

# Rancang Bangun Aplikasi Manajemen Kos Menggunakan Framework Flutter

**Daris Irfan Putrawanto <sup>\*1</sup>, Nur Cahyo Wibowo <sup>2</sup>, Reisa Permatasari <sup>3</sup>**

<sup>1,2,3</sup>Universitas Pembangunan Nasional “Veteran” Jawa Timur; Jl. Rungkut Madya No.1, Gn. Anyar, Kec. Gn. Anyar, Surabaya, Jawa Timur, (031) 8706369

<sup>1,2,3</sup>Jurusan Sistem Informasi, Fakultas Ilmu Komputer, UPN “Veteran” Jawa Timur, Surabaya  
e-mail: <sup>\*1</sup>20082010001@student.upnjatim.ac.id, <sup>2</sup>nurcahyo.si@upnjatim.ac.id,

<sup>3</sup>reisa.permatasari.sifo@upnjatim.ac.id

## Abstrak

*Indekos atau bahasa umumnya kos-kosan adalah sebuah bisnis yang menyediakan kamar-kamar untuk disewakan dengan jangka waktu tertentu sesuai dengan perjanjian antara penghuni kamar kos dan pemilik kos. Beberapa masalah timbul dari menghuni sebuah kos, seperti kurang mengenal antar penghuni, belum ada platform peminjaman fasilitas kos, pencatatan pembayaran kos, dan ajuan keluhan kos. Oleh karena itu, dibutuhkan sebuah aplikasi manajemen kos untuk dapat membantu pemilik dan pengelola kos dalam mengelola sebuah kos serta membantu penghuni kos dalam menghuni sebuah kos. Penelitian ini menggunakan metodologi waterfall. Dirancang dalam platform mobile menggunakan framework flutter dan bahasa pemrograman dart. Firebase dipilih sebagai database untuk kebutuhan penyimpanan data pada aplikasi ini. Black box test dipilih sebagai metode pengujian kepada pengguna langsung. Hasil yang didapatkan semua pengujian berhasil dilakukan. Dari pembuatan aplikasi manajemen kos ini diharapkan mampu menjawab kebutuhan pengguna yang berkaitan langsung dengan kos.*

**Kata kunci**—rancang bangun sistem, sistem informasi manajemen kos, flutter, firebase, aplikasi mobile

## Abstract

*Indekos or in general language kos-kosan is a business that provides rooms for rent for a certain period of time in accordance with the agreement between the boarding room occupant and the boarding house owner. Several problems arise from living in a boarding house, such as not getting to know each other, there is no platform for borrowing boarding facilities, recording boarding house payments, and submitting boarding house complaints. Therefore, a boarding house management application is needed to be able to help boarding house owners and managers in managing a boarding house and assist boarding house residents in occupying a boarding house. This research uses waterfall methodology. Designed on a mobile platform using the flutter framework and dart programming language. Firebase was chosen as the database for the data storage needs of this application. Black box test was chosen as a testing method for direct users. The results obtained were that all tests were carried out successfully. From the creation of this boarding house management application, it is hoped that it will be able to answer user needs that are directly related to boarding houses.*

**Keywords**— system design, boarding management information system, flutter, firebase, mobile applications

## 1. PENDAHULUAN

Perkembangan teknologi informasi membawa banyak perubahan pada aktivitas masyarakat. Perubahan aktivitas pada masyarakat dapat dilihat dari cara berkomunikasi, berinteraksi, hingga mendapatkan informasi dalam kehidupan sehari-hari. Segala inovasi

teknologi yang tercipta seiring perkembangan zaman dapat memudahkan seseorang dalam berkomunikasi, berinteraksi, serta bisa mendapatkan informasi yang tepat dan cepat. Perubahan aktivitas tidak terbatas pada masyarakat, melainkan perkembangan teknologi informasi juga menyebabkan perubahan pada sebuah organisasi dalam menjalankan bisnisnya [1].

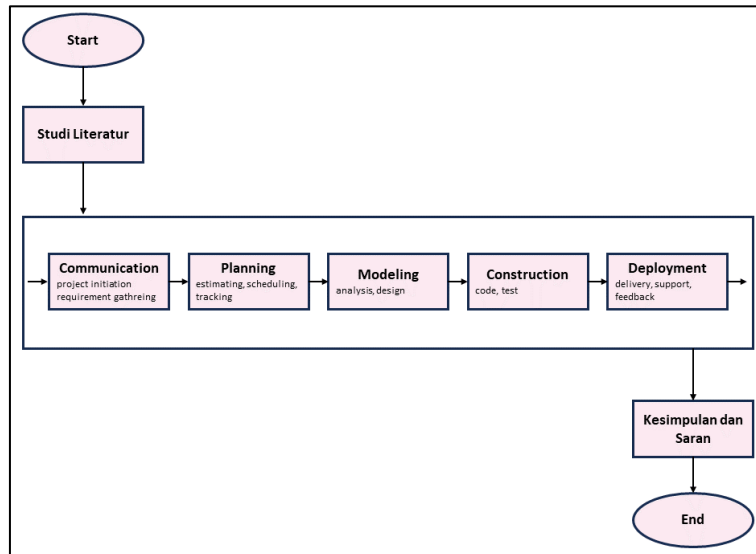
Indekos atau bahasa umumnya kos-kosan adalah sebuah bisnis yang menyediakan kamar-kamar untuk disewakan dalam rentan waktu tertentu yang sesuai dengan perjanjian yang ditentukan antara penghuni kamar kos dan pemilik kos [2]. Beberapa masalah timbul dari menghuni sebuah kos. Terkadang dalam satu hunian kos, ditempati oleh sekelompok orang yang sebelumnya telah mengenal satu sama lain. Tidak sedikit juga, orang yang belum tentu mengenal tetangga kosnya sendiri. Hal ini disebabkan kurang berinteraksinya orang tersebut dengan penghuni kos lain. Pada sebuah kos juga terdapat fasilitas yang dapat dipinjam dan digunakan bersama-sama oleh penghuni kos. Pembayaran sewa kos merupakan kewajiban seorang penghuni kos. Umumnya, pemilik kos akan secara rutin melakukan penagihan pembayaran kos kepada penyewa. Penagihan pembayaran biasanya dilakukan melalui WhatsApp atau media pengirim pesan singkat lainnya. Belum banyak kos yang menerapkan digitalisasi dalam melakukan pengelolaan kosnya.

Dari beberapa masalah yang diangkat diatas, maka dibutuhkan sebuah aplikasi manajemen kos. Aplikasi ini dirancang dan dibangun dalam basis android. Aplikasi manajemen kos ini tidak hanya dapat digunakan oleh satu kos saja, melainkan juga dapat digunakan oleh banyak pemilik kos dan penghuninya. Aplikasi ini menyediakan fitur yang dibutuhkan oleh pemilik, pengelola atau penghuni kos untuk memudahkan dalam mengelolanya. Fitur yang dapat ditemukan dalam aplikasi ini antara lain melihat daftar setiap penghuni kos, jatuh tempo pembayaran setiap penghuni kos, layanan peminjaman fasilitas, dan layanan pengelolaan kos. Aplikasi ini terbatas penggunaannya yaitu oleh pemilik kos, pengelola kos, dan penghuni kos.

Pada penelitian sebelumnya yang terkait dengan judul Pengembangan Aplikasi Mobile Manajemen Indekos berbasis Android dengan Metode *Human-Centered Design* (Studi Kasus: Kos Bu Parjo) [3], Aplikasi Aturkost Berbasis Web Untuk Pengelola Dan Penghuni Kost (Studi Kasus: Kost Jura) [4], Sistem Informasi Pembayaran Dan Pendataan Kost D'house Petukangan Jakarta Selatan [5], dan penelitian skripsi dengan judul Rancang Bangun Marketplace Sistem Informasi Kosku "Si Kosku" Di Wilayah Universitas Pembangunan Nasional "Veteran" Jawa Timur [6] terdapat pengembangan yang diterapkan pada aplikasi manajemen kos ini. Pengembangan yang dilakukan diharapkan dapat memudahkan pemilik dan pengelola kos dalam mengelola kos serta penghuni kos dalam beraktivitas selama menghuni sebuah kos. Aplikasi dilengkapi dengan data penghuni kos agar sesama penghuni dapat mengenal teman kos satu dengan yang lain, fitur peminjaman fasilitas kos agar mempermudah koordinasi dalam peminjaman fasilitas kos, layanan kos untuk mempermudah dalam pencatatan dan riwayat keluhan penghuni kos terhadap fasilitas yang ada, dan pencatatan keuangan untuk dapat mengetahui riwayat pembayaran kos setiap penghuni

## 2. METODE PENELITIAN

Penelitian ini dilakukan menggunakan metode waterfall. Metode *waterfall* adalah salah satu metode pengembangan sebuah perangkat lunak yang didalamnya menyediakan pendekatan siklus hidup (*lifecycle*) perangkat lunak dengan berurutan. Nama "*Waterfall*" digunakan karena metode ini melibatkan beberapa tahap yang harus dituntaskan satu per satu secara berangsur, menunggu dari tahap sebelumnya selesai sebelum melanjutkan lagi ke tahap berikutnya. *waterfall* adalah metodologi sekuensial dengan pendekatan sistematis yang meliputi tahapan analisis, desain, pengkodean, dan pengujian [7]. Gambar 1 berikut ini merupakan alur dalam metode penelitian yang digunakan selama penelitian dilakukan.



Gambar 1. Metode Penelitian Waterfall

### 3. HASIL DAN PEMBAHASAN

Hasil dan berisi tentang penjelasan dari tahapan yang telah dijelaskan sebelumnya pada metodologi penelitian serta penjelasan tentang kesuksesan dalam pengembangan aplikasi.

#### 3.1 *Studi Literatur*

Studi literatur pada penelitian ini dilakukan dengan cara mencari dan merangkum teori dari sejumlah buku, artikel, dan jurnal. Studi literatur juga didapatkan dari penelitian yang sebelumnya dilakukan sebagai pedoman awal penelitian ini.

#### 3.2 *Communication*

Tahap communication atau komunikasi merupakan tahapan awal dalam mengembangkan sebuah sistem informasi menggunakan metode *waterfall*. Wawancara dilaksanakan pada tiga kos berbeda guna mendapatkan data yang relevan. Ketiga kos tersebut antara lain Kosan Hub UPN, Kos Putra Pak Abror Klampis Aji dan Kos Putri Umik Yayuk UNTAG Surabaya.

Kosan Hub UPN adalah kos putra yang dimiliki oleh Bapak Wildan Al Akutsar yang terletak di Jalan Medokan Asri Barat MA1A Blok B-33, Medokan Ayu, Kecamatan Rungkut, Kota Surabaya. Kosan Hub UPN saat ini belum menerapkan digitalisasi. Pembayaran saat ini dilakukan dengan cara melakukan transfer langsung kepada pemilik kos dan data diarsipkan pada Excel. Saat ini belum ada pencatatan riwayat seseorang dalam penghuni kos. Saat terdapat keluhan dalam menghuni kos, akan dilaporkan langsung kepada pemilik kos melalui WhatsApp pribadi atau WhatsApp group kos. Fasilitas bersama digunakan bersama-sama tanpa adanya pemesanan terlebih dahulu.

Kos Putra Pak Abror Klampis Aji adalah kos putra yang dimiliki oleh Bapak Abror yang terletak di Jalan Klampis Aji Tengah 1 No. 5A, Klampis Asem, Kecamatan Sukolilo, Kota Surabaya. Kos Putra Pak Abror Klampis Aji saat ini belum menerapkan digitalisasi. Pembayaran saat ini dilakukan dengan cara melakukan transfer langsung kepada pemilik kos. Saat ini belum ada pencatatan riwayat seseorang dalam penghuni kos. Saat terdapat keluhan dalam menghuni kos, akan dilaporkan langsung kepada pemilik kos melalui WhatsApp.

Kos Putri Umik Yayuk UNTAG adalah kos putri yang dimiliki oleh ibu Yayuk yang terletak di Jalan Semolowaru Selatan 1 No. 20A, Semolowaru, Kecamatan Sukolilo, Kota Surabaya. Kos Putri Umik Yayuk UNTAG saat ini belum menerapkan digitalisasi. Pembayaran saat ini dilakukan dengan cara melakukan transfer langsung kepada pemilik kos. Saat ini belum ada pencatatan riwayat seseorang dalam penghuni kos. Saat terdapat keluhan dalam menghuni kos, akan dilaporkan langsung kepada pemilik kos melalui WhatsApp.

### 3.3 *Communication*

Tahapan planning berisi tentang perancangan awal dari aplikasi manajemen kos. Planning menghasilkan jadwal penelitian dan spesifikasi kebutuhan. Adapun spesifikasi kebutuhan *software* untuk melakukan pengembangan aplikasi manajemen kos sebagai berikut.

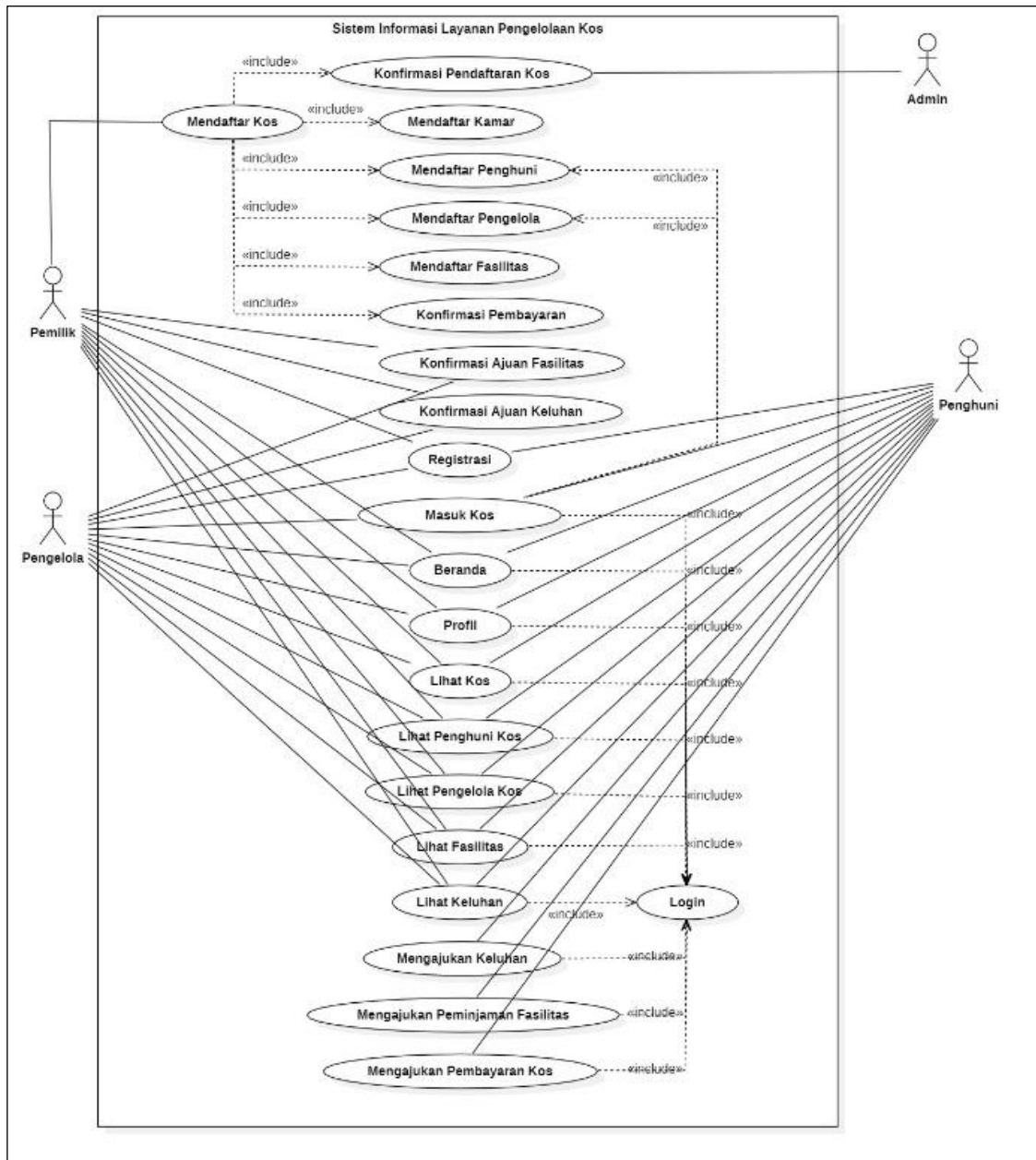
- a. Sistem Operasi (OS) Windows 11 Home 64-bit.
- b. Visual Studio Code v1.88.1
- c. Bahasa pemrograman Dart 3.0.6
- d. Framework Flutter channel stable 3.10.6
- e. Google Firebase Realtime Database 12.3.1
- f. Android Emulator Google Pixel 7 XL Android 11.0.

### 3.4 *Modeling*

Tahap *modeling* menjelaskan tentang rancangan dari model aplikasi manajemen kos. Tahapan dari modeling antara lain *use case diagram*, *activity diagram*, *sequence diagram*, dan desain antar muka. Berikut merupakan rincian *modeling* yang digunakan.

#### 3.4.1 *Use Case Diagram*

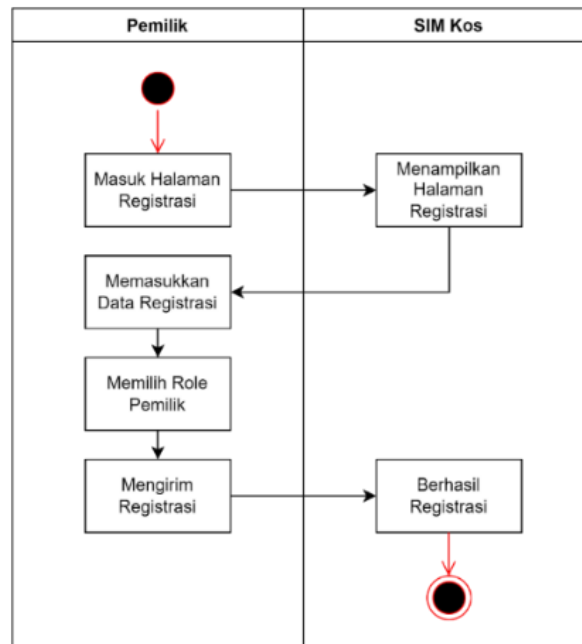
Gambar 2 merupakan *use case diagram* dari aplikasi manajemen kos. Terdapat 3 user yaitu pemilik, penghuni, dan pengelola. Gambar 2 berikut ini merupakan *use case diagram* yang digunakan dalam mengembangkan aplikasi manajemen kos.



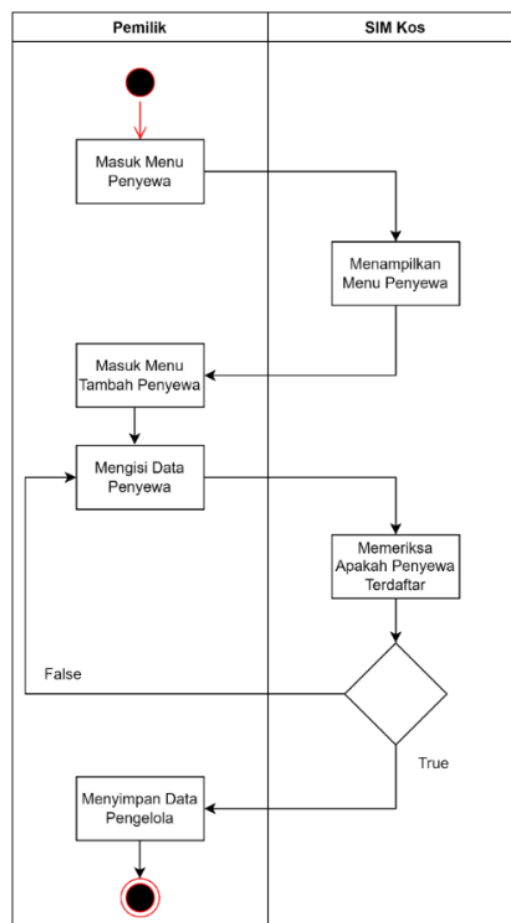
Gambar 2. Use Case Diagram

### 3.4.2 Activity Diagram

*Activity Diagram* merupakan gambaran dari sebuah proses bisnis atau sistem informasi yang ditampilkan dalam bentuk aliran kerja (*workflow*). *Activity diagram* yang digambarkan meliputi *activity diagram* untuk pemilik, penghuni, dan pengelola. Gambar 3 dan 4 berikut ini merupakan *activity diagram* yang digunakan dalam mengembangkan aplikasi manajemen kos.



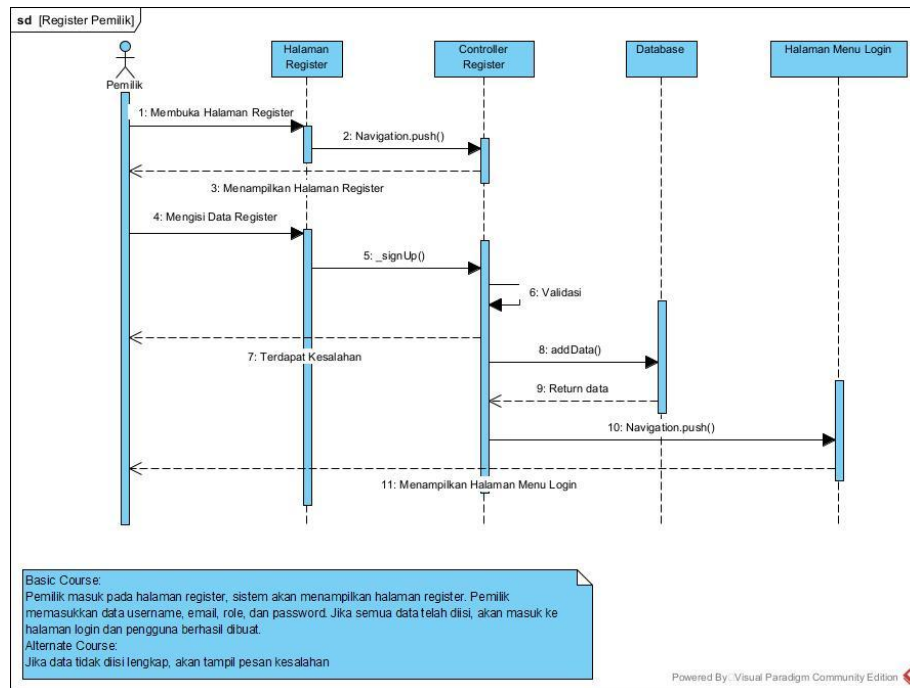
Gambar 3. Activity Diagram Register Pemilik



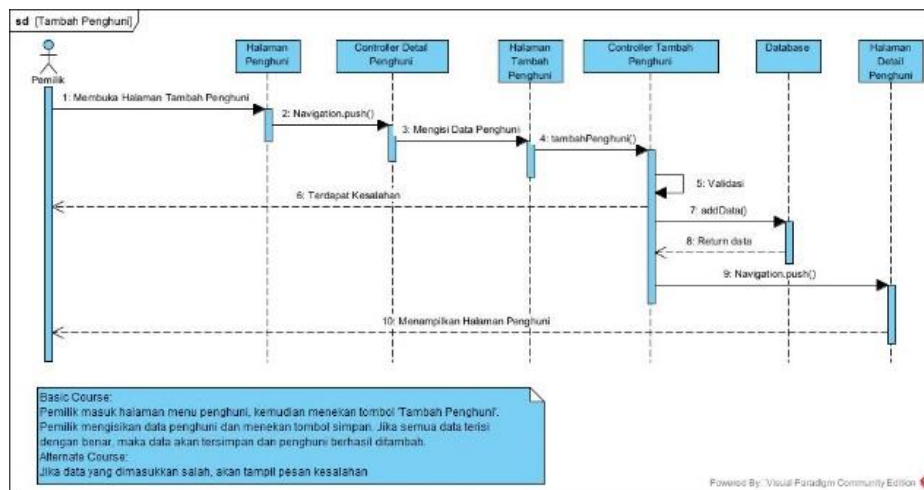
Gambar 4. Activity Diagram Tambah Penyewa

### 3.4.3 Sequence Diagram

*Sequence diagram* tersebut menggambarkan function dari setiap *class* dalam fitur yang ada dalam aplikasi manajemen kos. Gambar 5 dan 6 berikut ini merupakan *sequence diagram* yang digunakan dalam mengembangkan aplikasi manajemen kos.



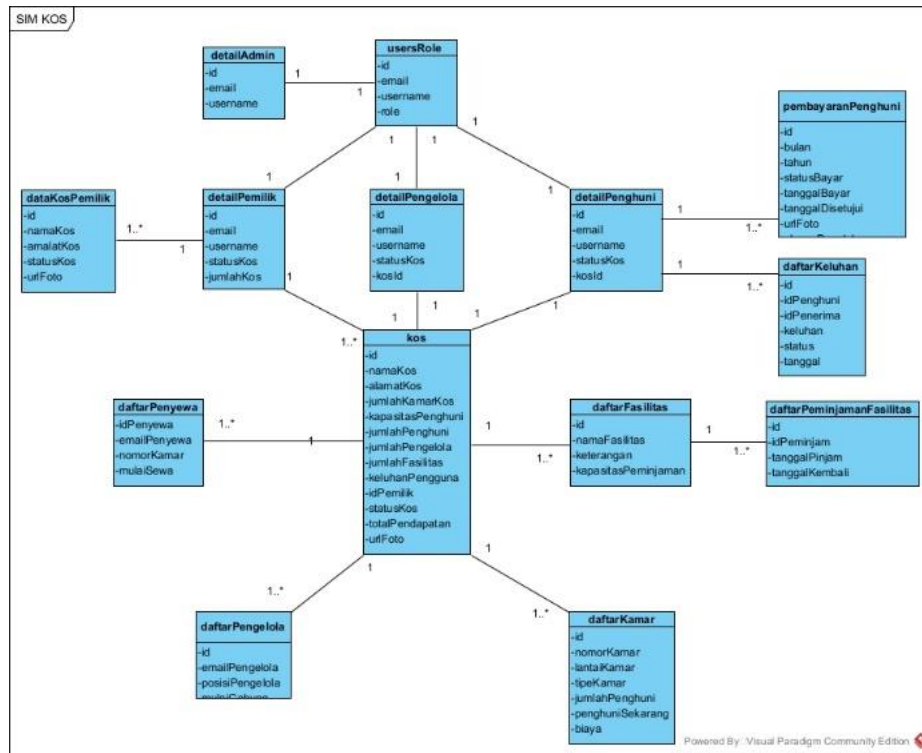
Gambar 5. Sequence Diagram Register Pemilik



Gambar 6. Sequence Diagram Tambah Penyewa

### 3.4.4 Class Diagram

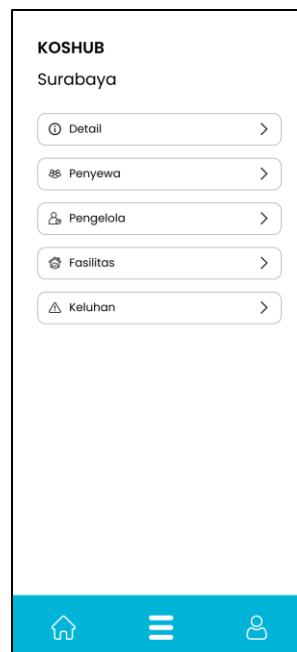
*Class diagram* adalah model hubungan satu kelas dengan kelas lainnya dalam database. *Class diagram* menjelaskan *field* yang dimiliki dan *constraint* yang menghubungkannya. Gambar 7 berikut ini merupakan *class diagram* yang digunakan dalam mengembangkan aplikasi manajemen kos.



Gambar 7. Class Diagram

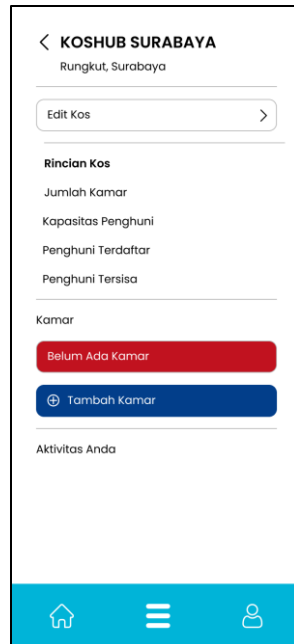
### 3.4.5 Desain Antar Muka

Pada setiap pembuatan sistem informasi terdapat proses perancangan desain antar muka yang memiliki tujuan untuk memudahkan pengguna saat menggunakan aplikasi manajemen kos. Gambar 8 dan 9 berikut ini merupakan desain antar muka yang digunakan dalam mengembangkan aplikasi manajemen kos.



Gambar 8. Halaman Menu Kos





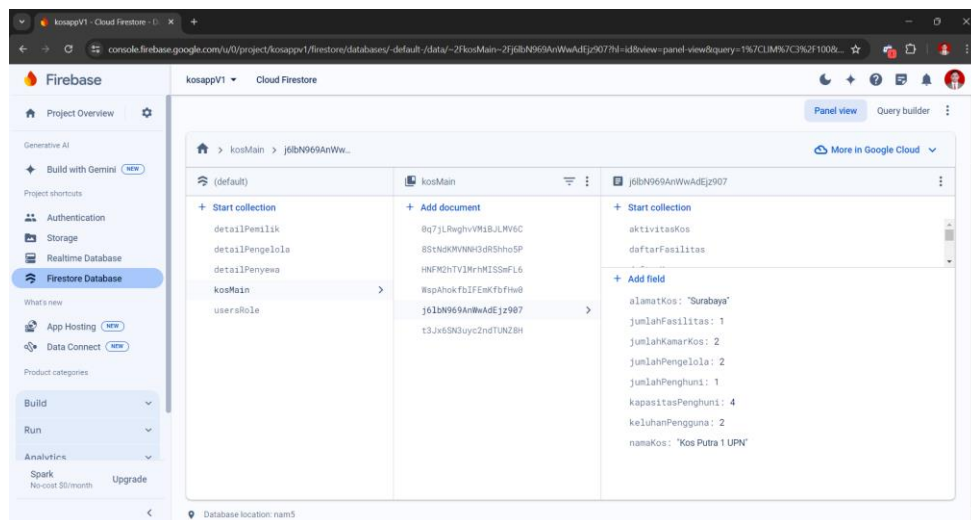
Gambar 9. Halaman Detail Kos

### 3.5 Construction

Tahap *construction* menjelaskan tentang membangun sistem dari desain yang dibuat sebelumnya ke proses untuk membuat aplikasi manajemen kos.

#### 3.5.1 Implementasi Database

Implementasi database adalah tahapan dari penerapan rancangan database yang dibuat sebelumnya dalam tahap *modeling*. Database aplikasi manajemen kos yaitu menggunakan Firebase. Firebase realtime database adalah salah satu basis data atau database yang disimpan pada layanan cloud dan mendukung penggunaan platform yang luas seperti platform website, android mobile, dan IOS mobile [8]. Gambar 10 merupakan implementasi dari Firebase Firestore yang digunakan untuk menyimpan data pengguna dari aplikasi manajemen kos.



Gambar 10. Firebase Firestore

### 3.5.2 Implementasi Sistem

Tahap implementasi sistem merupakan sebuah tahapan dimana akan dijelaskan cara kerja aplikasi manajemen kos dari kode yang telah dibangun. Aplikasi manajemen kos dikembangkan menggunakan bahasa pemrograman Dart dan Framework Flutter. Flutter merupakan *Software Development Kit* (SDK) yang dapat digunakan dalam melakukan pengembangan aplikasi dengan basis mobile yang memiliki performa yang tinggi dan dibuat oleh google [9]. Berikut merupakan rincian dari implementasi sistem aplikasi manajemen kos.

```
void _signUp() async {
  final formState = _formKey.currentState;
  if (formState != null && formState.validate()) { ...
  } else {
    showToast(message: "Lengkapi Data Anda");
  }
}

Future<void> addPemilikDetail(String email, String username) { ...

Future<void> addPenghuniDetail(String email, String username) { ...

Future<void> addPengelolaDetail(String email, String username) { ...

Future<void> addUser(String email, String username, String role) { ...
```

Gambar 11. Controller Halaman Register

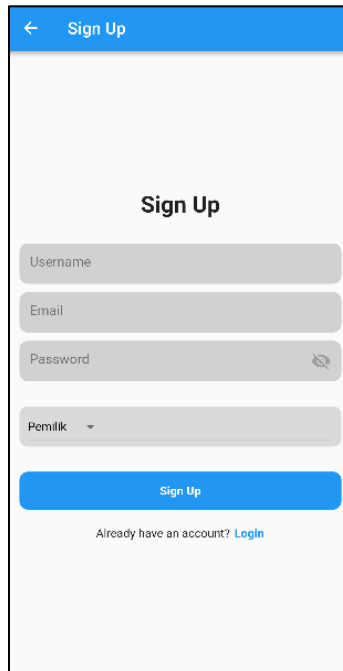
```
String username = _usernameController.text;
String email = _emailController.text;
String password = _passwordController.text;
String role = roleUser;

User? user = await _auth.signUpWithEmailAndPassword(email, password);
addUser(email, username, role);
if (role == "Penghuni") { ...
} else if (role == "Pengelola") { ...
} else { ...
}
```

Gambar 12. Model Halaman Register

```
Form(
  key: _formKey,
  child: Column(
    children: [
      FormContainerWidget( // FormContainerWidget ...
        SizedBox( // SizedBox ...
          FormContainerWidget( // FormContainerWidget ...
            SizedBox( // SizedBox ...
              FormContainerWidget( // FormContainerWidget ...
                SizedBox( // SizedBox ...
                  Container( // Container ...
                ), // Container ...
              ), // Container ...
            ), // Container ...
          ), // Container ...
        ), // Container ...
      ), // Column
    ), // Form
  ), // Form
  child: GestureDetector(
    onTap: () {
      _signUp();
    },
    child: Container( // Container ...
  ), // GestureDetector
```

Gambar 13. View Halaman Register



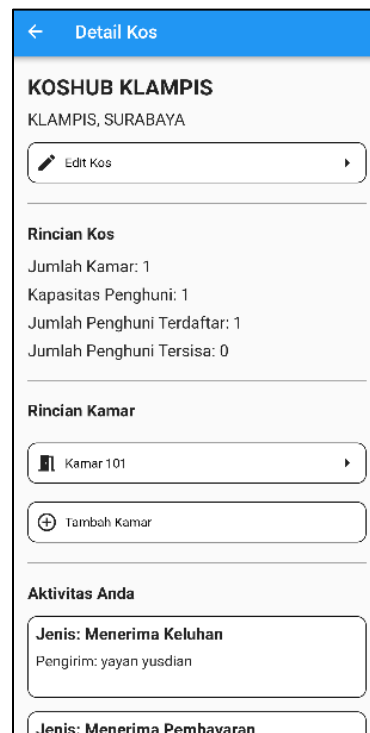
Gambar 14. Hasil Tampilan Halaman Register

```
FutureBuilder<DocumentSnapshot>(  
  future: FirebaseFirestore.instance.collection('koslala').doc(idKos).get(),  
  builder: (context, AsyncSnapshot<DocumentSnapshot> snapshot) {
```

Gambar 15. Controller Halaman Detail Kos

```
Text(  
  '${documentData["namaKos"]}'.toUpperCase(),  
  style: TextStyle(fontSize: 24, fontWeight: FontWeight.bold),  
) , // Text  
SizedBox( // SizedBox ---  
Text(  
  '${documentData["alamatKos"]}'.toUpperCase(),  
  style: TextStyle(fontSize: 18, fontWeight: FontWeight.normal),  
) , // Text  
SizedBox( // SizedBox ---  
GestureDetector(  
  onTap: () {  
    Navigator.push(  
      context,  
      MaterialPageRoute(builder: (context) => EditDetailKosPemilik(idKos: idKos)),  
    );  
  },  
  child: MenuContainerWidget(title: "Edit Kos", icon: Icons.edit),  
) , // GestureDetector  
Text(  
  "Rincian Kos",  
  style: TextStyle(fontSize: 18, fontWeight: FontWeight.bold),  
) , // Text  
SizedBox( // SizedBox ---  
Text(  
  'Jumlah Kamar: ${documentData["jumlahKamarKos"]}',  
  style: TextStyle(fontSize: 18, fontWeight: FontWeight.normal),  
) , // Text  
SizedBox( // SizedBox ---  
Text(  
  'Kapasitas Penghuni: ${documentData["kapasitasPenghuni"]}',  
  style: TextStyle(fontSize: 18, fontWeight: FontWeight.normal),  
) , // Text  
SizedBox( // SizedBox ---  
Text(  
  'Jumlah Penghuni Terdaftar: ${documentData["jumlahPenghuni"]}',  
  style: TextStyle(fontSize: 18, fontWeight: FontWeight.normal),  
) , // Text  
SizedBox( // SizedBox ---  
Text(  
  'Jumlah Penghuni Tersisa: ${documentData["kapasitasPenghuni"] - documentData["jumlahPenghuni"]}',  
  style: TextStyle(fontSize: 18, fontWeight: FontWeight.normal),  
) , // Text
```

Gambar 16. View Halaman Detail Kos



Gambar 17. Hasil Tampilan Halaman Detail Kos

### 3.5.3 Pengujian Sistem

Pengujian sistem adalah sebuah tahap di mana pengguna menguji sistem yang telah selesai dibuat. Tujuannya adalah untuk memastikan sistem yang dibuat tersebut sesuai dengan kebutuhan pengguna. Metode yang digunakan untuk menguji aplikasi manajemen kos ini adalah metode *black box testing*. *Black box testing* merupakan pengujian kualitas perangkat lunak yang berfokus pada fungsionalitas perangkat lunak [10]. Pengujian dilakukan kepada 3 kos yang masing-masing meliputi seorang pemilik, seorang penghuni, dan seorang pengelola. Tabel 1 hingga 3 berikut ini adalah rincian dari pengujian sistem yang dilakukan.

Tabel 1. Pengujian Sistem Pemilik

Skenario	Hasil	Status
Pemilik masuk menu Sign Up, mengisi data identitas, dan menekan tombol daftar	Menampilkan pesan user berhasil ditambahkan	Sukses pengguna 1, pengguna 2, pengguna 3
Pemilik masuk halaman Login, mengisi data identitas, dan menekan tombol Login	Berhasil login dan masuk beranda	
Pemilik masuk menu profil	Masuk halaman profil pemilik	
Pemilik masuk menu profil, menekan tombol edit profil, mengubah profil yang ada, dan menyimpan	Berhasil menyimpan profil baru	
Pemilik masuk menu kos	Masuk halaman detail kos yang terdaftar	
Pemilik menekan tombol 'Tambah Kos' pada menu kos, memasukkan data kos, dan menekan tombol simpan	Berhasil mendaftarkan kos	
Pemilik menekan tombol 'Tambah Kamar' pada detail kos, memasukkan data kamar, dan menekan	Berhasil membuat kamar	

tombol simpan		
Pemilik menekan tombol 'Hapus Kamar' pada menu rincian salah satu kamar	Berhasil menghapus data kamar	
Pemilik menekan tombol 'Tambah Penghuni' pada salah satu kamar terdaftar, memasukkan email penghuni, dan menekan tombol simpan	Berhasil menambahkan penghuni	
Pemilik menekan tombol 'Keluarkan Penghuni' pada salah satu penghuni terdaftar di menu detail penghuni	Berhasil mengeluarkan penghuni	
Pemilik memiliki salah satu riwayat pembayaran di menu detail penghuni dan melakukan konfirmasi dengan menekan tombol 'Setuju'	Berhasil menyetujui pembayaran penghuni	
Pemilik memiliki salah satu riwayat pembayaran di menu detail penghuni dan melakukan penolakan dengan menekan tombol 'Tolak'	Berhasil pembayaran penghuni dan penghuni berhak mengonfirmasi ulang pembayaran	
Pemilik menekan tombol 'Tambah Pengelola' pada menu detail pengelola, memasukkan email pengelola, dan menekan tombol simpan	Berhasil menambahkan pengelola	
Pemilik menekan tombol 'Keluarkan Pengelola' pada salah satu penghuni terdaftar di menu detail pengelola	Berhasil mengeluarkan pengelola	
Pemilik menekan tombol 'Tambah Fasilitas' pada detail kos, memasukkan data fasilitas, dan menekan tombol simpan	Berhasil membuat fasilitas	
Pemilik menekan tombol 'Hapus Fasilitas' pada menu rincian salah satu fasilitas	Berhasil menghapus data fasilitas	
Pemilik masuk menu detail fasilitas, menekan tombol fasilitas, mengubah data fasilitas yang ada, dan menyimpan data baru	Berhasil menyimpan data fasilitas baru	
Pemilik memiliki salah satuajuan peminjaman fasilitas pada menu detail fasilitas dan melakukan konfirmasi dengan menekan tombol 'Setuju' atau 'Tolak'	Berhasil menyetujui atau menolakajuan fasilitas	
Pemilik memiliki salah satuajuan keluhan pada menu detail keluhan dan melakukan konfirmasi dengan menekan tombol 'Setuju' atau 'Tolak'	Berhasil menyetujui atau menolak keluhan penghuni	

Tabel 2. Pengujian Sistem Penghuni

Skenario	Hasil	Status
Penghuni masuk menu Sign Up, mengisi data identitas, dan menekan tombol daftar	Menampilkan pesan user berhasil ditambahkan	Sukses pengguna 1, pengguna 2, pengguna 3
Penghuni masuk halaman Login, mengisi data identitas, dan menekan tombol Login	Berhasil login dan masuk beranda	
Penghuni masuk menu profil	Masuk halaman profil pengguna	
Penghuni masuk menu profil, menekan tombol edit profil, mengubah profil yang ada, dan menyimpan	Berhasil menyimpan profil baru	

Penghuni masuk menu kos	Masuk halaman detail kos	
Penghuni masuk menu pengelola	Masuk halaman daftar pengelola	
Penghuni masuk menu penghuni	Masuk halaman daftar penghuni	
Penghuni melakukan pembayaran kos pada halaman riwayat bayar	Masuk dalam menu riwayat bayar kos dan menampilkan pembayaran kos setiap bulan	
Penghuni masuk dalam salah satu menu pembayaran kos yang tersedia dan mengisikan bukti bayar kos.	Pembayaran kos berhasil diunggah	
Penghuni masuk dalam salah satu menu pembayaran kos yang ditolak pembayaran oleh pemilik dan mengisikan ulang bukti bayar kos.	Pembayaran kos berhasil diunggah ulang	
Penghuni masuk dalam menu fasilitas kos, memilih fasilitas yang tersedia, dan mengajukan peminjaman.	Berhasil melakukan peminjaman fasilitas	
Penghuni masuk dalam menu keluhan dan mengajukan keluhan.	Berhasil melakukan mengajukan keluhan	

Tabel 3. Pengujian Sistem Pengelola

Skenario	Hasil	Status
Pengelola masuk menu Sign Up, mengisi data identitas, dan menekan tombol daftar	Menampilkan pesan user berhasil ditambahkan	Sukses pengguna 1, pengguna 2, pengguna 3
Pengelola masuk halaman Login, mengisi data identitas, dan menekan tombol Login	Berhasil login dan masuk beranda	
Pengelola masuk menu profil	Masuk halaman profil pengelola	
Pengelola masuk menu profil, menekan tombol edit profil, mengubah profil yang ada, dan menyimpan	Berhasil menyimpan profil baru	
Pengelola masuk menu kos	Masuk halaman detail kos yang terdaftar	
Pengelola masuk menu pengelola	Masuk halaman daftar pengelola	
Pengelola masuk menu penghuni	Masuk halaman daftar penghuni	
Pengelola memiliki salah satu ajuan peminjaman fasilitas pada menu detail fasilitas dan melakukan konfirmasi dengan menekan tombol 'Setuju' atau 'Tolak'	Berhasil menyetujui atau menolak ajuan fasilitas	
Pengelola memiliki salah satu ajuan keluhan pada menu detail keluhan dan melakukan konfirmasi dengan menekan tombol 'Setuju' atau 'Tolak'	Berhasil menyetujui atau menolak keluhan penghuni	

#### 4. KESIMPULAN

Dari pembahasan penelitian ini, dapat disimpulkan bahwa aplikasi manajemen kos dapat membantu pemilik, penghuni, atau pengelola kos untuk dapat menunjang aktivitas selama menghuni kos. Model *waterfall* yang digunakan berhasil diterapkan dalam rancang bangun aplikasi manajemen kos. Penggunaan Firebase sebagai database sangat membantu dalam proses penyimpanan data dan pengolahan data secara realtime. Dari pengujian sistem, semua skenario yang dicoba telah berhasil diuji oleh pengguna menggunakan *black box testing*.

#### 5. SARAN

Saran sebagai pengembangan aplikasi manajemen kos untuk penelitian selanjutnya adalah mengembangkan fitur-fitur dalam manajemen kos, menyematkan pembayaran yang terintegrasi pada aplikasi untuk lebih memudahkan penghuni dalam melakukan pembayaran kos, mengembangkan untuk platform mobile IOS, dan melakukan *deployment* ke Google Play Store atau App Store.

#### UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada seluruh pihak yang telah memberi bimbingan, dukungan, dan berperan dalam penelitian ini, sehingga penelitian ini dapat diselesaikan dengan baik.

#### DAFTAR PUSTAKA

- [1] Z. Zulham, 2017, *Penerapan Teknologi Informasi Menentukan Keberhasilan Dunia Perusahaan Industri*, Jurnal Warta, Vol. 53.
- [2] M. Rahmawati and E. Harahap, 2021, *Analisis Keuntungan Usaha Kos-Kosan Menggunakan Program Linear Dengan Aplikasi Geogebra*, Jurnal Matematika, Vol. 20, No. 1, Hal 59–65.
- [3] A. P. Wicaksono, A. P. Kharisma, and R. S. Sianturi, 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, No. 8, Hal 3596–3605.
- [4] S. Steven and K. Christianto, 2021, *Aplikasi AturKost Berbasis Web Untuk Pengelola dan Penghuni Kost (Studi Kasus: Kost Jura)*, JBASE - Journal of Business and Audit Information Systems, Vol. 4, No. 2.
- [5] S. R. Sumiko, 2022, *Sistem Informasi Pembayaran dan Pendataan Kost D'house Petukangan Jakarta Selatan*, Jurnal Riset dan Aplikasi Mahasiswa Informatika (JRAMI), Vol. 3, No. 03.
- [6] C. C. Gildbrandsen, 2023, *Rancang Bangun Marketplace Sistem Informasi Kosku 'Si Kosku' di Wilayah Universitas Pembangunan Nasional 'Veteran' Jawa Timur*, Skripsi, Fakultas Ilmu Komputer, Universitas Pembangunan Nasional "Veteran" Jawa Timur, Surabaya.

- [7] S. Yuniar, E. D. Wahyuni, and R. Permatasari, 2023, *Rancang Bangun Sistem Informasi Pengerjaan Tugas Berbasis Teknik Pomodoro Menggunakan Metode Waterfall*, *Jutisi: Jurnal Ilmiah Teknik Informatika dan Sistem Informasi*, Vol. 12, No. 3.
- [8] E. A. W. Sanad, 2019, *Pemanfaatan Realtime Database di Platform Firebase pada Aplikasi E-Tourism Kabupaten Nabire*, *Jurnal Penelitian Enjiniring*, Vol. 22, No. 1, Hal 20–26.
- [9] Muslim, R. P. Sari, and S. Rahmayuda, 2022, *Implementasi Framework Flutter pada Sistem Informasi Perpustakaan Masjid*, *Coding: Jurnal Komputer dan Aplikasi*, Vol. 10, No. 1, Hal 46–59.
- [10] Y. D. Wijaya and M. W. Astuti, 2021, *Pengujian Blackbox Sistem Informasi Penilaian Kinerja Karyawan PT Inka (Persero) Berbasis Equivalence Partitions*, *Jurnal Digital Teknologi Informasi*, Vol. 4, No. 1, Hal 22.