

PERANCANGAN SISTEM MANAJEMEN KOS MENGGUNAKAN METODE SDLC AGILE DEVELOPMENT UNTUK MENINGKATKAN EFISIENSI PENGELOLAAN PENGHUNI DAN KEUANGAN

I Kadek Indra Mahayana¹, Putri Agung Permata Sari²

^{1,2}Program Studi Teknologi Informasi, Fakultas Teknik, Universitas Mahasaraswati Denpasar
Email: putriagung@unmas.ac.id

ABSTRAK: Pertumbuhan usaha properti sewa khususnya rumah kos di kota-kota besar Indonesia mendorong kebutuhan akan sistem pengelolaan yang lebih efisien. Permasalahan yang dihadapi pemilik kos di daerah Badung meliputi pencatatan data penghuni secara manual, monitoring masa sewa yang tidak terpantau, ketidakakuratan pencatatan pembayaran, dan tidak adanya laporan keuangan yang komprehensif. Penelitian ini bertujuan merancang aplikasi manajemen kos berbasis web menggunakan metode System Development Life Cycle (SDLC) Agile Development dengan pendekatan Object-Oriented Analysis and Design (OOAD). Data dikumpulkan melalui wawancara semi-terstruktur, kemudian dianalisis menggunakan teknik mapping untuk merumuskan kebutuhan sistem. Hasil penelitian menghasilkan artefak OOAD yang komprehensif meliputi use case diagram, activity diagram, sequence diagram, dan class diagram, serta perencanaan enam sprint Agile. Aplikasi dirancang dengan sepuluh kebutuhan fungsional dan lima kebutuhan non-fungsional yang mencakup manajemen data penghuni, pencatatan pembayaran, notifikasi jatuh tempo otomatis, pencatatan biaya maintenance, dan pelaporan keuangan real-time. Pengujian black box testing terhadap seluruh modul menunjukkan hasil yang valid. Penelitian ini menyimpulkan bahwa aplikasi yang dirancang berpotensi meningkatkan efisiensi pengelolaan penghuni dan keuangan secara signifikan, mengurangi potensi kesalahan pencatatan, serta memberikan visibilitas kondisi usaha yang lebih baik bagi pemilik kos.

Kata kunci: sistem manajemen, Agile Development, OOAD, UML, sistem informasi berbasis web

ABSTRACT: The growth of rental property businesses, particularly boarding houses (kos), in major Indonesian cities has driven the need for more efficient management systems. Problems faced by boarding house owners in the Badung area include manual tenant data recording, unmonitored lease period tracking, inaccurate payment records, and the absence of comprehensive financial reporting. This study aims to design a web-based boarding house management application using the System Development Life Cycle (SDLC) Agile Development method with an Object-Oriented Analysis and Design (OOAD) approach. Data were collected through semi-structured interviews with a boarding house owner managing 24 rooms, then analyzed using a mapping technique to formulate system requirements. The research produced comprehensive OOAD artifacts including use case diagrams, activity diagrams, sequence diagrams, and class diagrams, along with a six-sprint Agile development plan. The application was designed with ten functional requirements and five non-functional requirements covering tenant data management, payment recording, automatic due date notifications, maintenance cost tracking, and real-time financial reporting. Black box testing across all modules yielded valid results. This study concludes that the designed application has the potential to significantly improve tenant and financial management efficiency, reduce recording errors, and provide better business visibility for boarding house owners.

Keywords: Management system, Agile Development, OOAD, UML, web-based information system

PENDAHULUAN

Perkembangan usaha properti sewa, khususnya rumah kos, mengalami pertumbuhan signifikan di kota-kota besar Indonesia seiring meningkatnya jumlah mahasiswa dan tenaga kerja pendatang. Tingginya permintaan hunian sementara menjadikan usaha kos sebagai sektor properti dengan prospek yang menjanjikan, namun sebagian besar pengelolaan masih dilakukan secara konvensional tanpa dukungan sistem teknologi informasi yang memadai (Setiawan dkk., 2021). Ketergantungan pada metode manual menciptakan celah operasional yang berdampak langsung pada akurasi dan efisiensi pengelolaan secara keseluruhan.

Berdasarkan hasil wawancara dengan pemilik kos di Daerah Badung, ditemukan bahwa pengelolaan dilakukan secara manual. Permasalahan utama yang dihadapi meliputi: (1) data penghuni yang dicatat di buku sehingga mudah tercecer; (2) monitoring masa sewa yang tidak terpantau sehingga sering melewati jatuh tempo; (3) pencatatan pembayaran yang tidak akurat; (4) pengeluaran operasional yang tidak terorganisir; serta (5) tidak adanya laporan keuangan yang komprehensif. Pengelolaan kos yang masih dilakukan secara manual berisiko tinggi terhadap kesalahan pencatatan dan inefisiensi pelaporan, yang berdampak pada

kerugian operasional dan finansial bagi pemilik (Purnama, 2026).

Untuk mengatasi permasalahan tersebut, penelitian ini bertujuan merancang aplikasi manajemen kos berbasis web menggunakan metode *System Development Life Cycle (SDLC)* dengan pendekatan *Agile Development* dan *Object-Oriented Analysis and Design (OOAD)*. Metode *Agile* dipilih karena kemampuannya dalam mengakomodasi perubahan kebutuhan secara iteratif dan kolaboratif (Nova dkk., 2022), sementara *OOAD* digunakan untuk memodelkan sistem secara terstruktur melalui diagram UML yang mencakup use case diagram, activity diagram, sequence diagram, dan class diagram (Nasrull dan Sewaka, 2022).

Penerapan teknologi informasi dalam pengelolaan properti sewa terbukti mampu meningkatkan produktivitas dan mengurangi kesalahan administratif secara signifikan. Digitalisasi proses bisnis di sektor properti sewa tidak hanya meningkatkan efisiensi operasional, tetapi juga memperkuat kepercayaan penghuni terhadap transparansi pengelolaan (Prasetyo dan Dewantara, 2022). Dengan demikian, perancangan aplikasi ini tidak sekadar menjawab kebutuhan teknis pemilik kos, melainkan juga berkontribusi pada peningkatan kualitas layanan dan pengalaman penghuni secara keseluruhan (Setiadi dan Ulum, 2024).

JUDUL TEORI PENDUKUNG

Teori pendukung digunakan sebagai landasan konseptual dalam penelitian ini. Teori-teori yang dibahas berkaitan dengan Sistem Informasi Manajemen Kos dan teknologi yang digunakan dalam pengembangan sistem.

Sistem informasi manajemen kos

Sistem informasi manajemen kos merupakan perangkat lunak yang dirancang untuk membantu pemilik kos dalam mengelola seluruh aspek operasional usaha secara digital dan terintegrasi. Sistem ini mendefinisikan fungsi pengelolaan data penghuni, pencatatan pembayaran, dan pelaporan keuangan dalam satu platform terpadu yang dapat diakses kapan saja dan di mana saja (Nova dkk., 2022). Implementasi sistem informasi manajemen kos berbasis web terbukti mampu meningkatkan efisiensi operasional secara signifikan dibandingkan pengelolaan manual (Setiadi dan Ulum, 2024).

Digitalisasi pengelolaan properti sewa berkontribusi pada peningkatan kepuasan penyewa dan akurasi data keuangan pemilik

(Prasetyo dan Dewantara, 2022). Sistem informasi yang dirancang dengan baik memungkinkan pemilik kos untuk mengambil keputusan bisnis berdasarkan data yang akurat dan real-time, bukan lagi berdasarkan perkiraan atau estimasi kasar. Sistem manajemen kos berbasis web yang terintegrasi dengan modul keuangan mampu mengeliminasi praktik pencampuran keuangan usaha dan pribadi yang umum terjadi pada usaha kos skala kecil-menengah (Alvivi dan Voutama, 2024).

Agile Development

System Development Life Cycle (SDLC) merupakan serangkaian proses yang menjelaskan bagaimana perangkat lunak dirancang, dikembangkan, dipelihara, dan dibuat menjadi lebih efisien melalui beragam model dan pendekatan, salah satunya adalah model *Agile* yang bersifat iteratif dan adaptif terhadap perubahan kebutuhan (Andika dkk., 2024). Dalam konteks penelitian ini, metode *Agile* digunakan sebagai pendekatan *SDLC* yang paling sesuai karena sifatnya yang iteratif, adaptif, dan berpusat pada kolaborasi tim dan pengguna.

Metode *Agile Development* menekankan kolaborasi, fleksibilitas, dan pengiriman perangkat lunak yang berfungsi secara bertahap dalam siklus iterasi pendek yang disebut *sprint* (Nova dkk., 2022). *Agile* memberikan kemampuan adaptasi terhadap perubahan kebutuhan yang dinamis, menjadikannya pilihan utama dalam pengembangan aplikasi berbasis pengguna (Ramadhan, 2023). Implementasi *Agile* terbukti menghasilkan produk yang lebih sesuai dengan kebutuhan pengguna akhir dibandingkan metode konvensional seperti *Waterfall* (Halim dkk., 2024). Komparasi metodologi menunjukkan bahwa *Agile* unggul dalam proyek yang memiliki kebutuhan tidak pasti dan berubah-ubah, karakteristik yang sangat relevan dengan pengembangan sistem informasi untuk usaha kos (Zulfania, 2024).

Salah satu keunggulan utama *Agile* adalah kemampuannya menyertakan pengguna secara aktif dalam setiap iterasi pengembangan melalui mekanisme *sprint review* dan *retrospective*, di mana memungkinkan pemilik kos sebagai pengguna utama untuk memberikan umpan balik secara berkala di setiap akhir *sprint*, guna memastikan bahwa setiap fitur yang dibangun benar-benar menjawab permasalahan operasional yang dihadapi sehari-hari (Yarpiransa dkk., 2023).

Object-oriented Analysis and Design (OOAD)

Object-Oriented Analysis and Design (OOAD) merupakan pendekatan pengembangan perangkat lunak yang memandang sistem sebagai kumpulan objek yang saling berinteraksi. OOAD adalah cara baru dalam memandang masalah dengan menggunakan model yang sesuai dengan konsep dunia nyata, di mana setiap entitas direpresentasikan sebagai objek dengan atribut dan metode tersendiri (Nasrull dan Sewaka, 2022). Pendekatan ini memudahkan pemodelan sistem yang kompleks menjadi komponen-komponen yang terstruktur dan mudah dipahami oleh seluruh anggota tim pengembangan.

Dalam implementasinya, OOAD menggunakan *Unified Modeling Language (UML)* sebagai notasi standar untuk merepresentasikan rancangan sistem. UML menyediakan sekumpulan diagram seperti *use case*, *activity*, *sequence*, dan *class diagram* yang memungkinkan tim pengembang untuk mengkomunikasikan arsitektur sistem secara visual dan presisi (Supriatna dan Rima, 2022). Pendekatan *Object Oriented Analysis and Design (OOAD)* dalam perancangan sistem informasi menghasilkan struktur sistem yang terorganisir, fleksibel, dan mudah dikembangkan secara bertahap melalui pemodelan berorientasi objek yang terstruktur (Mahfuzhi dkk., 2025).

Software Requirements Specification (SRS)

Software Requirements Specification (SRS) adalah dokumen formal yang mendefinisikan kebutuhan fungsional dan non-fungsional dari suatu sistem perangkat lunak. SRS berfungsi sebagai kontrak antara pengembang dan pemangku kepentingan yang menggambarkan apa yang harus dilakukan oleh sistem secara terukur dan dapat diverifikasi (Andika dkk., 2024). Kebutuhan fungsional mencakup fitur-fitur utama yang harus dimiliki sistem, sedangkan kebutuhan non-fungsional mencakup aspek kualitas seperti performa, keamanan, dan kemudahan penggunaan (Rachmatika dkk., 2025).

Penyusunan SRS yang baik merupakan fondasi kritis keberhasilan pengembangan perangkat lunak, karena kesalahan dalam mendefinisikan kebutuhan pada tahap awal dapat mengakibatkan biaya perbaikan yang jauh lebih besar di tahap-tahap selanjutnya. Dalam konteks metodologi *Agile*, SRS tidak bersifat kaku, melainkan terus disempurnakan melalui

proses *backlog refinement* setiap *sprint* berlangsung. Pendekatan ini memastikan bahwa dokumen kebutuhan selalu mencerminkan pemahaman terkini dari seluruh pemangku kepentingan (Yarpiransa dkk., 2023). Integrasi antara SRS yang komprehensif dan proses *Agile* yang iteratif terbukti menghasilkan sistem yang lebih sesuai kebutuhan pengguna dan memiliki tingkat kepuasan yang lebih tinggi dibandingkan pengembangan berbasis dokumen kebutuhan statis (Zulfania, 2024).

METODE PENELITIAN

Penelitian ini menggunakan metode *System Development Life Cycle (SDLC) Agile Development* dengan pendekatan kualitatif deskriptif untuk fase analisis. Data primer dikumpulkan melalui wawancara terstruktur dengan pemilik kos di daerah Badung, sementara studi literatur digunakan untuk memperkuat landasan teoritis. Pendekatan OOAD diterapkan dalam fase perancangan sistem untuk menghasilkan model yang terstruktur dan dapat diimplementasikan (Andika dkk., 2024).

Pengumpulan data

Pengumpulan data primer dilakukan melalui wawancara semi-terstruktur dengan pemilik kos di daerah Badung yang mengelola 24 kamar. Wawancara mencakup 15 pertanyaan yang membahas proses bisnis pengelolaan kos, permasalahan yang dihadapi, dan kebutuhan sistem yang diharapkan. Hasil wawancara kemudian dianalisis menggunakan teknik mapping untuk mengidentifikasi kebutuhan fungsional dan non-fungsional sistem. Wawancara sebagai teknik pengumpulan data memungkinkan peneliti untuk menggali secara lebih mendalam aspek-aspek tertentu dari subjek penelitian, termasuk pandangan dan pengalaman, serta memberikan fleksibilitas dalam menyesuaikan alur percakapan sesuai respons narasumber, keunggulan yang tidak dimiliki oleh kuesioner yang terbatas dalam kedalaman informasi yang dapat diperoleh (Romdona dkk., 2025).

Selain wawancara, studi literatur dilakukan secara sistematis terhadap jurnal-jurnal ilmiah yang relevan dengan topik sistem informasi manajemen kos, metode *Agile*, dan pendekatan OOAD (Kurnia dan Wijaya, 2024).

Metode Agile Development

Pengembangan dilakukan dalam siklus *sprint* berdurasi dua minggu per iterasi. Setiap *sprint* terdiri dari *sprint planning*, *daily*

standup, *sprint review*, dan *sprint retrospective*. Pendekatan ini memastikan keterlibatan pemangku kepentingan secara berkelanjutan sehingga setiap iterasi menghasilkan perangkat lunak yang fungsional dan sesuai kebutuhan pengguna. Metode *Agile* terbukti efektif dalam pengembangan sistem informasi karena sifatnya yang fleksibel terhadap perubahan kebutuhan dan kemampuannya memfasilitasi interaksi intensif antara tim pengembang dengan pemangku kepentingan, sehingga menghasilkan produk yang sesuai dengan kebutuhan pengguna (Zulfania, 2024).

Tabel 1. Perencanaan *Sprint Agile* Development Aplikasi Manajemen Kos

Sprint	Durasi	Hasil
Sprint 1	2 Minggu	Analisis kebutuhan, SRS, wireframe UI/UX
Sprint 2	2 Minggu	Modul manajemen data penghuni dan kamar
Sprint 3	2 Minggu	Modul pencatatan pembayaran dan notifikasi
Sprint 4	2 Minggu	Modul keuangan dan biaya maintenance
Sprint 5	2 Minggu	Dashboard real-time dan laporan keuangan otomatis
Sprint 6	2 Minggu	Pengujian menyeluruh, perbaikan bug, deployment

Perencanaan enam *sprint* seperti yang disajikan pada Tabel 1 dirancang berdasarkan prioritas kebutuhan yang diperoleh dari hasil wawancara. *Sprint* pertama difokuskan pada fondasi analisis dan desain, sedangkan *sprint-sprint* berikutnya secara berurutan membangun modul-modul fungsional inti hingga pengujian akhir. Pendekatan bertahap ini memungkinkan deteksi dini terhadap potensi masalah teknis dan bisnis sebelum sistem diluncurkan secara penuh (Yarpriransa dkk., 2023).

HASIL DAN PEMBAHASAN

Berdasarkan hasil wawancara dengan narasumber, diperoleh gambaran komprehensif tentang proses bisnis pengelolaan kos dan permasalahan yang dihadapi. Narasumber mengungkapkan bahwa dari 24 kamar yang tersedia, tingkat hunian mencapai 20–22 kamar, menunjukkan tingginya permintaan tempat tinggal sementara di daerah Badung. Namun seluruh proses pengelolaan masih dilakukan secara manual, mencakup pencatatan data penghuni, monitoring masa sewa, pencatatan pembayaran, hingga penghitungan keuangan. Kondisi ini mengakibatkan beban kerja administrasi yang besar dan potensi kesalahan pencatatan.

Hasil *mapping* antara temuan wawancara, permasalahan yang diidentifikasi, dan kebutuhan sistem yang diperlukan disajikan pada Tabel 2.

Tabel 2. Hasil *Mapping* Wawancara ke Kebutuhan Sistem

Temuan	Masalah	Kebutuhan Sistem
Data penghuni dicatat manual di buku	Data mudah tercecer dan sulit dicari	Sistem menyimpan data penghuni secara digital
Monitoring masa sewa masih manual	Sering lupa jatuh tempo sewa	Fitur pengingat jatuh tempo otomatis
Pembayaran dicatat manual	Risiko lupa dan salah catat pembayaran	Pencatatan dan validasi pembayaran digital
Pengeluaran operasional tidak rapi	Sulit memantau arus kas bulanan	Modul pencatatan pemasukan dan pengeluaran
Biaya maintenance tercampur kebutuhan pribadi	Sulit menghitung keuntungan bersih	Kategori keuangan khusus maintenance
Tidak ada laporan keuangan otomatis	Pemilik sulit mengevaluasi usaha	Laporan keuangan otomatis berbasis data

Tabel 2 menunjukkan adanya korelasi langsung antara setiap temuan lapangan dengan permasalahan yang ditimbulkan dan solusi sistem yang dibutuhkan.

Software Requirements Specification (SRS)

Berdasarkan hasil analisis kebutuhan, dirumuskan *Software Requirements Specification (SRS)* yang mencakup kebutuhan fungsional dan non-fungsional sistem sebagaimana disajikan pada Tabel 3.

Tabel 3. *Software Requirements Specification (SRS)* Aplikasi Manajemen Kos

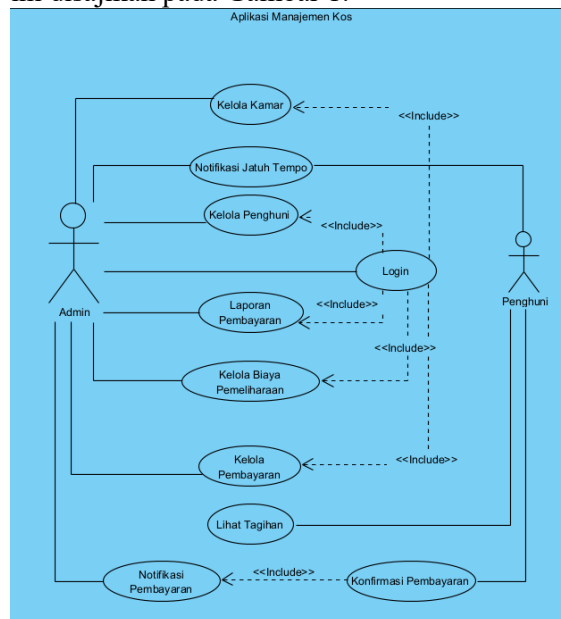
Kebutuhan Fungsional	Kebutuhan Non-Fungsional
1. Login 2. Mengelola data kamar 3. Mengelola data penghuni kos 4. Mencatat masa sewa penghuni 5. Konfirmasi pembayaran sewa	1. Sistem berbasis web dan mudah diakses 2. Antarmuka sederhana dan mudah dipahami 3. Data tersimpan aman dalam database 4. Sistem memproses data dengan cepat

6. Notifikasi jatuh tempo pembayaran	5. Dapat dikembangkan secara bertahap dengan Agile
7. Mencatat pemasukan dan pengeluaran	
8. Mencatat biaya maintenance bangunan	
9. Menghasilkan laporan keuangan otomatis	
10. Dashboard kondisi keuangan real-time	

Sepuluh kebutuhan fungsional yang dirumuskan dalam Tabel 3 mencerminkan seluruh aspek operasional pengelolaan kos, mulai dari manajemen data penghuni hingga pelaporan keuangan real-time. Sementara itu, lima kebutuhan non-fungsional menjamin bahwa sistem tidak hanya berfungsi dengan benar, tetapi juga dapat digunakan dengan mudah oleh pemilik kos yang tidak memiliki latar belakang teknologi.

Use Case Diagram

Identifikasi aktor dan *use case* pada sistem ini disajikan pada Gambar 1.



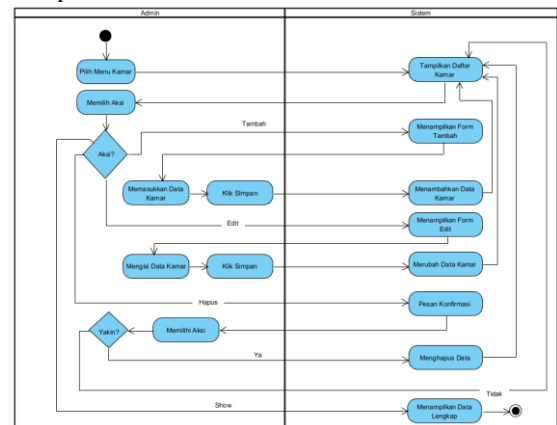
Gambar 1. Use Case Diagram

Pemisahan peran antara Admin (Pemilik Kos) dan Penghuni dalam use case diagram mencerminkan perbedaan tanggung jawab dan hak akses yang fundamental dalam operasional bisnis kos. Desain ini menerapkan prinsip least privilege, di mana setiap aktor hanya dapat mengakses fungsionalitas yang relevan dengan perannya, sehingga keamanan dan integritas data sistem dapat terjaga.

Activity Diagram

Activity diagram menggambarkan alur kerja proses bisnis dalam sistem secara visual,

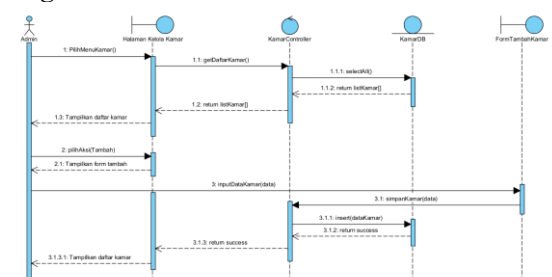
memperlihatkan urutan aktivitas dari awal hingga akhir beserta kondisi percabangan yang terjadi di dalamnya. Pada proses pengelolaan kamar, alur dimulai dari pemilik kos mengakses menu kamar, kemudian memilih aksi yang ingin dilakukan. Jika memilih tambah, sistem menampilkan form tambah dan pemilik memasukkan data kamar baru. Jika memilih edit, pemilik dapat mengubah data kamar yang sudah ada. Jika memilih hapus, sistem meminta konfirmasi terlebih dahulu sebelum menghapus data. Setiap aksi yang dilakukan akan divalidasi oleh sistem sebelum perubahan disimpan. Gambar 2 memperlihatkan *activity diagram* dari proses kelola kamar.



Gambar 2. Activity Diagram Kelola Kamar

Sequence Diagram

Gambar 3 memperlihatkan *sequence diagram* untuk skenario kelola kamar.

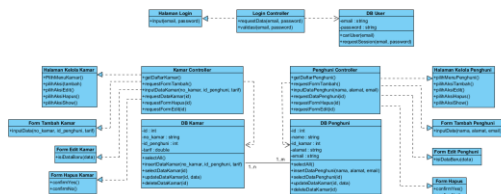


Gambar 3. Sequence Diagram Kelola Kamar

Pada skenario kelola kamar, alur interaksi dimulai dari Admin mengakses halaman kelola kamar, kemudian sistem memanggil controller untuk mengambil data kamar dari basis data dan menampilkan daftar kamar. Ketika Admin memilih tambah kamar, sistem menampilkan form tambah kamar. Admin mengisi dan mengirimkan data, controller memvalidasi dan menyimpan data baru ke basis data, kemudian mengembalikan status sukses kepada antarmuka pengguna.

Class Diagram

Class diagram dirancang untuk menggambarkan struktur statis sistem melalui identifikasi kelas, atribut, metode, serta hubungan antar objek yang terlibat dalam proses bisnis. *Class diagram* Sistem Manajemen Kos menggambarkan struktur kelas beserta hubungan antar kelas yang mendukung proses autentikasi, pengelolaan data kamar, dan pengelolaan data penghuni. Diagram ini menunjukkan interaksi antara antarmuka (form dan halaman), controller, serta basis data dalam menjalankan fungsi-fungsi utama sistem.

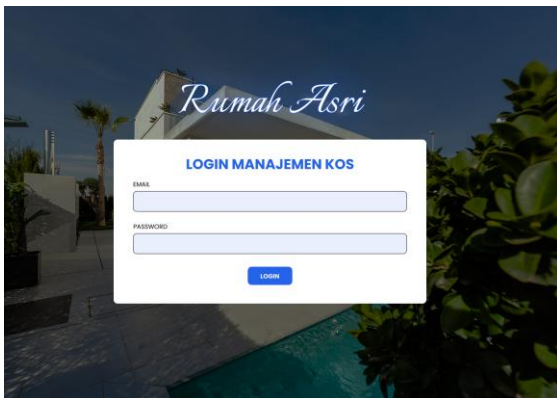


Gambar 4. *Class Diagram*

Desain User Interface dan User Experience

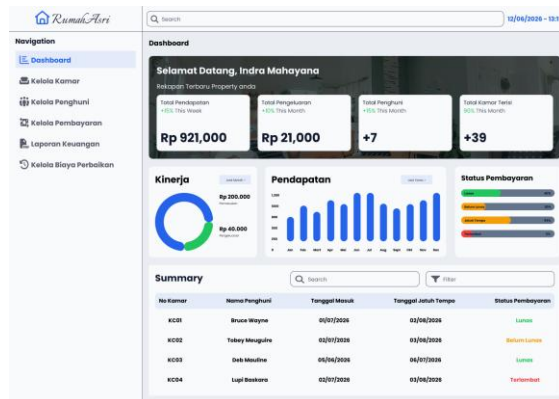
Perancangan UI/UX bertujuan untuk memastikan sistem tidak hanya berfungsi secara teknis, tetapi juga mudah digunakan oleh pengguna akhir. Pendekatan yang digunakan menekankan keterbacaan informasi, konsistensi elemen antarmuka, serta alur interaksi yang sederhana dan logis.

Halaman login digunakan sebagai proses autentikasi pengguna sebelum mengakses Sistem Manajemen Kos. Pengguna memasukkan email dan password, kemudian menekan tombol Login untuk masuk ke dalam sistem.



Gambar 5. Halaman Login

Halaman *Dashboard* menampilkan ringkasan informasi utama Sistem Manajemen Kos, meliputi statistik pendapatan, grafik kinerja, status pembayaran, dan data penghuni. Halaman ini juga dilengkapi menu navigasi untuk memudahkan pengguna mengakses seluruh fitur yang tersedia.



Gambar 6. Halaman *Dashboard*

Halaman Kelola Kamar digunakan untuk mengelola data kamar, meliputi informasi kamar, status, penghuni, dan tarif bulanan. Halaman ini juga menyediakan fitur pencarian, penyaringan, serta pengelolaan data kamar

No. Kamar	Tipe	Status	Nama Penghuni	Tarif Bulanan	Aksi
KC01	Single	Tersedia	-	900.000	[Edit] [Hapus]
KC02	Duo	Tersedia	-	1.200.000	[Edit] [Hapus]
KC03	Single	Tersedia	-	900.000	[Edit] [Hapus]
KC04	Single	Tersedia	-	900.000	[Edit] [Hapus]
KC05	Single	Tersedia	-	900.000	[Edit] [Hapus]
KC06	Single	Tersedia	-	900.000	[Edit] [Hapus]
KC07	Deluxe	Tersedia	-	1.800.000	[Edit] [Hapus]
KC08	Deluxe	Tersedia	Bruce Wayne	1.800.000	[Edit] [Hapus]
KC09	Duo	Tersedia	Dan Madsen	1.200.000	[Edit] [Hapus]
KC09	Duo	Tersedia	Lupin Berkas	1.200.000	[Edit] [Hapus]
KC10	Single	Tersedia	-	900.000	[Edit] [Hapus]
KC11	Single	Tersedia	-	900.000	[Edit] [Hapus]

Gambar 7. Halaman Kelola Kamar

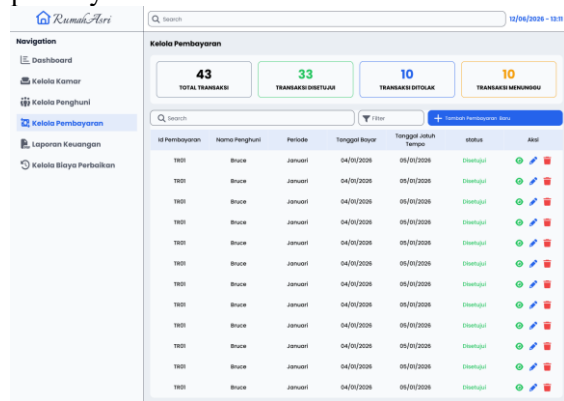
Halaman Kelola Penghuni menampilkan informasi identitas penghuni, nomor telepon, tanggal masuk, status, dan nomor kamar. Halaman ini juga menyediakan fitur pencarian, penyaringan, serta pengelolaan penambahan atau penghapusan data penghuni.

ID Penghuni	Nama Penghuni	No Telp	Tanggal Masuk	Status	No Kamar	Aksi
P01	Bruce	08192249	01/01/2026	Active	KC01	[Edit] [Hapus]
P02	Adi	08192249	01/01/2026	Active	KC02	[Edit] [Hapus]
P03	Michela	08192249	01/01/2026	Active	KC03	[Edit] [Hapus]
P04	Leon	08192249	01/01/2026	Active	KC04	[Edit] [Hapus]
P05	Sam	08192249	01/01/2026	Active	KC05	[Edit] [Hapus]
P06	ick	08192249	14/01/2026	Active	KC06	[Edit] [Hapus]
P07	Wid	08192249	26/01/2026	Keluar	-	[Edit] [Hapus]
P08	Chi yo	08192249	20/01/2026	Keluar	-	[Edit] [Hapus]
P09	Narto	08192249	30/01/2026	Keluar	-	[Edit] [Hapus]
P10	jean	08192249	30/01/2026	Active	KC07	[Edit] [Hapus]
P11	Sirkamdi	08192249	24/01/2026	Active	KC08	[Edit] [Hapus]
P12	Ayo	08192249	10/01/2026	Active	KC09	[Edit] [Hapus]

Gambar 8. Kelola Penghuni

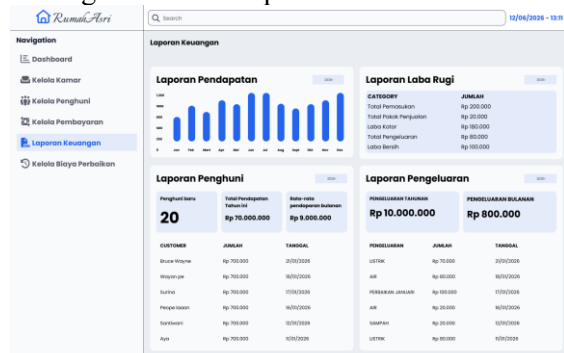
Halaman Kelola Pembayaran menampilkan informasi nama penghuni, periode tagihan, tanggal bayar, tanggal jatuh tempo, dan status pembayaran. Halaman ini juga menyediakan dasbor ringkasan status transaksi (disetujui, ditolak, menunggu), fitur pencarian,

penyaringan, serta pengelolaan persetujuan data pembayaran.



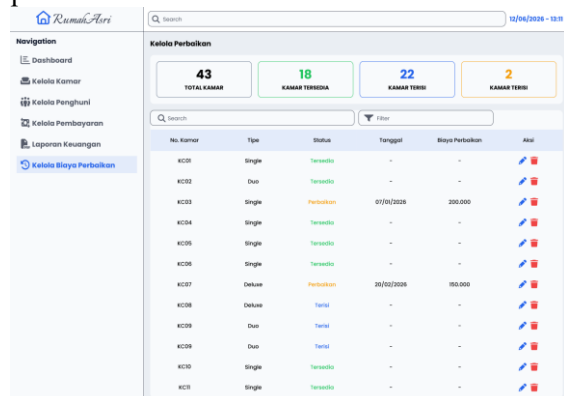
Gambar 9. Halaman Kelola Pembayaran

Halaman Laporan Keuangan menampilkan informasi grafik pendapatan bulanan, rincian laba rugi, daftar pemasukan dari penghuni, dan detail pengeluaran operasional. Halaman ini juga menyediakan dasbor visualisasi data, penyaringan tahun laporan, serta rekapitulasi keuangan secara komprehensif.



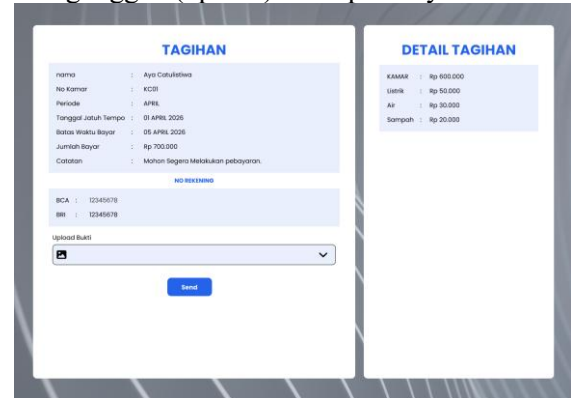
Gambar 10. Halaman Laporan Keuangan

Halaman Kelola Biaya Perbaikan digunakan untuk memantau status kamar dan biaya pemeliharaan, meliputi informasi nomor kamar, tipe kamar, status ketersediaan, tanggal, dan riwayat biaya perbaikan. Halaman ini juga menyediakan dasbor ketersediaan kamar, fitur pencarian, penyaringan, serta pengelolaan data perbaikan.



Gambar 11. Halaman Kelola Biaya Perbaikan

Halaman Bayar Tagihan digunakan untuk menampilkan instruksi dan memproses pembayaran sewa, meliputi informasi identitas penghuni, periode sewa, tanggal jatuh tempo, total jumlah bayar, rincian detail tagihan (kamar, listrik, air, sampah), dan nomor rekening tujuan. Halaman ini juga menyediakan fitur rincian biaya serta form untuk mengunggah (upload) bukti pembayaran.



Gambar 12. Halaman Bayar Tagihan

Black Box Testing

Aplikasi ini diuji menggunakan *black box testing* dengan hasil sebagai berikut:

Tabel 3. Hasil *Black box testing*

Kebutuhan	Skenario uji	Hasil yang diharapkan	Hasil pengujian
login	input data login (jika benar)	Tampil menu dashboard	Valid
	input data login (jika salah)	Muncul pesan error	Valid
dashboard	Input pencarian	Menampilkan data	Valid
Kelola kamar	CRUD data kamar	Menampilkan data dan data diproses sesuai perintah (insert, update, delete)	Valid
Kelola penghuni	CRUD data penghuni	Menampilkan data dan data diproses sesuai perintah (insert, update, delete)	Valid
Kelola pembayaran	CRUD data Pembayaran	Menampilkan data dan data diproses sesuai perintah (insert, update, delete)	Valid
Laporan Keuangan	Input data tahun	Menampilkan data sesuai dengan tahun	Valid

Kelola biaya perbaikan	CRUD data biaya perbaikan	Menampilkan data dan data diproses sesuai perintah (insert, update, delete)	Valid
Notifikasi email	Input tanggal jatuh tempo	Mengirim email ke penghuni dan admin	Valid
Pembayaran kamar	Input data transaksi	Mengirim data transaksi ke halaman pembayaran	Valid

SIMPULAN

Penelitian ini berhasil merancang aplikasi manajemen kos berbasis web menggunakan metode SDLC *Agile Development* dengan pendekatan *Object-Oriented Analysis and Design (OOAD)*. Berdasarkan hasil wawancara dengan pemilik kos di daerah Badung yang mengelola 24 kamar, teridentifikasi enam permasalahan utama pengelolaan manual yang kemudian dipetakan menjadi sepuluh kebutuhan fungsional dan lima kebutuhan non-fungsional dalam dokumen *Software Requirements Specification (SRS)*.

Proses perancangan menghasilkan artefak OOAD yang komprehensif, mencakup *use case diagram* dengan dua aktor utama (Admin dan Penghuni), *activity diagram* alur kelola kamar, *sequence diagram* interaksi sistem, serta *class diagram* dengan enam kelas utama yaitu Kamar, Admin, BiayaPemeliharaan, Penghuni, Pembayaran, dan Notifikasi. Perencanaan enam *sprint Agile* memberikan kerangka implementasi yang terstruktur dan adaptif terhadap perubahan kebutuhan.

Perancangan antarmuka (UI/UX) menghasilkan delapan halaman utama, meliputi *login*, *dashboard*, kelola kamar, kelola penghuni, kelola pembayaran, laporan keuangan, kelola biaya perbaikan, dan bayar tagihan. Hasil pengujian *black box testing* terhadap seluruh modul fungsional menunjukkan bahwa semua skenario uji menghasilkan keluaran yang valid sesuai harapan. Dengan demikian, aplikasi yang dirancang secara konseptual terbukti mampu mengotomasi proses administrasi kos dan berpotensi meningkatkan efisiensi pengelolaan penghuni serta keuangan secara signifikan bagi pemilik kos skala kecil-menengah.

Berdasarkan hasil dan keterbatasan penelitian ini, beberapa saran diajukan untuk

penelitian dan pengembangan selanjutnya. Penelitian selanjutnya disarankan untuk mengimplementasikan aplikasi secara penuh dan melakukan pengujian *usability* menggunakan instrumen seperti *System Usability Scale (SUS)* dengan melibatkan pemilik kos dan penghuni secara langsung, serta melengkapinya dengan pengujian performa dan keamanan data untuk memastikan sistem mampu beroperasi dengan baik di bawah beban pengguna yang besar. Dari sisi pengembangan fitur, aplikasi dapat diperluas dengan mengintegrasikan *payment gateway* untuk kemudahan transaksi digital, fitur marketplace pencarian kos, serta rekomendasi bisnis berbasis kecerdasan buatan. Cakupan penelitian juga dapat diperluas dengan melibatkan lebih banyak pemilik kos dari berbagai skala dan karakteristik agar kebutuhan sistem yang dirumuskan lebih representatif dan dapat digeneralisasikan secara lebih luas.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada pemilik kos di daerah Badung yang telah bersedia meluangkan waktu untuk diwawancarai dan memberikan informasi yang sangat berharga bagi penelitian ini. Ucapan terima kasih juga disampaikan kepada Fakultas Teknik Universitas Mahasaraswati Denpasar atas dukungan akademis yang diberikan.

DAFTAR PUSTAKA

- Alvivi, F. I. N. dan Voutama, A. 2024. 'Pengembangan Sistem Manajemen Kos Berbasis Web'. *Jurnal Sistem Informasi Universitas Suryadarma*.
- Andika, I., Steven, Steven Nevile, Satya, R. dan Farisi, A. 2024. 'Analisis Sistem Informasi Manajemen Proyek: Systematic Literature Review'. *Jurnal Teknik Informatika dan Sistem Informasi (JATISI)*, 11(1), hal. 220–230.
- Halim, A., Antoline, W., Arbecco, A., Gunawan, E. dan Wijaya, L. R. 2024. 'Rancang Bangun Aplikasi Penyedia Pekerjaan Lepas Trivial Menggunakan Metodologi Agile'. *Jurnal Ilmiah Matrik*, 26(1), hal. 48–58.
- Mahfuzhi, A., Hidayah, A., Rifqo, M. dan Justicia, A. 2025. 'Perancangan Sistem Informasi Kelurahan Berbasis Website dengan Metode Object Oriented Analysis and Design (OOAD) (Studi Kasus Kelurahan Masmambang, Kabupaten Seluma)'. *Jurnal Media Infotama*, 21(2), hal. 716–725. <https://doi.org/10.37676/jmi.v21i2.9308>

- Nasrull, A. N. dan Sewaka. 2022. 'Perancangan Sistem Informasi E-Sports di Indonesia Berbasis Website Menggunakan Metode OOAD (Object Oriented Analysis Design)'. *OKTAL: Jurnal Ilmu Komputer dan Sains*, 1(5), hal. 498–505.
- Nova, S. H., Widodo, A. P. dan Warsito, B. 2022. 'Analisis Metode Agile pada Pengembangan Sistem Informasi Berbasis Website: Systematic Literature Review'. *Techno.Com*, 21(1), hal. 139–148. <https://doi.org/10.33633/tc.v21i1.5659>
- Prasetyo, K. W. dan Dewantara, R. G. 2022. 'Developing A Homestay Lodging Management Information System at Palmyra Homestay Malang'. *J-INTECH: Journal of Information and Technology*, 10(2). <https://doi.org/10.32664/j-intech.v10i2.798>
- Rachmatika, P. A., Ain, R. N., Wahyudinarti, E. dan Fitri, A. S. 2025. 'Penerapan Metode Object Oriented Analysis and Design pada Aplikasi Sistem Informasi'. *Jurnal Informatika dan Teknik Elektro Terapan*, 13(1), hal. 1076–1083. <https://doi.org/10.23960/jitet.v13i1.5829>
- Ramadhan, Z. 2023. 'Mengoptimalkan Pengembangan Aplikasi Mobile Melalui Perbandingan Metode Pengembangan Perangkat Lunak (Waterfall, Prototype, Mobile-D, Agile, RAD)'. *SUBMIT Jurnal Ilmiah Teknologi Informasi dan Sains*, 3(2), hal. 13–19. <https://doi.org/10.36815/submit.v3i2.2993>
- Romdona, S., Junista, S. S. dan Gunawan, A. 2025. 'Teknik Pengumpulan Data: Observasi, Wawancara dan Kuesioner'. *JISOSEPOL: Jurnal Ilmu Sosial Ekonomi dan Politik*, 3(1), hal. 39–47. <https://doi.org/10.61787/taceee75>
- Setiadi, H. N. dan Ulum, M. B. 2024. 'Rancang Bangun Sistem Informasi Penyewaan Kamar Kost Berbasis Website (Studi Kasus CV. Busur Rukun Mandiri)'. *Indonesian Journal of Education and Computer Science*, 2(2), hal. 63–76. <https://doi.org/10.60076/indotech.v2i2.653>
- Setiawan, R., Supriatna, A. D. dan Kusuma, A. H. 2021. 'Rancang Bangun Sistem Informasi Pengelolaan Rumah Kos Deo Garut Berbasis Web'. *Jurnal Algoritma*, 17(2), hal. 368–377. <https://doi.org/10.33364/algoritma/v.17-2.368>
- Supriatna, E. dan Rima, R. 2022. 'Perancangan Sistem Informasi Peralatan Menggunakan Object Oriented Analysis and Design Model Rational Unified Process'. *INTERNAL (Information System Journal)*, 5(2), hal. 157–165. <https://doi.org/10.32627/internal.v5i2.633>
- Zulfania, N. 2024. 'Analisis Perbandingan Metodologi untuk Manajemen Proyek: Agile dan Waterfall'. *ResearchGate*. <https://doi.org/10.13140/RG.2.2.22828.27528>