

RANCANG BANGUN SISTEM INFORMASI *HELPDESK* PADA PT *LENTICE SOLUTIONS*

Yohannes Yahya Welim¹, Helga Indra Rukmana²

^{1,2} Program Studi Sistem Informasi, Fakultas Teknologi Informasi, Universitas Budi Luhur
Jl. Ciledug Raya, Petukangan Utara, Jakarta Selatan 12260
Email: yahyabl88@yahoo.com¹

Abstrak

PT *Lentice Solutions* merupakan usaha bisnis dalam bidang IT berupa pembuatan sistem aplikasi untuk perusahaan, pengadaan pelatihan, pembuatan sistem aplikasi *accounting*, dan pengembangan aplikasi *security*. Pada saat pengembangan sistem aplikasi yang ada di *customer*, sering ditemukan beberapa kesalahan sistem (*issue*) yang muncul. Hal ini menyebabkan terhambatnya proses bisnis dalam perusahaan. Jika ini terjadi maka *customer* melaporkan *issue* yang melalui email kepada *support Lentice* untuk segera dilakukan eskalasi oleh *engineer*. Sistem informasi *helpdesk* di PT *Lentice Solutions* masih menggunakan *email* yang dinilai belum efektif dalam penanganan eskalasi tiket, sehingga kinerja *engineer* tidak optimal. Penanganan tiket tidak optimal karena tidak adanya pengaturan waktu tiket. *Update progress* tiket dari *engineer* sering terlambat dikirim. Hal ini menyebabkan *customer* tidak puas dengan kinerja *engineer*. Oleh karena itu dibutuhkan sistem informasi yang memadai agar proses eskalasi menjadi lebih cepat dan tepat. Hasil yang dicapai adalah membantu *support* dan *engineer* dalam menangani tiket, *Customer* dapat mengetahui perkembangan eskalasi penanganan tiket melalui notifikasi *update* yang dikirim otomatis ke *email* setelah *engineer* melakukan *update* eskalasi dan Pimpinan dapat menilai kinerja *engineer* dengan melihat performa *engineer* berdasarkan tiket yang dikerjakan serta perusahaan dapat mengetahui seberapa efektif *action plan* yang digunakan saat eskalasi tiket dilaksanakan maka penanganan tiket menjadi lebih optimal.

Kata kunci : *tiket, engineer, customer*

1. PENDAHULUAN

1.1. Latar Belakang

Dalam dunia industri, perkembangan informasi begitu cepat. Perputaran data mengalir setiap hari. Hal ini mengakibatkan rentan terjadinya kesalahan dalam sistem yang seharusnya memberikan performa terbaik saat digunakan. Sebagai pelaku perancang dan pemegang kendali sistem, kitadituntut untuk menyelesaikan permasalahan yang ada dengan cepat, tepat, dan akurat sehingga akan mendapat hasil yang optimal.

PT *Lentice Solutions* adalah bentuk usaha bersama yang berperan menyelenggarakan fungsi bisnis dalam bidang IT berupa pembuatan sistem aplikasi untuk perusahaan, pengadaan pelatihan, pembuatan sistem aplikasi *accounting*, dan pengembangan aplikasi *security* pada sistem.

Dalam pengembangan sistem aplikasi yang ada di *customer*, sering ditemukan beberapa kesalahan sistem (*issue*) yang muncul. Hal ini menyebabkan terhambatnya proses bisnis dalam perusahaan. Jika ini terjadi maka *customer* akan segera melaporkan *issue* yang muncul melalui email kepada tim *support* PT *Lentice* untuk segera dilakukan eskalasi oleh *engineer*. Namun dengan proses yang masih menggunakan *email* sering membuat proses eskalasi berjalan tidak optimal.

Proses eskalasi permasalahan sistem ini berjalan optimal dan menghasilkan solusi penanganan yang akurat, maka diperlukan sistem informasi yang cukup handal. Berdasarkan alasan tersebut, penulis dalam penelitian ini mengambil judul “Sistem Informasi *Helpdesk* pada PT *Lentice Solutions*”

1.2. Rumusan Masalah

Berdasarkan analisis yang telah dilakukan maka rumusan masalah yang akan dibahas adalah:

1. Bagaimana penanganan tiket yang optimal ?
2. Bagaimana Pengaturan Waktu Tiket ?

3. Bagaimana *Update progress* tiket dari engineer terlambat pengirimannya ?
4. Bagaimana kinerja *engineer* ?

2. METODOLOGI

2.1. Tahap Analisa Sistem

Kegiatan yang dilakukan pada tahap ini adalah :

1. Menganalisa sistem yang ada yaitu dengan mempelajari dan mengetahui apa yang dikerjakan sistem yang ada.
2. Menspesifikasikan sistem yaitu dengan menspesifikasikan masukan yang digunakan, *database* yang ada, proses yang dilakukan dan keluaran yang dihasilkan.

Adapun tahapan-tahapan pada analisa sistem antara lain :

- a. *Activity Diagram*
Activity Diagram adalah alat untuk memodelkan alur kerja atau *workflow* sebuah proses bisnis dan urutan aktifitas didalam suatu proses.
- b. *Use Case Diagram*
Use Case Diagram adalah alat untuk mendeskripsikan fungsi dari sebuah sistem dari perspektif pengguna.
- c. *Use Case Description*
Use Case Description adalah alat untuk mendeskripsikan secara rinci mengenal *use case*.
- d. *Entity Relationship Diagram*
Entity Relationship Diagram adalah alat yang dapat mempresentasikan hubungan yang terjadi antara satu atau lebih komponen sistem.

2.2. Tahap Perancangan Sistem

Tahap Perancangan Sistem adalah merancang sistem secara rinci berdasarkan hasil analisa sistem yang ada, sehingga menghasilkan model sistem baru yang akan diusulkan, dengan disertai rancangan *database* dan spesifikasi program.

Adapun tahapan – tahapan pada rancangan sistem antara lain :

1. *Logical Record Structure*
Logical Record Structure terdiri dari *link-link* diantara tipe *record*. *Link* ini menunjukkan arah dari satu tipe *record* lainnya.
2. Normalisasi
Normalisasi digunakan untuk mengorganisasikan *file* dengan menghilangkan grup elemen yang berulang atau sebuah langkah (proses) untuk menyederhanakan hubungan elemen data di dalam *record*.
3. Relasi
Relasi digunakan untuk mendefinisikan dan mengilustrasikan model konseptual secara terperinci dengan adanya *primary key* dan *foreign key*.
4. Spesifikasi Basis Data
Spesifikasi basis data digunakan untuk menjelaskan tipe data yang ada pada model konseptual secara detail.

3. HASIL DAN PEMBAHASAN

3.1. Konsep Dasar Sistem

Sistem dalam bidang sistem informasi adalah Sekelompok komponen yang saling berhubungan, bekerja sama untuk mencapai tujuan bersama dengan menerima proses *input* serta menghasilkan *input* dalam proses transformasi yang teratur. (Agus Mulyanto, 2009).

Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, terkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk tujuan tertentu. (Yakub, 2012).

Konsep Dasar Informasi

Informasi adalah hasil dari pengolahan data dalam suatu bentuk yang lebih berguna dan berarti bagi penerimanya yang menggambarkan suatu kejadian-kejadian (*event*) yang nyata digunakan untuk pengambilan keputusan.

Data dan informasi merupakan sebuah pondasi untuk memahami konsep sistem informasi. (Agus Mulyanto, 2009).

Informasi dapat memiliki kualitas yang tergantung pada kriteria berikut :

1. Akurat
Informasi harus bebas dari kesalahan–kesalahan dan tidak biasa atau menyesatkan.
2. Tepat pada waktunya
Informasi yang diberikan atau datang pada penerima tidak boleh terlambat.
3. *Relevan*
Informasi tersebut mempunyai manfaat untuk pemakainya. Suatu sistem informasi dikatakan bernilai bila manfaatnya lebih efektif dibanding dengan biaya pendapatannya.
4. Lengkap
Informasi yang dikirim harus lengkap agar tidak salah pemahamannya.

3.2. Konsep Dasar Sistem Informasi

Pengertian sistem informasi adalah Sistem informasi yang saling berikatan untuk memberikan data, memproses, dan menyimpan satu sama lain yang membentuk satu kesatuan informasi. (Agus Mulyanto, 2009)

3.3. Konsep Dasar Analisa Sistem

Analisa sistem adalah penguraian suatu sistem informasi yang utuh ke dalam bagian komponennya yang bertujuan untuk mengidentifikasi dan mengevaluasi permasalahan, kesempatan, hambatan yang terjadi dan menentukan kebutuhan sehingga dapat diusulkan perbaikannya.

Adapun Langkah-langkah dalam analisa secara umum sebagai berikut :

1. Analisa Pendahuluan
Dalam analisa pendahuluan ini dilakukan pengumpulan informasi yang memperoleh gambaran secara menyeluruh mengenai tempat yang akan kita analisa. Pelaksanaan analisa sistem dirancang oleh analisa sistem dalam suatu dokumen tertulis yang disebut usulan pelaksanaan sistem. Dengan dihasilkannya dokumen tertulis tersebut adalah untuk mempertemukan pikiran pemakai informasi dengan analisis sistem untuk memenuhi kebutuhan pemakai informasi.
2. Pelaksanaan analisa sistem
Pelaksanaan analisa sistem didasarkan pada rancangan kerja yang dituangkan dalam usulan analisa sistem.
3. Penyusunan laporan hasil analisa sistem
Hasil akhir analisa sistem disajikan dalam bentuk suatu laporan yang berguna dalam pengambilan keputusan.

3.4. Teknologi Berbasis Web

Aplikasi berbasis web yang menggunakan *server web* untuk operasionalnya. *Server web* adalah perangkat lunak yang memberikan layanan data yang berfungsi menerima permintaan HTTP dari *browser* dan mengirimkan hasilnya dalam bentuk tampilan web, yang menggunakan *apache* sebagai *server web*nya dengan bahasa pemrograman sebagai berikut :

1. HTML (*Hypertext Markup Language*)
Merupakan bahasa standar yang digunakan untuk membuat halaman *web*. Saat ini penggunaannya dikendalikan oleh *World Wide Consortium (W3C)*. HTML berisi perintah-perintah yang telah terstruktur berupa *tag-tag* penyusun. Menuliskan *tag* HTML tidaklah sebatas hanya memasukkan perintah-perintah tertentu agar HTML dapat diakses oleh *browser* tetapi dalam *tag* HTML terdapat atribut yang bisa digunakan untuk merancang tampilan
2. PHP (*Hypertext Preprocessor*)
Adalah sebuah bahasa pemrograman yang berjalan dalam sebuah *server web*. Oleh karena itu, PHP dapat mengolah data dengan banyak tipe data. Bahasa pemrograman PHP adalah bahasa

kode yang tidak memerlukan sebuah kompilasi dalam penggunaannya. PHP dapat melayani banyak *server web*. PHP bersifat *open source* sehingga mudah untuk dikembangkan dan dapat dijalankan di berbagai macam sistem operasi. Sistem ini menggunakan PHP *versi 5.3.5*.

3. Framework

Adalah koleksi atau kumpulan potongan-potongan program yang disusun atau diorganisasikan sedemikian rupa sehingga dapat digunakan untuk membantu membuat aplikasi utuh tanpa harus membuat semua kodenya dari awal. *Codeigniter* adalah salah satu *framework* PHP yang dibuat berdasarkan metode *Model View Controller (MVC)* oleh *Rick Ellis* berbasis PHP 4, bersifat *open source* dan gratis. Konsep MVC ini memisahkan antara model untuk manipulasi basis data, *view* untuk manipulasi tampilan layar, dan *controller* untuk mengontrol aliran aplikasi.

3.5. Konsep Helpdesk

1. Pengertian Helpdesk

Helpdesk merupakan sistem manajemen untuk membantu menangani kebutuhan customer / user terkait dengan pertanyaan, pelayanan, *support* teknis, atau komplain terhadap produk dan jasa tertentu dengan memanfaatkan sistem penomoran (*request* tiket) untuk memudahkan penelusuran terhadap tindakan penyelesaian yang dikoordinasi oleh suatu tim. Tim ini biasanya terdiri dari satu atau beberapa orang yang bertugas yakni :

- a. Menampung, mengklasifikasi dan memberikan prioritas terhadap *request* tiket melalui berbagai jalur seperti *email*, *website*, atau telpon.
- b. Melakukan *assignment request* tiket kepada unit-unit yang bertanggung jawab dan memiliki kapabilitas untuk menyelesaikan permasalahan tersebut. Jika diperlukan melakukan eskalasi *request* tiket ke tingkatan manajemen yang lebih tinggi.
- c. Monitoring durasi dan status penyelesaian, dan melakukan *logging* terhadap seluruh tahapan tindakan yang telah diambil dan status penyelesaian *request* tiket.

2. Tujuan Helpdesk

Meminimalkan setiap problem yang terjadi dalam aktivitas pengguna yang bersangkutan dengan sistem yang terkait dengan *helpdesk* tersebut, dan dapat menghemat waktu dari berbagai problem yang terjadi. Sebagai suatu sistem yang didukung oleh perangkat organisasi, teknologi berbasis *web* dan kelompok tenaga ahli, pembangunan *helpdesk* bertujuan untuk :

- a. Menghimpun, mendokumentasikan dan menyebarkan pengetahuan dan pengalaman dalam mengatasi permasalahan operasi dan pemeliharaan.
- b. Menata, meregistrasi dan memobilisasi tenaga ahli (*experts*) yang masih aktif untuk memberikan kontribusi dalam penyelesaian permasalahan, terutama di daerah yang masih minim pengalaman.

3. Pemanfaatan Helpdesk di Berbagai Bidang

Berikut contoh pemanfaatan aplikasi *helpdesk* di berbagai bidang, antara lain :

- a. Perusahaan penyedia jasa layanan publik seperti *operator* telekomunikasi, *web hosting*, toko *online*, atau perusahaan yang memiliki pangsa pasar produk yang cukup luas, dengan tujuan untuk meningkatkan hubungan dengan *customer (customer relationship management)*.
- b. Departemen IT di berbagai instansi / perusahaan dalam melakukan tugasnya melakukan support dalam menyelesaikan permasalahan terkait infrastruktur jaringan, komputer atau *problem* perangkat lunak.
- c. Departemen Pemasaran atau Pengembangan Produk dalam menampung masukan-masukan dari para *customer* melalui sistem guna melakukan analisa-analisa atau perencanaan ke depan.
- d. Perusahaan pengembang *software* dalam mengatur *bug*, dan memonitor penyelesaiannya.

3.6. Uraian Prosedur

Pada sistem berjalan untuk proses eskalasi *issue* pada PT. *Lentice Solutions* melalui beberapa tahap, yaitu :

1. Proses Identifikasi *Issue*

Pada tahap awal *customer* melaporkan *issue* melalui *email* kepada *support* *Lentice*. *Support* menerima *email issue* dari *customer* lalu *support* mengecek *scope of work issue* tersebut apakah *issue* itu diterima atau bukan. Jika bukan, maka *support* akan merespon *email* ke *customer* untuk konfirmasi pengembalian *issue*. Tapi jika *issue* memang ditujukan untuk *Lentice*, maka *support* akan mengecek *Issue* untuk segera dicocokkan dengan *database Record Problem History and Solution Knowledge Base* yang ada di *server internal*. Kemudian *support* menugaskan *case issue* kepada *engineer level 2* serta merespon ke *customer* dengan mengirim *email* berupa nomer tiket.

2. Proses Analisa dan Eskalasi *Issue*

Engineer level 2 menerima *case issue* yang dikirim oleh *support* untuk dianalisa sesuai *knowledge base* yang sudah ada. Jika bisa dieskalasi secara langsung, maka *engineer* akan mengirimkan *email* respon ke *customer* kemudian dilanjutkan ke *support* terkait *progress* status *issue*. *Email* yang dikirim ke *support* selain status *issue* juga dikirim *action plan* (langkah-langkah eskalasi *issue*). Namun jika belum selesai maka *issue* akan di *assign* melalui *email* ke *engineer level 3* untuk eskalasi lanjut.

3. Proses Analisa dan *Support Issue*

Setelah menerima *issue* dari *engineer level 2* untuk dianalisa, jika sudah ditemukan solusinya, *engineer level 3* akan mengirimkan *email* berisi solusi alternatif kepada *engineer level 2*. Namun jika belum ditemukan, *engineer level 3* akan mengeskalasi *issue* ke pihak pengembang ketiga. *Issue* akan dianalisa dan dieskalasi lagi untuk nanti dikirimkan kembali hasil eskalasi dan solusi alternatifnya ke *engineer level 2*.

4. Proses *Closing* Tiket

Support menerima *email* dari *engineer level 2* terkait *issue* yang sudah dianalisa dan di eskalasi. Jika *issue* masih belum selesai maka *support* akan mengirim kembali ke *engineer level 2*. Tapi jika *issue* telah selesai, *support* akan *closing* tiket dan mengirim hasil eskalasi *issue* ke *customer*.

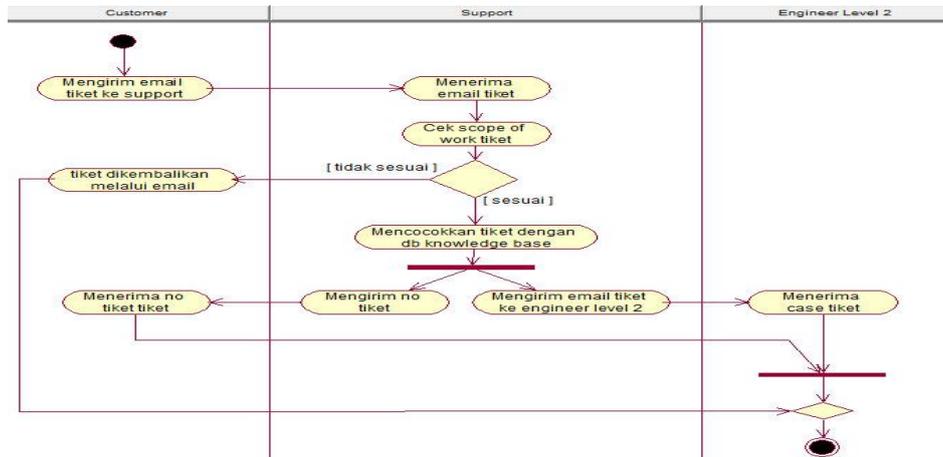
5. Proses *Reporting*

Setelah hasil *eskalasi* dan *closing issue* selesai, *support* kemudian membuat *daily report* berupa *detail open* dan *close issue* per hari ke *engineer level 3*. *Engineer level 3* menerima *email report* untuk diarsipkan.

3.7. Analisa Proses

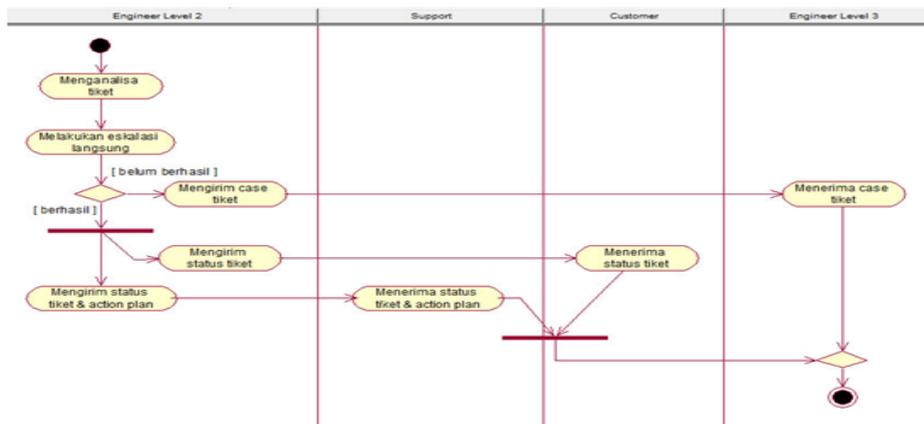
Menggambarkan diagram aktifitas yang menunjukkan penggunaan lampiran masukan dan keluaran yang dipakai pada sistem.

1. Activity Diagram Proses Identifikasi Tiket



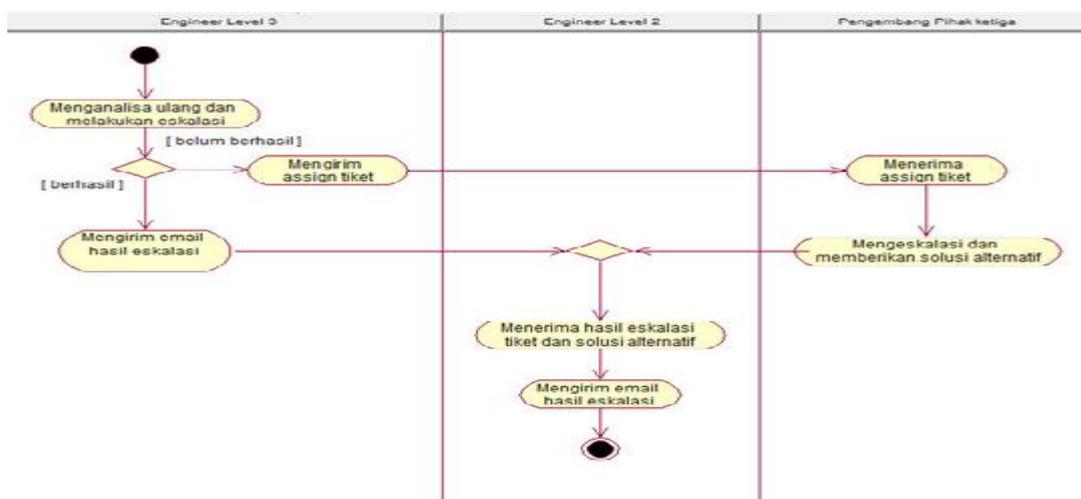
Gambar 1. Activity diagram Proses Identifikasi Tiket

2. Activity Diagram Proses Analisa dan Eskalasi Tiket



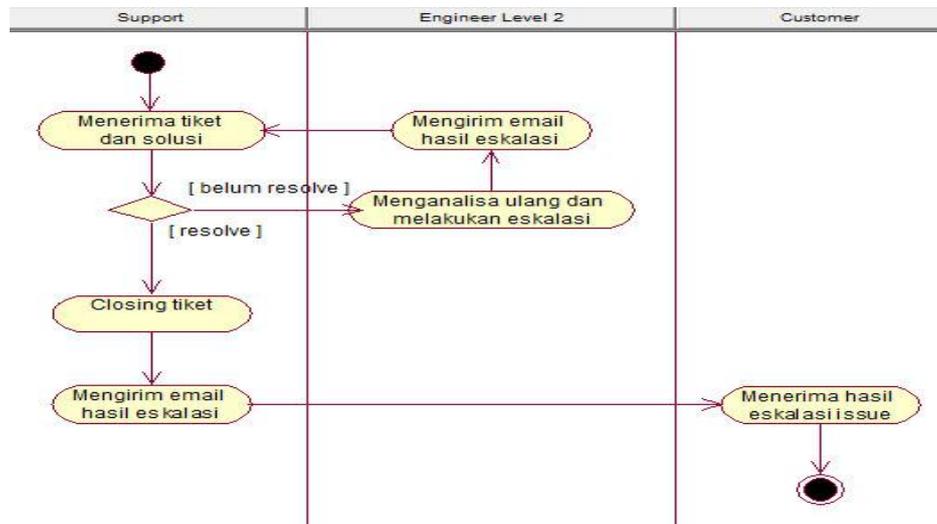
Gambar 2. Activity diagram Proses Analisa dan Eskalasi

3. Activity Diagram Proses Analisa dan Support Issue



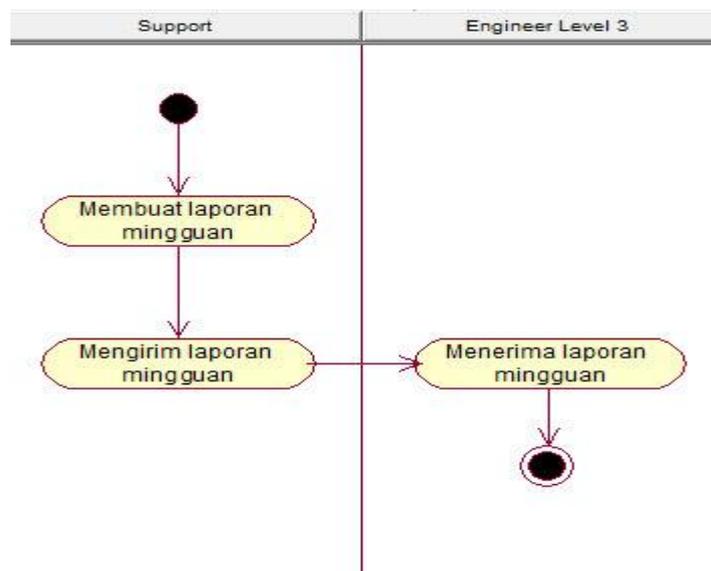
Gambar 3. Activity diagram Proses Analisa dan Support Issue

4. Activity Diagram Proses Resolving Issue dan Closing Tiket



Gambar 4. Activity diagram Proses Resolving Issue dan Closing Tiket

5. Activity Diagram Proses Reporting



Gambar 5. Activity diagram Proses Reporting

3.8. Package Diagram

Untuk mempermudah pembuatan use case maka proses dibagi menjadi beberapa package diagram pada gambar di bawah.

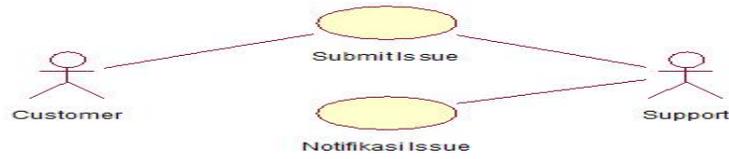


Gambar 6. Package Diagram

3.9. Use Case Diagram

1. *Front End*

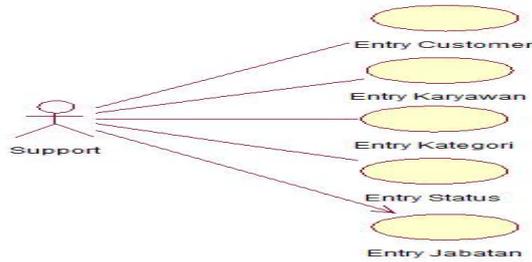
a. Use Case Transaksi



Gambar 7. Use Case Transaksi Front End

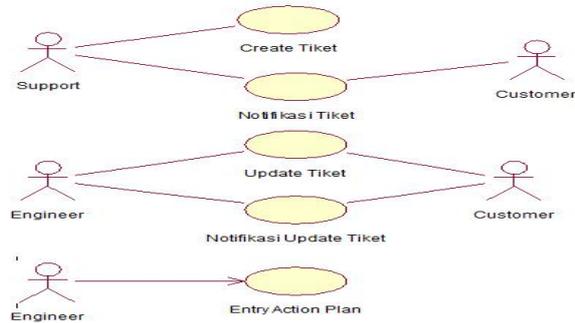
2. *Back End*

b. Use Case Master



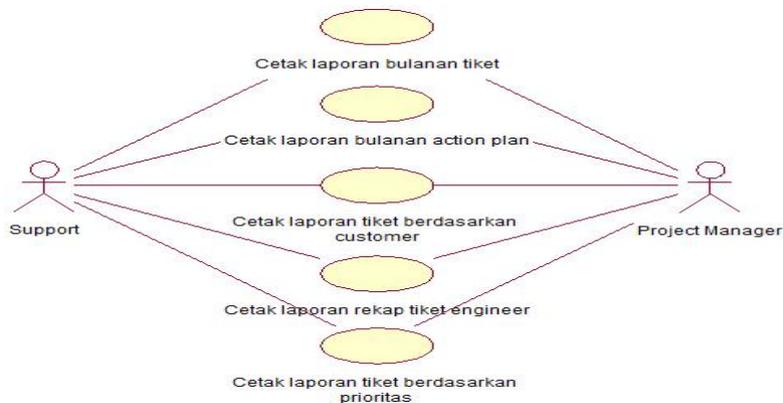
Gambar 8. Use Case Master

c. Use Case Transaksi



Gambar 9. Use case Transaksi Back End

d. Use Case Laporan



Gambar 10. Use Case Laporan

4. KESIMPULAN

Atas dasar analisis dan pembahasan di atas maka dapat ditarik beberapa kesimpulan sebagai berikut :

1. Dengan adanya sistem informasi *helpdesk* ini, dapat membantu *support* dan *engineer* memajemen tiket agar penanganannya tidak terlalu lama.
2. *Customer* bisa lebih cepat mengetahui perkembangan eskalasi penanganan tiket dengan adanya notifikasi *update* yang dikirim otomatis melalui *email* sesaat setelah *engineer* melakukan *update* eskalasi.
3. Atasan dapat menilai kinerja *engineer* dengan melihat performa *engineer* berdasarkan tiket yang dikerjakan.
4. Dengan sistem *helpdesk* ini, perusahaan dapat mengetahui seberapa efektif *action plan* yang digunakan pada saat eskalasi tiket sehingga penanganan tiket menjadi lebih optimal.

DAFTAR PUSTAKA

- Basuki, Awan P. 2010. Membangun Web Berbasis PHP dengan Framework Code Igniter. Penerbit: Lokomedia, Yogyakarta.
- Hermawan, Julius. 2007. Analisa Desain & Pemrograman Berorientasi Obyek dengan UML. Penerbit: Andi, Yogyakarta.
- Indrajani. 2011. Perancangan Basis Data dalam All in 1. Penerbit: PT Elex Media Komputindo, Jakarta.
- Laudon, Kenneth C. and Laudon, Jane P. 2009. Sistem Informasi Manajemen. Penerbit: Palgrave, Basingstoke.
- Mulyanto, Agus. 2008. Sistem informasi konsep dan aplikasi. Penerbit: Pustaka Belajar, Yogyakarta.
- P, Beynon Davies. 2009. Sistem Informasi Manajemen. Penerbit: Palgrave : Basingstoke, 2009.
- Rusmawan, UUS. 2011. Koleksi Program VB.NET Untuk Tugas Akhir dan Skripsi. Penerbit : PT. Elex Media Komputindo, Jakarta.
- Yakub. 2012. Pengantar Sistem Informasi. Penerbit: Graha Ilmu, Yogyakarta.