

# Penerapan *Helpdesk Ticketing System* Dalam Penanganan Keluhan Penggunaan Sistem Informasi Berbasis *Web*

Ryan Muhammad Bahrudin<sup>1</sup>, Mohammad Ridwan<sup>2</sup>, Hardjito S Darmojo<sup>3</sup>

<sup>1,2,3</sup>Program Studi Teknik Informatika, Universitas Islam Syekh Yusuf Tangerang  
Jl. Maulana Yusuf Tangerang 15118, telp. (021) 55270611-5527063 fax. 021-5581068

<sup>1</sup>ryanmuhammadbahrudin@gmail.com

<sup>2</sup>mridwan@unis.ac.id

<sup>3</sup>hardjito\_sd@yahoo.co.id

*Intisari*— Unit Pelaksana Tugas Information Technology (UPT IT) merupakan pusat penyedia informasi Akademik yang ada pada kampus Universitas Islam Syekh-Yusuf. ketika terjadi keluhan tentang gangguan Sistem Informasi Akademik (SINA) seperti nilai tidak sesuai, masalah registrasi atau masalah keuangan, user masih harus datang langsung ke kantor teknisi IT untuk melaporkan keluhannya, user juga bingung harus melapor kesiapa ketika teknisi IT tidak berada ditempat laporan-laporan yang masuk tidak terdokumentasi dengan baik. Untuk mengatasi masalah diatas dibutuhkan suatu aplikasi helpdesk ticketing untuk menjadi wadah user melaporkan masalah. Pendekatan pemrograman sistem ini menggunakan metode prototype, aplikasi ini dirancang memanfaatkan web application sebagai platformnya dengan MYSQL sebagai databasenya. Hasil dari penelitian ini adalah terancangnya aplikasi helpdesk ticketing untuk mempermudah user dalam melakukan pelaporan masalah, dan memudahkan Teknisi IT dalam mengelola laporan.

*Kata kunci*— Aplikasi Helpdesk, Prototype, PHP & MYSQL

*Abstract*— The Information Technology Task Force (UPT IT) is the center for providing Academic information on the campus of the Syekh-Yusuf Islamic University. when there is a complaint about the disruption of the Academic Information System (SINA) such as inappropriate value, registration problems or financial problems, the user still has to come directly to the IT technician's office to report his complaint, the user is also confused about reporting when IT technicians are not in the reports the entry is not well documented. To overcome the above problems a helpdesk ticketing application is needed to become a user forum to report problems. The development approach of this system uses the prototype method, this application is designed to use the web application as a platform with MYSQL as the database. The results of this study are the design of a helpdesk ticketing application to make it easier for users to report problems, and make it easier for IT Technicians to manage reports.

*Keywords*— Helpdesk Application, Prototype, PHP & MYSQL.

## I. PENDAHULUAN

### A. Latar Belakang

Tuntutan kebutuhan akan informasi dan penggunaan aplikasi komputer mendorong terbentuknya suatu aplikasi yang mampu menampung kegiatan dan permasalahan pada mahasiswa, dosen dan karyawan selama perkuliahan serta kegiatan operasional di kampus. *Helpdesk* adalah bantuan berupa informasi dan pengganti penyedia informasi yang menangani aktifitas dan keluhan dari

segenap *civitas* Universitas Islam Syekh Yusuf (UNIS) Tangerang.

Pada setiap akan memulai kegiatan belajar awal semester setiap mahasiswa diwajibkan untuk melakukan pembayaran registrasi, pengisian Kartu Rencana *Study* (KRS), dan pembayaran biaya kuliah. Dengan semakin majunya teknologi sistem Akademik UNIS melakukan migrasi sistem dari sistem yang lama bernama Akademik Informasi Sistem (AIS) menjadi Sistem Informasi Akademik (SINA). Ketika akan Ujian Tengah Semester (UTS) atau juga pada saat Ujian Akhir

Semester (UAS) sering sekali mahasiswa mengadakan laporan perihal registrasi ataupun masalah nilai yang tidak sesuai karena migrasi dari sistem lama ke sistem baru. *User* menyampaikan keluhan secara langsung atau melalui *Short Message Service (SMS)* atau *telephone*, karena pelaporan masih menggunakan *SMS* atau *telephone* pelapor terkadang hanya melaporkan masalah kepada satu orang teknisi *Information Technology (IT)* yang mana membuat teknisi *IT* tersebut tidak bisa memberikan solusi kepada *user* karena terlalu banyak laporan yang masuk kepada teknisi *IT* tersebut, *user* juga harus menunggu lama ketika bertanya melalui sms yang ditujukan hanya kepada satu orang teknisi *IT*, terkadang *user* bingung harus melaporkan kepada siapa ketika teknisi *IT* sedang sibuk atau tidak berada ditempat, *user* terkadang melaporkan masalah yang sama dengan *user* lainnya, laporan-laporan yang masuk hanya sebatas tersimpan pada *SMS* teknisi *IT*, jika *user* melaporkan melalui *telephone* maka laporan tersebut tidak dapat terdokumentasi. Karena keterbatasan *memory handphone* dalam menampung laporan-laporan yang tersimpan dalam *SMS* dan bercampur dengan *SMS* pribadi teknisi *IT* mengakibatkan laporan-laporan terdahulu hilang atau terhapus .

Dari permasalahan tersebut, dibutuhkan sebuah dokumentasi keluhan dengan merancang aplikasi *helpdesk* yang dapat memudahkan *user* dalam melaporkan keluhan kepada teknisi *IT* yang ada di kampus UNIS Tangerang dan memberikan informasi mengenai status keluhan.

Pada penelitian ini peneliti akan merancang bangun aplikasi *helpdesk* berbasis *website* sehingga menjadi suatu aplikasi yang mudah diakses oleh *user* dan teknisi *IT*. Peneliti berharap aplikasi ini dapat memudahkan *user* dalam hal pelaporan agar cepat direspon oleh teknisi *IT*, serta teknisi *IT* dapat melihat rekam jejak pelaporan tersebut.

#### *B. Rumusan Masalah*

Dari rincian uraian di atas, maka terdapat permasalahan yang dapat dirumuskan adalah:

1. Bagaimana caranya agar teknisi *IT* dapat mengatur laporan-laporan yang begitu banyak ?
2. Bagaimana cara memberikan solusi dari laporan atau pertanyaan *user* dengan cepat dan efisien ?
3. Bagaimana caranya agar *user* tidak bingung ketika teknisi *IT* sedang sibuk atau tidak berada ditempat ?
4. Bagaimana mengatasi pertanyaan-pertanyaan yang sama yang dilakukan berulang-ulang oleh *user* ?
5. Bagaimana mengatasi laporan-laporan yang tidak terdokumentasi dengan baik ?

#### *C. Batasan Masalah*

Dari latar belakang masalah yang penulis temukan dalam penelitian ini, penulis akan memberikan batasan masalah supaya permasalahan tidak melebar. Batasan masalah yang akan dieksplorasi oleh penulis adalah sebagai berikut, lingkup aplikasi ini hanya *IT helpdesk*, pengguna aplikasi ini hanya mahasiswa dan staf Unit Pelaksana Tugas (UPT) UNIS Tangerang, aplikasi tidak menggunakan *deadline* pertiket aplikasi dibangun berbasis *website* dengan *php Programming* dan *databasenya MySQL*.

#### *D. Manfaat Penelitian*

Manfaat dari penelitian ini terdiri dari dua macam, yaitu :

##### 1. Manfaat Teoritis

Hasil dari penelitian ini sebagai landasan dalam perancangan sistem aplikasi *helpdesk* secara berkelanjutan. Selain itu menjadi sebuah nilai tambah ilmu pengetahuan ilmiah di dunia pendidikan Indonesia.

##### 2. Manfaat Praktis

- a. Mempermudah teknisi *IT* dalam menerima laporan *user* secara berkala dan bisa di akses dimana saja kapan saja.

- b. Mempermudah *user* dalam melakukan pelaporan.
- c. Mempermudah dalam pengolahan data laporan.

#### *E. State Of The Art*

Beberapa literatur yang digunakan sebagai pedoman dan acuan untuk merancang aplikasi *Helpdesk* antara lain:

Wibowo & Metandi, (2017) melakukan penelitian serupa, peneliti merancang aplikasi *helpdesk* berbasis *web* pada Jurusan Teknologi Informasi Politeknik Negeri Samarinda, peneliti menggunakan metode pengembangan sistem *waterfall* hasil penelitian ini adalah *user* hanya melaporkan masalah yang terjadi tanpa ada informasi solusi yang ditawarkan karena aplikasi ini dirancang hanya untuk melaporkan fasilitas yang rusak<sup>1</sup>.

Rico, (2016) melakukan penelitian serupa yaitu merancang sistem *helpdesk* pada PT. Lontar *Papyrus Pulp & Paper* industri berbasis *web* dengan metode pengembangan sistem *waterfall*, hasil dari penelitian ini adalah aplikasi *helpdesk* dibangun menggunakan prioritas laporan, setelah menerima laporan admin akan mengirim proses penugasan kepada petugas melalui *SMS* atau *web*<sup>2</sup>.

Suryono dan Saptono, (2017) melakukan penelitian serupa dengan mengembangkan aplikasi *helpdesk* menggunakan *algoritma naive bayes classifier* hasil penelitian ini adalah terancangnya aplikasi *smart helpdesk* yang dapat mengklasifikasikan laporan tanpa peran operator atau admin.

Persamaan penelitian ini dengan penelitian sebelumnya yaitu aplikasi ini sama-sama dirancang berbasis *website*, terdapat *menu form* untuk mengirim laporan kepada petugas dan juga terdapat *menu informasi* untuk mendapatkan informasi dari

hasil pertanyaan yang diajukan *user* kepada petugas.

Perbedaan penelitian ini dengan penelitian terdahulu adalah pada penelitian ini aplikasi *helpdesk ticketing* berbasis *web* dibangun menggunakan metode pengembangan sistem *prototype* dan menggunakan bahasa pemrograman *php* dengan *framework CodeIgniter (CI)* dan *Mysql* sebagai *database*. Aplikasi ini dibangun menggunakan kategori masalah, pada penelitian terdahulu aplikasi dibangun tidak memiliki *menu* populer solusi, populer solusi diambil dari pertanyaan yang sering ditanyakan kepada teknisi. Jika terjadi masalah yang sama dikemudian hari dan teknisi *IT* tidak berada ditempat maka diharapkan *menu* populer solusi menjadi alternatif untuk memudahkan *user* menemukan solusi atas masalah yang dihadapi tanpa harus bertanya kepada teknisi *IT*.

## II. LANDASAN TEORI

### *A. Rancang Bangun*

Rancang bangun merupakan serangkaian prosedur untuk menerjemahkan hasil analisa dari sebuah sistem ke dalam bahasa pemrograman untuk mendeskripsikan dengan detail bagaimana komponen-komponen sistem di implementasikan<sup>4</sup>.

Perancangan adalah langkah pertama dalam fase pengembangan rekayasa produk atau *system*. Fase ini adalah inti teknis dari proses rekayasa perangkat lunak. Pada fase ini elemen-elemen dari model analisa dikonversikan. Dengan menggunakan satu dari sejumlah metode perancangan, fase perancangan akan menghasilkan perancangan data, perancangan antar muka, perancangan arsitektur dan perancangan prosedur (Manopo, 2016).

### *B. Aplikasi*

Aplikasi adalah sekumpulan perintah program yang di buat untuk mempermudah pengguna dalam melakukan tugas-tugas

tertentu yang berhubungan dengan pengelolaan informasi-informasi sehingga menghasilkan *output* yang sesuai dengan kebutuhan pengguna (Wibowo, 2017).

Aplikasi merupakan penggunaan dalam suatu komputer, instruksi (*instruction*) atau pernyataan (*statement*) yang disusun sedemikian rupa sehingga komputer dapat memproses *input* menjadi *output* (Irawan, 2017).

#### C. Helpdesk

*Helpdesk* adalah titik pusat pelaporan masalah dan selanjutnya dikelola atau dikordinasi. Dari sudut pandang yang lebih luas, juga dilihat sebagai bagian utama dari fungsi layanan, bertanggung jawab menjembatani sumber daya untuk menyelesaikan masalah. Pengguna *helpdesk* bisa *internal* atau *eksternal*, sehingga menjadikannya sangat penting dalam hal kelancaran jalannya organisasi dan kualitas bantuan yang ditawarkan kepada pelanggan, *Helpdesk* adalah nama yang umum digunakan untuk pusat bantuan terhadap *end-user*. Saat ini *helpdesk* makin dilihat sebagai satu kesatuan dengan fungsi layanan dan bertanggung jawab untuk menjembatani sumber daya untuk menyelesaikan masalah dan memenuhi kepuasan *user* (Wibowo, 2017).

#### D. Ticketing

Menurut (Hendi A, 2016), "*Ticketing* adalah sebuah karcis gangguan ( atau disebut juga laporan masalah) yang digunakan dalam suatu organisasi untuk melacak deteksi, pelaporan, dan resolusi dari beberapa jenis masalah. *Trouble* sistem *ticketing* berasal dari manufaktur sebuah dasar sistem pelaporan kertas. Sekarang kebanyakan berbasis *web* dan terkait dengan hubungan manajemen pelanggan seperti *call center* atau *e-business*"<sup>3</sup>.

#### E. Flowchart

"*Flowchart* adalah bagan-bagan yang mempunyai arus yang menggambarkan

langkah-langkah penyelesaian suatu masalah. *Flowchart* merupakan cara penyajian dari suatu algoritma" (Ginting, 2013).

*Flowchart* atau bagan alir adalah "Bagan (*chart*) yang menunjukkan alir (*flow*) di dalam program atau prosedur sistem secara logika. Bagan alir digunakan terutama untuk alat bantu komunikasi dan untuk dokumentasi"<sup>6</sup>.

#### F. Use Case

*Diagram Use Case* adalah gambaran grafis dari interaksi di antara unsur-unsur sistem. Hal ini digunakan dalam analisis sistem untuk mengidentifikasi, menjelaskan dan mengatur persyaratan sistem. *Diagram Use Case* ini menggambarkan proses apa saja yang dapat dilakukan oleh aktor (Ginanjari, 2016).

*Diagram Use Case* adalah *diagram dinamis* atau perilaku dalam UML. *Use Case Diagram* memodelkan fungsionalitas suatu sistem dengan menggunakan aktor dan *use case* (Wibowo, 2017).

#### G. Activity Diagram

*Diagram aktivitas* atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau *menu* yang ada pada perangkat lunak *Activity diagram* menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem"<sup>7</sup>.

"*Activity diagram* adalah teknik/cara untuk merepresentasikan *procedure logic*, *business proses*, dan *work flow*. Dalam beberapa hal *activity diagram* memainkan peran mirip *diagram alir*, tetapi perbedaan prinsip antara notasi *diagram alir* adalah *activity diagram* mendukung *behavior paralel*"<sup>8</sup>.

#### H. Sequence Diagram

*Sequence diagram* dapat menggambarkan jalannya suatu proses yang melibatkan objek dari *class* dalam aplikasi (Ilhamsyah, 2015).

“*Sequence diagram* menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek”<sup>10</sup>.

#### I. Diagram Class

*Diagram class* merupakan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain dari suatu sistem, juga memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem. *Class diagram* juga menunjukkan atribut-atribut dan operasi-operasi dari sebuah kelas dan *constraint* yang berhubungan dengan objek yang dikoneksikan, *class diagram* secara khas meliputi : Kelas (*Class*), Relasi *Assosiations*, *Generalitation* dan *Aggregation*, Atribut (*Attributes*), Operasi (*Operation/Method*) dan *Visibility*, tingkat akses objek *eksternal* kepada suatu operasi atau atribut. Hubungan antar kelas mempunyai keterangan yang disebut dengan *Multiplicity* atau *Cardinality*<sup>10</sup>.

#### J. Website

“*World Wide Web* atau (*WWW*) atau juga dikenal dengan *Web* adalah salah satu layanan yang didapat oleh pemakai komputer yang terhubung ke *internet*. *Web* ini menyediakan informasi bagi pemakai komputer yang terhubung ke *internet* dari sekedar sekumpulan informasi yang tidak berguna sama sekali menjadi informasi yang berguna, dari informasi yang gratis sampai informasi yang *komersial*. *Website* atau situs dapat diartikan sebagai kumpulan halaman-halaman yang digunakan untuk menampilkan informasi teks, gambar diam atau gerak, *animasi*, suara, dan atau gabungan dari semuanya itu baik yang bersifat *statis* maupun *dinamis* yang membentuk satu rangkaian bangunan yang saling terkait dimana masing- masing dihubungkan dengan jaringan-jaringan halaman” (Hakim & Utomo, 2015).

#### K. PHP

*PHP* pertama kali diciptakan oleh Rasmus Lerdorf pada tahun 1994. Awalnya, *php* digunakan untuk mencatat jumlah serta untuk mengetahui siapa saja pengunjung pada *homepage*-nya. Rasmus lerdorf adalah seorang pendukung *open source*. Oleh karena itu , ia mengeluarkan *Personal Home Page Tools* versi 1.0 secara gratis, kemudian menambah kemampuan *php* 1.0 dan meluncurkan *php* 2.0. Pada tahun 1996, *php* telah banyak digunakan dalam *website* di dunia. Sebuah kelompok pengembang *software* yang terdiri dari Rasmus, Zeew Zuraski, Andi Gutman, Stig Bakken, Shane Careveo. Salah satu kelebihan *php* adalah kemudahannya untuk berinteraksi dengan *database*. *php* dapat mendukung beberapa *database* secara langsung tanpa harus *menginstal* konektor seperti halnya bahasa pemrograman *java*. Dengan demikian *php* sangat *flexible* berhubungan dengan *php* (Manopo, 2016).

#### L. MySql

*MySql* merupakan *software* yang yang tergolong *database server* dan bersifat *Open Source*. *Open Source* menyatakan bahwa *software* ini dilengkapi dengan *source code* langsung dalam sistem operasi, dan bisa diperoleh dengan cara mengunduh di *internet* secara gratis. Hal menarik lainnya adalah *MySql* juga bisa bersifat *flatfrom*, *mysql* dapat dijalankan pada beberapa sistem operasi (Amin, 2016).

### III. METODE PENELITIAN

#### A. Pengumpulan Data

##### 1. Data Primer

Peneliti mengumpulkan data primer sebagai berikut :

##### a. Wawancara

Wawancara dilakukan dengan bagian *IT* yaitu Bapak Muhammad Ridwan untuk memperoleh data-data dan informasi yang dibutuhkan dalam perancangan Aplikasi *Helpdesk Ticketing*.

**b. Observasi**

Pengamatan dilakukan dengan melakukan peninjauan langsung pada Universitas Islam Syekh Yusuf terutama pada bagian IT, hal yang diamati adalah sistem *helpdesk* yang sedang berjalan.

**2. Data Sekunder**

Peneliti mengumpulkan data sekunder sebagai berikut :

Pada metode ini peneliti mempelajari buku-buku yang berhubungan dengan masalah yang dibahas, serta mencari informasi-informasi juga di jurnal yaitu mencari teori-teori yang berhubungan dengan permasalahan yang akan peneliti kembangkan.

**B. Metode Prototype**

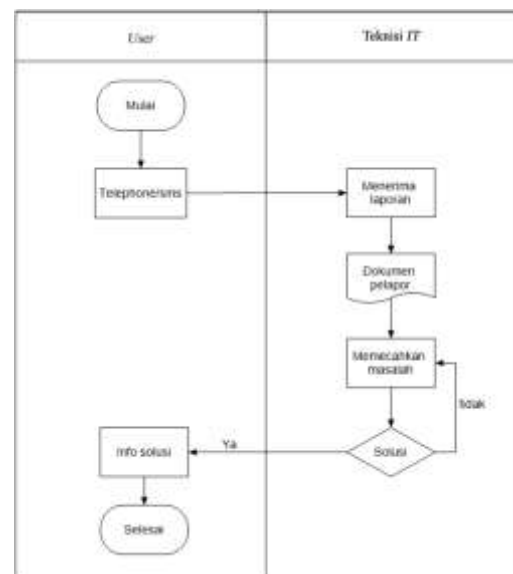
Penelitian ini menggunakan metode *prototype* untuk pengembangan aplikasi *helpdesk ticketing*. *Prototype* didefinisikan sebagai satu versi dari sebuah sistem potensial yang memberikan ide bagi para pengembang dan calon pengguna, bagaimana sistem akan berfungsi dalam bentuk yang telah selesai. Dasar dari pemikiran ini adalah membuat prototipe secepat mungkin, bahkan dalam waktu semalam, lalu memperoleh umpan balik dari pengguna yang akan memungkinkan *prototype* tersebut diperbaiki kembali dengan sangat cepat. Semua rancangan *diagram* atau model yang dibuat tidak diharuskan telah sempurna dan *final* dalam pendekatan *prototype*. Tujuan utama dari penyiapan rancangan adalah sebagai alat bantu dalam memberi gambaran sistem seperti materi dan *menu* yang perlu dimasukkan dalam *prototype* yang akan dikembangkan. Setelah rancangan terbentuk, dilanjutkan dengan mulai mengembangkan *prototype* <sup>8</sup>, tahap tahapan pada metode *prototype* sebagai berikut:



Gambar 1 Model *Prototype*

**IV. PERANCANGAN SISTEM**

**A. Sistem Yang Sedang Berjalan Di Obyek Penelitian**

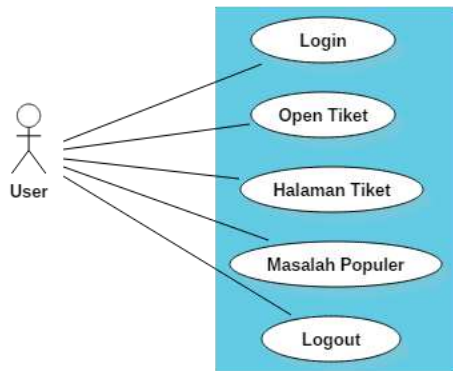


Gambar 2 *Flowchart* Sistem Yang Sedang Berjalan

Keterangan dari *flowchart* adalah sebagai berikut :

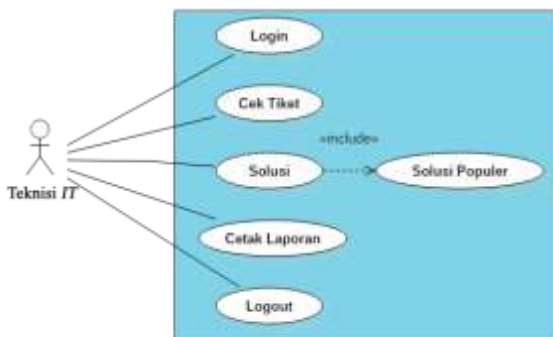
1. *User* melaporkan masalah kepada teknisi IT melalui *sms* atau *telephone*.
2. Teknisi IT menerima laporan kemudian.
3. Mendokumentasikan setelah itu.
4. Teknisi IT akan memecahkan masalah *user*.
5. Jika masalah dapat dipecahkan maka selanjutnya solusi akan diberikan kepada *user* jika tidak maka teknisi IT akan mengulangi sampai masalah terpecahkan.

*B. Use Case Sistem Usulan*



Gambar 3 Use Case User Sistem Yang Diusulkan

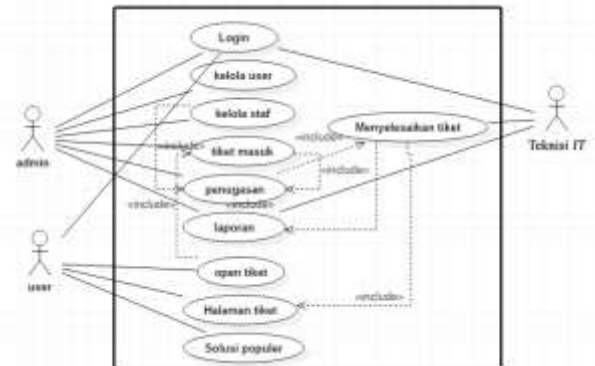
Berdasarkan gambar 3.3 *use case user* sistem yang diusulkan, *user* diharuskan *login* terlebih dahulu sebelum masuk ke *menu* utama *web helpdesk*, setelah *login user* dapat melihat 3 *menu* utama pada *web helpdesk* yaitu *open* tiket, halaman tiket dan masalah populer. Jika *user* ingin melaporkan masalah *user* dapat masuk pada *menu open* tiket untuk melaporkan atau bertanya masalah yang dihadapi oleh *user*. Setelah melaporkan masalah pada *menu open* tiket *user* dapat menunggu umpan balik dari teknisi *IT* pada *menu* halaman tiket. Terdapat juga *menu* masalah populer yang mana pada *menu* tersebut *user* dapat mencari solusi dari masalah yang dihadapi tanpa perlu bertanya kepada teknisi *IT*.



Gambar 4 Use Case Diagram Teknisi IT

Berdasarkan gambar 3.4 *use case user diagram* teknisi *IT* sebelum teknisi *IT* memberikan atau memecahkan masalah yang dihadapi oleh *user*, teknisi *IT* diharuskan

*login* setelah *login* teknisi *IT* akan mengecek tiket yang masuk setelah mengecek tiket masuk selanjutnya teknisi *IT* akan memecahkan masalah yang dilaporkan oleh *user*.

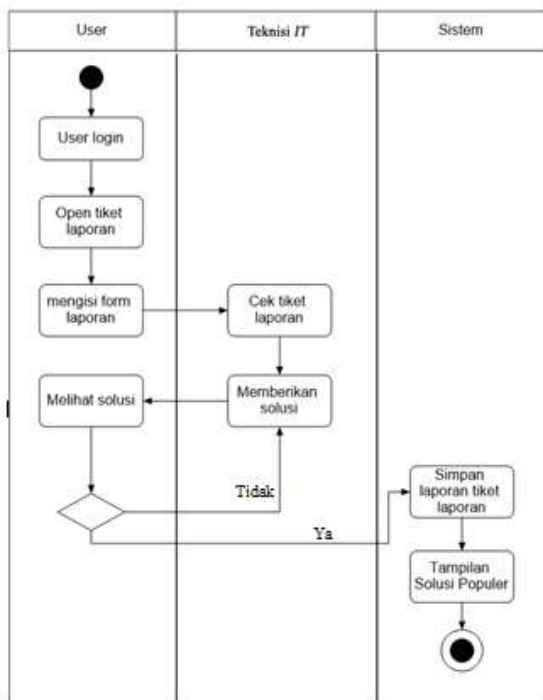


Gambar 5 Use Case Diagram Usulan

*C. Aktivity Diagram Sistem Usulan*

Berdasarkan gambar diatas *aktivity diagram* sistem yang diusulkan dimulai dari *user login* kemudian *user* membuka *menu open* tiket selanjutnya *user* mengisi *form* laporan, setelah itu teknisi *IT* akan mengecek tiket yang masuk dan memberikan solusi dari permasalahan tersebut, jika solusi yang diberikan teknisi *IT* memecahkan masalah *user* maka selanjutnya tiket tersebut akan disimpan, jika solusi tersebut tidak memecahkan masalah *user* maka teknisi *IT* akan kembali memberikan solusi lainnya sampai permasalahan selesai. Setelah tiket tersimpan pada *database* maka selanjutnya *website* akan menampilkan pertanyaan dan jawaban terpopuler atau terbanyak pada *menu* solusi populer halaman utama *website helpdesk*, sehingga *user* dapat menemukan solusi permasalahan tanpa perlu membuka tiket laporan.





Gambar 6 Aktivity Diagram Sistem Usulan

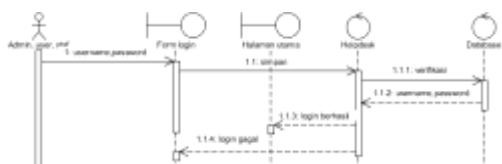
**D. Diagram Class**

*Class Diagram* untuk menggambarkan visualisasi dari struktur objek sistem yang diajukan.



Gambar 7 Class Diagram Aplikasi Helpdesk Ticketing

**E. Sequence Diagram**



Gambar 8 Sequence Diagram Login

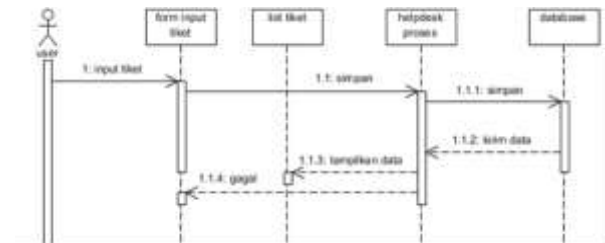
Kegiatan *login* dimulai dengan mengisi *username* dan *password* kemudian *helpdesk*

akan memverifikasi apakah *username* dan *password* sesuai dengan *database*, jika sesuai maka *login* berhasil dan menampilkan halaman utama aplikasi *helpdesk*, jika gagal maka *helpdesk* akan meminta mengisi ulang *username* dan *password*.



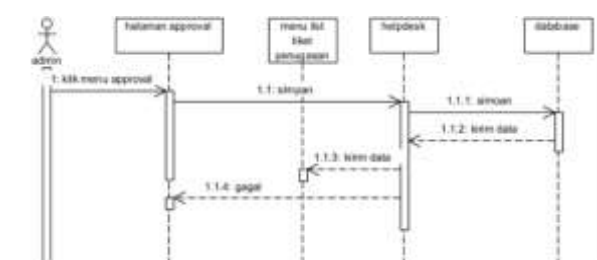
Gambar 9 Analisa Sequence Diagram Kelola Mahasiswa

Di gambar diatas tersebut *admin* menginput data mahasiswa kemudian *helpdesk* akan menyimpan data mahasiswa pada *database*, kemudian *helpdesk* akan menampilkan data mahasiswa pada *menu list* mahasiswa.



Gambar 10 Sequence Diagram Open Tiket

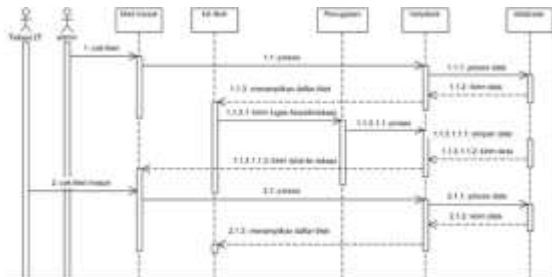
Pada gambar diatas *user* membuka tiket baru dengan *menginput* data tiket, setelah itu *helpdesk* akan menyimpan tiket pada *database*, jika berhasil maka tiket *user* akan ditampilkan *helpdesk* pada *menu list* tiket, jika gagal maka *helpdesk* akan meminta *user* mengisi ulang *form* tiket .



Gambar 11 Sequence Diagram Approval

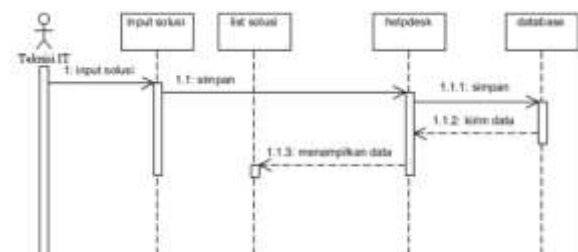


Berdasarkan sistem pada gambar diatas *admin* membuka aplikasi *helpdesk* setelah itu memilih *menu approval* pada halaman utama *helpdesk* kemudian *helpdesk* akan menampilkan *list* tiket masuk, *admin* memilih tiket yang akan di setujui kemudian akan menyimpan data pada *database* dan ditampilkan pada *menu list* tiket .



Gambar 12 Sequence Diagram Penugasan

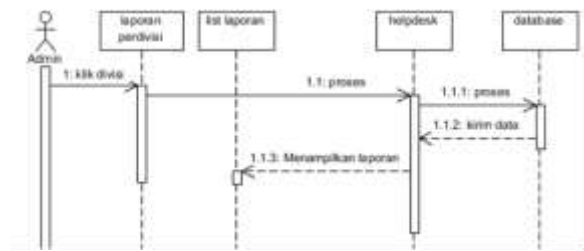
Pada gambar diatas *admin* melihat tiket masuk kemudian memilih tiket untuk disetujui, selanjutnya *helpdesk* akan memproses tiket dan memverifikasi tiket pada *database*, kemudian *helpdesk* akan menampilkan tiket yang disetujui pada *menu list* tiket. Selanjutnya *admin* menugaskan tiket kepada teknisi untuk diselesaikan, tiket diproses dan disimpan *helpdesk* pada *database*. Setelah tersimpan *helpdesk* akan mengirim tiket pada kotak masuk tiket teknisi, kemudian teknisi menyetujui tiket masuk, *helpdesk* memproses dan menyimpan pada *database* dan menampilkan tiket yang disetujui teknisi pada *menu list* tiket teknisi.



Gambar 13 Sequence Diagram Solusi

Pada gambar diatas teknisi *menginput* solusi kemudian *helpdesk* akan menyimpan solusi pada *database*, setelah tersimpan maka

*helpdesk* akan menampilkan solusi pada *menu list* solusi..



Gambar 14 Sequence Diagram Laporan

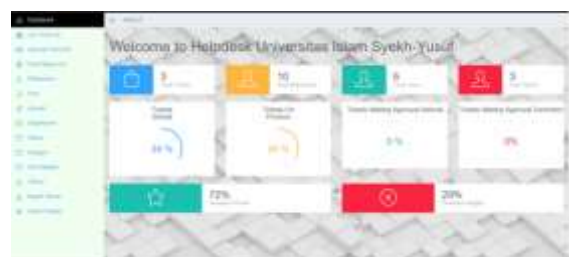
Pada gambar diatas *admin* membuka laporan kemudian *helpdesk* akan memproses laporan pada *database*, selanjutnya laporan ditampilkan *helpdesk* pada *menu list* laporan .

*F. Halaman Antar Muka Pengguna*



Gambar 15 Halaman Login

Pengguna aplikasi *helpdesk admin*, *user* dan teknisi *IT* diharuskan *login* terlebih dahulu.



Gambar 16 Halaman Dashboard Admin

Tampilan ini adalah *dashboard admin*, *admin* memiliki 13 menu pada halaman *helpdesk*.



Gambar 17 Halaman *Home User*

Pada gambar diatas adalah halaman utama *user*, dengan 3 menu tiket baru, tiket saya dan solusi populer.



Gambar 18 Halaman *Input* Tiket

Pada gambar diatas adalah halaman *form* tiket baru, *user* diharuskan mengisi *form – form* tiket sebelum disimpan.



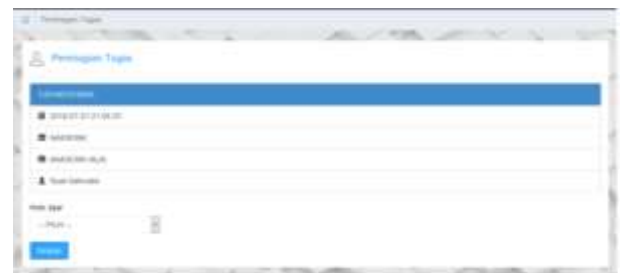
Gambar 19 Halaman *List* Tiket

Pada gambar diatas adalah halaman *list* tiket *user* yang telah disimpan dan sedang dalam proses.



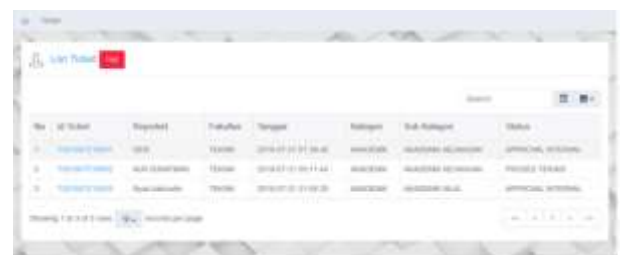
Gambar 20 Halaman *Approval*

Pada gambar diatas halaman persetujuan tiket masuk, jika *admin* setuju dengan tiket masuk maka bisa menggunakan aksi berwarna hijau, jika tidak setuju atau menolak tiket maka aksi berwarna merah.



Gambar 21 Halaman *Pembagian* Tugas

Pada gambar diatas adalah halaman pembagian tugas tiket masuk sesuai dengan kategori tiket dan spesialis teknisi.



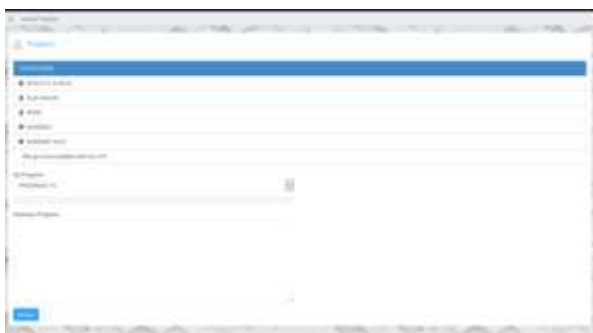
Gambar 22 Halaman *List* Tiket yang *Approval*

Pada gambar diatas adalah halaman list tiket yang telah disetujui dan tiket yang telah ditugaskan kepada teknisi.



Gambar 23 Halaman *List* Tiket Masuk Teknisi

Pada gambar diatas adalah laman *list* tiket masuk yang telah ditugaskan *admin* kepada teknisi, jika teknisi ingin memproses tiket masuk maka aksi warna hijau, aksi merah untuk status pending.



Gambar 24 Halaman *Input* Solusi

Pada gambar diatas adalah halaman teknisi untuk memecahkan masalah dan memberikan solusi kepada *user*.

#### G. Pengujian Sistem

Pada bagian ini penulis menggunakan metode pengujian yang menggunakan pendekatan *black box testing*, *black box* adalah metode yang dimanfaatkan untuk menemukan kesalahan/bug dan mendemonstrasikan fungsionalitas aplikasi saat dioperasikan, apakah masukan diproses dengan benar dan *keluaran* yang dihasilkan telah sesuai yang diharapkan. Pada *black,box* testing meliputi *security* dan *performance* sistem.

#### IV. KESIMPULAN

Berdasarkan perancangan pada Aplikasi *Helpdesk Ticketing*, maka diperoleh beberapa kesimpulan :

1. Aplikasi *Helpdesk Ticketing* dengan peran *admin* sebagai verifikasi tiket

- sesuai dengan kategori maka memudahkan teknisi *IT* dalam memberikan solusi karena tiket yang masuk sesuai dengan bidangnya
2. Aplikasi *Helpdesk Ticketing* membantu teknisi dalam memberikan solusi dari laporan-laporan *user*.
3. Aplikasi *helpdesk* bisa memudahkan *user* dalam melakukan laporan, *user* tidak perlu lagi bertemu langsung kepada teknisi untuk melakukan laporan.
4. Dengan adanya solusi populer diharapkan dapat membantu *user* mengatasi masalahnya tanpa perlu membuat tiket baru.
5. Admin dapat melihat laporan permasalahan sesuai dengan kategori masalah, dengan laporan tersebut diharapkan bisa menjadi bahan pertimbangan untuk pengembangan sistem.

#### REFERENSI

- [1] Fridayanthie EW, Mahdiati T. Rancang Bangun Sistem Informasi Permintaan Atk Berbasis Intranet (Studi Kasus: Kejaksaan Negeri Rangkasbitung). *J Inform.* 2016;13(2):126-137.
- [2] Ginting BS, Kom S, Kom M, et al. Sistem Informasi Tunggakan Premi Nasabah Menggunakan Pemrograman Visual Delphi Pada Asuransi Jiwa Bersama Bumi Putera 1912 Cabang Binjai. 2013;7(1):29-34.
- [3] Hendini A. Pemodelan UML Sistem Informasi Monitoring Penjualan dan Stok Barang. *J Khatulistiwa Inform.* 2016;IV(2):107-116.
- [4] Huda, S. (2013). Sistem Informasi Keuangan Berbasis Desktop Dengan Java Standard Edition & Mysql Di Sekolah Tinggi Teknologi Nurul Jadid Paiton Probolinggo Sekolah Tinggi Teknologi Nurul Jadid, (9011188), 26.
- [5] Ilhamsyah A, Lusiani T, Sutanto T. Rancang Bangun Sistem Informasi Monitoring Data Performansi Mitra Speedy Berbasis Web Pada Pt.Telkom Regional Timur (Persero). *J JSIKA.* 2015;1.

- [6] Irawan A, Setiyorini NK. Rancang Bangun Aplikasi helpdesk Dengan Menggunakan Pendekatan Knowledge Management System Pada Seksi Teknisi pt . Indah kiat pulp & paper tbk . *ProTekinfo*. 2017;4.
- [7] Manopo RI, Wowor H, Lumenta A. Perancangan Aplikasi Help Desk Di UPT-TIK Unsrat. *Tek Inform*. 2016;8(1).
- [8] Nurajizah S. Sistem Informasi Perpustakaan Berbasis Web Dengan Metode Prototype: Studi Kasus Sekolah Islam Gema Nurani Bekasi. *Pros SNIT 2015*. 2015:214-219.
- [9] Rico. Analisis Dan Perancangan Sistem Informasi It- Helpdesk ( Studi Kasus : Pt . Lontar Papyrus Pulp & Paper Industry ). *Teknol Inf*. 2016;10(2):296-305.
- [10] Suryono WD, Saptono R. Implementasi Pengembangan Smart Helpdesk di UPT TIK UNS Menggunakan Algoritma Naive Bayes Classifier. 2017:39-43.
- [11] Wibowo, N. I., Metandi, F., & Irwansyah. (2017). Rancang Bangun Aplikasi Helpdesk Berbasis Web Pada Jurusan Teknologi Informasi Politeknik Negeri Samarinda. *Just Ti*, 9, 10.