

PERANCANGAN SISTEM INFORMASI E-TICKETING DIVISI IT PT ASURANSI UMUM BUMIPUTERA MUDA 1967 DENGAN CODEIGNITER 3

Andri Gunawan¹; Imam Maliki²

Sistem Informasi^{1,2}
Universitas Indonesia Membangun^{1,2}

andri.gunawan9625@gmail.com¹
imam.maliki@inaba.ac.id²

Abstrak

Dalam mempercepat kinerja dan terpantaunya pekerjaan di PT Asuransi Umum Bumputera Muda 1967 terutama dalam hal permintaan terhadap Divisi IT diperlukan sebuah aplikasi yang dapat membantu memaksimalkan kinerja tersebut, permintaan kepada Divisi IT terkadang tidak termonitoring dengan baik, dan juga user kesulitan untuk dapat mengetahui status atau progres permintaan yang mereka ajukan, sehingga hal hasil mereka terus menanyakan hal itu melalui telepon, sedangkan pekerjaan yang lain pun harus Divisi IT kerjakan, ini membuat pekerjaan tidak dapat berjalan dengan efisien. Metodologi yang digunakan adalah metodologi waterfall, dimana setiap tingkatan akan diulang untuk dilakukan sebagai pengembangan. Sistem e-ticketing adalah aplikasi yang tepat untuk membantu semua pekerjaan yang dilakukan oleh perusahaan untuk mengatur jadwal pengerjaan dan melihat secara berkala hasil pekerjaan yang telah dilakukan. Website Helpdesk mempermudah untuk melakukan permintaan bantuan dari User cabang kepada Divisi IT, Mempermudah Supervisor untuk melakukan monitoring pekerjaan baik permintaan yang telah dibuat dan kapan permintaan tersebut dikerjakan, Mempermudah bagian Admin melakukan antrian pekerjaan yang harus dilakukan terlebih dahulu atau mana yang lebih prioritas untuk dikerjakan.

Kata Kunci: aplikasi *e-ticketing*, metodologi waterfall, mempermudah pekerjaan.

I. PENDAHULUAN

Dalam mempercepat kinerja dan terpantaunya pekerjaan di PT Asuransi Umum Bumiputera Muda 1967 terutama dalam hal permintaan terhadap Divisi IT diperlukan sebuah aplikasi yang dapat membantu memaksimalkan kinerja tersebut, permintaan kepada Divisi IT terkadang tidak termonitoring dengan baik, dan juga user kesulitan untuk dapat mengetahui status atau progres permintaan yang mereka ajukan, sehingga hal hasil mereka terus menanyakan hal itu melalui telepon, sedangkan pekerjaan yang lain pun

harus Divisi IT kerjakan, ini membuat pekerjaan tidak dapat berjalan dengan efisien. Permasalahan lain adalah dokumentasi yang masih terpisah-pisah sehingga kesulitan untuk mencari riwayat-riwayat permintaan tersebut mulai dari waktu respon, pengadaan permintaan, dan siapa yang merespon permintaan tersebut. Karena sistem yang berjalan saat sekarang ini masih menggunakan data dari excel.

Sehingga diharapkan adanya sistem informasi yang dapat membantu memaksimalkan pekerjaan tersebut agar user mengetahui progres dari permintaan terhadap Divisi IT dan adanya dokumentasi yang tidak terpisah untuk memudahkan pencarian data.

II. TINJAUAN LITERATUR

2.1 Landasan Teori

Landasan teori merupakan rujukan peneliti dalam melakukan penelitian

2.1.1 Pengertian Perancangan

Soetam mengatakan “Perancangan adalah sebuah proses untuk mendefinisikan sesuatu yang akan dikerjakan dengan menggunakan teknik yang bervariasi serta didalamnya melibatkan deskripsi mengenai arsitektur serta detail komponen dan juga keterbatasan yang akan dialami dalam proses pengerjaannya” (Setiyanto et al., 2019).

Sedangkan menurut Kadir (dalam Andrian, 2021) “Perancangan adalah pendefinisian kebutuhan-kebutuhan fungsional dengan mempersiapkan rancang bangun implementasi dengan menggambarkan bagaimana suatu sistem dibentuk yang berupa penggambaran, perencanaan dan pembuatan sketsa atau peraturan dari beberapa elemen yang terpisah kedalam satu kesatuan yang utuh dan berfungsi”

2.1.2 Pengertian Sistem

Tysoso menyatakan (dalam Setiyanto et al., 2019) “Sistem merupakan suatu kumpulan dari komponen - komponen yang membentuk satu kesatuan”.

Sedangkan menurut (dalam Angraini et al., 2020) “Sistem adalah suatu jaringan kerja dari prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan sasaran tertentu”.

2.1.3 Pengertian Informasi

Informasi merupakan hasil dari pengolahan data, akan tetapi tidak semua hasil dari pengolahan tersebut bisa menjadi informasi, hasil pengolahan data yang tidak memberikan makna atau arti tidak bermanfaat bagi seseorang bukanlah merupakan informasi bagi orang tersebut. (Yasir et al., 2020).

Loudon mengatakan Informasi adalah data yang telah diolah menjadi sebuah bentuk yang berguna bagi penerima dan sangat berarti dalam tindakan mengambil sebuah keputusan” (Yasir et al., 2020).

2.1.4 Pengertian Ticketing

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. (Adam et al., 2020).

Sedangkan menurut Hendi (dalam Ridwan, 2019) “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”.

Jadi bisa disimpulkan, *Ticketing* merupakan sebuah alat yang berfungsi untuk membuat antrian berdasarkan keperluan masing-masing jenis masalah, sehingga terdapat dokumentasi pada masalah tersebut.

2.1.5 Pengertian Helpdesk

Helpdesk adalah titik pusat pelaporan masalah dan selanjutnya

dikelola atau dikordinasi. Dari sudut pandang yang lebih luas, juga dilihat sebagai bagian utama dari layanan, bertanggung jawab fungsi menjembatani sumber daya untuk menyelesaikan masalah (Ridwan, 2019).

Helpdesk pada dasarnya adalah point dimana masalah atau issue dilaporkan dan diatur secara terurut dan diorganisasikan dari perspektif umum. Helpdesk juga berfungsi sebagai pelengkap dari sebuah fungsi pelayanan dan bertanggung jawab sebagai sumber dari pemecahan masalah atau issue lainnya (*helpdesk-world.com*) (Rachmawati et al., 2019).

2.1.6 CSS

Menurut Hasanah (dalam Susilo, 2018) CSS adalah singkatan dari Cascading Style-Sheet, yaitu sebuah pengembangan atas kode HTML yang sudah ada sebelumnya. Dengan CSS, bisa menentukan sebuah struktur dasar halaman web secara lebih mudah dan cepat, serta irit size.

2.1.7 PHP

PHP adalah bahasa pemrograman yang digunakan secara luas untuk penanganan pembuatan dan pengembangan sebuah web dan bias digunakan pada HTML. PHP merupakan singkatan dari “PHP : Hypertext Preprocessor”, dan merupakan bahasa yang disertakan dalam dokumen HTML, sekaligus bekerja di sisi server (server-side HTML-embedded scripting). Artinya sintaks dan perintah yang diberikan akan sepenuhnya

dijalankan di server tetapi disertakan pada halaman HTML biasa, sehingga script-nya tak tampak disisi client (Susilo, 2018).

Sedangkan menurut Kadir (Anggraini et al., 2020) PHP merupakan bahasa pemrograman yang ditunjuk untuk membuat aplikasi web. Ditinjau dari pemrosesannya, PHP tergolong berbasis *server side*. Artinya, pemrosesan dilakukan di server. Hal ini berkebalikan dengan bahasa seperti JavaScript, yang pemrosesannya dilakukan di sisi klient (client side).

2.1.8 Mysql

Kadir mengungkapkan, MySQL adalah nama database server. Database server adalah server yang berfungsi untuk menangani database.

Database adalah suatu pengorganisasian data dengan tujuan memudahkan penyimpanan dan pengaksesan data. MySQL tergolong sebagai database relasional. pada model ini, data dinyatakan dalam bentuk dua dimensi yaitu secara khusus dinamakan tabel, tabel tersusun atas baris dan kolom (Anggraini et al., 2020).

2.1.9 Database

Jubille Enterprise (dalam Ultariani & Putra, 2020) “Database adalah suatu aplikasi yang menyimpan sekumpulan data. Setiap database mempunyai API tertentu untuk membuat, mengakses, mengatur, mencari, menyalin data yang ada di dalamnya”.

Database yaitu kumpulan file- file yang berhubungan satu dengan yang lainnya, diatur sedemikian rupa sehingga dapat digunakan oleh beberapa program aplikasi database (Ultariani & Putra, 2020).

2.1.10 Relasional Database

Gat mengatakan (dalam Efendy, 2018) “Relational Database adalah mempresentasikan semua data dalam database sebagai tabel dua dimensi sederhana yang disebut relasi. Tabel-tabel itu serupa dengan file biasa, namun informasi dalam lebih dari satu file bisa dengan mudah di ekstrak dan di kombinasikan. Kadang kala tabel-tabel ini bisa di anggap sebagai file”.

2.1.11 Framework Codeigniter

CodeIgniter adalah Kerangka Pengembangan Aplikasi - toolkit - untuk orang yang membangun situs web menggunakan PHP. Tujuannya adalah untuk memungkinkan Anda mengembangkan proyek lebih cepat daripada yang Anda bisa jika Anda menulis kode dari awal, dengan menyediakan kumpulan pustaka yang kaya untuk tugas-tugas umum yang dibutuhkan, serta antarmuka sederhana dan struktur logis untuk mengakses pustaka ini (Rachman, 2018).

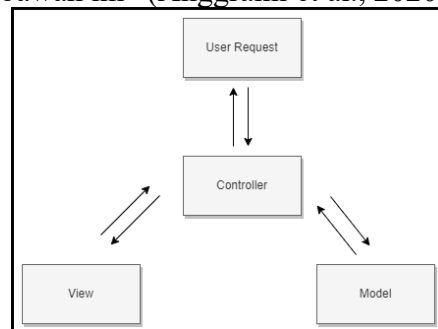
Basuki mengatakan, “framework codeigniter adalah sebuah framework PHP yang dapat membantu mempercepat developer dalam pengembangan aplikasi web berbasis PHP” (Anggraini et al., 2020).

Jadi Codeigniter 3 merupakan alat bantu berupa pustaka-pustaka dari pemrograman berbahasa PHP yang memudahkan para pembuatnya agar tidak menulis kode dari awal sehingga mempercepat proses pembuatan sebuah aplikasi berbasis WEB.

2.1.12 Model View Controller (MVC)

Basuki mengatakan “Model-

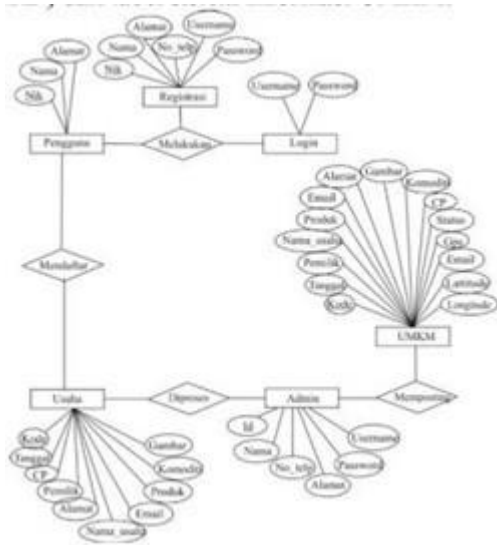
view-controller adalah framework PHP yang dibuat berdasarkan kaidah model-view-controller. Dengan MVC, maka memungkinkan pemisahan antara layer application- logic dan presentation. Sehingga, dalam sebuah pengembangan web, seorang programmer bisa berkonsentrasi pada core-sistem. Adapun alur program aplikasi berbasis framework codeigniter dapat dilihat pada gambar di bawah ini” (Anggraini et al., 2020).



Gambar 2. 1 Model View Controller

2.1.13 Entity Relationship Diagram (ERD)

Sri Mulyani (dalam Hendrawan & Nugrahanti, 2018) mengatakan “ERD merupakan tools yang digunakan untuk memodelkan struktur data dengan menggambarkan entitas dan hubungan antara entitas (relationship) secara abstrak (konseptual)” adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan requirement, membuat analisis & desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek.



Gambar 2. 2 ERD Desain

2.1.14 Unified Modeling Language (UML)

Menurut Rosa A.S dan M.Shalahudin (dalam Josi, 2017) UML (unified Modeling Language) adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan requirement, membuat analisis & desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek.

2.1.15 Use Case

Use case diagram merupakan salah satu diagram yang digunakan untuk memodelkan suatu sistem, use case diagram dapat menggambarkan sebuah interaksi antara aktor terhadap sistem (Anggraini et al., 2020).

Menurut Shalahuddin (dalam Heriyanto, 2018) mengungkapkan, "Diagram use case merupakan pemodelan untuk kelakuan (behavior) sistem informasi yang akan dibuat. Use case mendeskripsikan sebuah

interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, use case digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut." Berikut adalah simbol-simbol yang ada pada diagram use case:

Tabel 2. 1 Simbol Diagram Use Case

Simbol	Keterangan	Deskripsi
	Actor	Menspesifikasi kan himpunan peran yang pengguna mainkan ketika berinteraksi dengan use case
	Include	Hubungan antara suatu use case merupakan bagian / termasuk use case lainnya
	Use case	Deskripsi dari urutan aksi – aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi aktor
	Extend	Hubungan antara suatu use case merupakan pengembangan atau diteruskan pada use case lainnya
	Generalization	Hubungan generalisasi antar use case
	Association	Hubungan antara seseorang / suatu aktor dengan suatu use case

2.1.16 Class Diagram

Rosa (dalam Heriyanto, 2018) “Class Diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi. Atribut merupakan variabel-variabel yang dimiliki oleh suatu kelas, sedangkan operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas.” Berikut adalah simbol-simbol yang ada pada Class Diagram :

Tabel 2. 2 Simbol Class Diagram

Nama Komponen	Keterangan	Simbol
Class	<p>Class adalah blok – blok pembangun pada pemrograman berorientasi obyek.</p> <p>Sebuah Class digambarkan sebagai sebuah kotak yang terbagi atas 3 bagian, yaitu</p> <p>:Bagian atas adalah bagian nama dari Class. Bagian tengah mendefinisikan properti atau atribut Class.</p> <p>Bagian akhir mendefinisikan method – method dari sebuah Class.</p>	





Asosiasi	Merupakan sebuah <i>relationship</i> paling umum antara 2 <i>Class</i> dan dilambangkan oleh sebuah garis yang menghubungkan antara 2 <i>Class</i> . Garis disamping bisa melambangkan tipe – tipe <i>relationship</i> dan juga dapat menampilkan hukum – hukum multiplisitas pada sebuah <i>relationship</i> , antara lain : <i>One to One, One to Many, Many to Many</i>	
Composisi	Merupakan bagian dari <i>Class</i> yang lain dan memiliki relasi <i>composition</i> terhadap <i>Class</i> tempat yang bergantung tersebut dan digambarkan sebagai garis dengan ujung berbentuk jajaran genjang berisi atau solid.	
Dependensi	Digunakan untuk menunjukkan operasi pada suatu <i>Class</i> yang menggunakan <i>Class</i> yang lain dan dilambangkan sebagai sebuah panah bertitik – titik.	
Aggregasi	Mengidentifikasi keseluruhan bagian <i>relationship</i> dan biasanya disebut sebagai relasi.	


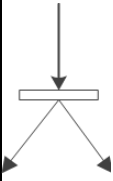


2.1.17 Activity Diagram

Rosa (dalam Heriyanto, 2018) mengatakan “Activity Diagram menggambarkan work flow (aliran kerja)

atau aktivitas dari sebuah sistem atau proses bisnis. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas dapat dilakukan oleh sistem.” Simbol-simbol yang digunakan dalam Activity Diagram sebagai berikut :

Tabel 2. 3 Simbol Sequence Diagram

Simbol	Keterangan	Deskripsi
	Start	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal
	Action	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja
	Decision	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu
	End	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir

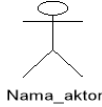
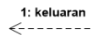



	Join (Penggabungan)	Asosiasi penggabungan dimana lebih dari satu dari aktivitas digabungkan menjadi akhir
	Fork Node	Digunakan untuk menunjuk kegiatan yang dilakukan secara parallel
	Join Node	Digunakan untuk menunjukkan kegiatan yang digabungkan
	Swimlane	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi

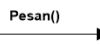
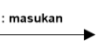

2.1.18 Sequence Diagram

Nofriadi Jurdan (dalam Heriyanto, 2018) “Sequence Diagram adalah alat yang sangat populer dalam pengembangan sistem informasi secara object-oriented untuk menampilkan interaksi antar objek.” Berdasarkan definisi tersebut, dapat disimpulkan bahwa Sequence Diagram adalah tool yang digunakan dalam pengembangan sistem.

Tabel 2. 4 Simbol Sequence Diagram

Simbol	Nama Simbol	Deskripsi

 <p>Atau</p> <div style="border: 1px solid black; padding: 2px; width: fit-content; margin: 5px auto;">Nama_aktor</div>	Aktor	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu
	Pesan Tipe Return	Metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda di awal frase nama aktor
	Garis Hidup/ lifeline	Menyatakan kehidupan suatu objek
<div style="border: 1px solid black; padding: 2px; width: fit-content; margin: 0 auto;">Nama_objek</div> 	Objek	Menyatakan objek yang berinteraksi pesan
	Waktu Aktif	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu

		aktif ini adalah sebuah tahapan yang dilakukan di dalamnya
	Pesan Tipe Call	Objek mengirim satu pesan ke objek lainnya
	Pesan Tipe Send	Menyatakan bahwa suatu objek mengirimkan data/ masukan / informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim
	Pesan Tipe Send	Menyatakan bahwa suatu objek mengirimkan data/ masukan / informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim

2.1.19 Context Diagram (CD)

CD memperlihatkan sistem yang dirancang secara keseluruhan, semua external entity harus digambarkan sedemikian rupa, sehingga terlihat data yang mengalir pada input-proses-output.

CD menggunakan tiga buah simbol yaitu: simbol untuk melambangkan external entity, symbol untuk melambangkan data flow dan simbol untuk melambangkan process. CD hanya boleh terdiri dari satu proses saja, tidak boleh lebih, dan pada CD tidak digambarkan data store. Proses pada CD biasanya tidak diberi nomor.

Diagram ini adalah diagram level tertinggi dari DFD yang menggambarkan

hubungan system dengan lingkungan luarnya (Soufitri, 2019).

2.1.20 Flowmap

Flowmap merupakan suatu diagram untuk menggambarkan aliran data/informasi antar bagian-bagian yang terkait dalam sistem. Atau Flow Map dapat juga diartikan sebagai alat bantu yang berbentuk sebuah diagram yang berfungsi untuk memetakan gambaran prosedur-prosedur yang terjadi dalam sistem.

Tabel 2. 5 Simbol Flowmap

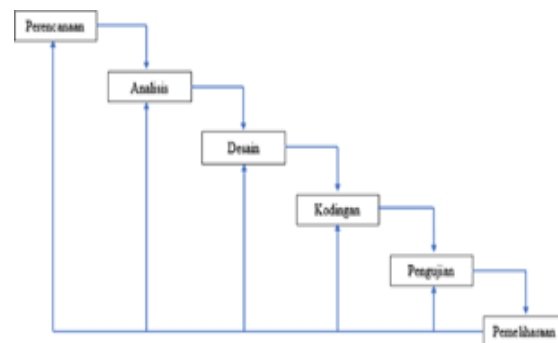
Simbol	Keterangan
	Dokumen : Menunjukkan berupa dokumen input dan ouput pada proses manual dan proses berbasis komputer.
	Proses Manual : Menunjukkan proses yang dilakukan secara manual.
	Penyimpanan Magnetik : Menunjukkan media penyimpanan data/informasi pada proses berbasis komputer.
	Arah Alir Dokumen : Menunjukkan arah aliran dokumen antar bagian terkait pada suatu sistem.
	Penghubung : Menunjukkan alir dokumen yang terpisah pada halaman alir dokumen yang sama.
	Proses Komputer : Menunjukkan proses yang dilakukan secara komputerisasi.

	Pengarsipan : Menunjukkan simpanan data pada proses manual.
	Input Keyboard : Menunjukkan input yang dimasukkan melalui keyboard.
	Penyimpanan Manual : Menunjukkan media penyimpanan data/informasi secara manual.

III. Metode Penelitian

Dalam membangun sistem ini penulis menggunakan metode waterfall dalam penyusunannya.

Berikut adalah penjelesan mengenai proses-proses pembuatan metode waterfall:



Gambar 2. 3 Metode Waterfall

a. Perencanaan

Dalam penelitian ini teknik pengumpulan data merupakan faktor penting demi keberhasilan penelitian. Hal ini berkaitan dengan bagaimana cara mengumpulkan data, siapa sumbernya, dan apa alat yang digunakan.

Perencanaan yang telah disusun oleh kami dilatar belakangi oleh hasil survey dari PT Asuransi Umum Bummiputera Muda 1967 Khusus nya Divisi TI yang menceritakan alur pekerjaan yang tidak dapat

termonitoring dengan baik dan terdokumentasi secara tidak efisien. Sehingga dapat ditarik kesimpulan untuk perencanaan pembangunan aplikasi *E-Ticketing* agar semua pekerjaan terekam pada aplikasi.

b. Analisa

Analisa ini merupakan tahap awal yang dilakukan oleh peneliti dalam mengembangkan sistem. Dalam analisa ini harus mendapatkan beberapa hal yang dianggap menunjang penelitian yang dilakukan, seperti mencari permasalahan yang ada, mengumpulkan data (data fisik, non fisik).

Analisa yang kita lakukan adalah melihat saat proses operasional berjalan, dan melihat apa yang bisa dilakukan oleh user pada saat melakukan permintaan terhadap Divisi IT. Pengumpulan data seperti jobdesk apa saja yang sering dilakukan oleh Divisi IT dan data-data seperti alamat email perusahaan.

c. Desain

Dalam hal ini kita mengambil kesimpulan yang di dapat pada bagian analisa dan perancangan yang dibuat pada langkah awal, dengan begitu desain aplikasi *E-Ticketing* dapat dibentuk tanpa adanya keraguan. UI dan UX adalah kunci utama pada desain aplikasi ini, karena agar memudahkan user dalam penggunaan aplikasi. Antar muka aplikasi dibagi menjadi 2 bagian, yaitu antar muka sebagai administrator yang dapat mengupdate proses pekerjaan, dan bagian user untuk membuat ticket permintaan dukungan pekerjaan pada Divisi IT.

d. Pengkodean

Dalam pembuatan aplikasi *E-Ticketing* peneliti membuatnya dengan bahasa pemrograman PHP dengan framework *codeigniter3* yang disisipkan *java script* maupun *css framework* yaitu *bootstrap* dalam pembangunannya agar tampilan terlihat menarik dan mudah dipahami.

e. Pengujian

Dalam tahapan ini peneliti hanya melibatkan user agar dimana ketika ada kerusakan sistem atau sistem mengalami *overload* trafik bisa langsung menghungi pihak admin dan bisa langsung di tangani admin, namun beda halnya ketika ada kebocoran informasi, yang menyebabkan informasi yang ada bocor ke pihak ketiga atau bisa di sebut heker, untuk pengujiannya kita bisa dilakukan menggunakan *sql injection*.

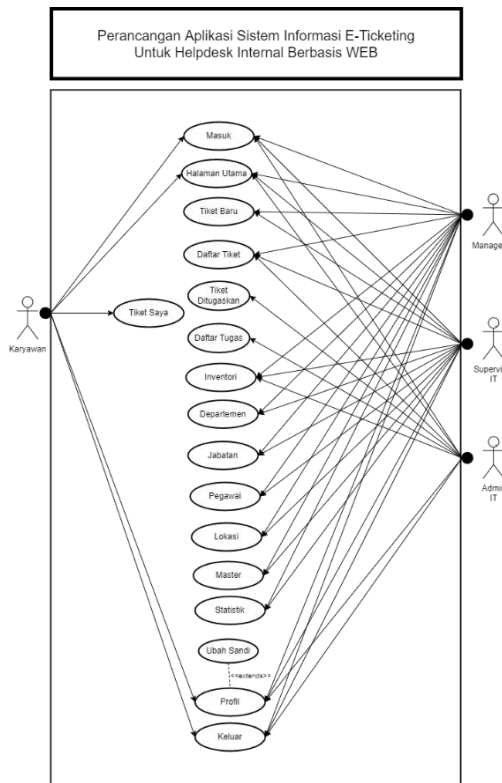
IV. HASIL DAN PEMBAHASAN

Dari penelitian diatas maka dihasilkan sebuah website dengan pearancangan sebagai berikut :

4.1 Perancangan Sistem

4.1.1 *Use Case Diagram Website*

Berikut adalah gambaran *use case diagram* pada website *e- ticketing*.



Gambar 4.1 Use Case Diagram

4.1.2 Spesifikasi Database

Tabel 4.1 Spesifikasi Database User

No	Nama Field	Type	Lebar	Keterangan
1	id_user	varchar	50	Primary Key
2	username	varchar	11	Username
3	Password	varchar	32	Password
4	level	varchar	10	Akses Level User

Tabel 4.2 Spesifikasi Database Inventori

No	Nama Field	Type	Lebar	Keterangan
1	id_brg	int	11	Primary Key
2	id_lokasi	int	11	Id Lokasi
3	no_inven	varchar	30	Nomer Inventori

4	nm_brg	varchar	30	Nama Barang
5	kategori	varchar	10	Kategori
6	type	varchar	50	Type
7	lic_windows	varchar	20	Lisensi Windows
8	lic_office	text		Lisensi Office
9	lic_software	varchar	20	Lisensi Software
10	keterangan	varchar	255	Keterangan
11	harga	varchar	12	Harga Beli
12	tgl_beli	date		Tanggal Beli

Tabel 4.3 Spesifikasi Database Tiket

No	Nama Field	Type	Lebar	Keterangan
1	id_ticket	varchar	5	Primary Key
2	tanggal	datetime	-	Tanggal
3	deadline	datetime	-	Target Akhir
4	last_update	datetime	-	Update Terakhir
5	tanggal_proses	datetime	-	Tanggal Proses
6	tanggal_selesai	datetime	-	Tanggal Selesai
7	reported	varchar	13	Laporan
8	id_sub_kategori	int	11	Id Sub Kategori
9	id_brg	Int	11	Id Inventori
10	problem_summary	varchar	50	Kendala Dialami
11	problem_detail	text		Detail Kendala
12	teknisi	varchar	50	Teknisi
13	status	int	11	Status

4.2 Tampilan Layar

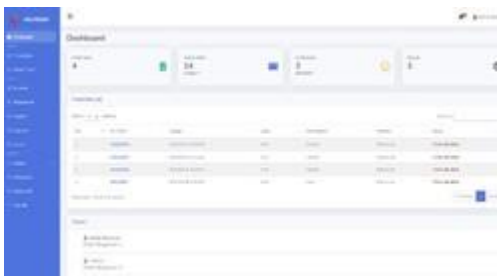
Berikut tampilan layar hasil dari pembuatan website e-ticketing

4.2.1 Tampilan Layar Halaman Login



Gambar 4. 1 Halaman Login

4.2.2 Tampilan Layar Halaman Utama



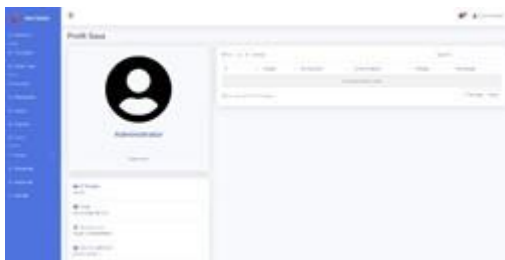
Gambar 4. 2 Tampilan Halaman Utama

4.2.3 Tampilan Layar Halaman Tiket



Gambar 4. 3 Tampilan Halaman Tiket

4.2.4 Tampilan Layar Halaman Profile



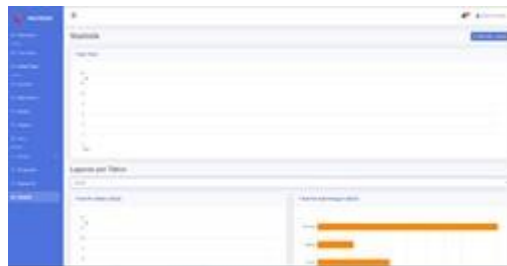
Gambar 4. 4 Tampilan Halaman Profile

4.2.5 Tampilan Layar Halaman Inventory



Gambar 4. 5 Tampilan Halaman Inventory

4.2.6 Tampilan Layar Halaman Statistik



Gambar 4. 6 Tampilan Halaman Statistik

4.3 Pengujian Dengan Black-box Testing

Abdullah mengatakan “Black Box Testing berfokus pada spesifikasi fungsional dari perangkat lunak. Tester dapat mendefinisikan kumpulan kondisi input dan melakukan pengujian pada spesifikasi fungsional program” (dalam Josi, 2017). Berikut adalah pengujian yang dilakukan menggunakan black box testing :

Tabel 4. 4 Black Box Testing

Kelas Uji	Yang diharapkan	Hasil
Login	Masuk kedalam aplikasi	Sesuai
Ubah Password	Berhasil ubah password	Sesuai
Pengelola	Berhasil menambah tiket	Sesuai

am Data Tiket	Berhasil mengedit tiket	Sesuai
Profile	Berhasi ubah profile	Sesuai
Logout	Session Aplikasi Hilang	Sesuai

V. KESIMPULAN DAN SARAN

5.1 Kesimpulan

Dalam pembuatan *website Ticketing* untuk *helpdesk* di PT Asuransi Umum Bumipuetra Muda 1967 pada Divisi IT kami memberikan kesimpulan sebagai berikut :

1. Website *Helpdesk* mempermudah untuk melakukan permintaan support dari User cabang kepada Divisi IT.
2. Mempermudah Supervisor untuk melakukan monitoring pekerjaan baik permintaan yang telah dibuat dan kapan permintaan tersebut dikerjakan.
3. Mempermudah bagian Admin melakukan antrian pekerjaan yang harus dilakukan terlebih dahulu atau mana yang lebih prioritas untuk dikerjakan.
4. Mempermudah dokumentasi terkait apa saja yang telah dikerjakan dan melakukan Analisa yang berulang menjadi lebih cepat.

5.2 Saran

Kami melihat dari beberapa keuntungan dalam penggunaan aplikasi *Helpdesk* yang digunakan, dan kami memberikan saran untuk aplikasi *Helpdesk* ini sabagai berikut :

1. Penambahan-penambahan fitur lebih banyak seperti pandataan pada setiap akun

user memliki nomer inventaris yang sesuai.

2. Monitoring koneksi internet di semua cabang menggunakan aplikasi tersebut agar tidak menggunakan tools pihak ketiga
3. Membuat aplikasi Helpdesk yang dibuat menjadi gerbang dari semua pekerjaan yang dilakukan oleh Divisi IT, agar semua dapat termonitoring dengan jelas dan tepat.
4. Membuat rating oleh Supervisor, pekerjaan yang dilakukan Admin agar Admin lebih semangat untuk bekerja.

VI. DAFTAR PUSTAKA

- Adam, S. I., Moedjahedy, J. H., & Lengkong, O. (2020). Pengembangan IT Helpdesk Ticketing Sistem Berbasis Web di Universitas Klabat. *CogITO Smart Journal*, 6(2), 217. <https://doi.org/10.31154/cogito.v6i2.273.217-228>
- Andrian, D. (2021). Penerapan Metode Waterfall Dalam Perancangan Sistem Informasi Pengawasan Proyek Berbasis Web. *Jurnal Informatika Dan Rekayasa Perangkat Lunak (JATIKA)*, 2(1), 85–93. <https://jim.teknokrat.ac.id/index.php/informatika>
- Anggraini, Y., Pasha, D., & Damayanti Setiawan, A. (2020). Sistem Informasi Penjualan Sepeda Berbasis Web Menggunakan Framework Codeigniter (Studi Kasus : Orbit Station). *Jurnal Teknologi Dan Sistem Informasi (JTSI)*, 1(2), 64–70.
- Efendy, Z. (2018). Normalisasi Dalam Desain Database. *Jurnal CorelIT*, 4(1), 34–43.

- Hendrawan, A. T., & Nugrahanti, F. (2018). Perancangan Sistem Informasi Usaha Mikro Kecil Dan Menengah (Ukm) Kabupaten Magetan Berbasis Website. 250–256.
- Heriyanto, Y. (2018). Perancangan Sistem Informasi Rental Mobil Berbasis Web Pada PT.APM Rent Car. *Jurnal Intra-Tech*, 2(2), 64–77.
- Josi, A. (2017). Penerapan Metode Prototyping Dalam Membangun Website Desa (Studi Kasus Desa Sugihan Kecamatan Rambang). *Jti*, 9(1), 50–57.
- Rachman, T. (2018). *Angewandte Chemie International Edition*, 6(11), 951–952. <https://codeigniter.com/userguide3/general/welcome.html>
- Rachmawati, U. A., Adam, S., & Alwi, S. H. (2019). Pembangunan Helpdesk Ticketing System Berbasis Web (Studi Kasus: Universitas Yarsi). *Jurnal Teknologi Informasi YARSI*, 6(1), 19–24. <https://doi.org/10.33476/jtiy.v6i1.597>
- Ridwan, M (2019). Penerapan Helpdesk Ticketing System Dalam Penanganan Keluhan Penggunaan Sistem Informasi Berbasis Web. (JUTIS) *Jurnal Teknik Informatika*, 7(1), 71–82.
- Setiyanto, R., Nurmaesah, N., & Rahayu, N. S. A. (2019). Perancangan Sistem Informasi Persediaan Barang Studi Kasus di Vahncollections Rudi. *Jurnal Sisfotek Global*, 9(1), 137–142. <https://shopee.co.id/vahncollections>.
- Soufitri, F. (2019). Perancangan Data Flow Diagram Untuk Sistem Informasi Sekolah (Studi Kasus Pada Smp Plus Terpadu). *Ready Star*, 2(1), 240–246.
- Susilo, M. (2018). Rancang Bangun Website Toko Online Menggunakan Metode Waterfall. *InfoTekJar (Jurnal Nasional Informatika Dan Teknologi Jaringan)*, 2(2), 98–105. <https://doi.org/10.30743/infotekjar.v2i2.171>
- Ultariani, N., & Putra, N. (2020). Visual Basic 2010 Dan Database Mysql. *Nursaka Putra*, 10(2), 220–229.
- Yasir, A., Kurniawan, H., Bondowoso, W. B., & Susanto, A. (2020). Sistem Informasi Perpustakaan Berbasis Web Pada Perpustakaan Universitas Dharmawangsa. In *Djtechno : Journal of Information Technology Research (Vol. 2, Issue 2)*.