

MENDETEKSI PENGELAPAN DANA PERJALANAN DINAS DENGAN APLIKASI INTERNAL TICKETING SYSTEM

Kevin Christianto¹, Deny²

^{1,2} Program Studi Sistem Informasi, Universitas Bunda Mulia Jakarta, Indonesia
E-mail : kevin.hikoza@gmail.com, deny.shaobin@gmail.com

Submission date: 2020-05-18

Accepted date: 2020-06-09

ABSTRACT

Employee official travel activities in a company have an important role in supporting business operations and developing the business wing of the company, especially for large companies that have many branch offices in various regions in Indonesia. With more and more official trips requiring official travel tickets, a system that can internally regulate how tickets are purchased and paid is needed. The system is needed to reduce unnecessary official travel activities and valid data to be compared with data from third-party systems in terms of payment of ticket reservations. Researchers conduct literature studies, interviews, and observations. Literature study conducted by researchers to find the concept of an appropriate methodology to develop the Internal Ticketing System application. Interviews are used to examine important input on the Internal Ticketing System application of several companies that have quite intensive business travel activities, so that they are suitable and useful to be implemented later in the field, and business observations are empowered by researchers to look directly at the ticket booking business process on official trips and conduct individual understanding from researchers. The results of this study were successfully made a prototype of the Internal Ticket System application called Business Travel by using a V-Shaped SDLC method. The prototype making method is more detailed because it has 4 rounds, namely requirements, specifications, high level design and low level unit design.

Keywords: Ticket, SDLC, V-Shaped, Prototype

ABSTRAK

Kegiatan perjalanan dinas karyawan dalam suatu perusahaan memiliki peranan penting dalam mendukung operasional bisnis dan mengembangkan sayap usaha perusahaan, terutama bagi perusahaan-perusahaan besar yang memiliki banyak kantor cabang di berbagai pelosok daerah di Indonesia. Dengan makin banyaknya perjalanan dinas yang membutuhkan tiket perjalanan dinas, semakin dibutuhkan juga sistem yang dapat mengatur secara internal bagaimana tiket tersebut dibeli dan dibayar. Sistem tersebut dibutuhkan untuk mengurangi kegiatan perjalanan dinas yang tidak dibutuhkan dan data yang valid untuk dibandingkan dengan data dari sistem pihak ketiga dalam hal pembayaran pemesanan tiket. Peneliti melakukan studi pustaka, wawancara, dan observasi. Studi pustaka dilakukan peneliti untuk menemukan konsep metodologi yang sesuai untuk melakukan pengembangan aplikasi *Internal Ticketing System*. Wawancara digunakan untuk meneliti masukan-masukan penting terhadap aplikasi *Internal Ticketing System* dari beberapa perusahaan yang memiliki kegiatan perjalanan dinas yang cukup intensif, agar sesuai dan berguna untuk nantinya dapat diimplementasikan pada lapangan, serta observasi bisnis diberdayakan oleh peneliti untuk melihat secara langsung proses bisnis pemesanan tiket dalam perjalanan dinas dan melakukan pemahaman individu dari peneliti. Hasil dari penelitian ini adalah menghasilkan dibuat prototipe dari aplikasi *Internal Ticketing System* yang disebut dengan *Business Trip* dengan menggunakan metodologi SDLC *V-Shaped*. Dengan metode ini pembuatan prototipe jadi lebih detail karena memiliki 4 tahapan yaitu *requirement*, *specification*, *high-level design* dan *unit design low level*.

Kata Kunci: Tiket, SDLC, V-Shaped, Prototipe



PENDAHULUAN

Software Development Life Cycle (SDLC) sebuah strategi dalam membuat atau merawat sebuah sistem aplikasi, dan *Waterfall*, *Vee-Methodology*, dan *Rational Unified Process* (RUP) adalah jenis-jenis SDLC pada tradisional umumnya (Suryantara & Andry, 2018).

Kegiatan perjalanan dinas karyawan dalam suatu perusahaan memiliki peranan penting dalam mendukung operasional bisnis dalam mengembangkan sayap usaha perusahaan. Tingginya intensitas kegiatan perjalanan dinas tentunya sangat memerlukan koordinasi yang tepat dan cepat antar kantor cabang perusahaan, dengan adanya penugasan dari atasan kepada karyawan untuk melakukan perjalanan dinas ke kantor cabang lainnya, karyawan harus membuat pemesanan tiket ke departemen yang bertugas untuk mengurus pembelian tiket pesawat perjalanan dinas.

Oleh karena itu, fenomena tersebut haruslah dapat dikontrol oleh perusahaan, dimulai dari bagaimana mengatur secara internal proses pembelian dan pembayaran tiket perjalanan dinas itu sendiri. Bisnis proses sendiri yang mendukung perjalanan dinas ini sudah dilakukan oleh perusahaan dan berguna mencapai nilai tambahan terkait pekerjaan, tetapi dalam hal pengembangan aplikasi yang mendukung terutama masalah pembelian tiket masih belum dijangkau oleh perusahaan (Ivgantius & Andry, 2019).

Tetapi, pada saat ini banyak perusahaan yang hanya bergantung kepada sistem yang disediakan oleh para *travel agent*, sehingga kemungkinan bisa saja terjadi pengelapan dana yang bermain antara para *travel agent* dan bagian internal dari perusahaan atau apabila sistem dari *travel agent* mempunyai kesalahan, perusahaan tidak memiliki data yang akurat. Melihat hal tersebut, peneliti melihat adanya celah yang dapat diatasi oleh perusahaan dengan membuat aplikasi yang mengontrol seluruh proses pembelian tiket perjalanan dinas tersebut.

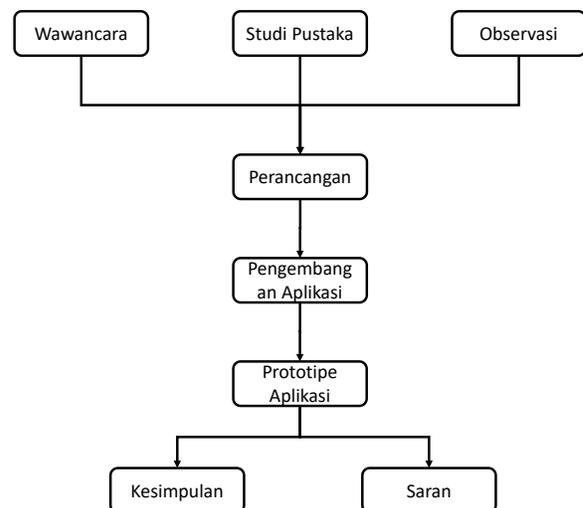
Guna mendukung keinginan peneliti untuk membuat aplikasi *Internal Ticketing System* yang disebut Aplikasi *Business Trip*, maka diperlukan suatu metode SDLC (*Systems Development Life Cycle*) yang sesuai dalam mengembangkan aplikasi tersebut. Pemilihan metodologi ini merupakan salah satu aspek penting dalam proses perjalanan manajemen proyek, karena kesalahan pemilihan metodologi dapat menyebabkan kekeliruan ataupun kesalahan proses dalam pengembangan sistem atau perangkat lunak tersebut (Kumar, N., Zadgaonkar, A. S., & Shukia, 2013).

Peneliti memilih menggunakan metode SDLC *V-Shaped* dikarenakan metode ini merupakan pengembangan dari metode SDLC *Waterfall* yang lebih difokuskan ke verifikasi dan validasi pada saat proses kebutuhan dan testing. Sehingga metode SDLC *V-Shaped* ini sangat cocok untuk digunakan sebagai metodologi pengembangan perangkat lunak yang tidak memiliki pengguna yang pasti dan kebutuhan yang spesifik dan digeneralkan yang kemungkinan dapat berubah terus-menerus atau ketidakpastian kebutuhan (Kute, S. S., & Thorat, 2014).

METODE PENELITIAN

Tahapan Penelitian

Penelitian ini melakukan prosedur langkah-langkah yang digambarkan pada diagram alir dibawah ini:



Sumber : (Kevin Christianto & Deny, 2020)

Gambar 1. Langkah - Langkah Prosedur Penelitian

Namun demikian model-model itu dapat dikelompokkan berdasarkan sifatnya yaitu statis atau dinamis (Budiyantara et al., 2020). Jenis diagram itu antara lain:

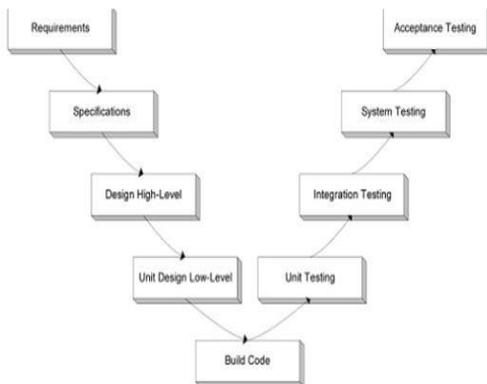
1. Tahap pertama: peneliti melakukan 3 metode pengumpulan data yang digunakan untuk menyelesaikan penelitian ini, yaitu studi pustaka, wawancara dan observasi.
 - a. Metode studi pustaka: Mengumpulkan data dari berbagai jenis buku, jurnal, artikel dan literatur lainnya yang akan digunakan sebagai landasan teori.
 - b. Metode wawancara: Melakukan tanya jawab kepada pengguna salah satu

- c. Metode Observasi: Melakukan pengamatan yang dilakukan dengan mengunjungi perusahaan terkait yang diambil peneliti sebagai objek dan melihat cara kerja langsung bagaimana proses pembelian tiket perjalanan dinas tersebut.
- 2. Tahap kedua: peneliti mulai melakukan penelitian dengan analisis data berupa perancangan dengan bantuan metode seperti *flowchart*, UML dan ERD sebagai dasar perancangan aplikasi ini.
- 3. Tahap ketiga: peneliti melakukan pengkodean yang telah dirancang pada tahap sebelumnya dengan menggunakan bahasa pemrograman PHP dan database MySQL.
- 4. Tahap keempat: peneliti melakukan proses prototipe pada objek penelitian dan menjalankan proses percobaan pada prototipe aplikasi, dan menentukan apakah aplikasi sudah sesuai dengan perancangan.
- 5. Tahap kelima: peneliti menutup penelitian ini dengan menarik kesimpulan dan saran dari hasil penelitian ini.

- 2. *Specifications*: Dalam proses ini spesifikasi secara keseluruhan dalam pembentukan infrastruktur dan *Hardware* penunjang dari kebutuhan *user* yang telah dikumpulkan pada proses *requirements*
- 3. *Design High-Level*: Desain di proses ini akan menggambarkan proses sistem secara garis besar pada luaran dari aplikasi.
- 4. *Unit Design Low-Level*: Setelah *High-Level*, lalu masuk ke bagian pendetailan desain sebelum masuk ke dalam proses *development*.
- 5. *Build Code*: Proses pengerjaan *development* dari aplikasi *Business Trip*.
- 6. *Unit Testing*: Skenario pengujian pada proses ini dibuatkan secara paralel saat *low-level* desain sedang dirancang juga dan dieksekusi setelah proses pengkodean selesai untuk beberapa *low-level* desain tersebut.
- 7. *Integration Testing*: Skenario pengujian pada proses ini dibuatkan secara paralel saat *high-level* desain sedang dirancang juga dan dieksekusi setelah proses pengkodean lebih dari 1 proses selesai untuk diintegrasikan dari proses yang satu ke proses yang lain
- 8. *System Testing*: Skenario pengujian terproses saat gambaran secara keseluruhan sistem sudah tergambar oleh peneliti dan akan di eksekusi setelah seluruh proses pengkodean selesai dan siap untuk masuk ke final proses pengujian.
- 9. *Acceptance Testing*: Skenario pengujian dibuat saat kebutuhan *user* sudah ter-*mapping* dengan baik dan dituangkan dalam skenario yang *user friendly* agar dapat di uji oleh *user* untuk melihat apakah hasil yang diharapkan sesuai dengan kebutuhan yang dijabarkan diawal.

Langkah Metode SDLC V-Shaped

Berikut langkah – langkah SDLC *V-Shaped* (Balaji, S., & Murugaiyan, 2012).



Sumber : (Kevin Christianto & Deny, 2020)

Gambar 2. Langkah - Langkah SDLC *V-Shaped Model*

Penelitian ini melakukan pengembangan aplikasi menggunakan SDLC *V-Shaped* dengan detail penelitian:

- 1. *Requirements*: Proses *requirements* di manajemen proyek ini meliputi proses wawancara dan observasi proses bisnis *Internal Ticketing System* pada beberapa tempat dan *user*.

HASIL DAN PEMBAHASAN

Requirement

Pada step awal ini, peneliti akan melakukan proses pengumpulan fakta sebagai analisa kebutuhan pada pembuatan aplikasi. Metode analisa yang digunakan oleh peneliti adalah wawancara dan observasi dengan para calon pengguna atau orang yang merasa dibutuhkannya aplikasi *Internal Ticketing System* ini ini.

Proses wawancara yang dilakukan peneliti mengambil sampel dengan wawancara langsung ke pengguna yang berhubungan langsung dengan proses perjalanan dinas dalam sebuah perusahaan. Wawancara dilakukan ke satu pengguna utama agar peneliti dapat lebih fokus untuk memecahkan dari masalah yang ada. Rangkuman dari wawancara

pengambilan analisa kebutuhan dapat dilihat dibawah ini:

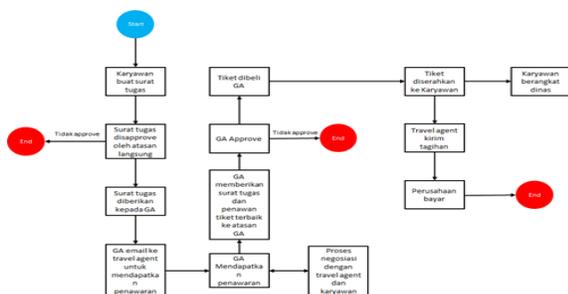
1. Analisa sistem berjalan

Dalam sistem yang berjalan, seluruh proses perjalanan dinas dimulai dari pengguna membuat surat tugas yang nanti akan di setujui oleh atasan langsung pengguna. Dari awal tersebut, nanti akan dilanjutkan dalam bagian GA, di GA-lah seluruh proses terjadi, mulai dari pemesanan tiket (yang dilakukan ke pihak eksternal yaitu agen perjalanan), proses negosiasi, persetujuan atasan GA hingga proses pencairan tiket pesawat. Setelah tiket pesawat cair dan pengguna melakukan perjalanan dinas. Agen perjalanan biasanya mengirimkan excel hasil tiket-tiket yang sudah dipesan oleh perusahaan tiap akhir bulan, dan perusahaan melakukan pembayaran berdasarkan excel yang disediakan oleh agen perjalanan itu.

2. Analisa harapan pengguna terhadap sistem yang akan datang

Pada sistem akan datang, diharapkan semua proses dapat dicakup kedalam sistem, agar lebih terstruktur dan tertata rapih. Sehingga data yang dihasilkan oleh sistem bisa menjadi bahan perbandingan terhadap nilai yang ditagihkan ke perusahaan oleh agen perjalanan tiap bulannya. Tetapi selain itu, pengguna tetap berharap masih ada proses yang manual tapi tetap dicatat dalam aplikasi seperti proses negosiasi atau proses perubahan jadwal tiket.

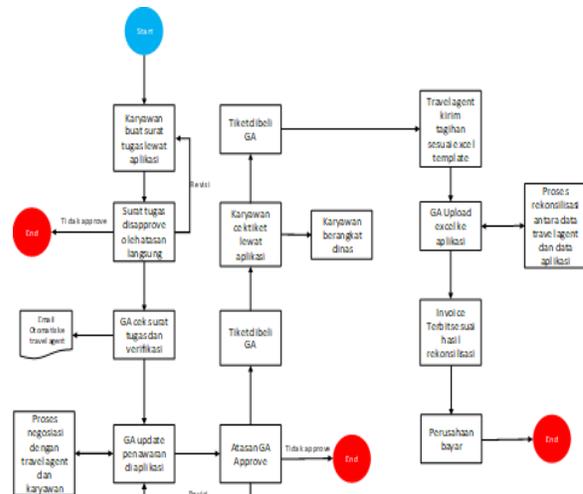
Dari hasil analisa diatas, dapat ditarik kesimpulan bahwa dibutuhkan sesuai yang terstruktur dan tercatat rapih untuk melakukan proses rekonsiliasi data dengan pihak eksternal. Sehingga ada 2 manfaat utama yaitu membuat prosedur perjalan dinas menjadi lebih baik dan memberikan proses rekonsiliasi dengan pihak external untuk proses pembayaran yang lebih terjamin untuk perusahaan. Untuk visualisasi dari analisa diatas, peneliti membuatkan proses *flowchart* untuk masing-masing dari proses analisa diatas, yaitu:



Sumber : (Kevin Christianto & Deny, 2020)

Gambar 3. Analisa sistem berjalan

Dari hasil visualisasi diatas, peneliti melihat terdapat beberapa proses yang bisa dibuat lebih baik atau dibantu oleh sistem, seperti proses pengiriman email ke agen perjalanan bisa dilakukan secara sistem, pengumpulan tiket yang sudah dibeli, dan proses rekonsiliasi yang diperlukan untuk proses pembayaran. Oleh sebab itu, peneliti dapat membuat proses akan datang dari sistem tersebut dalam menggunakan sistem menjadi:



Sumber : (Kevin Christianto & Deny, 2020)

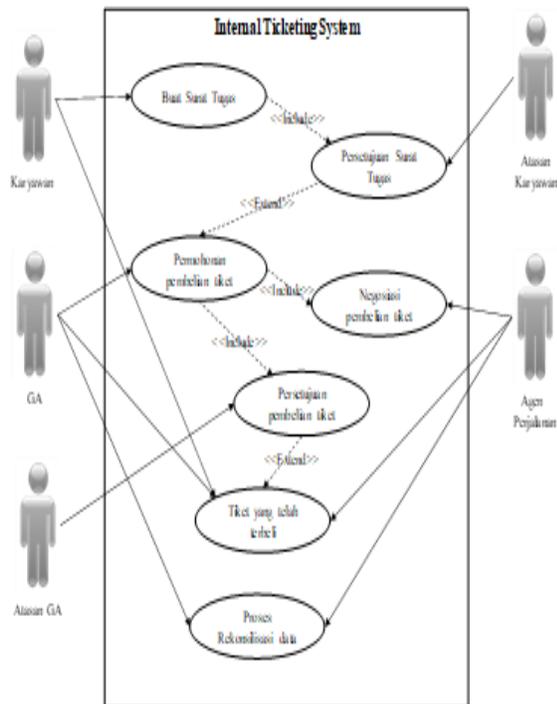
Gambar 4. Analisa sistem akan datang

Flowchart diatas menggambarkan proses perjalan dinas yang telah menggunakan sistem yang dibuat oleh peneliti. Dengan adanya sistem yang diajukan peneliti maka dapat dilihat terdapat beberapa hal yang membantu pengguna serta tambahan proses yang bisa menjaga proses penyelewengan dana pembelian tiket dengan proses rekonsiliasi dalam aplikasi.

Design High-Level

Langkah selanjutnya adalah melakukan proses perancangan secara garis besar. Hal ini agar peneliti dapat melihat secara keseluruhan dari batasan atau ruang lingkup dari sistem yang akan dikembangkan. Dalam perancangan ini peneliti akan menggunakan tools berupa UML (*Unified Modeling Language*) yaitu *use case diagram* dan *activity diagram*. Selain UML, peneliti juga menggunakan konsep perancangan ERD (*Entity Relationship Diagram*) dalam menggambarkan struktur *database* yang nantinya akan dibuat. Peneliti menggunakan kedua *tools* tersebut karena keduanya merupakan *tools* yang paling terkenal dan masih yang paling ideal digunakan untuk pengembangan sistem berorientasi terhadap objek ini.

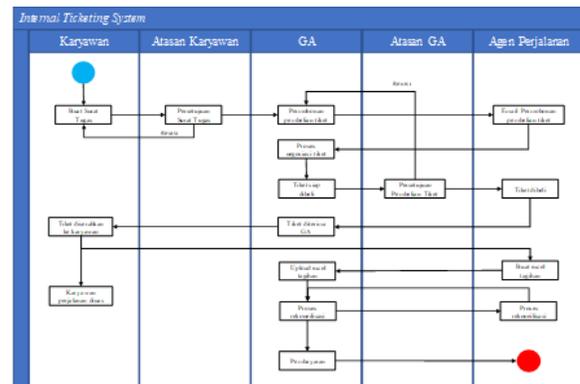
Pertama-tama peneliti akan menggambarkan *use case diagram*, karena dengan *use case diagram* ini pengguna akan memahami secara simpel apa yang nanti akan pengguna lakukan saat sistem ini diimplementasikan. Berikut perancangan dari *use case diagram* tersebut:



Sumber : (Kevin Christianto & Denny, 2020)
Gambar 5. *Use case diagram*

Dalam *use case* diatas, digambarkan nantinya dalam sistem ada 5 aktor, yaitu karyawan, atasan karyawan, GA, atasan GA dan agen perjalanan. Dimana aktor utamanya ada 3, yaitu karyawan, GA dan agen perjalanan. Dimana peran dalam aplikasi banyak perpusat di GA, tetapi awal dari proses tetap diajukan oleh karyawan. Dan dari *use case* diatas, dapat dilihat juga bahwa atasan hanya sebagai proses persetujuan saja untuk setiap inputan dari bawahannya. Dan di akhir proses terdapat proses rekonsiliasi yang menjadi fitur utama dalam hal mencegah hal penyelewengan dana saat pembayaran tiket-tiket yang telah digunakan oleh karyawan perusahaan.

Kedua peneliti juga merancang sistem ini menggunakan *activity diagram*, karena dengan *activity diagram* peneliti dapat melihat secara garis besar aktivitas-aktivitas apa yang ada dalam keseluruhan sistem yang akan dibangun. Dan berikut gambaran *activity diagram* yang dirancang oleh peneliti:



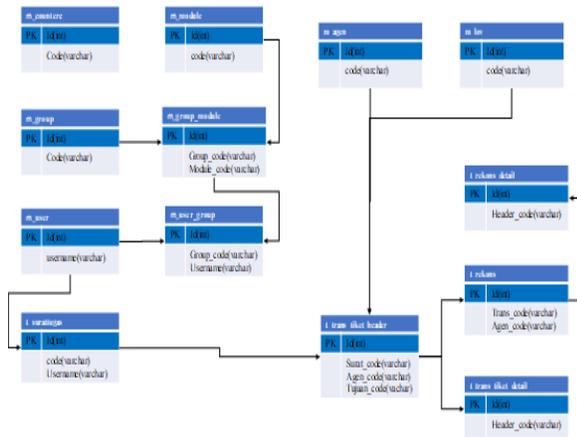
Sumber : (Kevin Christianto & Denny, 2020)
Gambar 6. *Activity diagram*

Dalam *activity diagram* diatas, dapat dibagi menjadi 3 aktivitas besar yaitu Pengajuan, proses tiket, hingga pembayaran tiket, yang penjelasannya dapat dilihat dibawah ini:

1. Langkah awal karyawan harus mengajukan surat tugas terlebih dahulu ke sistem
2. Lalu, surat tugas tersebut nantinya wajib melewati persetujuan dari atasan langsung karyawan.
3. Dan hanya surat tugas yang sudah disetujui oleh atasan langsung lah yang masuk ke dalam pekerjaan GA untuk diproses pemesanan tiketnya.
4. GA akan melihat surat tugas dan membuatkan pemesanan tiket berdasarkan surat tugas yang telah disetujui.
5. Setelah itu sistem akan secara otomatis mengirimkan email terkait tiket yang dibutuhkan oleh perusahaan ke agen perjalanan.
6. Lalu, akan ada proses negosiasi antara GA dan agen perjalanan diluar sistem untuk mendapatkan tiket yang pas untuk karyawan melakukan perjalanan dinas.
7. Setelah tiket siap untuk dibeli, maka GA akan mengajukan proses pembelian tiket ini untuk persetujuan dari atasan GA sebagai dasar untuk membeli tiket.
8. Tiket akan dibeli oleh agen perjalanan setelah atasan GA telah menyetujui permohonan yang diajukan GA.
9. Tiket yang sudah dibeli akan diberikan dari agen perjalanan ke GA, dan akan diterukan ke karyawan agar bisa melakukan perjalanan dinas.
10. Lalu, setiap akhir bulan agen perjalanan akan membuat excel tentang tiket-tiket yang sudah dibeli perusahaan untuk dibayar.
11. Excel tersebut akan diupload oleh GA ke dalam sistem dan melakukan proses rekonsiliasi data.

12. Setelah proses rekonsiliasi selesai, invoice akan keluar dan agen perjalanan akan dibayar sesuai dengan nilai akhir dari proses rekonsiliasi.

Terakhir peneliti melakukan perncangan *database* yang direpresentasikan dengan ERD sebagai fondasi awal dibuatnya sistem. Dengan ERD ini peneliti dapat memetakan entitas-entitas dalam sistem dan hubungan tiap entitas tersebut. Dan berikut gambaran *Entity Relationship Diagram* (ERD) yang dirancang oleh peneliti:



Sumber : (Kevin Christianto & Deny, 2020)
Gambar 7. ERD

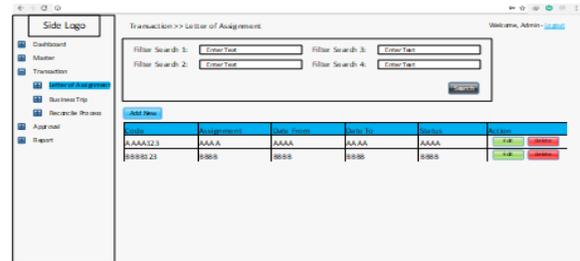
Dalam *entity relationship diagram* diatas, dapat lihat bahwa total tabel *database* yang dibuat oleh peneliti berjumlah 13 dengan beberapa bagian dari tabel tersebut, seperti tabel untuk transaksi, master dan konfigurasi untuk aplikasi..

Unit Design Low-Level

Pada langkah ini peneliti coba menjelaskan lebih detail dari desain besar yang telah dibuat dilangkah sebelumnya. Desain mengikuti halaman *login*, *master-master data* dan transaksi. Berikut desain untuk *login* dan transaksi yang ada dalam aplikasi nantinya:



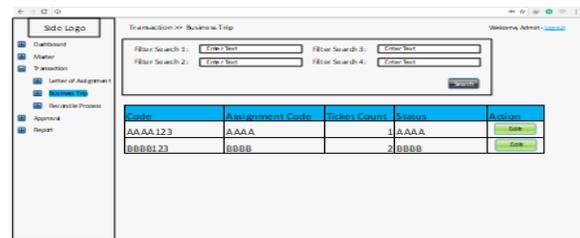
Sumber : (Kevin Christianto & Deny, 2020)
Gambar 8. Halaman Login



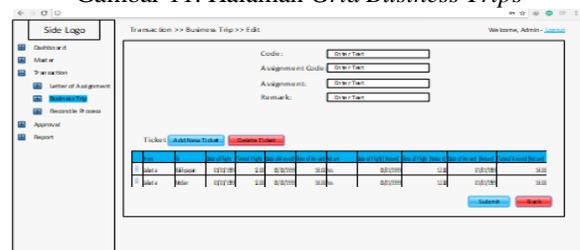
Sumber : (Kevin Christianto & Deny, 2020)
Gambar 9 Halaman Grid Letter of Assignment



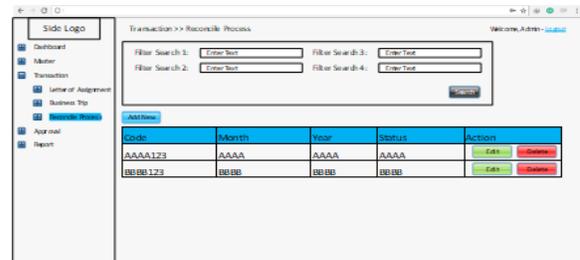
Sumber : (Kevin Christianto & Deny, 2020)
Gambar 10. Halaman Add/Edit Letter of Assignment



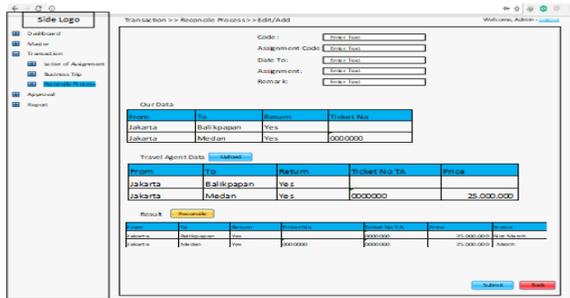
Sumber : (Kevin Christianto & Deny, 2020)
Gambar 11. Halaman Grid Business Trips



Sumber : (Kevin Christianto & Deny, 2020)
Gambar 12. Halaman Add/Edit Business Trips



Sumber : (Kevin Christianto & Deny, 2020)
Gambar 13. Halaman Grid Reconcile Process



Sumber : (Kevin Christianto & Deny, 2020)
Gambar 14. Halaman Add/Edit Reconcile Process

Acceptance Testing

Tabel 1. Bobot Alternatif Secara Keseluruhan

Nama Skenario	Keterangan	Hasil Uji
Uji Login	Tampilan halaman login dalam aplikasi	Sukses
Uji Dashboard	Tampilan halaman Dashboard dalam aplikasi	Sukses
Uji Master User - Grid	Tampilan halaman Master User - Grid dalam aplikasi	Sukses
Uji Master User - Edit	Tampilan halaman Master User - Edit dalam aplikasi	Sukses
Uji Master Group - Grid	Tampilan halaman Master Group - Grid dalam aplikasi	Sukses
Uji Master Group - Edit	Tampilan halaman Master Group - Edit dalam aplikasi	Sukses
Uji Master LoV - Grid	Tampilan halaman Master LoV - Grid dalam aplikasi	Sukses
Uji Master LoV - Edit	Tampilan halaman Master LoV - Edit dalam aplikasi	Sukses
Uji Master Travel Agent - Grid	Tampilan halaman Master Travel Agent - Grid dalam aplikasi	Sukses
Uji Master Travel Agent - Edit	Tampilan halaman Master Travel Agent - Edit dalam aplikasi	Sukses
Uji Transaction Letter of Assignment - Grid	Tampilan halaman Transaction Letter of Assignment - Grid dalam aplikasi	Sukses
Uji Transaction Letter of Assignment - Edit	Tampilan halaman Transaction Letter of Assignment - Edit dalam aplikasi	Sukses
Uji Transaction Business Trip - Grid	Tampilan halaman Transaction Business Trip - Grid dalam aplikasi	Sukses

Uji Transaction Business Trip - Edit	Tampilan halaman Transaction Business Trip - Edit dalam aplikasi	Sukses
Uji Transaction Reconcile Process - Grid	Tampilan halaman Transaction Reconcile Process - Grid dalam aplikasi	Sukses
Uji Transaction Reconcile Process - Edit	Tampilan halaman Transaction Reconcile Process - Edit dalam aplikasi	Sukses
Uji Approval Letter of Assignment - Grid	Tampilan halaman Approval Letter of Assignment - Grid dalam aplikasi	Sukses
Uji Approval Letter of Assignment - Edit	Tampilan halaman Approval Letter of Assignment - Edit dalam aplikasi	Sukses
Uji Approval Business Trip - Grid	Tampilan halaman Approval Business Trip - Grid dalam aplikasi	Sukses
Uji Approval Business Trip - Edit	Tampilan halaman Approval Business Trip - Edit dalam aplikasi	Sukses

Sumber : (Kevin Christianto & Deny, 2020)

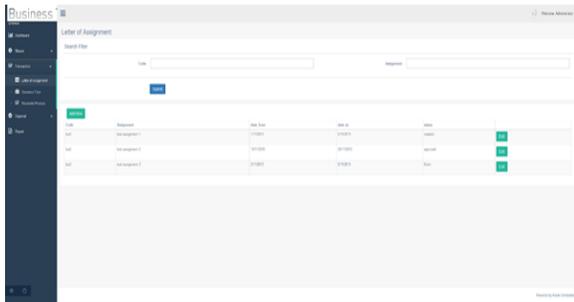
Dari tabel diatas pengguna menguji total dari 20 skenario penerimaan dari design aplikasi Business Trip. Dan dari 20 skenario pengujian yang telah dibuat peneliti, peneliti mendapatkan angka 100% pengujian sukses. Oleh karena itu, 20 skenario penerimaan deisng bisnis yang berjalan pada aplikasi Business Trip dan telah sukses terlewati yang berarti design bisnis yang telah berjalan dan digunakan oleh pengguna sudah bisa diakomodir oleh aplikasi Business Trip.

Implementation

Setelah melewati serangkaian proses metode V-Shaped, maka sistem aplikasi ini sudah bisa untuk diimplementasikan dalam bentuk prototipe sebagai tahap awal untuk pengembangan sistem secara garis besarnya. Berikut gambaran hasil prototipe dari Aplikasi Business Trip ini.

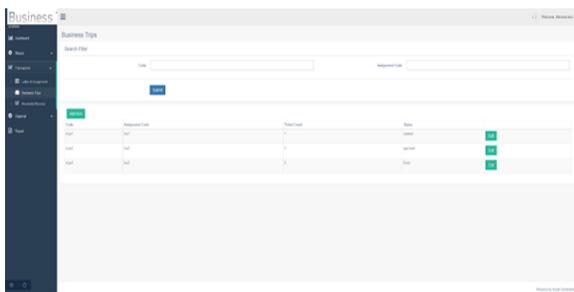


Sumber : (Kevin Christianto & Deny, 2020)
Gambar 15. Hasil Prototipe Login



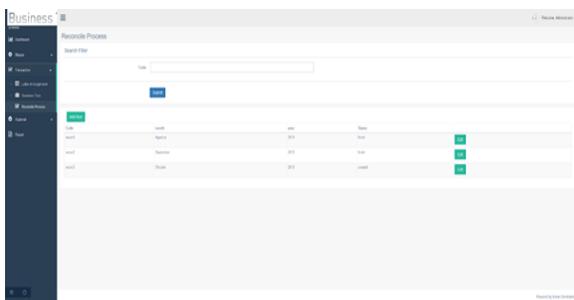
Sumber : Sumber : (Kevin Christianto & Deny, 2020)

Gambar 16. Hasil Prototipe *Transaction LoA Grid*



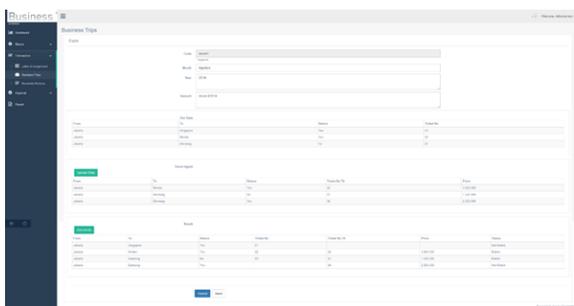
Sumber : Sumber : (Kevin Christianto & Deny, 2020)

Gambar 17. Hasil Prototipe *Transaction Business Trip Grid*



Sumber : (Kevin Christianto & Deny, 2020)

Gambar 18. Hasil Prototipe *Transaction Reconcile Grid*



Sumber : (Kevin Christianto & Deny, 2020)

Gambar 19. Hasil Prototipe *Transaction Reconcile Edit*

KESIMPULAN

Berdasarkan tujuan penelitian dan hasil penelitian di atas maka dapat disimpulkan beberapa hal berikut ini:

Dengan metode *V-shaped* peneliti dapat prototipe sistem aplikasi *Business Trip* yang support untuk proses rekonsiliasi untuk menghindari pengelapan dana. Metode *V-shaped* sendiri dalam proses penelitian dapat disimpulkan oleh peneliti bahwa metode ini lebih detail daripada metode *waterfall*, meskipun kedua metode tersebut sering disamakan, tetapi dalam pembuatan prototipe ada beberapa proses yang tidak dilalui dalam penelitian ini. Dan dalam proses pengumpulan analisa kebutuhan metode *V-shaped* lebih detail dengan pembagian proses menjadi 4 tahapan yaitu *requirement, specification, high-level design* dan *unit design low level*. Dengan tahapan tersebut peneliti berkesimpulan menghasilkan pengembangan prototipe yang meningkat dan lebih detail.

DAFTAR PUSTAKA

- Agus Budiyantra, Irwansyah, Egi Prengki, P. A. P. (2020). *Komparasi Algoritma Decision Tree, Naive Bayes Dan K-Nearest Neighbor Untuk Memprediksi Mahasiswa Lulus Tepat Waktu*.
- Balaji, S., & Murugaiyan, M. S. (2012). Waterfall V-Model Vs Agile: A Comparative Study on SDLC. *International Journal of Information Technology and Business Management*.
- Ivgantius, T. Z., & Andry, J. F. (2019). *Development of Warehouse Management System Using Extreme Programming*. 3(October), 39–46.
- Kevin Christianto, & Deny. (2020). *Mendeteksi Pengelapan Dana Perjalanan Dinas Dengan Aplikasi Internal Ticketing System*.
- Kumar, N., Zadgaonkar, A. S., & Shukia, A. (2013). Evolving a New Software Development Life Cycle Model SDLC-2013 with Client Satisfaction. *International Journal of Soft Computing and Engineering (IJSCE)*.
- Kute, S. S., & Thorat, S. D. (2014). A Review on Various Software Development Life Cycle. *International Journal of Research in Computer and Communication Technology (IJRCCT)*.
- Suryantara, I. G. N., & Andry, J. F. (2018). Development of Medical Record With Extreme Programming SDLC. *International Journal of New Media Technology*, 5(1), 47–53. <https://doi.org/10.31937/ijnmt.v5i1.706>