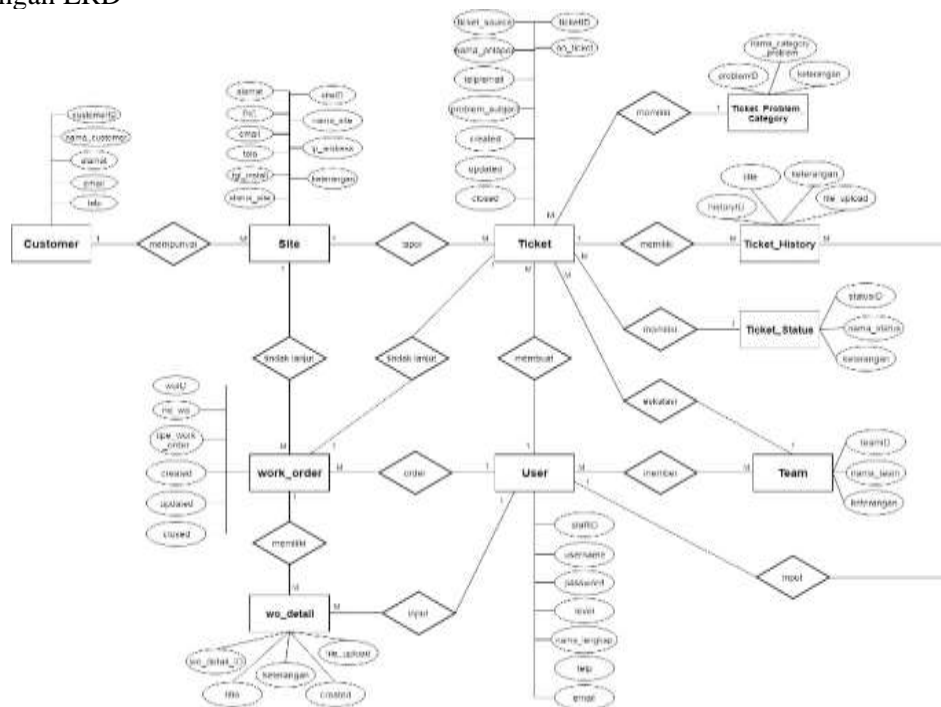
akan melakukan pengukuran hasil perbaikan apakah sudah memenuhi standar atau belum. Apabila hasil perbaikan belum memenuhi standar, NOC akan menginstruksikan teknisi untuk memaksimalkan kembali hasil perbaikan. Jika sudah memenuhi standar, NOC akan *update history* perbaikannya di tiket gangguan. Setelah selesai perbaikan dan layanan yang digunakan pelanggan telah normal kembali, tim *helpdesk* akan mengubah status tiket menjadi *Closed*.

### 3.2 Proses Design (*Design Workshop*)

Pada fase ini, merancang sebuah sistem guna menyelesaikan masalah yang ada pada proses pencatatan dan pengelolaan laporan keluhan dari pelanggan di PT. Satkomindo Mediyasa. Pada tahap ini, melibatkan langsung *user* untuk bisa memberikan komentar apabila terdapat ketidaksesuaian pada desain dengan apa yang dibutuhkan oleh *user*.

#### 3.2.1 Perancangan Basis Data

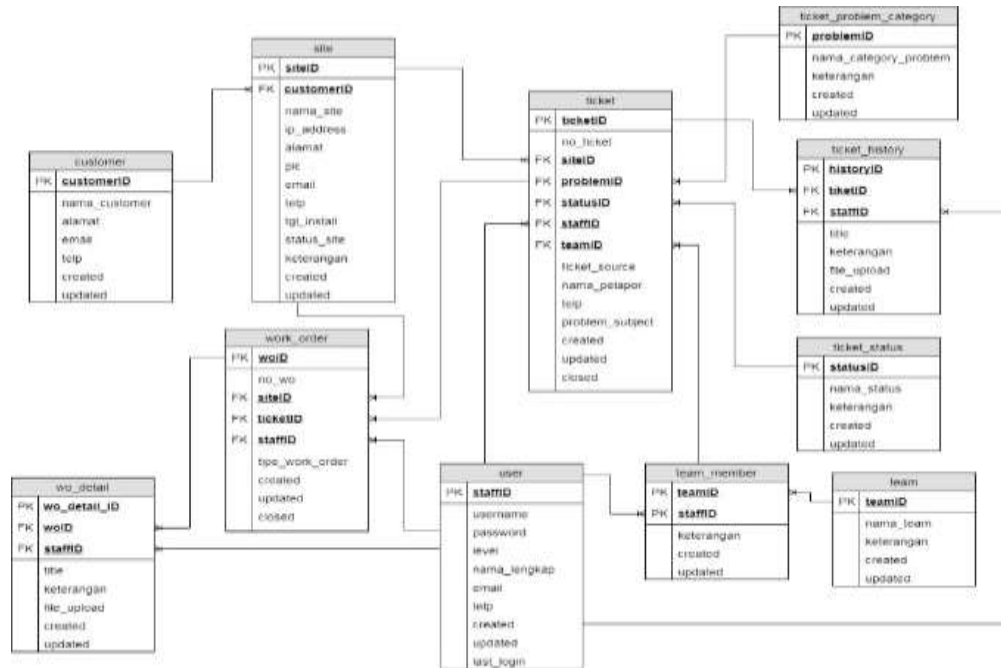
##### a. Perancangan ERD



Gambar. 4 Diagram ERD

##### b. Perancangan LRS





Gambar. 5 Diagram LRS

### 3.2.2 Perancangan Use Case Diagram

a. Penentuan Aktor

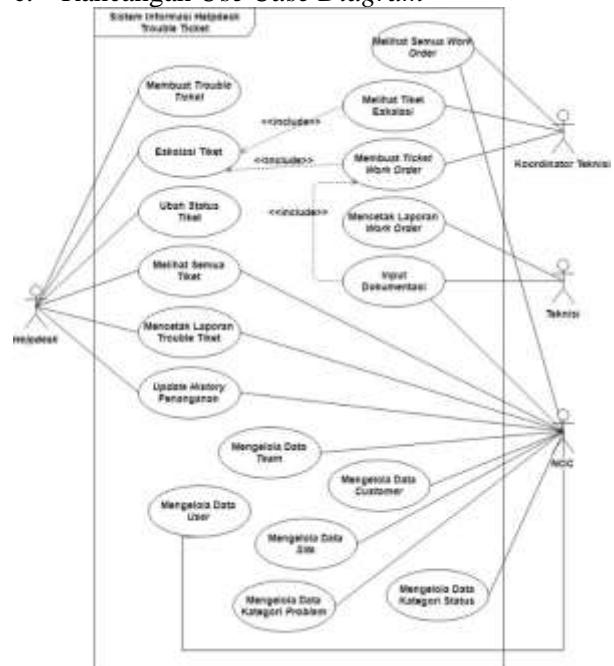
Pada sistem informasi helpdesk trouble ticket ini terdapat 4 buah aktor yang akan terlibat dengan sistem yaitu, *Helpdesk*, Koordinator Teknisi, Teknisi, NOC.

b. Identifikasi Use Case

c. Rancangan Use Case Diagram

Tabel 1 Tabel identifikasi use case

No	Deskripsi	Aktor
1	Membuat <i>Trouble Ticket</i>	<i>Helpdesk</i>
2	Eskalasi Tiket	<i>Helpdesk</i>
3	Ubah Status Tiket	<i>Helpdesk</i>
4	Melihat Semua Tiket	<i>Helpdesk</i> , NOC
5	Mencetak Laporan <i>Trouble Ticket</i>	<i>Helpdesk</i> , NOC
6	<i>Update History</i> Penanganan	<i>Helpdesk</i> , NOC
7	Melihat Semua <i>Work Order</i>	Koordinator Teknisi, NOC
8	Melihat Tiket Eskalasi	Koordinator Teknisi.
9	Membuat <i>Ticket Work Order</i>	Koordinator Teknisi
10	Mencetak Laporan <i>Work Order</i>	Koordinator Teknisi, Teknisi
11	<i>Input Dokumentasi</i>	Teknisi, NOC
12	Mengelola Data <i>User</i>	NOC
13	Mengelola Data <i>Team</i>	NOC
14	Mengelola Data <i>Customer</i>	NOC
15	Mengelola Data <i>Site</i>	NOC
16	Mengelola Data Kategori <i>Problem</i>	NOC
17	Mengelola Data Kategori Status	NOC



Gambar. 6 Perancangan use case diagram

### 3.2.3 Kodefikasi

Kodefikasi atau sistem pengkodean digunakan sebagai alat bantu pengolahan data yang dapat mempermudah dalam pemasukan dan pencarian data. Adapun pengkodean dalam sistem informasi *helpdesk trouble ticket* yang digunakan adalah sebagai berikut:

a. Kode Nomor *Trouble Ticket*



Gambar. 7 Kodifikasi Nomor Trouble Ticket

b. Kode Nomor Work Order

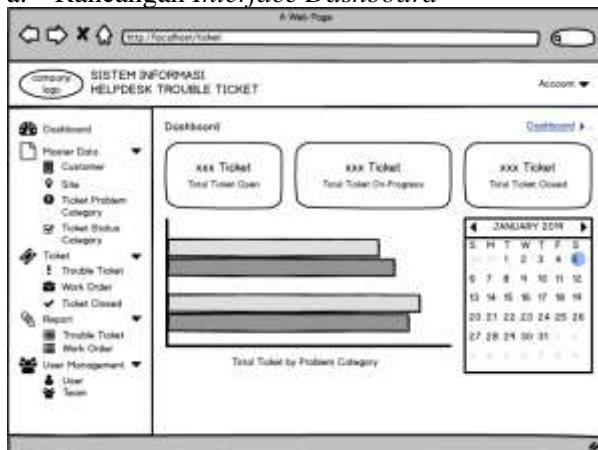


Gambar. 8 Kodifikasi Nomor Work Order

3.2.4 Perancangan Interface

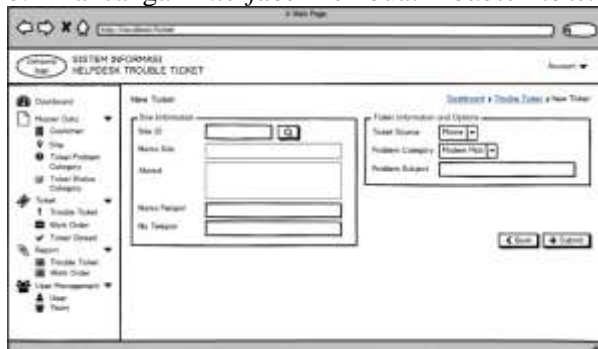
Perancangan *interface* atau antarmuka merupakan perancangan model aplikasi sebelum program dibuat seutuhnya, perancangan antarmuka pada sistem ini digunakan untuk menentukan bagaimana tampilan dari *website* sistem informasi *helpdesk trouble ticket*.

a. Rancangan Interface Dashboard



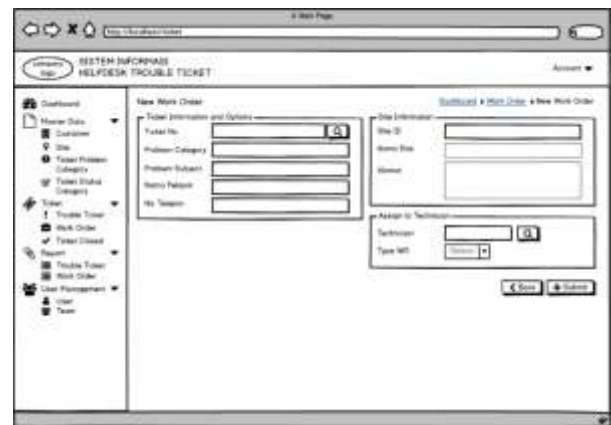
Gambar. 9 Rancangan Interface Dashboard

b. Rancangan Interface Membuat Trouble Ticket



Gambar. 10 Rancangan Interface Membuat Tiket Baru

c. Rancangan Interface Membuat Work Order



Gambar. 11 Rancangan Interface Membuat Work Order

4. Implementasi dan Pengujian

Setelah desain dari sistem dibuat, maka selanjutnya adalah tahap mengembangkan desain menjadi suatu program. Setelah program selesai dikembangkan, maka dilakukan proses pengujian terhadap program tersebut untuk menemukan apakah ada kesalahan atau ketidaksesuaian antara hasil dengan rancangan desain sebelum diaplikasikan pada perusahaan. Pada tahap ini, keterlibatan *user* sangat diperlukan agar sistem yang dikembangkan dapat memberikan kepuasan dan memenuhi kebutuhan *user*.

4.1 Implementasi

4.1.1 Implementasi Sistem

Sistem informasi *helpdesk trouble ticket* ini adalah aplikasi yang berbasis *website* sehingga untuk mengakses aplikasi harus menggunakan aplikasi *browser* seperti Mozilla Firefox, Google Chrome, atau aplikasi *browser* sejenis lainnya. Bahasa pemrograman PHP adalah yang digunakan untuk membangun aplikasi ini serta menggunakan MySQL sebagai *server* basis data. Tampilan antarmuka menggunakan bahasa pemrograman CSS (*Cascading Style Sheet*). Dalam implementasi *web server*, penulis menggunakan XAMPP versi 1.3.8 karena di dalam XAMPP sudah termasuk paket program yaitu Apache sebagai HTTP Server, MySQL database, dan PHP versi 5.5.15 sehingga tidak perlu untuk instal program satu per satu. Sebagai *text editor* dalam penulisan program, penulis menggunakan Sublime Text Editor versi 3.

4.1.2 Spesifikasi Software dan Hardware

Untuk implementasi sistem yang sudah dibangun, ada beberapa spesifikasi perangkat lunak dan perangkat keras yang dibutuhkan. Berikut adalah spesifikasi yang dibutuhkan untuk menjalankan sistem:

a. Perangkat Lunak

Kebutuhan perangkat lunak minimum yang digunakan untuk menjalankan sistem informasi *helpdesk trouble ticket* adalah sebagai berikut:

- 1) Sistem Operasi Windows 7 Ultimate 64-bit
- 2) Text Editor Sublime Text versi 3, digunakan untuk pengkodean sistem.
- 3) Xampp Server 1.8, digunakan sebagai PHP dan SQL *server*.
- 4) *Web Browser*: Mozilla Firefox, Google Chrome, Opera, dan sejenisnya.

b. Perangkat Keras

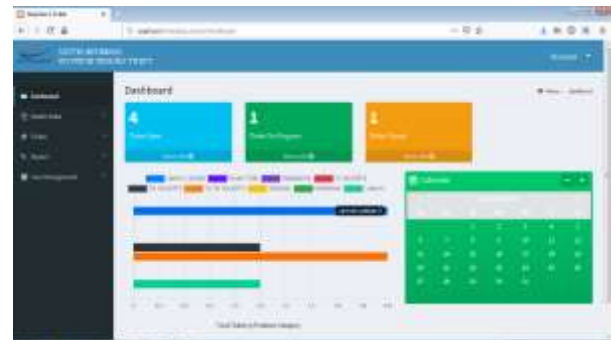
Kebutuhan perangkat keras minimum yang harus disediakan untuk menjalankan sistem adalah sebagai berikut:

- 1) Perangkat *Server*
  - Komputer dengan spesifikasi *Processor Intel Core 2 Duo T6600 2.20GHz*.
  - RAM 4 GB.
  - Hardisk 500 GB.
  - *Monitor/LCD*.
  - *Mouse dan Keyboard*.
  - LAN *card onboard*.
  - *Hub* atau *Switch* untuk koneksi jaringan antar komputer.
- 2) Perangkat *Client*
  - Laptop/Desktop dengan spesifikasi minimum *Processor Intel Pentium IV 1,8 GHz*.
  - RAM minimum 1 GB.
  - *Hardisk* minimum 80 GB.
  - *Monitor/LCD*.
  - *Mouse dan Keyboard*.
  - LAN *card onboard*.
  - *Hub* atau *switch* untuk koneksi jaringan antar komputer.

#### 4.1.3 Implementasi Desain dan Perancangan

Desain dan perancangan yang telah dibuat sebelumnya kemudian diimplementasikan ke dalam pembuatan sistem informasi *helpdesk trouble ticket*.

a. Halaman *Dashboard*



Gambar. 12 Halaman *Dashboard*

b. Halaman Membuat *Trouble Ticket*



Gambar. 13 Halaman Membuat *Trouble Ticket*

c. Halaman Membuat *Work Order*



Gambar. 14 Halaman Membuat *Work Order*

## 4.2 Pengujian

Pengujian merupakan bagian penting dalam pengembangan sebuah sistem, pengujian ditujukan untuk menemukan kesalahan-kesalahan pada sistem dan memastikan sistem yang dibangun telah sesuai dengan apa yang dirancang sebelumnya.

### 4.2.1 Rencana Pengujian

Pengujian pada sistem informasi *helpdesk trouble ticket* ini, penulis menggunakan metode pengujian *blackbox*. Metode pengujian *blackbox* akan lebih difokuskan pada keperluan fungsional dari *software/aplikasi* tersebut. Berikut adalah rencana dari pengujian yang akan dilakukan.



Tabel 2 Tabel Rencana Pengujian

Kelas Uji	Butir Uji	Jenis Pengujian
Login	Pengecekan <i>user</i> terdaftar	<i>Blackbox</i>
Pengelolaan Data <i>Customer</i>	Tambah data <i>customer</i>	<i>Blackbox</i>
	Edit data <i>customer</i>	<i>Blackbox</i>
	Pencarian data <i>customer</i>	<i>Blackbox</i>
Pengelolaan Data <i>Site</i>	Tambah data <i>site</i>	<i>Blackbox</i>
	Edit data <i>site</i>	<i>Blackbox</i>
	Pencarian data <i>site</i>	<i>Blackbox</i>
Pengelolaan Data Kategori <i>Problem</i>	Tambah data kategori <i>problem</i>	<i>Blackbox</i>
	Edit data kategori <i>problem</i>	<i>Blackbox</i>
	Pencarian data kategori <i>problem</i>	<i>Blackbox</i>
Pengelolaan Data Kategori Status	Tambah data kategori status	<i>Blackbox</i>
	Edit data kategori status	<i>Blackbox</i>
	Pencarian data kategori status	<i>Blackbox</i>
Pengelolaan Data <i>User</i>	Tambah <i>user</i>	<i>Blackbox</i>
	Edit <i>user</i>	<i>Blackbox</i>
	Pencarian <i>user</i>	<i>Blackbox</i>
Pengelolaan Data <i>Team</i>	Tambah <i>team</i>	<i>Blackbox</i>
	Edit <i>team</i>	<i>Blackbox</i>
	Tambah <i>member team</i>	<i>Blackbox</i>
	Pencarian <i>team</i>	<i>Blackbox</i>
<i>Trouble Ticket</i>	Membuat <i>trouble ticket</i>	<i>Blackbox</i>
	Eskalasi tiket	<i>Blackbox</i>
	Ubah status tiket	<i>Blackbox</i>
	Tambah <i>history penanganan</i>	<i>Blackbox</i>
	Pencarian <i>trouble ticket</i>	<i>Blackbox</i>
<i>Work Order</i>	Membuat tiket <i>work order</i>	<i>Blackbox</i>
	Tambah data dokumentasi	<i>Blackbox</i>
	Pencarian <i>work order</i>	<i>Blackbox</i>
<i>Report Trouble Ticket</i>	Cetak laporan <i>trouble ticket</i>	<i>Blackbox</i>
<i>Report Work Order</i>	Cetak laporan <i>work order</i>	<i>Blackbox</i>

## 5. Kesimpulan

Berdasarkan hasil implementasi sistem informasi *helpdesk trouble ticket* pada PT. Satkomindo Mediyasa, maka dapat disimpulkan sebagai berikut:

- 1) Dengan sistem informasi *helpdesk trouble tickets*, proses pencatatan dan *update* penanganan laporan keluhan pelanggan bisa dilakukan lebih efektif dan cepat karena pada sistem yang berjalan sebelumnya masih manual menggunakan *form excel*.
- 2) Proses eskalasi antar bagian bisa berjalan lebih cepat karena *user* hanya perlu memilih team yang dituju untuk eskalasi pada *trouble ticket* terdaftar di sistem.
- 3) Sistem informasi *helpdesk trouble ticket* yang dibangun telah dilengkapi dengan fitur mencetak laporan, disertai dengan perhitungan durasi waktu penanganan dari masing-masing laporan keluhan pelanggan.

## Referensi

- Aswati, S., & Siagian, Y. (2016). Model Rapid Application Development Dalam Rancang Bangun Sistem Informasi Pemasaran Rumah (Studi Kasus : Perum Perumnas Cabang Medan). *Seminar Nasional Sistem Informasi Indonesia*, 318-324.
- Basuki, A. P. (2014). *Proyek Membangun Website Berbasis PHP dengan Codeigniter*. Yogyakarta: Andi.
- Cassandra, C. (2015). Pengembangan Model Sistem Informasi Aplikasi Helpdesk Online PT. Mustika Memadata. *ComTech*,6(2), 173-184.
- Chintalois, S. (2015). *Perancangan Aplikasi Helpdesk Ticketing System Berbasis Web Pada Wilmar Group*. Medan: STMIK TIME.
- Mudhar, R. (2015). *Pembangunan Sistem Informasi Helpdesk Ticketing System Menggunakan Django Framework (Studi Kasus : SMK Saradan)*. Jakarta: Universitas Islam Negeri Syarif Hidayatullah.
- Noertjahyana, A. (2002). Studi Analisis Rapid Application Development Sebagai Salah Satu Alternatif Metode Pengembangan Perangkat Lunak. *Jurnal Informatika*,3(2), 74-79.
- Pratama, D. A. (2010). *Pembuatan Aplikasi Trouble Ticket Berbasis Website di PT. Multi Indosarva Sejati*. Surakarta: Universitas Sebelas Maret Surakarta.
- Solichin, A. (2010). *MySQL 5 Dari Pemula Hingga Mahir*. Jakarta: Universitas Budi Luhur.
- Susanto, A. (2017). *Sistem Informasi Manajemen : Konsep dan Pengembangan Secara Terpadu*. Bandung: Lingga Jaya.

- Wibowo, N. I., Metandi, F., & Irwansyah. (2017). Rancang Bangun Aplikasi Helpdesk Berbasis Web pada Jurusan Teknologi Informasi Politeknik Negeri Samarinda. *Jurnal Sains Terapan Teknologi Informasi*,9(2), 125-132.
- Yanto, R. (2016). *Manajemen Basis Data Menggunakan MySQL*. Yogyakarta: Deepublish.