

Implementasi Aplikasi Monitoring Kos Pondok Ina 2

Implementation of Monitoring Management Application at Pondok Ina 2

Akbar Zaki Rhomadon^{*1}, Rony Setiawan²

^{*1}Informatika Desain, Manajemen Informatika (Kampus Kota Bandung), Politeknik LP3I

²Sistem Informasi, Manajemen Informatika (Kampus Kota Bandung), STMIK LIKMI

*e-mail: akbarzakirhomadon@email.com, rony@plb.ac.id

Received:	Revised:	Accepted:	Available online:
19.02.2025	16.03.2025	11.04.2025	29.04.2025

Abstrak. Pengelolaan Kos Pondok Ina 2 sebelumnya dilakukan secara manual, yang mengakibatkan keterlambatan pencatatan pembayaran dan kesalahan dalam manajemen kamar. Untuk mengatasi hal tersebut, dikembangkan aplikasi berbasis Android dengan memanfaatkan Firebase sebagai basis data dan Kotlin sebagai bahasa pemrograman. Aplikasi ini memungkinkan pengelola untuk memantau status kamar secara interaktif melalui denah digital serta mencatat pembayaran penghuni secara realtime. Metode pengembangan menggunakan pendekatan Waterfall, dimulai dari analisis kebutuhan, perancangan sistem, pengembangan aplikasi, pengujian, hingga evaluasi. Implementasi aplikasi ini terbukti meningkatkan efisiensi pengelolaan kamar dan pembayaran secara signifikan.

Kata kunci: Aplikasi, Kosan, Manajemen, Android, Firebase

Abstract. The management of Pondok Ina 2 boarding house was previously conducted manually, leading to delays in payment records and room management errors. To address these issues, an Android-based application was developed utilizing Firebase as the database and Kotlin as the programming language. This application enables managers to monitor room statuses interactively through a digital layout and record tenant payments in real-time. The development method employed a Waterfall approach, starting from requirement analysis, system design, application development, testing, and evaluation. The implementation of this application has significantly improved the efficiency of room and payment management.

Keywords: Application, Boarding House, Management, Android, Firebase

1. PENDAHULUAN

Kemajuan teknologi yang pesat telah membawa banyak kemudahan, salah satunya dalam pengelolaan kos-kos. Aplikasi manajemen kos menjadi solusi yang efektif untuk membantu pengelola mengatur data kamar, status penghuni, dan pembayaran dengan lebih sistematis. Sebelumnya, pengelola masih menggunakan pencatatan manual yang sering kali menyebabkan keterlambatan dan kesalahan dalam mencatat status pembayaran penghuni, sehingga berpengaruh terhadap keuangan dan operasional kos.

Masalah utama yang dihadapi pengelola kos antara lain adalah ketidakjelasan status pembayaran penghuni, kesulitan dalam memantau status kamar, serta kesalahan pencatatan data penghuni dan kamar yang masih dilakukan secara manual. Rumusan masalah dalam penelitian ini mencakup bagaimana sistem pengelolaan kos Pondok Ina 2 berjalan saat ini, kendala yang dihadapi pengelola dalam mengelola data kamar dan pembayaran, serta solusi terbaik yang dapat diterapkan untuk meningkatkan efisiensi pengelolaan kos.

Kegunaan penelitian ini adalah untuk meningkatkan kemampuan analisis dan pengembangan sistem informasi bagi penulis, menyediakan referensi baru bagi pembaca dalam pengelolaan kos berbasis teknologi, dan membantu pengelola kos Pondok Ina 2 dalam mempermudah proses pengelolaan kamar dan pembayaran. Selain itu, penelitian ini diharapkan dapat meminimalkan kerugian akibat kesalahan pencatatan dan memantau status kamar secara transparan bagi pemilik kos.

2. TINJAUAN PUSTAKA

2.1. Aplikasi

Penggunaan sistem informasi berbasis aplikasi telah terbukti meningkatkan efisiensi pengelolaan data dalam berbagai bidang, termasuk dalam pengelolaan kos-kos. Menurut Laudon dan Laudon (2020), implementasi sistem informasi dapat mengoptimalkan proses pencatatan, monitoring, dan pengelolaan sumber daya dengan lebih sistematis dan real-time. Dalam konteks pengelolaan kos, aplikasi manajemen memungkinkan pemilik atau pengelola untuk dengan mudah memantau status kamar, mencatat transaksi pembayaran, serta mengurangi potensi kesalahan yang sering terjadi pada sistem pencatatan manual.

2.2. Figma

Menurut Ridho Nastainullah (2020) Figma adalah salah satu design tool berbasis cloud gratis yang bisa dijalankan di browser (web based) atau aplikasi desktop di OS Windows dan MAC OS yang mirip dengan Sketch atau Adobe XD untuk fungsionalitas dan fiturnya. Figma adalah tool yang biasa digunakan untuk membuat desain tampilan aplikasi mobile, website, desktop dan lainnya. Secara umum, Figma banyak digunakan oleh orang-orang untuk membuat desain UI/UX,

2.3. Android Studio

Android Studio adalah lingkungan pengembangan terintegrasi (Integrated Development Environment/IDE) resmi yang dikembangkan oleh Google bersama JetBrains, dirancang khusus untuk membangun aplikasi berbasis Android. Android Studio menggantikan Eclipse ADT sebagai IDE utama sejak versi stabil pertamanya dirilis pada Desember 2014.

IDE ini dilengkapi berbagai fitur penting seperti editor kode cerdas, alat debugging interaktif, dan Android Virtual Device (AVD) Manager yang memungkinkan pengujian aplikasi pada berbagai perangkat virtual tanpa perlu perangkat fisik. Fitur-fitur tersebut membantu meningkatkan efisiensi dan efektivitas dalam proses pengembangan aplikasi Android.

2.4. Kotlin

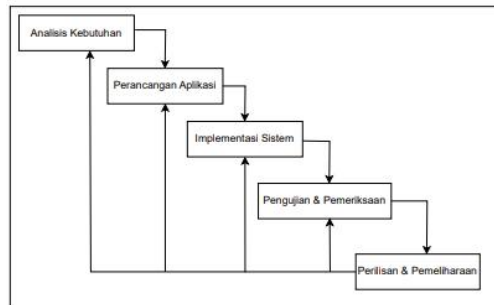
Kotlin adalah sistem pemrograman modern yang dirancang untuk pengembangan aplikasi Android. Menurut Sibarani, Munawar, & Wisnuadhi Kotlin merupakan Bahasa pemrograman berbasis Java Virtual Machine (JVM) untuk sistem operasi Android yang memadukan pemrograman object oriented dan pemrograman fungsional. Kotlin menawarkan sintaks yang ringkas, keamanan terhadap null-pointer exception, interoperabilitas penuh dengan Java, serta produktivitas pengembangan yang lebih tinggi. Dengan keunggulan tersebut,

2.5. Firebase

Firebase adalah sebuah platform layanan cloud yang menyediakan berbagai fitur backend seperti database realtime, autentikasi pengguna, dan penyimpanan data yang memudahkan pengembang aplikasi dalam membangun dan mengelola aplikasi tanpa harus mengurus server secara langsung. Menurut B. M. Ahmad dalam bukunya *Mobile App Development Essentials* (2022), Firebase memungkinkan sinkronisasi data secara instan antara pengguna dan server, sehingga sangat cocok digunakan untuk aplikasi yang membutuhkan pembaruan data secara realtime dan efisien.

3. METODE PENELITIAN

3.1 Pengembangan Sistem



Gambar 1. Model Waterfall Aplikasi

Metodologi Aplikasi Monitoring Kosan Pondok Ina 2 menggunakan model Waterfall sebagai panduan utama. Menurut Trisianto (2018: 12) menjelaskan bahwa Metode Waterfall adalah: “Metode Waterfall adalah suatu proses pengembangan perangkat lunak berurutan, di mana kemajuan dipandang sebagai terus mengalir ke bawah (seperti air terjun) melewati fase-fase perencanaan, pemodelan, implementasi (konstruksi), dan pengujian. Model ini dipilih karena memiliki alur kerja yang sederhana dan sistem, sehingga setiap tahap dapat diselesaikan secara mudah dan lebih terkendali untuk mengurangi kemungkinan terjadinya kesalahan.

Proses pengembangan dilakukan melalui beberapa tahapan, yaitu analisis kebutuhan, perancangan sistem, perancangan desain & implementasi aplikasi Berikut tahapannya:

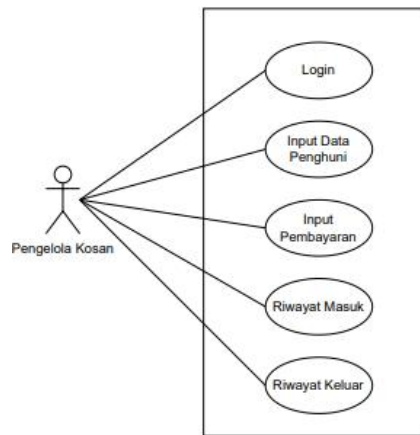
3.1.1 Analisis Kebutuhan

Peneliti melakukan analisis kebutuhan sistem dengan cara melakukan wawancara kepada pengelola Kosan Pondok Ina 2. Peneliti mengumpulkan informasi terkait keluhan, kebutuhan fitur, dan harapan pengelola terhadap aplikasi yang akan dikembangkan, seperti kebutuhan untuk pencatatan data penghuni, monitoring pembayaran, manajemen kamar, dan laporan riwayat penghuni. Wawancara digunakan karena merupakan teknik yang umum dalam *requirement elicitation* untuk menggali kebutuhan pengguna secara langsung, yang memungkinkan pengembang memahami proses bisnis dan ekspektasi pengguna dengan lebih akurat (Sommerville, 2011).

3.1.2 Perancangan Aplikasi

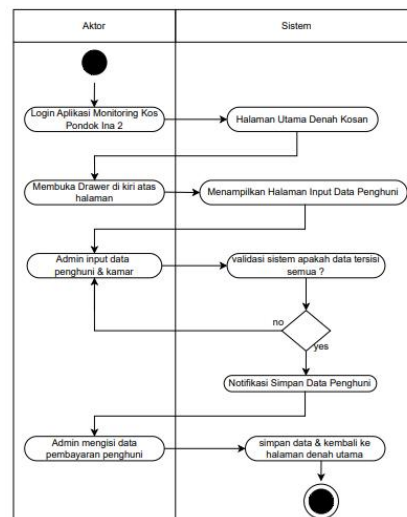
Peneliti melakukan perancangan aplikasi dengan menentukan teknologi pendukung dan fitur-fitur yang diperlukan berdasarkan hasil analisis kebutuhan. Proses ini mencakup perancangan alur navigasi aplikasi, struktur database, serta antarmuka pengguna (user interface) yang bertujuan memastikan sistem sesuai dengan kebutuhan pengguna akhir. Dalam pengembangan perangkat lunak, perancangan sistem merupakan tahap penting yang menjembatani kebutuhan fungsional dengan solusi teknis. Menurut Pressman (2014), perancangan perangkat lunak melibatkan transformasi kebutuhan menjadi representasi yang dapat diimplementasikan melalui struktur data, arsitektur sistem, dan elemen antarmuka.

3.1.2.1 Use Case Diagram Aplikasi

**Diagram 1.** Use Case Diagram Aplikasi

Peneliti menggunakan *Use Case Diagram* untuk memvisualisasikan interaksi antara pengguna dengan sistem aplikasi monitoring Kos Pondok Ina 2. Diagram ini membantu menggambarkan fungsi utama seperti login, pengelolaan data kamar dan penghuni, serta pencatatan pembayaran. Visualisasi ini digunakan agar proses pengembangan sistem lebih terarah dan sesuai kebutuhan pengguna. Menurut Satzinger et al. (2012), *Use Case Diagram* berperan penting dalam mengidentifikasi kebutuhan fungsional dan menjelaskan bagaimana aktor berinteraksi dengan sistem dalam konteks tertentu.

3.1.2.2 Activity Diagram Input Data Penghuni

**Gambar 2.** Activity Diagram Aplikasi

Peneliti membuat Activity Diagram sebagai salah satu alat bantu visual untuk menggambarkan alur aktivitas utama yang dilakukan oleh admin dalam menggunakan aplikasi monitoring Kos Pondok Ina 2. Activity Diagram ini dirancang untuk memetakan secara rinci

langkah-langkah proses yang harus dilalui oleh admin mulai dari saat membuka aplikasi, seperti dijelaskan oleh Dennis, Wixom, dan Tegarden (2015), penggunaan Activity Diagram sangat efektif dalam merepresentasikan alur kerja sistem secara visual, sehingga diagram ini tidak hanya memudahkan dalam analisis kebutuhan, tetapi juga sangat membantu pada tahap desain perangkat lunak.

3.1.2.3 Perancangan Database

Dua tabel utama dalam aplikasi ini adalah tabel data penghuni dan kamar, yang disimpan menggunakan Firebase Realtime Database. Tabel penghuni mencakup data seperti nama, NIK, tanggal masuk, status kamar, serta posisi koordinat pada denah digital, dan disimpan otomatis saat admin mengisi formulir. Sementara itu, tabel kamar mencatat ID kamar, lantai, nomor, tipe, dan status ketersediaan, yang terhubung secara langsung dengan data penghuni untuk pemantauan real-time. Struktur data ini mendukung integrasi antar fitur dalam aplikasi dan menjadi fondasi utama dalam proses monitoring kamar. Menurut Murach (2017), perancangan basis data yang baik memungkinkan pengelolaan data yang konsisten, efisien, dan mudah diakses dalam aplikasi real-time seperti sistem berbasis cloud.

Tabel 3. 1 Input Data Penghuni di Firebase

No	Nama Field	Tipe Data	Contoh Data	Keterangan
1	<i>jadikanSebagai</i>	String	Terisi	Status Kamar Penghuni
2	<i>nama</i>	String	Putri Sekar	Nama Penghuni
3	<i>nik</i>	String	217111500809008	Nomor Induk Kependudukan
4	<i>tanggalMasuk</i>	Date	01-02-2025	Tanggal Masuk Penghuni
5	<i>x</i>	Integer	100	Koordinat "Overlay Persegi" data
6	<i>y</i>	Integer	200	Koordinat "Overlay Persegi" data

Tabel 3. 2 Data Kamar di Firebase

No	Nama Field	Tipe Data	Contoh Data	Keterangan
1	<i>Id</i>	String	0l0CNN1CtGATYg5Jnyc	ID kamar di firebase
2	<i>Lantai</i>	String	Lantai 1	Letak Lantai Kamar
3	<i>noKamar</i>	String	101	No Kamar
4	<i>status</i>	String	Terisi	Status Kamar

3.1.3 Implementasi Sistem

Peneliti melakukan implementasi sistem dengan menerjemahkan hasil perancangan ke dalam bentuk aplikasi nyata. Peneliti membangun aplikasi menggunakan bahasa pemrograman yang sesuai, mengembangkan setiap fitur utama secara bertahap, serta mengintegrasikan seluruh komponen agar aplikasi dapat berjalan sesuai dengan fungsinya (Fairley, 2009; Larman, 2004).

3.1.4 Pengujian & Pemeriksaan

Peneliti melakukan pengujian terhadap aplikasi yang telah diimplementasikan untuk memastikan seluruh fitur berjalan sesuai dengan kebutuhan. Peneliti melakukan pemeriksaan fungsi, keamanan, dan kenyamanan penggunaan aplikasi, serta memperbaiki kesalahan atau bug yang ditemukan selama proses pengujian (Myers, Sandler, & Badgett, 2011; Kaner, Falk, & Nguyen, 1999).

3.1.5 Perilisan & Pemeliharaan

Peneliti melakukan implementasi sistem dengan menerjemahkan hasil perancangan ke dalam bentuk aplikasi nyata. Peneliti membangun aplikasi menggunakan bahasa pemrograman yang sesuai, mengembangkan setiap fitur utama secara bertahap, serta mengintegrasikan seluruh komponen agar aplikasi dapat berjalan sesuai dengan fungsinya (McConnell, 2004).

3.2 Pengumpulan Data

3.2.1 Observasi

Observasi dilakukan selama satu minggu terhadap proses pengelolaan data kos secara manual, mulai dari pencatatan penghuni baru, pembayaran, hingga pencatatan riwayat keluarmasuk. Tujuan dari observasi ini adalah untuk mengetahui kelemahan sistem konvensional serta mendefinisikan kebutuhan sistem yang akan dibangun.

3.2.2 Wawancara

Wawancara dilakukan kepada pemilik kos dan satu orang pengelola harian. Pertanyaan yang diajukan berfokus pada kesulitan dalam pengelolaan kos, kebutuhan terhadap fitur aplikasi, serta harapan terhadap sistem digital yang akan diterapkan.

3.2.3 Dokumentasi

Metode dokumentasi dilakukan dengan mengumpulkan berbagai data penunjang seperti format pencatatan manual sebelumnya, struk pembayaran, serta dokumentasi visual selama proses pembangunan dan pengujian sistem aplikasi.

4. HASIL DAN PEMBAHASAN

Hasil dari model Waterfall memudahkan peneliti dalam menyusun tahapan pengembangan secara sistematis dan terstruktur. Metode Waterfall merupakan pendekatan pengembangan perangkat lunak yang dilakukan secara berurutan, di mana setiap tahapan harus diselesaikan sebelum melanjutkan ke tahap berikutnya. Dengan mengikuti proses ini, peneliti dapat memastikan pengembangan aplikasi berjalan dengan terencana dan terkontrol. Berikut adalah tahapan penelitian yang diterapkan:

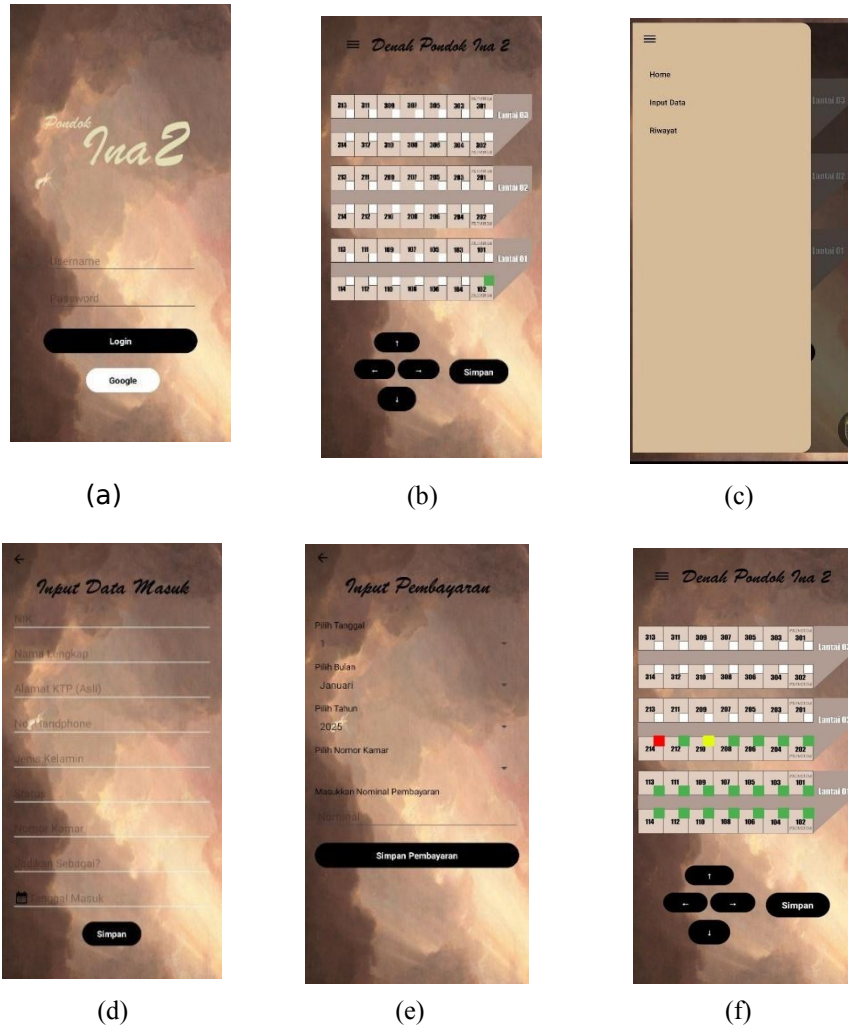
4.1 Analisis Kebutuhan

Peneliti menerima informasi dari pengelola dan pemilik Kos Pondok Ina 2 bahwa mereka membutuhkan suatu sistem untuk mempermudah pengelolaan dan monitoring keseluruhan kamar di Kos Pondok Ina 2. Adapun fitur-fitur yang akan diimplementasikan dalam aplikasi monitoring Kos Pondok Ina 2 adalah sebagai berikut:

1. Halaman utama denah monitoring kos
2. Input kamar kos
3. Input data penghuni
4. List kamar kos
5. List penghuni kos
6. Fitur satu arah mulai dari pemilihan kamar ke data penghuni dan ke pembayaran
7. Riwayat masuk penghuni
8. Riwayat keluar penghuni

4.2 Sistem & Antarmuka

4.2.1 Input data penghuni



Gambar 2. Urutan Tampilan Input Data Penghuni : (a) Halaman Login (b) Halaman Denah Utama (c) Drawer Pilihan (d) Halaman Input Data Penghuni (e) Halaman Pembayaran (f) Kembali ke Halaman Utama dan Pop up Detail Penghuni

Dalam alur penggunaannya, setelah pengguna berhasil login ke dalam sistem, mereka akan diarahkan ke halaman utama dan dapat mengakses menu "Input Penghuni". Urutan halaman bekerja secara logis, dimulai dari form pengisian, validasi data, hingga penyimpanan ke dalam basis data. Validasi input seperti pengecekan kelengkapan field diterapkan untuk mencegah data kosong masuk ke sistem. Desain antarmuka halaman input penghuni dibuat sederhana, responsif, dan mudah digunakan. Pengelola juga dapat langsung melihat daftar penghuni yang telah terinput dan memperbarui data bila diperlukan. Dengan demikian, sistem monitoring Kos Pondok Ina 2 menjadi lebih akurat, real-time, dan dapat diandalkan dalam mendukung operasional kos secara keseluruhan.

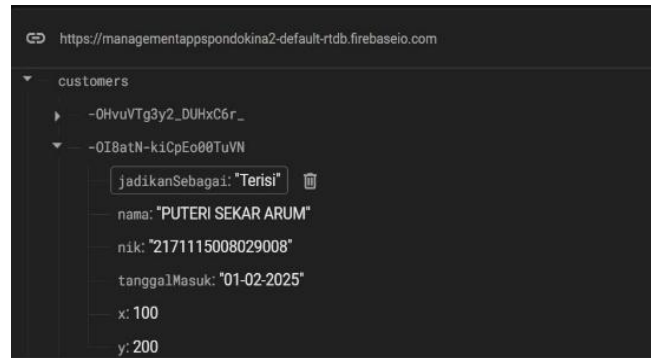
4.2.2 Riwayat Masuk & Keluar



Gambar 3. Tampilan Antarmuka Riwayat Masuk & Keluar

Dalam alur penggunaannya, setelah pengguna berhasil login ke dalam sistem, pengelola akan diarahkan ke halaman utama yang akan menampilkan denah kos secara keseluruhan. Dari halaman ini, pengguna dapat jugamengakses menu "Input Penghuni" melalui navigasi utama atau drawer menu yang tersedia. Urutan penggunaan aplikasi disusun secara logis dan berurutan, dimulai dari tahap pengisian formulir data penghuni, proses validasi data input, hingga penyimpanan otomatis ke dalam basis data Firebase secara real-time. Validasi input dilakukan secara ketat, termasuk pengecekan apakah semua field telah diisi dengan benar, seperti nama lengkap, NIK, jenis kelamin, dan nomor kamar. Langkah ini penting untuk menghindari kesalahan entri data yang berpotensi menimbulkan kekeliruan dalam pencatatan. Desain antarmuka halaman input penghuni dirancang dengan prinsip user interface (UI) yang intuitif dan user-friendly, memanfaatkan elemen visual seperti dropdown menu untuk pilihan data, tombol aksi yang jelas, serta struktur form yang tertata rapi agar mudah dipahami oleh pengguna non-teknis sekalipun. Tidak hanya meningkatkan efisiensi kerja pengelola, implementasi halaman input ini juga berperan dalam meminimalkan risiko kehilangan data, duplikasi informasi, atau ketidaksesuaian data yang kerap terjadi pada sistem manual berbasis kertas. Pengelola dapat langsung melihat daftar penghuni yang sudah terdaftar, memverifikasi detail informasi mereka, dan melakukan pembaruan data apabila diperlukan tanpa harus membuka dokumen terpisah. Selain itu, sistem penyimpanan terintegrasi dengan Firebase memastikan setiap perubahan data langsung tersimpan di cloud dan tersinkronisasi dengan perangkat lain, menjadikan sistem ini lebih responsif dan dapat diakses kapan saja. Dengan adanya mekanisme ini, proses pengelolaan penghuni kos menjadi lebih transparan, cepat, dan terstruktur, serta mendukung tujuan utama digitalisasi manajemen kos yang akurat, real-time, dan dapat diandalkan dalam operasional kos sehari-hari.

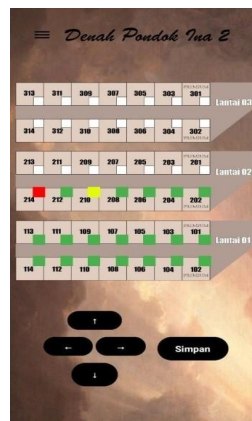
4.2.3 Database



Gambar 4. Database sinkron dengan Aplikasi

Sistem database menggunakan Firebase Realtime Database berhasil diintegrasikan dengan aplikasi monitoring Kos Pondok Ina 2. Data seperti penghuni, status kamar, dan pembayaran tersimpan secara real-time di cloud, memudahkan pengelola mengakses dan memperbarui tanpa sinkronisasi manual. Firebase juga otomatis memperbarui antarmuka aplikasi saat data berubah, meningkatkan efisiensi pencatatan harian. Selain itu, sistem ini menjamin keamanan dan keandalan data melalui backup dan autentikasi Firebase, sehingga pengelolaan kos menjadi lebih akurat dan cepat melalui perangkat Android (Google, 2021).

4.3 Pengujian & Pemeriksaan



Gambar 5. Denah Utama Setelah Penghuni Akad

Setelah aplikasi selesai dikembangkan, dilakukan uji coba secara langsung oleh pengelola kos Pondok Ina 2 untuk memastikan seluruh fitur berjalan sesuai kebutuhan. Pengujian dilakukan dengan skenario input data penghuni baru, pencatatan riwayat masuk dan keluar, serta penggunaan fitur monitoring kamar. Berdasarkan pengujian yang dilakukan selama satu minggu, tercatat sebanyak 25 data penghuni, 12 data riwayat masuk, dan 9 data riwayat keluar berhasil disimpan ke dalam sistem tanpa kendala berarti. Selain itu, pengguna menyampaikan bahwa proses pencatatan yang sebelumnya memerlukan waktu ± 15 menit kini dapat dilakukan hanya dalam waktu kurang dari 5 menit, menunjukkan adanya peningkatan efisiensi sebesar lebih dari

60%. Hasil pengujian ini menunjukkan bahwa aplikasi mampu membantu pengelolaan kos secara lebih cepat, akurat, dan terorganisir (Beizer, 1990)

4.3 Perilisan & Pemeliharaan



Gambar 6. Pop up Detail Penghuni Denah Utama

Setelah aplikasi diuji dan stabil, tahap berikutnya adalah perilisan kepada pengelola kos Pondok Ina 2 dalam bentuk file .apk yang dapat diinstal langsung di perangkat Android. Pengelola diberi panduan penggunaan fitur utama. Selanjutnya, dilakukan pemeliharaan untuk memastikan aplikasi tetap aman, berjalan baik, dan menyesuaikan kebutuhan pengguna. Hasil penggunaan menunjukkan peningkatan efisiensi pencatatan dan kemudahan pemantauan, sejalan dengan penelitian Pratama dan Nugroho (2020) serta Ramadhani et al. (2021) yang menegaskan manfaat digitalisasi dan aplikasi mobile dalam pengelolaan usaha kecil. Dengan tahap ini, aplikasi diharapkan terus berkembang sebagai solusi pengelolaan kos yang efisien dan praktis. Dengan adanya tahap perilisan dan pemeliharaan ini, diharapkan aplikasi monitoring Kos Pondok Ina 2 dapat terus berkembang dan menjadi solusi jangka panjang bagi pengelolaan kos yang efisien, praktis, dan berbasis teknologi.

5. KESIMPULAN

Berdasarkan hasil penelitian dan proses implementasi yang telah dilakukan, dapat disimpulkan bahwa pengembangan aplikasi monitoring Kos Pondok Ina 2 berbasis Android telah berhasil diterapkan dengan baik sebagai solusi digital dalam pengelolaan kos-kos. Aplikasi ini dirancang untuk menggantikan sistem pencatatan manual yang selama ini digunakan, yang rawan terhadap kesalahan, kehilangan data, serta menyulitkan proses pemantauan dan pengelolaan informasi penghuni kos.

Dari hasil pengujian, aplikasi dapat berjalan dengan stabil dan kompatibel di berbagai perangkat Android. Semua fungsi utama berjalan sesuai dengan yang diharapkan, mulai dari proses input data, pembaruan status kamar, hingga riwayat aktivitas penghuni. Selain itu, fitur login berbasis Firebase Authentication menambah tingkat keamanan dan kontrol terhadap akses aplikasi.

Implementasi aplikasi ini juga menunjukkan bahwa pemanfaatan teknologi digital dalam skala usaha kecil seperti kos-kos mampu meningkatkan efektivitas manajemen, mengurangi potensi kerugian akibat kesalahan pencatatan, dan menciptakan sistem informasi yang lebih terstruktur dan transparan.

Dengan demikian, aplikasi monitoring Kos Pondok Ina 2 tidak hanya menjadi alat bantu administrasi, tetapi juga sebagai bentuk inovasi dalam manajemen kos yang lebih modern dan adaptif terhadap perkembangan teknologi.

DAFTAR PUSTAKA

- [1] B. S. Bell and R. M. O’Keefe, "Visual programming in the real world," *Commun. ACM*, vol. 37, no. 6, pp. 76–85, Jun. 1994.
- [2] K. C. Laudon and J. P. Laudon, *Management Information Systems: Managing the Digital Firm*, 16th ed. Harlow, UK: Pearson, 2020.
- [3] B. M. Ahmad, *Mobile App Development Essentials*, 1st ed. Independently published, 2022.
- [4] R. Trisianto, *Rekayasa Perangkat Lunak*. Yogyakarta, Indonesia: Andi, 2018.
- [5] I. Sommerville, *Software Engineering*, 9th ed. Boston, MA, USA: Pearson Education, 2011.
- [6] R. S. Pressman and B. R. Maxim, *Software Engineering: A Practitioner’s Approach*, 8th ed. New York, NY, USA: McGraw-Hill, 2014.
- [7] J. W. Satzinger, R. B. Jackson, and S. D. Burd, *Systems Analysis and Design in a Changing World*, 6th ed. Boston, MA, USA: Cengage Learning, 2012.
- [8] A. Dennis, B. H. Wixom, and D. Tegarden, *Systems Analysis and Design*, 5th ed. Hoboken, NJ, USA: Wiley, 2015.
- [9] J. Murach, *Murach’s MySQL*, 2nd ed. Fresno, CA, USA: Mike Murach & Associates, 2017.
- [10] R. E. Fairley, *Software Engineering Concepts*, 3rd ed. New York, NY, USA: McGraw-Hill, 2009.
- [11] C. Larman, *Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development*, 3rd ed. Upper Saddle River, NJ, USA: Prentice Hall, 2004.
- [12] G. J. Myers, C. Sandler, and T. Badgett, *The Art of Software Testing*, 3rd ed. Hoboken, NJ, USA: Wiley, 2011.
- [13] C. Kaner, J. Falk, and H. Q. Nguyen, *Testing Computer Software*, 2nd ed. Hoboken, NJ, USA: Wiley, 1999.
- [14] N. S. Sibarani, G. Munawar, and B. Wisnuadhi, "Implementasi Firebase pada Aplikasi Android untuk Manajemen Data Real-Time," in *Prosiding Seminar Nasional Politeknik Negeri Bandung*, 2018. [Online]. Available: <https://jurnal.polban.ac.id/proceeding/article/view/1116>
- [15] R. Nastainullah, "Pengenalan Figma," 2020. [Online]. Available: <https://blogs.masterweb.com/panduan-figma/>
- [16] Android Developers, "Android Studio Overview," Android Developers. [Online]. Available: <https://developer.android.com/studio>
- [17] Google, "Firebase Realtime Database," Firebase Documentation, 2021. [Online]. Available: <https://firebase.google.com/docs/database>