

# **IMPLEMENTASI TEKNOLOGI PROGRESSIVE WEB APP (PWA) PADA SISTEM INFORMASI KOS-KOSAN**



Disusun Oleh:

N a m a : Nova Dwiyanto  
NIM : 20523211

**PROGRAM STUDI INFORMATIKA – PROGRAM SARJANA  
FAKULTAS TEKNOLOGI INDUSTRI  
UNIVERSITAS ISLAM INDONESIA  
2024**

HALAMAN PENGESAHAN DOSEN PEMBIMBING

**IMPLEMENTASI TEKNOLOGI PROGRESSIVE WEB APP  
(PWA) PADA SISTEM INFORMASI KOS-KOSAN**

**TUGAS AKHIR**



المعهد الإسلامي  
Yogyakarta, 27 Oktober 2024

Pembimbing,

( Irving Vitra Paputungan, S.T., M.Sc., Ph.D. )

## HALAMAN PENGESAHAN DOSEN PENGUJI

**IMPLEMENTASI TEKNOLOGI PROGRESSIVE WEB APP  
(PWA) PADA SISTEM INFORMASI KOS-KOSAN****TUGAS AKHIR**

Telah dipertahankan di depan sidang pengujian sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer dari Program Studi Informatika – Program Sarjana di Fakultas Teknologi Industri Universitas Islam Indonesia

Yogyakarta, 4 Nopember 2024

Tim Penguji

Irving Vitra Papatungan, S.T., M.Sc., Ph.D.



10/11/2024

**Anggota 1**

Beni Suranto, S.T., M.Soft.Eng.



**Anggota 2**

Kurniawan Dwi Irianto, S.T., M.Sc.

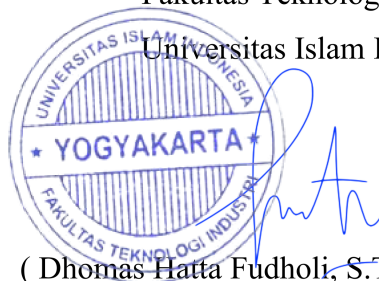


الجمعة المباركة  
Mengetahui,

Ketua Program Studi Informatika – Program Sarjana

Fakultas Teknologi Industri

Universitas Islam Indonesia



( Dhomas Hatta Fudholi, S.T., M.Eng., Ph.D. )

**HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR**

Yang bertanda tangan di bawah ini:

Nama : Nova Dwiyanto

NIM : 20523211

Tugas akhir dengan judul:

**IMPLEMENTASI TEKNOLOGI PROGRESSIVE WEB APP  
(PWA) PADA SISTEM INFORMASI KOS-KOSAN**

Menyatakan bahwa seluruh komponen dan isi dalam tugas akhir ini adalah hasil karya saya sendiri. Apabila di kemudian hari terbukti ada beberapa bagian dari karya ini adalah bukan hasil karya sendiri, tugas akhir yang diajukan sebagai hasil karya sendiri ini siap ditarik kembali dan siap menanggung risiko dan konsekuensi apapun.

Demikian surat pernyataan ini dibuat, semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta, 22 Oktober 2024



METERAI  
TEMPEL  
10000  
72AALX384856086

( Nova Dwiyanto )

## HALAMAN PERSEMBAHAN

Teruntuk kedua orang tua saya, saya mengucapkan terima kasih yang mendalam. Terima kasih atas dukungan, kasih sayang, dan pengorbanan yang tiada henti. Doa dan ridho kalian senantiasa menguatkan langkah saya. Setiap nilai yang kalian tanamkan dan setiap nasihat yang kalian berikan menjadi cahaya dalam perjalanan hidup saya. Saya berjanji untuk selalu menghargai segala usaha dan cinta yang telah kalian curahkan demi masa depan saya. Tanpa bimbingan dan kasih sayang kalian, saya tidak akan berada di titik ini. Semoga saya dapat membanggakan kalian dan meneruskan warisan nilai-nilai yang telah kalian ajarkan.

Teruntuk para dosen dan pengajar, saya mengucapkan terima kasih atas bimbingan dan ilmu yang sangat berharga yang telah kalian berikan. Terima kasih pula atas dedikasi dan inspirasi yang telah membantu membentuk pemikiran dan karakter saya. Setiap pelajaran yang disampaikan memiliki makna yang mendalam dan menjadi bekal untuk masa depan.

Teruntuk semua teman dan sahabat, saya mengucapkan terima kasih atas kebersamaan dan dukungan yang telah kalian berikan selama masa perkuliahan. Kalian telah menjadi bagian penting dalam perjalanan ini, dengan setiap tawa, cerita, dan momen berharga yang telah kita bagi. Setiap interaksi dan kenangan bersama kalian telah memberikan warna dan makna dalam hidup saya.

Terakhir, saya mengucapkan terima kasih kepada diri saya sendiri. Terima kasih telah berjuang, tetap berkomitmen, dan tidak menyerah meskipun banyak rintangan yang harus dihadapi. Setiap usaha yang dilakukan selama ini membawa kita kepada pencapaian yang ada saat ini. Semoga kita terus berkembang dan menjadi pribadi yang lebih baik di masa mendatang.

## HALAMAN MOTO

"Tantangan dan masalah dalam hidup adalah kesempatan untuk tumbuh."

"Jika melakukan sesuatu yang positif terasa sulit, mulailah dengan menghilangkan kebiasaan negatif terlebih dahulu."

"Disiplin berarti memaksa diri sendiri. Disiplin adalah jembatan antara tujuan dan pencapaian."

## KATA PENGANTAR

Puji syukur peneliti panjatkan kepada Allah SWT atas segala kesehatan dan nikmat yang telah diberikan, sehingga peneliti dapat menyelesaikan skripsi ini dengan baik. Atas izin-Nya, peneliti mampu menghadapi berbagai tantangan selama proses pengerjaan skripsi ini. Dalam penyusunan skripsi ini, peneliti mendapat banyak bantuan, dukungan, dan bimbingan dari berbagai pihak. Oleh karena itu, peneliti ingin mengucapkan terima kasih kepada:

1. Kedua orang tua dan kakak peneliti yang telah memberikan doa dan dukungan tanpa henti
2. Bapak Prof. Fathul Wahid, S.T., M.Sc., Ph.D. selaku Rektor Universitas Islam Indonesia
3. Bapak Dr. Raden Teduh Dirgahayu, S.T., M.Sc. selaku Ketua Program Studi Informatika Universitas Islam Indonesia
4. Bapak Irving Vitra Paputungan, S.T., M.Sc., Ph.D. selaku dosen pembimbing yang telah meluangkan waktu dan memberikan arahan hingga skripsi ini selesai

Meskipun tugas akhir ini masih memiliki banyak kekurangan karena keterbatasan ilmu dan pengalaman yang peneliti miliki, peneliti berharap skripsi ini dapat bermanfaat bagi yang membutuhkan.

Yogyakarta, 24 Oktober 2024



( Nova Dwiyanto )

## SARI

Kos-kosan adalah layanan penyediaan tempat tinggal yang menawarkan kamar dengan pembayaran sewa yang fleksibel. Namun, transaksi pada kebanyakan kos-kosan masih dilakukan secara manual, di mana penghuni harus mengirimkan bukti pembayaran kepada pemilik dan mengonfirmasi bahwa mereka telah membayar. Hal ini menyulitkan penghuni dalam memastikan bahwa pembayaran mereka diterima, dan pemilik juga harus mencocokkan bukti bayar dengan transaksi yang tercatat. Situasi ini tidak efisien dan dapat menimbulkan kerumitan administratif serta risiko kesalahan pencatatan. Oleh karena itu, dibutuhkan sistem informasi manajemen kos-kosan yang dapat mempermudah proses ini. Pangsa pasar ponsel secara global telah naik dari 0,67% pada Januari 2009 menjadi 47,96% pada Februari 2019. Hal ini mendorong kebutuhan akan aplikasi mobile dan situs web yang responsif untuk perangkat mobile. Namun, pengembangan perangkat mobile membutuhkan biaya yang lebih tinggi dibandingkan pengembangan sistem berbasis web. Artikel ini bertujuan untuk membuat sistem manajemen kos-kosan menggunakan teknologi Progressive Web App (PWA) yang dapat menyederhanakan proses pembayaran dan pencatatan keuangan serta dapat diinstal seperti aplikasi native, namun dengan langkah instalasi yang lebih mudah. Sistem yang dikembangkan berhasil mengotomatisasi pembayaran digital tanpa verifikasi manual, menghasilkan tagihan otomatis, dan mengirimkan pemberitahuan kepada penghuni. Implementasi PWA meningkatkan pengalaman pengguna dengan menyediakan akses cepat dan mudah melalui berbagai perangkat. Dengan demikian, proses transaksi pembayaran kos dan pencatatan transaksi menjadi lebih efisien, serta dapat meningkatkan kepuasan pengguna melalui aplikasi web yang memberikan pengalaman serupa aplikasi mobile.

Kata kunci: Kos-kosan, Sistem Informasi, Progressive Web App (PWA)

## GLOSARIUM

<i>Activity diagram</i>	Diagram yang menggambarkan alur kerja atau aktivitas dari sebuah sistem atau proses bisnis
<i>Blackbox</i>	Metode pengujian perangkat lunak yang berfokus pada pengujian fungsional tanpa melihat kode program
<i>Database</i>	Kumpulan data yang tersimpan secara sistematis dalam komputer
<i>Framework</i>	Kerangka kerja yang berisi kumpulan fungsi dasar untuk memudahkan pembuatan aplikasi
<i>Laravel</i>	Framework PHP yang memudahkan pengembangan aplikasi web dengan fitur-fitur modern
<i>Mobile first</i>	Pendekatan desain website yang mengutamakan tampilan pada perangkat mobile sebelum desktop
<i>MySQL</i>	Sistem manajemen basis data yang menggunakan bahasa SQL untuk mengelola data
<i>PHP</i>	Bahasa pemrograman yang digunakan untuk membuat website dinamis
<i>UML</i>	Unified Modeling Language - Bahasa pemodelan standar untuk menggambarkan struktur dan alur sistem perangkat lunak
<i>Use case diagram</i>	Diagram yang menggambarkan interaksi antara pengguna dengan sistem dan fungsi-fungsi yang tersedia
<i>Waterfall</i>	Model pengembangan sistem yang berjalan secara berurutan seperti air terjun, mulai dari analisis hingga pemeliharaan

## DAFTAR ISI

HALAMAN JUDUL .....	i
HALAMAN PENGESAHAN DOSEN PEMBIMBING.....	ii
HALAMAN PENGESAHAN DOSEN PENGUJI.....	iii
HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR.....	iv
HALAMAN PERSEMBAHAN .....	v
HALAMAN MOTO .....	vi
KATA PENGANTAR .....	vii
SARI.....	viii
GLOSARIUM.....	ix
DAFTAR ISI.....	x
DAFTAR TABEL.....	xii
DAFTAR GAMBAR .....	xiii
<b>BAB I PENDAHULUAN.....</b>	<b>1</b>
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah.....	4
1.3 Batasan Masalah .....	5
1.4 Tujuan Penelitian .....	5
1.5 Manfaat Penelitian .....	5
1.6 Metodologi Penelitian.....	5
1.7 Sistematika Penulisan.....	6
<b>BAB II TINJAUAN PUSTAKA .....</b>	<b>8</b>
2.1 Sistem Informasi Manajemen Kos-kosan .....	8
2.2 Metodologi Pengembangan Sistem Informasi .....	9
2.3 Metode Waterfall .....	11
2.4 Progressive Web App .....	12
2.5 Blackbox Testing .....	16
<b>BAB III METODOLOGI DAN PERANCANGAN SISTEM.....</b>	<b>18</b>
3.1 Analisis Kebutuhan.....	18
3.1.1 Pengumpulan Data.....	18
3.1.2 Analisis Kebutuhan Fungsional .....	20
3.1.3 Analisis Kebutuhan Non-fungsional.....	22
3.1.4 Identifikasi Pengguna.....	22

3.2	Perancangan Sistem .....	22
3.2.1	Use Case Diagram.....	23
3.2.2	Activity Diagram .....	25
3.2.3	Perancangan Progressive Web App.....	40
3.2.4	Perancangan Inteface .....	40
3.2.5	Perancangan Database .....	51
BAB IV IMPLEMENTASI HASIL DAN ANALISIS.....		55
4.1	Hasil Penelitian .....	55
4.1.1	Implementasi Progeressive Web App.....	55
4.1.2	Implementasi Sistem Informasi Antarmuka Admin .....	57
4.1.3	Implementasi Sistem Informasi Antarmuka Pengguna.....	76
4.1.4	Pengujian .....	82
4.2	Kelebihan dan Kekurangan Sistem.....	87
BAB V KESIMPULAN DAN SARAN .....		89
5.1	Kesimpulan .....	89
5.2	Saran.....	89
DAFTAR PUSTAKA.....		91
LAMPIRAN.....		95

**DAFTAR TABEL**

Tabel 2.1 Perbandingan penelitian sejenis.....	9
Tabel 2.2 Pengujian Sistem.....	16
Tabel 3.1 Sumber terkait.....	19
Tabel 3.2 Use Case Diagram sistem .....	24
Tabel 3.3 Users .....	52
Tabel 3.4 Rooms .....	53
Tabel 3.5 Room types .....	53
Tabel 3.6 Photos.....	53
Tabel 3.7 Transactions .....	53
Tabel 3.8 Messages .....	54
Tabel 3.9 Group accesses.....	54
Tabel 4.1 Pengujian Fungsional.....	83
Tabel 4.2 Responden pengujian fungsional .....	84
Tabel 4.3 Hasil kuesioner pengujian fungsional .....	84
Tabel 4.4 Pengujian non-fungsional .....	85
Tabel 4.5 Responden pengujian non-fungsional.....	86
Tabel 4.6 Hasil kuesioner pengujian non-fungsional .....	86
Tabel 4.7 Hasil pengujian non-fungsional .....	87

## DAFTAR GAMBAR

Gambar 2.1 Metode waterfall .....	11
Gambar 3.1 Metode waterfall .....	18
Gambar 3.2 Use Case Diagram sistem .....	24
Gambar 3.3 Mengelola data pnegguna .....	26
Gambar 3.4 Mengelola data tipe kamar .....	28
Gambar 3.5 Mengelola data kamar .....	30
Gambar 3.6 Generate tagihan otomatis.....	32
Gambar 3.7 Membuat dan memperbarui tagihan .....	33
Gambar 3.8 Melihat laporan keuangan .....	34
Gambar 3.9 Melihat daftar tagihan dan menghapus tagihan .....	35
Gambar 3.10 Melakukan obrolan .....	36
Gambar 3.11 Melihat daftar kamar .....	37
Gambar 3.12 Melihat tagihan .....	38
Gambar 3.13 Membayar tagihan.....	39
Gambar 3.14 Mobile .....	41
Gambar 3.15 Desktop .....	41
Gambar 3.16 Dashboard .....	44
Gambar 3.17 Daftar kamar .....	44
Gambar 3.18 Tambah kamar.....	44
Gambar 3.19 Detail kamar .....	44
Gambar 3.20 Tipe kamar .....	44
Gambar 3.21 Tambah dan perbarui tipe kamar .....	44
Gambar 3.22 Keuangan .....	45
Gambar 3.23 Status Pembayaran .....	45
Gambar 3.24 Detail Status Pembayaran .....	45
Gambar 3.25 Keuangan .....	45
Gambar 3.26 Transaksi .....	45
Gambar 3.27 Detail Transaksi .....	45
Gambar 3.28 Buat tagihan .....	46
Gambar 3.29 Perbarui tagihan .....	46
Gambar 3.30 Menu chat.....	46
Gambar 3.31 Chat .....	47

Gambar 3.32 Pengguna.....	47
Gambar 3.33 Tambah dan perbarui pengguna.....	47
Gambar 3.34 Detail pengguna.....	48
Gambar 3.35 Admin.....	48
Gambar 3.36 Tambah dan perbarui admin.....	48
Gambar 3.37 Profil.....	48
Gambar 3.38 Perbarui sistem.....	48
Gambar 3.39 Home.....	50
Gambar 3.40 Daftar tipe kamar.....	50
Gambar 3.41 Detail kamar.....	50
Gambar 3.42 Tagihan.....	50
Gambar 3.43 Detail tagihan.....	50
Gambar 3.44 Chat.....	50
Gambar 3.45 Profil.....	51
Gambar 3.46 Relasi antar tabel.....	52
Gambar 4.1 Sistem pada perangkat desktop.....	56
Gambar 4.2 Sistem pada perangkat mobile.....	56
Gambar 4.3 Halaman dashboard 1.....	57
Gambar 4.4 Halaman dashboard 2.....	57
Gambar 4.5 Halaman kamar.....	58
Gambar 4.6 Tambah kamar.....	58
Gambar 4.7 Perbarui data kamar.....	59
Gambar 4.8 Detail kamar.....	59
Gambar 4.9 Halaman tipe kamar.....	60
Gambar 4.10 Halaman tipe kamar.....	61
Gambar 4.11 Perbarui data tipe kamar.....	61
Gambar 4.12 Halaman keuangan 1.....	62
Gambar 4.13 Halaman keuangan 2.....	62
Gambar 4.14 Halaman status pembayaran.....	64
Gambar 4.15 Halaman status pembayaran detail.....	64
Gambar 4.16 Halaman laporan keuangan bulanan.....	65
Gambar 4.17 Halaman laporan keuangan tahunan.....	65
Gambar 4.18 Halaman daftar transaksi 1.....	66
Gambar 4.19 Halaman daftar transaksi 2.....	66

Gambar 4.20 Halaman buat tagihan.....	67
Gambar 4.21 Halaman perbarui tagihan .....	67
Gambar 4.22 Halaman menu chat.....	68
Gambar 4.23 Halaman chat.....	69
Gambar 4.24 Halaman pengguna aktif .....	70
Gambar 4.25 Halaman pengguna tidak aktif .....	70
Gambar 4.26 Tambah pengguna .....	70
Gambar 4.27 Perbarui data pengguna .....	70
Gambar 4.28 Detail pengguna .....	71
Gambar 4.29 Halaman admin .....	72
Gambar 4.30 Halaman admin (detail).....	72
Gambar 4.31 Halaman admin (perbarui) .....	72
Gambar 4.32 Halaman profil .....	74
Gambar 4.33 Halaman profil telepon.....	74
Gambar 4.34 Halaman profil tagihan.....	74
Gambar 4.35 Halaman profil password .....	74
Gambar 4.36 Halaman perbarui data sistem .....	75
Gambar 4.37 Halaman home .....	76
Gambar 4.38 Halaman tipe kamar .....	77
Gambar 4.39 Halaman detail kamar .....	78
Gambar 4.40 Halaman tagihan .....	79
Gambar 4.41 Halaman tagihan detail.....	79
Gambar 4.42 Halaman chat group .....	80
Gambar 4.43 Halaman chat admin.....	80
Gambar 4.44 Halaman profil .....	81
Gambar 4.45 Halaman profil (ubah telepon) .....	81
Gambar 4.46 Halaman Profil (ubah password).....	81

## **BAB I**

### **PENDAHULUAN**

#### **1.1 Latar Belakang**

Kos merupakan layanan penyediaan tempat tinggal yang menawarkan kamar atau ruangan untuk dihuni dengan membayar sejumlah biaya tertentu untuk setiap periode waktu tertentu. Kos-kosan adalah solusi yang sangat fleksibel dan sering kali dipilih oleh mahasiswa yang sedang mengejar pendidikan tinggi, pekerja muda yang sedang memulai karier, atau individu yang hanya membutuhkan tempat tinggal sementara. Menurut Rahmawati & Harahap (2021), rumah kos merupakan salah satu kebutuhan utama bagi mahasiswa yang sedang menempuh pendidikan di daerah yang berbeda dari kampung halaman mereka. Mahasiswa umumnya memandang rumah kos sebagai solusi utama. Karena itu, tidak mengherankan bahwa banyak ditemukan tempat kos di sekitar area dekat kampus atau universitas. Berdasarkan data yang dikeluarkan oleh Kementerian Pendidikan, Kebudayaan, Riset, dan Teknologi (Kemendikbud Ristek), pada tahun 2022, jumlah mahasiswa di Indonesia mencapai 9,32 juta orang, mengalami peningkatan sebanyak 4,02% dibandingkan dengan tahun sebelumnya yang mencatatkan 8,96 juta orang. Fasilitas yang ditawarkan dalam kos-kosan dapat bervariasi, mulai dari kamar tidur sederhana hingga unit dengan fasilitas lengkap seperti kamar mandi dalam, dapur bersama, dan ruang bersantai. Selain di wilayah dekat kampus, Kos-kosan cenderung ditemukan di lokasi-lokasi strategis seperti pusat kota, atau wilayah yang dekat dengan tempat-tempat kerja. Keberadaan kos-kosan memberikan alternatif yang lebih terjangkau dibandingkan dengan sewa apartemen atau rumah, dan sering kali tanpa perlu terikat dalam kontrak sewa jangka panjang. Dengan demikian, kos-kosan memainkan peran penting dalam memenuhi kebutuhan berbagai individu dalam masyarakat.

Menurut informasi dari Bank Indonesia (BI), tercatat bahwa pada bulan April 2023, volume transaksi uang elektronik mencapai Rp37,46 triliun. Pertumbuhan ini mencapai 1.017%, atau meningkat hampir sebelas kali lipat dibandingkan dengan April 2018 yang sebesar 3,49 triliun. Dengan merujuk pada informasi tersebut, saat ini transaksi menggunakan pembayaran digital atau *e-wallet* telah meningkat signifikan, termasuk dalam pembayaran yang umum dilakukan oleh penghuni kos. Ini disebabkan oleh kenyamanan yang ditawarkan oleh pembayaran digital atau *e-wallet*. Menurut Fadhilah dkk. (2021), sistem pembayaran digital memiliki berbagai manfaat, termasuk efisiensi waktu karena memungkinkan transaksi di berbagai lokasi, kenyamanan dengan berbagai metode pembayaran online, serta pengendalian

biaya melalui pencatatan otomatis yang akurat dan biaya transaksi yang lebih rendah. Meskipun demikian, pembayaran kos menggunakan *e-wallet* masih memerlukan langkah tambahan di mana penghuni kos harus secara manual mengirimkan bukti pembayaran kepada pemilik kos setelah melakukan pembayaran. Hal ini membutuhkan usaha tambahan dan berpotensi menimbulkan ketidaknyamanan. Menurut Rosalina Widyayanti (2020), Sistem pembayaran Non-Tunai dapat mencatat transaksi non-digital menjadi digital, secara rinci semua transaksi akan terekam secara otomatis di dalam sistem, mulai dari tanggal transaksi dan waktu transaksi, jenis produk atau layanan, jumlah modal yang dikeluarkan, serta jumlah pembayaran dari pelanggan. Semua data ini otomatis tercatat dalam sistem untuk memberikan gambaran yang komprehensif mengenai setiap transaksi. Namun, dari perspektif pemilik kos, seringkali mereka menghadapi kesulitan dalam membedakan antara penghuni yang sudah membayar dengan yang belum, serta mencocokkan bukti bayar dengan transaksi yang telah masuk. Situasi ini menciptakan banyak kendala dalam manajemen kos-kosan, termasuk kerumitan administratif, risiko kesalahan pencatatan, dan keterlambatan dalam pelacakan pembayaran.

Dalam era digital saat ini, di mana teknologi telah mengubah cara kita melakukan banyak hal. diperlukan sebuah solusi yang modern dan efisien untuk manajemen kos-kosan, yang akan memudahkan penghuni dalam melakukan pembayaran, mengoptimalkan proses administrasi bagi pemilik kos, dan meningkatkan kenyamanan dan kemudahan bagi semua pihak yang terlibat. Menurut Nitami, Munthe, & Masrizal (2021), Sistem informasi adalah sekumpulan komponen informasi yang saling berhubungan saling mengumpulkan atau mendapatkan, memproses, menyimpan, dan mendistribusikan informasi. Komponen-komponen tersebut saling terhubung dan berfungsi secara bersinergi dalam sistem informasi. Dengan sistem informasi, pemilik kos dapat mengotomatisasi banyak aspek dalam operasional manajemen kos-kosan, termasuk pencatatan pembayaran dan pelacakan penghuni. Sistem informasi yang terintegrasi dengan payment gateway memungkinkan penghuni untuk membayar sewa mereka secara online tanpa harus mengirim bukti pembayaran fisik. Hal ini tidak hanya mengurangi beban administratif pemilik kos, tetapi juga memberikan kemudahan dan kenyamanan kepada penghuni. Menurut Maysaroh & Diansyah (2022), payment gateway adalah layanan yang berfungsi untuk menerima pembayaran dalam transaksi digital. Layanan ini beroperasi secara otomatis, memudahkan proses transaksi, serta memastikan semua transaksi keuangan diproses dengan aman dan real-time. Payment gateway memungkinkan sistem informasi untuk membuat tagihan kepada setiap pengguna. Hal ini memungkinkan sistem informasi untuk memantau

pengguna yang telah melakukan pembayaran dan yang belum. Selain itu, sistem informasi dapat membantu dalam mengelola kontrak sewa dan mengingatkan tentang tanggal jatuh tempo. Semua ini berkontribusi pada meningkatkan efisiensi operasional dan memberikan pengalaman yang lebih baik kepada pemilik kos dan penghuni. Dengan demikian, sistem informasi kos-kosan adalah langkah progresif dalam menghadapi perubahan teknologi dan memaksimalkan efisiensi dalam manajemen kos-kosan.

Menurut Tahirshah (2019), antara tahun 2009 hingga 2018, terjadi peningkatan yang signifikan dalam persentase halaman web global yang diakses melalui perangkat ponsel, dari 0,7% menjadi 52,2%. Peningkatan ini sejalan dengan pertumbuhan pangsa pasar ponsel secara global, yang naik dari 0,67% pada Januari 2009 menjadi 47,96% pada Februari 2019. Dengan meningkatnya penggunaan ponsel pintar dan akses internet melalui perangkat seluler, perusahaan mulai memberikan perhatian lebih besar terhadap kebutuhan pengguna seluler. Hal ini terlihat dari pengembangan aplikasi mobile native dan perancangan situs web yang responsif untuk perangkat mobile. Namun, pengembangan aplikasi mobile seringkali memerlukan sumber daya yang lebih besar, baik dari segi waktu maupun biaya. Proses pengembangan ini mencakup beberapa aspek, seperti pengembangan berbasis platform, pemeliharaan, dan pembaruan yang berkelanjutan.

Parbat Zait dkk. (2018) menyatakan bahwa pengguna aplikasi harus melalui beberapa tahapan, seperti pendaftaran di toko aplikasi, memeriksa kapasitas penyimpanan, mengunduh, menyelesaikan instalasi, dan akhirnya membuka aplikasi. Penelitian menunjukkan bahwa, rata-rata, setiap aplikasi kehilangan sekitar 20% pengguna di setiap tahap dari interaksi awal hingga saat pengguna benar-benar mulai menggunakannya. Hal ini menunjukkan bahwa banyak pengguna merasa proses ini terlalu rumit, yang dapat berdampak negatif bagi perusahaan dan pengembang aplikasi.

Progressive Web App (PWA) muncul sebagai solusi efektif untuk mengatasi tantangan dalam pengembangan aplikasi mobile. PWA adalah jenis aplikasi web yang memanfaatkan teknologi terkini untuk menciptakan pengalaman pengguna yang mirip dengan aplikasi native. Salah satu keunggulan PWA adalah kemampuannya untuk diakses melalui browser web dan dapat diinstal langsung dari browser dengan proses instalasi yang mudah, sehingga memungkinkan pengembang untuk mengerjakan aplikasi sekali saja dan dapat dijalankan di berbagai platform. Ini secara signifikan dapat mengurangi biaya pengembangan dan pemeliharaan. Selain itu, PWA dirancang untuk memberikan kinerja optimal dan responsif, dengan fitur seperti mode offline, notifikasi push, dan kemampuan untuk ditempatkan di layar

utama perangkat. Dengan demikian, PWA menjadi pilihan yang menarik untuk menyajikan aplikasi secara efisien dan memberikan pengalaman pengguna yang unggul di berbagai perangkat. Menurut Nugraha, Anjara, & Faizah (2022), “PWA merupakan salah satu metode pengembangan perangkat lunak yang canggih dengan mengintegrasikan aplikasi native, aplikasi web, dan desktop. Pendekatan ini memanfaatkan teknologi terkini dan fitur-fitur modern browser untuk menciptakan aplikasi yang mirip dengan aplikasi asli. PWA dapat dianggap sebagai kumpulan teknologi, konsep desain, dan prinsip-prinsip yang menyatukan kelebihan dari berbagai jenis aplikasi. Beberapa perusahaan besar seperti AliExpress, BookMyShow, Forbes, Pinterest, dan Flipkart telah sukses menerapkan PWA. Keberhasilan ini menunjukkan bahwa pengembangan PWA sangat menjanjikan dan dapat diterapkan di berbagai segmen industri.”

## 1.2 Rumusan Masalah

Terdapat beberapa kendala dalam proses transaksi pada kos-kosan. Meskipun sebagian besar pembayaran kos saat ini sudah menggunakan pembayaran digital atau *e-wallet*, masih diperlukan langkah tambahan di mana penghuni kos harus secara manual mengirimkan bukti pembayaran kepada pemilik kos setelah melakukan pembayaran. Selain itu, pemilik kos harus mencocokkan bukti bayar yang dikirim oleh penghuni kos, hal ini menciptakan banyak kendala dalam manajemen kos-kosan, termasuk kerumitan administratif, risiko kesalahan pencatatan, dan keterlambatan dalam pelacakan pembayaran.

Pengembangan aplikasi mobile seringkali memerlukan sumber daya yang lebih besar, baik dalam hal waktu maupun biaya. Sebuah studi juga menunjukkan bahwa, secara rata-rata, setiap aplikasi kehilangan sekitar 20% pengguna karena memerlukan serangkaian langkah untuk dapat menggunakan aplikasi tersebut.

Dari permasalahan tersebut, diangkat beberapa pertanyaan penelitian, yaitu:

- a. Bagaimana cara mempermudah transaksi pembayaran kos?
- b. Bagaimana membantu pemilik kos untuk mempermudah pencatatan transaksi pembayaran kos?
- c. Bagaimana membangun sebuah sistem informasi berbasis web yang memberikan pengalaman pengguna yang mirip dengan aplikasi native dan dapat digunakan tanpa melewati langkah yang panjang agar tidak kehilangan pengguna?

### **1.3 Batasan Masalah**

Berdasarkan latar belakang dan rumusan masalah yang sudah diuraikan di atas. Adapun batasan masalah yang akan diuraikan meliputi:

- a. Penelitian ini berfokus pada permasalahan manajemen pembayaran kos-kosan dalam konteks meningkatnya penggunaan pembayaran digital di Indonesia.
- b. Masalah yang diteliti terbatas pada kesulitan dalam verifikasi dan pelacakan pembayaran kos, baik dari sisi penghuni maupun pemilik kos.

### **1.4 Tujuan Penelitian**

Tujuan dari penelitian ini adalah untuk membangun Sistem Informasi berbasis web dengan teknologi PWA yang mendukung proses pembayaran dan pencatatan keuangan, serta memberikan pengalaman pengguna yang mirip dengan aplikasi native dan dapat digunakan tanpa melewati langkah yang panjang agar tidak kehilangan pengguna.

### **1.5 Manfaat Penelitian**

Dengan adanya penelitian ini diharapkan dapat memberikan manfaat sebagai berikut:

- a. Mempermudah proses transaksi pembayaran kos.
- b. Mempermudah pencatatan transaksi.

### **1.6 Metodologi Penelitian**

#### **a. Pengumpulan data**

Tahap Pengumpulan data dilakukan pengumpulan data yang diperlukan oleh sistem melalui dua metode, observasi dan studi literatur. Observasi melibatkan pengumpulan informasi dari kegiatan pembayaran yang umumnya dilakukan oleh penghuni kos. Sementara itu, Studi literatur melibatkan pencarian sumber informasi dengan penelusuran literatur sebelumnya yang relevan dengan tema yang sama. Proses pengambilan data dilakukan hingga memperoleh data atau informasi yang cukup sebelum masuk ke dalam tahap perancangan sistem. Data yang diperoleh akan diolah agar dapat menjadi dasar pengembangan sistem informasi yang sedang dibuat.

#### b. Perancangan sistem

Dalam pengembangan sistem, perlu adanya desain yang optimal untuk memastikan bahwa sistem yang dibuat memiliki dokumentasi yang lengkap dan memenuhi kebutuhan pengguna. Dalam penelitian ini, perancangan dibagi menjadi beberapa bagian yaitu perancangan *use case diagram*, *Activity Diagram*, perancangan antarmuka, dan Entity Relationship Diagram (ERD). Hal ini dilakukan untuk memastikan bahwa setiap aspek sistem direncanakan dengan baik agar sesuai dengan harapan pengguna.

#### c. Implementasi sistem

Implementasi merupakan tahap dimana sistem informasi yang telah direncanakan dibangun sesuai dengan perencanaan sebelumnya. Pada tahap ini, dilakukan perubahan hasil perancangan sistem menjadi suatu sistem informasi. Sistem informasi dalam skripsi ini memanfaatkan Bahasa pemrograman *PHP* dengan *framework Laravel* dan menggunakan *database MySQL*. Sistem informasi yang akan dibangun juga mengintegrasikan Midtrans sebagai payment gateway dan mengimplementasikan Progressive Web App (PWA).

#### d. Pengujian dan Evaluasi sistem

Tahap terakhir dari penelitian ini melibatkan pengujian dan evaluasi sistem. Pengujian dilakukan dengan mengakses sistem informasi pada suatu perangkat. Proses pengujian dilakukan dengan membandingkan untuk memastikan bahwa semua fitur yang telah dikembangkan beroperasi tanpa kendala dan bahwa setiap halaman berfungsi sesuai dengan perencanaan. Hasil pengujian ini digunakan sebagai landasan untuk mengevaluasi sejauh mana sistem yang dikembangkan memenuhi tujuan dan kebutuhan yang diinginkan.

### 1.7 Sistematika Penulisan

Penyusunan format penulisan dirancang dengan tujuan mempermudah pembaca untuk lebih memahami laporan tugas akhir ini. Berikut adalah struktur penulisan yang digunakan dalam penelitian ini:

#### a. BAB I PENDAHULUAN

Bagian pendahuluan menjelaskan dasar-dasar yang mendorong penulis untuk melakukan penelitian ini. Bab ini mencakup latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, dan sistematika penulisan.

b. **BAB II TINJAUAN PUSTAKA**

Bagian tinjauan pustaka mencakup teori-teori dan konsep-konsep penting yang menjadi dasar dalam memahami topik penelitian serta membantu dalam merumuskan solusi atau pengembangan lebih lanjut.

c. **BAB III METODOLOGI DAN PERANCANGAN SISTEM**

Bagian ini menjelaskan metode penelitian dan perancangan sistem, serta memberikan gambaran umum tentang sistem yang dikembangkan sebagai pedoman dalam proses pembuatan sistem.

d. **BAB IV IMPLEMENTASI HASIL DAN ANALISIS**

Bagian ini mencakup analisis implementasi sistem dan evaluasi terhadap sistem yang telah berhasil dibuat.

e. **BAB V PENUTUP**

Bagian penutup memuat rangkuman dari temuan yang diperoleh selama pelaksanaan tugas akhir, beserta saran untuk pengembangan sistem pada tahap berikutnya.

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Sistem Informasi Manajemen Kos-kosan**

Sistem informasi merupakan "sekumpulan komponen informasi yang saling berhubungan, yang bertujuan untuk mengumpulkan, memproses, menyimpan, dan mendistribusikan informasi" (Nitami dkk., 2021). Dalam konteks indekos, atau yang dikenal sebagai kos-kosan, penerapan sistem informasi sangat relevan untuk mengelola berbagai aspek operasional secara lebih efisien dan terstruktur. Kos-kosan adalah layanan penyewaan tempat tinggal sementara dengan sistem pembayaran berkala, umumnya bulanan. Menurut Hartono dan Gianawati (2013), seperti dikutip oleh Christianto & Isputrawan (2022), rumah kos berfungsi sebagai tempat tinggal sementara bagi individu yang membutuhkan tempat tinggal yang praktis dan terjangkau, seperti mahasiswa atau pekerja. Kos-kosan umumnya dilengkapi dengan fasilitas dasar, seperti kamar tidur, kamar mandi, listrik, dan akses internet.

Sistem informasi manajemen kos-kosan merupakan sebuah aplikasi berbasis digital yang dirancang untuk mempermudah pengelolaan berbagai aspek kos, mulai dari administrasi penghuni, pemeliharaan fasilitas, hingga sistem pembayaran. Aldi, Ghiffari, Praptono, & Sagita (2020) menyatakan bahwa sistem ini berperan penting dalam mengotomatisasi proses pengelolaan kos, yang mencakup aktivitas pemantauan fasilitas, pengelolaan data penghuni, serta pencatatan transaksi pembayaran. Penelitian yang dilakukan oleh Choldun, Fauziah, & Sentanu (2022) menunjukkan bahwa sistem informasi manajemen kos-kosan umumnya dilengkapi dengan fitur-fitur yang mendukung pengelolaan data kos secara menyeluruh, termasuk fitur untuk memantau ketersediaan kamar, mengelola fasilitas, dan memproses transaksi pembayaran. Hal ini memudahkan pemilik kos untuk melakukan kontrol dan pengawasan secara lebih efisien.

Lebih lanjut, Khumaidah & Riyantomo (2019) mengembangkan sebuah sistem serupa yang memungkinkan penyewa untuk melakukan transaksi sewa secara daring, serta menyediakan laporan keuangan bulanan dan tahunan yang merinci data penyewa, status pembayaran, dan kondisi keuangan kos. Hal ini memberikan nilai tambah berupa transparansi dan efisiensi dalam pengelolaan kos-kosan. Rahmatya, Simangunsong, & Wicaksono (2022) juga menambahkan bahwa sistem informasi ini berfungsi sebagai alat bantu pemilik kos dalam memantau transaksi pemesanan kamar, mengelola laporan keuangan, serta memberikan informasi tagihan secara cepat dan akurat kepada penyewa.

Pengertian sistem informasi manajemen kos-kosan, berdasarkan uraian di atas, adalah sebuah sistem digital yang terintegrasi dan dirancang untuk mempermudah pengelolaan seluruh aspek operasional kos-kosan. Sistem ini mencakup pengelolaan data penghuni, fasilitas, pembayaran, hingga penyediaan laporan keuangan yang tersusun secara terstruktur dan transparan. Selain itu, sistem ini juga berperan sebagai media pemasaran bagi pemilik kos serta sarana akses informasi bagi penyewa. Dengan fitur pengelolaan daring, dan pencatatan otomatis, sistem ini menawarkan solusi yang lebih efisien dan akurat dibandingkan metode pengelolaan manual, serta membantu meningkatkan transparansi dan akuntabilitas dalam pengelolaan kos-kosan.

## 2.2 Metodologi Pengembangan Sistem Informasi

Sebelumnya, sudah terdapat beberapa penelitian yang serupa dengan penelitian tentang pengembangan sistem informasi manajemen kos-kosan. Penelitian-penelitian tersebut dapat dilihat pada Tabel 2.1 Perbandingan penelitian sejenis.

Tabel 2.1 Perbandingan penelitian sejenis

Metode	Peneliti	Penjelasan	Kelebihan dan Kekurangan
<i>Waterfall</i>	1. Rahmatya dkk. (2022) 2. Khumaidah & Riyantomo (2019) 3. Luh dkk. (2021)	Metode pengembangan yang mengikuti proses linier dan berurutan. Setiap tahap (perencanaan, analisis, desain, implementasi, pengujian, pemeliharaan) harus selesai sebelum melanjutkan ke tahap berikutnya.	+ Tahapan pengembangan jelas, terstruktur, dan mudah diterapkan. + Kebutuhan sistem terdefinisi di awal, meminimalkan kesalahan. + Dokumentasi lengkap, memudahkan pemeliharaan.
			– Tidak fleksibel menghadapi perubahan di tengah proses. – Hanya cocok untuk proyek yang sejak awal sudah memiliki kebutuhan dan tujuan yang jelas.
<i>Prototype</i>	1. Gitaswari dkk. (2023) 2. Wicaksono, Kharisma, & Sianturi (2022)	Metode pengembangan yang berfokus pada pembuatan model awal atau prototipe. Dievaluasi oleh pengguna untuk mendapatkan umpan	+ Waktu pengembangan lebih cepat. + Biaya pengembangan lebih kecil. + Prototipe mempermudah visualisasi dan pengujian kebutuhan.

		balik yang membantu penyempurnaan sistem sebelum versi final dibuat.	<ul style="list-style-type: none"> <li>- Kurangnya Skalabilitas.</li> <li>- Ketergantungan pada umpan balik Pengguna.</li> </ul>
Agile	1. Aldi dkk. (2020)	Metode pengembangan yang efisien dan fleksibel untuk pengembangan jangka pendek, serta mampu beradaptasi dengan perubahan tanpa menghambat proses pengembangan yang sedang berlangsung.	<ul style="list-style-type: none"> <li>+ Cocok untuk proyek dengan waktu pengembangan cepat.</li> <li>+ Fleksibilitas dalam Penyesuaian Kebutuhan.</li> </ul>
			<ul style="list-style-type: none"> <li>- Membutuhkan komitmen tinggi dari pengembang dan pengguna.</li> <li>- Sulit dikelola pada Proyek Besar.</li> </ul>

Berdasarkan perbandingan beberapa metodologi yang disajikan pada Tabel 2.1, metode *Waterfall* lebih unggul dalam pengembangan sistem informasi kos-kosan berbasis Progressive Web App (PWA). Metode ini menawarkan pendekatan yang terstruktur dan berurutan, di mana setiap tahap pengembangan direncanakan dan dijalankan dengan jelas. Hal ini sangat cocok untuk proyek yang memiliki kebutuhan sistem yang sudah terdefinisi sejak awal, sehingga setiap tahap dapat dilaksanakan dengan tepat, tanpa banyak perubahan di tengah proses.

Keunggulan utama metode *Waterfall* adalah kontrol yang ketat pada setiap fase pengembangan, yang dimulai dari perencanaan hingga pemeliharaan, memastikan bahwa setiap tahap selesai sebelum melanjutkan ke tahap berikutnya. Selain itu, metode ini juga menghasilkan dokumentasi yang lengkap dan terperinci, yang sangat berguna untuk pemeliharaan jangka panjang dan pengembangan lebih lanjut.

Dibandingkan metode lain seperti Prototype yang bergantung pada umpan balik pengguna, *Waterfall* meminimalkan penyesuaian berulang karena kebutuhan sudah ditetapkan di awal. *Waterfall* juga lebih mudah dikelola daripada metode Agile, yang memerlukan fleksibilitas tinggi dan sulit untuk menghadapi tantangan dalam proyek besar. *Waterfall* memastikan setiap komponen diuji secara menyeluruh sebelum lanjut ke tahap berikutnya, berbeda dengan Agile yang melakukan pengujian paralel dengan pengembangan, yang bisa meningkatkan risiko kesalahan terlewat.

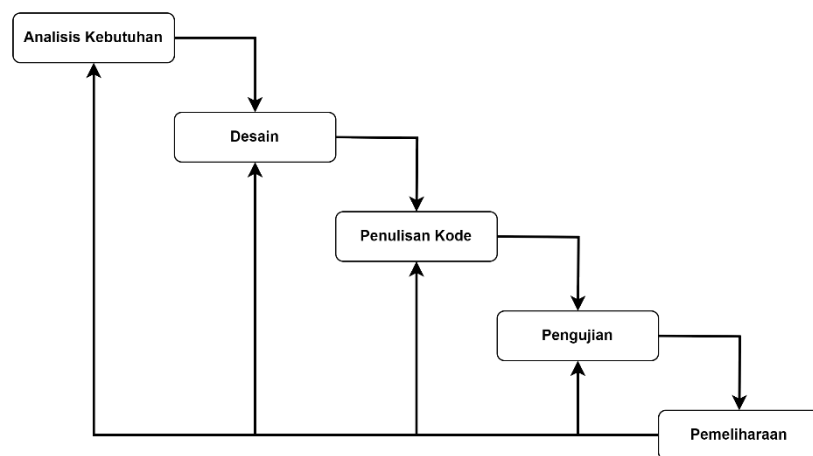
Berdasarkan pembahasan tentang metodologi pengembangan sistem informasi di atas, metode *Waterfall* menjadi pilihan yang sangat relevan untuk pengembangan sistem informasi manajemen kos-kosan ini. Pendekatan *Waterfall*, yang memiliki struktur yang jelas dan urutan

tahapan yang sistematis, memungkinkan setiap fase pengembangan dilaksanakan dengan baik dan terdokumentasi secara menyeluruh. Dengan karakteristiknya yang sesuai untuk proyek yang memiliki spesifikasi dan persyaratan yang jelas sejak awal, metode ini diharapkan dapat memberikan hasil yang efektif dan efisien dalam pengembangan sistem ini.

### 2.3 Metode Waterfall

Metode *Waterfall* merupakan salah satu model pengembangan perangkat lunak yang paling banyak digunakan dalam industri teknologi. Metode ini pertama kali diperkenalkan oleh Winston Royce pada tahun 1970 dan dikenal sebagai model yang berurutan dan sistematis. Proses pengembangannya dibagi menjadi beberapa tahap, yang harus diselesaikan satu per satu sebelum melanjutkan ke tahap berikutnya (Abdul Wahid, 2020).

Metode *Waterfall* sering diterapkan pada proyek yang memiliki spesifikasi dan persyaratan yang jelas dari awal. Dengan menggunakan pendekatan ini, pengembang dapat memastikan setiap tahap terdefinisi dengan baik dan terdokumentasi secara menyeluruh, sehingga memudahkan pengawasan dan pengelolaan proyek.



Gambar 2.1 Metode waterfall

Tahapan yang dilalui adalah sebagai berikut:

- Analisis Kebutuhan: Mengumpulkan dan mendokumentasikan kebutuhan pengguna melalui wawancara, survei, atau observasi.
- Desain: Merancang sistem berdasarkan kebutuhan yang telah dianalisis, mencakup arsitektur, *database*, dan antarmuka.

- c. Penulisan Kode: Menulis dan mengintegrasikan kode secara bertahap sesuai desain yang telah dibuat.
- d. Pengujian: Menguji sistem untuk memastikan fungsionalitas sesuai spesifikasi dan menemukan bug.
- e. Pemeliharaan: Memelihara dan memperbaiki sistem setelah implementasi, serta menyesuaikan dengan kebutuhan baru.

*Kelebihan Metode Waterfall:*

- a. Tahapan pengembangan jelas, terstruktur, dan mudah diterapkan.
- b. Kebutuhan sistem terdefinisi di awal, meminimalkan kesalahan.
- c. Dokumentasi lengkap, memudahkan pemeliharaan.

*Kekurangan Metode Waterfall:*

- a. Tidak fleksibel menghadapi perubahan di tengah proses.
- b. Tidak cocok untuk proyek dengan spesifikasi yang sering berubah.

Secara keseluruhan, metode *Waterfall* menawarkan pendekatan yang terstruktur dan sistematis dalam pengembangan perangkat lunak, namun penggunaannya lebih sesuai untuk proyek yang memiliki spesifikasi yang stabil dan persyaratan yang jelas dari awal.

## **2.4 Progressive Web App**

Dalam metode *Waterfall*, salah satu tahap penting adalah merancang teknologi yang sesuai dengan kebutuhan sistem yang telah didefinisikan sejak awal. Dalam pengembangan aplikasi, terdapat beberapa pendekatan teknologi yang bisa dipertimbangkan, seperti Progressive Web App (PWA), native app, cross-platform app, dan hybrid app.

- a. Progressive Web App (PWA) adalah teknologi yang menggabungkan keunggulan aplikasi web dan aplikasi native, menghadirkan solusi atas keterbatasan aplikasi berbasis web dan performa browser seluler yang seringkali terbatas (Amrullah, Salim, & Rachman Manga, 2021).
- b. Aplikasi native adalah aplikasi yang dikembangkan secara spesifik untuk satu platform, seperti iOS atau Android, dengan menggunakan bahasa pemrograman dan *framework* yang sesuai (Ilhami, 2021).

- c. Aplikasi cross-platform adalah aplikasi yang dikembangkan menggunakan *framework* yang memungkinkan kode yang sama dapat berjalan di lebih dari satu platform (Ilhami, 2021)
- d. Hybrid app adalah jenis aplikasi mobile yang dibangun dengan menggabungkan teknologi web dan teknologi native. Pendekatan ini memungkinkan pengembang untuk menciptakan aplikasi yang dapat beroperasi di berbagai platform mobile (Zaki Hasibuan & Suendri, 2023).

Berikut adalah perbandingan kelebihan dan kekurangan dari keempat pendekatan dalam pengembangan aplikasi yang telah disebutkan:

Jenis Aplikasi	Kelebihan	Kekurangan
Progressive Web App (PWA)	<ul style="list-style-type: none"> <li>• Mudah diakses dari browser tanpa perlu instalasi dari app store</li> <li>• Lebih ringan dibanding aplikasi native</li> <li>• Biaya pengembangan lebih murah</li> <li>• Bisa bekerja offline dengan caching data</li> <li>• Memiliki kemampuan untuk menerima notifikasi push</li> <li>• Tidak perlu pembaruan di app store.</li> </ul>	<ul style="list-style-type: none"> <li>• Tidak memiliki akses penuh terhadap fitur hardware perangkat seperti aplikasi native</li> <li>• Performa biasanya lebih lambat dibanding aplikasi native</li> <li>• Fungsionalitas terbatas pada fitur-fitur yang didukung oleh browser</li> <li>• Kurang optimal untuk aplikasi yang membutuhkan performa tinggi.</li> </ul>
Native App	<ul style="list-style-type: none"> <li>• Performa optimal karena dibuat khusus untuk platform tertentu</li> <li>• Akses penuh terhadap fitur perangkat keras</li> <li>• Pengalaman pengguna terbaik karena menggunakan komponen UI/UX asli dari platform</li> <li>• Mendukung fitur yang lebih kompleks dan berat.</li> </ul>	<ul style="list-style-type: none"> <li>• Biaya pengembangan tinggi karena memerlukan pengembangan terpisah untuk masing-masing platform (iOS, Android)</li> <li>• Membutuhkan pemeliharaan dan pembaruan terpisah di masing-masing platform</li> <li>• Proses distribusi aplikasi lebih panjang karena harus melalui proses approval di app store.</li> </ul>
Cross-Platform App	<ul style="list-style-type: none"> <li>• Menghemat waktu dan biaya pengembangan karena satu kode bisa dijalankan di berbagai platform</li> <li>• Pemeliharaan lebih mudah karena satu basis kode untuk banyak platform</li> </ul>	<ul style="list-style-type: none"> <li>• Meskipun hampir menyerupai native, performa masih belum sebaik aplikasi native</li> <li>• Akses ke fitur hardware perangkat bisa terbatas atau membutuhkan penyesuaian khusus</li> </ul>

	<ul style="list-style-type: none"> <li>• Mendekati performa native, terutama dengan <i>framework</i> modern seperti Flutter atau React Native.</li> </ul>	<ul style="list-style-type: none"> <li>• Desain UI/UX mungkin tidak sepenuhnya mengikuti standar native dari masing-masing platform (Android dan iOS).</li> </ul>
Hybrid App	<ul style="list-style-type: none"> <li>• Kode tunggal yang dapat digunakan di banyak platform</li> <li>• Biaya pengembangan dan pemeliharaan lebih rendah dibandingkan dengan aplikasi native</li> <li>• Pengembangan lebih cepat karena menggunakan teknologi web yang familiar (HTML, CSS, JavaScript).</li> </ul>	<ul style="list-style-type: none"> <li>• Performa lebih lambat dibandingkan dengan aplikasi native dan cross-platform</li> <li>• UI/UX kurang mulus dan kadang terasa "kurang alami" di platform tertentu karena tidak sepenuhnya mengikuti pedoman desain native</li> <li>• Akses ke fitur perangkat keras terbatas, seringkali membutuhkan plugin tambahan atau kustomisasi khusus.</li> </ul>

Memilih menggunakan Progressive Web App (PWA) memiliki beberapa alasan kuat dibandingkan dengan jenis aplikasi lain, terutama ketika mempertimbangkan kebutuhan bisnis, efisiensi biaya, dan jangkauan pengguna. Berikut adalah beberapa alasan utama:

a. Biaya Pengembangan dan Pemeliharaan Lebih Rendah

Menurut (Herman & Frederick, 2023), biaya pengembangan untuk aplikasi native mobile biasanya lebih tinggi dibandingkan dengan PWA. Hal ini karena PWA hanya memerlukan satu versi aplikasi yang dapat diakses di berbagai platform, baik desktop maupun mobile. Ini jauh lebih efisien dibandingkan dengan Native Apps, yang memerlukan pengembangan terpisah untuk iOS dan Android. Meskipun Cross-Platform Apps menggunakan satu basis kode, namun tetap memerlukan adaptasi untuk berbagai platform.

b. Instalasi Mudah tanpa App Store

PWA, yang merupakan kombinasi dari aplikasi mobile dan web, memungkinkan pengguna untuk mengakses aplikasi secara langsung tanpa perlu melalui proses pengunduhan atau instalasi terlebih dahulu (Putra, Satwika, & Nirmala, 2020). Selain itu, Progressive Web Apps (PWA) dapat diinstal di perangkat mobile dan desktop melalui browser (Kurniawan, 2020). Berbeda dengan Native Apps, Cross-Platform Apps, dan Hybrid Apps, PWA tidak memerlukan file aplikasi yang diunduh maupun distribusi melalui app store. Hal ini

memungkinkan pengguna untuk langsung menginstal aplikasi dari browser tanpa harus melalui proses unduhan atau persetujuan di toko aplikasi.

c. Jangkauan Pengguna Lebih Luas

Nurwanto (2019) menjelaskan bahwa Progressive Web Applications (PWA) dapat digunakan di berbagai platform, yang menunjukkan bahwa PWA memiliki jangkauan yang lebih luas. Dibandingkan dengan Native Apps yang hanya tersedia di perangkat dengan spesifikasi tertentu, PWA menawarkan akses universal tanpa bergantung pada perangkat atau sistem operasi tertentu. Selain itu, Hidayat dkk. (2023) juga menyatakan bahwa PWA juga unggul dalam kecepatan pembuatan dan kemudahan akses melalui berbagai perangkat.

d. Pembaruan Lebih Mudah dan Cepat

PWA dapat diakses tanpa melalui proses download melalui media penyimpanan maupun Google Playstore atau Apple Store (Matiini, Setiyadi, Setiawan, & Ramli, 2021). PWA dijalankan melalui browser web, sehingga pembaruan dilakukan langsung dari server tanpa memerlukan instalasi ulang. Proses pembaruan ini lebih cepat dibandingkan dengan Native Apps, Hybrid Apps, dan Cross-Platform Apps, yang memerlukan proses pembaruan manual melalui platform distribusi. Pengguna dari aplikasi jenis ini harus mengunduh dan menginstal pembaruan untuk mendapatkan versi terbaru, yang dapat memperlambat akses mereka terhadap fitur dan perbaikan terbaru.

Dengan berbagai keunggulan yang dimilikinya, penggunaan Progressive Web App (PWA) dalam pengembangan sistem informasi kos-kosan sangat dianjurkan. Penggunaan Progressive Web App (PWA) memiliki banyak keuntungan dibandingkan dengan jenis aplikasi lainnya, terutama dalam hal biaya, efisiensi, dan jangkauan pengguna. PWA lebih murah dalam pengembangan dan pemeliharaan karena hanya memerlukan satu versi aplikasi yang dapat diakses di berbagai platform tanpa perlu membuat versi terpisah untuk setiap sistem operasi. Selain itu, pengguna dapat mengakses PWA langsung melalui browser tanpa harus mengunduh dari toko aplikasi. PWA juga dapat diakses di perangkat dengan spesifikasi rendah dan tidak tergantung pada sistem operasi tertentu. Terakhir, PWA memudahkan pembaruan, karena pengguna bisa mendapatkan versi terbaru tanpa proses rumit. Dengan semua kelebihan ini, PWA menjadi pilihan menarik bagi pengembang dan bisnis untuk memberikan pengalaman pengguna yang lebih baik dan lebih mudah diakses.

## 2.5 Blackbox Testing

Setelah proses pengembangan sistem informasi kos-kosan berbasis Progressive Web App (PWA) melalui tahapan-tahapan yang terstruktur dalam metode *Waterfall*, tahap berikutnya yang sangat krusial adalah pengujian sistem. Pengujian ini bertujuan untuk memastikan bahwa aplikasi yang dikembangkan tidak hanya berfungsi sesuai spesifikasi, tetapi juga memberikan pengalaman pengguna yang optimal, sejalan dengan keunggulan teknologi PWA yang sudah dijelaskan sebelumnya.

Pengujian sistem perangkat lunak adalah langkah kritis dalam siklus pengembangan perangkat lunak untuk memastikan bahwa aplikasi berfungsi sesuai dengan persyaratan yang ditetapkan. Proses pengujian ini mencakup berbagai metode yang dirancang untuk mengidentifikasi kesalahan, bug, dan kekurangan dalam sistem sebelum perangkat lunak diterapkan secara luas. Pengujian sistem bertujuan untuk mengevaluasi apakah perangkat lunak mampu menjalankan tugas-tugas yang diharapkan dan memenuhi standar kualitas yang diinginkan. Melalui metode seperti black-box testing, white-box testing, dan kombinasi keduanya (grey-box testing), pengembang dapat mengevaluasi fungsionalitas eksternal maupun logika internal dari perangkat lunak, memastikan keamanan, kinerja, dan stabilitas sistem yang dikembangkan.

Verma, Khatana, & Chaudhary (2017) menjelaskan jenis pengujian black-box dan white-box testing yang disajikan sebagai berikut:

Tabel 2.2 Pengujian Sistem

Aspek	Blackbox Testing	Whitebox Testing
Nama Lain	Dikenal juga sebagai Specification Based Technique	Dikenal juga sebagai Structural Testing Technique
Fokus Pengujian	Berfokus pada fungsionalitas sistem tanpa memperhatikan struktur internal	Berfokus pada struktur internal aplikasi seperti kode, cabang, loop, dan kondisi
Pengetahuan yang Dibutuhkan	Tidak membutuhkan pengetahuan tentang struktur internal atau kode sumber aplikasi	Mebutuhkan pemahaman mendalam tentang kode sumber dan struktur internal aplikasi
Pendekatan Utama	Berdasarkan pada spesifikasi dan persyaratan aplikasi	Berdasarkan pada analisis kode dan logika program
Kelebihan	+ Tidak memerlukan pengetahuan teknis, cocok untuk pengujian oleh pihak non-teknis + Menghindari bias	+ Memberikan cakupan lebih mendalam karena menguji jalur logika kode + Memungkinkan optimasi kode

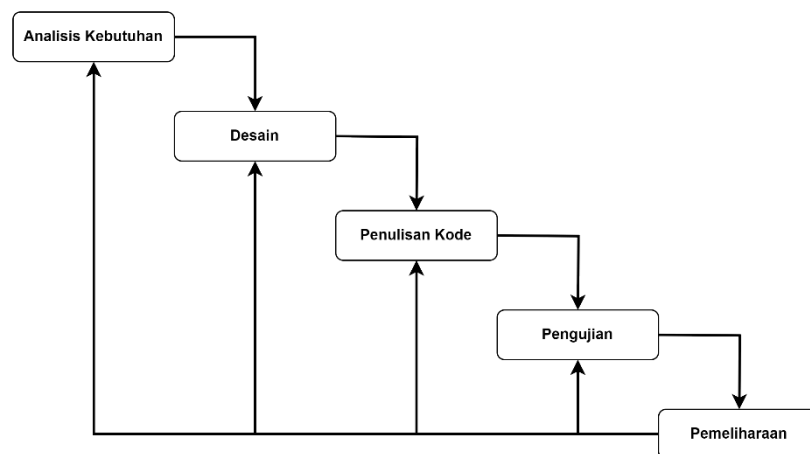
	<ul style="list-style-type: none"> <li>pengembang karena dilakukan oleh pihak ketiga</li> <li>+ Efisien untuk sistem yang besar</li> </ul>	<ul style="list-style-type: none"> <li>+ Dapat dimulai lebih awal dalam siklus pengembangan</li> </ul>
Kelemahan	<ul style="list-style-type: none"> <li>- Tidak bisa mendeteksi kesalahan dalam logika internal kode</li> <li>- Sulit merancang kasus uji jika spesifikasi tidak jelas</li> <li>- Tidak cocok untuk pengujian segmen kode yang kompleks</li> </ul>	<ul style="list-style-type: none"> <li>- Memerlukan pemahaman teknis yang tinggi</li> <li>- Pengujian memerlukan lebih banyak waktu dan persiapan</li> <li>- Tidak selalu memungkinkan menguji setiap jalur logika</li> </ul>

*Blackbox* Testing sering dipilih karena fokusnya pada kebutuhan pengguna dan spesifikasi fungsional sistem, tanpa memerlukan pemahaman mendalam tentang kode. Metode ini memungkinkan pihak ketiga atau tim non-teknis melakukan pengujian, mengurangi risiko bias dari pengembang. Pengujian lebih cepat dilakukan karena tidak membutuhkan keahlian teknis yang mendalam, dan memastikan aplikasi sesuai dengan kebutuhan pengguna tanpa terpengaruh oleh detail internal sistem.

Menurut Verma dkk. (2017) *Blackbox* Testing memiliki beberapa kelebihan, diantaranya adalah pengujian dilakukan berdasarkan kebutuhan pelanggan, dilakukan oleh pihak ketiga untuk menghindari bias pengembang, dan tidak memerlukan penguji dengan latar belakang teknis. Selain itu, teknik ini lebih efisien digunakan pada sistem yang lebih besar, dengan kasus uji yang dapat dirancang segera setelah spesifikasi selesai.

## BAB III METODOLOGI DAN PERANCANGAN SISTEM

Pada bab ini akan dijelaskan tahapan-tahapan yang dilakukan untuk mengembangkan sistem informasi berdasarkan metode yang digunakan. Metode pengembangan sistem yang digunakan dalam penelitian ini adalah metode *waterfall*. Pada bab ini dilakukan analisis kebutuhan dan perancangan sistem. Analisis kebutuhan dilakukan untuk mengetahui kebutuhan sistem sesuai kebutuhan pengguna. Kemudian dilanjutkan pada tahap perancangan sistem informasi yang akan dibangun. Tahapan dari metode *waterfall* dapat dilihat pada gambar dibawah ini.



Gambar 3.1 Metode waterfall

### 3.1 Analisis Kebutuhan

Analisis kebutuhan menentukan kebutuhan-kebutuhan sistem informasi yang akan dibuat. Informasi yang dihasilkan dari analisis ini digunakan sebagai acuan untuk pengembangan sistem informasi manajemen kos-kosan yang akan dibangun.

#### 3.1.1 Pengumpulan Data

Pada tahap pengumpulan data, dilakukan pencarian informasi terkait sistem informasi manajemen kos-kosan. Proses pengumpulan data ini melibatkan observasi langsung di beberapa kos serta mencari informasi dari berbagai sumber. Pendekatan ini bertujuan untuk memahami berbagai aspek dan perkembangan terkini terkait pengembangan manajemen kos, serta memanfaatkan temuan-temuan yang ada dari penelitian sebelumnya.

a. Observasi

Pada tahap pengumpulan data melalui metode observasi, dilakukan peninjauan pada tiga tempat kos yang berbeda, yaitu Kos Pak Arif Squad, Kos Griya Widi, dan Kost Putra Ahsan. Dari ketiga kos tersebut, terungkap bahwa sistem pembayaran kos sudah dapat menggunakan *e-wallet*. Namun, dalam proses pembayaran tagihan, masih diperlukan langkah tambahan di mana penghuni kos harus secara manual mengirimkan bukti pembayaran kepada pemilik kos setelah melakukan pembayaran. Praktik ini tidak hanya menuntut usaha tambahan, tetapi juga menimbulkan ketidaknyamanan bagi para penghuni kos. Selain itu, pemilik kos juga perlu mengirimkan pengingat pembayaran kos melalui WhatsApp secara manual jika terdapat penghuni yang lupa atau belum membayar.

b. Sumber Terkait

Pada tahap pengumpulan data, dilakukan pencarian informasi dari berbagai sumber yang berkaitan dengan manajemen kos. Proses ini melibatkan pengumpulan data dari tiga sumber yang dianggap relevan dalam konteks manajemen kos. Dengan memanfaatkan ketiga sumber ini, diharapkan dapat memperoleh informasi yang komprehensif dan mendukung dalam pengembangan sistem informasi manajemen kos yang efektif dan efisien.

Tabel 3.1 Sumber terkait

Peneliti	Fitur	Saran dari peneliti
Luh dkk. (2021)	<ul style="list-style-type: none"> <li>• Pengelolaan data penghuni.</li> <li>• Pengelolaan data tagihan</li> <li>• Verifikasi pembayaran kos.</li> <li>• Laporan keuangan yang dapat diekspor.</li> <li>• Penghuni dapat melihat dan membayar tagihan.</li> </ul>	<ul style="list-style-type: none"> <li>• Perlu diperluas platformnya agar dapat digunakan di android.</li> <li>• Penambahan fitur pembayaran online.</li> </ul>
Fatimah Isnay Nur Alvivi & Voutama (2024)	<ul style="list-style-type: none"> <li>• Pengelolaan data kamar.</li> <li>• Pengelolaan data tagihan.</li> <li>• Verifikasi pembayaran kos.</li> <li>• Penghuni dapat melihat dan membayar tagihan.</li> </ul>	<ul style="list-style-type: none"> <li>• Penambahan fitur notifikasi pembayaran.</li> <li>• Perlu diintegrasikan dengan layanan pembayaran pihak ketiga.</li> <li>• Perlu dikembangkan fitur laporan keuangan.</li> </ul>

Wicaksono, Kharisma, & Sianturi (2022)	<ul style="list-style-type: none"> <li>• Pengelolaan data penghuni.</li> <li>• Pengelolaan data kamar kos.</li> <li>• Laporan pendapatan serta pengeluaran kos.</li> </ul>	<ul style="list-style-type: none"> <li>• Penambahan fitur arsip penghuni.</li> <li>• Penambahan fitur pengingat tagihan melalui whatsapp.</li> <li>• Perlu diperluas platformnya agar dapat diakses melalui Web, Android, dan iOS.</li> </ul>
--	--	---

Dari dua sumber penelitian yang telah dijelaskan, fitur-fitur yang telah ada pada sistem menjadi referensi penting dalam pengembangan sistem baru. Analisis fitur-fitur tersebut dapat memberikan wawasan mendalam terkait fungsionalitas yang perlu diintegrasikan dalam pengembangan sistem yang baru. Selain itu, saran-saran yang diberikan oleh penulis pada kedua penelitian juga menjadi panduan berharga untuk meningkatkan kualitas sistem. Integrasi fitur seperti pengiriman pengingat tagihan, dan perluasan platform aplikasi menjadi hal-hal yang dapat menjadi fokus dalam pengembangan sistem yang lebih baik dan lebih komprehensif. Dengan memanfaatkan masukan dari penelitian sebelumnya, diharapkan sistem yang baru dapat memberikan solusi yang lebih efektif dan responsif terhadap kebutuhan manajemen kos-kosan.

### 3.1.2 Analisis Kebutuhan Fungsional

Kebutuhan fungsional adalah sekumpulan fitur atau kemampuan yang diinginkan dari suatu sistem perangkat lunak. Kebutuhan fungsional mencakup aspek-aspek spesifik yang harus diimplementasikan agar sistem dapat memenuhi kebutuhan pengguna. Berdasarkan data-data yang ada, Didapatkan kebutuhan fungsional sistem sebagai berikut:

#### a. Pengelolaan Data Pengguna

Sistem menyediakan fitur untuk mengelola data pengguna. Admin harus memiliki kewenangan untuk menambahkan, mengubah, atau menghapus data pengguna sesuai kebutuhan.

#### b. Pengelolaan Data Kamar Kos

Sistem menyediakan fitur untuk mengelola data kamar kos. Admin dapat menambahkan kamar baru ke dalam sistem, serta mengelola status kamar seperti "tersedia" atau "ditempati". Fitur ini memberikan fleksibilitas dalam manajemen properti kos.

c. Informasi Kamar

Sistem dapat menampilkan data kamar yang mencakup informasi seperti status, fasilitas yang tersedia, dan biaya sewa. Fitur ini memudahkan pengguna dalam mengelola dan memantau kondisi kamar yang ada.

d. Pengelolaan Tagihan

Sistem ini menyediakan fitur untuk membuat, menampilkan, menghapus, dan memperbarui tagihan. Penghapusan dan pembaruan hanya diperbolehkan untuk tagihan dengan status tidak dibayar atau sudah kadaluarsa, guna menjaga rekam jejak dan transparansi, serta mencegah kesalahan penghapusan. Admin dapat menampilkan informasi tagihan yang mencakup rincian seperti jumlah pembayaran, tanggal jatuh tempo, dan status pembayaran. Fitur ini membantu penghuni memahami kondisi tagihan mereka dengan cepat dan jelas.

e. Generate Tagihan Otomatis

Sistem dapat secara otomatis membuat tagihan bulanan untuk setiap pengguna, disesuaikan dengan detail kamar dan fasilitas yang digunakan untuk menentukan jumlah tagihan. Ini mengurangi risiko kesalahan dan menghemat waktu admin, memastikan setiap penghuni menerima tagihan tepat waktu tanpa perlu dilakukan secara manual.

f. Mengirim notifikasi tagihan

Sistem dapat mengirim notifikasi kepada pengguna terkait tagihan kos, melalui whatsapp. Notifikasi mencakup jumlah yang harus dibayarkan, tanggal jatuh tempo, dan kode pembayaran, membantu penghuni tetap terinformasi dan menghindari keterlambatan pembayaran.

g. Proses Pembayaran

Sistem mendukung proses pembayaran tagihan kos, memungkinkan pengguna melakukan pembayaran melalui ATM atau *e-wallet*. Setelah pembayaran selesai, sistem secara otomatis memperbarui status pembayaran di *database*, memastikan bahwa informasi yang ditampilkan selalu terkini.

h. Riwayat Transaksi

Sistem dapat menampilkan daftar riwayat transaksi keuangan, memungkinkan admin untuk melihat semua transaksi terkait pembayaran tagihan bulanan kos. Informasi transaksi mencakup tanggal, nama pengirim, dan jumlah uang yang dibayarkan.

i. Laporan Keuangan

Sistem dapat menyajikan laporan keuangan bulanan yang komprehensif. Fitur ini

memudahkan pengguna untuk mengunduh atau mencetak laporan guna memantau kinerja keuangan properti kos.

j. **Obrolan**

Sistem menyediakan fitur obrolan yang memungkinkan pengguna berkomunikasi langsung dengan admin melalui chat pribadi atau berpartisipasi dalam grup yang melibatkan seluruh penghuni. Fitur ini mendukung interaksi yang lebih efektif dan responsif di antara semua pihak yang terlibat, memudahkan pengguna untuk menyampaikan pertanyaan, masalah, atau berdiskusi tentang topik yang relevan.

### **3.1.3 Analisis Kebutuhan Non-fungsional**

Kebutuhan non-fungsional adalah aspek-aspek atau karakteristik sistem yang berkaitan dengan kinerja, keamanan, dan ketersediaan. Kebutuhan ini melibatkan aspek-aspek seperti kualitas, performa, dan sifat-sifat lain yang mempengaruhi cara sistem beroperasi.. Berikut adalah kebutuhan non-fungsional sistem:

- a. Pengguna harus dapat memahami dan menggunakan fitur sistem tanpa pelatihan tambahan.
- b. Sistem dapat terintegrasi dengan layanan pembayaran online untuk memudahkan proses pembayaran.

### **3.1.4 Identifikasi Pengguna**

Berdasarkan hasil analisis kebutuhan fungsional yang telah diperoleh, teridentifikasi dua pengguna yang dapat menggunakan sistem informasi ini, yaitu penghuni dan admin.

a. **Admin**

Admin akan memiliki tanggung jawab untuk mengelola data kamar, pengguna dan transaksi.

b. **Penghuni**

Penghuni dapat memanfaatkan sistem ini untuk melihat dan membayar tagihan, mengakses informasi mengenai kamar kos, serta mengecek riwayat transaksi dan status pembayaran.

## **3.2 Perancangan Sistem**

Tahap perancangan sistem memiliki tujuan untuk menyusun rancangan sistem secara terinci sebelum memulai tahap implementasi dengan merujuk pada kebutuhan fungsional yang telah diidentifikasi sebelumnya. Dalam tahap ini, dilakukan perancangan diagram struktur dan

perilaku sistem, perancangan interface, perancangan *database* serta perancangan Progressive Web App.

Diagram yang digunakan untuk perancangan struktur dan perilaku sistem adalah diagram *UML (Unified Modeling Language)*. Menurut Dharwiyanti (2013) “*Unified Modeling Language (UML)* adalah sebuah 'bahasa' yang telah menjadi standar dalam industri untuk visualisasi, merancang, dan mendokumentasikan sistem piranti lunak. *UML* menawarkan sebuah standar untuk merancang model sebuah sistem.” Pada *UML*, terdapat beberapa jenis diagram, yaitu:

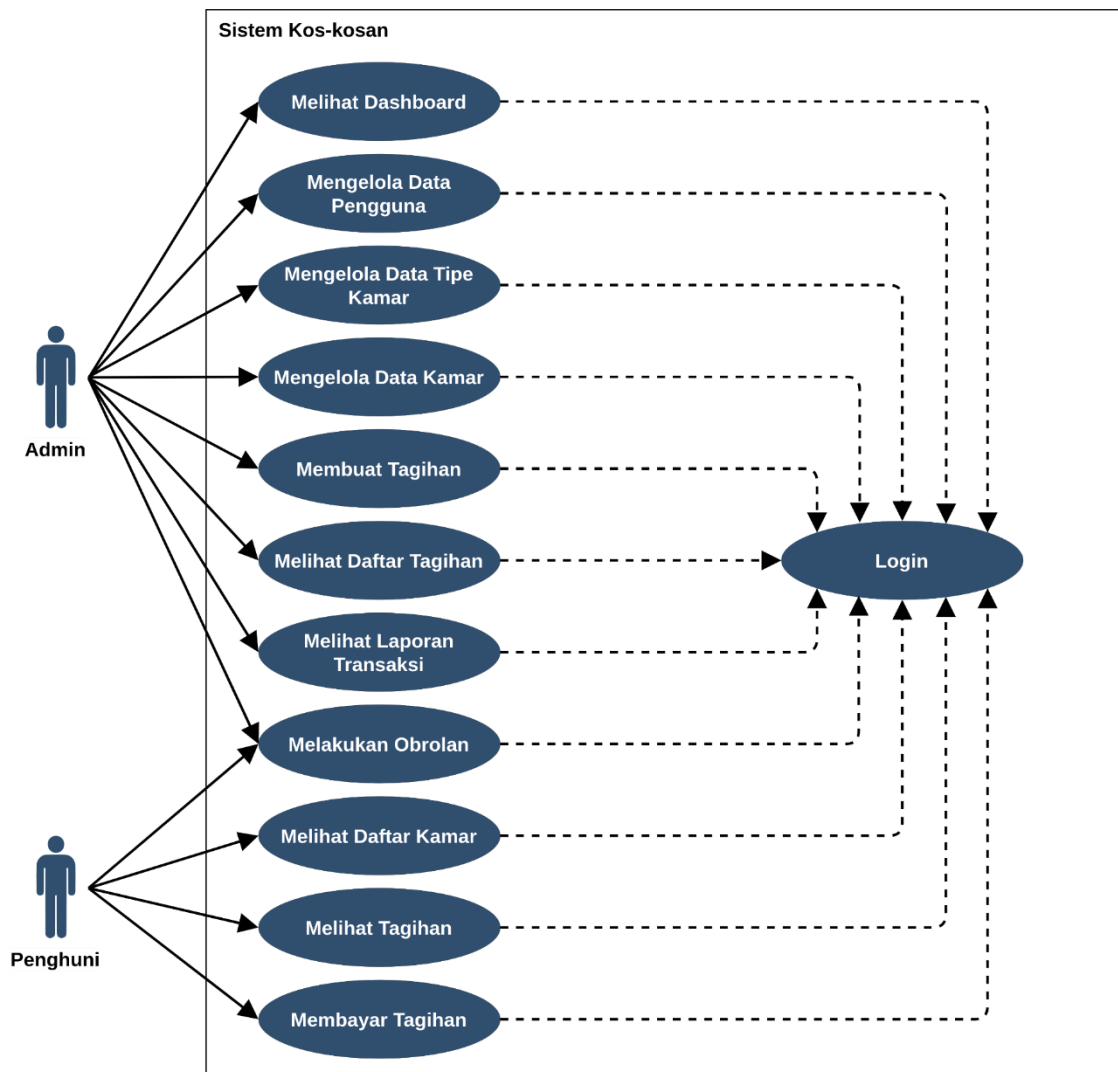
- a. *Use case diagram*
- b. Class diagram
- c. Statechart diagram
- d. *Activity diagram*
- e. Sequence diagram
- f. Collaboration diagram
- g. Component diagram
- h. Deployment diagram

Pada tahap ini, diagram *UML* yang digunakan dalam perancangan diagram struktur dan perilaku sistem adalah *Use case diagram* dan *Activity diagram*.

### **3.2.1 Use Case Diagram**

Diagram use-case adalah diagram yang mengilustrasikan fungsi-fungsi yang diharapkan dari suatu sistem. Menurut Astuti (2009), use-case menggambarkan interaksi antara aktor dengan sistem yang sedang dirancang, serta membantu memahami fungsi yang ada dalam sistem dan siapa saja yang berhak menggunakan fungsi tersebut. Selain itu, seperti yang diungkapkan oleh Dharwiyanti (2013), diagram use-case juga berperan penting dalam merumuskan kebutuhan sistem, berkomunikasi dengan klien mengenai rancangan, serta membantu perancangan test case untuk semua fitur yang ada dalam sistem.

Pada sistem yang dirancang. Sistem dapat digunakan oleh dua aktor, yaitu penghuni dan admin. Penjelasan lebih lanjut terkait interaksi setiap aktor pada sistem dapat dilihat pada Gambar 3.2.



Gambar 3.2 Use Case Diagram sistem

Rancangan use case diatas merupakan gambaran interaksi setiap aktor yang terlibat dalam sistem. Admin memiliki alur kerja melihat dashboard, mengelola data pengguna, mengelola data tipe kamar, mengelola data kamar, membuat tagihan, melihat daftar tagihan, melihat laporan transaksi dan melakukan obrolan. Penghuni memiliki alur kerja melakukan obrolan, melihat daftar kamar, melihat daftar tagihan dan membayar tagihan. Penjelasan dari *use case diagram* diatas dapat dilihat pada Tabel 3.2.

Tabel 3.2 Use Case Diagram sistem

No.	Nama Use Case	Aktor	Deskripsi
1	Login	Admin dan Penghuni	Proses otorisasi pengguna untuk masuk ke dalam sistem.

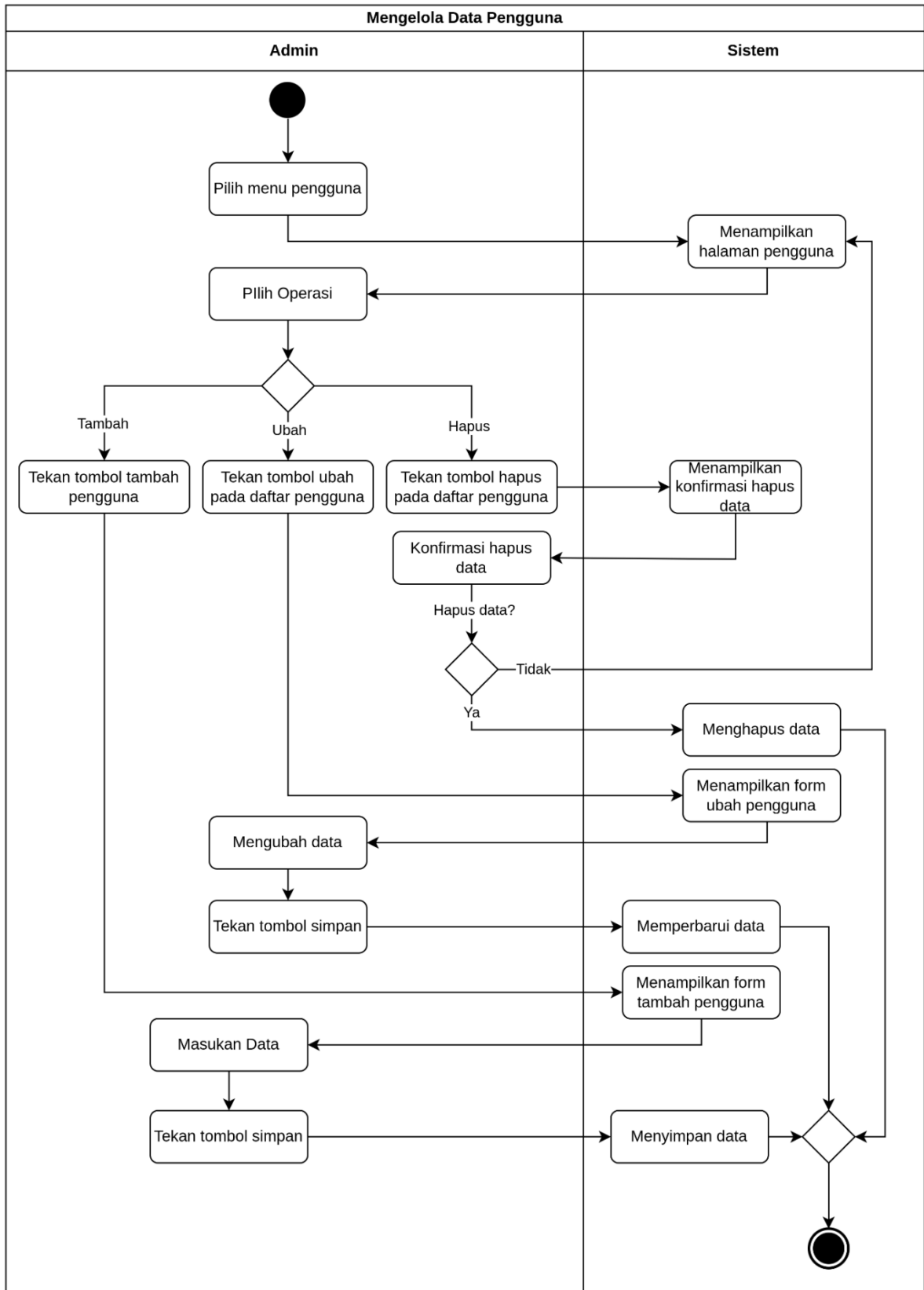
2	Melihat Dashboard	Admin	Proses menampilkan visualisasi informasi terkait manajemen kos secara ringkas dan akses ke berbagai fitur administrasi.
3	Mengelola Data Pengguna	Admin	Proses mengelola dan memperbarui informasi pengguna dalam sistem.
4	Mengelola Data Tipe Kamar	Admin	Proses mengatur dan memperbarui informasi tipe-tipe kamar yang tersedia di kos.
5	Mengelola Data Kamar	Admin	Proses mengelola dan memperbarui informasi terkait kamar yang tersedia di kos.
6	Membuat Tagihan	Admin	Proses pembuatan dan pengaturan tagihan untuk penghuni kos.
7	Melihat Daftar Tagihan	Admin	Proses melihat daftar tagihan yang telah dibuat untuk para penghuni.
8	Melihat Laporan Transaksi	Admin	Proses melihat dan menganalisis laporan transaksi yang terjadi dalam sistem.
9	Melakukan Obrolan	Admin dan Penghuni	Proses melakukan komunikasi antara penghuni dan admin kos.
10	Melihat Daftar Kamar	Penghuni	Proses melihat daftar kamar yang tersedia dan detail terkait.
11	Melihat Tagihan	Penghuni	Proses melihat tagihan yang harus dibayar oleh penghuni.
12	Membayar Tagihan	Penghuni	Proses melakukan pembayaran tagihan kos melalui sistem.

### 3.2.2 Activity Diagram

Menurut Tohari 2014, yang dikutip oleh Siking, Koniyo, & Yassin (2023), Diagram aktivitas digunakan untuk memodelkan alur kerja dan urutan aktivitas dalam suatu proses bisnis, sehingga pembuatan diagram aktivitas pada tahap awal pemodelan proses memberikan keuntungan untuk memahami secara komprehensif keseluruhan proses tersebut.

*Activity diagram* pada sistem ini terdiri dari sepuluh aktivitas yang memperlihatkan langkah-langkah yang terlibat dalam proses sistem. Dalam diagram aktivitas ini, setiap aktivitas dijelaskan secara rinci, termasuk tindakan yang diambil, aliran kerja, serta hubungan dan ketergantungan antara satu aktivitas dengan aktivitas lainnya. Penjelasan yang mendalam ini membantu dalam memahami secara komprehensif bagaimana sistem beroperasi dan berinteraksi dengan aktor yang terlibat.

a. Mengelola Data Pengguna



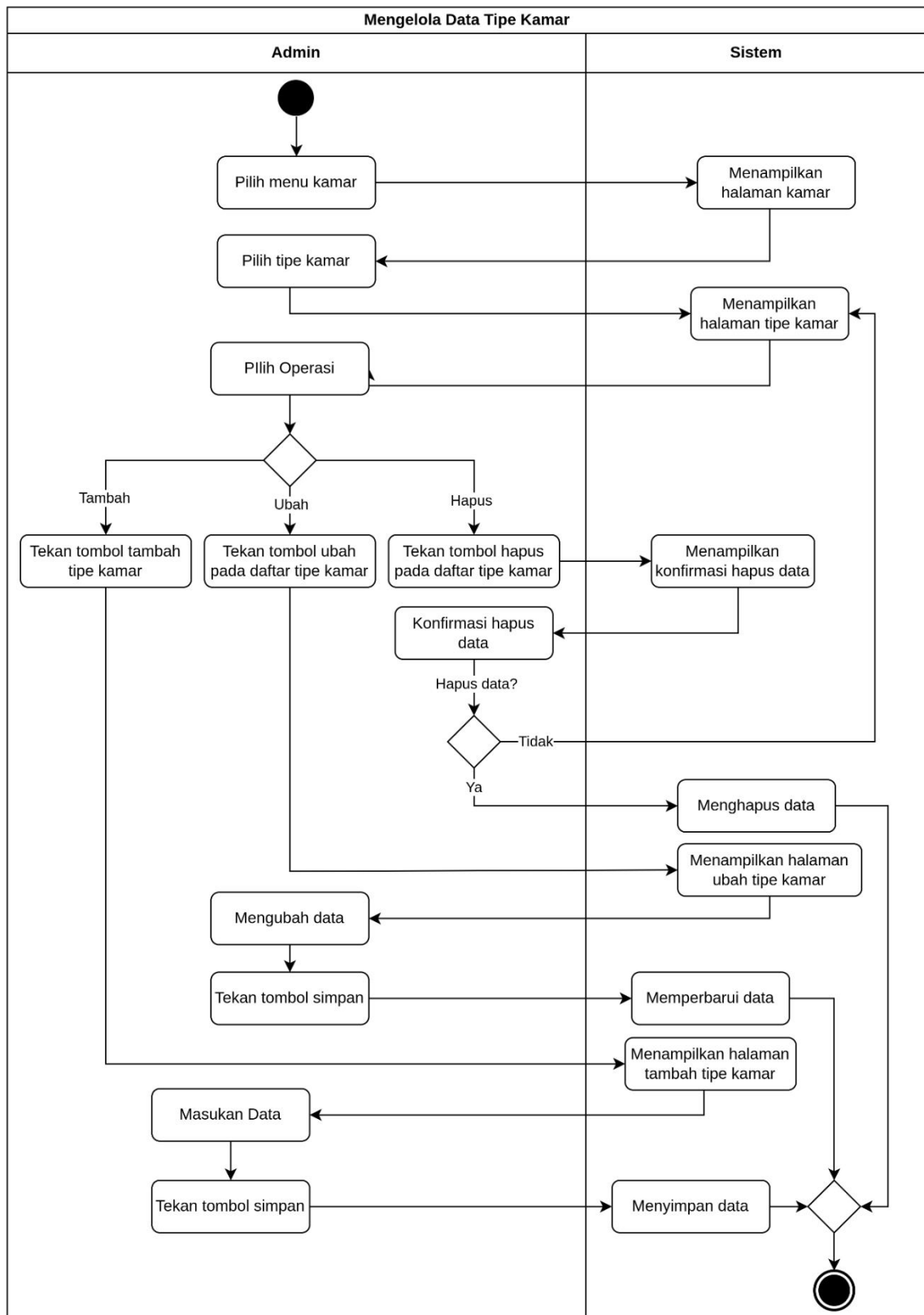
Gambar 3.3 Mengelola data pnegguna

Diagram activity "Mengelola Data Pengguna" menggambarkan alur proses yang dilakukan oleh seorang admin dalam mengelola data pengguna di sebuah sistem. Proses dimulai dengan admin memilih menu pengguna untuk memulai pengelolaan data. Setelah memilih menu, admin memiliki tiga pilihan: menambah, mengubah, atau menghapus data pengguna.

Jika admin memilih untuk menambah pengguna, sistem akan menampilkan form di mana admin dapat memasukkan data pengguna baru. Setelah data dimasukkan, admin menyimpan informasi tersebut, dan sistem menambahkan pengguna baru ke dalam *database*. Jika admin memilih untuk mengubah data pengguna, sistem akan menampilkan form dengan data pengguna tersebut. Admin kemudian dapat mengubah data yang diperlukan dan menyimpannya, setelah itu sistem memperbarui data di dalam *database*.

Jika admin memilih untuk menghapus pengguna, sistem akan meminta konfirmasi sebelum melanjutkan. Jika admin mengonfirmasi, sistem akan menghapus data pengguna yang dipilih dari *database*. Jika admin membatalkan konfirmasi, data tidak akan dihapus dan proses kembali ke daftar halaman pengguna.

b. Mengelola Data Tipe Kamar



Gambar 3.4 Mengelola data tipe kamar

Diagram activity "Mengelola Data Tipe Kamar" ini menjelaskan langkah-langkah yang diambil oleh seorang admin dalam mengelola data tipe kamar di sebuah sistem. Proses dimulai ketika admin memilih menu kamar kemudian memilih tipe kamar untuk mengakses data tipe kamar yang ada. Selanjutnya, admin memiliki tiga pilihan: menambah, mengubah, atau menghapus data tipe kamar.

Jika admin memilih untuk menambah tipe kamar, sistem akan menampilkan halaman tambah data tipe kamar untuk memasukkan data tipe kamar baru. Setelah data diisi, admin menyimpannya, dan sistem akan menambahkan tipe kamar baru tersebut ke dalam *database*. Jika admin memilih untuk mengubah tipe kamar, sistem akan menampilkan halaman yang memuat data tipe kamar tersebut. Admin dapat mengubah data yang diperlukan dan kemudian menyimpannya, setelah itu sistem akan memperbarui data tipe kamar di dalam *database*.

Jika admin memilih untuk menghapus tipe kamar, sistem akan menampilkan konfirmasi untuk memastikan bahwa admin benar-benar ingin menghapus data tersebut. Jika admin mengonfirmasi, sistem akan menghapus tipe kamar dari *database*. Jika admin membatalkan konfirmasi, data tidak akan dihapus dan proses kembali ke daftar tipe kamar.



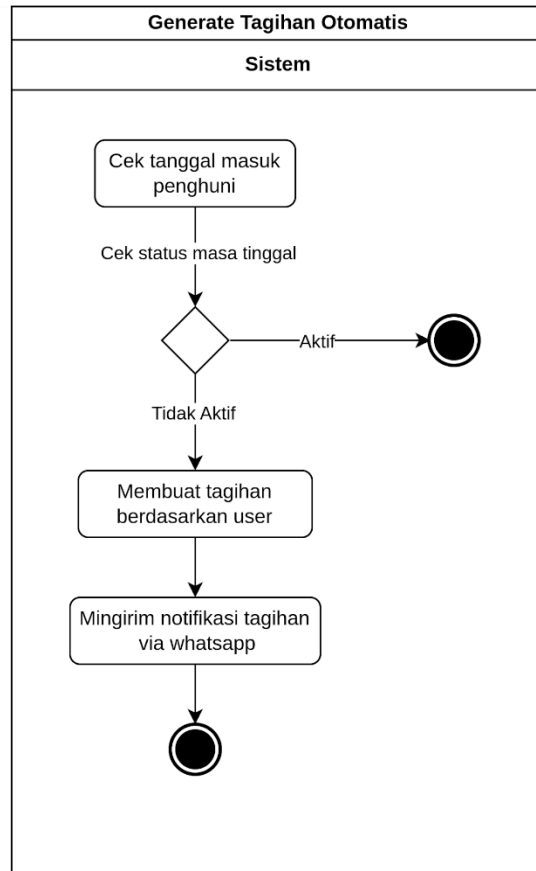
Diagram activity "Mengelola Data Kamar" ini menjelaskan alur proses yang dilakukan oleh seorang admin dalam mengelola data kamar di dalam sebuah sistem. Proses ini dimulai dengan admin memilih menu kamar dari antarmuka sistem. Setelah memilih menu kamar, admin memiliki tiga pilihan: menambah, mengubah, atau menghapus data kamar.

Jika admin memilih untuk menambah kamar, sistem akan menampilkan form di mana admin dapat memasukkan data kamar baru. Setelah data dimasukkan, admin menekan tombol simpan, dan sistem akan menambahkan data kamar baru tersebut ke dalam *database*.

Jika admin memilih untuk mengubah data kamar, sistem akan menampilkan form dengan data kamar tersebut. Admin kemudian dapat memperbarui informasi yang diperlukan, misalnya. Setelah perubahan dilakukan, admin menyimpan perubahan tersebut, dan sistem akan memperbarui data kamar di dalam *database*.

Jika admin memilih untuk menghapus data kamar, sistem akan menampilkan pesan konfirmasi untuk memastikan bahwa admin benar-benar ingin menghapus kamar tersebut. Jika admin mengonfirmasi, sistem akan menghapus data kamar dari *database*. Namun, jika admin membatalkan konfirmasi, data kamar tidak akan dihapus, dan proses akan kembali ke daftar kamar.

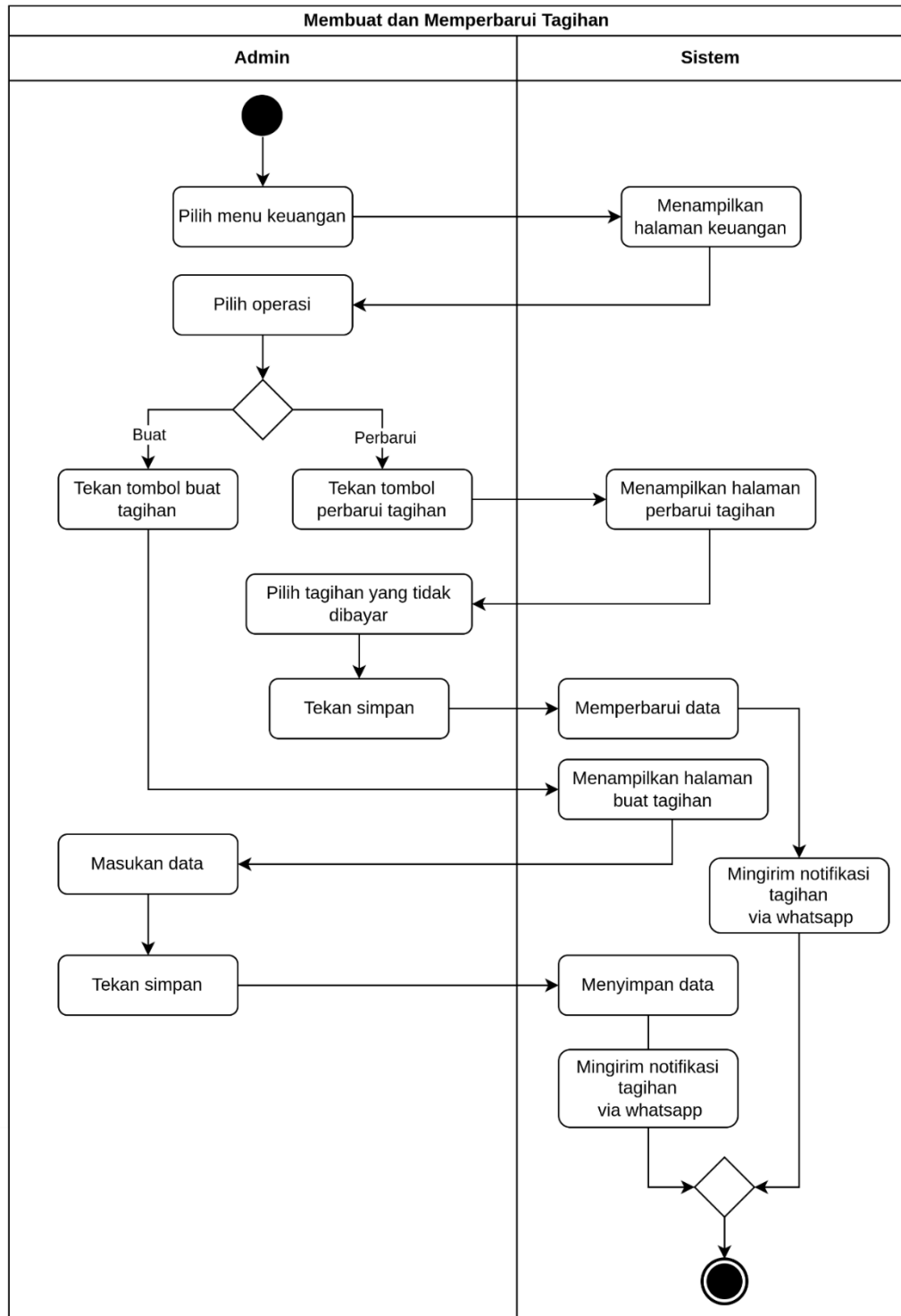
## d. Generate Tagihan Otomatis



Gambar 3.6 Generate tagihan otomatis

Diagram "Generate Tagihan Otomatis" menunjukkan bagaimana sistem secara otomatis mengelola pembuatan dan pengiriman tagihan kepada penghuni. Proses dimulai dengan sistem mengecek tanggal masuk setiap penghuni untuk menentukan status masa tinggal mereka. Jika masa tinggal sudah tidak aktif, sistem melanjutkan dengan membuat tagihan yang sesuai berdasarkan data penghuni yang tersimpan. Setelah tagihan dibuat, sistem secara otomatis mengirimkan notifikasi tagihan melalui WhatsApp ke penghuni tersebut. Jika masa tinggal masih aktif, proses dihentikan dan tagihan tidak dibuat. Alur ini memastikan bahwa tagihan dibuat dan dikirim secara efisien tanpa perlu campur tangan manual dari admin.

## e. Membuat dan Memperbarui Tagihan



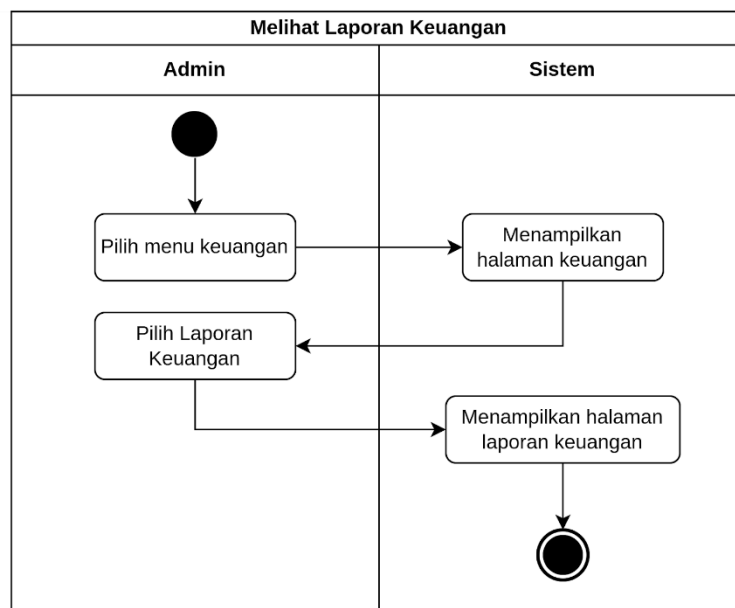
Gambar 3.7 Membuat dan memperbarui tagihan

Diagram "Membuat dan Memperbarui Tagihan" menggambarkan proses di mana admin dapat membuat atau memperbarui tagihan yang tidak dibayar dalam sistem. Proses dimulai

ketika admin memilih menu keuangan, lalu memilih operasi yang diinginkan, yaitu membuat atau memperbarui tagihan. Jika admin memilih untuk membuat tagihan, sistem akan menampilkan halaman pembuatan tagihan, di mana admin dapat memasukkan data yang diperlukan. Setelah data dimasukkan, admin menekan tombol simpan, dan sistem akan menyimpan tagihan serta mengirimkan notifikasi tagihan melalui WhatsApp ke penghuni.

Jika admin memilih untuk memperbarui tagihan, sistem akan menampilkan halaman pembaruan tagihan. Admin kemudian memilih tagihan yang akan diperbarui dan menekan tombol simpan. Sistem akan memperbarui tagihan serta mengirimkan notifikasi tagihan melalui WhatsApp ke penghuni. Proses ini digunakan untuk mengelola tagihan yang tidak dibayar atau sudah kadaluarsa serta untuk membuat tagihan di luar tagihan bulanan.

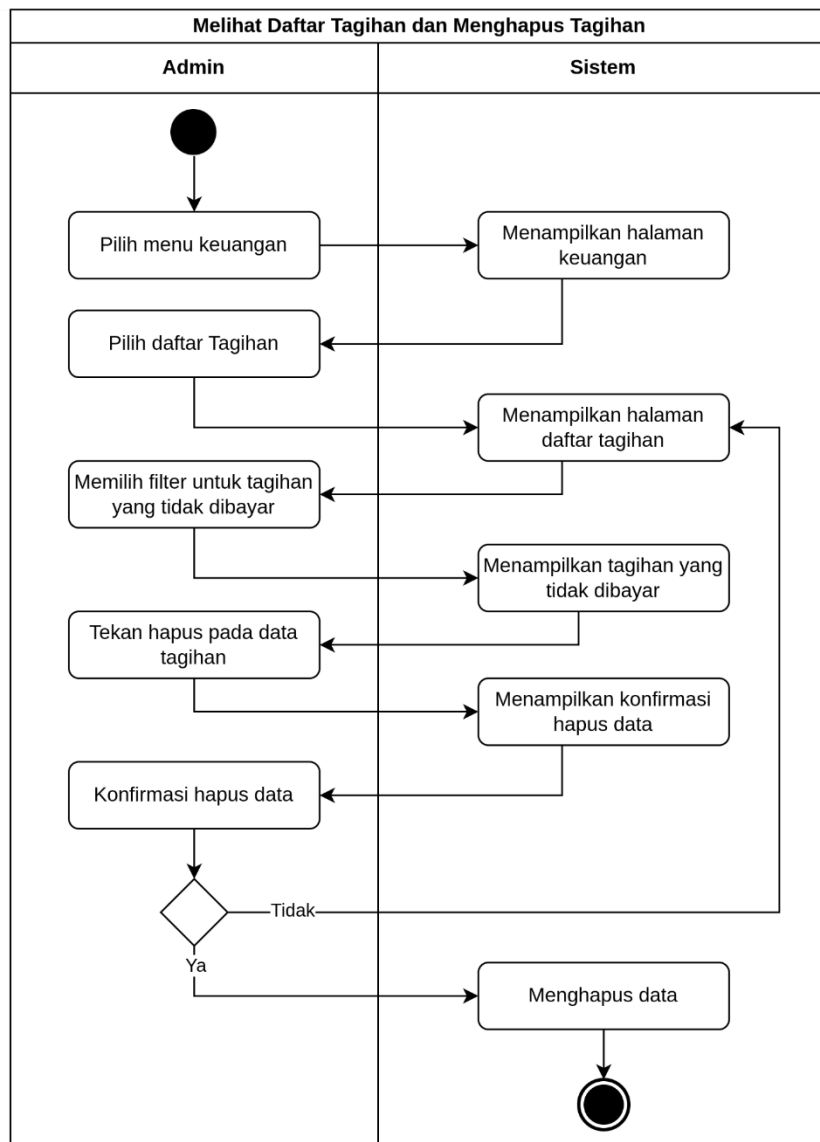
#### f. Melihat Laporan Keuangan



Gambar 3.8 Melihat laporan keuangan

Diagram "Melihat Laporan Keuangan" menunjukkan proses sederhana di mana admin mengakses laporan keuangan dalam sistem. Admin memilih menu keuangan, lalu sistem menampilkan halaman dengan opsi terkait keuangan. Admin memilih "Laporan Keuangan" dan sistem menampilkan laporan terkait tagihan penghuni kos.

## g. Melihat Daftar Tagihan dan Menghapus Tagihan

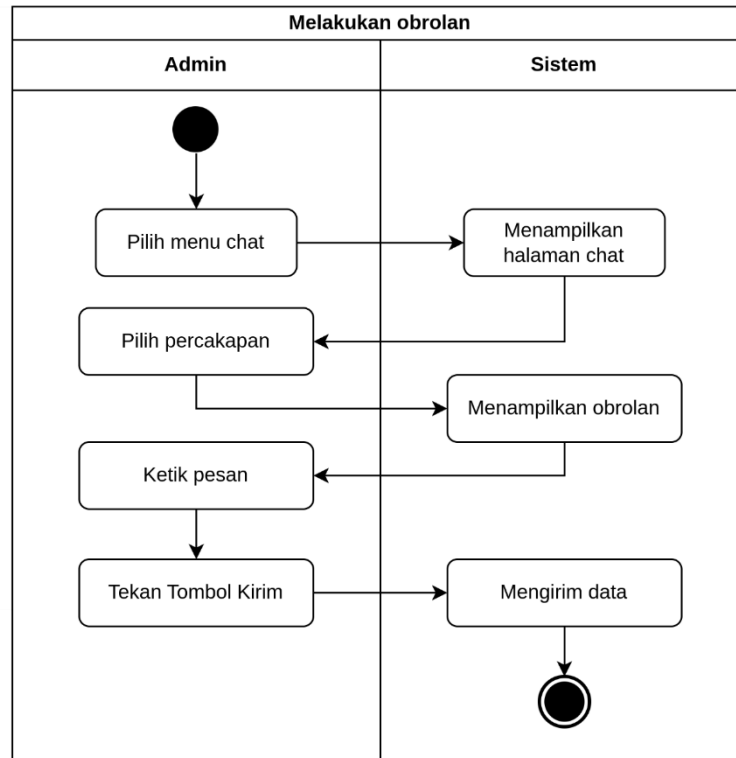


Gambar 3.9 Melihat daftar tagihan dan menghapus tagihan

Diagram "Melihat Daftar Tagihan dan Menghapus Tagihan" menunjukkan proses di mana admin mengakses dan mengelola tagihan dalam sistem. Admin memulai dengan memilih menu keuangan, dan sistem menampilkan halaman keuangan. Admin kemudian memilih "daftar tagihan," dan sistem menampilkan semua tagihan yang ada. Setelah itu, admin dapat memilih filter untuk melihat tagihan yang tidak dibayar, dan sistem akan menampilkan tagihan yang tidak dibayar. Jika admin ingin menghapus tagihan, admin dapat menekan opsi hapus pada data tagihan, yang kemudian diikuti dengan tampilan konfirmasi dari sistem. Setelah

admin mengkonfirmasi penghapusan, sistem akan menghapus data tagihan tersebut.

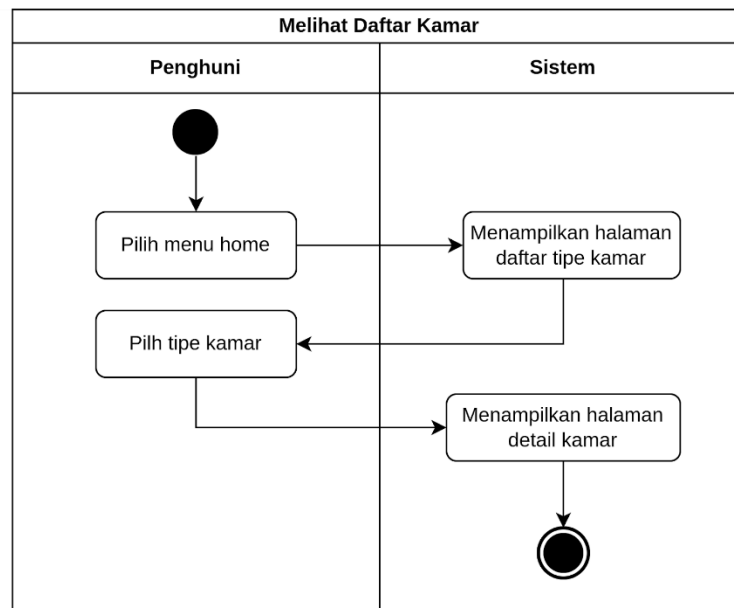
#### h. Melakukan Obrolan



Gambar 3.10 Melakukan obrolan

Diagram "Melakukan Obrolan" menjelaskan proses komunikasi melalui sistem chat yang dilakukan oleh admin atau penghuni. Proses dimulai ketika aktor (baik admin maupun penghuni) memilih menu chat dari antarmuka sistem. Setelah menu chat dipilih, sistem akan menampilkan halaman chat yang memungkinkan aktor untuk melihat percakapan yang ada. Aktor kemudian memilih percakapan yang ingin dilanjutkan. Sistem akan menampilkan obrolan yang terkait dengan pilihan tersebut, memungkinkan aktor untuk melanjutkan percakapan yang sudah ada atau memulai yang baru. Setelah percakapan dipilih, aktor mengetik pesan yang ingin disampaikan. Setelah pesan selesai diketik, aktor menekan tombol kirim, dan sistem akan mengirimkan pesan tersebut ke penerima yang dituju. Langkah-langkah ini memungkinkan admin dan penghuni untuk berinteraksi melalui sistem secara efisien dan terorganisir.

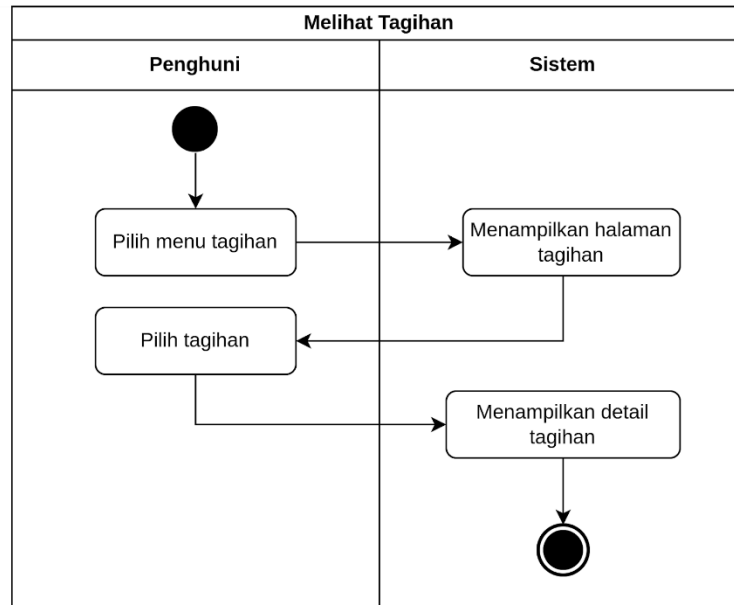
## i. Melihat Daftar Kamar



Gambar 3.11 Melihat daftar kamar

Diagram "Melihat Daftar Kamar" ini menunjukkan bagaimana seorang penghuni dapat melihat informasi kamar melalui sistem. Proses dimulai ketika penghuni memilih menu home. Setelah itu, sistem menampilkan daftar tipe kamar yang tersedia. Penghuni kemudian memilih tipe kamar yang ingin dilihat, dan sistem akan menampilkan halaman yang berisi detail dari kamar tersebut. Proses ini memungkinkan penghuni untuk dengan mudah melihat informasi lengkap mengenai tipe kamar yang tersedia.

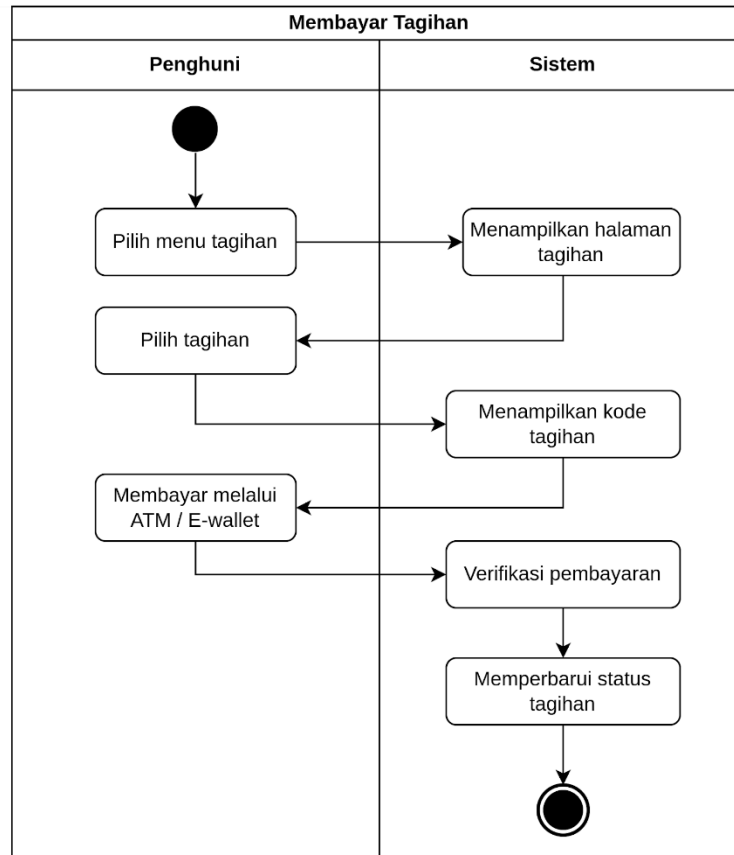
## j. Melihat Tagihan



Gambar 3.12 Melihat tagihan

Diagram "Melihat Tagihan" ini menggambarkan bagaimana seorang penghuni dapat melihat rincian tagihan melalui sistem. Proses dimulai dengan penghuni memilih menu tagihan. Setelah itu, sistem akan menampilkan halaman yang berisi daftar tagihan yang ada. Penghuni kemudian memilih tagihan yang ingin dilihat, dan sistem akan menampilkan detail lengkap dari tagihan tersebut. Alur ini memungkinkan penghuni untuk dengan mudah mengakses dan memeriksa rincian tagihan yang perlu dibayar.

## k. Membayar Tagihan



Gambar 3.13 Membayar tagihan

Diagram "Membayar Tagihan" ini menjelaskan alur proses yang dilakukan oleh penghuni saat membayar tagihan melalui sistem. Proses dimulai ketika penghuni memilih menu tagihan pada antarmuka sistem. Setelah itu, sistem menampilkan halaman yang berisi daftar tagihan yang perlu dibayar. Penghuni kemudian memilih tagihan yang ingin dibayarkan, dan sistem menampilkan kode tagihan yang diperlukan untuk proses pembayaran. Penghuni melanjutkan dengan melakukan pembayaran melalui ATM atau *e-wallet*, menggunakan kode tagihan yang telah diberikan. Setelah pembayaran dilakukan, sistem akan memverifikasi transaksi tersebut. Jika verifikasi berhasil, sistem secara otomatis memperbarui status tagihan menjadi lunas.

### 3.2.3 Perancangan Progressive Web App

#### a. Desain Responsif

Desain responsif merupakan aspek kunci dalam pembangunan sebuah website, terutama untuk memastikan bahwa tampilan dan tata letak situs web dapat menyesuaikan diri dengan berbagai ukuran layar dan perangkat, termasuk ponsel pintar dan tablet. Desain yang responsif memungkinkan pengguna untuk mengakses dan menjelajahi situs web secara optimal tanpa mengalami kendala visual atau fungsionalitas, terlepas dari perangkat yang digunakan.

#### b. Manifest File

Manifest file, atau `manifest.json`, adalah komponen esensial dalam pengembangan PWA. File ini menyediakan informasi meta terkait aplikasi yang membantu menentukan tampilan dan perilaku aplikasi ketika diakses oleh pengguna. Di dalam manifest file, terdapat detail seperti ikon aplikasi, nama aplikasi, deskripsi singkat, dan tema warna yang akan diterapkan. Manifest file juga memungkinkan PWA diinstal di perangkat pengguna layaknya aplikasi native, dengan ikon aplikasi muncul di layar utama perangkat. Dengan adanya manifest file, PWA dapat memiliki identitas visual yang konsisten dan menarik bagi pengguna.

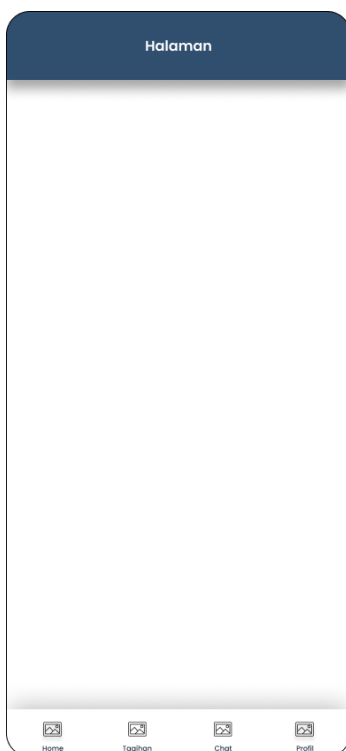
Manifest file berisi berbagai detail penting yang harus disertakan untuk memperkaya pengalaman pengguna dalam aplikasi. Beberapa elemen yang perlu dimasukkan dalam manifest file antara lain: nama aplikasi, nama pendek aplikasi, URL awal, warna latar belakang, warna tema, dan mode tampilan aplikasi dengan mode “standalone” agar aplikasi terlihat seperti aplikasi native tanpa elemen browser. Selain itu, ikon-ikon aplikasi perlu disediakan dalam berbagai ukuran, mulai dari 72x72 piksel hingga 512x512 piksel, untuk memastikan tampilan yang optimal di berbagai perangkat. Gambar “splash” juga harus disiapkan untuk berbagai resolusi layar.

### 3.2.4 Perancangan Inteface

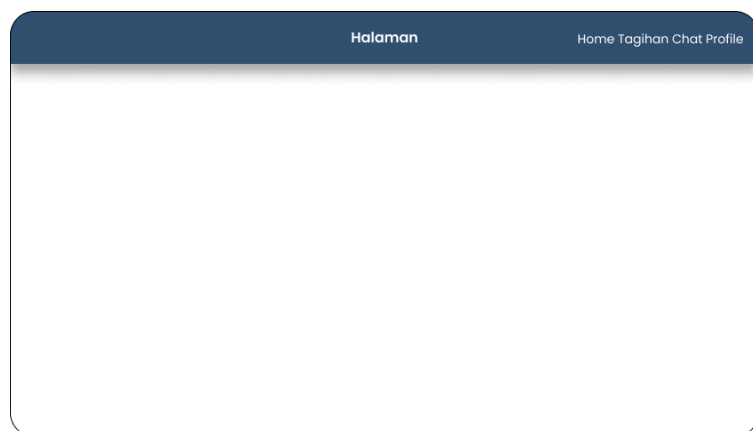
Merancang antarmuka merupakan aspek penting dalam proses perancangan sistem informasi. Tujuan utama dari perancangan antarmuka adalah untuk mempermudah pengembangan sistem dan memberikan gambaran yang jelas tentang implementasi sistem yang akan dilakukan. Perancangan antarmuka dalam penelitian ini disusun berdasarkan kebutuhan aktor yang akan menggunakan sistem.

Dengan penerapan Progressive Web App, rancangan antarmuka harus responsif agar memberikan pengalaman pengguna yang mirip dengan aplikasi native, baik di perangkat mobile maupun desktop. Oleh karena itu, saat sistem diakses melalui desktop, navigasi ditempatkan di bagian atas, sedangkan saat diakses melalui perangkat mobile, navigasi dipindahkan ke bagian bawah. Penempatan navigasi ini dirancang untuk memaksimalkan kemudahan akses dan kenyamanan pengguna, sesuai dengan rancangan antarmuka yang telah dibuat.

Rancangan interface ini mengikuti pendekatan *mobile first*, di mana pengembangan antarmuka dimulai dari desain untuk perangkat mobile terlebih dahulu. Dengan metode ini, navigasi dan elemen penting lainnya ditempatkan di lokasi yang mudah dijangkau oleh pengguna perangkat mobile, seperti di bagian bawah layar. Setelah rancangan untuk mobile selesai, interface diperluas untuk mencakup perangkat dengan layar yang lebih besar seperti tablet dan desktop, di mana navigasi dapat ditempatkan di bagian atas untuk mengikuti kebiasaan pengguna pada perangkat tersebut. Pendekatan ini memastikan bahwa antarmuka tetap intuitif dan efisien, terlepas dari perangkat yang digunakan.



Gambar 3.14 Mobile



Gambar 3.15 Desktop

## A. Interface Admin

### a. Halaman Dashboard

Halaman ini menampilkan berbagai visualisasi data yang penting, seperti grafik pendapatan bulanan, persentase pembayaran kos untuk bulan tersebut, total pendapatan bulanan, grafik jumlah kamar yang terisi setiap bulan, dan jumlah kamar yang tersedia. Dapat dilihat pada Gambar 3.16.

### b. Halaman Menu Kamar

Halaman ini menampilkan data semua kamar yang tersedia, dengan opsi untuk memfilter berdasarkan tipe dan status kamar, serta fitur pencarian. Pada halaman ini, terdapat tombol "Tambah Kamar," "Ubah," "Hapus," dan "Detail," yang masing-masing akan menampilkan popup untuk melakukan aksi terkait. Dapat dilihat pada Gambar 3.17, Gambar 3.18, dan Gambar 3.19

### c. Halaman Tipe Kamar

Halaman ini menampilkan daftar tipe-tipe kamar yang ada. Tersedia tombol "Tambah Tipe Kamar," "Ubah," "Hapus," dan "Detail" untuk mengelola data tipe kamar. Dapat dilihat pada Gambar 3.20.

### d. Halaman Tambah dan Perbarui Tipe Kamar

Halaman ini memungkinkan admin untuk menambah atau memperbarui tipe kamar, tergantung pada aksi sebelumnya. Admin harus mengisi informasi yang diperlukan, dan setelah selesai, data kamar bisa disimpan dengan menekan tombol "Simpan". Dapat dilihat pada Gambar 3.21.

### e. Halaman Menu Keuangan

Halaman ini menampilkan visualisasi data keuangan, serta menyediakan menu untuk melihat status pembayaran per kamar, laporan keuangan, daftar transaksi, dan membuat tagihan. Dapat dilihat pada Gambar 3.22.

### f. Halaman Status Pembayaran

Halaman ini menampilkan data pembayaran untuk semua kamar. Kamar yang tidak terisi tidak akan ditampilkan. Kotak berwarna biru tua menunjukkan bahwa pembayaran sudah dilakukan, sementara kotak berwarna biru muda menunjukkan bahwa pembayaran belum dilakukan. Dapat dilihat pada Gambar 3.23 dan Gambar 3.24.

### g. Halaman Laporan Keuangan

Halaman ini menampilkan laporan keuangan yang dapat di filter berdasarkan bulan dan tahun. Terdapat juga tombol untuk mengeksport data. Dapat dilihat pada Gambar 3.25.

h. Halaman Daftar Transaksi

Halaman ini menampilkan seluruh data transaksi yang berasal dari tagihan kamar kos, dengan status apakah tagihan sudah dibayar, belum dibayar, atau tidak dibayar. Dapat dilihat pada Gambar 3.26 dan Gambar 3.27.

i. Halaman Buat dan Perbarui Tagihan

Halaman ini memungkinkan admin untuk membuat tagihan tambahan selain tagihan bulanan yang dihasilkan secara otomatis, serta memperbarui tagihan yang belum dibayar, dengan memperbarui tenggat pembayaran dan kode pembayaran. Dapat dilihat pada Gambar 3.28 dan Gambar 3.29.

j. Halaman Menu Chat

Halaman ini menampilkan daftar pengguna yang masih aktif dan grup chat. Pengguna atau grup yang dapat dihubungi ditampilkan di sini. Dapat dilihat pada Gambar 3.30.

k. Halaman Chat

Halaman ini menampilkan percakapan antara pengguna dengan seseorang yang dipilih pada halaman Menu Chat. Pengguna dapat membaca dan mengirim pesan dalam obrolan yang sedang berlangsung. Dapat dilihat pada Gambar 3.31.

l. Halaman Menu Pengguna

Halaman ini menampilkan data seluruh penghuni, dengan opsi untuk memfilter berdasarkan status aktif atau tidak aktif, serta fitur pencarian. Pada halaman ini terdapat tombol untuk "Tambah Pengguna," "Ubah," "Hapus," dan "Detail," yang semuanya akan membuka popup untuk melakukan aksi terkait. Dapat dilihat pada Gambar 3.32, Gambar 3.33, dan Gambar 3.34.

m. Halaman Admin

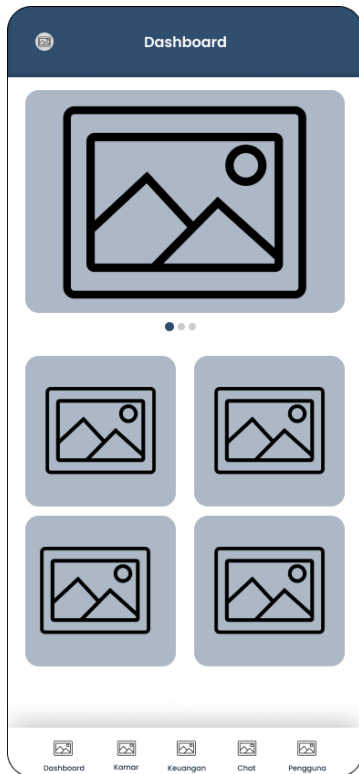
Halaman ini khusus untuk mengelola akun dengan role sebagai admin. Pada halaman ini terdapat tombol "Tambah Admin," "Ubah," dan "Hapus" untuk mengelola akun admin. Dapat dilihat pada Gambar 3.35 dan Gambar 3.36.

n. Halaman Profil

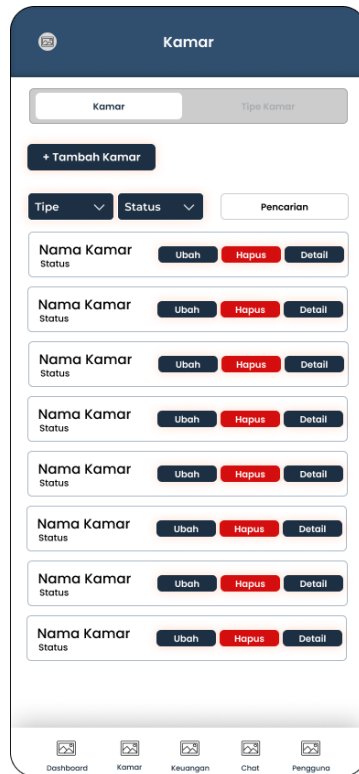
Halaman ini menampilkan informasi akun yang sedang login, serta menyediakan pengaturan dan pengelolaan data terkait sistem. Dapat dilihat pada Gambar 3.37.

o. Halaman Perbarui Data Sistem

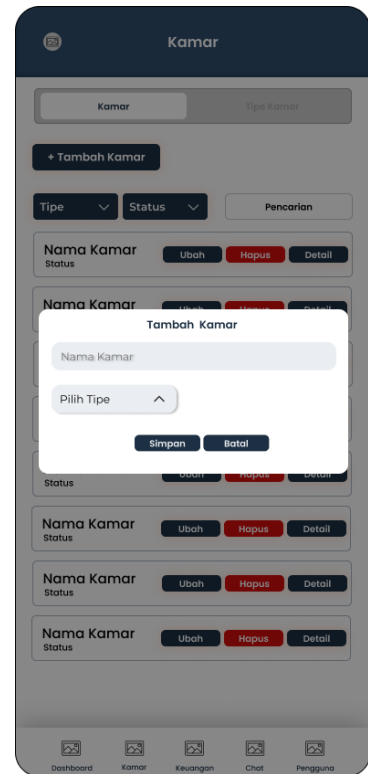
Halaman ini digunakan untuk memperbarui atau mengubah data sistem yang mencakup deskripsi, fasilitas, aturan dan tentang kami. Dapat dilihat pada Gambar 3.38.



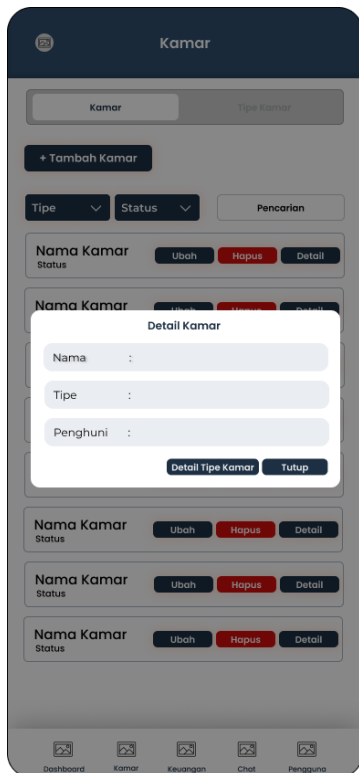
Gambar 3.16 Dashboard



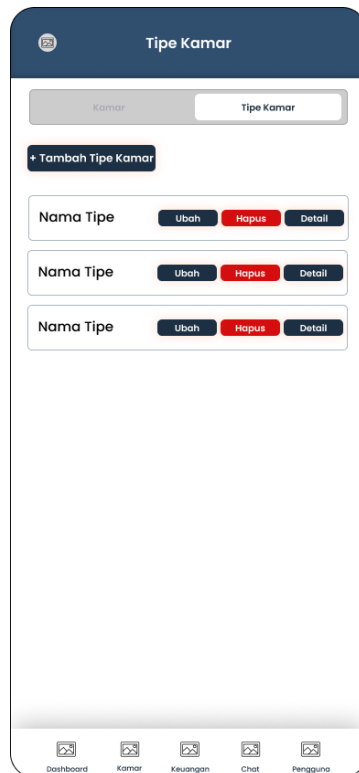
Gambar 3.17 Daftar kamar



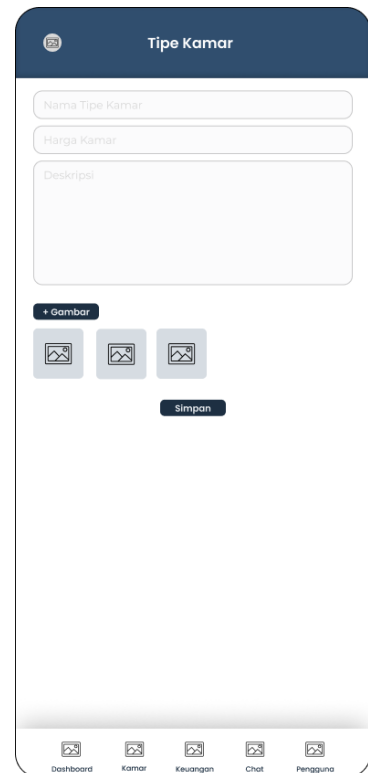
Gambar 3.18 Tambah kamar



Gambar 3.19 Detail kamar



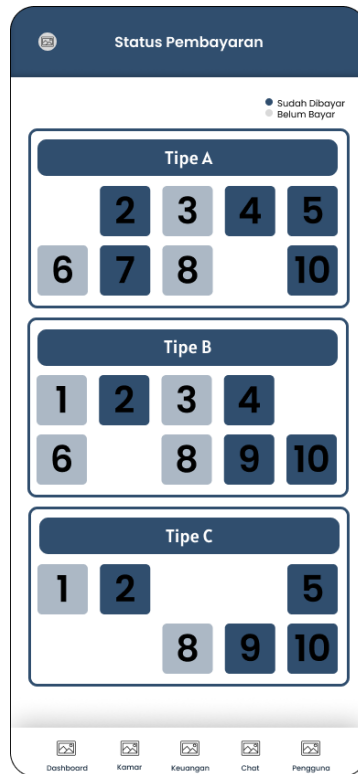
Gambar 3.20 Tipe kamar



Gambar 3.21 Tambah dan perbarui tipe kamar



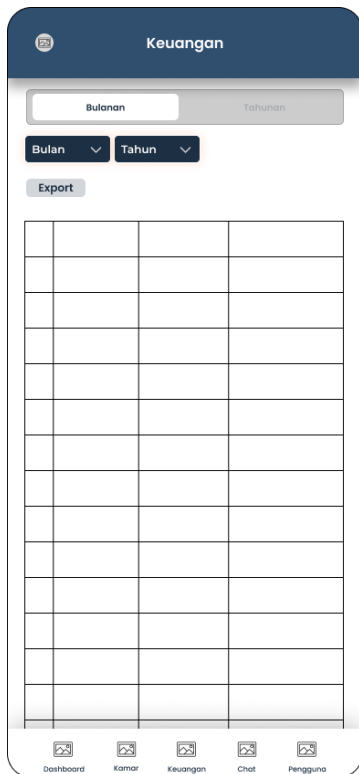
Gambar 3.22 Keuangan



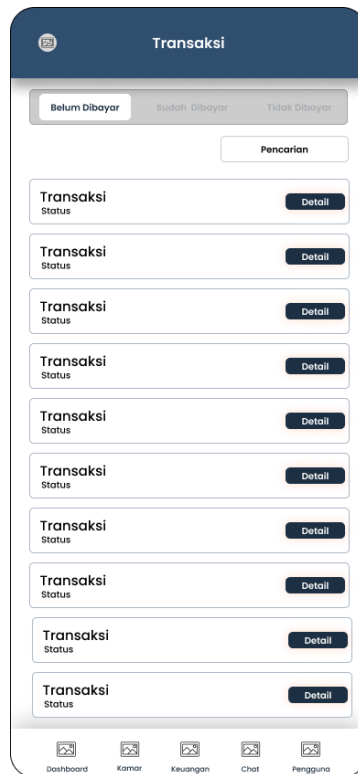
Gambar 3.23 Status Pembayaran



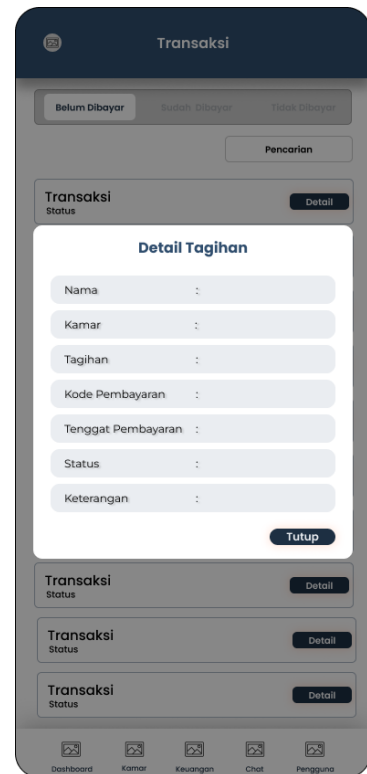
Gambar 3.24 Detail Status Pembayaran



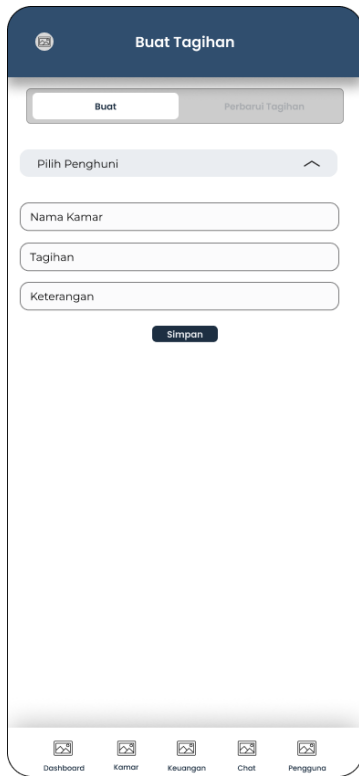
Gambar 3.25 Keuangan



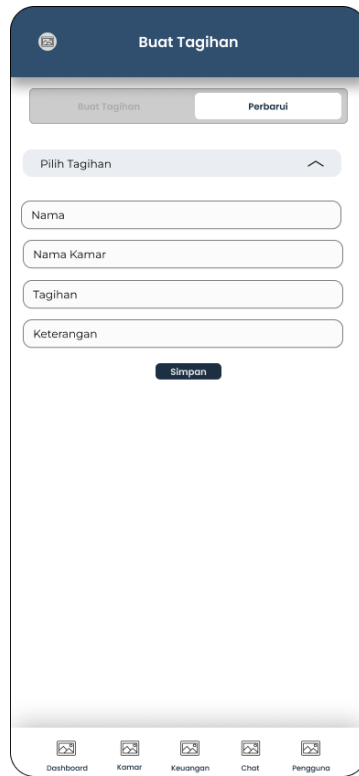
Gambar 3.26 Transaksi



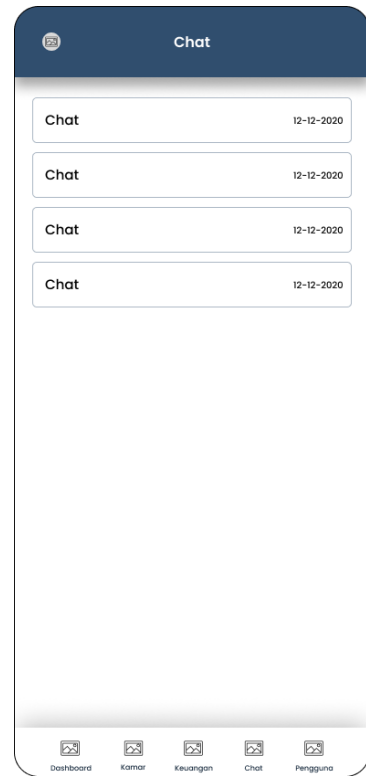
Gambar 3.27 Detail Transaksi



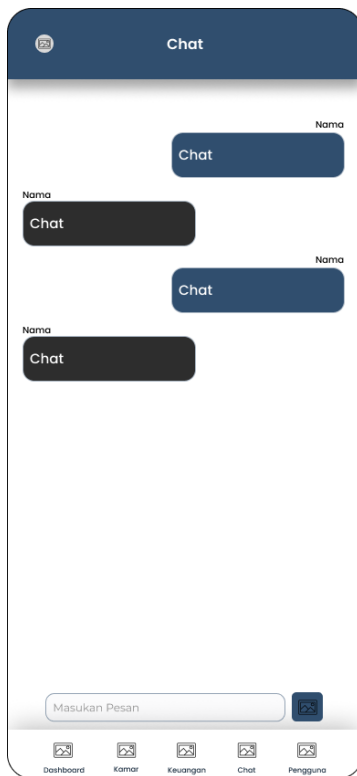
Gambar 3.28 Buat tagihan  
IIII



Gambar 3.29 Perbarui tagihan



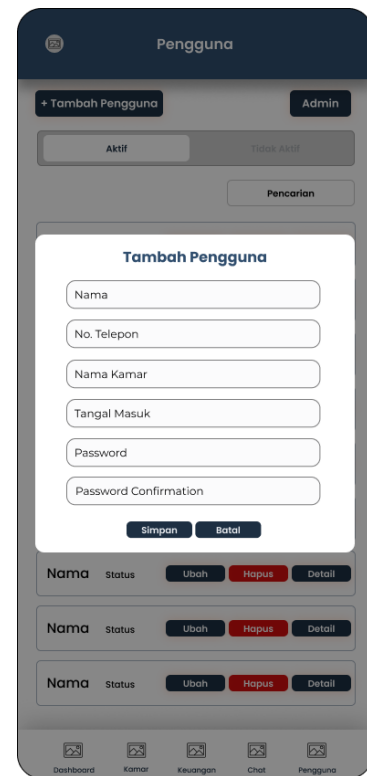
Gambar 3.30 Menu chat



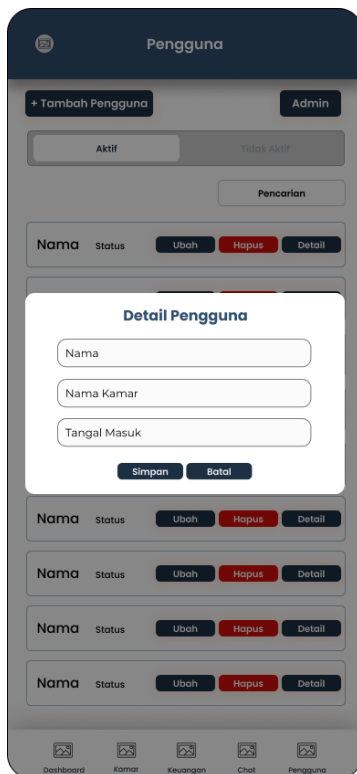
Gambar 3.31 Chat



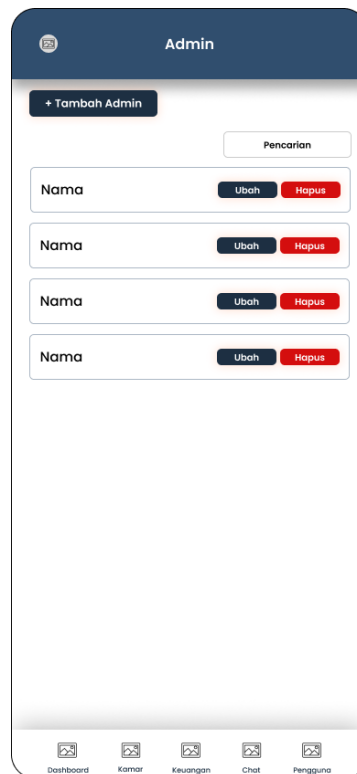
Gambar 3.32 Pengguna



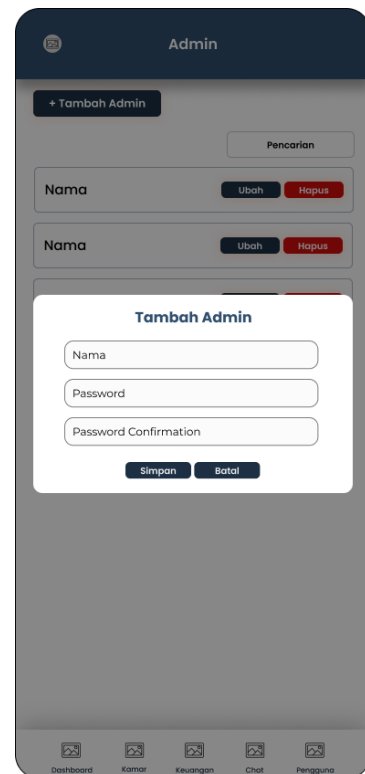
Gambar 3.33 Tambah dan perbarui pengguna



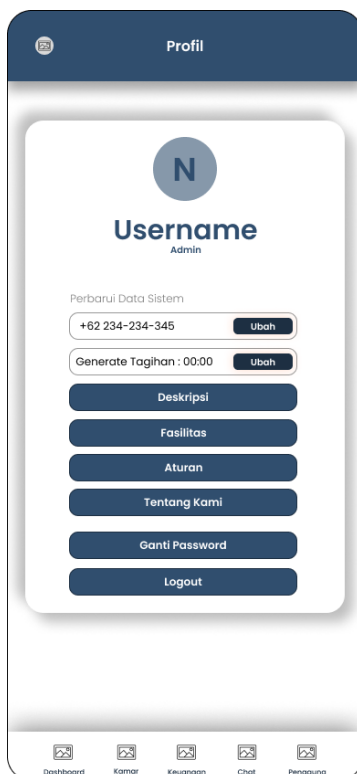
Gambar 3.34 Detail pengguna



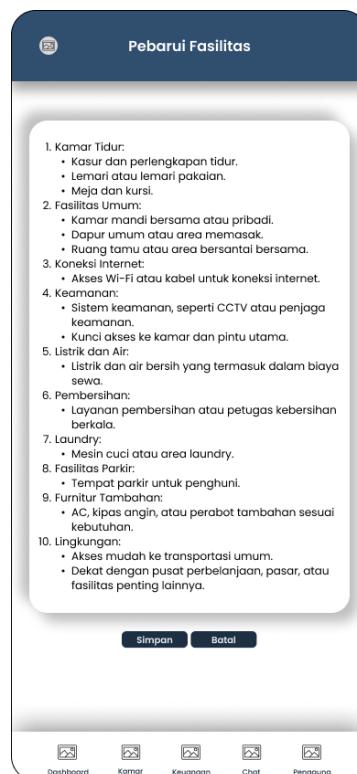
Gambar 3.35 Admin



Gambar 3.36 Tambah dan perbarui admin



Gambar 3.37 Profil



Gambar 3.38 Perbarui sistem

## B. Interface Penghuni

### a. Halaman Home

Halaman Home dapat diakses tanpa perlu login. Pada halaman ini, tersedia beberapa menu yang dapat diakses, seperti "Kamar", "Fasilitas", "Aturan" dan "Tentang Kami". Dapat dilihat pada Gambar 3.39.

### b. Halaman Daftar Tipe Kamar

Halaman Daftar Tipe Kamar juga dapat diakses tanpa perlu login. Halaman ini menampilkan berbagai tipe kamar yang tersedia, lengkap dengan informasi mengenai status ketersediaan dan harga setiap tipe kamar. Dapat dilihat pada Gambar 3.40.

### c. Halaman Detail Kamar

Halaman Detail Kamar dapat diakses tanpa perlu login. Halaman ini memberikan informasi detail tentang kamar berdasarkan tipe yang dipilih, termasuk daftar kamar yang tersedia beserta statusnya, foto kamar, dan daftar fasilitas yang disediakan. Dapat dilihat pada Gambar 3.41.

### d. Halaman Menu Tagihan

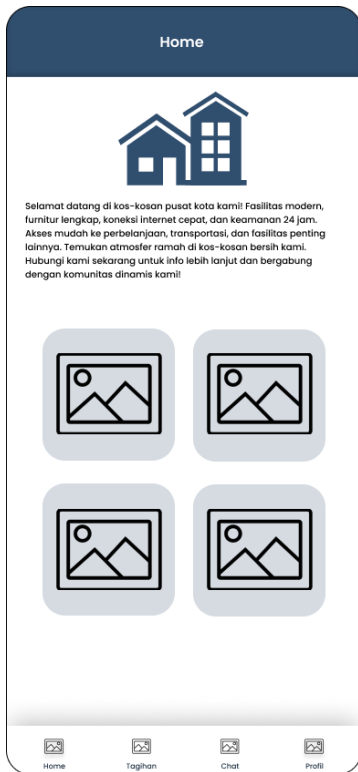
Halaman ini menampilkan daftar tagihan bagi penghuni kos, baik yang sudah terbayar maupun yang belum terbayar. Pada halaman ini, terdapat tombol "Detail" yang memungkinkan pengguna untuk melihat rincian tagihan kos, termasuk kode pembayaran dan tenggat waktu pembayaran. Dapat dilihat pada Gambar 3.42 dan Gambar 3.43.

### e. Halaman Chat

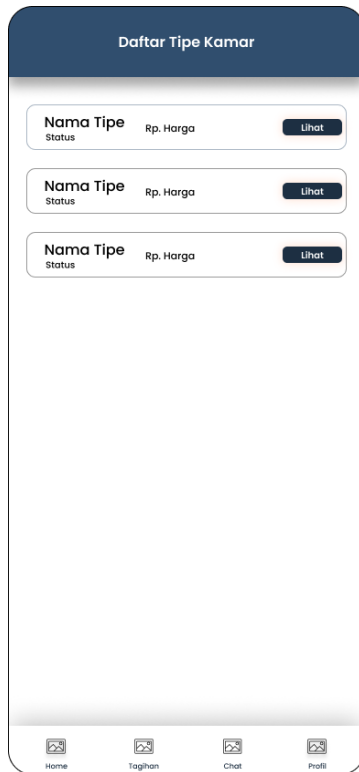
Halaman ini menyediakan dua jenis fitur chat, yaitu chat dengan admin dan chat grup. Pengguna dapat berinteraksi langsung dengan admin untuk keperluan komunikasi pribadi, serta berpartisipasi dalam diskusi grup yang melibatkan seluruh pengguna. Dapat dilihat pada Gambar 3.44.

### f. Halaman Menu Profil

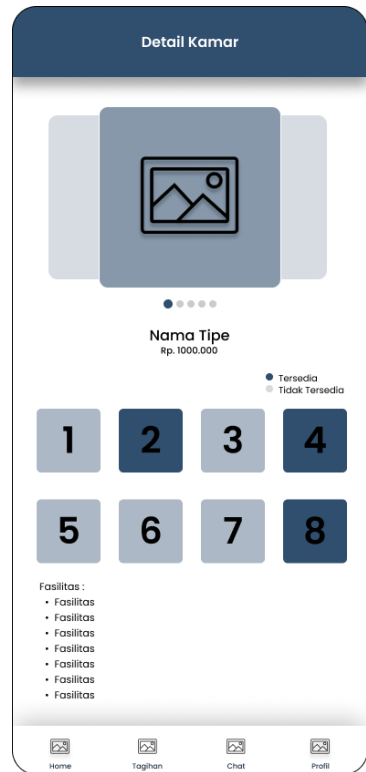
Halaman ini berisi informasi terkait username pengguna dan informasi akun yang sedang login. Selain itu, halaman ini juga menyediakan menu untuk mengubah nomor telepon dan password, serta tombol untuk logout dari akun. Dapat dilihat pada Gambar 3.45.



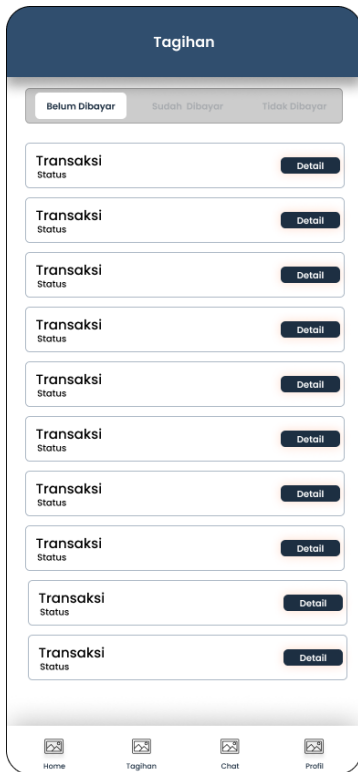
Gambar 3.39 Home



Gambar 3.40 Daftar tipe kamar



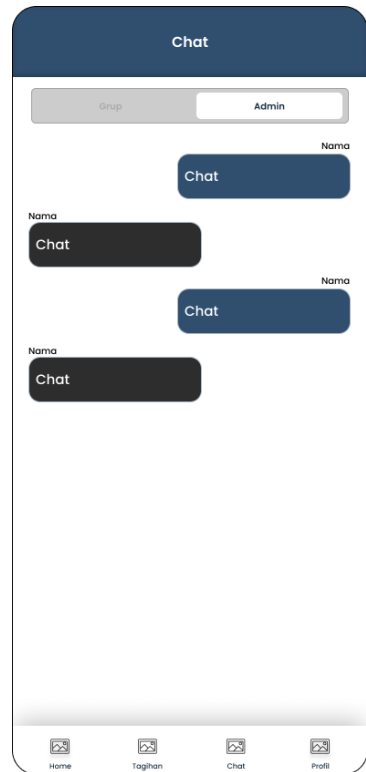
Gambar 3.41 Detail kamar



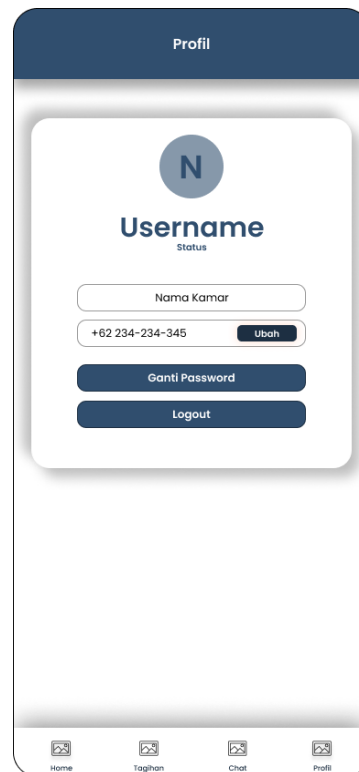
Gambar 3.42 Tagihan



Gambar 3.43 Detail tagihan



Gambar 3.44 Chat



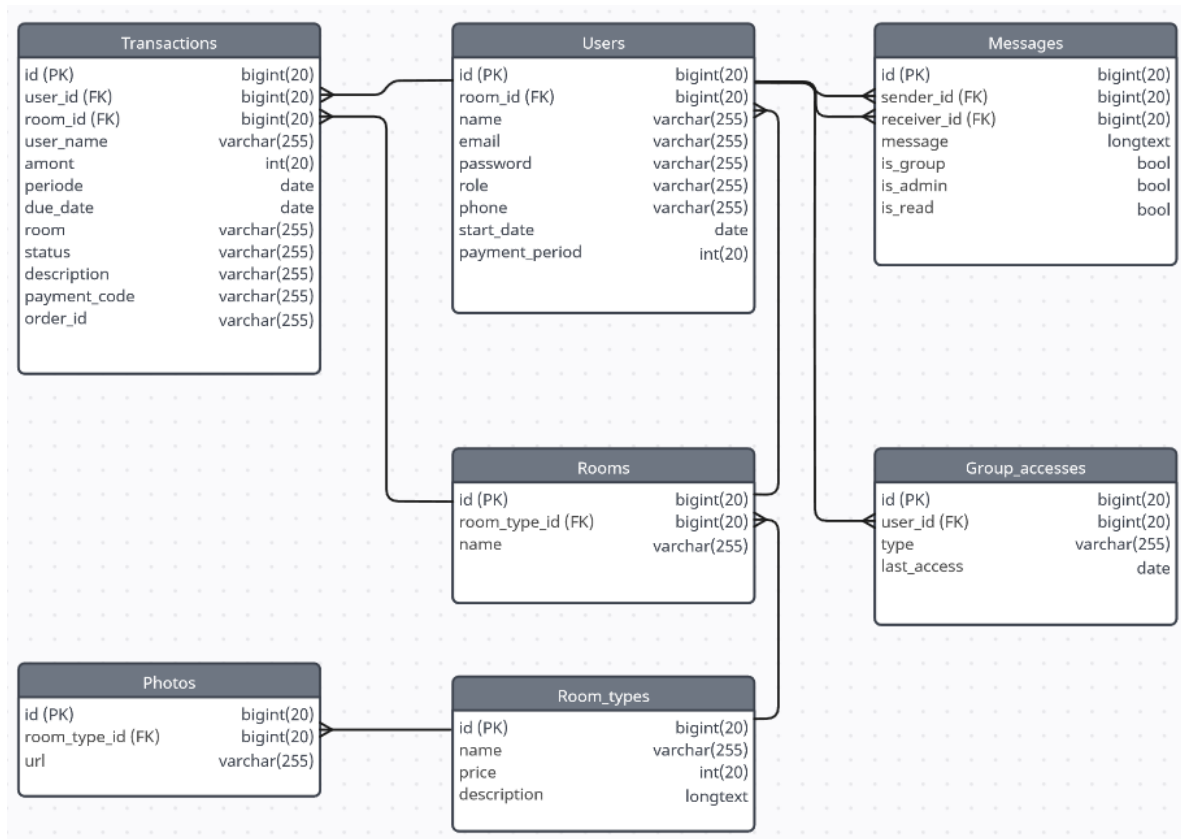
Gambar 3.45 Profil

### 3.2.5 Perancangan Database

Perancangan *database* merupakan proses penyusunan dan penentuan struktur penyimpanan data yang optimal untuk mendukung fungsionalitas sistem secara efisien. Tujuan dari perancangan *database* adalah untuk menciptakan basis data yang dapat menyimpan, mengelola, dan mengakses informasi dengan efektif.

#### A. Perancangan Relasi Antar Tabel

Perancangan relasi antar tabel merujuk pada proses merancang struktur dan hubungan antara tabel-tabel dalam basis data relasional. Tujuan utama dari perancangan relasi antar tabel adalah menciptakan struktur basis data yang efisien dan konsisten.



Gambar 3.46 Relasi antar tabel

## B. Perancangan Struktur Tabel

Perancangan struktur tabel merujuk pada proses merancang tabel-tabel dalam basis data. Setiap tabel dalam basis data memiliki struktur yang terdiri dari kolom atau atribut yang merepresentasikan jenis data yang berbeda dan baris atau catatan yang menyimpan data konkret.

### a. Tabel Users

Tabel users berfungsi sebagai tempat penyimpanan data pengguna, meliputi penghuni dan admin.

Tabel 3.3 Users

No.	Nama Kolom	Type Data	Constraint
1	id	bigint (20)	Primary key
2	room_id	bigint (20)	Foreign key
3	name	varchar (255)	
4	password	varchar (255)	
5	role	varchar (255)	
6	phone	varchar (255)	

7	start_date	date	
8	payment_period	int (20)	

b. Tabel Rooms

Tabel kamar berfungsi sebagai tempat penyimpanan data kamar.

Tabel 3.4 Rooms

No.	Nama Kolom	Tipe Data	Constraint
1	id	bigint (20)	Primary key
2	room_type_id	bigint (20)	Foreign key
3	name	varchar (255)	

c. Tabel Room\_types

Table tipe berfungsi sebagai tempat penyimpanan data tipe kamar.

Tabel 3.5 Room types

No.	Nama Kolom	Tipe Data	Constraint
1	id	bigint (20)	Primary key
2	name	varchar (255)	
3	price	int (20)	
4	description	longtext	

d. Tabel Photos

Tabel gambar berfungsi sebagai tempat penyimpanan data url gambar.

Tabel 3.6 Photos

No.	Nama Kolom	Tipe Data	Constraint
1	id	bigint (20)	Primary key
2	room_type_id	bigint (20)	Foreign key
3	url	varchar (255)	

e. Tabel Transactions

Table transaction berfungsi sebagai tempat penyimpanan data transaction.

Tabel 3.7 Transactions

No.	Nama Kolom	Tipe Data	Constraint
1	id	bigint (20)	Primary key
3	room_id	bigint (20)	Foreign key
2	user_id	bigint (20)	Foreign key
4	user_name	varchar (255)	
5	amount	int (20)	
6	periode	date	

7	due_date	date	
8	room	varchar (255)	
9	status	varchar (255)	
10	description	varchar (255)	
11	payment_code	varchar (255)	
12	order_id	varchar (255)	

f. Tabel Messages

Table transaction berfungsi sebagai tempat penyimpanan data message.

Tabel 3.8 Messages

No.	Nama Kolom	Tipe Data	Constraint
1	id	bigint (20)	Primary key
3	sender_id	bigint (20)	Foreign key
2	receiver_id	bigint (20)	Foreign key
4	message	longtext	
5	is_group	bool	
6	is_admin	bool	
7	is_read	bool	

g. Tabel Group\_accesses

Tabel group\_accesses berfungsi untuk menyimpan data mengenai waktu terakhir pengguna mengakses grup, data tersebut berfungsi sebagai indikator apakah pengguna sudah membaca chat terbaru atau belum.

Tabel 3.9 Group accesses

No.	Nama Kolom	Tipe Data	Constraint
1	id	bigint (20)	Primary key
2	user_id	bigint (20)	Foreign key
3	type	varchar (255)	
4	last_access	date	

## **BAB IV**


### **IMPLEMENTASI HASIL DAN ANALISIS**


#### **4.1 Hasil Penelitian**

Hasil penelitian ini merupakan tahapan lanjutan setelah perancangan sistem informasi berbasis Progressive Web App (PWA) untuk sistem informasi kos-kosan. Penelitian ini mencakup berbagai aspek implementasi serta pengujian terhadap sistem yang telah dikembangkan. Berikut ini dijelaskan secara rinci hasil-hasil yang diperoleh dari pengembangan sistem informasi tersebut.

##### **4.1.1 Implementasi Progeressive Web App**

Sistem yang dikembangkan telah berhasil mengimplementasikan Progressive Web App (PWA), baik untuk versi desktop maupun mobile. Pada kedua platform ini, sistem dapat diakses dengan tampilan yang menyerupai aplikasi native. Hal ini tercermin dari antarmuka yang sederhana dan fungsional, memberikan pengalaman yang intuitif dan efisien bagi pengguna. Tidak ada lagi elemen-elemen khas browser seperti tab atau toolbar, sehingga penggunaan aplikasi terasa lebih fokus dan lebih sederhana, seolah-olah menggunakan aplikasi native yang diunduh dari toko aplikasi.

Pada versi desktop, sistem dapat diakses layaknya sebuah aplikasi native tanpa perlu masuk melalui browser. Setelah instalasi PWA, pengguna cukup meluncurkan aplikasi langsung dari desktop tanpa harus membuka peramban terlebih dahulu, yang mana ini memberikan pengalaman yang lebih praktis dan cepat. Pengguna akan merasakan aplikasi ini seperti aplikasi desktop pada umumnya, dengan performa yang stabil dan user interface yang responsif terhadap berbagai ukuran layar. Dapat dilihat pada .

Di sisi mobile, sistem ini juga mendukung akses langsung dari home screen perangkat, tanpa perlu membuka browser. Dengan kemampuan PWA, pengguna bisa “menginstal” aplikasi langsung dari browser ke layar utama perangkat seluler mereka, memberikan pengalaman yang sama seperti aplikasi native di perangkat mobile. Dapat dilihat pada .



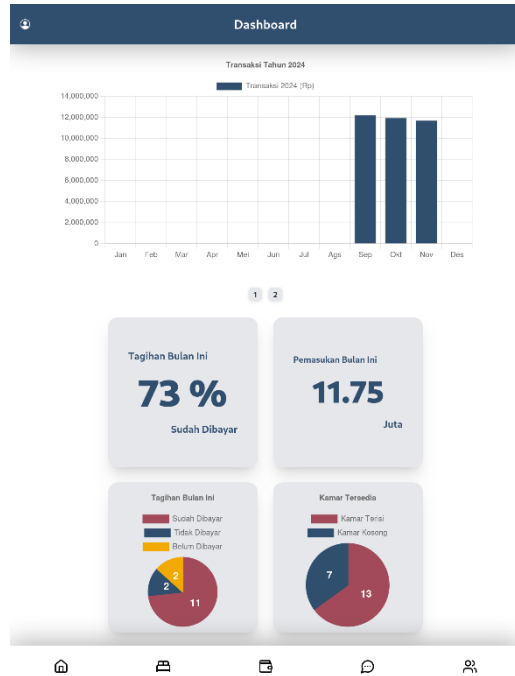
Gambar 4.1 Sistem pada perangkat desktop



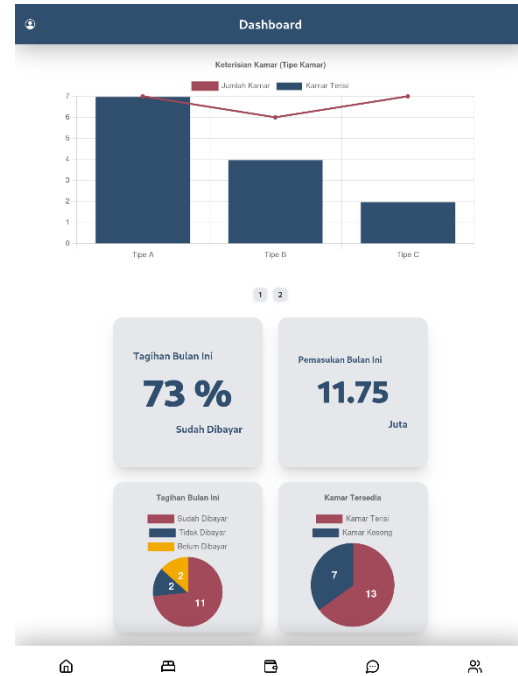
Gambar 4.2 Sistem pada perangkat mobile

## 4.1.2 Implementasi Sistem Informasi Antarmuka Admin

### a. Halaman Dashboard



Gambar 4.3 Halaman dashboard 1



Gambar 4.4 Halaman dashboard 2

Halaman dashboard ini merupakan halaman utama yang dirancang khusus untuk admin. Pada halaman ini, admin dapat dengan mudah mengakses dan memonitor berbagai informasi penting yang disajikan dalam bentuk visual yang menarik dan informatif. Di bagian atas dashboard, terdapat carousel yang menampilkan grafik batang. Grafik pertama menampilkan data transaksi tahun 2024 dalam satuan Rupiah, yang menggambarkan jumlah transaksi setiap bulan. Grafik kedua menampilkan ketersediaan kamar berdasarkan tipe kamar, memperlihatkan jumlah kamar yang tersedia dan yang terisi untuk setiap tipe.

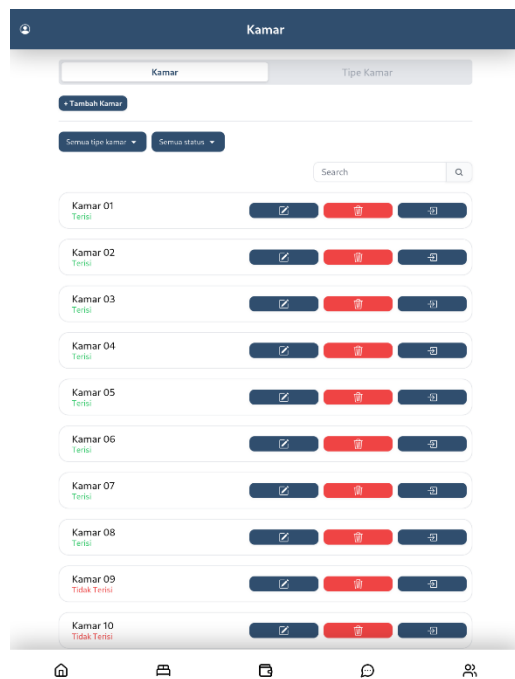
Di bawah grafik ketersediaan kamar, terdapat empat kotak informasi utama.

1. **Tagihan Bulan Ini:** Menunjukkan persentase tagihan yang telah dibayar dalam bentuk persentase, dengan tampilan yang mencolok dan mudah dipahami.
2. **Pemasukan Bulan Ini:** Menampilkan jumlah pemasukan bulanan dalam satuan juta Rupiah, memberikan gambaran mengenai arus kas saat ini.

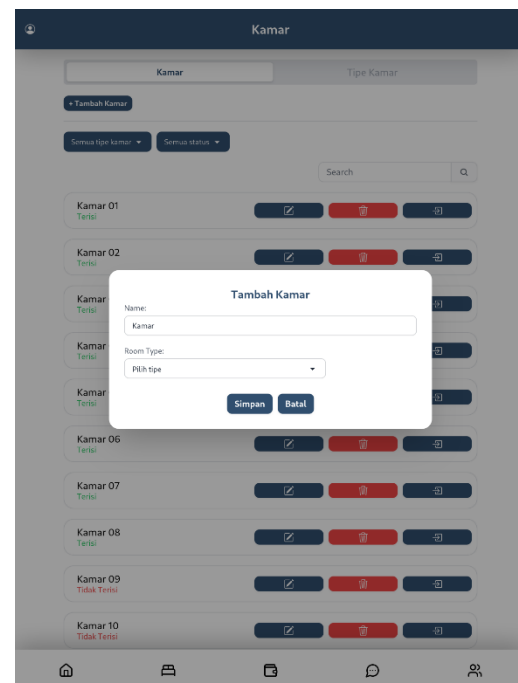
3. Tagihan Bulan Ini: Diagram ini memperlihatkan perbandingan antara tagihan yang sudah dibayar dan yang belum dibayar, memudahkan admin untuk melihat sejauh mana tagihan bulanan telah terpenuhi.
4. Kamar Tersedia: Diagram ini menunjukkan kondisi keterisian kamar saat ini, dengan perbandingan antara kamar yang terisi dan kamar yang kosong.

Setiap kotak informasi interaktif ini memiliki fungsi untuk mengarahkan admin ke halaman terkait untuk melihat detail lebih lanjut ketika kotak tersebut di klik.

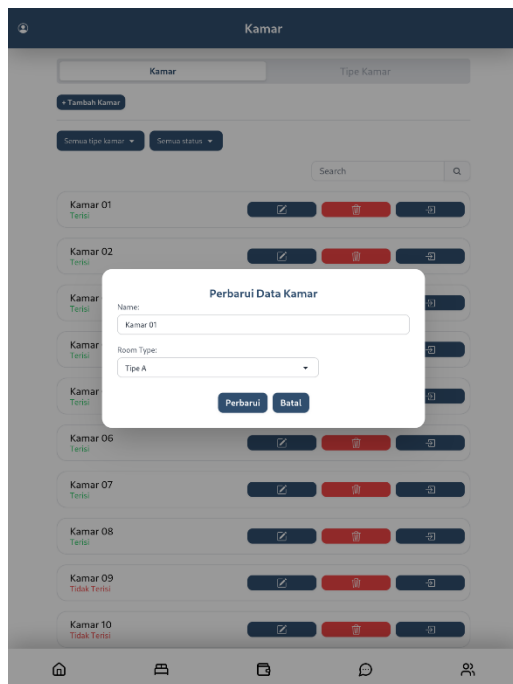
#### b. Halaman Menu Kamar



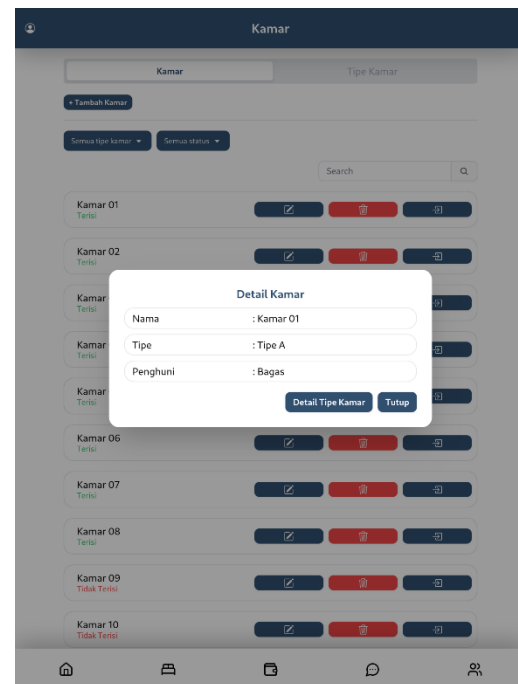
Gambar 4.5 Halaman kamar



Gambar 4.6 Tambah kamar



Gambar 4.7 Perbarui data kamar



Gambar 4.8 Detail kamar

Halaman Kamar adalah halaman yang dirancang khusus untuk memudahkan admin dalam mengelola data kamar di sistem kos-kosan. Pada halaman ini, admin dapat melihat daftar kamar yang tersedia, beserta status keterisiannya, apakah kamar tersebut "Terisi" atau "Tidak Terisi."

Fitur utama pada halaman ini meliputi:

1. **Filter dan Pencarian:** Admin dapat memfilter data kamar berdasarkan tipe kamar dan status keterisian. Filter ini memungkinkan admin untuk menyaring kamar sesuai dengan kebutuhan, seperti menampilkan hanya kamar yang kosong atau hanya kamar dengan tipe tertentu. Selain itu, terdapat juga kotak pencarian yang mempermudah admin dalam menemukan kamar dengan cepat berdasarkan nama.
2. **Tambah Kamar:** Untuk menambah data kamar baru, admin dapat mengklik tombol "Tambah Kamar" yang akan memunculkan jendela pop-up di halaman yang sama. Dalam jendela ini, admin dapat memasukkan informasi kamar baru, seperti nama kamar dan tipe kamar. Proses ini dilakukan secara langsung tanpa perlu berpindah halaman, sehingga meningkatkan efisiensi kerja admin.
3. **Detail dan Pembaruan Data Kamar:** Admin juga memiliki opsi untuk melihat detail kamar dengan mengklik tombol "Detail" di setiap baris kamar. Jendela pop-up akan muncul yang menampilkan informasi lengkap mengenai kamar tersebut, termasuk

nama kamar, tipe kamar, dan nama penghuni jika ada. Selain itu, admin dapat memperbarui informasi kamar dengan mengklik tombol "Edit," yang juga akan membuka jendela pop-up pada halaman yang sama, memungkinkan perubahan data seperti tipe kamar atau informasi lainnya.

4. Hapus Kamar: Setiap baris kamar memiliki tombol "Hapus" yang memungkinkan admin untuk menghapus data kamar jika diperlukan. Tindakan ini memudahkan admin dalam menjaga data tetap terorganisir dan hanya menyimpan informasi yang relevan.

### c. Halaman Tipe Kamar



Gambar 4.9 Halaman tipe kamar

Halaman Tipe Kamar dirancang untuk memudahkan admin dalam mengelola berbagai tipe kamar yang tersedia di sistem kos-kosan. Di halaman ini, admin dapat melihat daftar tipe kamar yang ada dengan opsi untuk melakukan tindakan lebih lanjut.

Fitur utama pada halaman ini meliputi:

1. Tambah Tipe Kamar: Admin dapat menambah tipe kamar baru dengan mengklik tombol "Tambah Tipe Kamar". Saat tombol ini diklik, admin akan diarahkan ke halaman lain yang khusus untuk memasukkan informasi lengkap mengenai tipe kamar baru. Hal ini memberikan fleksibilitas lebih dalam pengelolaan data dengan tampilan yang lebih mendetail.

2. Edit Tipe Kamar: Setiap baris tipe kamar memiliki tombol "Edit" yang memungkinkan admin untuk memperbarui informasi terkait tipe kamar tersebut. Saat tombol "Edit" di klik, admin akan diarahkan ke halaman lain yang memungkinkan pengeditan informasi secara rinci, seperti perubahan nama tipe kamar atau detail lainnya.
3. Hapus Tipe Kamar: Admin juga dapat menghapus tipe kamar yang sudah tidak diperlukan lagi dengan mengklik tombol "Hapus". Tindakan ini dilakukan langsung di halaman yang sama, memastikan bahwa tipe kamar yang sudah tidak relevan dapat segera dihapus dari sistem.

#### d. Halaman Tambah dan Perbarui Tipe Kamar

Gambar 4.10 Halaman tipe kamar

Gambar 4.11 Perbarui data tipe kamar

Pada halaman Tambah dan Perbarui Tipe Kamar, admin bisa dengan mudah mengelola detail tipe kamar, baik untuk menambahkan tipe kamar baru atau memperbarui yang sudah ada, seperti Nama Tipe Kamar, Harga Sewa, Deskripsi Kamar dan Photo Kamar. Selain itu, admin juga dapat menghapus foto kamar yang sudah ada dengan mengklik foto tersebut.

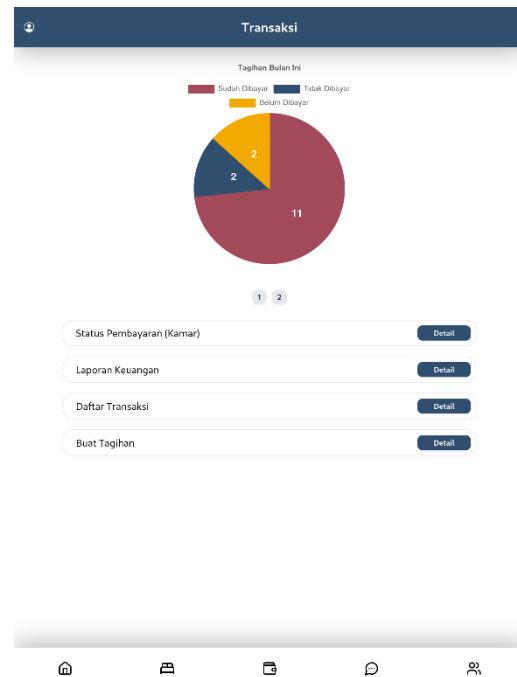
Setelah semua informasi dimasukkan atau diperbarui, admin dapat menekan tombol "Simpan" untuk menyimpan tipe kamar baru atau memperbarui tipe kamar yang sudah ada.

Dengan tindakan ini, tipe kamar akan tersimpan dalam sistem dengan informasi yang akurat dan terbaru.

#### e. Halaman Menu Keuangan



Gambar 4.12 Halaman keuangan 1



Gambar 4.13 Halaman keuangan 2

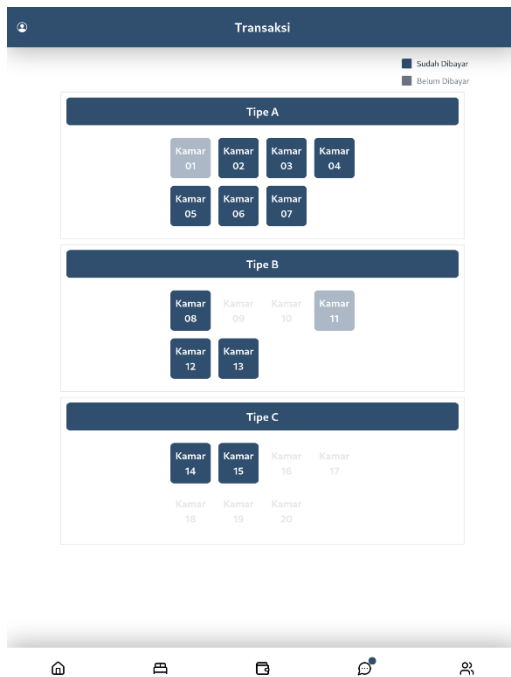
Halaman Keuangan dirancang untuk memberikan admin akses lengkap dalam memantau, mengelola, dan membuat laporan terkait arus kas yang berasal dari pembayaran kos. Halaman ini menampilkan rangkuman visual serta akses cepat ke berbagai laporan dan data transaksi keuangan.

Fitur utama pada halaman ini meliputi:

1. Carousel Chart: Di bagian atas halaman, terdapat carousel yang menampilkan grafik dan diagram terkait transaksi keuangan. Carousel ini memberikan gambaran visual yang jelas mengenai kondisi keuangan saat ini.
  - Grafik Pemasukan Bulanan: Grafik batang yang menampilkan jumlah pemasukan yang diperoleh setiap bulan sepanjang tahun ini.
  - Pie Chart Status Pembayaran: Diagram lingkaran yang memperlihatkan perbandingan antara tagihan yang sudah dibayar dan yang belum dibayar untuk bulan ini.

2. Menu Navigasi Keuangan: Di bawah carousel chart, terdapat beberapa pilihan menu yang memberikan akses cepat ke berbagai laporan dan data keuangan yang lebih detail. Setiap menu dilengkapi dengan tombol "Detail" yang akan mengarahkan admin ke halaman yang sesuai:
- Status Pembayaran (Kamar): Menyediakan detail mengenai status pembayaran per kamar, membantu admin memantau pembayaran dari setiap penghuni dengan lebih rinci.
  - Laporan Keuangan: Menampilkan laporan keuangan yang hanya berisi pemasukan dari pembayaran kos. Laporan ini membantu admin untuk melacak uang masuk dan memberikan gambaran yang jelas mengenai pendapatan bulanan.
  - Daftar Transaksi: Berisi daftar lengkap transaksi uang masuk yang dilakukan oleh penghuni kos. Ini memberikan transparansi dan kemudahan dalam melacak setiap pembayaran yang diterima.
  - Buat Transaksi: Menu ini dirancang untuk membuat data tagihan baru. Secara default, tagihan akan dibuat secara otomatis setiap bulan berdasarkan data saat pengguna pertama kali masuk. Namun, jika ada tagihan tambahan di luar tagihan bulanan reguler, admin dapat membuatnya melalui menu ini. Selain itu, pada menu ini admin dapat memperbarui tagihan yang tidak dibayar.

## f. Halaman Status Pembayaran



Gambar 4.14 Halaman status pembayaran



Gambar 4.15 Halaman status pembayaran detail

Halaman Transaksi ini memungkinkan admin untuk melihat status pembayaran kamar secara langsung. Pada halaman ini, terdapat daftar kamar yang ditandai dengan warna yang menunjukkan status pembayaran:

- Biru: Menandakan kamar yang pembayarannya sudah dilakukan.
- Abu-abu: Menandakan kamar yang pembayarannya belum dilakukan.
- Putih: Menandakan kamar tidak terisi

Admin dapat melihat detail status pembayaran dengan mengklik salah satu kamar yang tersedia. informasi rinci mengenai kamar tersebut, termasuk:

- Nama Kamar: Menampilkan nomor kamar.
- Penghuni: Nama penghuni yang menempati kamar tersebut.
- Status Pembayaran: Memberikan informasi apakah pembayaran sudah dilakukan atau belum.
- Tanggal Pembayaran: Menunjukkan tanggal jatuh tempo pembayaran untuk kamar tersebut.

## g. Halaman Laporan Keuangan

	Nama	Kamar	Periode	Jumlah	Status
1	Dr. Brown Pauck	Kamar 08	08-2024	1.250.000	Sudah Dibayar
2	Roman Murray DDS	Kamar 29	08-2024	750.000	Sudah Dibayar
3	Hermann Durgan	Kamar 26	08-2024	750.000	Sudah Dibayar
4	Miss Wilma Siedemann	Kamar 12	08-2024	100.000	Sudah Dibayar
5	Mrs. Zoe Jaskolski	Kamar 07	08-2024	1.250.000	Belum Dibayar
6	Elmira Towne	Kamar 18	08-2024	100.000	Belum Dibayar
7	Miss Anvilla Swaniawski IV	Kamar 27	08-2024	750.000	Belum Dibayar
	Total			4.950.000	

Gambar 4.16 Halaman laporan keuangan bulanan

	Periode	Tagihan		Terbayar		%
		#	Rp	#	Rp	
1	01-2024	50	37.500.000	45	33.750.000	90 %
2	02-2024	50	37.500.000	45	33.750.000	90 %
3	03-2024	50	5.000.000	45	4.500.000	90 %
4	04-2024	50	5.000.000	45	4.500.000	90 %
5	05-2024	50	62.500.000	45	56.250.000	90 %
6	06-2024	50	62.500.000	45	56.250.000	90 %
7	07-2024	50	62.500.000	45	56.250.000	90 %
8	08-2024	7	4.950.000	4	2.850.000	57 %
	Total				248.100.000	

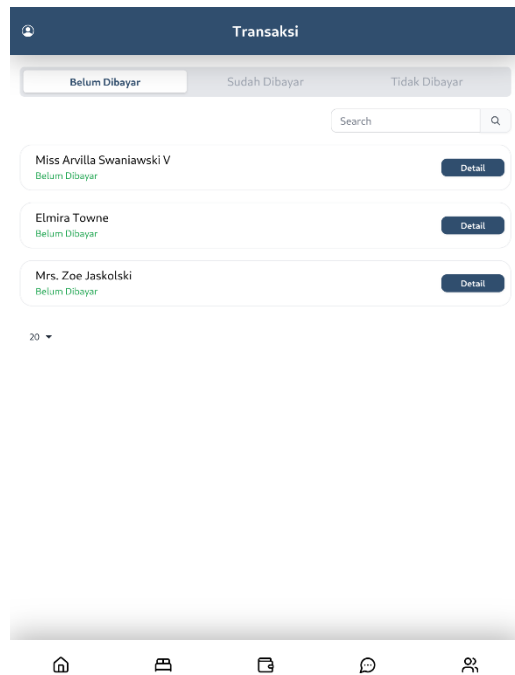
Gambar 4.17 Halaman laporan keuangan tahunan

Halaman Laporan Keuangan dirancang untuk memberikan admin akses lengkap terhadap laporan arus kas yang berasal dari pembayaran kos. Halaman ini memungkinkan admin untuk melihat dan menganalisis data keuangan baik secara bulanan maupun tahunan, membantu dalam pengelolaan keuangan yang lebih efektif.

Fitur utama pada halaman ini meliputi:

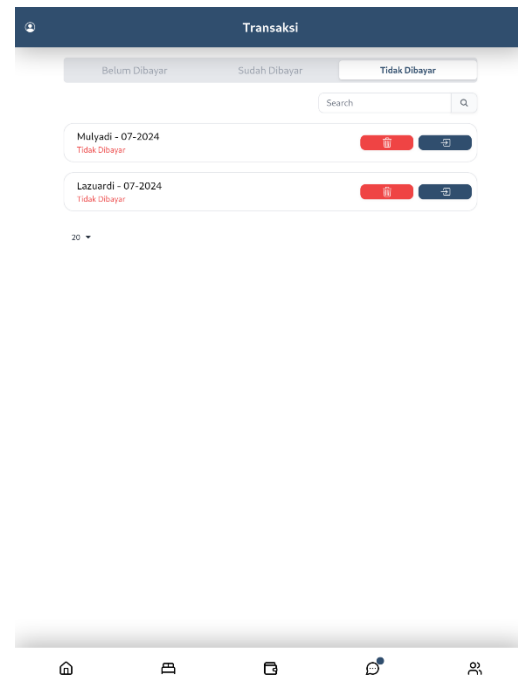
1. Pilihan Tampilan Bulanan dan Tahunan: Admin bisa memilih melihat laporan keuangan per bulan atau per tahun, membantu fokus pada detail transaksi bulanan atau gambaran keuangan tahunan.
2. Laporan Bulanan: Menampilkan pembayaran yang diterima selama bulan tertentu, termasuk nama penghuni, nomor kamar, periode pembayaran, jumlah, dan status pembayaran. Ini membantu admin melacak semua transaksi bulan tersebut.
3. Laporan Tahunan: Menyajikan ringkasan total tagihan dan pembayaran per bulan dalam setahun, termasuk jumlah dan persentase pembayaran yang diterima. Ini memberikan wawasan tentang tren keuangan sepanjang tahun.
4. Fitur Ekspor Laporan: Admin bisa mengekspor laporan keuangan untuk analisis lebih lanjut atau pelaporan eksternal dengan mudah menggunakan tombol "Export".

## h. Halaman Daftar Transaksi



Gambar 4.18 Halaman daftar transaksi

1



Gambar 4.19 Halaman daftar transaksi

2

Halaman Daftar Transaksi dirancang untuk memberikan admin tampilan yang terorganisir mengenai status pembayaran dari setiap tagihan kos. Halaman ini memudahkan admin dalam melacak, mengelola, dan menindaklanjuti pembayaran berdasarkan statusnya. Admin dapat melihat daftar penghuni beserta status pembayarannya, seperti "Belum Dibayar", "Sudah Dibayar", atau "Tidak Dibayar". Setiap status pembayaran ditandai dengan indikator warna. Selain itu, ada fitur pencarian yang memungkinkan admin untuk mencari transaksi berdasarkan nama penghuni, serta opsi untuk melihat detail transaksi lebih lanjut dengan mengklik tombol "Detail". Data tagihan yang berstatus "Tidak Dibayar" juga dapat dihapus dengan menekan tombol "Hapus" yang tersedia di setiap data tagihan.

### i. Halaman Buat dan Perbarui Tagihan

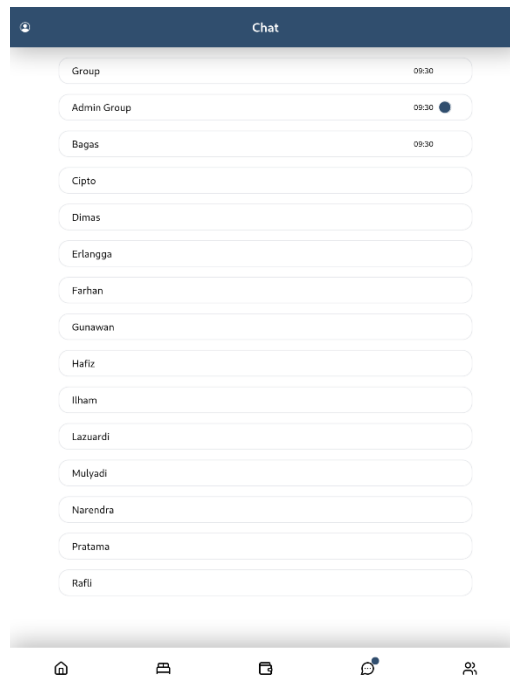
Gambar 4.20 Halaman buat tagihan

Gambar 4.21 Halaman perbarui tagihan

Pada halaman ini, admin dapat membuat tagihan tambahan di luar tagihan bulanan yang sudah dihasilkan secara otomatis oleh sistem. Admin dapat memilih penghuni, kamar, serta memasukkan jumlah tagihan dan keterangan yang diperlukan. Setelah semua informasi diisi, admin dapat menekan tombol "Simpan" untuk menyimpan tagihan baru tersebut.

Selain itu, halaman ini juga memungkinkan admin untuk memperbarui tagihan yang tidak dibayar oleh penghuni. Admin dapat memilih tagihan yang ingin diperbarui dari daftar tagihan yang tersedia. Setelah tagihan dipilih, kolom-kolom di bawahnya akan otomatis terisi dengan data tagihan yang dipilih. Selanjutnya admin dapat menyimpan perubahan dengan menekan tombol "Perbarui." Sistem hanya akan memperbarui nomor tagihan dan tenggat waktu pembayaran tanpa mengubah detail lainnya.

## j. Halaman Menu Chat



Gambar 4.22 Halaman menu chat

Halaman Chat dirancang untuk memfasilitasi komunikasi antara admin dan penghuni kos-kosan, serta menyediakan ruang untuk diskusi grup jika diperlukan. Halaman ini menampilkan daftar percakapan aktif yang dapat diakses oleh admin untuk memantau dan merespons pesan dengan cepat. Daftar percakapan mencakup percakapan grup yang melibatkan semua pengguna aktif, baik admin maupun penghuni, grup khusus admin yang hanya berisi admin, serta percakapan individu dengan setiap pengguna aktif.

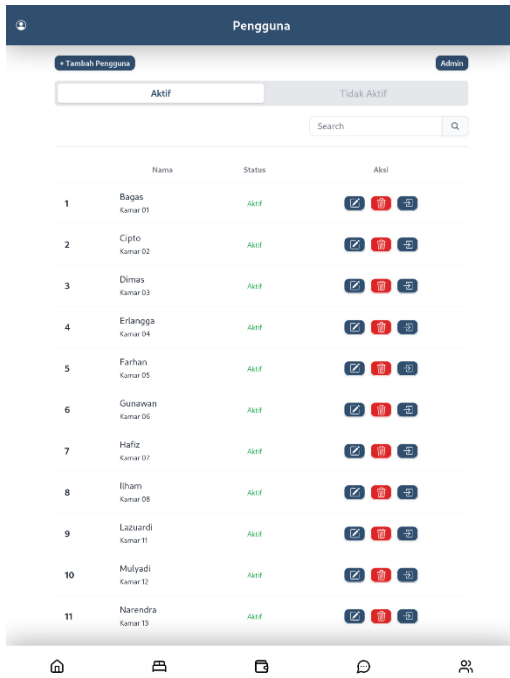
## k. Halaman Chat



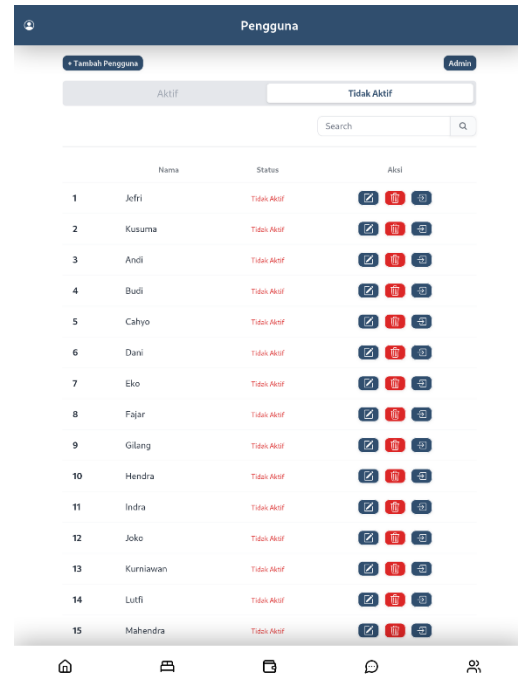
Gambar 4.23 Halaman chat

Pada halaman ini, ditampilkan percakapan berdasarkan percakapan yang dipilih dari menu chat sebelumnya. Halaman ini menampilkan riwayat percakapan secara kronologis, sehingga admin dapat dengan mudah mengikuti dan melanjutkan diskusi yang sedang berlangsung. Setiap pesan ditandai dengan tanggal dan waktu pengiriman, serta dilengkapi dengan identitas pengirim, baik itu dari admin maupun penghuni. Di bagian bawah halaman, tersedia kolom input untuk memasukkan pesan baru yang akan dikirim dalam percakapan ini. Admin dapat mengetikkan pesan di kolom tersebut dan mengirimkannya dengan menekan tombol kirim.

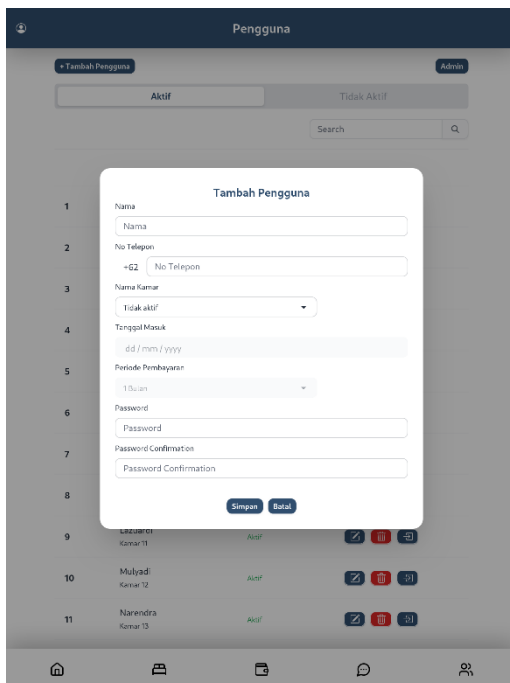
## 1. Halaman Menu Pengguna



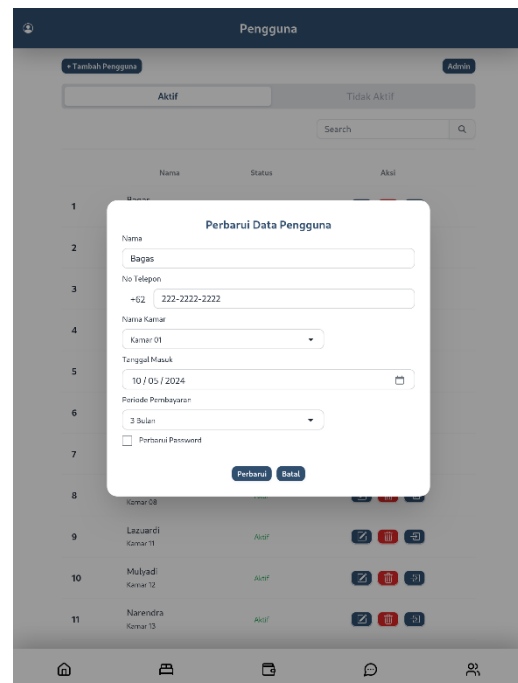
Gambar 4.24 Halaman pengguna aktif



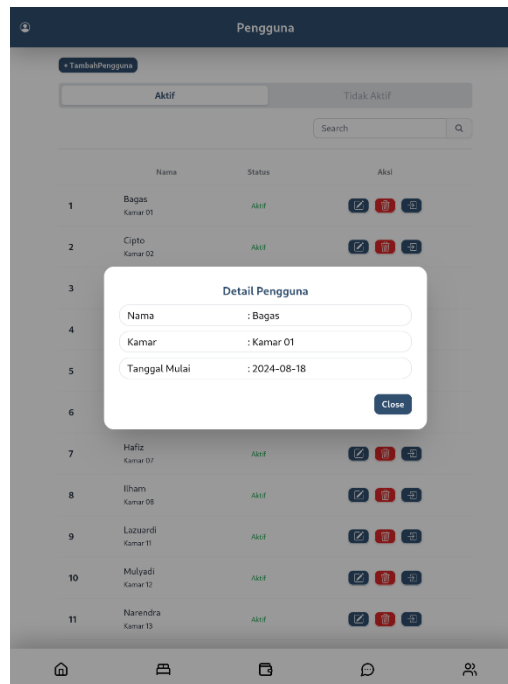
Gambar 4.25 Halaman pengguna tidak aktif



Gambar 4.26 Tambah pengguna



Gambar 4.27 Perbarui data pengguna



Gambar 4.28 Detail pengguna

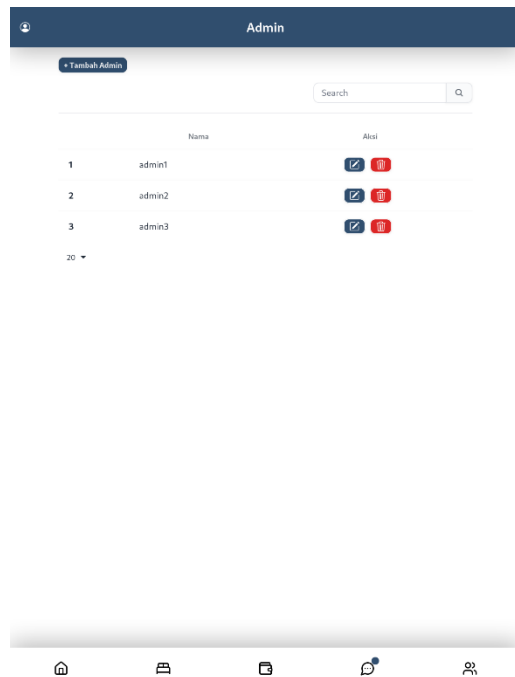
Pada halaman Pengguna, terdapat daftar data pengguna yang bisa di filter berdasarkan status pengguna, seperti "Aktif" atau "Tidak Aktif". Di halaman ini, admin memiliki berbagai opsi pengelolaan data pengguna.

**Tambah Pengguna:** Terdapat tombol "Tambah Pengguna" yang memungkinkan admin untuk menambahkan pengguna baru ke dalam sistem. Ketika tombol ini diklik, sebuah jendela pop-up akan muncul di halaman yang sama, memungkinkan admin untuk memasukkan data pengguna baru seperti nama, nomor telepon, kamar yang ditempati, serta informasi login.

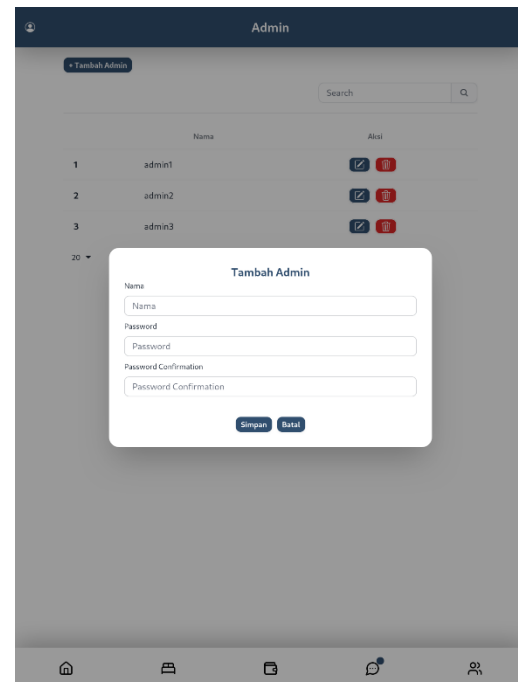
**Detail dan Pembaruan Data:** Admin dapat melihat detail lengkap dari data pengguna dengan mengklik tombol "Detail" pada setiap baris pengguna. Detail yang ditampilkan mencakup informasi seperti nama, nomor kamar, dan tanggal mulai. Selain itu, admin juga dapat memperbarui informasi pengguna yang ada dengan mengklik tombol "Edit," yang juga akan membuka jendela pop-up pada halaman yang sama. Pop-up ini memungkinkan admin untuk mengubah informasi yang diperlukan, seperti mengganti password, nomor telepon, atau status kamar.

**Hapus Pengguna:** Untuk menghapus data pengguna, admin dapat mengklik tombol "Hapus" yang ada di setiap baris pengguna. Tindakan ini memungkinkan admin untuk mengelola data pengguna secara efisien, memastikan bahwa hanya data yang relevan yang tetap ada di dalam sistem.

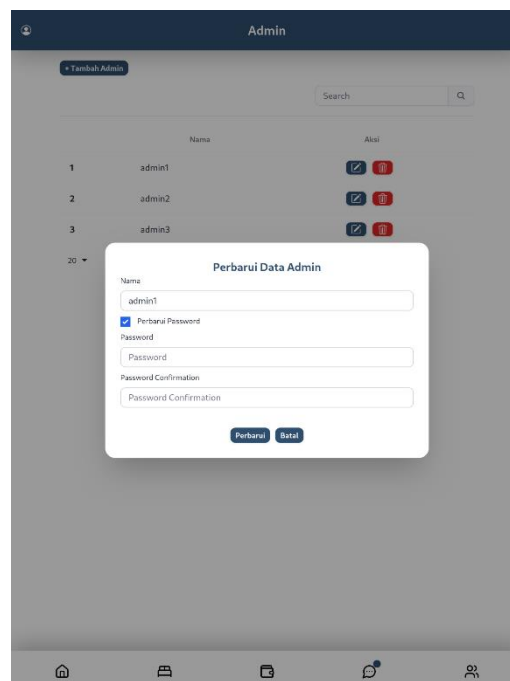
### m. Halaman Admin



Gambar 4.29 Halaman admin



Gambar 4.30 Halaman admin (detail)



Gambar 4.31 Halaman admin (perbarui)

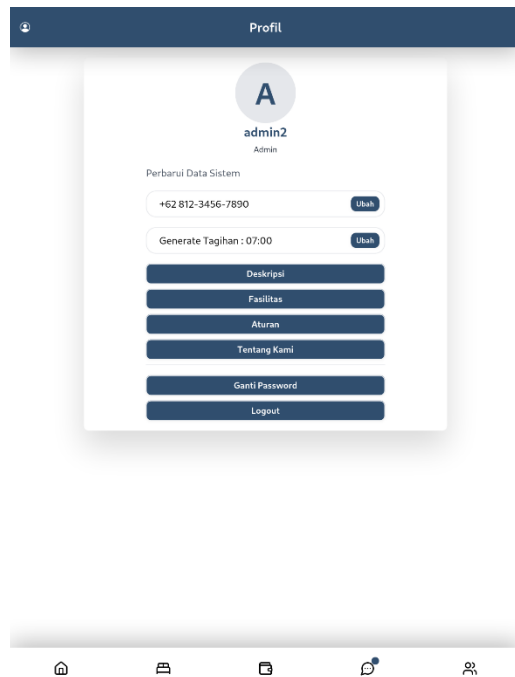
Pada halaman ini, admin dapat mengelola daftar admin yang terdaftar dalam sistem. Di bagian atas, terdapat tombol "+ Tambah Admin" yang memungkinkan admin untuk menambahkan akun admin baru. Setelah tombol ini diklik, sebuah form akan muncul, memungkinkan admin untuk memasukkan nama, password, dan konfirmasi password untuk

akun admin yang baru. Setelah semua informasi diisi, admin dapat menekan tombol "Simpan" untuk menambahkan admin baru ke dalam sistem.

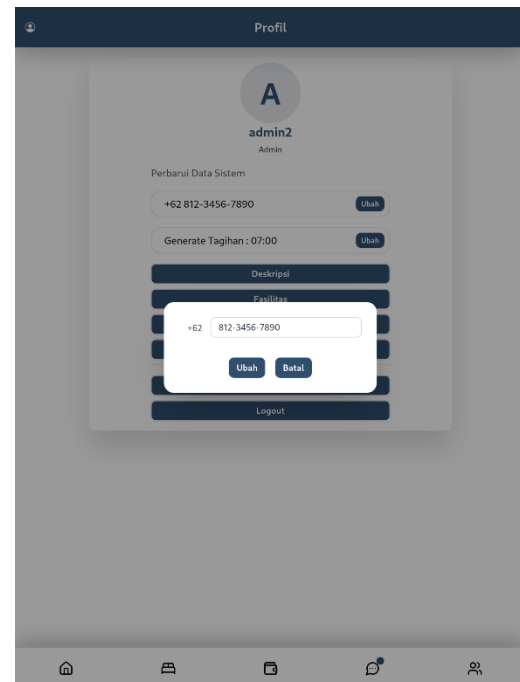
Selain itu, terdapat daftar admin yang sudah terdaftar, ditampilkan dalam bentuk tabel yang mencakup nama-nama admin dan opsi untuk mengedit atau menghapus akun admin tersebut. Admin dapat mengklik ikon pensil untuk memperbarui informasi admin, seperti nama atau password. Ketika admin memilih untuk memperbarui data, sebuah form akan muncul dengan informasi yang sudah terisi sebelumnya. Admin dapat mengubah nama, memilih untuk memperbarui password, dan kemudian menekan tombol "Perbarui" untuk menyimpan perubahan.

Setiap baris dalam tabel juga memiliki ikon tempat sampah yang memungkinkan admin menghapus akun admin yang bersangkutan dari sistem. Fitur pencarian di sebelah kanan atas juga disediakan untuk memudahkan admin dalam mencari akun tertentu berdasarkan nama.

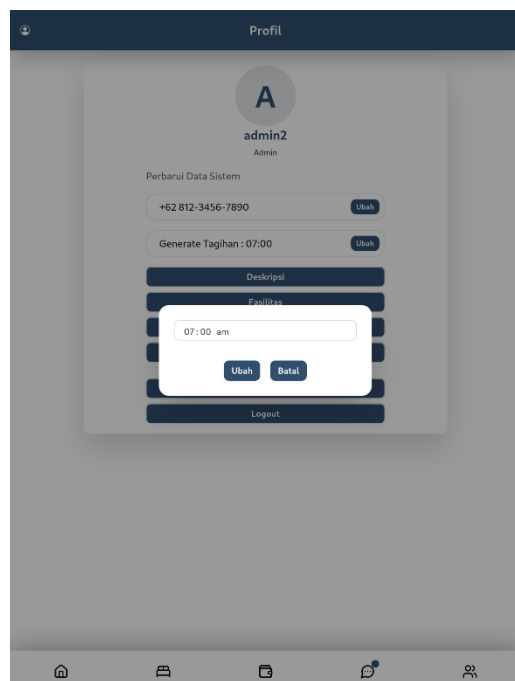
## n. Halaman Profil



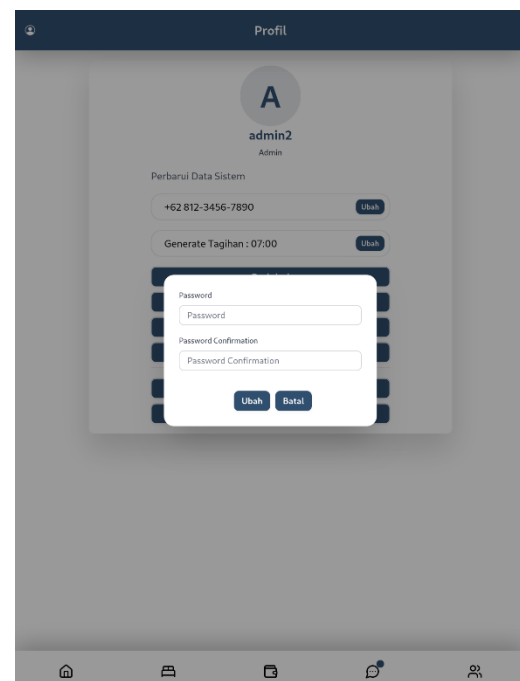
Gambar 4.32 Halaman profil



Gambar 4.33 Halaman profil telepon



Gambar 4.34 Halaman profil tagihan



Gambar 4.35 Halaman profil password

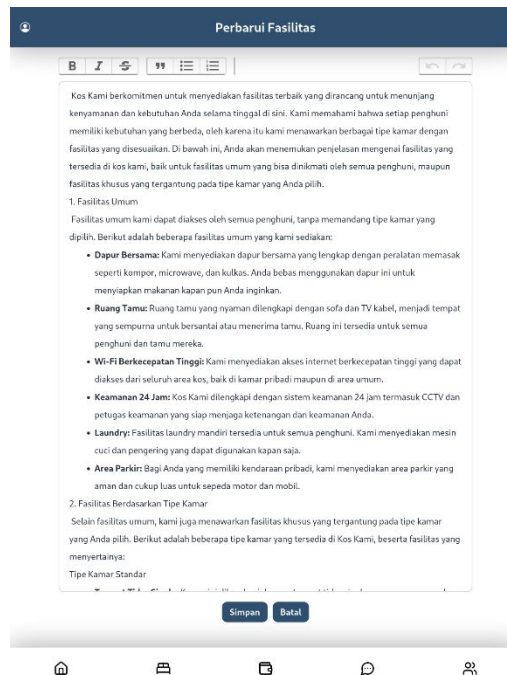
Pada halaman Profil ini, admin dapat melihat dan mengelola informasi pribadi serta beberapa pengaturan sistem yang terkait dengan perannya. Admin dapat memperbarui nomor telepon dan waktu pengaturan otomatis untuk pembuatan tagihan dengan mengklik tombol

"Ubah" di sebelah masing-masing informasi. Ketika admin mengklik tombol "Ubah", sebuah form akan muncul untuk memasukkan data baru, yang kemudian dapat disimpan dengan menekan tombol "Ubah" pada form tersebut.

Selain itu, terdapat beberapa tombol navigasi seperti "Deskripsi", "Fasilitas", "Aturan", dan "Tentang Kami". Mengklik salah satu dari tombol ini akan membawa admin ke halaman lain yang khusus untuk mengubah data terkait masing-masing topik.

Admin juga memiliki opsi untuk mengganti password dengan mengklik tombol "Ganti Password", yang akan membuka form untuk memasukkan password baru dan konfirmasi password. Setelah perubahan selesai dilakukan, admin dapat kembali ke halaman utama atau melakukan logout dengan menekan tombol "Logout" di bagian bawah.

#### o. Halaman Perbarui Data Sistem



Gambar 4.36 Halaman perbarui data sistem

Halaman ini digunakan untuk memperbarui data sistem yang mencakup deskripsi, fasilitas, aturan, dan tentang kami yang ada pada menu profil. Di sini, admin dapat mengedit dan memperbarui informasi secara langsung melalui editor teks yang tersedia. Setelah melakukan perubahan, admin dapat menyimpan data baru dengan menekan tombol "Simpan," atau membatalkan perubahan dengan menekan tombol "Batal." Halaman ini memastikan

bahwa informasi yang ditampilkan kepada penghuni dan calon penghuni selalu akurat dan terkini.

### 4.1.3 Implementasi Sistem Informasi Antarmuka Pengguna

#### a. Halaman Home



Gambar 4.37 Halaman home

Halaman Home ini merupakan halaman utama yang memberikan penghuni gambaran singkat tentang kos-kosan. Disini, penghuni dapat menemukan informasi dasar tentang fasilitas, layanan.

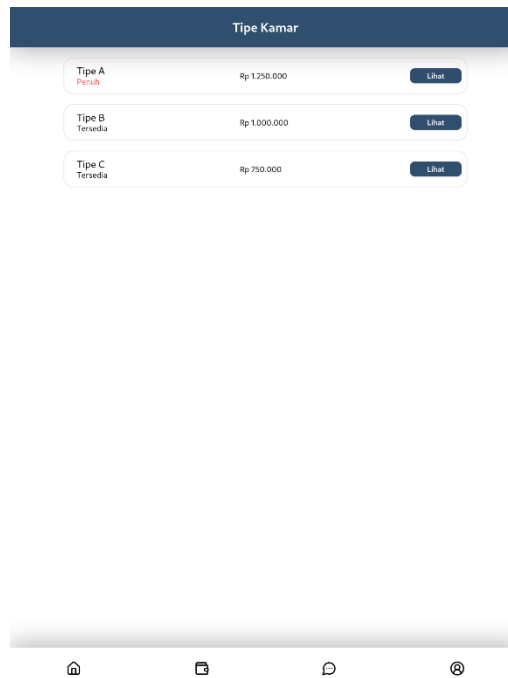
Pada halaman ini, terdapat tombol "Install on Phone" yang memberikan petunjuk langkah demi langkah untuk menambahkan website ke home screen pada perangkat mobile. Fitur ini memudahkan pengunjung untuk mengakses website dengan cepat langsung dari layar utama ponsel mereka.

Di bawah deskripsi, terdapat empat tombol utama yang membantu pengunjung menjelajahi informasi lebih lanjut:

- Kamar: Mengarahkan pengunjung ke halaman yang menampilkan berbagai jenis kamar yang tersedia.
- Fasilitas: Memberikan informasi lengkap tentang fasilitas yang ditawarkan, seperti ruang tamu bersama, dapur, layanan laundry, dan area parkir.

- Aturan: Menampilkan aturan kos yang harus dipatuhi oleh semua penghuni, untuk memastikan kenyamanan dan ketertiban di kos-kosan.
- Tentang Kami: Menyediakan informasi lebih mendalam tentang kos-kosan, termasuk sejarah, visi, misi, dan informasi kontak.

#### b. Halaman Daftar Tipe Kamar

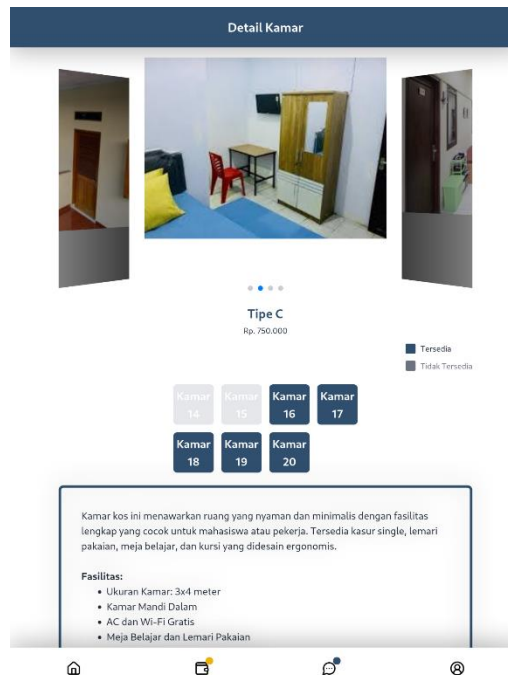


Gambar 4.38 Halaman tipe kamar

Halaman Tipe Kamar ini dirancang untuk memudahkan calon penghuni dalam memilih jenis kamar yang sesuai dengan kebutuhan dan anggaran mereka. Di halaman ini, terdapat daftar tipe kamar yang tersedia di kos-kosan, lengkap dengan status ketersediaan dan harga per bulannya.

Setiap tipe kamar dilengkapi dengan tombol "Lihat" yang memungkinkan calon penghuni untuk melihat lebih detail mengenai fasilitas, ukuran, dan foto kamar tersebut. Ini memudahkan pengunjung dalam membuat keputusan sebelum melakukan reservasi atau kunjungan.

### c. Halaman Detail Kamar



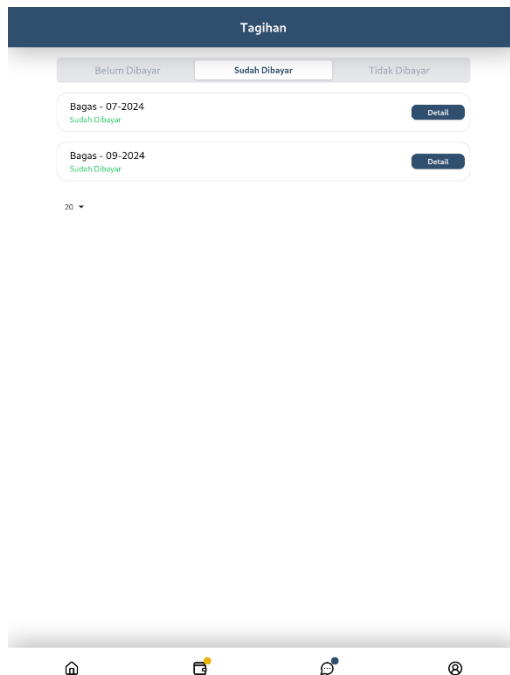
Gambar 4.39 Halaman detail kamar

Halaman Detail Kamar ini memberikan informasi lengkap dan visual menarik tentang kamar yang tersedia di kos-kosan Anda. Pengunjung dapat melihat foto-foto kamar dan koridor untuk mendapatkan gambaran suasana dan tata letak ruang.

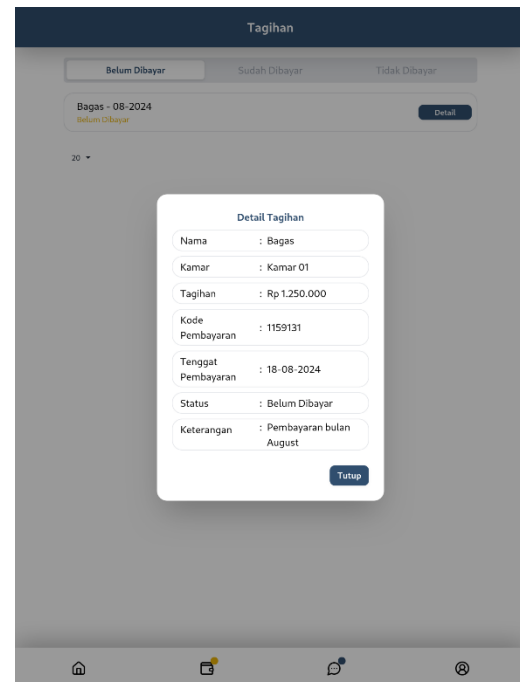
Di bawah carousel foto, terdapat informasi mengenai tipe kamar dan harganya, membantu calon penghuni memahami pilihan yang ada. Nomor kamar yang tersedia ditampilkan dengan warna biru, sementara yang sudah terisi atau tidak tersedia ditandai dengan warna abu-abu, memudahkan pengunjung melihat ketersediaan kamar.

Bagian bawah halaman mencantumkan fasilitas yang terkait dengan tipe kamar, seperti ukuran, kamar mandi, AC, Wi-Fi, dan perabotan. Ini membantu calon penghuni menilai kenyamanan dan kebutuhan mereka sebelum memutuskan.

#### d. Halaman Menu Tagihan



Gambar 4.40 Halaman tagihan



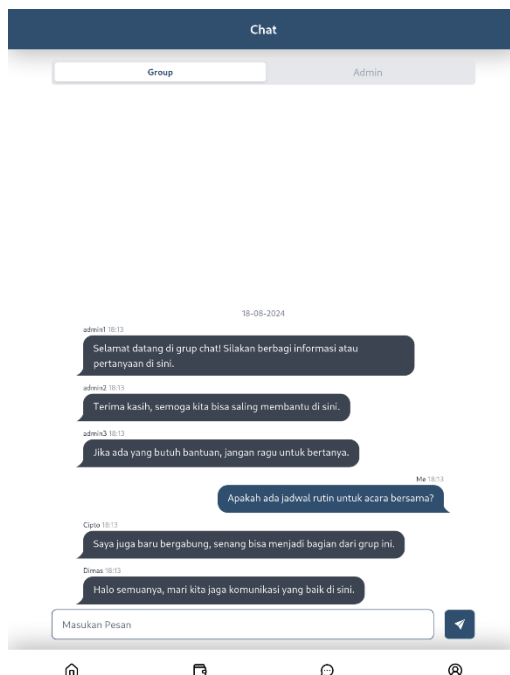
Gambar 4.41 Halaman tagihan detail

Halaman Tagihan ini memudahkan penghuni untuk memantau status pembayaran kos mereka. Terdapat tiga tab yang mengelompokkan tagihan berdasarkan status: Belum Dibayar, Sudah Dibayar, dan Tidak Dibayar. Penghuni dapat dengan mudah beralih antar-tab untuk melihat tagihan sesuai statusnya.

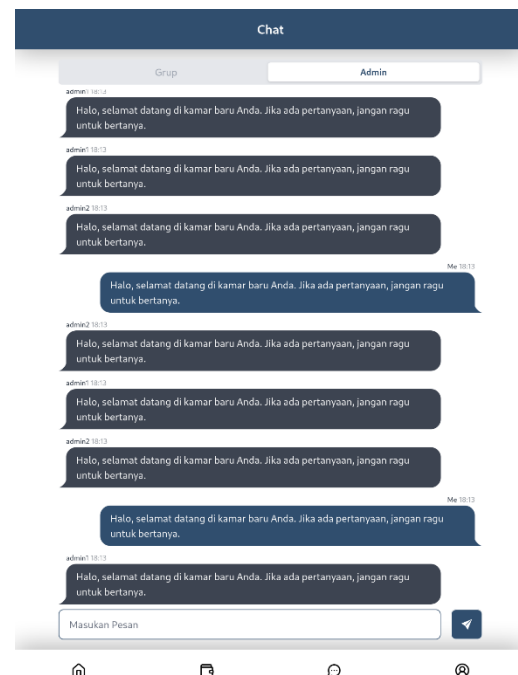
Di tengah halaman, daftar tagihan ditampilkan sesuai dengan tab yang dipilih. Setiap entri menyertakan periode tagihan dan status pembayaran, serta tombol "Detail" yang memungkinkan penghuni melihat informasi lengkap mengenai tagihan tersebut.

Ketika tombol "Detail" di klik, muncul jendela pop-up yang menampilkan rincian tagihan, seperti nama penghuni, nomor kamar, jumlah tagihan, kode pembayaran, tenggat waktu, status pembayaran, dan keterangan tambahan terkait tagihan tersebut.

### e. Halaman Chat



Gambar 4.42 Halaman chat group



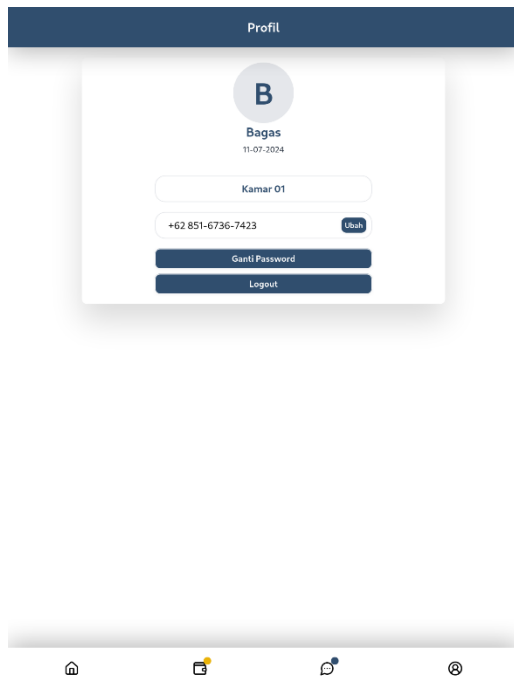
Gambar 4.43 Halaman chat admin

Halaman Chat ini dirancang sebagai pusat komunikasi antara penghuni kos dan admin, serta antar penghuni dalam grup. Halaman ini terbagi menjadi dua tab utama: Admin dan Grup, yang memudahkan pengguna untuk berinteraksi sesuai kebutuhan mereka.

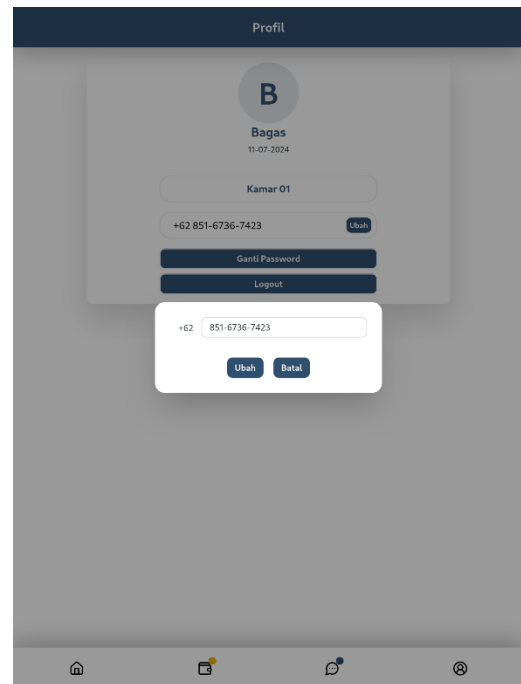
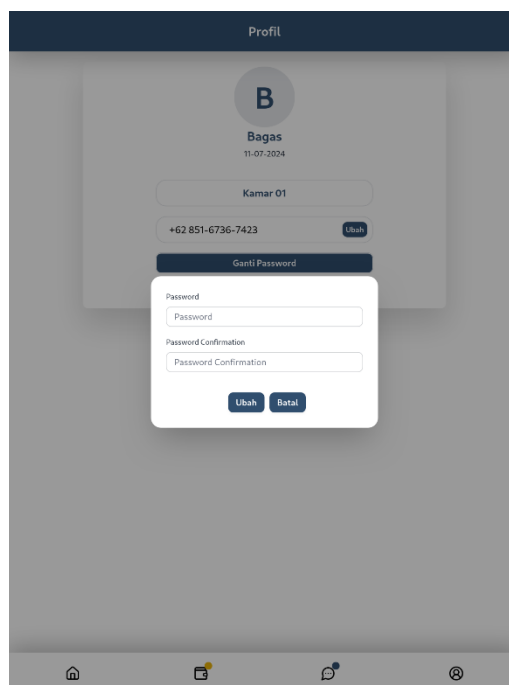
Setiap pesan yang dikirim atau diterima ditampilkan dalam bentuk balon percakapan, dengan identitas pengirim dan waktu pesan yang tercantum dengan jelas. Hal ini memungkinkan penghuni untuk dengan mudah mengikuti percakapan dan mendapatkan respon dari admin secara langsung.

Di bagian bawah halaman, terdapat kolom input teks di mana penghuni dapat mengetikkan pesan mereka dan mengirimkannya dengan mengklik tombol kirim.

## f. Halaman Menu Profil



Gambar 4.44 Halaman profil

Gambar 4.45 Halaman profil  
(ubah telepon)Gambar 4.46 Halaman Profil  
(ubah password)

Halaman Profil ini memberikan penghuni kontrol penuh atas informasi penting terkait akun mereka. Di bagian atas, penghuni dapat melihat nama, nomor kamar, dan tanggal mulai tinggal di kos. Informasi ini membantu memastikan bahwa data yang ditampilkan sesuai dengan riwayat mereka.

Penghuni juga memiliki opsi untuk memperbarui nomor telepon. Dengan menekan tombol "Ubah" di sebelah nomor telepon yang terdaftar, mereka dapat memasukkan nomor baru melalui jendela pop-up yang muncul. Setelah perubahan selesai, penghuni bisa menyimpan nomor baru dengan menekan tombol "Ubah" atau membatalkan perubahan dengan tombol "Batal".

Selain itu, penghuni dapat menjaga keamanan akun mereka dengan mengganti kata sandi melalui tombol "Ganti Password". Ini akan membuka jendela pop-up di mana mereka bisa memasukkan dan mengonfirmasi kata sandi baru sebelum menyimpan perubahan.

Tombol "Logout" di bagian bawah halaman memungkinkan penghuni untuk keluar dari akun mereka dengan aman setelah selesai menggunakan sistem.

#### **4.1.4 Pengujian**

Pengujian dilakukan untuk memastikan bahwa sistem bekerja sesuai dengan rencana yang telah ditetapkan. Pengujian dibagi menjadi dua jenis. Pertama, pengujian fungsional yang menggunakan metode black-box testing, di mana setiap fitur atau fungsi dari sistem diuji tanpa memperdulikan cara kerjanya di dalam. Tujuannya untuk memastikan bahwa setiap fungsi yang ada berfungsi dengan benar sesuai kebutuhan. Kedua, pengujian non-fungsional yang menggunakan System Usability Scale (SUS), yang bertujuan untuk mengukur sejauh mana sistem mudah digunakan oleh pengguna. Dengan kedua jenis pengujian ini, kami dapat memastikan bahwa sistem tidak hanya berjalan dengan baik, tetapi juga mudah digunakan oleh penggunanya.

##### **a. Pengujian Fungsional**

Dalam pengujian fungsional ini, terdapat empat responden yang akan melakukan pengujian terhadap setiap test case. Kriteria keberhasilan ditetapkan sebagai "semua harus berhasil" sehingga jika suatu testcase berhasil diuji oleh keempat responden, maka test case tersebut dinyatakan berhasil. Namun, jika ada satu responden yang gagal dalam pengujian suatu testcase, maka test case tersebut dinyatakan gagal.

Tabel 4.1 Pengujian Fungsional

No.	Fungsional	Testcase	
1	Pengelolaan Data Pengguna	1.1	Menambahkan data pengguna baru
		1.2	Mengedit data pengguna
		1.3	Menghapus data pengguna
2	Pengelolaan Data Kamar Kos	2.1	Menambahkan tipe kamar kos
		2.2	Mengedit tipe kamar kos
		2.3	Menghapus tipe kamar kos
		2.4	Menambahkan data kamar kos
		2.5	Mengedit data kamar kos
		2.6	Menghapus data kamar kos
3	Informasi Kamar	3.2	Menampilkan informasi kamar (status, fasilitas, biaya sewa, tipe kamar)
4	Pengelolaan Tagihan	4.2	Membuat tagihan baru
		4.2	Menampilkan daftar tagihan beserta statusnya
		4.3	Menghapus tagihan yang tidak dibayar
5	Generate Tagihan Otomatis	5.1	Pembuatan tagihan otomatis
6	Mengirim Notifikasi Tagihan	6.1	Mengirim notifikasi tagihan kepada pengguna (WhatsApp)
		6.1	Memastikan notifikasi mencakup jumlah tagihan, tanggal jatuh tempo, dan kode pembayaran
7	Proses Pembayaran	7.1	Pembayaran tagihan menggunakan ATM atau <i>e-wallet</i>
		7.2	Memperbarui status pembayaran secara otomatis setelah transaksi berhasil
		7.3	Pembaruan otomatis status tagihan setelah melewati masa tenggang (status menjadi "Tidak Dibayar")
8	Riwayat Transaksi	8.1	Menampilkan riwayat transaksi pembayaran tagihan (tanggal, nama, jumlah pembayaran)
9	Laporan Keuangan	9.1	Menyajikan laporan keuangan bulanan dan tahunan yang bisa diunduh
10	Obrolan	10.1	Melakukan obrolan antar pengguna dan admin melalui chat pribadi
		10.2	Melakukan grup obrolan antara penghuni dan admin

Tabel 4.2 Responden pengujian fungsional

No.	Nama Responden
1	Dhea Khansa Nabila
2	Arif Budiawan
3	Zunanda Ibrahim
4	Bintang Ahmad Ramadhan

Tabel 4.3 Hasil kuesioner pengujian fungsional

Test Case	Responden 1	Responden 2	Responden 3	Responden 4	Hasil
1.1	Berhasil	Berhasil	Berhasil	Berhasil	Berhasil
1.2	Berhasil	Berhasil	Berhasil	Berhasil	Berhasil
1.3	Berhasil	Berhasil	Berhasil	Berhasil	Berhasil
2.1	Berhasil	Berhasil	Berhasil	Berhasil	Berhasil
2.2	Berhasil	Berhasil	Berhasil	Berhasil	Berhasil
2.3	Berhasil	Berhasil	Berhasil	Berhasil	Berhasil
2.4	Berhasil	Berhasil	Berhasil	Berhasil	Berhasil
2.5	Berhasil	Berhasil	Berhasil	Berhasil	Berhasil
2.6	Berhasil	Berhasil	Berhasil	Berhasil	Berhasil
3.1	Berhasil	Berhasil	Berhasil	Berhasil	Berhasil
4.1	Berhasil	Berhasil	Berhasil	Berhasil	Berhasil
4.2	Berhasil	Berhasil	Berhasil	Berhasil	Berhasil
4.3	Berhasil	Berhasil	Berhasil	Berhasil	Berhasil
5.1	Berhasil	Berhasil	Berhasil	Berhasil	Berhasil
6.1	Berhasil	Berhasil	Berhasil	Berhasil	Berhasil
6.2	Berhasil	Berhasil	Berhasil	Berhasil	Berhasil
7.1	Berhasil	Berhasil	Berhasil	Berhasil	Berhasil
7.2	Berhasil	Berhasil	Berhasil	Berhasil	Berhasil
7.3	Berhasil	Berhasil	Berhasil	Berhasil	Berhasil
8.1	Berhasil	Berhasil	Berhasil	Berhasil	Berhasil
9.1	Berhasil	Berhasil	Berhasil	Berhasil	Berhasil
10.1	Berhasil	Berhasil	Berhasil	Berhasil	Berhasil
10.2	Berhasil	Berhasil	Berhasil	Berhasil	Berhasil

Berdasarkan hasil pengujian fungsional menggunakan metode black-box yang dilakukan oleh empat responden terhadap 23 test case yang mewakili berbagai fitur dalam sistem, dapat disimpulkan bahwa seluruh test case yang diuji berhasil sesuai dengan harapan, seperti yang ditunjukkan pada Tabel 4.3.

#### b. Pengujian Non-Fungsional

Pengujian non-fungsional dilakukan untuk mengukur sejauh mana sistem atau aplikasi dapat digunakan dengan mudah oleh pengguna. Pengujian ini dilakukan terhadap empat responden menggunakan System Usability Scale (SUS), terdiri dari 16 pertanyaan terkait pengalaman pengguna dalam menggunakan sistem informasi. Metode penilaian dalam pengujian ini menggunakan skala 1-5, dengan pertanyaan ganjil (1, 3, 5, dll.) merupakan pertanyaan positif, di mana semakin tinggi nilai, semakin baik pengalaman pengguna. Sedangkan pertanyaan genap (2, 4, 6, dll.) merupakan pertanyaan negatif, di mana nilai yang lebih rendah menunjukkan pengalaman pengguna yang lebih baik.

Skala Penilaian:

- 1: Sangat Tidak Setuju
- 2: Tidak Setuju
- 3: Netral
- 4: Setuju
- 5: Sangat Setuju

Metode penilaian SUS dilakukan dengan cara:

1. Untuk pertanyaan bernomor ganjil, nilai yang diberikan responden dikurangi 1
2. Untuk pertanyaan bernomor genap, nilai 5 dikurangi nilai yang diberikan responden
3. Hasil perhitungan dari semua pertanyaan dijumlahkan
4. Total nilai kemudian dikalikan 1,5625 untuk mendapatkan skor akhir SUS dalam rentang 0 hingga 100.

Tabel 4.4 Pengujian non-fungsional

No.	Pertanyaan
1	Saya merasa mudah untuk menginstal aplikasi melalui browser.
2	Saya merasa proses instalasi aplikasi terlalu rumit.
3	Proses instalasi aplikasi berlangsung dengan cepat.
4	Saya merasa tidak ada konfirmasi yang jelas mengenai keberhasilan instalasi.

5	Ikon aplikasi mudah ditemukan di Home Screen atau Desktop setelah instalasi.
6	Saya merasa kesulitan mengakses aplikasi melalui ikon di Home Screen atau Desktop.
7	Tampilan aplikasi berfungsi dengan baik pada berbagai ukuran layar.
8	Menu navigasi aplikasi sulit diakses.
9	Transisi antar halaman dalam aplikasi berjalan dengan lancar dan cepat.
10	Tata letak antarmuka aplikasi tidak konsisten di seluruh halaman.
11	Ukuran font dalam aplikasi mudah dibaca.
12	Saya merasa scrolling dalam aplikasi tidak lancar.
13	Aplikasi berjalan dengan lancar dan tanpa gangguan (lag).
14	Fitur-fitur dalam aplikasi sering kali tidak berfungsi dengan baik.
15	Saya merasa saya dapat menggunakan aplikasi tanpa bantuan.
16	Saya merasa aplikasi ini terlalu kompleks untuk digunakan.

Tabel 4.5 Responden pengujian non-fungsional

No.	Nama Responden
1	Dhea Khansa Nabila
2	Arif Budiawan
3	Zunanda Ibrahim
4	Bintang Ahmad Ramadhan

Tabel 4.6 Hasil kuesioner pengujian non-fungsional

No.	Responden 1	Responden 2	Responden 3	Responden 4
Q1	4	4	5	5
Q2	1	2	1	1
Q3	5	5	5	5
Q4	2	1	1	2
Q5	5	4	5	5
Q6	1	1	2	1
Q7	5	4	5	5
Q8	1	2	2	2
Q9	4	4	5	5
Q10	1	2	1	1
Q11	5	5	4	4
Q12	1	1	2	2
Q13	5	4	4	4

Q14	2	2	1	1
Q15	5	4	5	4
Q16	1	2	2	2

Tabel 4.7 Hasil pengujian non-fungsional

Nama Responden	Total Skor	Nilai SUS
Responden 1	60	93,75
Responden 2	52	81,25
Responden 3	56	87,5
Responden 2	60	93,75
Rata-rata		89,06

Berdasarkan hasil pengujian System Usability Scale (SUS) yang dilakukan terhadap 4 responden, aplikasi ini memiliki tingkat usability yang sangat baik dengan skor rata-rata 89,06, seperti yang dapat dilihat pada Tabel 4.7.

## 4.2 Kelebihan dan Kekurangan Sistem

### Kelebihan Sistem

- a) Sistem dapat menghasilkan tagihan bulanan secara otomatis, sehingga meminimalkan kesalahan manual dan menghemat waktu.
- b) Sistem dapat mengirimkan notifikasi pembayaran tagihan melalui WhatsApp, memastikan penghuni selalu mendapatkan informasi pembayaran tepat waktu.
- c) Sistem mendukung instalasi web sebagai Progressive Web App (PWA) pada perangkat yang mendukung, memungkinkan akses cepat dan mudah dari layar utama perangkat pengguna.
- d) Sistem memiliki antarmuka pengguna yang ramah dan mudah digunakan, memudahkan pengguna untuk menavigasi dan mengelola informasi.
- e) Sistem memungkinkan penghuni untuk memantau status pembayaran mereka secara real-time, memberikan transparansi dan kontrol penuh atas keuangan mereka.

### Kekurangan Sistem

- a) Fitur chat tidak memiliki dukungan untuk pengiriman file atau multimedia, yang dapat membatasi komunikasi antara penghuni dan admin.
- b) Sistem belum mengimplementasikan fitur-fitur PWA secara maksimal, seperti notifikasi push langsung dari aplikasi dan dukungan penuh untuk mode offline.

## BAB V

### KESIMPULAN DAN SARAN

#### 5.1 Kesimpulan

Berdasarkan penelitian yang telah dilakukan, sistem informasi kos-kosan berbasis Progressive Web Application (PWA) berhasil diimplementasikan menggunakan *framework Laravel* dan *database MySQL*, dengan metodologi pengembangan perangkat lunak *Waterfall*. Metode ini dipilih karena tahapan pengembangannya yang terstruktur dan sesuai dengan kebutuhan sistem yang telah terdefinisi sejak awal. Pengumpulan data dilakukan melalui observasi langsung pada kos-kosan serta perbandingan dengan penelitian sejenis.

Sistem yang dikembangkan memiliki dua jenis pengguna, yaitu penghuni kos sebagai pengguna dan pengelola sistem sebagai admin. Pada tahap perancangan, dibuat diagram use case dan *activity diagram* untuk memodelkan alur kerja sistem, serta perancangan antarmuka dengan pendekatan *mobile-first* untuk meningkatkan pengalaman pengguna di perangkat mobile. Aset dan file PWA juga dirancang untuk mendukung fitur-fitur aplikasi agar dapat diakses selayaknya aplikasi native tanpa perlu melalui toko aplikasi.

Pengujian sistem dilakukan menggunakan metode *blackbox testing* dengan jumlah test case sebanyak 82, dan seluruhnya berhasil sesuai harapan. Hal ini menunjukkan bahwa fungsi-fungsi pada sistem berjalan sesuai dengan spesifikasi yang dirancang. Dengan demikian, sistem informasi kos-kosan ini mampu mempermudah proses manajemen kos-kosan, baik dari sisi admin maupun penghuni kos, serta mendukung aksesibilitas yang lebih baik melalui teknologi PWA.

Secara keseluruhan, sistem informasi kos-kosan ini berhasil diimplementasikan sesuai dengan tujuan awal penelitian, yaitu memudahkan proses manajemen kos-kosan, baik dari sisi penghuni kos dalam memproses pembayaran kos-kosan, maupun dari sisi admin yang mengelola data kos-kosan dan penghuni. Dengan dukungan teknologi PWA, sistem ini juga mampu memberikan pengalaman pengguna yang lebih baik. Implementasi ini diharapkan dapat menjadi solusi efektif bagi pengelola kos-kosan dalam menghadapi tantangan manajemen yang lebih kompleks di era digital ini.

#### 5.2 Saran

Sistem informasi kos-kosan berbasis Progressive Web Application (PWA) yang telah diimplementasikan dalam penelitian ini sudah memenuhi tujuan dan rumusan masalah yang

dijelaskan pada bab pertama. Dari segi fungsionalitas, sistem telah berjalan dengan baik, namun masih terdapat beberapa kekurangan yang dapat menjadi perhatian untuk pengembangan lebih lanjut. Oleh karena itu, penelitian selanjutnya diharapkan dapat melakukan pengembangan terhadap beberapa fitur yang masih kurang optimal. Beberapa saran perbaikan untuk peningkatan sistem ke depannya adalah sebagai berikut:

a) Peningkatan Fitur Chat

Fitur chat pada sistem saat ini masih terbatas hanya pada pengiriman teks, tanpa dukungan untuk pengiriman file atau multimedia (seperti gambar, video, atau dokumen). Hal ini membatasi komunikasi antara penghuni dan admin, terutama jika ada kebutuhan untuk saling berbagi bukti pembayaran atau gambar terkait kondisi kamar. Pengembangan lebih lanjut disarankan untuk menambahkan dukungan pengiriman file, sehingga interaksi antar pengguna dan admin menjadi lebih efisien dan fleksibel.

b) Optimalisasi Fitur PWA

Meskipun sistem ini telah diimplementasikan menggunakan teknologi PWA, beberapa fitur unggulan PWA belum dimanfaatkan secara maksimal. Fitur seperti push notification yang memungkinkan aplikasi mengirimkan notifikasi secara langsung kepada pengguna tanpa perlu membuka aplikasi, belum diterapkan. Fitur ini akan sangat berguna untuk mengingatkan pengguna tentang pembayaran, pembaruan status kamar, atau pengumuman penting dari admin. Selain itu, dukungan penuh untuk mode offline juga belum optimal, terutama pada pengelolaan data saat pengguna tidak terhubung ke internet. Dengan pengembangan lebih lanjut, sistem ini dapat menjadi lebih responsif dan memberikan pengalaman pengguna yang lebih baik, meskipun dalam kondisi jaringan yang tidak stabil.

Dengan adanya pengembangan pada kedua aspek ini, diharapkan sistem informasi kos-kosan berbasis PWA ini dapat lebih optimal dalam memenuhi kebutuhan pengguna, baik dari sisi penghuni kos maupun admin. Pengguna diharapkan dapat berinteraksi dengan sistem secara lebih mudah dan fleksibel, serta mendapatkan akses yang lebih baik terhadap fitur-fitur yang mendukung kenyamanan dan kemudahan dalam pengelolaan kos-kosan.

## DAFTAR PUSTAKA

- Abdul Wahid, A. (2020). Analisis Metode Waterfall Untuk Pengembangan Sistem Informasi. *Jurnal Ilmu-Ilmu Informatika Dan Manajemen STMIK*, (November), 1–5.
- Aldi, M., Ghiffari, A., Praptono, I. B., & Sagita, H. (2020). Perancangan Sistem Informasi Manajemen Pengelolaan Hunian Sewa Berbasis Website Pada Kos-Kosan Pondok Salma. *Telkom University Open Library*, 7(2), 6106–6113.
- Amrullah, A., Salim, Y., & Rachman Manga, A. (2021). Implementasi Progressive Web App Terhadap Aplikasi E-Commerce Sebagai Solusi Untuk Meningkatkan Kinerja Aplikasi Berbasis Web INFORMASI ARTIKEL ABSTRAK. *Buletin Sistem Informasi Dan Teknologi Islam*, 2(3), 213–221.
- Astuti, R. (2009). Pemodelan Analisis Berorientasi Objek dengan Use Case. *Media Informatika*, Vol. 8, pp. 73–81. Retrieved from [https://jurnal.likmi.ac.id/Jurnal/7\\_2009/Pemodelan\\_Analisis\\_rini\\_.pdf](https://jurnal.likmi.ac.id/Jurnal/7_2009/Pemodelan_Analisis_rini_.pdf)
- Choldun, I., Fauziah, D., & Sentanu, B. D. (2022). Perancangan Sistem Informasi Pengelolaan Kost Putri Sufia. *Improve*, 14(1), 14–17. Retrieved from <https://ejurnal.poltekpos.ac.id/index.php/improve/article/view/2170%0Ahttps://ejurnal.poltekpos.ac.id/index.php/improve/article/download/2170/1007>
- Christianto, C., & Isputrawan, F. (2022). Pengembangan Aplikasi E-Kost Berbasis Website Menggunakan Metode Microservice. *JBASE - Journal of Business and Audit Information Systems*, 5(1), 52–65. <https://doi.org/10.30813/jbase.v5i1.3472>
- Dharwiyanti, S. (2013). *Pengantar Unified Modeling Language (UML)*. <http://rosni-gj.staff.gunadarma.ac.id/Downloads/files/14321/10.+Unified+Modeling+Language.pdf>. Diakses tanggal: 3 Mei 2021. (pp. 1–13). pp. 1–13.
- Fadhilah, J., Layyinna, C. A. A., Khatami, R., & Fitroh, F. (2021). Pemanfaatan Teknologi Digital Wallet Sebagai Solusi Alternatif Pembayaran Modern: Literature Review. *Journal of Computer Science and Engineering (JCSE)*, Vol. 2, pp. 89–97. <https://doi.org/10.36596/jcse.v2i2.219>
- Fatimah Isnay Nur Alvivi, S., & Voutama, A. (2024). Pengembangan Sistem Manajemen Kos Berbasis Web Di Kos Jannati. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 8(2), 1765–1774. <https://doi.org/10.36040/jati.v8i2.7891>
- Fitri Zuyina Nur Azizah, Adine Pramesti Gitaswari, Dwiana Maryamas Sofa, Rexana Widyadhana Agusta, & Zulva Nur Fadila. (2023). Sistem Informasi Pengelolaan Data Kos

- Annisa Menggunakan Framework Code Igniter 4. *Jurnal Informatika Dan Multimedia*, Vol. 14, pp. 7–14. <https://doi.org/10.33795/jim.v14i2.467>
- Herman, H., & Frederick, F. (2023). Progressive Web Apps: Pengembangan dan Studi Penerimaan pada Mahasiswa Indonesia Menggunakan Scrum dan UTAUT. *Jurnal Teknologi Terpadu*, 9(1), 22–28. <https://doi.org/10.54914/jtt.v9i1.603>
- Hidayat, R., Hasbi, H. F., Putri, D. M., Ningrum, S. W., Supriani, N., & Farida, M. (2023). Rancangan User Interface untuk Meningkatkan Performa Pada E-Commerce Menggunakan Progressive Web Apps (PWA). *Technology and Informatics Insight Journal*, 2(2), 70–82. <https://doi.org/10.32639/tij.v2i2.781>
- Ilhami, M. (2021). Tren dan Peluang Cross-Platform Mobile App untuk Developer Pemula. *KONSTELASI: Konvergensi Teknologi Dan Sistem Informasi*, 1(2), 402–411. <https://doi.org/10.24002/konstelasi.v1i2.4320>
- Khumaidah, & Riyantomo, A. (2019). Sistem Informasi Manajemen Kost Menggunakan Framework Laravel (Studi Kasus Pada Kost Griya Stonen). *Jurnal Ilmiah Cendekia Eksakta*, 1(1), 112–118.
- Kurniawan, A. A. (2020). Analisis Performa Progressive Web Application (Pwa) Pada Perangkat Mobile. *Jurnal Ilmiah Informatika Komputer*, 25(1), 18–31. <https://doi.org/10.35760/ik.2020.v25i1.2510>
- Luh, N., Ambaradewi, G., Putu, N., Resmiati, A., Studi, P., Informatika, T., ... Kusuma, A. H. (2021). MANAJEMEN INFORMASI RUMAH KOST “ Restra .” *Jurnal Algoritma*, 17(2), 29–41.
- Matiini, G., Setiyadi, R., Setiawan, A., & Ramli, M. (2021). Pengembangan Aplikasi Progressive Web Application (PWA) Untuk Pembelajaran dan Evaluasi Kelas English Grammar Online Course. *Jurnal Pendidikan Edutama*, 8(2), 163. <https://doi.org/10.30734/jpe.v8i2.984>
- Maysaroh, S., & Diansyah, D. (2022). Pengaruh Peer To Peer Lender (P2P) Dan Payment Gateway Terhadap Kinerja UMKM Pada Masa Pandemi Covid-19 Dengan E-commerce Sebagai Variabel Moderating. *Business Management Journal*, Vol. 18, p. 131. <https://doi.org/10.30813/bmj.v18i2.3021>
- Nitami, A., Munthe, A. A., & Masrizal. (2021). Sistem Informasi Reservasi Hotel Rantauprapat Berbasis Web dengan Framework Codeigniter. *Journal of Student Development Information System (JoSDIS)*, Vol. 1, pp. 7–17.
- Nugraha, D., Anjara, F., & Faizah, S. (2022). Comparison of Web Based and PWA in Online

- Learning. *Proceedings of the 5th FIRST T1 T2 2021 International Conference (FIRST-T1-T2 2021)*, 9. <https://doi.org/10.2991/ahe.k.220205.035>
- Nurwanto, N. (2019). Penerapan Progressive Web Application (PWA) pada E-Commerce. *Techno.Com*, 18(3), 227–235. <https://doi.org/10.33633/tc.v18i3.2400>
- Putra, I. B. P. A., Satwika, I. P., & Nirmala, B. P. W. (2020). Penerapan Progressive Web App (PWA) Pada Aplikasi Manajemen Dokter Gigi (Studi Kasus : Praktek Drg. Ketut Astina). *Journal of Informatics Engineering and Technology*, 1(1), 50–58.
- Rahmatya, M. D., Simangunsong, D. E. S., & Wicaksono, M. F. (2022). e-Kos sebagai Sistem Informasi Pengelolaan Kos pada Mazasi's House. *Jurnal Teknologi Dan Informasi*, 12(2), 176–190. <https://doi.org/10.34010/jati.v12i2.8027>
- Rahmawati, M., & Harahap, E. (2021). Analisis Keuntungan Usaha Kos-Kosan Menggunakan Program Linear Dengan Aplikasi Geogebra Profit Analysis of a Boarding House Business using Linear Programming with Geogebra Application. *Jurnal Matematika*, Vol. 20, pp. 59–65.
- Rosalina Widyayanti, E. (2020). Pengaruh Kecenderungan Pergeseran Sistem Pembayaran Dari Tunai ke Non-Tunai/Online Payment Terhadap Peningkatan Pendapatan Usaha (Studi pada UMKM di Yogyakarta). *STIE Widya Wiwaha*, pp. 187–200. Retrieved from <http://eprint.stieww.ac.id/1102/1/14 Evi Rosalina Widyayanti.pdf>
- Siking, A., Koniyo, M. H., & Yassin, R. M. T. (2023). Unified Modelling Language (UML) dalam Perancangan Sistem Informasi Pelayanan Pengujian Material Berbasis Web Pada Dinas Pekerjaan Umum dan Penataan Ruang Provinsi Gorontalo. *Journal of System and Information Technology*, 3(2), 204–213.
- Tahirshah, F. S. (2019). *Comparison between Progressive Web App and Regular Web App*. Retrieved from [www.bth.se](http://www.bth.se)
- Verma, A., Khatana, A., & Chaudhary, S. (2017). A Comparative Study of Black Box Testing and White Box Testing. *International Journal of Computer Sciences and Engineering*, 5(12), 301–304. <https://doi.org/10.26438/ijcse/v5i12.301304>
- Wicaksono, A. P., Kharisma, A. P., & Sianturi, R. S. (2022). Pengembangan Aplikasi Mobile Manajemen Indekos berbasis Android dengan Metode Human-Centered Design (Studi Kasus: Kos Bu Parjo). *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, Vol. 6, pp. 3596–3605. Retrieved from <https://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/11398>
- Zait, A., Berteau, P. E., Gove, Jenny, Heilmann Chris, Dutton Sam, Clark Sarah, Kurtuldu

Mustafa, Warrender Ryan, Cassells Shane, Demidova Olga, A. C., Fabris, N., Luburic, R., Rcvd, S., ... Γ.Δουμουλάκης. (2018). Evaluation and Implementation of Progressive Web Application. *International Journal of Interactive Mobile Technologies (IJIM)*, Vol. 7, pp. 1–8. Retrieved from [http://www.theseus.fi/bitstream/handle/10024/142997/PWAthesis.pdf?sequence=1&isAllowed=y%0Ahttp://ccf.ee.ntu.edu.tw/~cchen/course/simulation/lec13.ppt.%0Awww.ijarcsse.com%0Ahttps://www.academia.edu/35910194/Design\\_and\\_Implementation\\_of\\_Treasury\\_Applicati](http://www.theseus.fi/bitstream/handle/10024/142997/PWAthesis.pdf?sequence=1&isAllowed=y%0Ahttp://ccf.ee.ntu.edu.tw/~cchen/course/simulation/lec13.ppt.%0Awww.ijarcsse.com%0Ahttps://www.academia.edu/35910194/Design_and_Implementation_of_Treasury_Applicati)

Zaki Hasibuan, M. I., & Suendri, S. (2023). Implementasi Hybrid App Framework dalam Membangun Aplikasi Kesehatan Mental berbasis Mobile. *Edumatic: Jurnal Pendidikan Informatika*, 7(1), 141–150. <https://doi.org/10.29408/edumatic.v7i1.17478>

# LAMPIRAN

## Lembar Kuesioner Pengujian Sistem

Nama : *Dhea Khansa Nabala*

### A. Pengujian Fungsional

Jawablah dengan memberikan tanda centang pada kolom jawaban yang tersedia, apakah "Gagal" atau "Berhasil."

No.	Status		No.	Status	
	Gagal	Berhasil		Gagal	Berhasil
1.1		✓	4.3		✓
1.2		✓	5.1		✓
1.3		✓	6.1		✓
2.1		✓	6.2		✓
2.2		✓	7.1		✓
2.3		✓	7.2		✓
2.4		✓	7.3		✓
2.5		✓	8.1		✓
2.6		✓	9.1		✓
3.1		✓	10.1		✓
4.1		✓	10.2		✓
4.2		✓			

### B. Pengujian Non-Fungsional

Jawablah pertanyaan berikut dengan memberi tanda centang pada kolom jawaban yang tersedia. Penilaian dilakukan dalam skala sebagai berikut: 1 = Sangat Tidak Setuju, 2 = Tidak Setuju, 3 = Netral, 4 = Setuju, 5 = Sangat Setuju.

No.	1	2	3	4	5	No.	1	2	3	4	5
Q1				✓		Q9				✓	
Q2	✓					Q10	✓				
Q3					✓	Q11					✓
Q4		✓				Q12	✓				
Q5					✓	Q13					✓
Q6	✓					Q14		✓			
Q7					✓	Q15					✓
Q8	✓					Q16	✓				

Yogyakarta, 09 November 2024  
Responded



( *Dhea Khansa Nabala* )

**Lembar Kuesioner Pengujian Sistem**

Nama : **Arif Budhawan**

**A. Pengujian Fungsional**

Jawablah dengan memberikan tanda centang pada kolom jawaban yang tersedia, apakah "Gagal" atau "Berhasil."

No.	Status		No.	Status	
	Gagal	Berhasil		Gagal	Berhasil
1.1		✓	4.3		✓
1.2		✓	5.1		✓
1.3		✓	6.1		✓
2.1		✓	6.2		✓
2.2		✓	7.1		✓
2.3		✓	7.2		✓
2.4		✓	7.3		✓
2.5		✓	8.1		✓
2.6		✓	9.1		✓
3.1		✓	10.1		✓
4.1		✓	10.2		✓
4.2		✓			

**B. Pengujian Non-Fungsional**

Jawablah pertanyaan berikut dengan memberi tanda centang pada kolom jawaban yang tersedia. Penilaian dilakukan dalam skala sebagai berikut: 1 = Sangat Tidak Setuju, 2 = Tidak Setuju, 3 = Netral, 4 = Setuju, 5 = Sangat Setuju.

No.	1	2	3	4	5	No.	1	2	3	4	5
Q1				✓		Q9				✓	
Q2		✓				Q10		✓			
Q3					✓	Q11					✓
Q4	✓					Q12	✓				
Q5				✓		Q13				✓	
Q6	✓					Q14		✓			
Q7				✓		Q15				✓	
Q8		✓				Q16		✓			

Yogyakarta, 8 November 2024  
Responded



( Arif Budhawan )

**Lembar Kuesioner Pengujian Sistem**

Nama : Zunanda Ibrahim

**A. Pengujian Fungsional**

Jawablah dengan memberikan tanda centang pada kolom jawaban yang tersedia, apakah "Gagal" atau "Berhasil."

No.	Status		No.	Status	
	Gagal	Berhasil		Gagal	Berhasil
1.1		✓	4.3		✓
1.2		✓	5.1		✓
1.3		✓	6.1		✓
2.1		✓	6.2		✓
2.2		✓	7.1		✓
2.3		✓	7.2		✓
2.4		✓	7.3		✓
2.5		✓	8.1		✓
2.6		✓	9.1		✓
3.1		✓	10.1		✓
4.1		✓	10.2		✓
4.2		✓			

**B. Pengujian Non-Fungsional**

Jawablah pertanyaan berikut dengan memberi tanda centang pada kolom jawaban yang tersedia. Penilaian dilakukan dalam skala sebagai berikut: 1 = Sangat Tidak Setuju, 2 = Tidak Setuju, 3 = Netral, 4 = Setuju, 5 = Sangat Setuju.

No.	1	2	3	4	5	No.	1	2	3	4	5
Q1					✓	Q9					✓
Q2	✓					Q10	✓				
Q3					✓	Q11				✓	
Q4	✓					Q12		✓			
Q5					✓	Q13				✓	
Q6		✓				Q14	✓				
Q7					✓	Q15					✓
Q8		✓				Q16		✓			

Yogyakarta, 10 November 2024  
Responded

  
 ( Zunanda Ibrahim )

**Lembar Kuesioner Pengujian Sistem**

Nama : *Bintang Ahmad Kanadhon*

**A. Pengujian Fungsional**

Jawablah dengan memberikan tanda centang pada kolom jawaban yang tersedia, apakah "Gagal" atau "Berhasil."


No.	Status		No.	Status	
	Gagal	Berhasil		Gagal	Berhasil
1.1		✓	4.3		✓
1.2		✓	5.1		✓
1.3		✓	6.1		✓
2.1		✓	6.2		✓
2.2		✓	7.1		✓
2.3		✓	7.2		✓
2.4		✓	7.3		✓
2.5		✓	8.1		✓
2.6		✓	9.1		✓
3.1		✓	10.1		✓
4.1		✓	10.2		✓
4.2		✓			

**B. Pengujian Non-Fungsional**

Jawablah pertanyaan berikut dengan memberi tanda centang pada kolom jawaban yang tersedia. Penilaian dilakukan dalam skala sebagai berikut: 1 = Sangat Tidak Setuju, 2 = Tidak Setuju, 3 = Netral, 4 = Setuju, 5 = Sangat Setuju.

No.	1	2	3	4	5	No.	1	2	3	4	5
Q1					✓	Q9					✓
Q2	✓					Q10	✓				
Q3					✓	Q11				✓	
Q4		✓				Q12		✓			
Q5					✓	Q13				✓	
Q6	✓					Q14	✓				
Q7					✓	Q15				✓	
Q8		✓				Q16	✓				

Yogyakarta, 11 November 2024  
Responded

  
 ( *Bintang Ahmad Kanadhon* )