

APLIKASI SYSTEM MANAGEMENT TASK DAN PENILAIAN KERJA (KPI) PADA PT INTISOFT MITRA SEJAHTERA

Maria Ocktaviani Zega¹, Desi Chandrika², Ricki Siswanto³, Febrianti Supardinah⁴

¹41814120005@student.mercubuana.ac.id, ²41814120189@student.mercubuana.ac.id,

³41814120150@student.mercubuana.ac.id, ⁴febrianti.supardinah@gmail.com

^{1,2,3,4} Universitas Mercubuana Jakarta

Abstrak

Tujuan penelitian ini adalah untuk membuat sebuah aplikasi yang dapat memberikan manfaat bagi PT Intisoft Mitra Sejahtera dalam mengelola tiket yang disampaikan oleh *client*, mengatur kebijakan modifikasi program, dan dapat melihat penilaian terhadap kinerja karyawan yang mengerjakan tiket serta mempermudah *client* dalam mendapatkan informasi yang cepat mengenai kendala yang mereka alami dengan mengubahnya menjadi tiket pada aplikasi tersebut. Metode penelitian yang digunakan yaitu dengan cara melakukan analisis (survei sistem berjalan, wawancara, dan observasi) dan metode perancangan (UML), metode *testing* (*Black box testing*). Analisis dilakukan dengan cara melakukan studi lapangan dan melakukan wawancara kepada Direktur PT Intisoft Mitra Sejahtera beserta divisi yang terkait dalam rencana pembuatan aplikasi ini. Hasil yang dicapai yaitu aplikasi *web* dan *android* yang berfungsi untuk mengelola *task* dan penilaian kerja yang dapat membantu mengawasi, serta mendokumentasikan permasalahan yang ada pada perusahaan dan juga menilai kinerja karyawan dalam menangani laporan permasalahan yang telah dilaporkan oleh *client*.

Kata kunci: Manajemen tugas, UML, Key Performance Indikator

Abstract

The purpose of this study is to create an application that can provide benefits to PT Intisoft Mitra Sejahtera in managing the tickets delivered by the client, arrange program modification policies, and can see the assessment of the performance of employees who work on the ticket and facilitate the client in obtaining quick information about obstacles which they experience by turning it into a ticket on the application. The research method used is by conducting an analysis (walking system survey, interview, and observation) and design method (UML), testing method (Black box testing). The analysis is carried out by conducting field studies and conducting interviews with the Director of PT Intisoft Mitra Sejahtera along with related divisions in the plan for making this application. The results achieved are web and android applications that function to manage work assignments and assessments that can help oversee, and document problems that exist in the company and also assess employee performance in handling reports of problems reported by the client.

Keywords: Task Management, UML, Key Performance Indikator

1. Pendahuluan

Core banking bisa didefinisikan sebagai penyediaan produk dan layanan perbankan melalui *platform* elektronik seperti ATM, Internet perbankan dan seluler atau terlepas dari kecurangan (Nitsure, 2003). Karenanya pelanggan dapat mengakses akun mereka dari cabang bank mereka, terlepas dari cabang mana rekening dibuka. Saat ini transaksi dalam kehidupan sehari – hari sudah banyak yang menggunakan transaksi non tunai maka transaksi didalam bank pun sangatlah banyak sehingga dapat mengakibatkan terjadinya kesalahan dalam penginputan data atau pemberian informasi.

Sekarang, solusi *Core banking* (CBS) membantu proses *front-end* dan *back-end* otomatis dari bank mencapai pemrosesan terpusat dan halus. Aplikasi ini menawarkan satu tampilan pelanggan dan memfasilitasi otomatisasi lintas saluran pengiriman. Konsep CBS telah membantu bank menjadi toko serba ada untuk semua kebutuhan keuangan pelanggan ritel dan korporat dengan menawarkan berbagai layanan dalam satu atap[1].

Dikarenakan banyaknya *client* yang dimiliki PT Intisoft Mitra Sejahtera maka tidak jarang saat menangani permasalahan yang terjadi banyak hal yang terlewatkan sebagai contoh: saat ini sangat sulit untuk manajemen masalah yang terjadi pada *client*. Metode saat ini yang di gunakan adalah menggunakan *Whatsapp* group. Hal ini sangat menyulitkan untuk merekap *complain* mana yang sudah tertangani, yang sedang di tangani, dan yang belum tertangani. Sering terjadi *client* pun memaksa segera diselesaikan walaupun hal tersebut bukanlah sifat SLA yang harus segera di tangani



seperti permintaan mereka. Permasalahan selanjutnya yang timbul adalah untuk *maintenance source* aplikasi. Dikarenakan manajemen menggunakan *Powerbuilder* dalam pengembangan maka sulit untuk menggunakan aplikasi *source version control (SVC)* sehingga seringkali dalam perekapan dan *update* aplikasi ke *client* menimbulkan masalah-masalah akibat versi yang tertumpuk atau pun kurangnya *update* secara *database/library* aplikasi yang terkait dalam perubahan/perbaikan lainnya.

Dalam Penilaian kerja karyawan manajemen sangatlah berpengaruh, bila penilaian tersebut dianggap tidak adil atau tidak memiliki dasar penilaian maka penilaian tersebut dapat membuat kinerja karyawan menurun bahkan dapat menyebabkan keluar masuknya karyawan. Tidak jarang pula penilaian karyawan menjadi permasalahan antar karyawan yang menyebabkan tidak nyamannya lingkungan kerja.

2. Metode

2.1. Metodologi Penelitian

Metode pengembangan sistem yang digunakan yaitu metode System Development Life Cycle (SDLC). System Development Life Cycle adalah seluruh proses yang membangun, menyebarkan, menggunakan dan memperbarui sistem informasi. Dengan metode SDLC Proses pengembangan sistem menjadi lebih mudah diterapkan dan diaplikasikan sehingga pengembangan lebih mudah dijadwalkan dan dikontrol [2]. Berikut tahap penelitian yang dilakukan:

Tabel 1. Tahapan Penelitian

Kegiatan	Rincian Kegiatan
Perencanaan	Identifikasi permasalahan Penjadwalan pengerjaan
Pengumpulan Data	Wawancara Observasi Studi literatur
Analisa	Analisa sistem berjalan Analisis PIECES Analisa perangkat keras dan perangkat lunak
Perancangan	Perancangan basisdata Perancangan antarmuka
Implementasi	Pembuatan basisdata Pembuatan antarmuka Pengkodean program

1. Perencanaan dan Penjadwalan

Pada tahap ini penulis mengidentifikasi permasalahan dengan membuat latar belakang, perumusan masalah, ruang lingkup dan mencari tujuan dan manfaat pada penelitian ini. Setelah itu penulis menentukan jadwal pengerjaan penelitian ini.

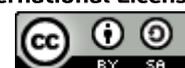
2. Pengumpulan data

Penulis melakukan wawancara dengan mengajukan pertanyaan secara lisan kepada seluruh staff yang terkait tentang *manajemen task* dan penilaian kinerja untuk mendapatkan gambaran umum perusahaan tentang sistem penanganan masalah dan penilaian kinerja pada perusahaan. Setelah itu penulis melakukan pengamatan langsung pada PT. Intisoft Mitra Sejahtera beserta client untuk mendapatkan data dan informasi yang dibutuhkan. Studi lapangan ini meliputi pengambilan data melalui dokumen tertulis maupun elektronik dari PT. Intisoft Mitra Sejahtera. Penulis juga melakukan studi literatur dengan cara mencari sumber-sumber yang berhubungan dengan manajemen task dan penilaian kinerja yang diangkat dalam penelitian. Studi literatur yang dilakukan didapat dari berbagai sumber, jurnal, buku dokumentasi, internet dan pustaka.

3. Analisa

Pada tahap analisa, penulis melakukan analisa sistem berjalan saat ini. Kemudian melakukan analisis *PIECES* untuk memperoleh pokok permasalahan yang lebih spesifik. Selanjutnya penulis menganalisa kebutuhan fungsional dari sistem yang akan dikembangkan dengan menggunakan *usecase diagram*, *activity diagram*, dan *class diagram*. Serta menganalisa kebutuhan perangkat keras dan perangkat lunak yang dibutuhkan sistem.

4. Perancangan



Pada tahap perancangan, penulis akan mulai merancang permodelan sistem dengan *UML* dan basis data sistem yang akan dikembangkan. Kemudian setelah itu, penulis akan mulai membuat racangan antarmuka pengguna yang akan menampilkan tampilan aplikasi.

5. Implementasi

Pada tahap ini, penulis mulai membuat basis data dengan bahasa *postgreSQL*. Kemudian setelah itu penulis akan mulai melakukan pengkodean program dengan *laravel* untuk *web* serta *Android Studio* untuk aplikasi androidnya. Setelah selesai diimplementasikan maka penulis melakukan *testing* pada aplikasi tersebut.

2.2. Analisis PIECES

PIECES adalah untuk mengoreksi atau memperbaiki sistem informasi bagi pengambil keputusan dalam suatu organisasi. Berikut ini daftar identifikasi masalah yang sesuai dengan yang dihadapi oleh perusahaan [3]:

Tabel 2. Analisis *Pieces*

Aspek	Kendala	Solusi
<i>Performance</i>	Pembagian tugas yang tidak merata sehingga ada <i>Staff</i> yang pekerjaannya menjadi <i>overload</i> dan proses penyelesaian jadi tidak sesuai dengan permintaan sehingga banyak mengalami proses <i>review</i> .	Dibuatkan fitur [<i>handle ticket</i>] yang membantu divisi IT dalam mengerjakan tiket yang sudah mereka handle sendiri, dan melihat ada berapa tiket yang ia handle dan sejauh mana pengerjaannya.
<i>Information</i>	client tidak bisa mengetahui siapa yang sedang mengerjakan masalah yang sudah dilaporkan dan sudah sampai mana pengerjaannya. Dan client kurang memahami kejelasan penggunaan mandays.	Adanya Dashboard client yang memperlihatkan siapa yang mengerjakan laporan permasalahan dan sudah sampai sejauh mana, serta client dapat melihat sisa mandaysnya.
<i>Economy</i>	Dengan semakin banyaknya client yang ditangani dengan biaya yang berbeda-beda setiap client, seringkali sulit dalam melakukan control biaya yang seharusnya menjadi beban client terutama dalam biaya Mandays Free setiap 1 tahun Maintenance dan tidak ada perhitungan biaya mandays yang jelas baik dari sisi client maupun manajemen.	Dengan adanya aplikasi ini perhitungan biaya mandays akan lebih jelas sesuai dengan adanya MOU maintenance di awal. Client pun bisa mengetahui sisa mandaysnya dan mereka bisa mengetahui bahwa memerlukan biaya untuk modifikasi program apabila mandaysnya habis.
<i>Control</i>	Dengan semakin banyaknya client yang ditangani dengan permasalahan yang berbeda-beda setiap client, Saat ini sangat sulit untuk memanager masalah yang terjadi pada client. Dengan menggunakan Whatsapp sangat menyulitkan untuk merekap complain mana yang sudah tertangani, yang sedang di tangani, dan yang belum tertangani.	Dibuatkan sistem manajemen task untuk melihat banyaknya tiket yang sudah di post, sudah dikerjakan, atau belum terhandle. Agar proses dokumentasi menjadi terkontrol.
<i>Efficiency</i>	Pembuatan laporan permasalahan masih menggunakan via whatsapp sehingga ada beberapa pesan yang sering kali terlewat dan tidak terbaca.	Dibuatkan fitur [<i>create tiket</i>] agar client dapat mudah melaporkan permasalahan pada aplikasi yang sudah diimplementasikan.

<i>Services</i>	Dengan sistem yang berjalan sekarang perubahan belum bisa tersusun dalam sebuah dokumen sehingga sangat sulit untuk memastikan apakah aplikasi sudah sesuai dengan permintaan client atau belum, sehingga menimbulkan banyak complain dari client, karna aplikasi belum stabil dan menimbulkan error di sisi yang lain.	Dengan adanya aplikasi ini alur pengerjaan suatu masalah menjadi sangat terarah. dan QA dapat bekerja dengan baik dan mampu memastikan aplikasi tersebut sesuai permintaan client.
-----------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

2.3. Penelitian Terkait

Penelitian terkait atau penelitian terdahulu adalah salah satu acuan bagi peneliti dalam melakukan penelitian. Penelitian terdahulu dapat menjadi referensi atau pembandingan serta pembeda bagi penelitian yang dilakukan saat ini untuk mengefisienkan dan mengefektifkan pengembangan ilmu pengetahuan.

Penelitian [4] Menjelaskan bagaimana cara mengembangkan aplikasi dengan metode pengembangan *SDLC* sehingga dapat membangun aplikasi *Mylibrary* Telkomsel yang mampu melakukan proses pengelolaan informasi dan proses pengarsipan dokumen yang dapat dilakukan dengan terkomputerisasi dan dokumen tersimpan dalam basis data secara teratur.

Penelitian terkait manajemen task dan KPI telah dilakukan oleh beberapa peneliti antara lain [5]-[10]. Penelitian oleh [5] mengusulkan aplikasi berbasis *web* mampu mencatat data *user*, proyek, *task*, waktu yang dapat digunakan sebagai alternatif dalam manajemen tugas

Penelitian oleh [6] berfokus pada *Tracking* Insinyur yang akan melakukan proses tugas pemeliharaan, tugas distribusi ke divisi yang relevan serta proses pemantauan dan hasilnya, maka akan mendapatkan laporan terkait kinerja Insinyur dan dapat diambil tindakan *tracking* akan masalah pada masa depan.

Penelitian oleh [7] memberikan gambaran bahwa dengan menerapkan metode *360 degree* dalam menentukan perpanjangan kontrak kerja pegawai dapat memberikan hasil berupa keputusan penilaian kinerja pegawai khususnya pegawai kontrak yang berupa pernyataan bahwa pegawai dapat melanjutkan kontrak kerja atau tidak dapat melanjutkan kontrak kerja berdasarkan nilai kualitatif yang diperoleh dari hasil akhir penilaian kinerja pegawai yang dinilai.

Penelitian oleh [8] menjelaskan bahwa sistem PMB Akademi akan mempermudah dalam mengkoordinasi panitia-panitia penerima mahasiswa baru dengan memaksimalkan sumber daya manusia yang ada untuk mencapai suatu tujuan tertentu. Selain itu dengan memanfaatkan sistem ini dapat mengetahui kinerja dari stakeholders dalam mencapai suatu target.

Penelitian oleh [9] mengusulkan aplikasi *web* yang membantu tim manajemen dan pihak manajemen dalam proses menjalankan kegiatan dari kerjasama dengan Dokter-Dokter Referral sehingga dapat menghasilkan *profit* yang maksimal dari pasien-pasien yang dirujuk oleh Dokter-Dokter referral. Menu-menu aplikasi seperti *Report*, *Dashboard* yang ada pada aplikasi ini sangat berfungsi untuk memantau dan memberikan informasi *Performance* dari tiap-tiap Dokter Referral dengan cepat dan mudah dianalisa oleh manajemen.

Penelitian oleh [10] menjelaskan bahwa dengan metode *SCOR* diperoleh 36 *Key Performance Indicator* yang dapat disesuaikan dengan model *SCOR* yaitu *plan*, *source*, *deliver*, *make* (process), dan *return*. Hasil pembobotan dengan menggunakan pembobotan *AHP* pada hierarki tingkat 1 yang memiliki bobot tertinggi adalah *make* (process) dengan nilai bobot 0,534. Pada hierarki tingkat 2 bobot tertinggi terdapat pada variabel *reliability* dengan total bobot 0,739. Sedangkan nilai bobot tertinggi pada hierarki tingkat 3 (*Key Performance Indicator*) adalah pada *KPI 24* Keandalan kinerja karyawan dalam mengolah menjadi produk jadi dengan total bobot 0,180. Hasil ini dapat digunakan untuk penelitian selanjutnya yakni mengukur kinerja rantai pasok dari sebuah organisasi atau perusahaan.

Penelitian diatas merupakan penelitian terkait yang dapat digunakan untuk menyelesaikan penelitian ini. Berdasarkan penelitian di atas, belum ada yang menggabungkan sistem *task management* dengan *KPI* ke dalam satu aplikasi.

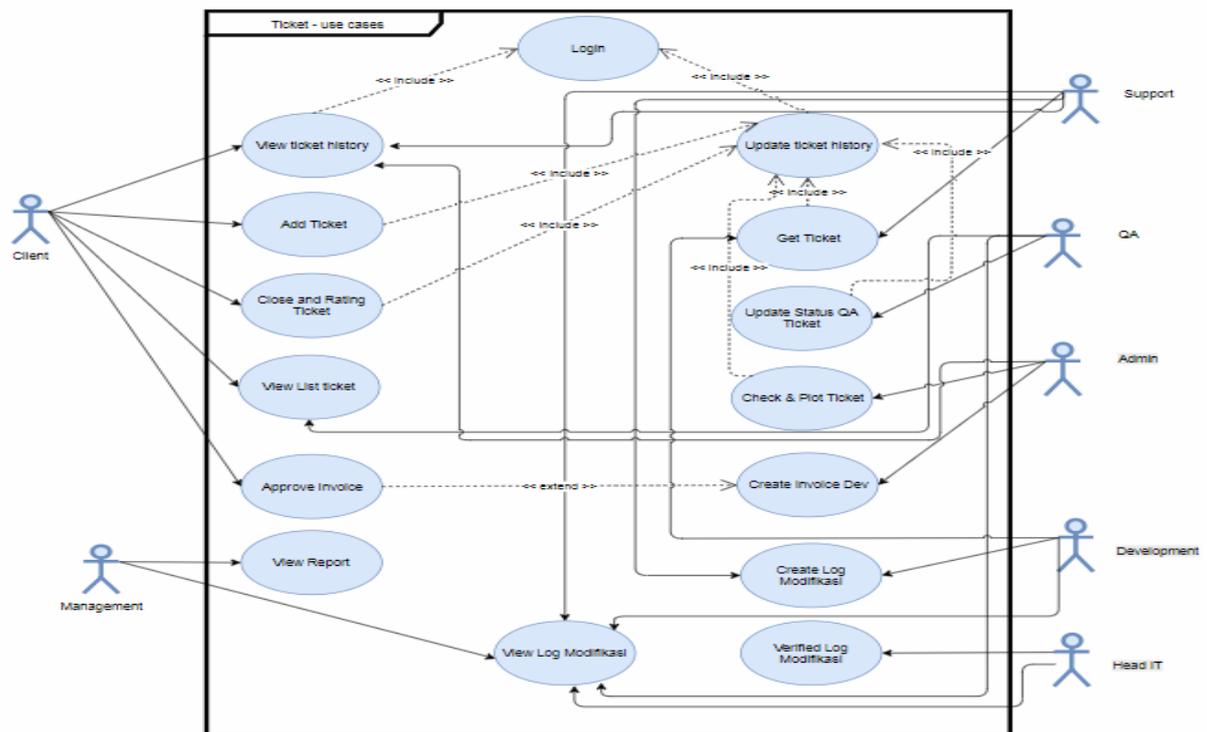
3. Hasil dan Pembahasan

Hasil pengembangan Aplikasi Sistem Management Task dan Penilaian Kerja (KPI) pada PT Intisoft Mitra Sejahtera digambarkan pada bentuk Unified Modeling Language (UML) dan User Interface (UI).

Pada usecase diagram menggambarkan adanya beberapa aktivitas user dengan aplikasi antara lain *View Ticket History*, *Add Ticket*, *Close and Rating Ticket*, *View List Ticket*, *Approve Invoice*,



View Report, Update Ticket History, Get Ticket, Update Status QA Ticket, Check dan Plot Ticket, Create Invoice Development, Create Log Modifikasi, Verified Log Modifikasi, dan View Log Modifikasi seperti yang digambarkan pada Gambar 1.



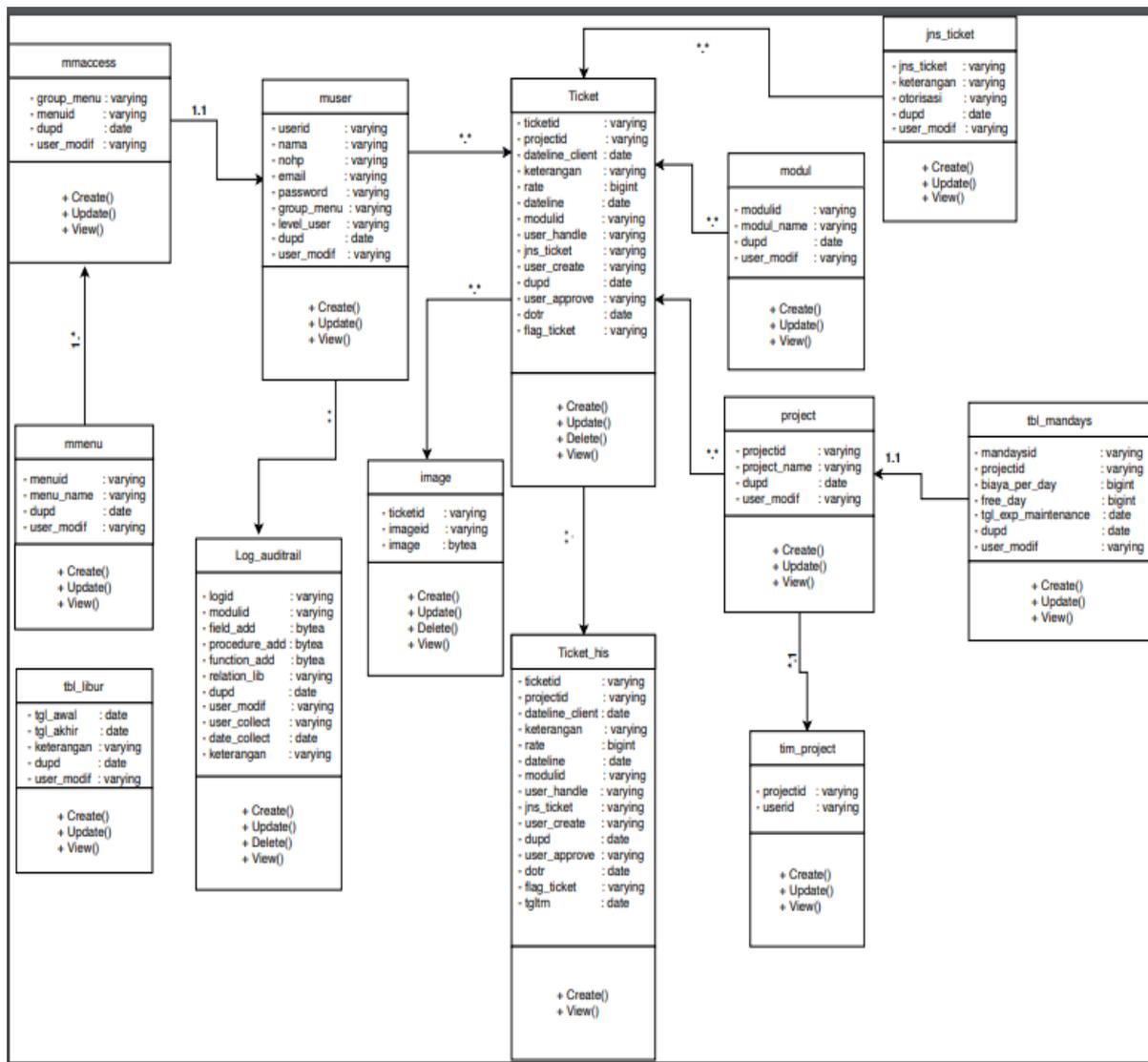
Gambar 1. Use Case Diagram

Hal pertama yang harus dilakukan untuk mengakses aplikasi ini yaitu *Login* dengan cara memasukkan username dan password. Aktivitas *View Ticket History* dilakukan untuk dapat melihat *history ticket* yang sudah dan belum dikerjakan. Aktivitas *Add Ticket* untuk dapat membuat ticket laporan permasalahan atau modifikasi aplikasi. Pada aktivitas *Close and Rating Ticket*, *Client* dapat menyelesaikan ticket dan melakukan penilaian atas ticket yang telah dikerjakan dengan memberikan rating kepada IT staff yang mengerjakan tiket tersebut. Aktivitas *View List Ticket* dilakukan oleh *Client* dan *QA* untuk melihat *ticket* mana yang sudah dan yang belum terselesaikan. Aktivitas *Approve Invoice* dilakukan oleh *client* untuk dapat menyetujui *invoice* yang sudah dibuat oleh *admin*. Pada tahap ini, *admin* harus membuat *invoice* berdasarkan sisa mandays dan jenis tiket yang dilaporkan.

Aktivitas *View Report* dilakukan oleh *Management* untuk dapat melihat *report* aplikasi baik penilaian maupun *ticketing*. Dengan *report* ini maka *management* dapat membuat perhitungan terkait penilaian karyawan berdasarkan kinerja karyawan dalam mengatasi permasalahan. Pada aktivitas *Update Ticket History*, *client* harus melakukan *add ticket* dan *development/support* mengambil *ticket* laporan permasalahan, melakukan pengerjaan sampai *client* melakukan *close* dan memberikan rating maka *history ticket* akan terupdate otomatis dan dapat melihat sampai mana permasalahan yang dilaporkan ditangani. Aktivitas *Get Ticket* dilakukan oleh *Development*, *Support* untuk mengambil tiket permasalahan yang sudah dibuat oleh *Client*. Aktivitas *Update Status QA Ticket* dilakukan oleh *QA* untuk melakukan update status QA terkait program yang sudah di testing dan akan diimplementasikan ke *Client*. Aktivitas *Check dan Plot Ticket* dilakukan oleh *Admin* untuk melakukan pengecekan dan melakukan pembagian tiket ke bagian yang dituju yaitu *Development* atau *Support* sesuai kategori tiket yang dilaporkan. Pada aktivitas *Create Invoice Development* dilakukan oleh *Admin* untuk melakukan pembuatan *invoice* terkait pengembangan. Nantinya *invoice* ini akan diterima oleh *client* untuk persetujuan pengembangan atau modifikasi aplikasi.

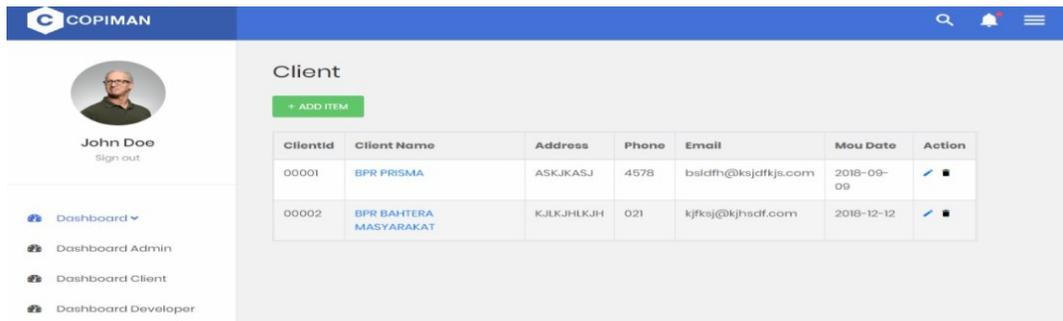
Aktivitas *Create Log Modifikasi* dilakukan oleh *Development*, *Support* untuk melakukan log modifikasi untuk melihat apa saja yang dimodifikasi atau ditambahkan dalam sebuah program. Aktivitas *Verified Log Modifikasi* dilakukan oleh *Head IT* untuk memverifikasi file update source code dan memastikan bahwa *source* program telah diterima dan siap diimplementasikan ke *Client*. Dan pada

aktivitas View Log Modifikasi Management, Support, QA, Development, Head IT dapat melihat melihat log versi modifikasi program. Berdasarkan aktivitas diatas maka diperlukan adanya class diagram untuk menggambarkan struktur data pada aplikasi yang dapat dilihat pada gambar 2.



Gambar 2. Class Diagram

Pada aplikasi web proses *ticketing* diatur oleh sistem mulai dari *add ticket* sampai klien menutup dan memberi *rating* tiket. Aplikasi ini juga dapat mengelola mandays dari menambahkan modifikasi sampai persetujuan faktur oleh klien. Dan untuk log auditrail ketika IT *push source code* aplikasi, sistem akan menampilkan Daftar Log Modifikasi yang belum diverifikasi oleh Kepala IT dan Kepala IT dapat memverifikasi data jika *source code* untuk update telah diterima. Jika sudah diverifikasi, sistem akan memperbarui status di log dan menyatakan bahwa log telah diverifikasi. IT dapat menarik laporan modifikasi log untuk menjadi panduan dalam mengimplementasikan pembaruan aplikasi berikutnya.



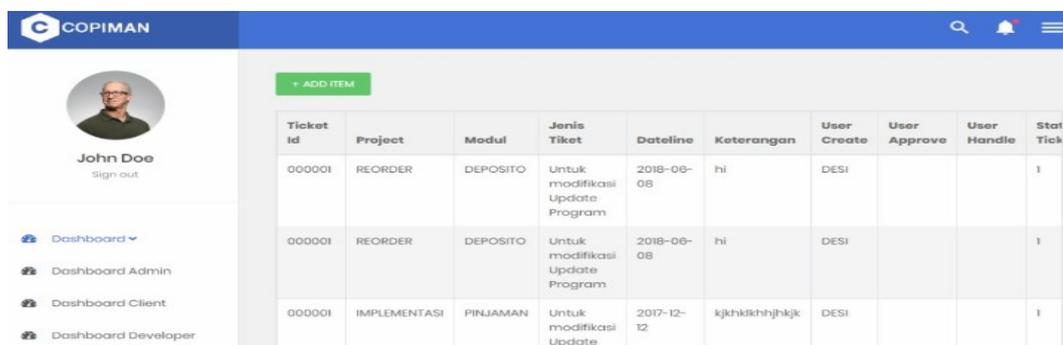
Gambar 3. Dashboard Aplikasi Client



Gambar 4. Menu Add log Auditrial

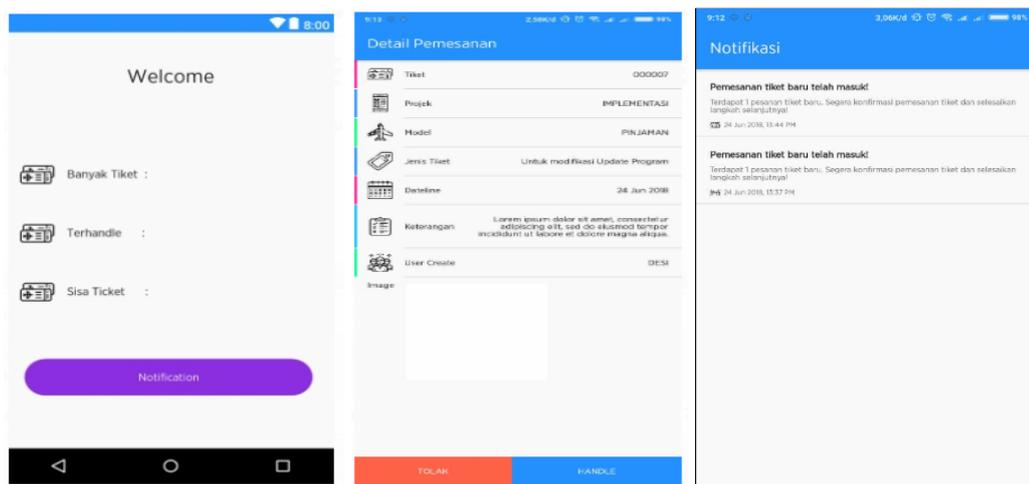


Gambar 5. Menu Add Project



Gambar 6. Daftar Detail Tiket

Aplikasi Android ditujukan bagi para *developer* untuk dapat memperoleh informasi tiket sepenuhnya. dalam aplikasi ini pengembang dapat melihat beberapa tiket masuk, pemberitahuan tiket, dan rincian tiket. Aplikasi android ini dibangun dengan tujuan proses penanganan yang cepat sesuai dengan prioritas dan jadwal yang ditentukan.



Gambar 7. Tampilan Aplikasi Android

4. Kesimpulan dan Saran

4.1 Kesimpulan

Kesimpulan dari penelitian ini adalah :

1. Aplikasi yang dibuat lebih terintegasi dibanding sebelumnya yang masih menggunakan *Whatsapp Group*. Pembuatan tiket di *client* sudah sistematis sehingga *client* dengan mudah membuat pengaduan masalah. Tiket yang masuk segera dapat ditangani oleh PT Intisoft Mitra Sejahtera di bagian yang sesuai. Dan proses penanganan masalah tersebut dapat dimonitor yang belum ditangani sampai yang sudah diselesaikan.
2. Pada aplikasi ini terdapat modul *mandays* sehingga perhitungan biaya *mandays* akan lebih jelas sesuai dengan adanya *MOU maintenance* di awal. *Client* pun bisa mengetahui sisa *mandays*nya dan mereka bisa mengetahui apakah memerlukan biaya untuk modifikasi program jika *mandays*nya habis.
3. Aplikasi ini juga menampilkan modul *List Log Modification* sehingga :
 - a. Kepentingan *Head IT* dalam memonitor kinerja masing-masing *developer* dapat dilakukan dengan baik. Modul ini mengharuskan *Head IT* menverifikasi seluruh kegiatan perubahan *script*, *coding*, dan lainnya pada masing-masing *developer*.
 - b. Saat ada kendala pada proses *monitoring* tersebut dan *developer* sebelumnya tidak ada, maka pencatatan Log Modifikasi ini dapat menjadi acuan *developer* lain yang menggantikannya.
4. Adanya *Mobile Application* yang dibangun untuk para *developer* menghasilkan informasi yang cepat terhadap tiket yang masuk, sehingga proses penanganan bisa cepat sesuai dengan prioritas dan *timeline* yang ditentukan.
5. Dengan proses yang lebih terstruktur maka pembuatan *report* dan *dashboard* dapat dibuat dari segala dimensi, penilaian kerja akan menjadi lebih objektif karna manajemen dapat melihat detail sebagai bahan perhitungan penilaian kinerja karyawan.

4.2 Saran

Adapun saran yang perlu ditambahkan untuk pengembangan aplikasi selanjutnya, yaitu :

1. Desain tampilan perlu ditingkatkan agar lebih menarik.
2. Menambahkan beberapa fitur aplikasi WEB pada aplikasi android agar kinerja para *developer* menjadi lebih efisien.
3. Menambahkan perhitungan KPI ke dalam sistem agar sistem penilaian kerja menjadi lebih efektif dan efisien.

Daftar Pustaka

- [1] Giddalah, Rapallo, "Customer Preference Attributes In The Usage Of Core Banking Service: An Emperical Investigation." International Journal of Social Sciences and Management., Vol. 1, pp. 108-112, 2014.
- [2] Satzinger, W, J., Jackson, B, R., dan Burd, D, S. "SYSTEMS ANALYSIS AND DESIGN IN A CHANGING WORLD". Boston: Cengage Learning, 2012.

- [3] W, James. "Systems Analysis and Design: Traditional, Best Practices 4th Ed." Texas: Texas Tech University, 2012.
- [4] R. Sahara , H. Prastiawan dan D. Rizal, "Rancang Bangun Sistem Informasi Mylibrary Telkomsel Berbasis Website (Studi Kasus: PT. Telekomunikasi Selular)" Jurnal Format., vol 6 , No. 2, pp. 106-118, 2016.
- [5] W. Arsal, T. Brotoharsono, dan F. Hepanie, "Aplikasi Task Manager Berbasis WEB", jurnal Fakultas Ilmu Terapan Universitas Telkom, 2011.
diakses dari: <https://openlibrary.telkomuniversity.ac.id/pustaka/files/97700/resume/aplikasi-task-manager-berbasis-web.pdf> 13 November 2018.
- [6] A. A. Puteri , Tursina ,dan A. S. Sukamto, "Sistem Pendukung Keputusan Penilaian Kinerja Pegawai dengan Metode 360 Degree," Jurnal Teknik Informatika Universitas Tanjungpura. 2015.
diakses dari :
<http://download.portalgaruda.org/article.php?article=382683&val=2313&title=Sistem%20Pendukung%20Keputusan%20Penilaian%20Kinerja%20Pegawai%20dengan%20Metode%20360%20Degree> 12 November 2018.
- [7] A. Wardhana dan I. Wiati, "Application Of Task Management and Tracking Engineer at PT. Prodata System Technology Based on WEB", International Research Journal of Computer Science (IRJCS)., vol. 4, No. 1, pp. 41-39, 2017.
- [8] M. S. Hapudin and Jani Sujatmoko, "Perancangan Program Project Management Berbasis Android pada Sistem PMB" Jurnal ICT Penelitian dan Penerapan Teknologi AKADEMI TELKOM SANDHY PUTRA Jakarta, 2017.
diakses dari : <http://ejournal.akademitelkom.ac.id/index.php/ictjurnal/article/download/64/46> 18 November 2018.
- [9] Ardiansyah dan Z. Alamsyah, "Perancangan Sistem Pengelolaan Dokter Referral PT. Nitrasanata Dharma" Jurnal Format., vol. 6, no. 1, pp. 27-38, 2017.
- [10] Ariani, M. Ulya, dan A. A. Jakfar, 2017 , "Penentuan dan Pembobotan Key Performance Indicator (KPI) Sebagai Alat Pengukuran Kinerja Rantai Pasok Produksi Keju Mozarella di CV. Brawijaya Dairy Industry" AGROINTEK., vol. 11, no 1, pp. 27-36, 2017.

