

BAB 3

METODE PENELITIAN

Penelitian ini merupakan jenis penelitian berbasis perancangan dan pengembangan. Kegiatan penelitian dimulai dengan mengidentifikasi latar belakang permasalahan, memetakan alur proses yang terlibat, menelusuri akar permasalahan, lalu merancang serta membangun sebuah sistem yang bertujuan untuk mengurangi atau mengatasi permasalahan tersebut.

3.1 BAHAN DAN ALAT PENELITIAN

Dalam penelitian ini, bahan yang digunakan berupa informasi-informasi dari berbagai sumber. Data primer diperoleh melalui wawancara langsung dengan pihak perusahaan penyedia layanan internet lokal terkait proses penerimaan dan penanganan keluhan.

Instrumen penelitian yang digunakan meliputi:

1. **Wawancara**, untuk menggali informasi dari pihak perusahaan terkait proses penerimaan keluhan, kendala yang dihadapi, serta harapan terhadap sistem monitoring ke depannya.
2. **Observasi langsung**, terhadap proses penanganan keluhan dan kekurangan sistem yang telah berjalan sebelumnya

Selain bahan penelitian, alat yang digunakan adalah perangkat komputer/laptop dengan spesifikasi yang mendukung untuk menjalankan sistem operasi dan perangkat lunak pendukung lainnya. Sistem operasi dan program aplikasi yang digunakan dalam pengembangan sistem *monitoring* tiket pelayanan aduan gangguan internet adalah sebagai berikut:

1. Sistem Operasi: Windows 10.
2. Server *Local*: XAMPP.
3. *Database*: MySQL.
4. *Text Editor*: Visual Studio Code.
5. Bahasa Pemrograman: Python
6. *Framework Web*: Django

7. Frontend: CSS *Tailwind*

Penggunaan alat dan bahan tersebut bertujuan untuk mendukung proses pengembangan sistem secara optimal, serta memastikan bahwa data yang dikumpulkan akurat, relevan, dan mendukung tercapainya tujuan penelitian.

3.2 JALAN PENELITIAN

Penelitian ini dilakukan melalui beberapa tahapan yang sistematis untuk memastikan bahwa proses perancangan dan pengembangan sistem *monitoring* tiket pelayanan aduan gangguan internet berjalan dengan terarah dan sesuai tujuan. Adapun tahapan jalannya penelitian dijelaskan sebagai berikut:

1. Studi Literatur

Pada tahap awal, dilakukan pengumpulan informasi dan referensi dari buku, jurnal ilmiah, artikel, serta dokumen terkait sistem informasi, *monitoring* gangguan jaringan, dan manajemen tiket gangguan. Tujuannya adalah untuk memahami konsep-konsep dasar serta teknologi yang relevan dalam pengembangan sistem.

2. Tahap Analisis Kebutuhan

Pada tahap ini, peneliti melakukan pengumpulan data yang relevan dengan kebutuhan penelitian. Data diperoleh melalui wawancara dengan pimpinan atau petugas yang bertanggung jawab dalam proses *monitoring* di PT Lintas Data Prima. Wawancara ini bertujuan untuk menggali informasi terkait permasalahan layanan dan kebutuhan sistem yang dibutuhkan perusahaan.

3. Analisis Kebutuhan Sistem

Berdasarkan hasil observasi, dan wawancara, dilakukan analisis kebutuhan sistem baik dari sisi fungsional (fitur yang dibutuhkan) maupun non-fungsional (kinerja, keamanan, dan *usability*). Hasil analisis ini digunakan sebagai dasar dalam proses perancangan sistem.

4. Perancangan Sistem

Pada tahap ini dilakukan perancangan sistem *monitoring* tiket pelayanan aduan gangguan internet meliputi:

a. Perancangan *use case* diagram

- b. Perancangan *Activity* diagram
 - c. Perancangan *Class* diagram
 - d. Perancangan antarmuka (UI)
 - e. Perancangan struktur tabel basis data
5. Implementasi Sistem
- Setelah desain selesai, sistem mulai diimplementasikan menggunakan bahasa pemrograman dan teknologi yang sesuai. Implementasi mencakup pengembangan *backend*, *frontend*, serta integrasi dengan *database*.
6. Pengujian Sistem
- Sistem diuji menggunakan metode *black box testing* untuk memastikan bahwa setiap fungsi berjalan sesuai dengan spesifikasi. Uji coba dilakukan dengan melibatkan *user internal* untuk mendapatkan *feedback* terhadap kinerja dan kemudahan penggunaan sistem.
7. Evaluasi dan Perbaikan
- Berdasarkan hasil pengujian dan masukan pengguna, dilakukan evaluasi untuk memperbaiki kekurangan sistem. Tahap ini bersifat iteratif dan bertujuan untuk memastikan bahwa sistem yang dikembangkan benar-benar sesuai kebutuhan pengguna.
8. Dokumentasi dan Penyusunan Laporan
- Tahap akhir adalah pendokumentasian seluruh proses penelitian dan pengembangan sistem ke dalam laporan skripsi. Dokumentasi mencakup penjelasan tahapan, diagram sistem, kode program, serta kesimpulan dan saran dari hasil.

3.3 ANALISIS KEBUTUHAN SISTEM

Tahap ini bertujuan untuk mengidentifikasi kebutuhan pengguna terhadap sistem *monitoring* tiket pelayanan aduan gangguan internet. Proses analisis dilakukan berdasarkan hasil observasi, dan wawancara kepada pihak internal perusahaan ISP (seperti teknisi dan *customer service*). Analisis ini mencakup:

1. **Kebutuhan fungsional**, seperti fitur pelaporan gangguan, pelacakan status tiket, notifikasi pembaruan, dan *dashboard*.

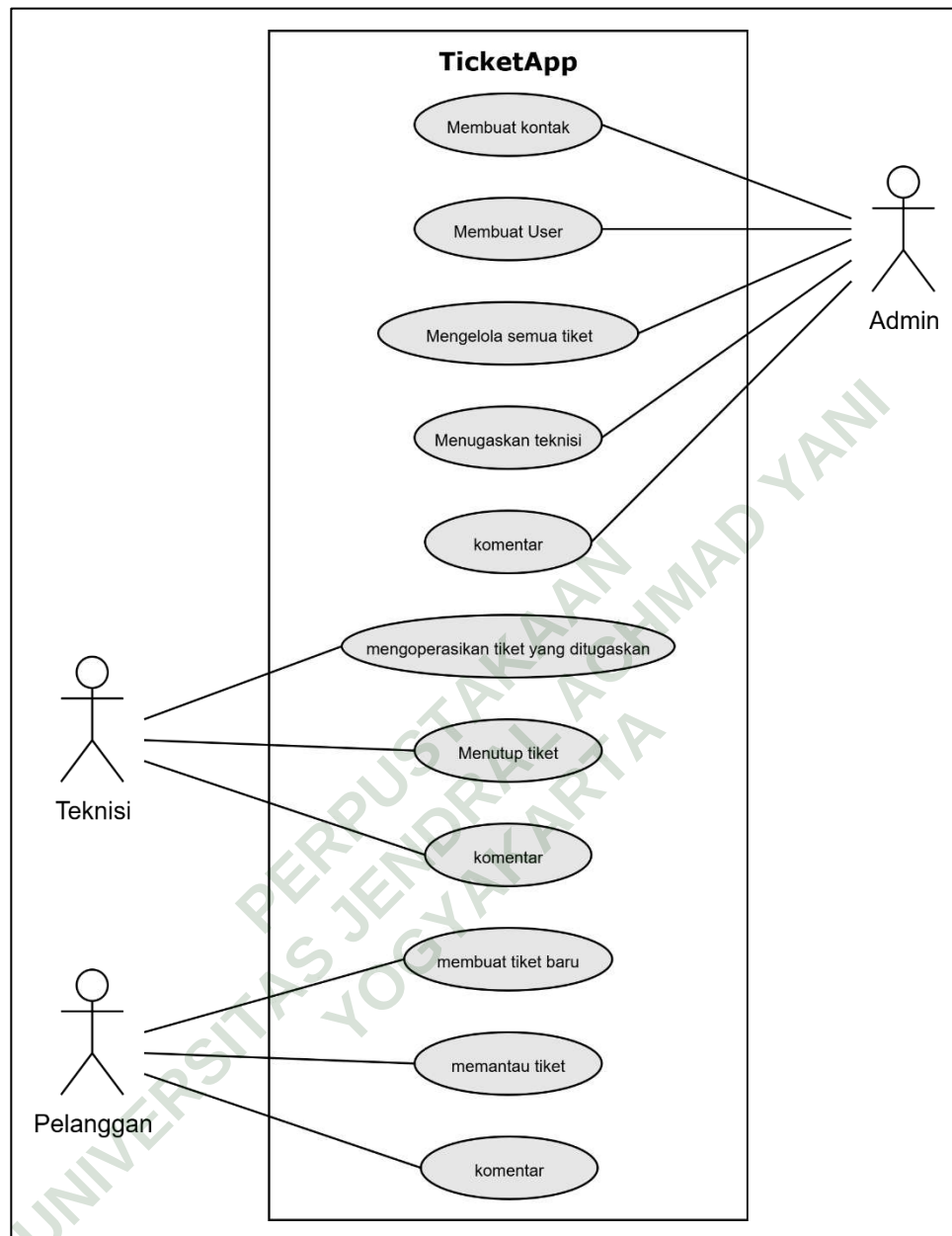
2. **Kebutuhan non-fungsional**, seperti kemudahan penggunaan (*usability*), kecepatan sistem (*performance*), dan keamanan data (*security*).
3. **Kebutuhan perangkat**, termasuk perangkat keras dan lunak yang mendukung pengembangan dan operasional sistem

3.4 RANCANGAN SISTEM

Berikut rancangan sistem pada sistem *monitoring* pelayanan aduan gangguan internet di PT. Lintas Data Prima berbasis Aplikasi Web TicketApp.

3.4.1 Use case Diagram

Use case diagram pada sistem TicketApp menggambarkan tiga aktor utama, yaitu Admin, Teknisi, dan Pelanggan, yang masing-masing memiliki akses terhadap fitur tertentu. Admin dapat membuat kontak, membuat *user*, mengelola semua tiket, membuat komentar, dan menugaskan teknisi untuk menangani tiket. Teknisi memiliki peran dalam melihat tiket yang ditugaskan, menutup tiket setelah menyelesaikan tugas, serta memberikan komentar pada tiket pelanggan sebagai bagian dari proses penanganan. Sementara itu, pelanggan dapat membuat tiket baru untuk melaporkan masalah, memantau perkembangan tiket yang mereka ajukan, dan membalas komentar teknisi guna memberikan informasi tambahan atau klarifikasi. Diagram ini menunjukkan alur kerja sistem yang memungkinkan kolaborasi efektif antara pelanggan, teknisi, dan admin dalam menyelesaikan tiket layanan pada Gambar 3.1.



Gambar 3.1 Use Case Diagram

Tabel 3.1 skenario *use case* TicketApp menampilkan hubungan antara aktor, fungsi sistem yang digunakan, dan deskripsi singkat alur interaksi. Tabel ini membantu memahami peran masing-masing pengguna dalam menjalankan proses pada sistem.

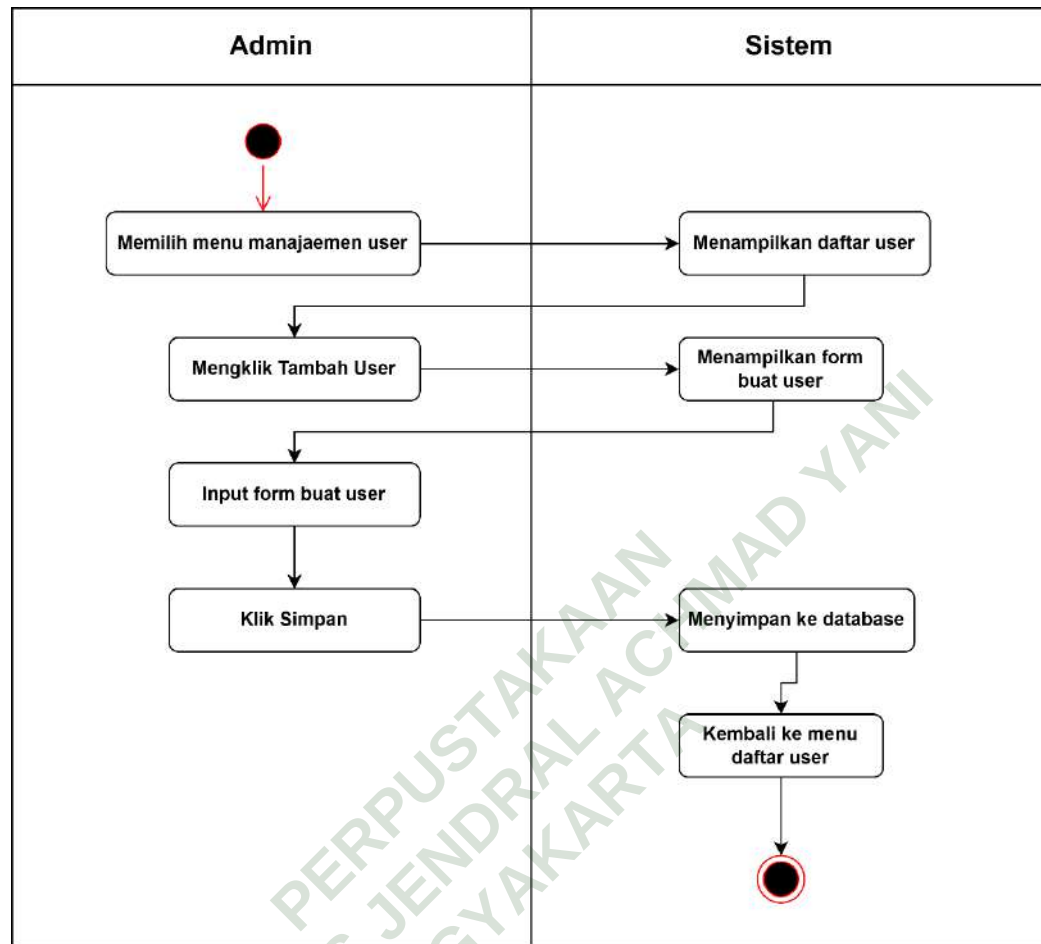
Tabel 3.1 Skenario *Use Case Diagram*

Aktor	<i>Use case</i>	Deskripsi
Admin	Membuat <i>user</i>	Admin <i>login</i> ke sistem, mengisi data <i>user</i> baru (nama, <i>email</i> , lokasi, nomer hp, peran), lalu menyimpannya.
Admin	Menugaskan Teknisi	Admin masuk ke sistem, memilih tiket yang belum diproses, dan menetapkan teknisi yang sesuai.
Admin	Membuat kontak	Admin menambahkan kontak baru (pelanggan/teknisi), melengkapi detail dan menyimpannya.
Admin	Komentar tiket	Admin masuk ke detail tiket dan berkomentar untuk teknisi maupun pelanggan
Teknisi	Melihat tiket yang ditugaskan	Teknisi masuk ke sistem dan melihat daftar tiket yang telah ditugaskan kepadanya.
Teknisi	Mengomentari tiket pelanggan	Teknisi menulis komentar atau penjelasan tambahan di tiket pelanggan.
Teknisi	Menutup Tiket	Setelah menyelesaikan masalah, teknisi membuka tiket dan mengubah status menjadi "Selesai".
Pelanggan	Membuat Tiket Baru	Pelanggan <i>login</i> , mengisi formulir keluhan atau permintaan, lalu mengirimkannya
Pelanggan	Memantau tiket	Pelanggan melihat status dan perkembangan tiket yang sudah mereka buat.

Pelanggan	Membalas komentar	Pelanggan membalas komentar teknisi pada tiket untuk memberi klarifikasi atau info.
-----------	-------------------	---

3.4.2 Activity Diagram Admin Menambahkan User

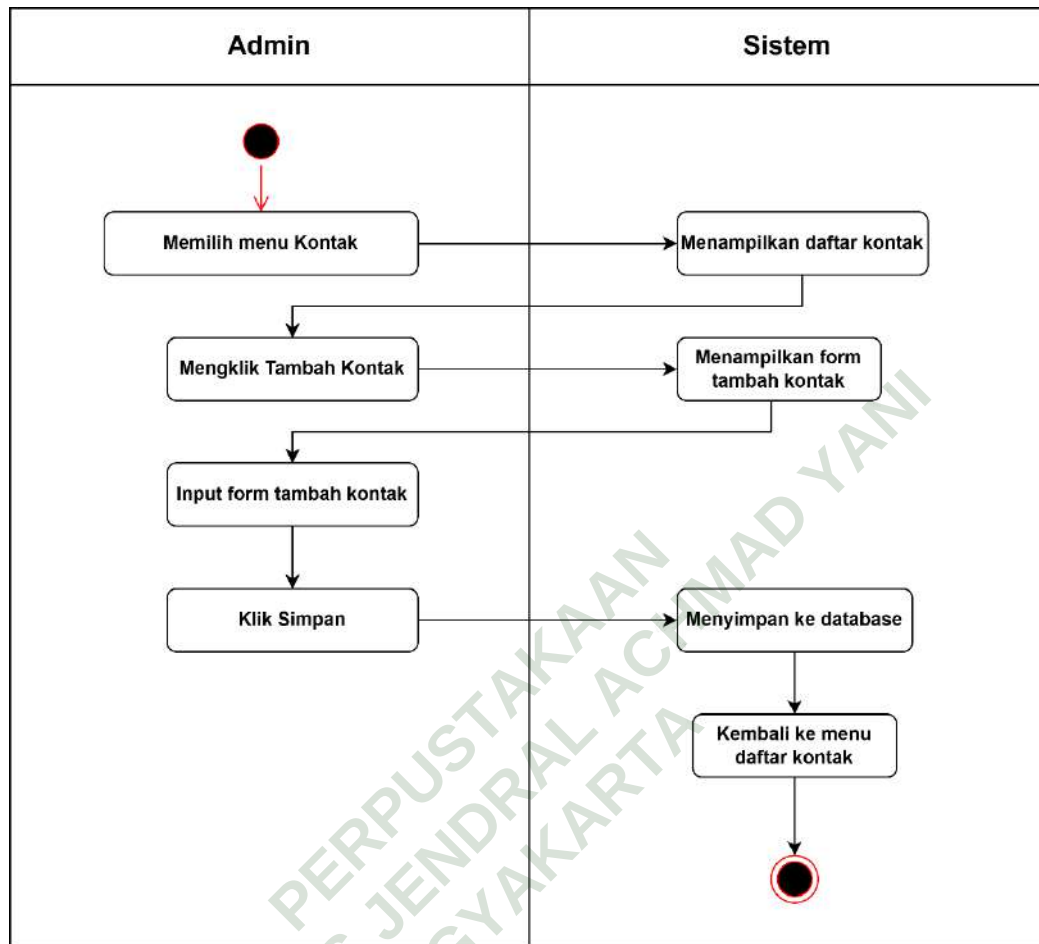
Activity Diagram menggambarkan proses admin menambahkan *user* baru ke dalam sistem. Proses dimulai ketika admin memilih menu manajemen *user*, kemudian sistem merespons dengan menampilkan daftar *user* yang sudah ada. Selanjutnya, admin melakukan aksi dengan mengklik tombol "Tambah User", dan sistem akan menampilkan *form input* untuk data *user* baru. Admin kemudian mengisi *form* tersebut dengan data yang diperlukan (seperti nama, *email*, peran, dll.), lalu menekan tombol "Simpan". Setelah itu, sistem akan menyimpan data ke dalam *database* dan kemudian secara otomatis kembali ke tampilan daftar *user*, yang menandakan akhir dari alur proses ini. Diagram ini menunjukkan interaksi antara admin dan sistem secara terstruktur untuk memastikan bahwa penambahan *user* dilakukan dengan benar dan sistematis pada gambar 3.2.



Gambar 3.2 Activity Diagram Admin Membuat User

3.4.3 Activity Diagram Admin Membuat Kontak

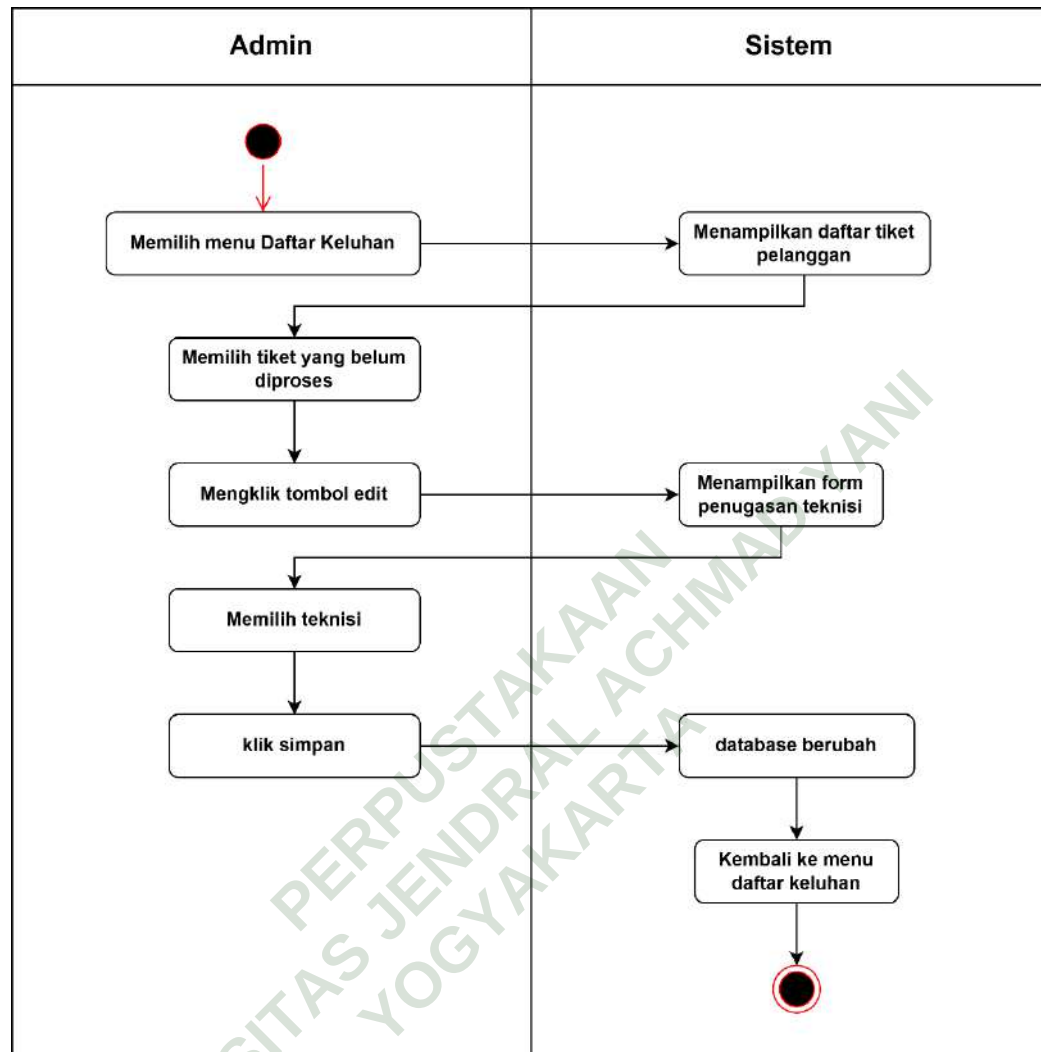
Diagram aktivitas tersebut menunjukkan alur proses admin menambahkan kontak ke dalam sistem. Aktivitas dimulai ketika admin memilih menu Kontak, dan sistem akan menampilkan daftar kontak yang sudah ada. Setelah itu, admin mengklik tombol "Tambah Kontak", lalu sistem merespons dengan menampilkan *form* penambahan kontak. Admin kemudian mengisi *form* kontak dengan informasi yang diperlukan (seperti nama, nomor, atau *email* kontak), lalu menekan tombol "Simpan". Sistem kemudian akan menyimpan data kontak ke dalam *database*, dan setelah penyimpanan berhasil, sistem akan mengembalikan tampilan ke daftar kontak. Proses ini berakhir di titik akhir aktivitas. Diagram ini menggambarkan langkah-langkah yang jelas dan sistematis dalam proses penambahan data kontak oleh admin pada gambar 3.3.



Gambar 3.3 Activity Diagram Admin Menambahkan Kontak

3.4.4 Activity Diagram Admin Penugasan Teknisi

Activity diagram admin menugaskan tiket ke teknisi dimulai ketika admin memilih menu Daftar Keluhan, yang memicu sistem untuk menampilkan daftar tiket pelanggan. Selanjutnya, admin memilih tiket yang belum diproses dan menekan tombol edit, sehingga sistem menampilkan *form* penugasan teknisi. Pada tahap ini, admin memilih teknisi yang akan menangani tiket tersebut dan mengklik tombol simpan. Setelah itu, sistem akan memperbarui data di *database* dan mengembalikan tampilan ke menu daftar keluhan sebagai akhir dari proses penugasan pada gambar 3.4.

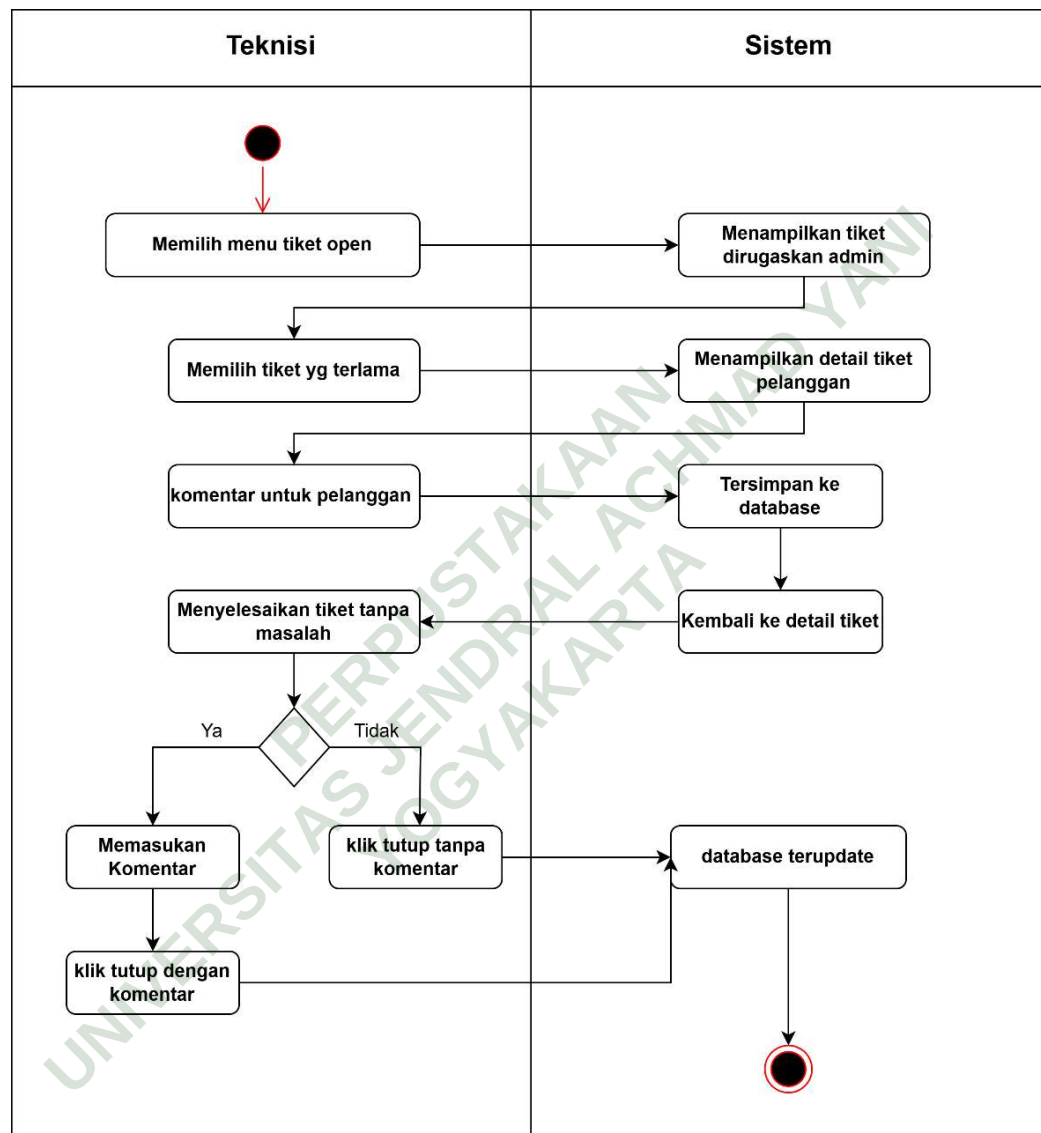


Gambar 3.4 Activity Diagram Admin Penugasan Teknisi

3.4.5 Activity Diagram Teknisi Menugaskan Tiket

Activity diagram teknisi menyelesaikan tiket menggambarkan alur kerja teknisi dalam menangani tiket gangguan yang ditugaskan oleh admin. Proses dimulai ketika teknisi memilih menu tiket *open*, lalu sistem menampilkan daftar tiket yang telah dirugaskan. Teknisi kemudian memilih tiket yang paling lama dan sistem menampilkan detail tiket tersebut. Jika diperlukan, teknisi dapat memberikan komentar untuk pelanggan, yang kemudian disimpan oleh sistem ke dalam *database* dan kembali ke halaman detail tiket. Setelah permasalahan diselesaikan, teknisi menentukan apakah ingin menambahkan komentar. Jika ya, maka teknisi memasukkan komentar dan menekan tombol tutup dengan komentar. Jika tidak,

teknisi langsung menekan tutup tanpa komentar. Akhirnya, sistem memperbarui data di dalam *database* sebagai penanda bahwa tiket telah selesai diproses pada gambar 3.5.

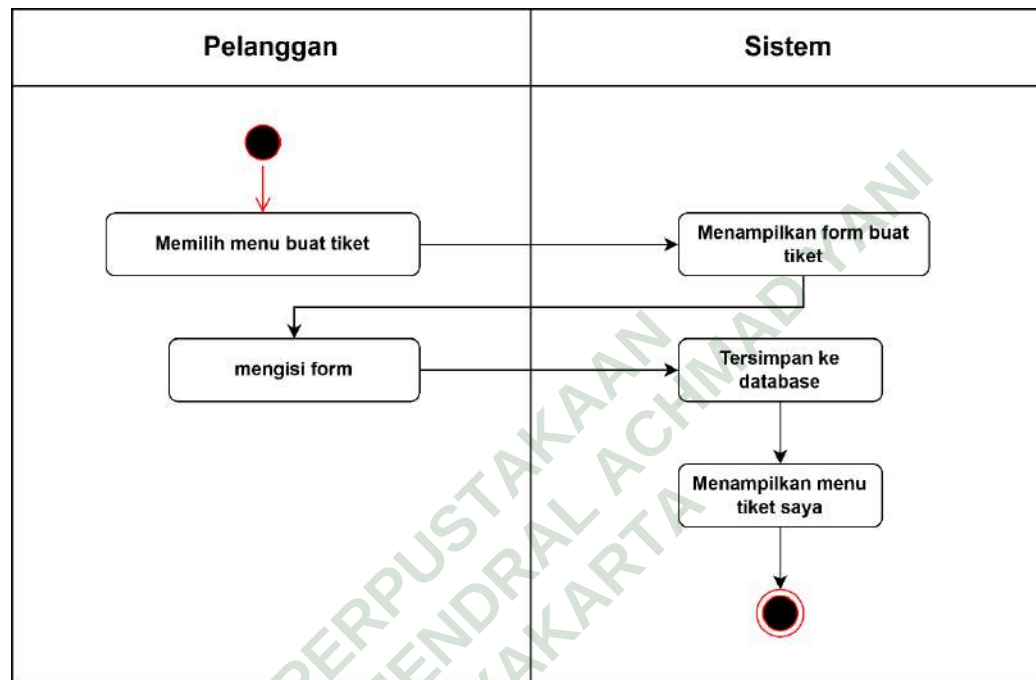


Gambar 3.5 Activity Diagram Teknisi Menugaskan Tiket

3.4.6 Activity Diagram Pelanggan Membuat Tiket

Activity diagram pelanggan membuat tiket baru menjelaskan proses ketika pelanggan mengajukan laporan gangguan atau permintaan bantuan. Proses diawali saat pelanggan memilih menu buat tiket, kemudian sistem menampilkan *form* pembuatan tiket. Pelanggan mengisi *form* tersebut sesuai dengan keluhan atau

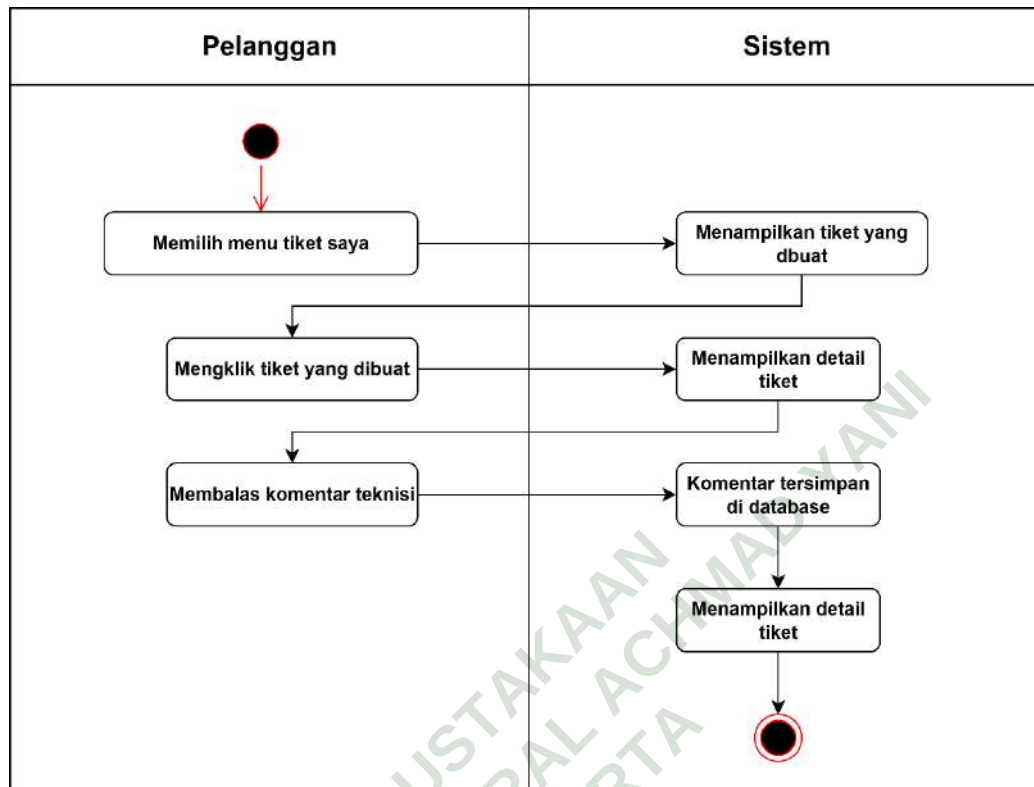
permasalahan yang dialami. Setelah *form* diisi dan dikirim, data akan tersimpan ke dalam *database* oleh sistem. Selanjutnya, sistem mengarahkan pelanggan ke menu tiket saya untuk melihat tiket yang telah dibuat sebagai konfirmasi bahwa laporan telah berhasil dikirim pada gambar 3.6.



Gambar 3.6 Activity Diagram Pelanggan Membuat tiket baru

3.4.7 Activity Diagram Pelanggan Memantau Tiket

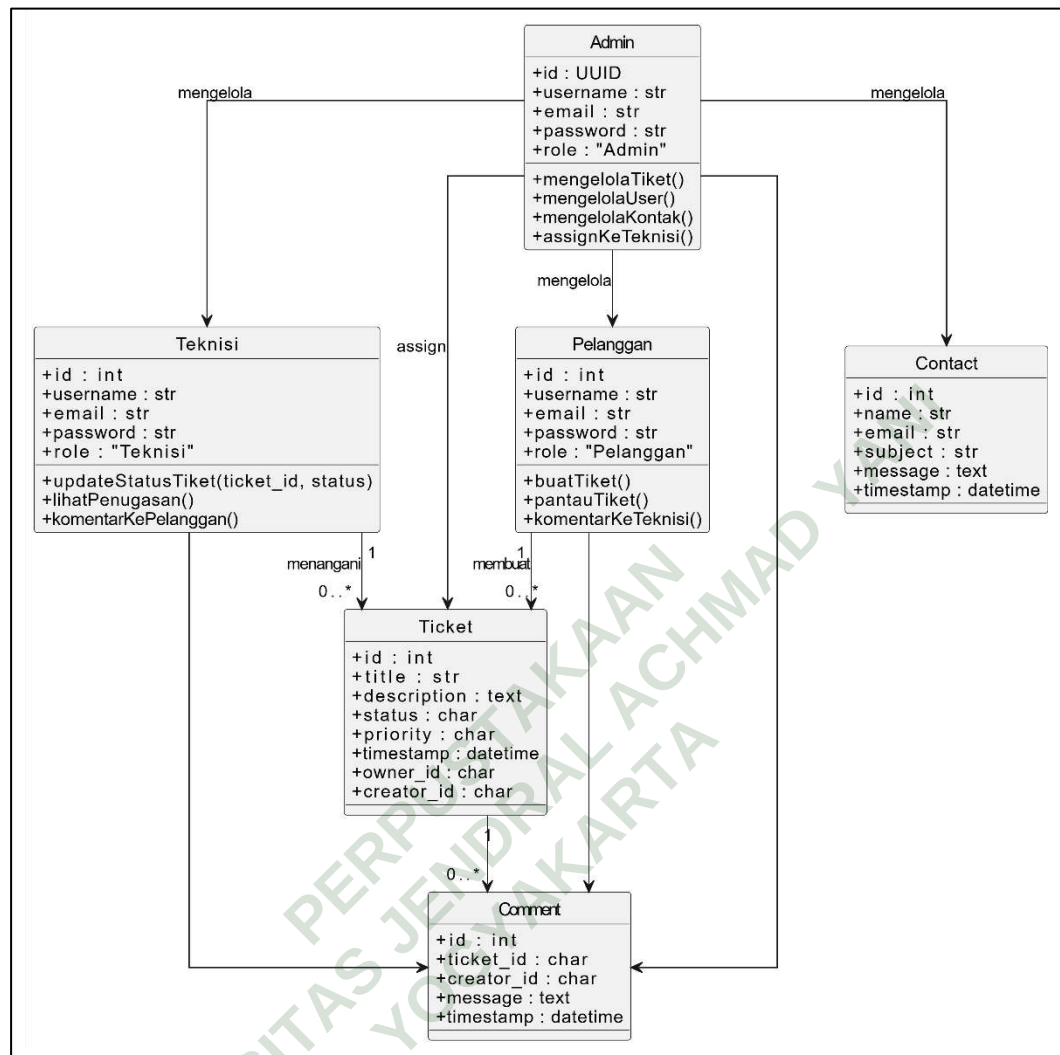
Activity diagram pelanggan memantau tiket menjelaskan alur interaksi pelanggan saat memantau perkembangan tiket yang telah dibuat. Proses dimulai ketika pelanggan memilih menu tiket saya, lalu sistem akan menampilkan daftar tiket yang telah dibuat oleh pelanggan. Pelanggan kemudian mengklik salah satu tiket untuk melihat detailnya, dan sistem menampilkan informasi lengkap tiket tersebut. Jika terdapat komentar dari teknisi, pelanggan dapat memberikan balasan komentar. Balasan tersebut akan disimpan oleh sistem ke dalam *database* dan detail tiket diperbarui serta ditampilkan kembali, sebagai bentuk umpan balik yang terus-menerus antara pelanggan dan teknisi pada gambar 3.7.



Gambar 3.7 Activity Diagram Pelanggan Membuat Tiket

3.4.8 Class Diagram

Class diagram pada gambar tersebut menggambarkan sistem manajemen tiket layanan yang melibatkan tiga peran utama: admin, teknisi, dan pelanggan. Admin memiliki hak untuk mengelola pengguna (teknisi dan pelanggan), mengelola tiket, mengelola kontak dari pengguna, serta menetapkan teknisi untuk menangani tiket. Pelanggan dapat membuat tiket, memantau statusnya, dan memberikan komentar kepada teknisi, sementara teknisi menerima penugasan tiket dari admin, memperbarui status tiket, serta memberi komentar balik kepada pelanggan. Setiap tiket dibuat oleh pelanggan dan ditangani oleh teknisi, serta dapat memiliki banyak komentar. Komentar sendiri memiliki relasi dengan tiket dan penciptanya (teknisi dan pelanggan). Selain itu, entitas *contact* menyimpan informasi dari pengguna yang mengirimkan pesan, yang juga dikelola oleh admin. Diagram ini memvisualisasikan alur interaksi antara pengguna dan sistem secara menyeluruh dengan fokus pada tanggung jawab masing-masing aktor.



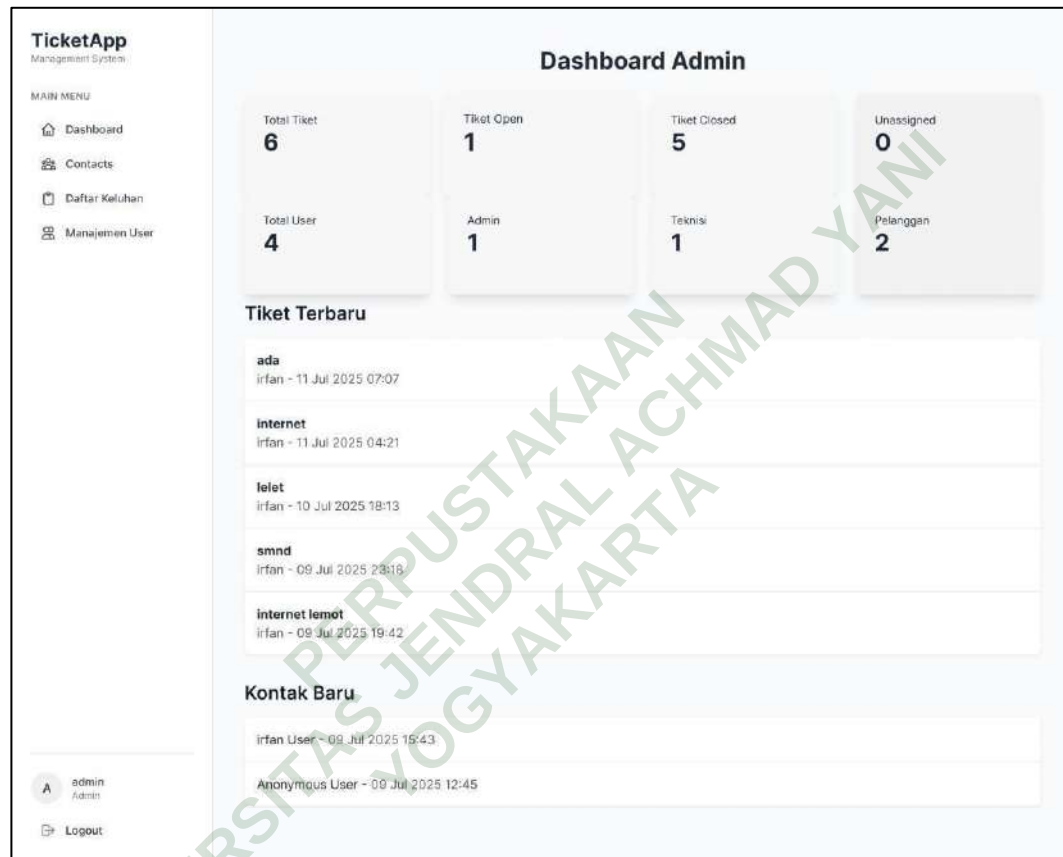
Gambar 3.8 Class Diagram

3.5 DESAIN INTERFACE

Desain antarmuka merupakan kegiatan merancang tampilan dan interaksi antara pengguna dengan sistem agar lebih mudah digunakan, efisien, dan nyaman. Mencakup prinsip seperti konsistensi desain, kemudahan membaca, navigasi yang terarah, serta pemanfaatan elemen visual seperti warna, ikon, dan susunan layout. Tujuannya adalah untuk memberikan pengalaman pengguna yang optimal dan mempermudah proses penggunaan aplikasi atau sistem.

3.5.1 Tampilan *Dashboard Admin*

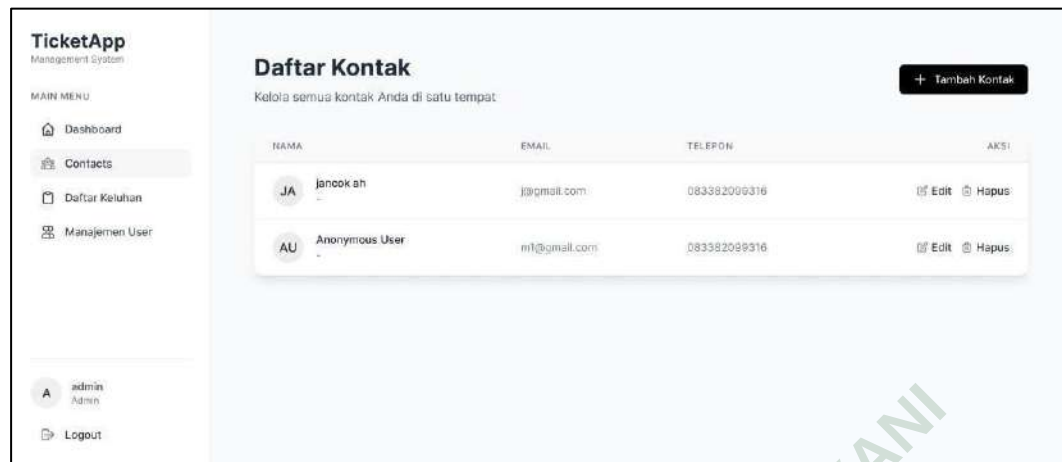
Dashboard admin menampilkan ringkasan jumlah tiket, pengguna, dan status tiket. Terdapat juga daftar tiket dan kontak terbaru serta menu navigasi untuk mengelola sistem.



Gambar 3.9 Tampilan *dashboard* admin

3.5.2 Tampilan Daftar Kontak Admin

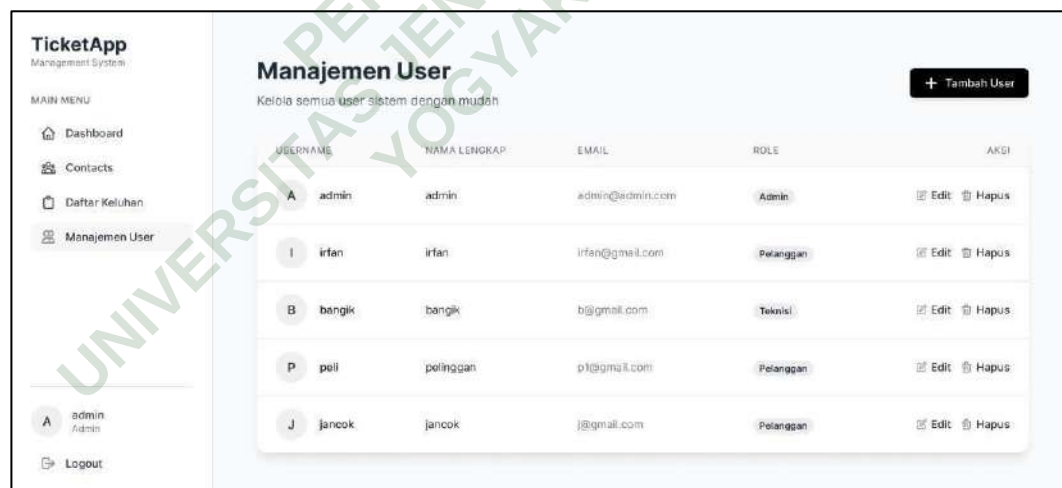
Tampilan daftar kontak admin menampilkan daftar kontak yang terdiri dari nama, *email*, dan nomor telepon. Admin dapat menambahkan kontak baru melalui tombol tambah kontak, serta melakukan aksi edit atau hapus pada setiap entri yang tersedia.



Gambar 3.10 Tampilan daftar tiket

3.5.3 Tampilan Manajemen User

Tampilan manajemen *user* menampilkan daftar seluruh pengguna sistem beserta informasi seperti *username*, nama lengkap, *email*, dan lain-lain (admin, teknisi, atau pelanggan). Admin dapat menambah pengguna baru melalui tombol tambah *user*, serta melakukan aksi edit dan hapus untuk setiap *user* yang terdaftar guna mempermudah pengelolaan akses sistem.

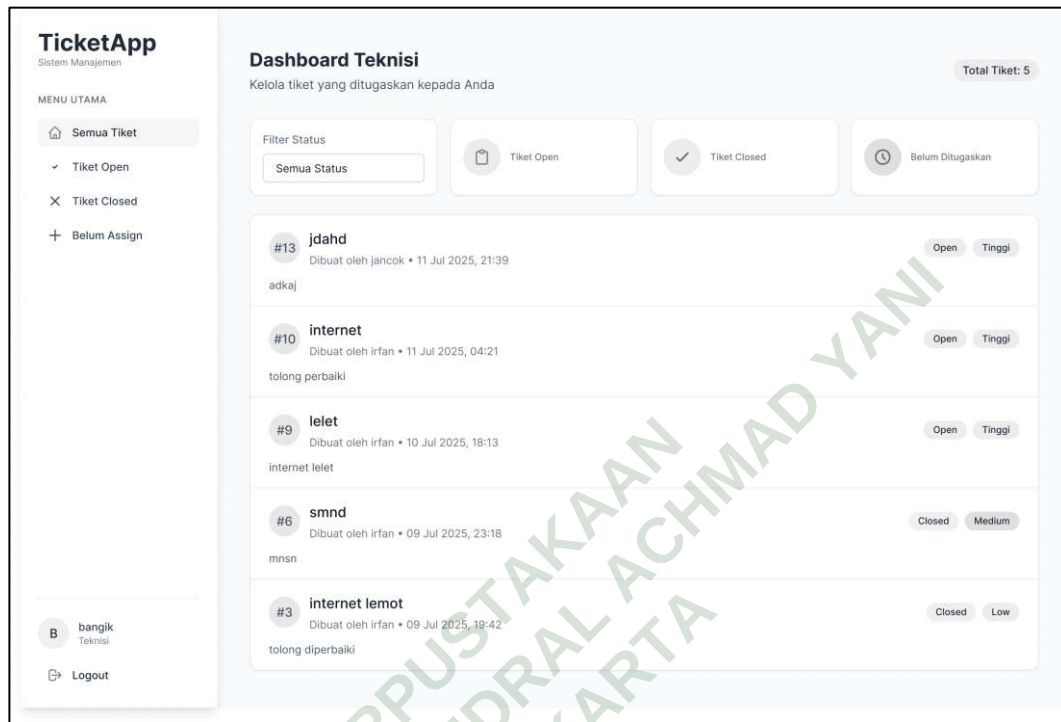


Gambar 3.11 Tampilan Manajemen User

3.5.4 Tampilan Dashboard Teknisi

Tampilan *dashboard* teknisi menyajikan daftar tiket yang telah dialokasikan kepada teknisi, disertai informasi seperti nama pembuat, waktu pengajuan, isi keluhan, status (*open/closed*), dan tingkat prioritas (tinggi, sedang, rendah). Teknisi

dapat menggunakan filter status seperti tiket *open*, tiket *closed*, atau belum ditugaskan untuk mempermudah pengaturan dan pemantauan tugas.



Gambar 3.12 Tampilan *Dashboard* Teknisi

3.5.5 Tampilan Detail Tiket Teknisi

Tampilan detail tiket teknisi menampilkan informasi lengkap mengenai tiket, seperti judul, status, prioritas, pembuat tiket, tanggal pembuatan, dan deskripsi keluhan. Selain itu, tersedia detail lokasi pelanggan, termasuk alamat, kota, provinsi, dan kode pos. Teknisi dapat menutup tiket dengan atau tanpa komentar melalui *form* yang disediakan, serta memberikan atau melihat komentar pada bagian bawah halaman untuk komunikasi tambahan terkait penyelesaian masalah.

TicketApp
Sistem Manajemen

MENU UTAMA

- Semua Tiket
- Tiket Open
- Tiket Closed
- Belum Assign

Detail Tiket

Judul: internet lemot
Status: Open
Prioritas: Medium
Dibuat oleh: jancok
Tanggal dibuat: 12 Jul 2025 17:33
Deskripsi: wifi jelek

Lokasi Pelanggan

Alamat: jl raya
Kota: yogya
Provinsi: yogyakarta
Kode Pos: 1231
Negara: indonesia
No. Telepon: None

Tutup Tiket

Tutup tanpa Komentar

Komentar Penutupan

Tulis komentar penutupan...

Tutup dengan Komentar

Komentar

Belum ada komentar.

Tambah Komentar

Tulis komentar Anda...

Kirim

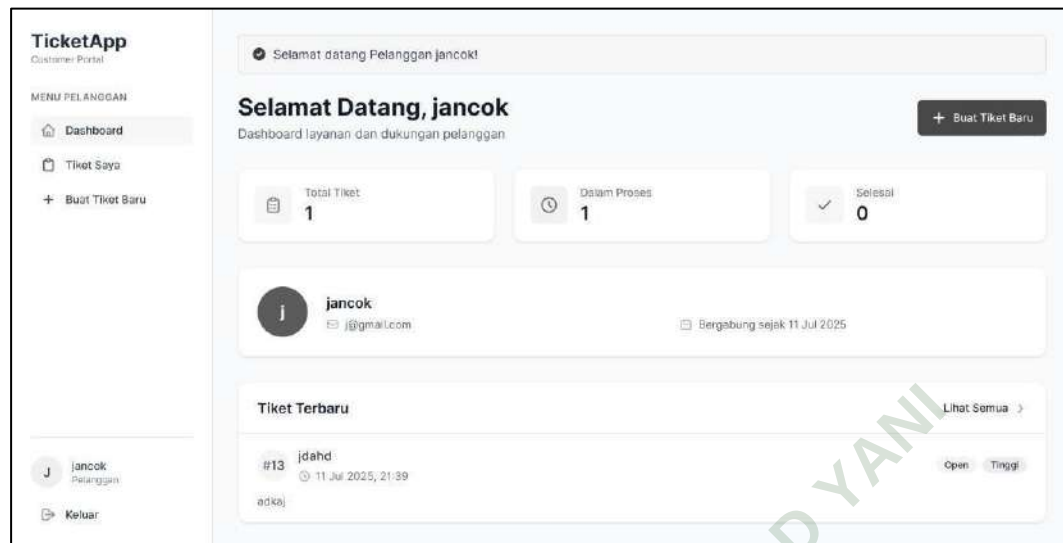
B bangik
Teknisi

Logout

Gambar 3.13 Tampilan Detail Teknisi

3.5.6 Tampilan *Dashboard* Pelanggan

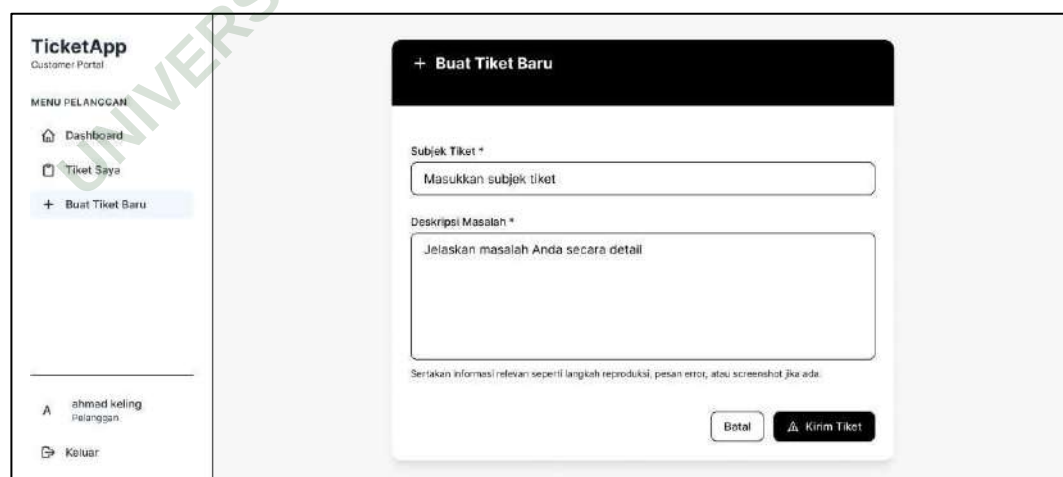
Dashboard pelanggan menampilkan ringkasan tiket yang diajukan, termasuk jumlah total, tiket dalam proses, dan tiket selesai. Pengguna dapat melihat informasi akun, daftar tiket terbaru, serta mengakses menu untuk melihat semua tiket atau membuat tiket baru. Fungsinya adalah memudahkan pelanggan memantau dan mengelola permintaan layanan.



Gambar 3.14 Tampilan Dashboar Pelanggan

3.5.7 Tampilan Buat Tiket Baru

Tampilan buat tiket baru memungkinkan pelanggan mengajukan permintaan bantuan dengan mengisi subjek tiket dan deskripsi masalah secara jelas dan detail. *Form* ini juga menyarankan pelanggan untuk menyertakan informasi tambahan seperti langkah reproduksi atau tangkapan layar. Tersedia dua tombol: batal untuk membatalkan pengisian, dan kirim tiket untuk mengirimkan laporan ke sistem. Tampilan ini dirancang sederhana agar pelanggan dapat melaporkan masalah dengan cepat dan mudah.



Gambar 3.15 Tampilan Buat Tiket Baru

3.5.8 Tampilan Detail Tiket *Close* Pelanggan

Tampilan detail tiket menampilkan informasi lengkap tiket yang telah diajukan pelanggan, seperti id tiket, waktu dibuat, subjek, deskripsi, status, dan prioritas. Jika status tiket sudah *closed*, akan muncul peringatan bahwa pelanggan tidak dapat lagi menambahkan komentar. Halaman ini memudahkan pelanggan untuk memantau perkembangan tiket dan mengetahui apakah tiket masih aktif atau sudah ditutup.



Gambar 3.16 Tampilan Detail Tiket *close* Pelanggan