

# Perancangan Aplikasi IT Help Desk Menggunakan Platform Node.js Pada Mittasys

Dwi Nurul Huda<sup>1</sup>, Aggry Saputra<sup>2</sup>, Yulinda<sup>3</sup>

<sup>1,2,3</sup>Jurusan Teknik Informatika STT Indonesia Tanjungpinang  
Jln. Pempa Air No. 28 Tanjungpinang Kepulauan Riau Indonesia

<sup>1</sup>dwi.nurulhuda@gmail.com

<sup>2</sup>aggrysaputra@gmail.com

<sup>3</sup>yulindahuangg@gmail.com

Mittasys merupakan perusahaan yang bergerak dibidang jasa pemrograman. Perusahaan ini belum memiliki sistem yang mampu menangani masalah hubungan perusahaan dengan pelanggan. Dalam layanan purna jual, perusahaan belum dapat memaksimalkan pelayanan dalam bentuk keluhan ataupun pertanyaan pelanggan. Berdasarkan permasalahan yang ada, penulis membangun sebuah perangkat lunak *help desk* dalam bentuk aplikasi *web*. Aplikasi *web* ini menggunakan *waterfall* sebagai metode pengembangannya. Perancangan aplikasi ini menggunakan *platform Node.js* dan menggunakan *database Microsoft SQL Server*. Dari hasil perancangan perangkat lunak ini diharapkan dapat membantu perusahaan dalam menangani setiap pelanggan yang membutuhkan sehingga dapat meningkatkan pelayanan serta menghasilkan keakuratan dalam pencarian data yang dibutuhkan.

*Kata kunci* : *Help Desk, Node.js, Microsoft SQL Server, Keluhan, Pertanyaan.*

*Abstract*— *Mittasys is a company engaged in programming services. The company have not had a system that is able to handle the company's relationship with customers. In after-sales service, the company has not been able to maximize service of customer complaints or questions. Based on the existing problems, the authors built a help desk software web application. This web application uses waterfall as a method of development. The design of this application uses Node.js platform and uses a Microsoft SQL Server as a database. From the results of software design, it is expected to help the company in handling every customer who needs so as to improve service and produce accuracy in finding the data needed.*

*Keywords* : *Complaints, Help Desk, Microsoft SQL Server, Node.js, Questions*

## I. PENDAHULUAN

Inovasi teknologi informasi yang berkembang dengan pesat mempunyai pengaruh besar terhadap sektor bisnis. Pada dunia bisnis yang semakin kompetitif, membuka peluang tumbuhnya berbagai usaha baru yang membuat persaingan semakin ketat. Inovasi yang tidak lagi terbatas pada teknologi tinggi, namun telah menjadi fenomena global yang mempengaruhi semua sektor kehidupan. Bagaimana sebuah informasi mampu menggerakkan perdagangan barang melalui transmisi arus informasi dan gagasan seiring dengan meningkatnya jumlah pengguna arus informasi.

Penelusuran informasi terkait kebutuhan setiap individu sangat mengandalkan mesin pencarian yang terhubung dengan internet. Setiap orang bisa menjadi selektif dalam memilih produk ataupun layanan jasa yang ditawarkan oleh sebuah perusahaan. Dalam hal ini untuk mengoptimalkan waktu, memerlukan suatu media yang berisi informasi yang bermanfaat dapat membuatnya mudah memahami produk ataupun layanan dengan cepat agar tidak membuang-buang waktu. Selain itu, juga dapat membangun citra bahwa perusahaan dapat dipercaya dengan informasi yang berhubungan dengan perusahaan dan target pelanggan.

Dewasa ini, perusahaan harus menyadari pentingnya menjaga hubungan dengan pelanggan. Untuk mewujudkan hal tersebut, perusahaan dapat memberikan layanan kepada pelanggan setelah melakukan pembelian produk/jasa atau biasa disebut *after sales services*. Menjaga kualitas *after sales services* yang baik berarti dapat membangun dan menjaga hubungan baik jangka panjang dengan pelanggan. Melalui

layanan ini, perusahaan menjadi lebih mengetahui dan mempelajari ekspektasi pelanggan.

Setiap perusahaan tentunya akan menyiapkan layanan terbaik untuk setiap pelanggannya. Namun tidak bisa dipungkiri bahwa pelayanan perusahaan tidak luput dalam ketidaksempurnaan. Dalam waktu yang singkat, pelanggan yang terlibat akan menemui ketidakpuasan pada performa layanan perusahaan. Untuk dapat mengevaluasi layanan perusahaan, setiap keluhan ataupun pertanyaan yang diajukan oleh pelanggan harus dapat ditampung dan kemudian ditindaklanjuti.

Mittasys merupakan perusahaan yang bergerak dibidang jasa pemrograman. Perusahaan ini belum memiliki sistem yang mampu menangani masalah hubungan perusahaan dengan pelanggan. Dalam *after sales services*, perusahaan belum dapat memaksimalkan pelayanan dalam bentuk keluhan ataupun pertanyaan pelanggan. Permasalahan tersebut akan berdampak buruk terhadap loyalitas pelanggan jika tidak segera diberikan solusi.

Berdasarkan permasalahan diatas, maka diusulkan adanya sebuah aplikasi *Help Desk* yang berfungsi sebagai media komunikasi dua arah antara perusahaan dengan pelanggan. Pelanggan dapat mengajukan keluhan dan pertanyaan tentang program yang digunakan dengan aplikasi ini. Peran lainnya adalah dapat mengarahkan pelanggan untuk berinventasi pada layanan yang ditawarkan untuk membantu bisnis perusahaan mencapai tujuan sehingga menjadi salah satu strategi dalam meningkatkan kualitas pelayanan yang baik.

I. METODOLOGI PENELITIAN

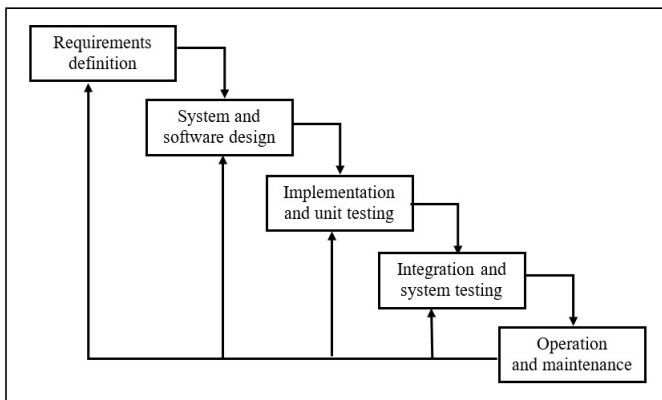
A. Metode Pengumpulan Data

Dalam pelaksanaan kegiatan pengumpulan data yang menjadi dasar dan untuk melengkapi laporan skripsi ini, maka digunakan teknik yang umum dalam kegiatan ilmiah, yaitu:

1. Wawancara (*Interview*), yaitu kegiatan berupa tanya jawab langsung dengan orang yang berkaitan dengan pengelolaan data keluhan atau pertanyaan. Wawancara ini dilakukan kepada Bapak Victor, S.Kom selaku pimpinan Mittasys.
2. Observasi, pengamatan langsung ke perusahaan dimana dilaksanakan penelitian ini. Metode ini cukup efektif karena langsung meninjau ke bagian yang mengelola data. Observasi dilakukan dengan mengunjungi langsung ke kantor Mittasys di Jl. Ahmad Yani No. 3A Tanjungpinang.
3. Studi Literatur, pengumpulan data yang diambil dari perusahaan yang bersangkutan.

B. Metode Pengembangan Perangkat Lunak

Dalam hal ini, tahap pengembangan sistem yang digunakan dalam laporan ini adalah menggunakan Metodologi *Life Cycle (Waterfall)*.



Gambar 1. Metode Pengembangan *Waterfall* Menurut Somerville (2011)

1. Requirement Definition

Tahap ini menguraikan kebutuhan aplikasi yang utuh menjadi komponen-komponen aplikasi untuk mengetahui bagaimana aplikasi dibangun. Informasi ini biasanya dapat diperoleh melalui wawancara, survei atau diskusi. Informasi tersebut dianalisis untuk mendapatkan dokumentasi kebutuhan pengguna untuk digunakan pada tahap selanjutnya.

2. System dan Software Design

Tahap ini merupakan tahap perancangan sistem. Pada tahap ini perancangan difokus pada perancangan struktur data, arsitektur perangkat lunak, representasi *interface* dan algoritma dalam pembuatan aplikasi. Tahap ini membantu dalam menspesifikasikan kebutuhan *hardware* dan sistem serta mendefinisikan arsitektur sistem secara keseluruhan.

3. Implementation and Unit Testing

Pada tahap ini dilakukan penerjemahan hasil rancangan kedalam bahasa pemrograman komputer. Setelah *design* selesai diterjemahkan dilakukan pengujian terhadap sistem yang telah dibuat tadi dengan tujuan untuk menemukan kesalahan yang terjadi dan kemudian diperbaiki. Selain itu dalam tahap ini juga dilakukan pemeriksaan terhadap modul

yang dibuat, apakah sudah memenuhi fungsi yang diinginkan atau belum.

4. Integration and System Testing

Pada tahap *integration and system testing* ini merupakan tahapan akhir dari pembuatan aplikasi *help desk*, pada tahap ini sistem sudah bisa digunakan oleh peneliti.

5. Operation and Maintenance

Ini merupakan tahap terakhir dalam model *waterfall*. *Software* yang sudah jadi dijalankan serta dilakukan pemeliharaan. Pemeliharaan termasuk dalam memperbaiki kesalahan yang tidak ditemukan pada langkah sebelumnya. Perhatikan implementasi unit sistem dan peningkatan jasa sistem sebagai kebutuhan baru.

II. HASIL DAN PEMBAHASAN

A. Analisa Sistem yang Berjalan

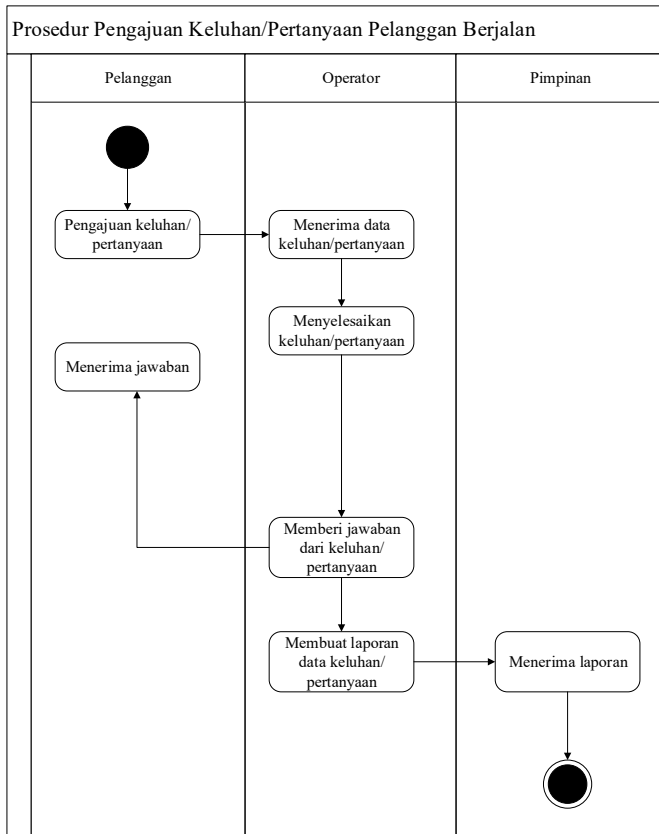
*Node.js* adalah perangkat lunak yang didesain untuk mengembangkan aplikasi berbasis *web* dan ditulis dalam sintaks bahasa pemrograman *JavaScript*. Ditulis dengan bahasa *JavaScript* dan dijalankan pada *Windows, Mac OS X*, dan *Linux* tanpa perubahan kode program. *Node.js* memiliki pustaka peladen HTTP sendiri sehingga memungkinkan untuk menjalankan peladen *web* tanpa menggunakan program peladen *web* seperti *Apache* atau *Lighttpd*.

Analisis sistem merupakan sebuah istilah yang secara kolektif mendeskripsikan fase-fase awal pengembangan sistem. Analisis sistem adalah teknik pemecahan masalah yang menguraikan bagian-bagian komponen dengan mempelajari, mengidentifikasi dan mengevaluasi berbagai macam masalah atau hambatan yang timbul pada sistem sehingga nantinya bisa dilakukan penanggulangan, perbaikan dan juga pengembangan.

Analisis sistem yang berjalan bertujuan untuk mengetahui lebih jelas bagaimana cara kerja sistem tersebut dan masalah yang dihadapi sistem untuk dapat dijadikan landasan usulan perancangan analisa sistem yang sedang berjalan yang dilakukan berdasarkan urutan kejadian yang ada.

Dibawah ini adalah prosedur pengajuan keluhan/pertanyaan pada pelanggan

1. Pelanggan mengajukan keluhan/pertanyaan.
2. Operator menerima keluhan/pertanyaan dari pelanggan.
3. Operator menganalisa data keluhan/pertanyaan.
4. Operator memberi jawaban dari keluhan/pertanyaan yang diajukan pelanggan. Kemudian jawaban diberikan kepada pelanggan
5. Operator memberikan jawaban kepada pelanggan.
6. Pelanggan menerima jawaban.
7. Operator membuat laporan data keluhan/pertanyaan dari pelanggan.
8. Pimpinan menerima laporan dari operator.



Gambar 2. Activity Diagram berjalan Pengajuan Keluhan/Peranyaan Pelanggan

**B. Analisa Kelemahan Sistem**

Terdapat beberapa kelemahan yang ada dalam sistem pada saat ini, yaitu:

1. Pencatatan data (keluhan/pertanyaan) masih dilakukan secara manual.
2. Data yang masuk harus dianalisis dan dibahas terlebih dahulu.
3. Tidak adanya prosedur umum dalam pengajuan keluhan/pertanyaan pelanggan terhadap perusahaan.
4. Data yang dicatat sering hilang sehingga tidak adanya *feedback* kepada pelanggan.

**C. Analisa Kebutuhan Sistem**

Berdasarkan penelitian yang dilakukan, diketahui bahwa kebutuhan sistem saat ini adalah penyediaan aplikasi yang dapat menangani keluhan/pertanyaan program dan *database* yang terintegrasi, sehingga akan membantu pelanggan dalam memperoleh informasi dan perbaikan lebih cepat.

**D. Analisa Pengguna**

Analisis pengguna mencirikan siapa saja pengguna dari perangkat lunak yang dispesifikasikan dan apa saja haknya terhadap perangkat lunak tersebut. Pada aplikasi ini ada beberapa pengguna, yaitu:

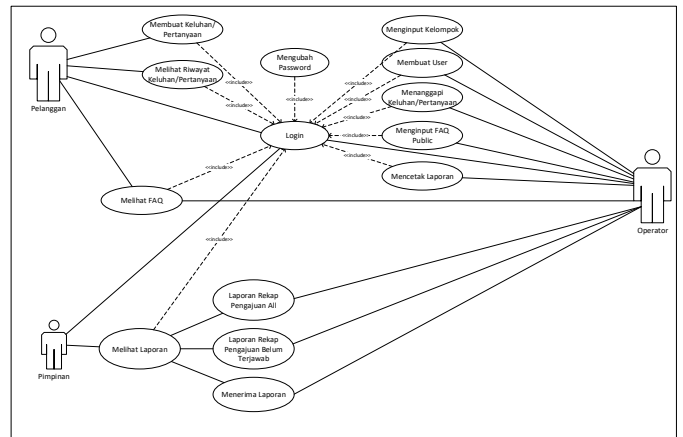
1. Pelanggan

Pada aplikasi ini pelanggan merupakan orang yang mengakses dan menggunakan aplikasi ini untuk melakukan pengajuan keluhan/pertanyaan.

2. Operator  
Operator merupakan orang yang bertanggungjawab mengelola data keluhan/pertanyaan kepada pelanggan dan pimpinan.
3. Pimpinan  
Pimpinan memiliki tanggungjawab untuk memantau dan menerima data yang diberikan oleh operator.

**E. Use Case Diagram yang diusulkan**

UML menyediakan serangkaian gambar dan diagram yang sangat baik. Beberapa diagram memfokuskan diri pada ketanggahan teori *object-oriented* dan sebagian lagi memfokuskan pada detail rancangan dan konstruksi. Semuanya dimaksudkan sebagai sarana komunikasi antartim *programmer* maupun dalam pengguna. *Use case* diagram menjelaskan interaksi yang terjadi antara aktor dengan sistem. Aktor yang terdapat dalam aplikasi ini adalah: pelanggan, operator dan pimpinan.



Gambar 3. Use Case Diagram yang diusulkan

Tabel I  
Deskripsi Use Case Diagram

No	Aktor	Use Case	Keterangan
1	Pelanggan, Operator, Pimpinan	Login	Setiap aktor harus melakukan login terlebih dahulu untuk melakukan <i>entry</i> data pada aplikasi
2	Pelanggan, Operator, Pimpinan	Mengubah <i>password</i>	Setiap aktor dapat mengubah <i>password</i>
3	Pelanggan	Membuat Keluhan/Pertanyaan	Pelanggan menginput data keluhan/pertanyaan pada aplikasi
4	Pelanggan	Melihat Riwayat Keluhan/Pertanyaan	Pelanggan melihat riwayat data keluhan/pertanyaan pada aplikasi
5	Operator	Membuat User	Operator menginput user baru untuk digunakan oleh pelanggan dalam mengakses
6	Operator	Menanggapi Keluhan/Pertanyaan	Operator menanggapi keluhan/pertanyaan yang diajukan oleh pelanggan
7	Operator	Mencetak Laporan	Operator mencetak laporan
8	Operator	Membuat <i>FAQ Public</i>	Operator menginput <i>FAQ Public</i> yang akan muncul di

			halaman <i>FAQ</i>
9	Operator	Menginput Kelompok	Operator menginput kelompok untuk digunakan sebagai data <i>master</i> di <i>form</i> pengajuan
10	Operator, Pelanggan	Melihat <i>FAQ</i>	Operator dan pelanggan dapat mengakses halaman <i>FAQ</i> dengan catatan harus <i>login</i> ke aplikasi terlebih dahulu
11	Operator, Pimpinan	Melihat Laporan	Operator dan Pimpinan dapat melihat laporan, yang terdiri dari: laporan rekap pengajuan <i>all</i> dan laporan rekap pengajuan belum terjawab

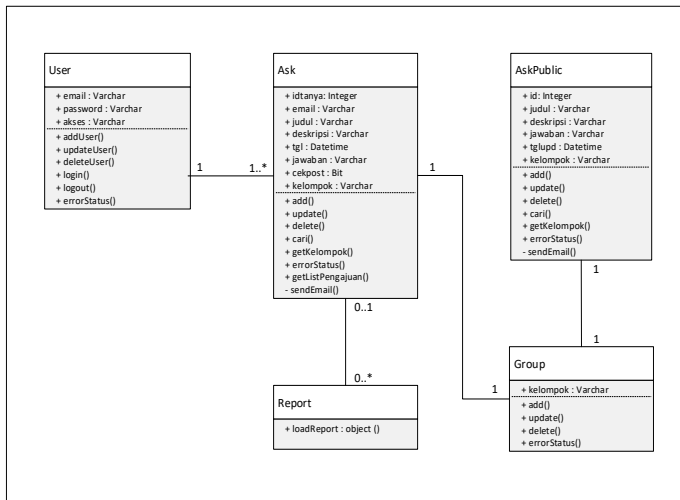
6	jawaban	Varchar	500
7	cekpost	Bit	
8	kelompok	Varchar	30

Tabel V  
Tabel tb tanyaublic

No	Nama Field	Tipe	Ukuran
1	id	Int	
2	judul	Varchar	255
3	deskripsi	Varchar	500
4	jawaban	Varchar	500
5	tglupd	Datetime	
6	kelompok	Varchar	30

F. Class Diagram yang diusulkan

Class Diagram atau diagram kelas merupakan diagram yang menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Diagram kelas memiliki 3 bagian utama, yaitu *attribute*, *operation*, dan *name*. Kelas-kelas yang ada pada struktur sistem harus dapat melakukan fungsi-fungsi sesuai dengan kebutuhan sistem. Berikut adalah gambaran *class* diagram yang diusulkan:



Gambar 4 Class Diagram yang diusulkan

G. Perancangan Struktur File

Tabel II  
Tabel tb user

No	Nama Field	Tipe	Ukuran
1	Email	Varchar	255
2	Password	Varchar	255
3	Akses	Varchar	20

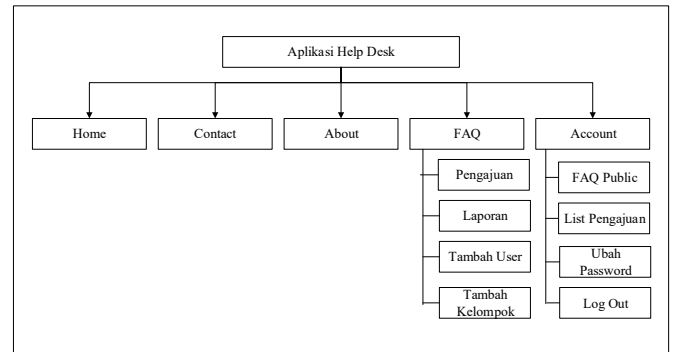
Tabel III  
Tabel tb kelompok

No	Nama Field	Tipe	Ukuran
1	kelompok	Varchar	50

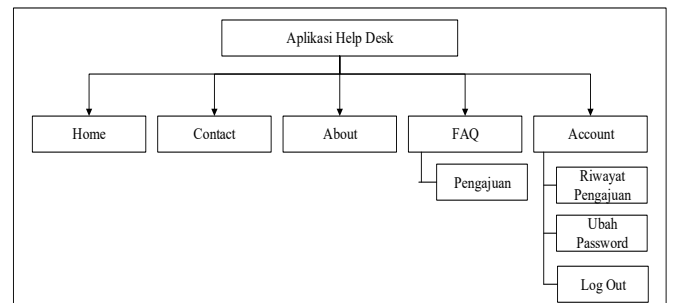
Tabel IV  
Tabel tb tanya

No	Nama Field	Tipe	Ukuran
1	id tanya	Int	
2	email	Varchar	255
3	judul	Varchar	255
4	deskripsi	Varchar	500
5	tgl	Datetime	

H. Perancangan Struktur Menu



Gambar 5 Perancangan Struktur Menu Operator

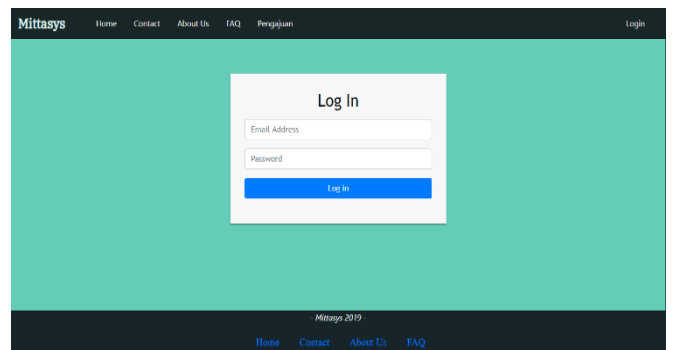


Gambar 6 Perancangan Struktur Menu Pelanggan

I. Implementasi Sistem

1. Halaman Login

Halaman *login* merupakan halaman yang akan tampil ketika *user* masuk ke aplikasi.



Gambar 7 Halaman Login

2. Halaman Home

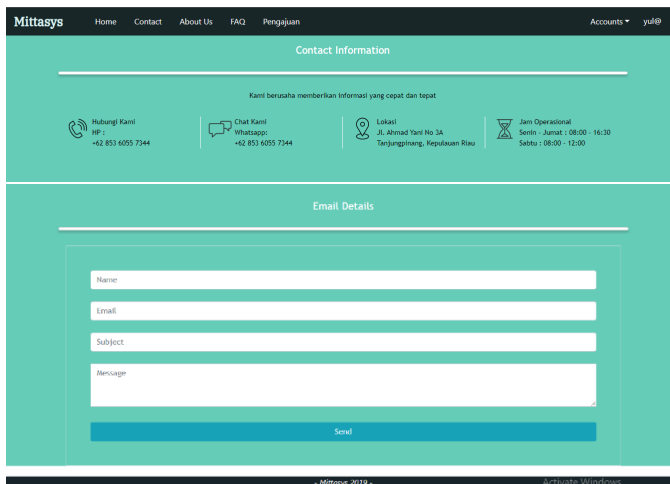
Setelah *user* berhasil menjalankan proses *login* maka, tampilan halaman *home* akan muncul.



Gambar 8 Halaman Home

### 3. Halaman Contact

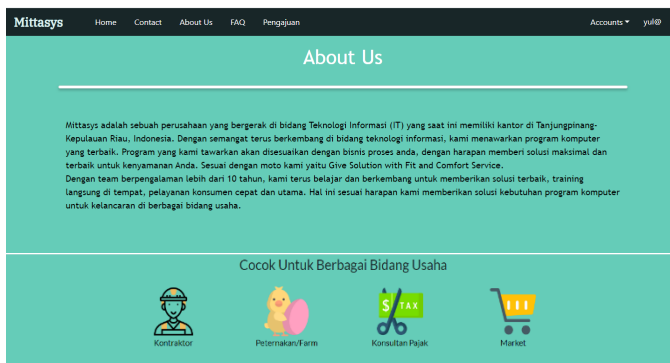
Halaman *contact* menampilkan informasi mengenai operasional perusahaan serta *form email* untuk menanyakan seputar jasa yang lebih lanjut.



Gambar 9 Halaman Contact

### 4. Halaman About

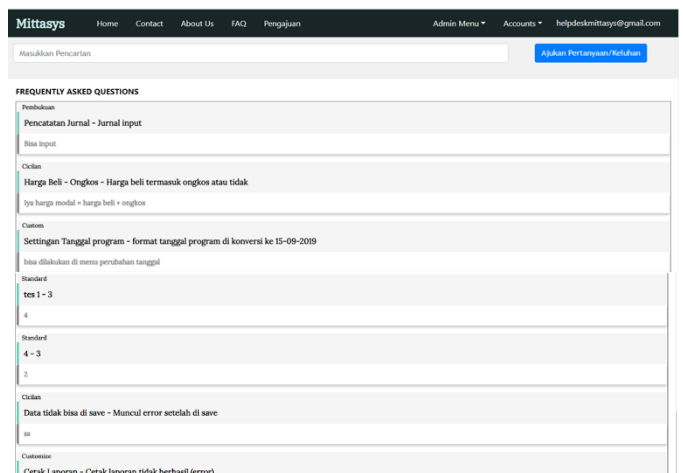
Halaman *About* berisikan informasi tentang sejarah perusahaan dan jenis-jenis jasa yang ditawarkan.



Gambar 9 Halaman About

### 5. Halaman FAQ

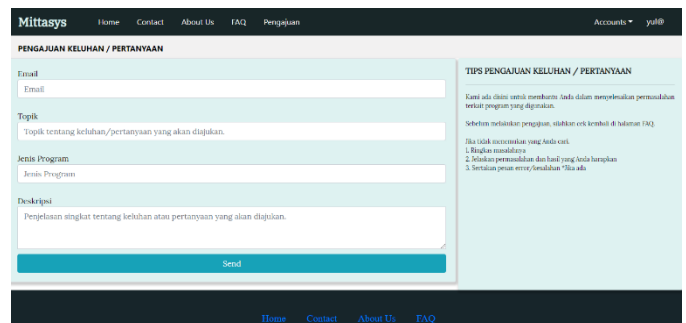
Halaman *FAQ* berisikan pertanyaan dan jawaban yang sering ditanyakan oleh pelanggan.



Gambar 10 Halaman FAQ

### 6. Halaman Pengajuan

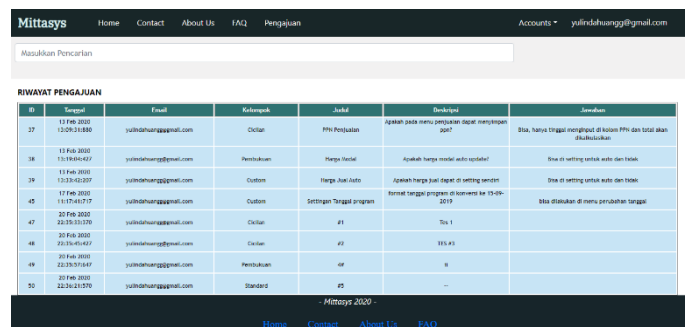
Halaman pengajuan berfungsi sebagai form untuk pelanggan menginputkan keluhan/pertanyaan pada perusahaan.



Gambar 11 Halaman Pengajuan

### 7. Halaman Riwayat Pengajuan

Halaman riwayat pengajuan menampilkan data sesuai dengan *email* pelanggan yang melakukan pengajuan.



Gambar 12 Halaman Riwayat Pengajuan

### 8. Halaman List Pengajuan

Halaman ini merupakan halaman yang diakses oleh operator. Semua data pengajuan pelanggan akan di jawab di menu ini.

ID	Tanggal	Email	Judul	Deskripsi	Jawaban	Aksi
50	20 Feb 2020	yulinda@gmail.com	#5			Standard Edit Hapus
49	20 Feb 2020	yulinda@gmail.com	#4			Pembakuan Edit Hapus
48	20 Feb 2020	yulinda@gmail.com	#2	TES #3		Ceklan Edit Hapus
47	20 Feb 2020	yulinda@gmail.com	#1	Tes 1		Ceklan Edit Hapus
46	20 Feb 2020	helpdeskmittasys@gmail.com	Pencetakan Jurnal	Jurnal Input	Bisa input	Pembakuan Edit Hapus
45	17 Feb 2020	yulinda@gmail.com	Setingan tanggal program	format tanggal program di konversi ke 15-09-2019	bisa dilakukan di menu perubahan tanggal	Custom Edit Hapus
39	13 Feb 2020	yulinda@gmail.com	Kejaga Jual Auto	Apakah harga jual dapat di setting sendiri?	Bisa di setting untuk auto dan tidak	Custom Edit Hapus
38	13 Feb 2020	yulinda@gmail.com	Keraja Modal	Apakah harga modal auto update?	Bisa di setting untuk auto dan tidak	Pembakuan Edit Hapus
37	13 Feb 2020	yulinda@gmail.com	PPH Penjualan	Apakah pada menu penjualan dapat menyimpan pph?	Bisa, harga thggl menginput di kolom PPH dan total akan diotomatiskan	Ceklan Edit Hapus
28	05 Feb 2020	stand@gmail.com	#3	Login error		Customize Edit Hapus
27	05 Feb 2020	stand@gmail.com	#2	Data yang sudah di simpan, tidak muncul		Customize Edit Hapus
26	05 Feb 2020	stand@gmail.com	#1	error not found		Customize Edit Hapus
25	05 Feb 2020	stand@gmail.com	Apakah harga jual dapat di setting sendiri?		Bisa di setting untuk auto dan tidak	Standard Edit Hapus
24	05 Feb 2020	stand@gmail.com	Apakah harga modal auto update?		Bisa di setting untuk auto dan tidak	Standard Edit Hapus
23	05 Feb 2020	stand@gmail.com	Apakah pada menu penjualan dapat menyimpan pph?		Bisa, harga thggl menginput di kolom PPH dan total akan diotomatiskan	Standard Edit Hapus

Gambar 13 Halaman List Pengajuan

9. Halaman FAQ Public

Halaman ini diinput oleh operator berupa data pertanyaan yang sering ditanyakan oleh pelanggan, dan data yang diinput di menu ini akan muncul di halaman FAQ.

Grup	Judul	Deskripsi	Jawaban	Tgl Update	Aksi
Ceklan					
Standard	#50	100000	121111		Edit Hapus
Pembakuan	Default	Program Default	123		Edit Hapus
Ceklan	Harga Beli - Ongkos	Harga beli termasuk ongkos atau tidak	Ya harga modal + harga beli - ongkos	17 Feb 2020 15:02:28.647	Edit Hapus
Pembakuan	1	2	3	16 Feb 2020 22:30:21.000	Edit Hapus
Standard	tes 1	1	2	16 Feb 2020 22:30:17.720	Edit Hapus
Standard	4	1	2	16 Feb 2020 19:27:06.883	Edit Hapus
Ceklan	Data tidak bisa di save	Rancor error setelah di save	0	16 Feb 2020 19:00:02.310	Edit Hapus

Gambar 14 Halaman FAQ Public

10. Halaman Ubah Password

Halaman ini digunakan oleh user untuk mengubah password.

Gambar 15 Halaman Ubah Password

11. Halaman Tambah User

Halaman tambah user digunakan oleh operator untuk menambahkan user baru.

Email	Password	Alias	Aksi
id@		id@	Edit Hapus
id@		Admin	Edit Hapus
yulinda@gmail.com		Admin	Edit Hapus
yulinda@gmail.com		Admin	Edit Hapus

Gambar 16 Halaman Tambah User

12. Halaman Tambah Kelompok

Halaman tambah kelompok adalah master jenis program yang muncul pada halaman pengajuan yang diinputkan oleh operator.

Kelempok	Status	Aksi
Jual Beli	Standar	Edit Hapus
Mulang Putang	Standar	Edit Hapus
Hotel	Standar	Edit Hapus
Import Ekspor	Standar	Edit Hapus
Bangkit	Standar	Edit Hapus
Penjualan	Standar	Edit Hapus
Ceklan	Standar	Edit Hapus
Custom	Standar	Edit Hapus
Pembakuan	Standar	Edit Hapus
Standard	Standar	Edit Hapus

Gambar 17 Halaman Tambah Kelompok

13. Laporan Rekap Pengajuan

Laporan rekap pengajuan adalah laporan akhir untuk melihat data-data pengajuan keluhan.pertanyaan yang diinput oleh pelanggan.

ID	Tanggal	Kelempok	Email	Judul-Deskripsi
21	05-02-2020	Standard	ke@mittasys.com	Tidak bisa save - setelah di data produk, dan muncul modal save. Muncul error "Item tidak bisa di simpan"
22	05-02-2020	Pembakuan	ke@mittasys.com	Item error laporan tidak muncul. Cetak laporan rekening rekening M yang muncul 0?
23	05-02-2020	Standard	stand@gmail.com	Apakah pada menu penjualan dapat menyimpan pph?
24	05-02-2020	Standard	stand@gmail.com	Apakah harga modal auto update?
25	05-02-2020	Standard	stand@gmail.com	Apakah harga jual dapat di setting sendiri?
26	05-02-2020	Customize	stand@gmail.com	#1 - error not found
27	05-02-2020	Customize	stand@gmail.com	#2 - Data yang sudah di simpan, tidak muncul
28	05-02-2020	Customize	stand@gmail.com	#3 - Login error
38	05-02-2020	Customize	stand@gmail.com	Cetak Laporan - Cetak laporan tidak berhalat muncul
37	13-02-2020	Ceklan	yulinda@gmail.com	PPH Penjualan - Apakah pada menu penjualan dapat menyimpan pph?
38	13-02-2020	Pembakuan	yulinda@gmail.com	Harga Modal - Apakah harga modal auto update?

Gambar 18 Laporan Rekap Pengajuan

III. KESIMPULAN

Perancangan Aplikasi IT Help Desk Menggunakan Platform Node.js Pada Mittasys dibuat atas tindak lanjut dari sistem yang sedang berjalan. Aplikasi ini diharapkan dapat membantu perusahaan dalam menangani permasalahan yang ada. Adapun kesimpulan yang dapat di ambil dari pembuatan aplikasi ini adalah: memberikan kemudahan layanan data dalam mengakses informasi dengan menggunakan fitur pencarian data terkait dengan kebutuhan, perancangan aplikasi Help Desk menggunakan platform node.js dapat diakses oleh pelanggan untuk menginput pengajuan keluhan atau pertanyaan. Selain itu, aplikasi dapat menampilkan riwayat data pengajuan sesuai akses login. Operator dan pelanggan dapat dengan mudah mengetahui pengajuan data yang masuk dan yang sudah ditanggapi dengan fitur auto email sebagai notifikasi dalam proses pengajuan.

REFERENSI

- [1] Brown Ethan, *Web Development with Node and Express*, O'Reilly Media. Inc, 2019, USA
- [2] Chan Syahrial, *Membuat Aplikasi Database dengan PowerBuilder 12.6 dan MySQL*, PT. Elex Media Komputindo, 2017, Jakarta.

- [3] Elisabet Yunaeti dan Rita Irviani, *Pengantar Sistem Informasi*, CV. ANDI OFFSET, 2017, Yogyakarta
- [4] Emma Utami & Sukrisno, *Mengoptimalkan Query pada Microsoft SQL Server*, CV. ANDI OFFSET, 2008, Yogyakarta
- [5] Hernita P, *Panduan Praktis Menguasai Pemrograman Web dengan Javascript*, Wahana Komputer, 2010, Yogyakarta
- [6] Hutahaean Jeperson, *Konsep Sistem Informasi*, Deepublish, 2015, Yogyakarta
- [7] Muhammad Mushuliddin dan Oktafianto, *Analisis dan Perancangan Sistem Informasi Menggunakan Metode Terstruktur dan UML*, CV. ANDI OFFSET, 2016, Yogyakarta
- [8] Ombro George, *Sams Teach Yourself Node.js in 24 Hours*, SAMS, 2012, USA
- [9] Yudho Yanto & Ardhi Wijayanto, *Yuk Berbisnis dengan Laravel dan Android*, PT. Elex Media Komputindo, 2019, Jakarta