

---

**Dokumen Pengembangan Perangkat Lunak  
Rekayasa Perangkat Lunak  
untuk  
Rancang Bangun Sistem Pendukung Keputusan  
Pemilihan Kos Daerah Surabaya dengan Metode  
Fuzzy TOPSIS Berbasis Mobile**

**Tiara Ratna Sari**

**Disiapkan oleh  
081211632014**



**S1 Sistem Informasi Universitas Airlangga  
Surabaya, Agustus 20**

## Revision History

Name	Date	Reason For Changes	Version



**Latar Belakang**

Kota Surabaya sebagai kota metropolitan kedua di Indonesia memiliki lebih dari 69 sarana Pendidikan Tinggi, baik berupa Universitas, Perguruan Tinggi, Sekolah Tinggi maupun Institut yang tersebar di seluruh daerah Surabaya. Setiap tahun, Pendidikan Tinggi menerima ribuan mahasiswa baru. Contohnya tahun 2015 Universitas Airlangga 5215 mahasiswa baru. Dari ribuan mahasiswa baru tersebut, hampir separuhnya bukan berasal dari kota Surabaya. Mahasiswa baru tersebut bisa berasal dari seluruh wilayah nusantara. Maka, selama menempuh studi di Pendidikan Tinggi di Surabaya, mereka harus tinggal di tempat orang. Bagi mereka yang memiliki keluarga atau kerabat dapat tinggal bersama kerabat tersebut, sedangkan bagi yang tidak memiliki kerabat mereka harus mencari tempat tinggal lain. Beberapa Pendidikan Tinggi menyediakan asrama di lingkungan kampus, tetapi karena jumlah mahasiswa yang dapat ditampung terbatas, sebagian besar sisanya harus mencari tempat tinggal lain berupa kos / kontrakan.

Mencari kos bukanlah hal yang mudah bagi mahasiswa baru. Karena mahasiswa baru berasal dari luar kota Surabaya dan baru pertama kali datang di Surabaya, dan tidak memiliki kenalan, mobilitas dan waktu yang mereka miliki untuk mencari kos sangat terbatas. Mahasiswa baru pun tidak memiliki informasi yang memadai mengenai lokasi dan spesifikasi kos-kosan di sekitar kampus. Walaupun pihak Badan Eksekutif Mahasiswa (Organisasi Mahasiswa) di kampus biasanya telah membuka stan informasi dan memberikan selebaran informasi kos-kosan, kebanyakan dari mahasiswa baru tidak tahu arah dan terpaksa harus mencari dari pintu ke pintu. Setelah menemukan beberapa pilihan kos, menentukan kos yang terbaik pun bukan perkara mudah karena setiap kos menawarkan fasilitas dan biaya yang berbeda-beda. Mahasiswa baru biasanya tidak mencari kos seorang diri, biasanya ditemani orangtua, saudara, atau teman. Seringkali masing-masing memiliki kriteria dan prioritas yang berbeda yang membuat pencarian kos semakin memakan waktu. Seluruh proses ini harus dilakukan secara manual karena belum ada sistem yang membantu seluruh proses pemilihan kos.

Sistem yang bisa membantu adalah Sistem Pendukung Keputusan (SPK). Konsep SPK dikemukakan pertama kali oleh Scott Morton pada tahun 1971 (Turban, 2005). Beliau mendefinisikan cikal bakal SPK tersebut sebagai “sistem berbasis komputer yang interaktif, yang membantu pengambilan keputusan dengan menggunakan data dan model untuk memecahkan persoalan-persoalan tak terstruktur”. SPK tidak dimaksudkan untuk

mengotomasikan pengambilan keputusan, tetapi memberikan perangkat interaktif yang memungkinkan pengambil keputusan dapat melakukan berbagai analisis dengan menggunakan model-model yang tersedia (Kadir, 2003).

Salah satu metode SPK adalah logika *Fuzzy* yang diperkenalkan pertama kali oleh Lotfi A. Zadeh. Pada logika *Fuzzy*, komponen utama yang sangat berpengaruh adalah fungsi keanggotaan. Fungsi keanggotaan merepresentasikan derajat kedekatan antara suatu obyek terhadap atribut tertentu. Menurut Tettamanzi (2001). Logika *Fuzzy* merupakan kerangka matematis yang digunakan untuk merepresentasikan ketidakpastian, ketidakjelasan, ketidaktepatan, kekurangan informasi, dan kebenaran parsial.

Di dalam SPK juga dikenal istilah *Multiple Criteria Decision Making* (MCDM) yang digunakan untuk menetapkan alternatif terbaik dari sejumlah alternatif berdasarkan beberapa kriteria tertentu. Berdasarkan tujuannya, MCDM dapat dibagi menjadi 2 model, yaitu *Multi Attribute Decision Making* (MADM) dan *Multi Objective Decision Making* (MODM). MADM digunakan untuk menyelesaikan masalah masalah dalam ruang diskret. Sedangkan MODM digunakan untuk menyelesaikan masalah-masalah dalam ruang kontinyu. Dalam kasus pemilihan kos akan dilakukan seleksi terhadap beberapa alternatif dalam jumlah terbatas maka digunakan MADM.

Salah satu metode dalam MADM adalah *Technique for Order Preference by Similarity to Ideal Solution* (TOPSIS). Sebagaimana dinyatakan oleh Hwang (1981) bahwa TOPSIS didasarkan pada konsep dimana alternatif terpilih yang terbaik tidak hanya memiliki jarak terpendek dari solusi ideal positif, namun juga memiliki jarak terpanjang dari solusi ideal negative. Menurut Chung-Hsing Yeh (2002) Konsep ini banyak digunakan pada beberapa model MADM untuk menyelesaikan masalah keputusan secara praktis karena konsepnya sederhana dan mudah dipahami, komputasinya efisien, dan memiliki kemampuan untuk mengukur kinerja relatif dari alternatif-alternatif keputusan dalam bentuk matematis sederhana.

Menurut Sri Kusumadewi (2006) apabila data tentang atribut suatu alternatif tidak disajikan dengan lengkap, mengandung ketidakpastian atau tidak konsistena, maka metode MADM biasa tidak dapat digunakan. Untuk mengatasi masalah tersebut beberapa penelitian telah dilakukan dengan menggunakan *Fuzzy MADM* yang terbukti memiliki kinerja yang sangat baik. *Fuzzy MADM* adalah suatu metode yang digunakan untuk mencari alternatif optimal dari sejumlah alternatif dengan kriteria tertentu. Untuk itulah dipilih metode *Fuzzy TOPSIS*, yang akan menggabungkan *fuzzy* dan TOPSIS. Penggunaan *fuzzy* dilakukan pada kriteria linguistik yang memiliki ketidak pastian, setelah itu dihitung menggunakan metode

TOPSIS untuk mengetahui alternatif terbaik. Keuntungan yang didapatkan dengan menggunakan metode Fuzzy TOPSIS, ketidakpastian dan ketidakjelasan dari prespektif subjektif dan pengalaman pengambil keputusan dapat efektif diwakili dan mencapai keputusan yang lebih efektif (Ertugrul, 2007).

SPK dapat dikembangkan dalam berbagai *platform*, salah satunya adalah *mobile*. Sistem operasi *smartphone* yang paling banyak digunakan adalah Android. Android, Inc. didirikan di Palo Alto, California, pada bulan Oktober 2003 oleh Andy Rubin (pendiri *Danger*), Rich Miner (pendiri *Wildfire Communications, Inc.*), Nick Sears (mantan VP *T-Mobile*), dan Chris White (kepala desain dan pengembangan antarmuka *WebTV*). Android adalah sebuah sistem operasi untuk perangkat *mobile* berbasis linux yang mencakup sistem operasi, *middleware* dan aplikasi. Android menyelesaikan *platform* terbuka bagi para pengembang untuk menciptakan aplikasi mereka (Safaat, 2012).

Menindak lanjuti dari masalah yang dihadapi mahasiswa baru dari luar kota, maka dalam penelitian ini dibangun sistem pendukung keputusan. Sistem tersebut dimaksudkan untuk mendukung keputusan dengan metode *Fuzzy TOPSIS* dengan *mobile* yang diharapkan mampu membantu mahasiswa baru luar kota untuk mencari tempat tinggal sementara / kos sesuai prioritas yang dimiliki masing-masing individu.

### **Ruang Lingkup**

Dari identifikasi masalah yang terpapar di atas diperoleh gambaran dimensi permasalahan yang begitu luas. Namun menyadari adanya keterbatasan waktu dan kemampuan, maka penulis memandang perlu memberi batasan masalah secara jelas dan terfokus. Selanjutnya masalah yang menjadi obyek penelitian dibatasi sebagai berikut:

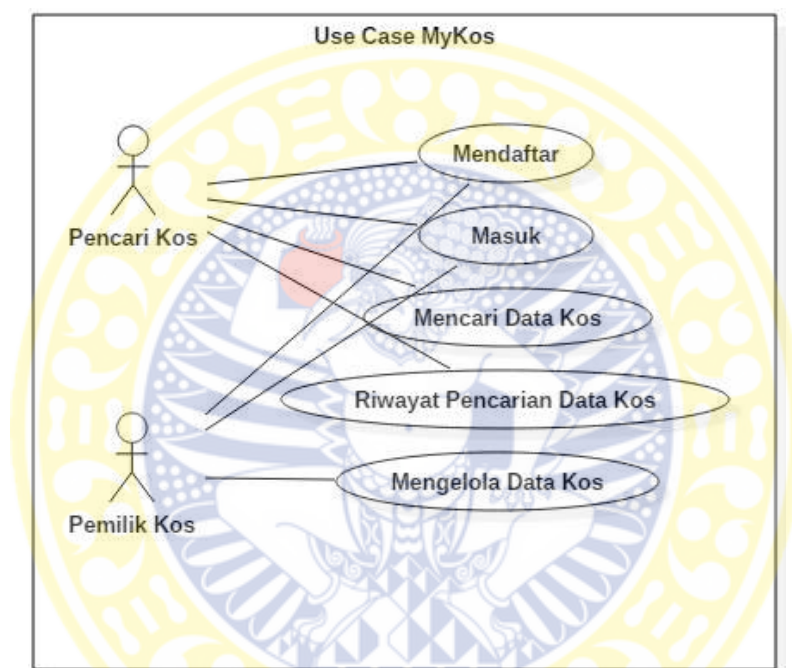
1. Data kos yang digunakan dalam penelitian ini adalah kos yang terletak di Kota Surabaya.
2. Data identitas dan fasilitas kos di Kota Surabaya yang dipakai dalam penelitian ini menggunakan sumber dari infokos.net.
3. Penyebaran Kuesioner kepada mahasiswa luar kota yang sedang belajar di Perguruan Tinggi Surabaya menggunakan *sample* 6 perguruan tinggi di Surabaya.



Produk yang kami buat ini bersifat mobile dan memerlukan jaringan internet untuk mengaksesnya. Kami ingin membuat tampilan yang menarik agar pengguna tidak bosan dengan tampilan kami.

## 2.1 Perancangan Sistem

Pemodelan hasil analisis proses perancangan sistem akan digambarkan dengan menggunakan *use case diagram* dan *activity diagram*. *Use case diagram* yang dimiliki sistem ini dapat dilihat pada Gambar 2.1.



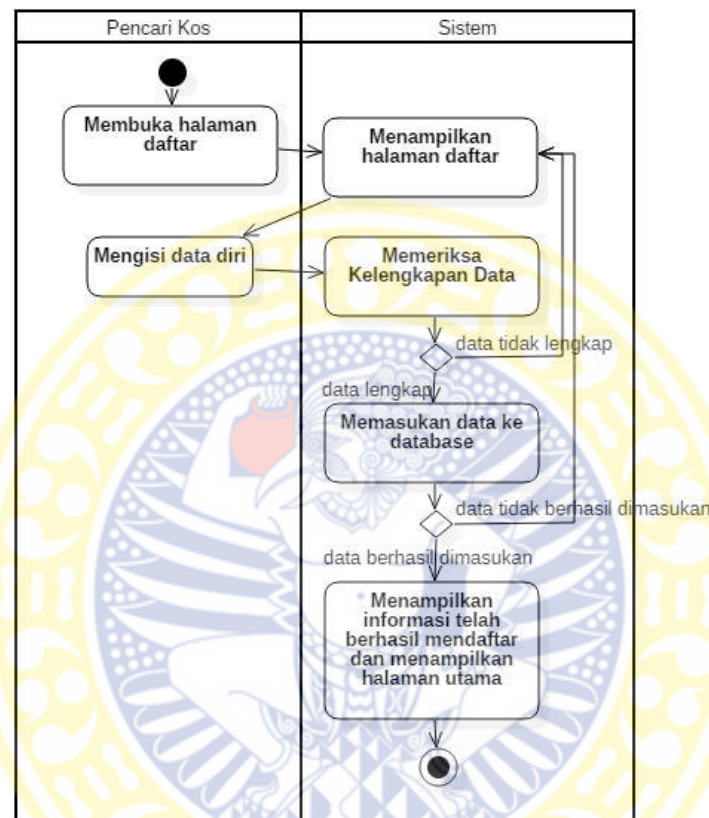
**Gambar 2.1** Use Case Diagram

Didalam sistem ini ada dua aktor yang memakai sistem, yaitu pencari kos dan pemilik kos. Fitur-fitur yang diberikan untuk pencari kos adalah mendaftar ke sistem, masuk ke sistem, mencari kos dengan metode FTOPSIS, dan riwayat pencarian kos. Fitur yang diberikan untuk pemilik kos adalah mendaftar ke sistem, masuk ke sistem, dan mengelola data kos yang dimiliki. Pada fitur mengelola data kos terdapat 3 kegiatan yaitu, dapat menambah data kos, mengubah data kos, dan menghapus data kos.

### 2.1.1 Activity Diagram Pencari Kos Daftar/Sign

Pada Aktivitas ini Pencari Kos mendaftarkan diri pada sistem agar data diri yang dimiliki dapat tersimpan untuk memudahkan pencarian kos sesuai kebutuhan. Ketika Pencari membuka halaman daftar pencari, maka sistem akan membuka halaman daftar pencari.

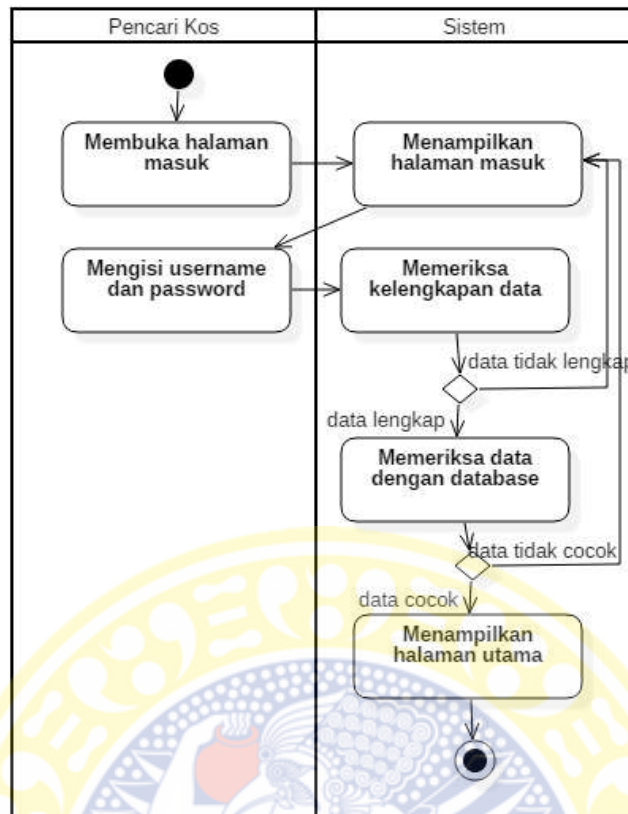
Pencari mengisi universitas, fakultas, program studi, nama, alamat, telp, jenis kelamin, *username*, *password*, dan *retype password* lalu menekan tombol *signup*. Sistem akan memeriksa kelengkapan data apabila telah lengkap data akan disimpan ke dalam *database* ketika data belum lengkap maka akan muncul peringatan. Setelah data berhasil masuk kedalam *database* sistem akan menampilkan informasi berhasil mendaftar dan membuka halaman utama pencari. *Activity Diagram* Pencari Kos Daftar/*Sign* dilihat pada Gambar 2.2.



**Gambar 2.2** *Activity Diagram* Pencari Kos Daftar/*Sign*

### 2.1.2 *Activity Diagram* Pencari Kos Masuk/*Login*

Pada Aktivitas ini Pencari Kos ingin masuk kepada sistem agar dapat menggunakan sistem. Ketika pencari membuka halaman login pencari, sistem akan menampilkan halaman login pencari. Pencari mengisi *username* dan *password* sesuai yang dimiliki. Sistem akan mencari data yang sesuai pada *database*. Ketika sistem menemukan data yang cocok sistem akan membuka halaman utama pencari, Ketika sistem tidak menemukan data yang cocok sistem tetap pada halaman masuk. *Activity Diagram* Pencari Kos Masuk/*Login* dapat dilihat Gambar 2.3.



**Gambar 2.3** Activity Diagram Pencari Kos Masuk/Login

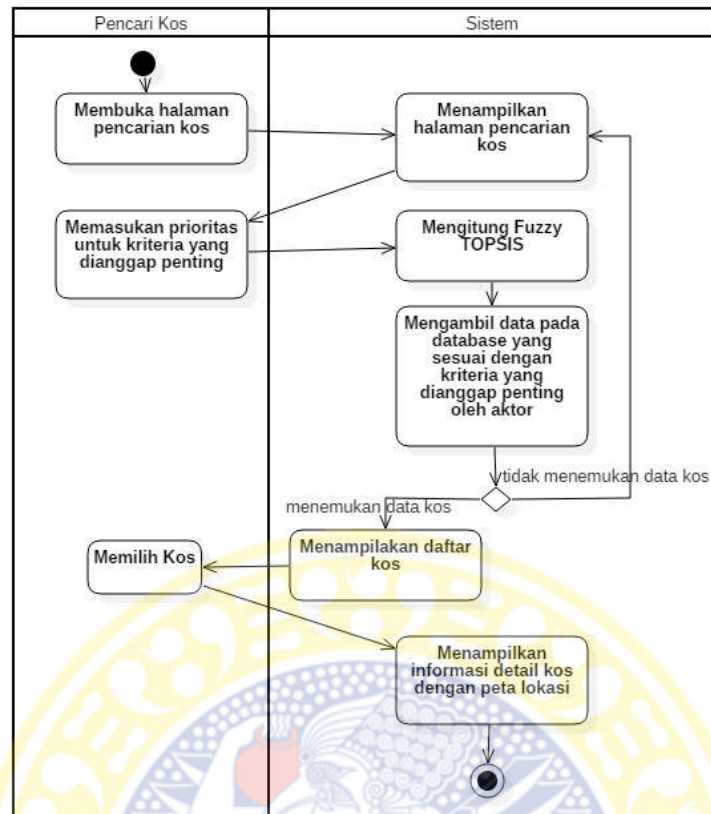
### 2.1.3 Activity Diagram Pencari Kos Mencari Data Kos

Pada Aktivitas ini Pencari Kos mencari data kos sesuai kriteria dan prioritas yang dimiliki pencari. Pencari dapat memilih sendiri kriteria dan berapa banyak jumlah kriteria yang diinginkan, serta dapat menentukan urutan prioritas yang diinginkan. Pencari membuka halaman pencarian data kos, sistem akan menampilkan halaman data kos. Pencari mengisi prioritas pada kriteria yang diinginkan dengan menekan tombol. Sistem mengambil data yang sesuai dengan kriteria pada *database*. Setelah mendapatkan data sistem akan menghitung metode TOPSIS dan menampilkan daftar kos sesuai perhitungan metode TOPSIS. Activity Diagram Pencari Kos Mencari Data Kos dapat dilihat pada Gambar 2.4.

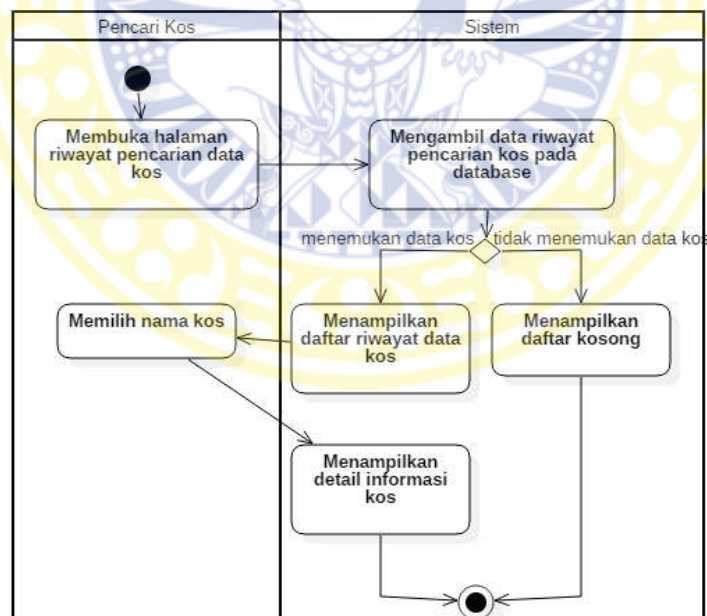
### 2.1.4 Activity Diagram Pencari Kos Riwayat Pencarian Data Kos

Pada Aktivitas ini Pencari Kos menyimpan data kos yang diinginkan pada fitur riwayat pencarian. Pencari kos dapat melihat kembali data kos yang telah disimpan. Pencari kos membuka halaman riwayat pencarian kos, sistem akan menampilkan halaman riwayat pencarian kos. Sistem mengambil data riwayat pada database dan menampilkan dalam bentuk daftar riwayat. Activity Diagram Pencari Kos Riwayat Pencarian Data Kos dilihat pada Gambar 2.5.





**Gambar 2.4** Activity Diagram Pencari Kos Mencari Data Kos

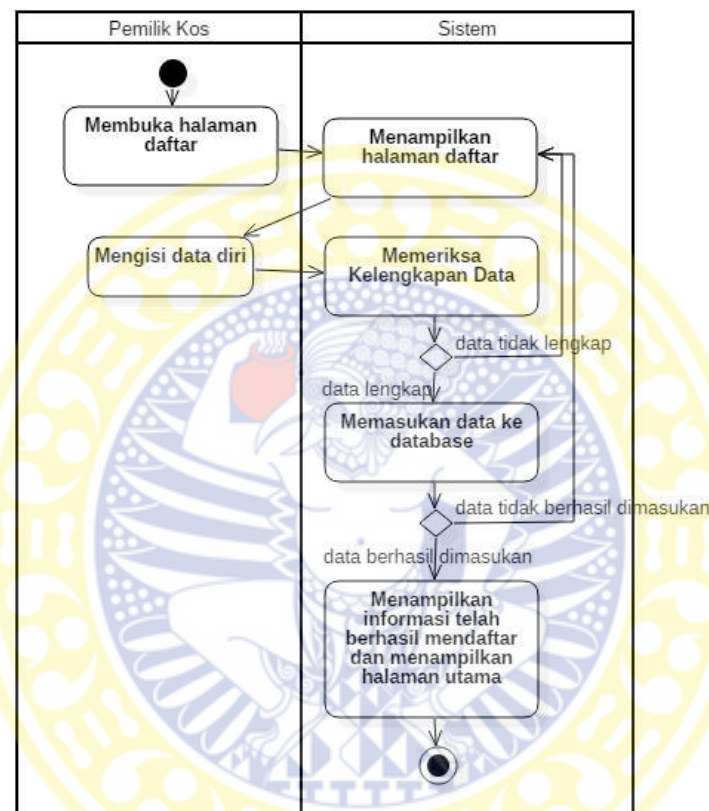


**Gambar 2.5** Activity Diagram Pencari Kos Riwayat Pencarian Data Kos

#### 2.1.5 Activity Diagram Pemilik Kos Daftar/Sign

Pada Aktifitas ini Pemilik Kos mendaftarkan diri pada sistem agar data diri yang dimiliki dapat tersimpan agar dapat mendaftarkan kos yang dimiliki pada sistem. Ketika

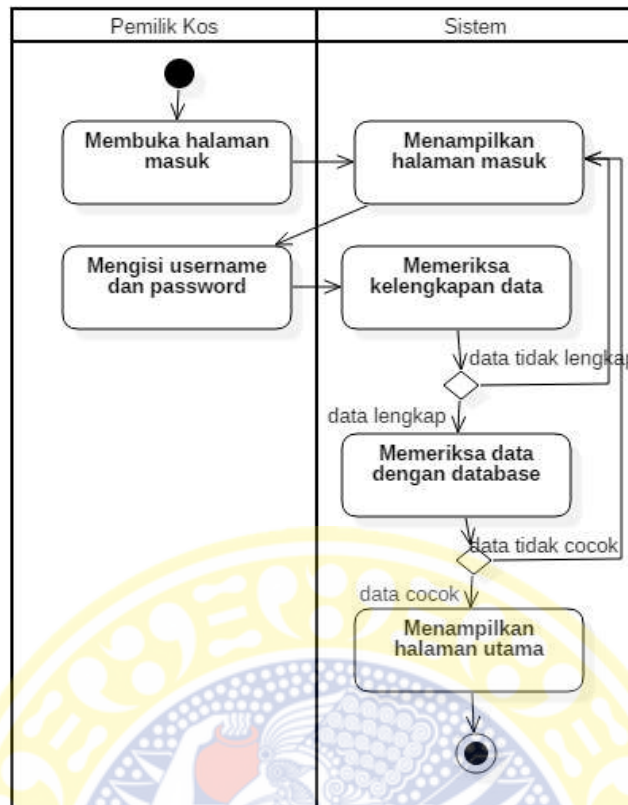
Pencari membuka halaman daftar pemilik, maka sistem akan membuka halaman daftar pemilik. Pemilik mengisi nama, alamat, telp, *username*, *password*, dan *retype password* lalu menekan tombol *signup*. Sistem akan memeriksa kelengkapan data apabila telah lengkap data akan disimpan ke dalam *database* ketika data belum lengkap maka akan muncul peringatan. Setelah data berhasil masuk kedalam *database* sistem akan menampilkan informasi berhasil mendaftar dan membuka halaman utama pemilik. *Activity Diagram* Pemilik Kos Daftar/*Sign up* dilihat pada Gambar 2.6.



**Gambar 2.6** Activity Diagram Pemilik Kos Daftar/*Sign up*

#### 2.1.6 Activity Diagram Pemilik Kos Masuk/Login

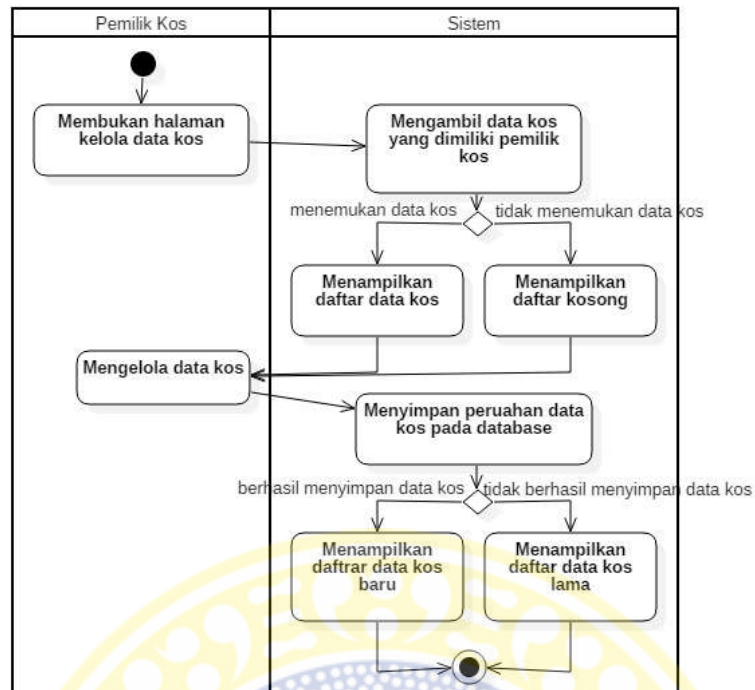
Pada Aktifitas ini Pemilik Kos masuk pada sistem agar dapat menggunakan sistem sesuai *username* dan *password* yang dimiliki. Ketika pemilik membuka halaman login pemilik, sistem akan menampilkan halaman login pemilik. Pemilik mengisi *username* dan *password* sesuai yang dimiliki. Sistem akan mencari data yang sesuai pada *database*. Ketika sistem menemukan data yang cocok sistem akan membuka halaman utama pemilik, Ketika sistem tidak menemukan data yang cocok sistem tetap pada halaman masuk. *Activity Diagram* Pemilik Kos Masuk/*Login* dapat dilihat Gambar 2.7.



**Gambar 2.7** Activity Diagram Pemilik Kos Masuk/Log in

### 2.1.7 Activity Diagram Pemilik Kos Mengelola Data Kos

Pada Aktivitas ini Pemilik Kos mengelola data kos, terdapat 3 aktivitas yang dapat dilakukan oleh pemilik kos. Aktivitas yang dapat dilakukan adalah dapat menambah data kos, mengubah data kos, dan juga menghapus data kos. Pemilik membuka halaman mengelola data kos, sistem akan membuka halaman mengelola data kos. Sistem akan mengambil data pada *database*, ketika sistem menemukan data kos maka ditampilkan daftar data kos tetapi ketika sistem tidak menemukan data kos sistem menampilkan daftar kosong. Pemilik dapat menekan tombol ➕ untuk menambah data kos, menekan tombol ➖ untuk menghapus data kos dan menekan tombol ✏️ untuk mengubah data kos. Sistem akan menyimpan perubahan pengelolaan data kos terbaru dan menampilkannya dalam bentuk daftar kos. Activity Diagram Pencari Kos Mencari Data Kos dilihat pada Gambar 2.8.



**Gambar 2.8** Activity Diagram Pemilik Kos Mengelola Data Kos

## 2.2 Perancangan Basis Data

Perancangan basis data akan digambarkan menggunakan *Conceptual Data Model* (CDM) dapat dilihat pada Gambar 2.9 dan *Physical Data Model* (PDM) dapat dilihat pada Gambar 2.10. Sistem ini akan mencatat beberapa data penting seperti data lokasi, data universitas pencari kos, data diri pemilik, data diri pencari, data kos dan fasilitas kos yang akan direpresentasikan kedalam 10 tabel data sebagai berikut:

### 1. Kecamatan

Perancangan basis data untuk tabel kecamatan merupakan representasi data kecamatan. Data kecamatan yang diperlukan adalah nama kecamatan seluruh Surabaya. Detail perancangan tabel kecamatan dapat dilihat pada Tabel 42.1.

**Tabel 2.1** Tabel Kecamatan

No.	Nama Atribut	Deskripsi	Tipe & Panjang	Keterangan
1.	id_kecamatan	Identitas kecamatan	Varchar(5)	<i>Primary Key</i>
2.	nama_kecamatan	Nama kecamatan	Varchar(20)	-

### 2. Kelurahan

Perancangan basis data untuk tabel kelurahan merupakan representasi data kelurahan. Data kecamatan yang diperlukan adalah nama kelurahan yang ada di Surabaya. Detail perancangan tabel kelurahan dapat dilihat pada Tabel 2.2.



**Tabel 2.2** Tabel Kelurahan

No.	Nama Atribut	Deskripsi	Tipe & Panjang	Keterangan
1.	id_kelurahan	Identitas kelurahan	Varchar(7)	<i>Primary Key</i>
2.	nama_kelurahan	Nama kelurahan	Varchar(20)	-
3.	id_kecamatan	Identitas kecamatan yang dimiliki kelurahan	Varchar(5)	<i>Foreign Key</i> dari Tabel Kecamatan

### 3. Universitas

Perancangan basis data untuk tabel universitas merupakan representasi untuk data universitas. Data universitas yang diperlukan adalah nama universitas yang ada di Surabaya. Detail perancangan tabel data universitas dapat dilihat pada Tabel 2.3.

**Tabel 2.3** Tabel Universitas

No.	Nama Atribut	Deskripsi	Tipe & Panjang	Keterangan
1.	id_universitas	Identitas universitas	Varchar(10)	<i>Primary Key</i>
2.	nama_universitas	Nama universitas	Varchar(50)	-

### 4. Fakultas

Perancangan basis data untuk tabel fakultas merupakan representasi untuk data fakultas yang dimiliki tiap universitas yang ada di Surabaya. Data fakultas yang diperlukan adalah nama fakultas. Detail perancangan tabel data kelurahan dapat dilihat pada Tabel 2.4.

**Tabel 2.4** Tabel Fakultas

No.	Nama Atribut	Deskripsi	Tipe & Panjang	Keterangan
1.	id_fakultas	Identitas fakultas	Varchar(10)	<i>Primary Key</i>
2.	nama_fakultas	Nama fakultas	Varchar(50)	-
3.	id_universitas	Identitas universitas yang dimiliki fakultas	Varchar(10)	<i>Foreign Key</i> dari Tabel Universitas

### 5. Program Studi

Perancangan basis data untuk tabel program studi merupakan representasi untuk data program studi yang dimiliki tiap universitas yang ada di Surabaya. Data program studi yang diperlukan adalah nama program studi dan letak program studi. Detail perancangan tabel data program studi pada Tabel 2.5.

### 6. Pemilik Kos

Perancangan basis data untuk tabel pemilik kos merupakan representasi untuk data pemilik kos yang mendaftar pada sistem. Data pemilik kos yang diperlukan adalah nama, telp, username, dan password pemilik. Detail perancangan tabel data pemilik kos dapat dilihat dapat dilihat pada Tabel 2.6.



**Tabel 2.5** Tabel Program Studi

No.	Nama Atribut	Deskripsi	Tipe & Panjang	Keterangan
1.	id_prodi	Identitas fakultas	Varchar(10)	<i>Primary Key</i>
2.	nama_prodi	Nama fakultas	Varchar(50)	-
3.	latitude	Garis lintang bumi yang menunjakan lokasi gedung program studi	Float(10,6)	-
4.	longitude	Garis bujur bumi yang menunjukan lokasi gedung program studi	Float(10,6)	-
5.	id_fakultas	Identitas fakultas yang dimiliki program studi	Varchar(10)	<i>Foreign Key</i> dari Tabel Fakultas

**Tabel 2.6** Tabel Pemilik Kos

No.	Nama Atribut	Deskripsi	Tipe & Panjang	Keterangan
1.	id_pemilik	Identitas pemilik kos	Varchar(20)	<i>Primary Key</i>
2.	nama_pemilik	Nama pemilik kos	Varchar(30)	-
3.	telp_pemilik	Telp pemilik kos	Varchar(12)	-
4.	username_pemilik	Username pemilik kos	Varchar(12)	-
5.	password_pemilik	Password pemilik kos	Varchar(20)	-

## 7. Pencari Kos

Perancangan basis data untuk tabel pencari kos merupakan representasi data pencari kos yang mendaftar pada sistem. Data pencari kos yang diperlukan adalah nama, jenis kelamin, program studi, username, dan password pencari. Detail perancangan tabel data pencari kos dapat dilihat pada Tabel 2.7.

**Tabel 2.7** Tabel Pencari Kos

No.	Nama Atribut	Deskripsi	Tipe & Panjang	Keterangan
1.	id_pencari	Identitas pencari kos	Varchar(20)	<i>Primary Key</i>
2.	nama_pencari	Nama pencari kos	Varchar(30)	-
3.	jk_pencari	Jenis kelamin pencari kos	Varchar(1)	-
4.	username_pencari	Username pencari kos	Varchar(12)	-
5.	password_pencari	Password pencari kos	Varchar(20)	-
6.	id_prodi	Identitas program studi yang dimiliki pencari kos	Varchar(10)	<i>Foreign Key</i> dari Tabel Program Studi

## 8. Riwayat Pencarian

Perancangan basis data untuk tabel riwayat pencarian merupakan representasi untuk data daftar pencarian yang telah dilakukan pencari kos. Data riwayat pencarian yang

diperlukan adalah identitas pencari dan identitas kos. Detail perancangan tabel data riwayat pencarian dapat dilihat pada Tabel 2.8.

**Tabel 2.8** Tabel Riwayat Pencarian

No.	Nama Atribut	Deskripsi	Tipe & Panjang	Keterangan
1.	id_riwayat	Identitas pencari kos	Varchar(10)	<i>Primary Key</i>
2.	id_pencari	Identitas pencari kos	Varchar(20)	<i>Foreign Key</i> dari Tabel Pencari Kos
3.	id_kos	Identitas program studi yang dimiliki pencari kos	Varchar(10)	<i>Foreign Key</i> dari Tabel Identitas Kos

## 9. Identitas Kos

Perancangan basis data untuk tabel identitas kos merupakan representasi untuk data identitas kos, fasilitas kos, dan fasilitas terdekat dengan kos. Data identitas kos yang diperlukan adalah nama kos, alamat kos, lokasi kos, biaya sewa kos, jenis kelamin penghuni kos, jumlah penghuni per kamar, meja kursi, lemari, tempat tidur, kamar mandi luar, kipas angin, AC, Tv, Kulkas, dispenser, internet, ruang tamu, dapur, mesin cuci, pembantu, cuci gosok, cctv security, kolam renang, parkir motor, parkir mobil, tempat makan, warnet, mall, apotek dokter, atm bank, supermarket, dan kendaraan umum. Detail perancangan tabel data riwayat pencarian dapat dilihat pada Tabel 2.9.

**Tabel 2.9** Tabel Identitas Kos

No.	Nama Atribut	Deskripsi	Tipe & Panjang	Keterangan
1.	id_kos	Identitas pencari kos	Varchar(10)	<i>Primary Key</i>
2.	nama_kos	Nama pencari kos	Varchar(50)	-
3.	alamat_kos	Alamat pencari kos	Varchar(50)	-
4.	latitude_kos	Garis lintang bumi menunjukan lokasi kos	Float(10,6)	-
5.	longitude_kos	Garis bujur bumi yang menunjukan lokasi kos	Float(10,6)	-
6.	biaya_kos	Biaya sewa kos	Integer(7)	-
7.	jk	Jenis kelamin penghuni kos	Varchar(1)	-
8.	jmlh_penghuni	Jumlah penghuni dalam satu kamar	Varchar(1)	-
9.	meja_kursi	Fasilitas meja dan kursi yang dimiliki kos	Varchar(1)	-
10.	lemari	Fasilitas lemari yang dimiliki kos	Varchar(1)	-
11.	tmpt_tidur	Fasilitas tempat tidur yang dimiliki kos	Varchar(1)	-
12.	km_luar	Fasilitas kamar mandi luar yang dimiliki kos	Varchar(1)	-

**Lanjutan Tabel 2.9** Tabel Identitas Kos

No.	Nama Atribut	Deskripsi	Tipe & Panjang	Keterangan
13.	kipas_angin	Fasilitas kipas angin yang dimiliki kos	Varchar(1)	-
14.	AC	Fasilitas AC yang dimiliki kos	Varchar(1)	-
15.	Tv	Fasilitas televisi yang dimiliki kos	Varchar(1)	-
16.	kulkas	Fasilitas kulkas yang dimiliki kos	Varchar(1)	-
17.	dispenser	Fasilitas dispenser yang dimiliki kos	Varchar(1)	-
18.	internet	Fasilitas internet yang dimiliki kos	Varchar(1)	-
19.	ruang_tamu	Fasilitas meja ruang tamu yang dimiliki kos	Varchar(1)	-
20.	dapur	Fasilitas dapur yang dimiliki kos	Varchar(1)	-
21.	mesin_cuci	Fasilitas mesin cuci yang dimiliki kos	Varchar(1)	-
22.	pembantu	Fasilitas pembantu yang dimiliki kos	Varchar(1)	-
23.	cuci_gosok	Fasilitas cuci gosok yang dimiliki kos	Varchar(1)	-
24.	cctv_security	Fasilitas cctv/security yang dimiliki kos	Varchar(1)	-
25.	klm_renang	Fasilitas kolam renang yang dimiliki kos	Varchar(1)	-
26.	parkir_motor	Fasilitas parkir motor yang dimiliki kos	Integer(2)	-
27.	parkir_mobil	Fasilitas parkir mobil yang dimiliki kos	Integer(2)	-
28.	tmpt_makan	Jarak fasilitas umum tempat makan yang berada didekat kos	Double(2.2)	-
29.	warnet	Jarak fasilitas umum warnet yang berada didekat kos	Double(2.2)	-
30.	mall	Jarak fasilitas umum mall yang berada didekat kos	Double(2.2)	-
31.	apotek_dokter	Jarak fasilitas umum apotek/dokter yang berada didekat kos	Double(2.2)	-
32.	atm_bank	Jarak fasilitas umum atm/bank didekat kos	Double(2.2)	-
33.	supermarket	Jarak fasilitas umum supermarket yang berada didekat kos	Double(2.2)	-

**Lanjutan Tabel 2.9** Tabel Identitas Kos

No.	Nama Atribut	Deskripsi	Tipe & Panjang	Keterangan
34.	kendaraan_umum	Jarak fasilitas umum kendaraan umum yang berada didekat kos	Double(2.2)	-
35.	id_pemilik	Identitas pemilik yang dimiliki identitas kos	Varchar(20)	<i>Foreign Key</i> dari Tabel Pemilik kos
36.	id_kelurahan	Identitas kelurahan yang dimiliki identitas kos	Varchar(7)	<i>Foreign Key</i> dari Tabel Kelurahan
37.	id_tfn	Identitas fuzzy triangular number yang dimiliki identitas kos	Varchar(10)	<i>Foreign Key</i> dari Tabel Tfn Kos

#### 10. Tfn Kos

Perancangan basis data untuk tabel tfn kos representasi data TFN berdasarkan identitas kos. Data tfn yang diperlukan biaya sewa, jenis kelamin penghuni, jumlah penghuni per kamar, meja kursi, lemari, tempat tidur, kamar mandi luar, kipas angin, AC, Tv, Kulkas, dispenser, internet, ruang tamu, dapur, mesin cuci, pembantu, cuci gosok, cctv *secutiry*, kolam renang, parkir motor, parkir mobil, tempat makan, warnet, mall, apotek, bank, supermarket, dan kendaraan umum. Detail perancangan tabel data riwayat pencarian pada Tabel 2.10.

**Tabel 2.10** Tabel Tfn

No.	Nama Atribut	Deskripsi	Tipe & Panjang	Keterangan
1.	id_tfn	Identitas pencari kos	Varchar(10)	<i>Primary Key</i>
2.	biaya_tfn	<i>Fuzzy Triangular Number</i> untuk biaya sewa kos	Varchar(20)	-
3.	jk_tfn	<i>Fuzzy Triangular Number</i> untuk jenis kelamin penghuni kos	Varchar(1)	-
4.	jmlh_penghuni_tfn	<i>Fuzzy Triangular Number</i> untuk jumlah penghuni dalam satu kamar	Varchar(20)	-
5.	meja_kursi_tfn	<i>Fuzzy Triangular Number</i> untuk fasilitas meja dan kursi yang dimiliki kos	Varchar(1)	-
6.	lemari_tfn	<i>Fuzzy Triangular Number</i> untuk fasilitas lemari yang dimiliki kos	Varchar(1)	-
7.	tmpt_tidur_tfn	<i>Fuzzy Triangular Number</i> untuk fasilitas tempat tidur yang dimiliki kos	Varchar(1)	-



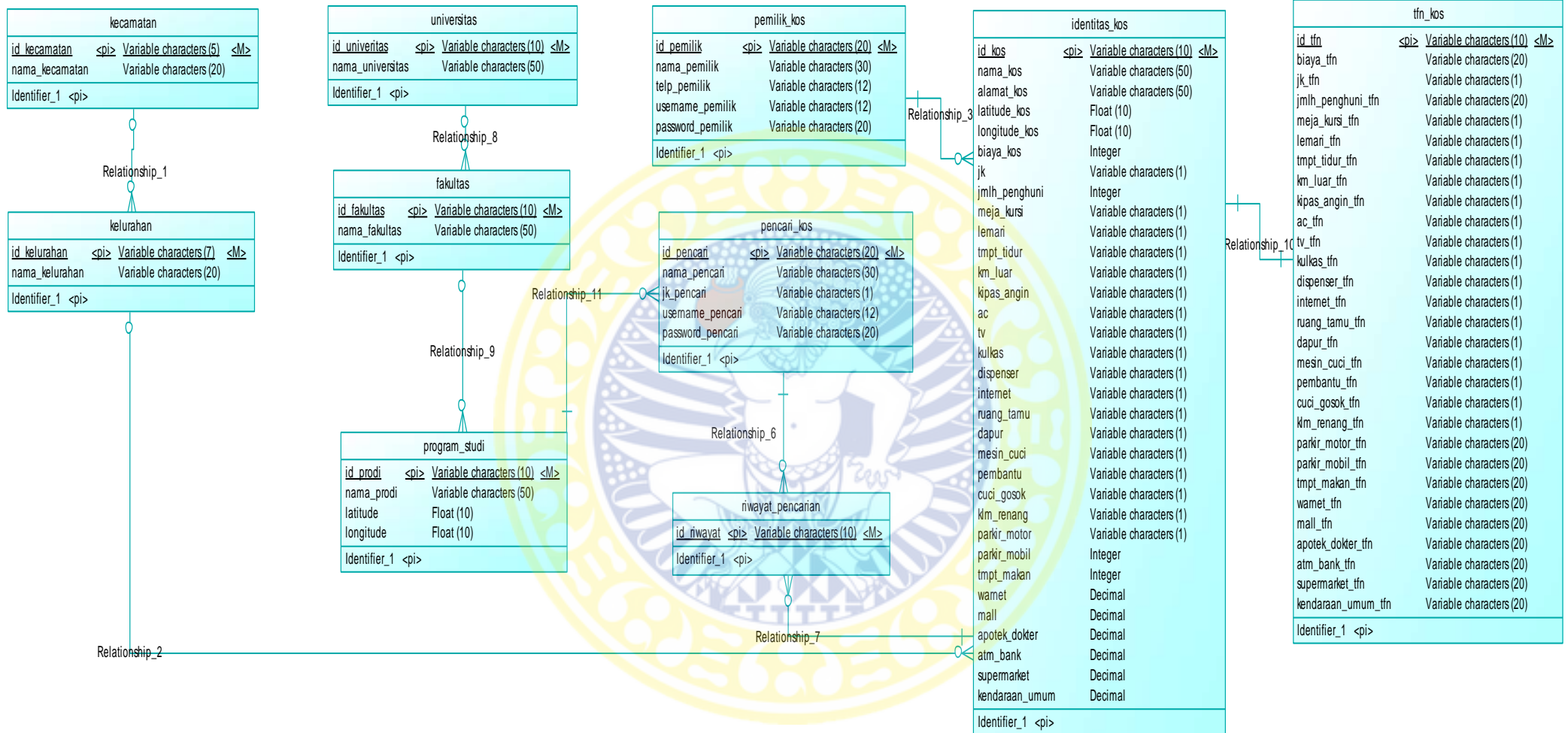
**Lanjutan Tabel 2.10 Tabel Tfn**

No.	Nama Atribut	Deskripsi	Tipe & Panjang	Keterangan
8.	km_luar_tfn	<i>Fuzzy Triangular Number</i> untuk fasilitas kamar mandi yang dimiliki kos	Varchar(1)	-
9.	kipas_angin_tfn	<i>Fuzzy Triangular Number</i> untuk fasilitas kipas angin yang dimiliki kos	Varchar(1)	-
10.	ac_tfn	<i>Fuzzy Triangular Number</i> untuk fasilitas AC yang dimiliki kos	Varchar(1)	-
11.	tv_tfn	<i>Fuzzy Triangular Number</i> untuk fasilitas televisi yang dimiliki kos	Varchar(1)	-
12.	kulkas_tfn	<i>Fuzzy Triangular Number</i> untuk fasilitas kulkas yang dimiliki kos	Varchar(1)	-
13.	dispenser_tfn	<i>Fuzzy Triangular Number</i> untuk fasilitas dispenser yang dimiliki kos	Varchar(1)	-
14.	internet_tfn	<i>Fuzzy Triangular Number</i> untuk fasilitas internet yang dimiliki kos	Varchar(1)	-
15.	ruang_tamu_tfn	<i>Fuzzy Triangular Number</i> untuk fasilitas meja ruang tamu yang dimiliki kos	Varchar(1)	-
16.	dapur_tfn	<i>Fuzzy Triangular Number</i> untuk fasilitas dapur yang dimiliki kos	Varchar(1)	-
17.	mesin_cuci_tfn	<i>Fuzzy Triangular Number</i> untuk fasilitas mesin cuci yang dimiliki kos	Varchar(1)	-
18.	pembantu_tfn	<i>Fuzzy Triangular Number</i> untuk fasilitas pembantu	Varchar(1)	-
19.	cuci_gosok_tfn	<i>Fuzzy Triangular Number</i> untuk fasilitas cuci gosok	Varchar(1)	-
20.	cctv_security_tfn	<i>Fuzzy Triangular Number</i> untuk fasilitas cctv/security/satpam yang dimiliki kos	Varchar(1)	-
21.	klm_renang_tfn	<i>Fuzzy Triangular Number</i> untuk fasilitas kolam renang yang dimiliki kos	Varchar(1)	-
22.	parkir_motor_tfn	<i>Fuzzy Triangular Number</i> untuk fasilitas parkir motor yang dimiliki kos	Integer(1)	-

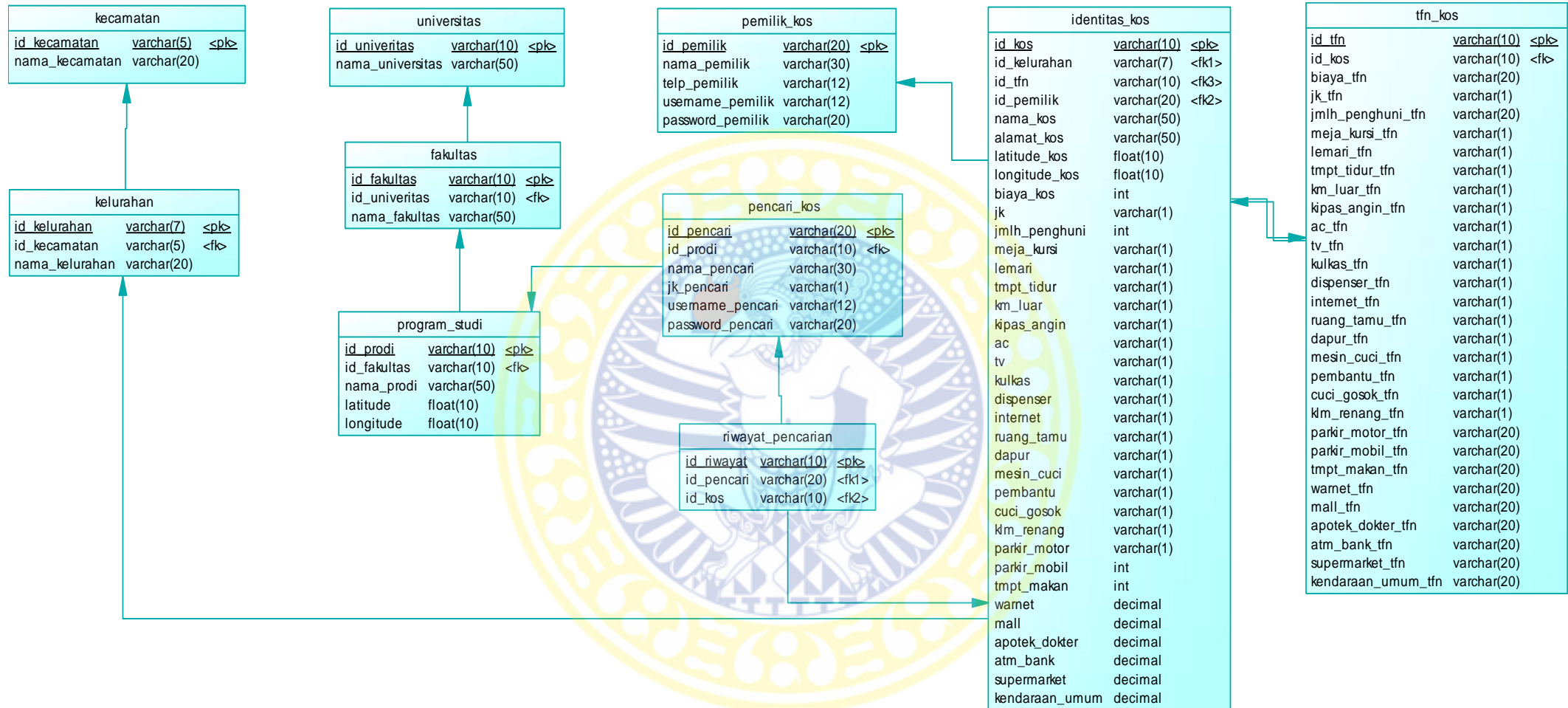


**Lanjutan Tabel 2.10** Tabel Tfn

No.	Nama Atribut	Deskripsi	Tipe & Panjang	Keterangan
23.	parkir_mobil_tfn	<i>Fuzzy Triangular Number</i> untuk fasilitas parkir mobil yang dimiliki kos	Varchar(20)	-
24.	tmpt_makan_tfn	<i>Fuzzy Triangular Number</i> untuk jarak fasilitas umum tempat makan/depot/ <i>restaurant</i> yang berada didekat kos	Varchar(20)	-
25.	warnet_tfn	<i>Fuzzy Triangular Number</i> untuk jarak fasilitas umum warnet yang berada didekat kos	Varchar(20)	-
26.	mall_tfn	<i>Fuzzy Triangular Number</i> untuk jarak fasilitas umum mall yang berada didekat kos	Varchar(20)	-
27.	apotek_dokter_tfn	<i>Fuzzy Triangular Number</i> untuk jarak fasilitas umum apotek/dokter yang berada didekat kos	Varchar(20)	-
28.	atm_bank_tfn	<i>Fuzzy Triangular Number</i> untuk jarak fasilitas umum atm/bank yang berada didekat kos	Varchar(20)	-
29.	supermarket_tfn	<i>Fuzzy Triangular Number</i> untuk jarak fasilitas umum supermarket yang berada didekat kos	Varchar(20)	-
30.	kendaraan_umum_tfn	<i>Fuzzy Triangular Number</i> untuk jarak fasilitas umum kendaraan umum yang berada didekat kos	Varchar(20)	-



**Gambar 2.9** Conceptual Data Model



**Gambar 2.10** *Physical Data Model*

## 2.3 Perancangan Antarmuka

Perancangan antar muka akan dilakukan agar desain yang dibuat dapat mudah dipahami oleh user.

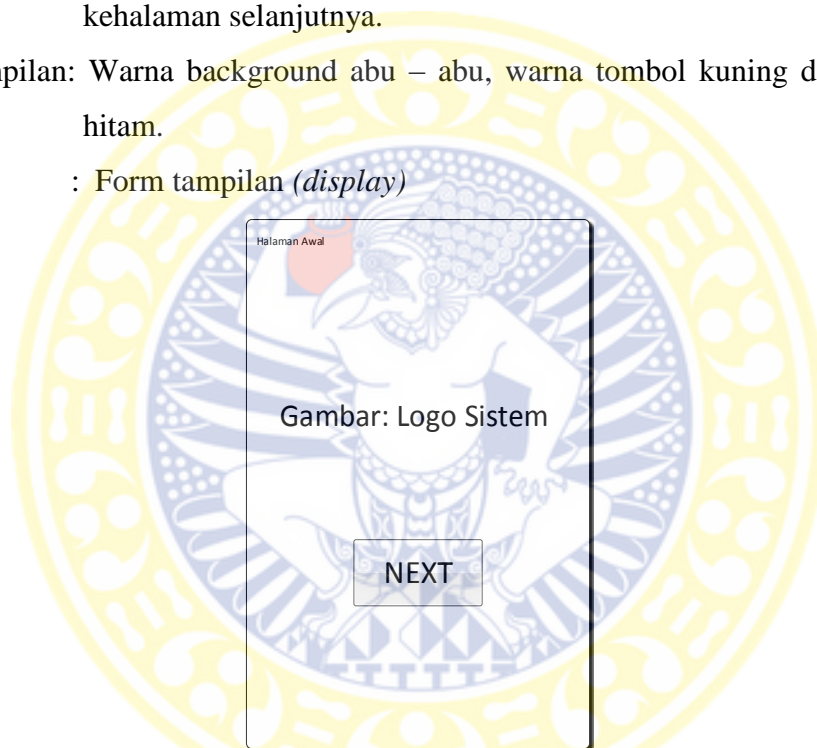
### 1. Halaman Awal

Halaman ini adalah halaman yang pertama kali akan dilihat oleh aktor (pemilik dan pencari). Halaman awal dapat dilihat pada Gambar 4.25 dan Deskripsi Objek dapat dilihat pada Tabel 4.54.

Deskripsi Isi : Di halaman ini akan ditampilkan gambar dan tombol untuk pergi kehalaman selanjutnya.

Desain Tampilan: Warna background abu – abu, warna tombol kuning dan font berwarna hitam.

Jenis : Form tampilan (*display*)



**Gambar 4.1** Perancangan Halaman Awal

**Tabel 4.1** Deskripsi Objek Halaman Awal

Objek	Jenis	Keterangan
btnNext	<i>Button</i>	Menuju halaman masuk
gambarLogo	Gambar	Menampilkan gambar logo dari sistem

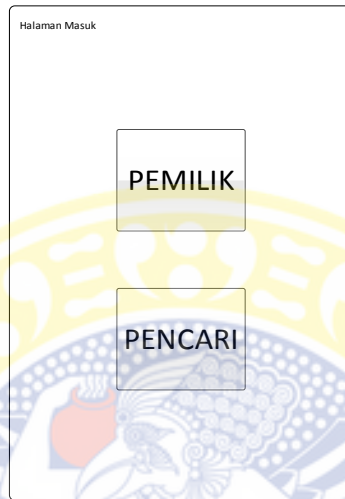
### 2. Halaman Masuk

Pada halaman ini aktor akan dibagi menjadi dua bagian yang merepresentasikan diri, yaitu pencari dan pemilik kos. Halaman masuk dapat dilihat pada Gambar 4.26 dan Deskripsi Objek dapat dilihat pada Tabel 4.55.

Deskripsi Isi : Halaman ini akan ditampilkan tombol untuk pergi kehalaman selanjutnya sebagai pemilik kos atau pencari kos.

Desain Tampilan: Warna background abu – abu, warna tombol kuning dan font berwarna hitam.

Jenis : Form tampilan (*display*)



**Gambar 4.2** Perancangan Halaman Masuk

**Tabel 4.2** Deskripsi Objek Halaman Masuk

Objek	Jenis	Keterangan
btnPemilik	<i>Button</i>	Menuju halaman khusus untuk pemilik kos
btnPencari	<i>Button</i>	Menuju halaman khusus untuk pencari kos

### 3. Halaman *Signup*/Daftar sebagai Pemilik

Pada halaman ini, pemilik mendaftarkan diri dengan cara memasukan data diri. Halaman *signup*/daftar sebagai pemilik dapat dilihat pada Gambar 4.27 dan Deskripsi Objek dapat dilihat pada Tabel 4.56.

Deskripsi Isi : Halaman ini merupakan halaman masukan untuk mendaftar sebagai pemilik kos. Masukan yang dibutuhkan nama, telpon, *username*, *password*, dan *retype password*.

Desain Tampilan: Warna background kombinasi abu – abu serta putih, warna tombol kuning dan font berwarna hitam.

Jenis :Form masukan (*input*)



Halaman Signup Pemilik

Nama

Telp

Username

Password

Retype Password

SIGN UP

**Gambar 4.3** Perancangan Halaman Daftar Pemilik

**Tabel 4.3** Deskripsi Objek Halaman Daftar Pemilik

Objek	Jenis	Keterangan
Text Nama	<i>Edit Text</i>	<i>Field</i> untuk memasukkan nama pemilik kos
Text Telp	<i>Edit Text</i>	<i>Field</i> untuk memasukkan telpon pemilik kos
Text <i>Username</i>	<i>Edit Text</i>	<i>Field</i> untuk memasukkan <i>username</i> pemilik kos
Text <i>Password</i>	<i>Edit Text</i>	<i>Field</i> untuk memasukkan <i>password</i> pemilik kos
Text <i>Retype Password</i>	<i>Edit Text</i>	<i>Field</i> untuk memasukkan <i>retype password</i> pemilik kos
btnSignup	<i>Button</i>	Menuju halaman khusus untuk pemilik kos

#### 4. Halaman *Login/Masuk* sebagai Pemilik

Pada halaman ini, pemilik dapat masuk kedalam sistem. Halaman *Login/Masuk* sebagai Pemilik dapat dilihat pada Gambar 4.28 dan Deskripsi Objek dapat dilihat pada Tabel 4.57.

Deskripsi Isi : Halaman ini merupakan halaman masukan untuk masuk ke dalam sistem sebagai pemilik kos. Masukan yang dibutuhkan adalah *username* dan *password*.

Desain Tampilan: Warna background kombinasi abu – abu serta putih, warna tombol kuning dan font berwarna hitam.

Jenis : Form masukan (*input*)

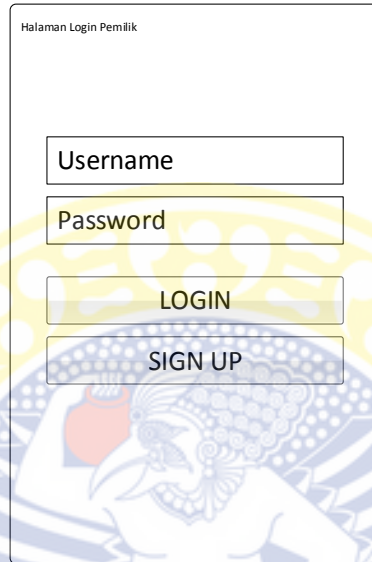
#### 5. Halaman Utama Pemilik

Pada halaman ini, pemilik memiliki dua layout yang dapat digeser. Layout pertama untuk menampilkan data diri pemilik dan layout kedua untuk menampilkan pemberitahuan seputar pengelolaan data kos. Halaman utama pemilik data diri dapat dilihat pada Gambar 4.29 dan deskripsi objek dapat dilihat pada Tabel 4.58.

Deskripsi Isi : Halaman menampilkan data diri pemilik dan tombol untuk ke halaman ubah data diri.

Desain Tampilan: Warna background abu – abu, warna tombol kuning dan font berwarna hitam.

Jenis : Form tampilan (*display*)



**Gambar 4.4** Perancangan Halaman Masuk Pemilik

**Tabel 4.4** Deskripsi Objek Halaman Masuk Pemilik

Objek	Jenis	Keterangan
Text <i>Username</i>	<i>Edit Text</i>	Field untuk memasukkan <i>username</i> pemilik kos
Text <i>Password</i>	<i>Edit Text</i>	Field untuk memasukkan <i>password</i> pemilik kos
btnLogin	<i>Button</i>	Menuju halaman utama pemilik kos
btnSignup	<i>Button</i>	Menuju halaman daftar untuk pemilik kos



**Gambar 4.5** Perancangan Halaman Utama Pemilik Data Diri

**Tabel 4.5** Deskripsi Objek Perancangan Halaman Utama Pemilik Data Diri

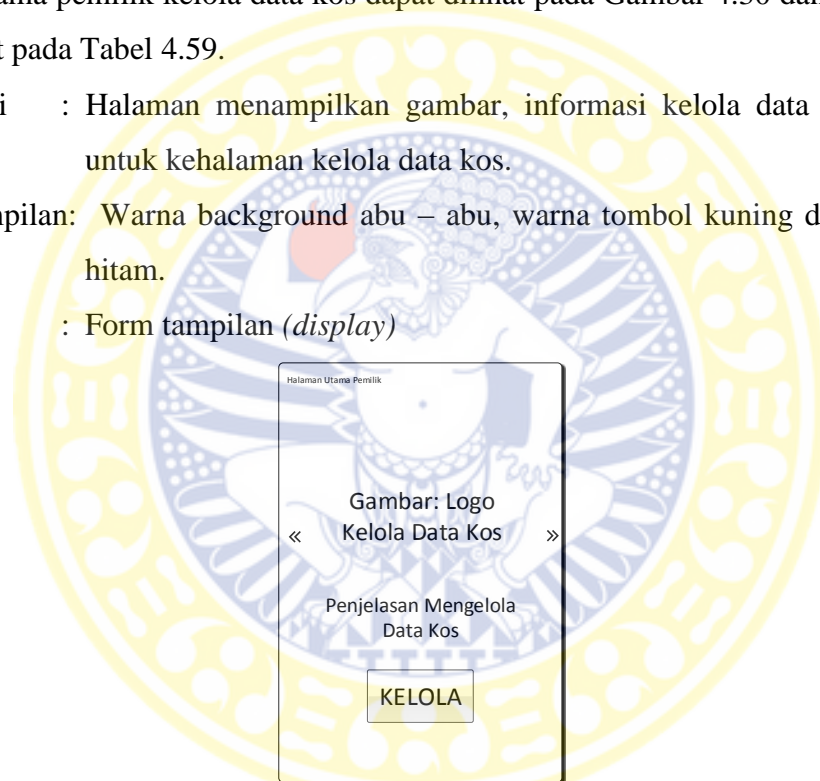
Objek	Jenis	Keterangan
gmbrLogoPemilik	Gambar	Menampilkan gambar logo pemilik
Text Nama	<i>Text Area</i>	<i>Field</i> untuk menampilkan nama pemilik kos
Text Telp	<i>Text Area</i>	<i>Field</i> untuk menampilkan telpon pemilik kos
Text <i>Username</i>	<i>Text Area</i>	<i>Field</i> untuk menampilkan <i>username</i> pemilik kos
Text <i>Password</i>	<i>Text Area</i>	<i>Field</i> untuk menampilkan <i>password</i> pemilik kos
btnUbah	<i>Button</i>	Menuju halaman mengubah data pemilik kos

Halaman utama pemilik kelola data kos dapat dilihat pada Gambar 4.30 dan Deskripsi Objek dapat dilihat pada Tabel 4.59.

Deskripsi Isi : Halaman menampilkan gambar, informasi kelola data kos, dan tombol untuk kehalaman kelola data kos.

Desain Tampilan: Warna background abu – abu, warna tombol kuning dan font berwarna hitam.

Jenis : Form tampilan (*display*)



**Gambar 4.6** Perancangan Halaman Utama Pemilik Kelola Data Kos

**Tabel 4.6** Deskripsi Objek Perancangan Halaman Utama Pemilik Kelola Data Kos

Objek	Jenis	Keterangan
gmbrKelola Data Kos	Gambar	Menampilkan gambar logo kelola data kos
Text Info Kelola Data Kos	<i>Text Area</i>	<i>Field</i> untuk menampilkan informasi kelola data kos
btnKelola	<i>Button</i>	Menuju halaman mengelola data kos

## 6. Halaman Ubah Data Diri Pemilik

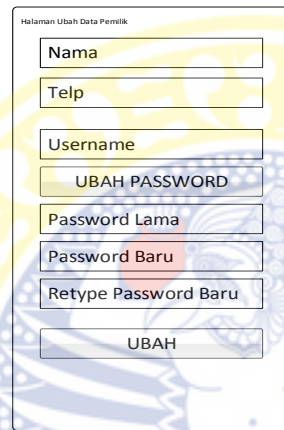
Pada halaman ini, pemilik dapat mengubah data diri yang telah tersimpan agar data diri selalu terbaru. Halaman utama pemilik dapat dilihat pada Gambar 4.31 dan Deskripsi Objek

dapat dilihat pada Tabel 4.60.

Deskripsi Isi : Halaman ini merupakan halaman masukan untuk mengubah data diri pemilik kos. Masukan yang bisa diubah adalah nama, telpon, dan *username*. Masukan yang harus diisi untuk mengubah *password* adalah *password* lama, *password* baru, dan *retype password* baru.

Desain Tampilan: Warna background kombinasi abu – abu serta putih, warna tombol kuning dan font berwarna hitam.

Jenis : Form masukan (*input*)



**Gambar 4.7** Perancangan Halaman Ubah Data Diri Pemilik

**Tabel 4.7** Deskripsi Objek Perancangan Halaman Ubah Data Diri Pemilik

Objek	Jenis	Keterangan
Text Nama	<i>Edit Text</i>	<i>Field</i> untuk memasukkan nama pemilik kos
Text Telp	<i>Edit Text</i>	<i>Field</i> untuk memasukkan telpon pemilik kos
Text <i>Username</i>	<i>Edit Text</i>	<i>Field</i> untuk memasukkan <i>username</i> pemilik kos
Text <i>Password</i> Lama	<i>Edit Text</i>	<i>Field</i> untuk memasukkan <i>password</i> lama pemilik kos
Text <i>Password</i> Baru	<i>Edit Text</i>	<i>Field</i> untuk memasukkan <i>password</i> baru pemilik kos
Text <i>Retype Password</i> Baru	<i>Edit Text</i>	<i>Field</i> untuk memasukkan <i>retype password</i> baru pemilik kos
btnUbah Password	<i>Button</i>	Mengubah <i>password</i> pemilik kos
btnUbah	<i>Button</i>	Mengubah data diri untuk pemilik kos

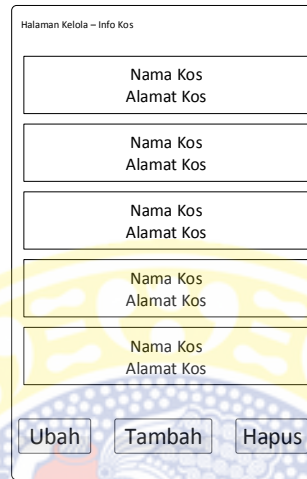
## 7. Halaman Daftar Kelola Data Kos

Pada halaman ini, pemilik dapat melihat daftar kelola data kos yang dimiliki dan juga memilih aktifitas pengelolaan yang dapat dilakukan dengan data kos. Halaman daftar kelola data kos dapat dilihat pada Gambar 4.32 dan deskripsi objek dapat dilihat pada Tabel 4.61.

Deskripsi Isi : Pada halaman ini ditampilkan daftar kelola data kos.

Desain Tampilan: Warna background kombinasi abu – abu serta putih, warna tombol kuning dan font berwarna hitam.

Jenis : Form tampilan (*display*)



**Gambar 4.8** Perancangan Halaman Daftar Kelola Data Kos

**Tabel 4.8** Deskripsi Objek Perancangan Halaman Daftar Kelola Data Kos

Objek	Jenis	Keterangan
TextNama Kos	<i>Text Area</i>	<i>Field</i> untuk menampilkan nama kos
TextAlamat Kos	<i>Text Area</i>	<i>Field</i> untuk menampilkan alamat kos
btnUbah	<i>Button</i>	Mengubah data kos
btnTambah	<i>Button</i>	Menambah data kos
btnHapus	<i>Button</i>	Menghapus data kos

#### 8. Halaman Detail Informasi Kelola Data Kos

Pada halaman ini, pemilik dapat melakukan aktifitas yang telah dipilih pada halaman sebelumnya. Jika pemilik memilih menambah data maka halaman ini akan disajikan tanpa data, apanila pemilik memilih mengubah data yang telah ada maka halaman ini akan disajikan dengan data yang telah tersimpan sebelumnya. Halaman ini akan dibagi menjadi tiga layout yang dapat digeser. Halaman detail informasi kelola data kos informasi utama dapat dilihat pada Gambar 4.33 dan deskripsi objek dapat dilihat pada Tabel 4.62.

Deskripsi Isi : Halaman ini merupakan halaman masukan untuk menambah dan mengubah data kos. Masukan yang dibutuhkan adalah nama kos, alamat kos, kecamatan, kelurahan, biaya sewa, jenis kelamin penghuni, dan jumlah penghuni tiap kamar.



Desain Tampilan: Warna background kombinasi abu – abu serta putih, warna tombol kuning dan font berwarna hitam.

Jenis : Form masukan (*input*)

**Gambar 4.9** Perancangan Halaman Detail Informasi Kelola Data Kos Informasi Utama

**Tabel 4.9** Deskripsi Objek Perancangan Halaman Detail Informasi Kelola Data Kos Informasi Utama

Objek	Jenis	Keterangan
Text Nama	<i>Edit Text</i>	<i>Field</i> untuk memasukkan nama kos
Text Alamat	<i>Edit Text</i>	<i>Field</i> untuk memasukkan alamat kos
Text Kecamatan	<i>Drop down list</i>	Menampilkan daftar fakultas dari data yang telah disediakan
Text Kelurahan	<i>Drop down list</i>	Menampilkan daftar program studi dari data yang telah disediakan
TextBiaya Sewa	<i>Edit Text</i>	<i>Field</i> untuk memasukkan biaya sewa kos
Text Jenis Kelamin	<i>Edit Text</i>	<i>Field</i> untuk memasukkan jenis kelamin penghuni kos
Text Jumlah Penghuni	<i>Edit Text</i>	<i>Field</i> untuk memasukkan jumlah penghuni perkamar

Halaman detail informasi kelola data kos fasilitas kos dapat dilihat pada Gambar 4.34 dan deskripsi objek dapat dilihat pada Tabel 4.63.

Deskripsi Isi : Halaman ini merupakan halaman masukan untuk menambah dan mengubah data kos. Masukan yang dibutuhkan adalah fasilitas meja dan kursi, lemari, tempat tidur, kipas angin, AC, kamar mandi luar, internet, ruang tamu, dapur, kolam renang, Tv, cuci gosok, mesin cuci, dispenser, pembantu, kulkas, cctv, jumlah parkir motor, dan jumlah parkir mobil.

Desain Tampilan: Warna background kombinasi abu – abu serta putih, warna tombol kuning dan font berwarna hitam.

Jenis : Form masukan (*input*)

Jenis : Form masukan (*input*)

**Gambar 4.10** Perancangan Halaman Detail Informasi Kelola Data Kos Fasilitas Kos

**Tabel 4.10** Deskripsi Objek Perancangan Halaman Detail Informasi Kelola Data Kos Fasilitas Kos

Objek	Jenis	Keterangan
cbMeja Kursi	<i>Check Box</i>	Check Box untuk data meja kursi
cbLemari	<i>Check Box</i>	Check Box untuk data lemari
cbTempat Tidur	<i>Check Box</i>	Check Box untuk data tempat tidur
cbKipas Angin	<i>Check Box</i>	Check Box untuk data kipas angina
cbAc	<i>Check Box</i>	Check Box untuk data AC
cbKamar Mandi Luar	<i>Check Box</i>	Check Box untuk data kamar mandi luar
cbInternet	<i>Check Box</i>	Check Box untuk data internet
cbRuang Tamu	<i>Check Box</i>	Check Box untuk data ruang tamu
cbDapur	<i>Check Box</i>	Check Box untuk data dapur
cbKolamRenang	<i>Check Box</i>	Check Box untuk data kolam renang
cbTv	<i>Check Box</i>	Check Box untuk data Tv
cbCuci Gosok	<i>Check Box</i>	Check Box untuk data cuci gosok
cbMesin cuci	<i>Check Box</i>	Check Box untuk data mesin cuci
cbDispenser	<i>Check Box</i>	Check Box untuk data dispenser
cbPembantu	<i>Check Box</i>	Check Box untuk data pembantu
cbKulkas	<i>Check Box</i>	Check Box untuk data kulkas
cbCctv	<i>Check Box</i>	Check Box untuk data CCTV
Text Parkir Motor	<i>Edit Text</i>	Field untuk memasukkan jumlah parkir motor
Text Parkir Mobil	<i>Edit Text</i>	Field untuk memasukkan jumlah parkir mobil

Halaman detail informasi kelola data kos fasilitas umum terdekat dapat dilihat pada Gambar 4.35 dan deskripsi objek dapat dilihat pada Tabel 4.64.

Deskripsi Isi : Halaman ini merupakan halaman masukan untuk menambah dan mengubah data kos. Masukan yang dibutuhkan jarak fasilitas umum terdekat seperti restaurant, warnet, mall, apotek, bank, kendaraan umum dan supermarket.

Desain Tampilan: Warna background kombinasi abu – abu serta putih, warna tombol kuning dan font berwarna hitam.

Jenis : Form masukan (*input*)



**Gambar 4.11** Perancangan Halaman Detail Informasi Kelola Data Kos Fasilitas Umum Terdekat

**Tabel 4.11** Deskripsi Objek Perancangan Halaman Detail Informasi Kelola Data Kos Fasilitas Umum Terdekat

Objek	Jenis	Keterangan
Text Restaurant	<i>Edit Text</i>	<i>Field</i> untuk memasukkan jarak kos dengan restaurant
Text Warnet	<i>Edit Text</i>	<i>Field</i> untuk memasukkan jarak kos dengan warnet
Text Mall	<i>Edit Text</i>	<i>Field</i> untuk memasukkan jarak kos dengan mall
Text Apotek	<i>Edit Text</i>	<i>Field</i> untuk memasukkan jarak kos dengan apotek
Text Bank	<i>Edit Text</i>	<i>Field</i> untuk memasukkan jarak kos dengan bank
Text Kendaraan Umum	<i>Edit Text</i>	<i>Field</i> untuk memasukkan jarak kos dengan kendaraan umum
Text Supermarket	<i>Edit Text</i>	<i>Field</i> untuk memasukkan jarak kos dengan supermarket
btnSave	<i>Button</i>	Memasukan data kos

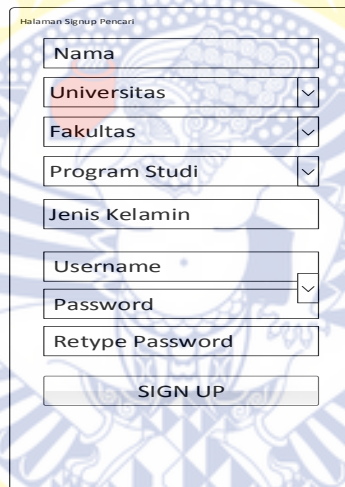
#### 9. Halaman *Signup*/Daftar sebagai Pencari

Pada halaman ini, pencari mendaftarkan diri dengan cara memasukan data diri. Halaman *signup*/daftar sebagai pencari dapat dilihat pada Gambar 4.36 dan deskripsi objek dapat dilihat pada Tabel 4.65.

Deskripsi Isi : Di halaman ini akan masukan untuk mendaftar sebagai pencari kos. Masukan yang dibutuhkan adalah nama, jenis kelamin, universitas, fakultas, program studi, *username*, *password*, dan *retype password*.

Desain Tampilan: Warna background kombinasi abu – abu serta putih, warna tombol kuning dan font berwarna hitam.

Jenis : Form masukan (*input*)



**Gambar 4.12** Perancangan Halaman Daftar Pencari

#### 10. Halaman *Login*/Masuk sebagai Pencari

Pada halaman ini, pencari dapat masuk kedalam sistem. Halaman *Login*/Masuk sebagai Pencari dapat dilihat pada Gambar 4.37 dan deskripsi objek dapat dilihat pada Tabel 4.66.

Deskripsi Isi : Halaman ini merupakan halaman masukan untuk masuk ke dalam sistem sebagai pencari kos. Masukan yang dibutuhkan adalah *username* dan *password*.

Desain Tampilan: Warna background kombinasi abu – abu serta putih, warna tombol kuning dan font berwarna hitam.

Jenis : Form masukan (*input*)



**Tabel 4.12** Deskripsi Objek Halaman Daftar Pencari

Objek	Jenis	Keterangan
Text Nama	Text Area	<i>Field</i> untuk memasukkan nama pemilik kos
Text Universitas	Drop down list	Menampilkan daftar universitas dari data yang telah disediakan
Text Fakultas	Drop down list	Menampilkan daftar fakultas dari data yang telah disediakan
Text Program Studi	Drop down list	Menampilkan daftar program studi dari data yang telah disediakan
Text Jenis Kelamin	Drop down list	Menampilkan jenis kelamin dari data yang telah disediakan
Text <i>Username</i>	Text Area	<i>Field</i> untuk memasukkan <i>username</i> pemilik kos
Text <i>Password</i>	Text Area	<i>Field</i> untuk memasukkan <i>password</i> pemilik kos
Text <i>Retype Password</i>	Text Area	<i>Field</i> untuk memasukkan <i>retype password</i> pemilik kos
btnSignup	button	Menuju halaman khusus untuk pencari kos

**Gambar 4.13** Perancangan Halaman Masuk Pencari

**Tabel 4.13** Deskripsi Objek Perancangan Halaman Masuk Pencari

Objek	Jenis	Keterangan
Text <i>Username</i>	<i>Edit Text</i>	<i>Field</i> untuk memasukkan <i>username</i> pencari kos
Text <i>Password</i>	<i>Edit Text</i>	<i>Field</i> untuk memasukkan <i>password</i> pencari kos
btnLogin	<i>Button</i>	Menuju halaman utama pencari kos
btnSignup	<i>Button</i>	Menuju halaman daftar untuk pencari kos

## 11. Halaman Utama Pencari

Pada halaman ini, pencari memiliki tiga layout yang dapat digeser. Layout pertama untuk menampilkan data diri pencari, layout kedua untuk menampilkan pemberitahuan seputar pencarian data kos, dan layout ketiga untuk menampilkan pemberitahuan seputar riwayat pencarian data kos. Halaman utama pencari data diri dapat dilihat pada Gambar 4.38 dan deskripsi objek dapat dilihat pada Tabel 4.67.

Deskripsi Isi : Halaman menampilkan data diri pemilik dan tombol untuk ke halaman ubah data diri.

Desain Tampilan: Warna background abu – abu, warna tombol kuning dan font berwarna hitam.

Jenis : Form tampilan (*display*)

**Gambar 4.14** Perancangan Halaman Utama Pencari Data Diri

**Tabel 4.14** Deskripsi Objek Perancangan Halaman Utama Pencari Data Diri

Objek	Jenis	Keterangan
gambarLogo Pencari	Gambar	Menampilkan gambar logo pencari
Text Nama	<i>Text Area</i>	<i>Field</i> untuk menampilkan nama pencari kos
Text Jenis Kelamin	<i>Text Area</i>	<i>Field</i> untuk menampilkan jenis kelamin pencari kos
Text Universitas	<i>Text Area</i>	<i>Field</i> untuk menampilkan universitas pencari kos
Text Fakultas	<i>Text Area</i>	<i>Field</i> untuk menampilkan fakultas pencari kos
Text Program Studi	<i>Text Area</i>	<i>Field</i> untuk menampilkan program studi pencari kos
Text <i>Username</i>	<i>Text Area</i>	<i>Field</i> untuk menampilkan <i>username</i> pencari kos
Text <i>Password</i>	<i>Text Area</i>	<i>Field</i> untuk menampilkan <i>password</i> pencari kos
btnUbah	<i>Button</i>	Menuju halaman mengubah data pencari kos

Halaman utama pencari cari data kos dapat dilihat pada Gambar 4.39 dan deskripsi objek dapat dilihat pada Tabel 4.68.

Deskripsi Isi : Halaman menampilkan gambar, informasi cari data kos, dan tombol untuk kehalaman cari data kos.

Desain Tampilan : Warna background abu – abu, warna tombol kuning dan font berwarna hitam.

Jenis : Form tampilan (*display*)



**Gambar 4.15** Perancangan Halaman Utama Pencari Cari Data Kos

**Tabel 4.15** Deskripsi Objek Perancangan Halaman Utama Pencari Cari Data Kos

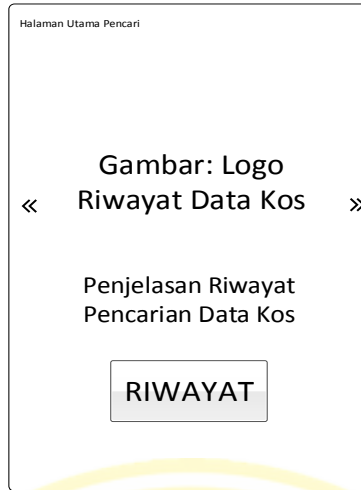
Objek	Jenis	Keterangan
Gambar Logo Cari Data Kos	Gambar	Menampilkan gambar logo cari data kos
Text Info Cari Data Kos	<i>Text Area</i>	<i>Field</i> untuk menampilkan informasi cari data kos
btnCari	<i>Button</i>	Menuju halaman mencari data kos

Halaman utama pencari riwayat pencarian data kos dapat dilihat pada Gambar 4.40 dan deskripsi objek dapat dilihat pada Tabel 4.69.

Deskripsi Isi : Halaman menampilkan gambar, informasi riwayat pencarian kos, dan tombol untuk kehalaman riwayat pencarian kos.

Desain Tampilan: Warna background abu – abu, warna tombol kuning dan font berwarna hitam.

Jenis : Form tampilan (*display*)



**Gambar 4.16** Perancangan Halaman Utama Pencari Riwayat Pencarian

**Tabel 4.16** Deskripsi Objek Perancangan Halaman Utama Pencari Riwayat Pencarian

Objek	Jenis	Keterangan
Gambar Logo Riwayat Pencarian Data Kos	Gambar	Menampilkan gambar logo riwayat pencarian data kos
Text Info Riwayat Pencarian Data Kos	<i>Text Area</i>	<i>Field</i> untuk menampilkan informasi riwayat pencarian data kos
btnRiwayat	<i>Button</i>	Menuju halaman riwayat pencarian data kos

## 12. Halaman Ubah Data Diri Pencari

Pada halaman ini, pencari dapat mengubah data diri yang telah tersimpan agar data diri selalu terbaru. Halaman ubah data diri pencari dapat dilihat pada Gambar 4.41 dan Deskripsi Objek dapat dilihat pada Tabel 4.70.

Deskripsi Isi : Di halaman ini akan masukan untuk mengubah data diri pencari kos. Masukan yang dibutuhkan adalah nama, alamat, telpon, jenis kelamin, universitas, fakultas, program studi, *username*, *password*, dan *retype password*.

Desain Tampilan: Warna background kombinasi abu – abu serta putih, warna tombol kuning dan font berwarna hitam.

Jenis : Form masukan (*input*)

## 13. Halaman Pencarian Data Kos

Pada halaman ini, pencari dapat mencari data kos sesuai kriteria dan prioritas yang diinginkan. Halaman utama pencari data diri dapat dilihat pada Gambar 4.42 dan deskripsi objek dapat dilihat pada Tabel 4.71.



Deskripsi Isi : Halaman ini merupakan halaman masukan untuk mencari data kos.  
Masukan yang dibutuhkan adalah *username* dan *password*.

Desain Tampilan: Warna background abu – abu, warna tombol kuning dan font berwarna hitam.

Jenis : Form masukan (*input*)

**Gambar 4.17** Perancangan Halaman Ubah Data Diri Pencari

**Tabel 4.17** Deskripsi Objek Halaman Ubah Data Diri Pencari

Objek	Jenis	Keterangan
Text Nama	Text Area	<i>Field</i> untuk memasukkan nama pemilik kos
Text Universitas	Drop down list	Menampilkan daftar universitas dari data yang telah disediakan
Text Fakultas	Drop down list	Menampilkan daftar fakultas dari data yang telah disediakan
Text Program Studi	Drop down list	Menampilkan daftar program studi dari data yang telah disediakan
Text Alamat	Text Area	<i>Field</i> untuk memasukkan alamat pemilik kos
Text Telp	Text Area	<i>Field</i> untuk memasukkan telpon pemilik kos
Text Jenis Kelamin	Drop down list	Menampilkan jenis kelamin dari data yang telah disediakan
Text <i>Username</i>	Text Area	<i>Field</i> untuk memasukkan <i>username</i> pemilik kos
Text <i>Password</i>	Text Area	<i>Field</i> untuk memasukkan <i>password</i> pemilik kos
Text <i>Retype Password</i>	Text Area	<i>Field</i> untuk memasukkan <i>retype password</i> pemilik kos
btnSignup	button	Menuju halaman khusus untuk pencari kos

Halaman Cari - Input Prioritas Kos

Jarak	<input type="text"/>	Biaya Kos	<input type="text"/>
Jenis Kelamin	<input type="text"/>	Jmlh Penghuni	<input type="text"/>
Meja Kursi	<input type="text"/>	Lemari	<input type="text"/>
Tempat Tidur	<input type="text"/>	Km Dalam	<input type="text"/>
Kipas Angin	<input type="text"/>	AC	<input type="text"/>
Tv	<input type="text"/>	Kulkas	<input type="text"/>
Dispenser	<input type="text"/>	Internet	<input type="text"/>
Ruang Tamu	<input type="text"/>	Dapur	<input type="text"/>
Mesin Cuci	<input type="text"/>	Pembantu	<input type="text"/>
Cuci Gosok	<input type="text"/>	CCTV	<input type="text"/>
Kolam Renang	<input type="text"/>		
Parkir Motor	<input type="text"/>	Parkir Mobil	<input type="text"/>
Restaurant	<input type="text"/>	Warnet	<input type="text"/>
Mall	<input type="text"/>	Apotek	<input type="text"/>
Bank	<input type="text"/>	Supermarket	<input type="text"/>
Kendaraan Umum	<input type="text"/>		

**Gambar 4.18** Perancangan Halaman Pencarian Data Kos

#### 14. Halaman Daftar Pencarian Data Kos

Pada halaman ini, pemilik dapat melihat daftar pencarian data kos yang dimiliki dan juga memilih nama kos yang ingin dilihat. Halaman daftar pencarian data kos pada Gambar 4.43 dan deskripsi objek pada Tabel 4.72.

Deskripsi Isi : Pada halaman ini ditampilkan daftar pencarian data kos.

Desain Tampilan: Warna background kombinasi abu – abu serta putih dan font berwarna hitam.

Jenis : Form tampilan (*display*)

#### 15. Halaman Detail Informasi Pencarian Data Kos

Pada halaman ini, pencari dapat melihat data kos, fasilitas yang dimiliki, dan fasilitas umum terdekat dengan kos. Halaman ini akan dibagi menjadi tiga layout yang dapat digeser. Halaman detail informasi pencarian data kos informasi utama pada Gambar 4.44 dan deskripsi objek pada Tabel 4.73.

Deskripsi Isi : Menampilkan nama, alamat, kecamatan, kelurahan, biaya sewa, jenis kelamin penghuni dan penghuni tiap kamar.

Desain Tampilan: Warna background kombinasi abu – abu serta putih, warna tombol kuning dan font berwarna hitam.

Jenis : Form tampilan (*display*)

**Tabel 4.18** Deskripsi Objek Perancangan Halaman Pencarian Data Kos

Objek	Jenis	Keterangan
Text Jarak	<i>Edit Text</i>	<i>Field</i> untuk memasukkan bobot jarak
Text Biaya	<i>Edit Text</i>	<i>Field</i> untuk memasukkan bobot biaya
Text Jenis Kelamin	<i>Edit Text</i>	<i>Field</i> untuk memasukkan bobot jenis kelamin penghuni kos
Text Jmlh Penghuni	<i>Edit Text</i>	<i>Field</i> untuk memasukkan bobot jumlah penghuni tiap kamar
Text Meja Kursi	<i>Edit Text</i>	<i>Field</i> untuk memasukkan bobot meja kursi
Text Lemari	<i>Edit Text</i>	<i>Field</i> untuk memasukkan bobot lemari
Text Tempat Tidur	<i>Edit Text</i>	<i>Field</i> untuk memasukkan bobot tempat tidur
Text Kipas Angin	<i>Edit Text</i>	<i>Field</i> untuk memasukkan bobot kipas angin
Text AC	<i>Edit Text</i>	<i>Field</i> untuk memasukkan bobot AC
Text KM Luar	<i>Edit Text</i>	<i>Field</i> untuk memasukkan bobot kamar mandi luar
Text Internet	<i>Edit Text</i>	<i>Field</i> untuk memasukkan bobot internet
Text Ruang Tamu	<i>Edit Text</i>	<i>Field</i> untuk memasukkan bobot ruang tamu
Text Dapur	<i>Edit Text</i>	<i>Field</i> untuk memasukkan bobot dapur
Text KolamRenang	<i>Edit Text</i>	<i>Field</i> untuk memasukkan bobot kolam renang
Text Tv	<i>Edit Text</i>	<i>Field</i> untuk memasukkan bobot Tv
Text Cuci Gosok	<i>Edit Text</i>	<i>Field</i> untuk memasukkan bobot cuci gosok
Text Mesin cuci	<i>Edit Text</i>	<i>Field</i> untuk memasukkan bobot mesin cuci
Text Dispenser	<i>Edit Text</i>	<i>Field</i> untuk memasukkan bobot dispenser
Text Pembantu	<i>Edit Text</i>	<i>Field</i> untuk memasukkan bobot pembantu
Text Kulkas	<i>Edit Text</i>	<i>Field</i> untuk memasukkan bobot kulkas
Text Cctv	<i>Edit Text</i>	<i>Field</i> untuk memasukkan bobot CCTV
Text Parkir Motor	<i>Edit Text</i>	<i>Field</i> untuk memasukkan bobot parkir motor
Text Parkir Mobil	<i>Edit Text</i>	<i>Field</i> untuk memasukkan bobot parkir mobil
Text Restaurant	<i>Edit Text</i>	<i>Field</i> untuk memasukkan bobot restaurant
Text Warnet	<i>Edit Text</i>	<i>Field</i> untuk memasukkan bobot warnet
Text Mall	<i>Edit Text</i>	<i>Field</i> untuk memasukkan bobot mall
Text Apotek	<i>Edit Text</i>	<i>Field</i> untuk memasukkan bobot apotek
Text Bank	<i>Edit Text</i>	<i>Field</i> untuk memasukkan bobot bank
Text Kendaraan Umum	<i>Edit Text</i>	<i>Field</i> untuk memasukkan bobot kendaraan umum
Text Supermarket	<i>Edit Text</i>	<i>Field</i> untuk memasukkan bobot supermarket
btnCari	<i>Button</i>	Mencari data kos yang sesuai dengan kriteria dan prioritas yang telah dimasukkan
BtnClear	<i>Button</i>	Mengulang bobot menjadi satu kembali

Halaman Cari – Daftar Kos

Nama Kos  
Alamat Kos

Nama Kos  
Alamat Kos

Nama Kos  
Alamat Kos

Nama Kos  
Alamat Kos

Nama Kos  
Alamat Kos

Nama Kos  
Alamat Kos

**Gambar 4.19** Perancangan Halaman Daftar Pencarian Data Kos

**Tabel 4.19** Deskripsi Objek Perancangan Halaman Daftar Pencarian DataKos

Objek	Jenis	Keterangan
TextNama Kos	<i>Text Area</i>	<i>Field</i> untuk menampilkan nama kos
TextAlamat Kos	<i>Text Area</i>	<i>Field</i> untuk menampilkan alamat kos

Halaman Cari – Detail Info Kos

Nama Kos  
Alamat Kos  
Nomor Telp Pemilik Kos  
Nama Pemilik Kos

« Kelurahan Kelurahan Kos »

Kecamatan Kecamatan Kos

Biaya Sewa Biaya Sewa Kos

Jenis Kelamin Jenis Kelamin Penghuni Kos

Jumlah Penghuni Jumlah Penghuni Per Kamar

PETA RIWAYAT

**Gambar 4.20** Perancangan Perancangan Halaman Detail Informasi Pencarian Data Kos Informasi Utama



**Tabel 4.20** Deskripsi Objek Perancangan Halaman Detail Informasi Pencarian Data Kos Informasi Utama

Objek	Jenis	Keterangan
Text Nama	<i>Text Area</i>	<i>Field</i> untuk menampilkan nama kos
Text Alamat	<i>Text Area</i>	<i>Field</i> untuk menampilkan alamat kos
Text Telp Pemilik Kos	<i>Text Area</i>	<i>Field</i> untuk menampilkan telp pemilik kos
TextNamaPemilik Kos	<i>Text Area</i>	<i>Field</i> untuk menampilkan nama pemilik kos
Text Kecamatan	<i>Text Area</i>	<i>Field</i> untuk menampilkan kecamatan
Text Kelurahan	<i>Text Area</i>	<i>Field</i> untuk menampilkan kelurahan
TextBiaya Sewa	<i>Text Area</i>	<i>Field</i> untuk menampilkan biaya sewa kos
Text Jenis Kelamin	<i>Text Area</i>	<i>Field</i> untuk menampilkan jenis kelamin penghuni kos
Text Jumlah Penghuni	<i>Text Area</i>	<i>Field</i> untuk menampilkan jumlah penghuni perkamar
btnPeta	<i>Button</i>	Menuju peta letak kos
btnRiwayat	<i>Button</i>	Memasukan data pada riwayat pencarian

Halaman detail informasi pencarian data kos fasilitas kos dapat dilihat pada Gambar 4.45 dan deskripsi objek dapat dilihat pada Tabel 4.74.

Deskripsi Isi : Halaman ini menampilkan fasilitas meja dan kursi, lemari, tempat tidur, kipas angin, AC, kamar mandi luar, internet, ruang tamu, dapur, kolam renang, Tv, cuci gosok, mesin cuci, dispenser, pembantu, kulkas, cctv, jumlah parkir motor, dan jumlah parkir mobil.

Desain Tampilan: Warna background kombinasi abu – abu serta putih, warna tombol kuning dan font berwarna hitam.

Jenis : Form tampilan (*display*)

Halaman Cari - Detail Info Kos

**Fasilitas Kos**

☒ Meja Kursi    ☒ Lemari

☒ Tempat Tidur    ☒ Kipas Angin

☒ Ac    ☒ Kamar Mandi Luar

☒ Internet    ☒ Ruang Tamu

☒ Dapur    ☒ Kolam Renang    >>

☒ Tv    ☒ Cuci Gosok

☒ Mesin Cuci    ☒ Dispenser

☒ Pembantu    ☒ Kulkas

☒ CCTV

Jumlah Parkir Motor    Jumlah Parkir Motor

**Gambar 4.21** Perancangan Halaman Detail Informasi Pencarian Data Kos Fasilitas Kos

Halaman detail informasi pencarian data kos fasilitas umum terdekat dapat dilihat pada Gambar 4.46 dan deskripsi objek dapat dilihat pada Tabel 4.75.

Deskripsi Isi : Halaman ini menampilkan jarak fasilitas umum terdekat seperti restaurant, warnet, mall, apotek, bank, kendaraan umum dan supermarket.

Desain Tampilan: Warna background kombinasi abu – abu serta putih, warna tombol kuning dan font berwarna hitam.

Jenis : Form tampilan (*display*)

**Tabel 4.21** Deskripsi Objek Perancangan Halaman Halaman Detail Informasi Pencarian Data Kos Fasilitas Kos

Objek	Jenis	Keterangan
cbMeja Kursi	<i>Check Box</i>	Check Box untuk data meja kursi
cbLemari	<i>Check Box</i>	Check Box untuk data lemari
cbTempat Tidur	<i>Check Box</i>	Check Box untuk data tempat tidur
cbKipas Angin	<i>Check Box</i>	Check Box untuk data kipas angina
cbAc	<i>Check Box</i>	Check Box untuk data AC
cbKamar Mandi Luar	<i>Check Box</i>	Check Box untuk data kamar mandi luar
cbInternet	<i>Check Box</i>	Check Box untuk data internet
cbRuang Tamu	<i>Check Box</i>	Check Box untuk data ruang tamu
cbDapur	<i>Check Box</i>	Check Box untuk data dapur
cbKolamRenang	<i>Check Box</i>	Check Box untuk data kolam renang
cbTv	<i>Check Box</i>	Check Box untuk data Tv
cbCuci Gosok	<i>Check Box</i>	Check Box untuk data cuci gosok
cbMesin cuci	<i>Check Box</i>	Check Box untuk data mesin cuci
cbDispenser	<i>Check Box</i>	Check Box untuk data dispenser
cbPembantu	<i>Check Box</i>	Check Box untuk data pembantu
cbKulkas	<i>Check Box</i>	Check Box untuk data kulkas
cbCctv	<i>Check Box</i>	Check Box untuk data CCTV
Text Parkir Motor	<i>Text Area</i>	<i>Field</i> untuk menampilkan jumlah parkir motor
Text Parkir Mobil	<i>Text Area</i>	<i>Field</i> untuk menampilkan jumlah parkir mobil

Halaman Riwayat Pencarian – Detail Info Kos

Jarak Kos dengan Fasilitas Umum

Restaurant	Jarak Restaurant	km
Wamet	Jarak Wamet	km
Mall	Jarak Mall	km
Apotek	Jarak Apotek	km
Bank	Jarak Bank	km
Kendaraan Umum	Jarak Kendaraan Umum	km
Supermarket	Jarak Supermarket	km

KEMBALI

**Gambar 4.22** Perancangan Halaman Detail Informasi Pencarian Data Kos Fasilitas Umum Terdekat

**Tabel 4.22** Deskripsi Objek Perancangan Halaman Detail Informasi Pencarian Data Kos Fasilitas Umum Terdekat

Objek	Jenis	Keterangan
Text Restaurant	<i>Text Area</i>	<i>Field</i> untuk menampilkan jarak kos dengan restaurant
Text Warnet	<i>Text Area</i>	<i>Field</i> untuk menampilkan jarak kos dengan warnet
Text Mall	<i>Text Area</i>	<i>Field</i> untuk menampilkan jarak kos dengan mall
Text Apotek	<i>Text Area</i>	<i>Field</i> untuk menampilkan jarak kos dengan apotek
Text Bank	<i>Text Area</i>	<i>Field</i> untuk menampilkan jarak kos dengan bank
Text Kendaraan Umum	<i>Text Area</i>	<i>Field</i> untuk menampilkan jarak kos dengan kendaraan umum
Text Supermarket	<i>Text Area</i>	<i>Field</i> untuk menampilkan jarak kos dengan supermarket
btnKembali	<i>Button</i>	Kembali ke daftar kos

## 16. Halaman Daftar Riwayat Pencarian Data Kos

Pada halaman ini, pemilik dapat melihat daftar riwayat pencarian data kos yang telah disimpan. Halaman Daftar Riwayat Pencarian Data Kos dapat dilihat pada Gambar 4.47 dan Deskripsi Objek dapat dilihat pada Tabel 4.76.

Deskripsi Isi : Pada halaman ini ditampilkan daftar pencarian data kos.

Desain Tampilan: Warna background kombinasi abu – abu serta putih dan font berwarna hitam.

Jenis : Form tampilan (*display*)

**Gambar 4.23** Perancangan **Daftar Riwayat Pencarian Data Kos**

**Tabel 4.23** Deskripsi Objek Perancangan Halaman Daftar Kos

Objek	Jenis	Keterangan
TextNama Kos	<i>Text Area</i>	<i>Field</i> untuk menampilkan nama kos
TextAlamat Kos	<i>Text Area</i>	<i>Field</i> untuk menampilkan alamat kos
btnHapus	<i>Button</i>	Menghapus data kos dari daftara riwayat pencarian

#### 17. Halaman Detail Informasi Riwayat Pencarian Data Kos

Pada halaman ini, pencari dapat melihat data kos, fasilitas yang dimiliki, dan fasilitas umum terdekat dengan kos. Halaman ini akan dibagi menjadi tiga layout yang dapat digeser. Halaman detail informasi riwayat pencarian data kos informasi utama dapat dilihat pada Gambar 4.48 dan deskripsi objek dapat dilihat pada Tabel 4.77.

Deskripsi Isi : Halaman ini akan melampirkan nama kos, alamat kos, kecamatan, kelurahan, biaya sewa kos, jenis kelamin penghuni kos, dan jumlah penghuni kos tiap kamar.

Desain Tampilan: Warna background kombinasi abu – abu serta putih, warna tombol kuning dan font berwarna hitam.

Jenis : Form tampilan (*display*)

Halaman Riwayat Pencarian – Detail Info Kos

Nama Kos  
Alamat Kos  
Nomor Telp Pemilik Kos  
Nama Pemilik Kos

<< Kelurahan Kelurahan Kos >>

Kecamatan Kecamatan Kos

Biaya Sewa Biaya Sewa Kos

Jenis Kelamin Jenis Kelamin Penghuni Kos

Jumlah Penghuni Jumlah Penghuni Per Kamar

PETA

**Gambar 4.24** Perancangan Perancangan Halaman Detail Informasi Riwayat Pencarian Data Kos Informasi Utama

**Tabel 4.24** Deskripsi Objek Perancangan Halaman Detail Informasi Riwayat Pencarian Data Kos Informasi Utama

Objek	Jenis	Keterangan
Text Nama	<i>Text Area</i>	<i>Field</i> untuk menampilkan nama kos
Text Alamat	<i>Text Area</i>	<i>Field</i> untuk menampilkan alamat kos
Text Telp Pemilik Kos	<i>Text Area</i>	<i>Field</i> untuk menampilkan telp pemilik kos
TextNamaPemilik Kos	<i>Text Area</i>	<i>Field</i> untuk menampilkan nama pemilik kos
Text Kecamatan	<i>Text Area</i>	<i>Field</i> untuk menampilkan kecamatan
Text Kelurahan	<i>Text Area</i>	<i>Field</i> untuk menampilkan kelurahan
TextBiaya Sewa	<i>Text Area</i>	<i>Field</i> untuk menampilkan biaya sewa kos
Text Jenis Kelamin	<i>Text Area</i>	<i>Field</i> untuk menampilkan jenis kelamin penghuni kos
Text Jumlah Penghuni	<i>Text Area</i>	<i>Field</i> untuk menampilkan jumlah penghuni perkamar
btnPeta	<i>Button</i>	Menuju peta letak kos

Halaman detail informasi riwayat pencarian data kos fasilitas kos dapat dilihat pada Gambar 4.49 dan deskripsi objek dapat dilihat pada Tabel 4.78.

Deskripsi Isi : Halaman ini menampilkan fasilitas meja dan kursi, lemari, tempat tidur, kipas angin, AC, kamar mandi luar, internet, ruang tamu, dapur, kolam renang, Tv, cuci gosok, mesin cuci, dispenser, pembantu, kulkas, cctv, jumlah parkir motor, dan jumlah parkir mobil.

Desain Tampilan: Warna background kombinasi abu – abu serta putih, warna tombol kuning dan font berwarna hitam.

Jenis : Form tampilan (*display*)



Halaman Riwayat Pencarian – Detail Info Kos

**Fasilitas Kos**

<input checked="" type="checkbox"/> Meja Kursi	<input checked="" type="checkbox"/> Lemari
<input checked="" type="checkbox"/> Tempat Tidur	<input checked="" type="checkbox"/> Kipas Angin
<input checked="" type="checkbox"/> Ac	<input checked="" type="checkbox"/> Kamar Mandi Luar
<input checked="" type="checkbox"/> Internet	<input checked="" type="checkbox"/> Ruang Tamu
<input checked="" type="checkbox"/> Dapur	<input checked="" type="checkbox"/> Kolam Renang
<input checked="" type="checkbox"/> Tv	<input checked="" type="checkbox"/> Cuci Gosok
<input checked="" type="checkbox"/> Mesin Cuci	<input checked="" type="checkbox"/> Dispenser
<input checked="" type="checkbox"/> Pembantu	<input checked="" type="checkbox"/> Kulkas
<input checked="" type="checkbox"/> CCTV	

Jumlah Parkir Motor      Jumlah Parkir Mobil

**Gambar 4.25** Perancangan Halaman Detail Informasi Riwayat Pencarian Data Kos Fasilitas Kos

**Tabel 4.25** Deskripsi Objek Perancangan Halaman Halaman Detail Informasi Riwayat Pencarian Data Kos Fasilitas Kos

Objek	Jenis	Keterangan
cbMeja Kursi	Check Box	Check Box untuk data meja kursi
cbLemari	Check Box	Check Box untuk data lemari
cbTempat Tidur	Check Box	Check Box untuk data tempat tidur
cbKipas Angin	Check Box	Check Box untuk data kipas angina
cbAc	Check Box	Check Box untuk data AC
cbKamar Mandi Luar	Check Box	Check Box untuk data kamar mandi luar
cbInternet	Check Box	Check Box untuk data internet
cbRuang Tamu	Check Box	Check Box untuk data ruang tamu
cbDapur	Check Box	Check Box untuk data dapur
cbKolam Renang	Check Box	Check Box untuk data kolam renang
cbTv	Check Box	Check Box untuk data Tv
cbCuci Gosok	Check Box	Check Box untuk data cuci gosok
cbMesin cuci	Check Box	Check Box untuk data mesin cuci
cbDispenser	Check Box	Check Box untuk data dispenser
cbPembantu	Check Box	Check Box untuk data pembantu
cbKulkas	Check Box	Check Box untuk data kulkas
cbCctv	Check Box	Check Box untuk data CCTV
Text Parkir Motor	Text Area	Field untuk menampilkan jumlah parkir motor
Text Parkir Mobil	Text Area	Field untuk menampilkan jumlah parkir mobil

Halaman detail informasi riwayat pencarian data kos fasilitas umum terdekat pada Gambar 4.50 dan deskripsi objek pada Tabel 4.79.

Deskripsi Isi : Halaman ini menampilkan jarak fasilitas umum terdekat dengan kos seperti restaurant, warnet, mall, apotek, bank, kendaraan umum dan supermarket.

Desain Tampilan: Warna background kombinasi abu – abu serta putih, warna tombol kuning dan font berwarna hitam.

Jenis : Form tampilan (*display*)

The screenshot shows a web interface titled 'Halaman Riwayat Pencarian – Detail Info Kos'. Below the title is a section 'Jarak Kos dengan Fasilitas Umum'. It contains a list of facilities with corresponding distance fields: Restaurant, Warnet, Mall, Apotek, Bank, Kendaraan Umum, and Supermarket. Each facility name is followed by 'Jarak [Facility] km'. Navigation arrows '<<' and '>>' are on the left and right of the list. A 'KEMBALI' button is at the bottom right. A large, faint watermark of a university seal is visible in the background.

**Gambar 4.26** Perancangan Halaman Detail Informasi Riwayat Pencarian Data Kos Fasilitas Umum Terdekat

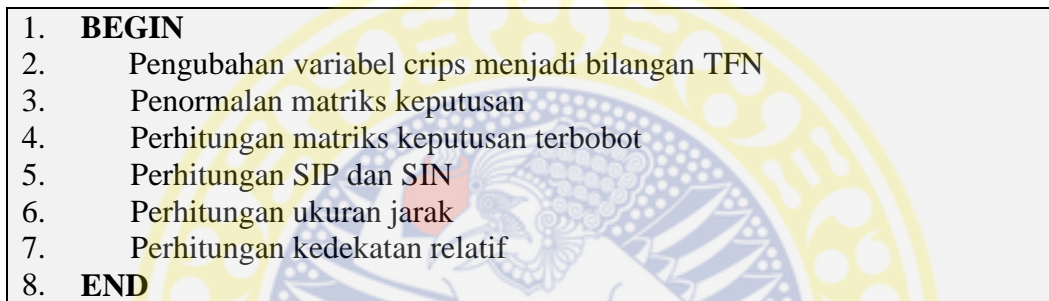
**Tabel 4.26** Deskripsi Objek Perancangan Halaman Detail Informasi Riwayat Pencarian Data Kos Fasilitas Umum Terdekat

Objek	Jenis	Keterangan
Text Restaurant	<i>Text Area</i>	<i>Field</i> untuk menampilkan jarak kos dengan restaurant
Text Warnet	<i>Text Area</i>	<i>Field</i> untuk menampilkan jarak kos dengan warnet
Text Mall	<i>Text Area</i>	<i>Field</i> untuk menampilkan jarak kos dengan mall
Text Apotek	<i>Text Area</i>	<i>Field</i> untuk menampilkan jarak kos dengan apotek
Text Bank	<i>Text Area</i>	<i>Field</i> untuk menampilkan jarak kos dengan bank
Text Kendaraan Umum	<i>Text Area</i>	<i>Field</i> untuk menampilkan jarak kos dengan kendaraan umum
Text Supermarket	<i>Text Area</i>	<i>Field</i> untuk menampilkan jarak kos dengan supermarket
btnKembali	<i>Button</i>	Kembali ke daftar kos

Hasil Implementasi sistem berdaraskan perancangan telah dilakukan dengan menggunakan *use case diagram* dan *activity diagram*. Aplikasi mobile sistem pendukung keputusan rekomendasi kos di Kota Surabaya akan diimplementasikan menggunakan Bahasa pemrograman Java dan berbasis Android.

#### 2.4.1 Implementasi Algoritma Fuzzy TOPSIS

Algoritma Umum dari Sistem Pendukung Keputusan Pemilihan Kos Daerah Surabaya dengan Metode Fuzzy TOPSIS dapat dilihat pada Gambar 4.51.



**Gambar 4.27** Algoritma Umum SPK Pemilihan Kos Daerah Surabaya

Algoritma Umum dimulai dengan mengubah variabel crips data kos menjadi bilangan TFN sesuai dengan fungsi keanggotaan masing-masing kategori. Setelah menjadi bilangan TFN dimulailah perhitungan yang pertama yaitu menormalkan matriks keputusan. Kemudian dikalikan dengan bobot yang telah ditentukan oleh pengguna berdasarkan urutan prioritas yang dimiliki. Mencari mean merupakan proses selanjutnya untuk mencari nilai terbesar dan terkecil dari semua alternatif disetiap kriteria. Mean terbesar sama dengan SIP sedangkan mean terkecil sama dengan SIN. Kemudian mencari nilai jarak dari semua alternatif setiap kriteria dengan SIP setiap kriteria dan dari semua alternatif setiap kriteria dengan SIN setiap kriteria. Perhitungan terakhir adalah perhitungan kedekatan relatif yang akan dilanjutkan dengan mengurutkan nilai kedekatan relatif terbesar hingga terkecil. Urutan kedekatan relatif adalah hasil rekomendasi kos terbaik sampe terburuk dari Sistem Pendukung Keputusan Pemilihan Kos Daerah Surabaya berdasarkan kriteria dan prioritas pengguna.

Langkah pertama merupakan perubahan variabel dari variabel *crisp* menjadi variabel TFN. Algoritma perubahan variabel dapat dilihat pada Gambar 4.52. Ditentukan batasan keanggotaan himpunan *fuzzy* setiap kriteria terlebih dahulu. Apabila variabel *crip* lebih dari sama

dengan c, maka kategori pertama memiliki nilai 0. Apabila variabel *crisp* lebih dari sama dengan b dan variabel *crisp* kurang dari c maka nilai kategori pertama adalah variabel *crisp* dikurangi b dibagi dengan c dikurangi b.

```

1.  Fungsi varLinguistik (Parameter data)
2.  BEGIN
3.      double kategori1, kategori2, kategori3
4.      Array batas // mengambil batasan himpunan fuzzy setiap kriteria
5.      IF (data >= batas[2]) THEN
6.          kategori1 = 0
7.      ELSE ((data >= batas[1])&&(data <= batas[2]))
8.          kategori1 = (data - batas[1])/(batas[2] - batas[1])
9.      ENDIF
10.     IF ((data <= batas[3])||(data >= batas[5])) THEN
11.         kategori2 = 0
12.     ELSE ((data > batas[3])||(data <= batas[4])) THEN
13.         kategori2 = (data - batas[3])/(batas[4] - batas[3])
14.     ELSE ((data > batas[4])||(data <= batas[5])) THEN
15.         kategori2 = ((-data) + batas[5])/(batas[5] - batas[4])
16.     ENDIF
17.     IF (data <= batas[6]) THEN
18.         kategori3 = 0
19.     ELSE ((data >= batas[6])&&(data <= batas[7])) THEN
20.         kategori3 = (data - batas[6])/(batas[7] - batas[6])
21.     ELSE (data >= batas[7])
22.         kategori3 = 1
23.     ENDIF
24.     double max = kategori1
25.     Array tfn // menampung hasil TFN
26.     tfn[0] = 0.01, tfn[1] = 0.01, tfn[2] = 4
27.     IF (max < kategori2) THEN
28.         max = kategori2
29.     tfn[0] = 2
30.     tfn[1] = 4.5
31.     tfn[2] = 7
32.     ELSEIF (max < kategori3) THEN
33.         max = kategori3
34.     tfn[0] = 5
35.     tfn[1] = 10
36.     tfn[2] = 10
37.     ENDIFEND
38.     RETURN tfn
39.
40. END

```

**Gambar 4.28** Algoritma Pengubahan Variable *Crips* Menjadi Bilangan *Triangular Fuzzy Number*



Nilai untuk kategori kedua memiliki tiga kemungkinan. Kemungkinan pertama apabila variabel *crisp* kurang dari sama dengan a atau variabel *crisp* lebih dari sama dengan c, maka kategori kedua memiliki nilai 0. Kemungkinan kedua apabila variabel *crisp* lebih dari a atau variabel *crisp* kurang dari sama dengan b, maka kategori kedua memiliki nilai variabel *crisp* dikurangi a dibagi dengan b dikurangi a. Kemungkinan ketiga apabila variabel *crisp* lebih dari b atau variabel *crisp* kurang dari sama dengan c, maka kategori kedua memiliki nilai minus variabel *crisp* ditambah c dibagi dengan c dikurangi b.

Nilai untuk kategori kedua memiliki tiga kemungkinan. Kemungkinan pertama apabila variabel *crisp* kurang dari sama dengan a, maka kategori ketiga memiliki nilai 0. Kemungkinan kedua apabila variabel *crisp* lebih dari a dan variabel *crisp* kurang dari sama dengan b, maka kategori ketiga memiliki nilai variabel *crisp* dikurangi a dibagi dengan b dikurangi a. Kemungkinan ketiga apabila variabel *crisp* lebih dari b, maka kategori ketiga memiliki nilai 1.

Langkah kedua merupakan perhitungan menormalkan matriks keputusan. Pada perhitungan ini terdapat empat kategori, yaitu atribut biaya variabel *crisp*, atribut biaya variabel *fuzzy*, atribut keuntungan variabel *crisp*, dan atribut keuntungan variabel *fuzzy*.

Algoritma menormalkan matriks keputusan dengan atribut biaya dengan bilang *fuzzy* dapat dilihat pada Gambar 4.53. Didefinisikan terlebih dahulu array hasil untuk menampung hasil dari menormalkan matriks keputusan dan array tfn untuk menampung nilai x yang terkecil. Array tfn pertama kali akan diisi oleh nilai x kriteria baris pertama. Selanjutnya akan diulang sebanyak baris dari kriteria untuk mencari nilai x terkecil dari seluruh baris kriteria. Ketika a kriteria lebih kecil dari a terkecil maka a terkecil digantikan dengan a kriteria. Ketika b kriteria lebih kecil dari b terkecil maka b terkecil digantikan dengan b kriteria. Ketika c kriteria lebih kecil dari c terkecil maka c terkecil digantikan dengan c kriteria.

Diulang sebanyak baris dari kriteria untuk menormalkan matriks keputusan dengan cara membagi nilai terkecil dengan nilai kriteria. Hasil a sama dengan a terkecil dibagi dengan c kriteria. Hasil b sama dengan b terkecil dibagi dengan b kriteria. Hasil c sama dengan c terkecil dibagi dengan a kriteria.

Algoritma menormalkan matriks keputusan dengan atribut keuntungan dengan bilang *fuzzy* dapat dilihat pada Gambar 4.54. Didefinisikan terlebih dahulu array hasil untuk menampung hasil dari menormalkan matriks keputusan dan array tfn untuk menampung nilai a,b, dan c yang terbesar Array tfn pertama kali akan diisi oleh nilai a,b, dan c kriteria baris pertama.



Selanjutnya akan diulang sebanyak baris dari kriteria untuk mencari nilai nilai a,b, dan c terbesar dari seluruh baris kriteria. Ketika a kriteria lebih besar dari a terbesar maka a terbesar digantikan dengan a kriteria. Ketika b kriteria lebih besar dari b terbesar maka b terbesar digantikan dengan b kriteria. Ketika c kriteria lebih besar dari c terbesar maka c terbesar digantikan dgn c kriteria.

```

1.  Fungsi langkahSatuBiayaFuzzy (Parameter kriteria)
2.  BEGIN
3.      Array hasil          // Array dua dimensi untuk menampung
                             hasil menormalkan matriks keputusan
4.      Array tfn_min        // Array satu dimensi untuk menampung TFN
                             paling kecil
5.      tfn_min[0] = kriteria[0][0]
6.      tfn_min[1] = kriteria[0][1]
7.      tfn_min[2] = kriteria[0][2]
8.      FOR i ← 1 TO kriteria.row
9.          IF (kriteria[i][0] < tfn_min[0]) THEN
10.             tfn_min[0] = kriteria[i][0]
11.          ELSE (kriteria[i][1] < tfn_min[1]) THEN
12.             tfn_min[1] = kriteria[i][1]
13.          ELSE (kriteria[i][2] < tfn_min[2]) THEN
14.             tfn_min[2] = kriteria[i][2]
15.          ENDIF
16.      ENDFOR
17.      FOR i ← 0 TO kriteria.row
18.          hasil[i][0] = tfn_min[0] / kriteria[i][2]
19.          hasil[i][1] = tfn_min[1] / kriteria[i][1]
20.          hasil[i][2] = tfn_min[2] / kriteria[i][0]
21.      ENDFOR
22.      RETURN hasil
23.  END

```

**Gambar 4.29** Algoritma Penormalan Matriks Keputusan Atribut Biaya Bilangan Fuzzy

Diulang sebanyak baris dari kriteria untuk menormalkan matriks keputusan dengan cara membagi nilai kriteria dengan nilai terbesar. Hasil a sama dengan a kriteria dibagi dengan c terbesar. Hasil b sama dengan b kriteria dibagi dengan b terbesar. Hasil c sama dengan c kriteria dibagi dengan a terbesar.

Algoritma menormalkan matriks keputusan dengan atribut keuntungan dengan bilangan *crisp* dapat dilihat pada Gambar 4.55. Didefinisikan terlebih dahulu array hasil untuk menampung hasil dari menormalkan matriks keputusan, tfn\_max untuk menampung nilai kriteria yang terbesar dan hitung untuk menampung hasil normalisasi setiap kriteria. Tfn\_max pertama kali akan diisi oleh nilai kriteria baris pertama. Selanjutnya akan diulang sebanyak baris dari kriteria untuk mencari nilai terbesar dari seluruh baris kriteria. Ketika nilai kriteria lebih besar dari nilai tfn\_max maka nilai tfn\_max digantikan dengan nilai kriteria. Diulang sebanyak baris dari kriteria untuk menormalkan matriks keputusan. Nilai hitung tiap kriteria merupakan hasil membagi nilai

kriteria dengan nilai tfn\_max. Hasil a, b dan c sama dengan nilai hitung.

```

1. Fungsi langkahSatuKentunganFuzzy (Parameter kriteria)
2. BEGIN
3.   Array hasil           // Array dua dimensi untuk menampung
                           hasil menormalkan matriks keputusan
4.   Array tfn_max         // Array satu dimensi untuk menampung TFN
                           paling besar
5.   tfn_max[0] = kriteria[0][0]
6.   tfn_max[1] = kriteria[0][1]
7.   tfn_max[2] = kriteria[0][2]
8.   FOR i ← 1 TO kriteria.row
9.     IF (kriteria[i][0] > tfn_max[0]) THEN
10.      tfn_max[0] = kriteria[i][0]
11.     ELSE (kriteria[i][1] > tfn_max[1]) THEN
12.      tfn_max[1] = kriteria[i][1]
13.     ELSE (kriteria[i][2] > tfn_max[2]) THEN
14.      tfn_max[2] = kriteria[i][2]
15.     ENDIF
16.   ENDFOR
17.   FOR i ← 0 TO kriteria.row
18.     hasil[i][0] = kriteria[i][0] / tfn_max[2]
19.     hasil[i][1] = kriteria[i][1] / tfn_max[1]
20.     hasil[i][2] = kriteria[i][2] / tfn_max[0]
21.   ENDFOR
22.   RETURN hasil
23. END

```

**Gambar 4.30** Algoritma Penormalan Matriks Keputusan Atribut Keuntungan Bilangan Fuzzy

```

1. Fungsi langkahSatuKentunganCrips (Parameter kriteria)
2. BEGIN
3.   Array hasil           //Array dua dimensi untuk menampung hasil
                           menormalkan matriks keputusan
4.   double tfn_max         //bilangan untuk menampung nilai
                           kriteria paling besar
5.   double hitung          //bilangan untuk menampung hasil
                           normalisasi tiap kriteria
6.   tfn_max = kriteria[0]
7.   FOR i ← 1 TO kriteria.row
8.     IF (kriteria[i] > tfn_max) THEN
9.      tfn_max = kriteria[i]
10.    ENDIF
11.  ENDFOR
12.  FOR i ← 0 TO kriteria.row
13.    hitung = kriteria[i] / tfn_max
14.    hasil[i][0] = hitung
15.    hasil[i][1] = hitung
16.    hasil[i][2] = hitung
17.  ENDFOR
18.  RETURN hasil
19. END

```

**Gambar 4.31** Algoritma Penormalan Matriks Keputusan Atribut Keuntungan Bilangan *Crips*

Langkah ketiga merupakan perhitungan matriks keputusan terbobot. Sebelum dilakukan perhitungan matriks keputusan terbobot diperlukan bobot setiap kriteria terlebih dahulu.

Algoritma perhitungan bobot setiap kriteria dapat dilihat pada Gambar 4.56. Didefinisikan terlebih dahulu Array bobot untuk menampung nilai bobot. Nilai x adalah hasil pembagian urutan prioritas pengguna dengan 6. Sedangkan index adalah hasil pengurangan 6 dengan hasil sisa pembagian urutan prioritas dengan 6. Nilai x dan index dihitung untuk mendapatkan hasil bila melebihi 6 maka sisa dari pembagian harus digabung dengan sisa pembagian karena prioritas dianggap hanya sedikit perbedaannya.

```

1.  Fungsi menghitungBobot (Parameter w, spare)
2.  BEGIN
3.      Array bobot//array satu dimensi menampung bobot
      String R = 0.0,0.1,0.3
      String MR = 0.1,0.3,0.5
      String M = 0.3,0.5,0.7
      String MT = 0.5,0.7,0.9
      String T = 0.7,0.9,1.0
      String ST = 0.9,1.0,1.0
9.      int x = urutan / 6;
10.     int index = 6 - (urutan % 6);
11.     int i = 1;
12.     WHILE (x >= 0)
13.         WHILE (i < 7)
14.             IF (i <= index) THEN
15.                 FOR j ← 0 TO x
16.                     IF (i = 1) THEN
17.                         bobot[index] = ST
18.                     ELSE (i = 2) THEN
19.                         bobot[index] = T
20.                     ELSE (i = 3) THEN
21.                         bobot[index] = MT
22.                     ELSE (i = 4) THEN
23.                         bobot[index] = M
24.                     ELSE (i = 5) THEN
25.                         bobot[index] = MR
26.                     ELSE (i = 6) THEN
27.                         bobot[index] = R
28.                     ENDIF
29.                 ENDFOR
30.             ELSE (i > index) THEN
31.                 FOR (int j = 0; j < x + 1; j++)
32.                     IF (i = 1) THEN
33.                         bobot[index] = ST
34.                     ELSE (i = 2) THEN
35.                         bobot[index] = T
36.                     ELSE (i = 3) THEN
37.                         bobot[index] = MT
38.

```

**Gambar 4.56** Algoritma Perhitungan Bobot Setiap Kriteria

```

31.         ELSE (i = 4) THEN
32.             bobot[index] = M
33.         ELSE (i = 5) THEN
34.             bobot[index] = MR
35.         ELSE (i = 6) THEN
36.             bobot[index] = R
37.         ENDIFENDFOR
38.     ENDIF
39. ENDWHILE
40. ENDWHILE
41. RETURN bobot
42. END

```

**Lanjutan Gambar 4.56** Algoritma Perhitungan Bobot Setiap Kriteria

Setelah mengetahui bobot setiap kriteria maka perhitungan matriks keputusan terbobot bias dilakukan. Algoritma perhitungan matriks keputusan terbobot dapat dilihat pada Gambar 4.57. Diulang sebanyak baris dari kriteria untuk matriks keputusan terbobot dengan cara mengali nilai kriteria dengan nilai bobot. Kriteria a sama dengan a kriteria dikali dengan a bobot. Kriteria b sama dengan b kriteria dikali dengan b bobot. Kategori c sama dengan c kriteria dikali dengan c bobot.

```

1. Fungsi langkahDua (Parameter kriteria, bobot)
2. BEGIN
3.   FOR i ← 0 TO kriteria.row
4.     kriteria[i][0] = kriteria[i][0] * bobot[0]
5.     kriteria[i][1] = kriteria[i][1] * bobot[1]
6.     kriteria[i][2] = kriteria[i][2] * bobot[2]
7.   ENDFOR
8.   RETURN kriteria
9. END

```

**Gambar 4.32** Algoritma Perhitungan Matriks Keputusan Terbobot

Langkah keempat merupakan perhitungan SIP dan SIN. Perhitungan SIP dan SIN sama dengan menghitung mean dari setiap kriteria, mean dengan nilai paling besar merupakan nilai SIP sedangkan mean dengan nilai paling kecil merupakan nilai SIN. Algoritma perhitungan SIP dan SIN pada Gambar 4.58.

Untuk algoritma perhitungan ini didefinisikan terlebih dahulu Array mean untuk menampung hasil mean setiap kriteria, Array data\_ke untuk menampung index mean paling besar dan index mean paling kecil, bilangan max untuk menampung hasil mean terbesar, dan bilangan min untuk menampung hasil mean terkecil. Diulang sebanyak baris dari kriteria untuk menghitung mean. Mean sama dengan pangkat dua a kriteria dikurangi pangkat dua b kriteria ditambah pangkat dua b kriteria ditambah pangkat dua c kriteria dikurangi a kriteria dikali b

kriteria ditambah b kriteria dikali c kriteria dibagi dengan tiga kali minus a kriteria dikurangi b kriteria ditambah b kriteria ditambah c.

```

1.  Fungsi langkahTiga (Parameter kriteria)
2.  BEGIN
3.      Array mean          // Array dua dimensi untuk menampung
                             hasil mean setiap alternatif
4.      Array data_ke       // Array satu dimensi untuk menampung
                             index tempat hasil mean paling besar
                             dan kecil
5.      Double max          // Bilangan untuk menampung hasil mean
                             terbesar
6.      Double min          // Bilangan untuk menampung hasil mean
                             terkecil
7.      FOR i ← 0 TO kriteria.row
8.          FOR i ← 0 TO kriteria.row
9.              mean[i] = ((kriteria[i][0]^2) - (kriteria[i][1]^2)) +
                           (kriteria[i][1]^2) + (kriteria[i][2]^2) -
                           (kriteria[i][0] * kriteria[i][1]) +
                           (kriteria[i][1] * kriteria[i][2])) / (3 *
                           (-kriteria[i][0] - kriteria[i][1] +
                           kriteria[i][1] + kriteria[i][2]))
10.          ENDFOR
11.          max = mean[0]
12.          min = mean[0]
13.          FOR i ← 1 TO kriteria.row
14.              IF (max < mean[i]) THEN
15.                  max = mean[i]
16.                  data_ke[0] = i
17.              ELSE IF (min < mean[i]) THEN
18.                  min = mean[i]
19.                  data_ke[1] = i
20.              ENDIF
21.          ENDFOR
22.          RETURN data_ke
23.      END

```

**Gambar 4.33** Algoritma Perhitungan SIP dan SIN

Nilai max dan min pertama kali akan diisi oleh mean baris pertama. Diulang sebanyak baris kriteria untuk mencari nilai terbesar dan terkecil dari mean. Apabila nilai max lebih kecil dari mean, maka nilai max sama dengan mean dan data\_ke baris pertama akan diisi oleh index mean. Apabila nilai min lebih kecil dari mean, maka nilai min sama dengan mean dan data\_ke baris kedua akan diisi oleh index mean.

Langkah kelima merupakan perhitungan jarak setiap alternatif terhadap SIP dan SIN setiap kriteria. Algoritma perhitungan jarak setiap alternatif terhadap SIP setiap kriteria dapat dilihat pada Gambar 4.59. Algoritma perhitungan jarak setiap alternatif terhadap SIN setiap kriteria dapat dilihat pada Gambar 4.60.

Perhitungan jarak setiap alternatif terhadap SIP setiap kriteria dimulai dengan mendefinisikan Array D\_max untuk menampung hasil jarak terhadap SIP dan nilai max .



Diulang sebanyak baris kriteria untuk menghitung D\_max. D\_max sama dengan 1 dikurangi a kriteria index max dikurangi c kriteria dibagi dengan minus c kriteria dikurangi b kriteria dikurangi b kriteria index max dikurangi a kriteria index max.

```

1. Fungsi langkahEmpatMax (Parameter kriteria, max)
2. BEGIN
3.   Array D_max    // Array satu dimensi untuk menampung hasil jarak
                     terhadap solusi ideal positif
4.   FOR i ← 0 TO kriteria.row
5.     D_max[i]= (1-((kriteria[max][0] – kriteria[i][2]) / ((-(kriteria[i][2]
6.               - (kriteria[i][1])) – (kriteria[max][1] -
7.               kriteria[max][0])))
8.   ENDFOR
9.   RETURN D_max
10. END

```

**Gambar 4.34** Algoritma Perhitungan Jarak terhadap SIP

```

1. Fungsi langkahEmpatMin (Parameter kriteria, min)
2. BEGIN
3.   Array D_min    // Array satu dimensi untuk menampung hasil jarak
                     terhadap solusi ideal negatif
4.   FOR i ← 0 TO kriteria.row
5.     D_min[i]= (1-((kriteria[min][2] – kriteria[min][0]) / ((kriteria[i][1]
6.               - kriteria[i][0]) + (kriteria[min][2] - kriteria[min][1])))
7.   ENDFOR
8.   RETURN D_min
9. END

```

**Gambar 4.35** Algoritma Perhitungan Jarak terhadap SIN

Pehitungan jarak setiap alternatif SIN setiap kriteria dimulai dengan mendefinisikan Array D\_min untuk menampung hasil jarak terhadap SIN dan nilai min. Diulang sebanyak baris kriteria untuk menghitung D\_min. D\_min sama dengan 1 dikurangi c kriteria index min dikurangi a kriteria dibagi dengan b kriteria dikurangi a kriteria ditambah c kriteria index min dikurangi b kriteria index min.

Langkah keenam merupakan perhitungan kedekatan relative. Algoritma perhitungan kedekatan relatif dapat dilihat pada Gambar 4.61. Untuk algoritma perhitungan ini didefinisikan terlebih dahulu Array identitas untuk menyimpan identitas kos, Array C untuk menampung nilai preferensi setiap alternatif, String move untuk menampung identitas kos yang akan dipindah sesuai urutan, dan bilangan max untuk menampung nilai terbesar dari nilai preferensi setiap alternatif.

```

1. Fungsi langkahLima (Parameter S_plus, S_min)
2. BEGIN
3.   Array identitas // array untuk menyimpan identitas kos
4.   Array C // Array satu dimensi untuk menampung hasil solusi
        ideal negative
5.   String move // kalimat untuk menampung idntitas kos yang akan
        di pindah
6.   double max // bilangan untuk menampung nilai terbesar dari C
7.   FOR i ← 0 TO S_plus.length
8.     C[i]= (S_min[i] / (S_min[i] + S_plus[i]))
9.   ENDFOR
10.  FOR j ← 1 TO C.length-1
11.    FOR i ← 0 TO C.length-1
12.      IF (C[i] > C[i + 1]) THEN
13.        max = C[i]
14.        move = identitas[i]
15.        C[i] = C[i + 1]
16.        identitas[i] = identitas[i+1]
17.        C[i + 1] = max
18.        Identitas[i + 1] = move
19.      ENDFOR
20.    RETURN D_min
21.  END

```

**Gambar 4.36** Algoritma Perhitungan Kedekatan Relatif

Nilai preferensi setiap alternatif diperoleh dengan membagi jarak alternatif terhadap SIN dengan jarak alternatif terhadap SIN ditambah jarak alternatif terhadap SIP. Kedekatan relative merupakan urutan dari nilai preferensi setiap alternatif dari nilai terbesar sampai nilai terkceil. Diulang sebanyak panjang nilai preferensi dikurangi dan kembali diulang sebanya panjang nilai preferensi dikurangi satu untuk mendapatkan urutan nilai preferensi setiap alternatif. Apabila nilai preferensi saat ini lebih besar dari nilai preferensi selanjutnya, maka max sama dengan nilai preferensi, move akan diisi oleh identitas kos saat ini, nilai preferensi saat ini diisi dengan nilai preferensi selanjutnya, identitas saat ini diisi dengan identitas kos selanjutnya, nilai prefrensi selanjutnya akan diisi oleh max, dan identitas kos selanjutnya diisi oleh move.

#### 2.4.2 Deskripsi Akhir Sistem

Deskripsi akhir system merupakan tahapan untuk menampilkan hasil implementasi antarmuka berdasarkan perancangan antarmuka yang telah dibuat sebelumnya. Berikut ini adalah hasil implementasi yang telah dibuat:

##### 1. Halaman Awal

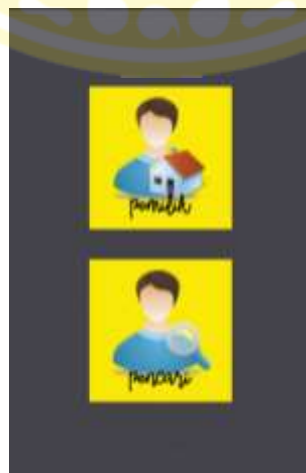
Halaman ini adalah halaman yang pertama kali akan dilihat oleh aktor (pemilik dan pencari). Halaman ini adalah pertanda bahwa telah masuk pada sistem. Implementasi halaman awal dapat dilihat pada Gambar 4.62.



**Gambar 4.37** Implementasi Halaman Awal

##### 2. Halaman Masuk

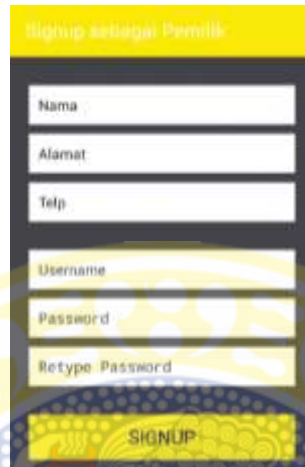
Pada halaman ini aktor akan memilih salah satu tombol yang merepresentasikan diri, yaitu untuk tombol pencari dan untuk tombol pemilik. Implementasi halaman masuk dapat dilihat pada Gambar 4.63.



**Gambar 4.38** Implementasi Halaman Masuk

### 3. Halaman *Signup*/Daftar sebagai Pemilik

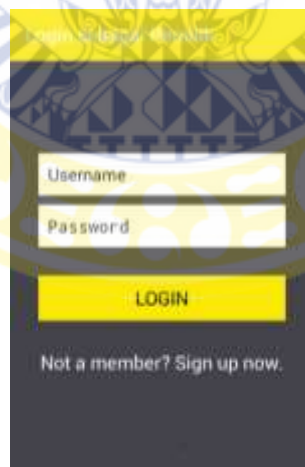
Pada halaman ini, pemilik mendaftarkan diri dengan cara memasukan data diri sesuai permintaan di halaman *signup*. Implementasi halaman *signup*/daftar sebagai pemilik dapat dilihat pada Gambar 4.64.

The image shows a mobile application interface for a 'Signup sebagai Pemilik' (Signup as Owner) page. The title bar is yellow with the text 'Signup sebagai Pemilik'. Below the title bar, there are several input fields: 'Nama' (Name), 'Alamat' (Address), 'Telp.' (Phone), 'Username', 'Password', and 'Retype Password'. At the bottom of the form is a yellow button labeled 'SIGNUP'. The background of the app is dark blue with a large, faint, circular watermark of a university seal in the center.

**Gambar 4.39** Implementasi Halaman *Signup*/Daftar sebagai Pemilik

### 4. Halaman *Login*/Masuk sebagai Pemilik

Pada halaman ini, pemilik dapat masuk kedalam sistem dengan cara memasukan username dan password yang dimiliki oleh pemilik. Implementasi halaman *Login*/Masuk sebagai Pemilik dapat dilihat pada Gambar 4.65.

The image shows a mobile application interface for a 'Login sebagai Pemilik' (Login as Owner) page. The title bar is yellow with the text 'Login sebagai Pemilik'. Below the title bar, there are two input fields: 'Username' and 'Password'. At the bottom of the form is a yellow button labeled 'LOGIN'. Below the button is a link that says 'Not a member? Sign up now.'. The background of the app is dark blue with a large, faint, circular watermark of a university seal in the center.

**Gambar 4.40** Implementasi Halaman *Login*/Masuk sebagai Pemilik

### 5. Halaman Utama Pemilik

Pada halaman ini, pemilik memiliki dua layout yang dapat digeser. Layout pertama untuk menampilkan data diri pemilik dan tombol untuk mengubah data diri. Layout kedua untuk

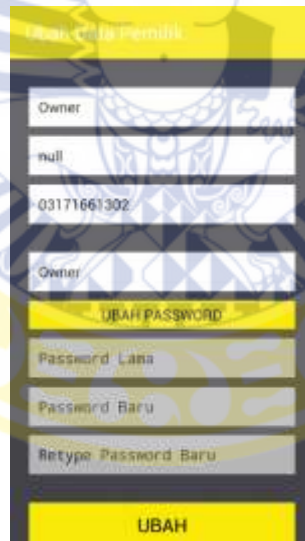
menampilkan pemberitahuan seputar pengelolaan data kos dan tombol untuk mengelola data kos. Implementasi halaman utama pemilik dapat dilihat pada Gambar 4.66



**Gambar 4.41** Implementasi Halaman Utama Pemilik

#### 6. Halaman Ubah Data Diri Pemilik

Pada halaman ini, pemilik dapat mengubah data diri agar data diri selalu terbaru. Implementasi halaman utama pemilik pada Gambar 4.67.

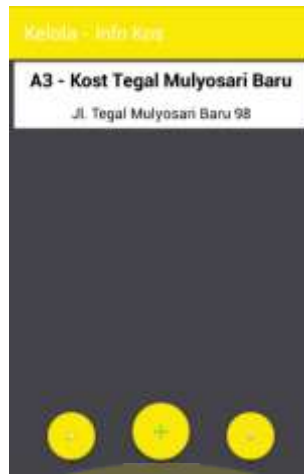


**Gambar 4.42** Implementasi Halaman Ubah Data Diri Pemilik

#### 7. Halaman Daftar Kelola Data Kos

Pada halaman ini, pemilik dapat melihat daftar kelola data kos yang dimiliki dan juga memilih aktifitas pengelolaan yang dapat dilakukan dengan data kos. Implementasi halaman Daftar Kelola Data Kos dapat dilihat pada Gambar 4.68.





**Gambar 4.43** Implementasi Halaman Daftar Kelola Data Kos

8. Halaman Detail Informasi Kelola Data Kos

Pada halaman ini, pemilik dapat melakukan aktifitas yang telah dipilih pada halaman sebelumnya. Jika pemilik memilih menambah data maka halaman ini akan disajikan tanpa data, apabila pemilik memilih mengubah data yang telah ada maka halaman ini akan disajikan dengan data yang telah tersimpan sebelumnya. Implementasi halaman detail informasi kelola data kos dapat dilihat pada Gambar 4.69.

**Gambar 4.44** Implementasi Halaman Detail Informasi Kelola Data Kos

9. Halaman *Signup*/Daftar sebagai Pencari

Pada halaman ini, pencari mendaftarkan diri dengan cara memasukkan data diri sesuai permintaan di halaman *signup*. Implementasi halaman *signup*/daftar sebagai pencari dapat dilihat pada Gambar 4.70.

**Gambar 4.45** Implementasi Halaman *Signup*/Daftar sebagai Pencari

#### 10. Halaman *Login*/Masuk sebagai Pencari

Pada halaman ini, pencari dapat masuk kedalam sistem dengan cara memasukan username dan password yang dimiliki oleh pencari. Implementasi halaman *Login*/Masuk sebagai Pencari dapat dilihat pada Gambar 4.71.

**Gambar 4.46** Implementasi Halaman *Login*/Masuk sebagai Pencari

#### 11. Halaman Utama Pencari

Pada halaman ini, pencari memiliki tiga layout yang dapat digeser. Layout pertama untuk menampilkan data diri pencari dan tombol untuk mengubah data diri. Layout kedua untuk menampilkan pemberitahuan pencarian data kos dan tombol untuk mencari data kos. Layout

ketiga untuk menampilkan pemeritahuan riwayat pencarian data kos dan tombol untuk melihat daftar riwayat kos. Implementasi halaman utama pencari pada Gambar 4.72.

## 12. Halaman Ubah Data Diri Pencari

Pada halaman ini, pencari dapat mengubah data diri yang telah tersimpan agar data diri selalu terbaru. Implementasi halaman utama pencari dapat dilihat pada Gambar 4.73.



**Gambar 4.47** Halaman Utama Pencari



**Gambar 4.48** Implementasi Halaman Utama Pencari

### 13. Halaman Pencarian Data Kos

Pada halaman ini, pencari akan diminta memasukan nilai bobot dengan cara menekan tombol sesuai kriteria yang di inginkan. Bobot yang dimasukan adalah urutan prioritas yang dimiliki pencari untuk setiap kriteria. Implementasi halaman pencarian data kos dapat dilihat pada Gambar 4.74.

Cari - Input Prioritas Kos	
JARAK	BIAYA SEWA
JENIS KAMAR	JUMLAH PENGHUNI
MEJA KURSI	LEMARI
TEMPAT TIDUR	KOR DALAM
KIPAS ANGIN	AC
TV	KULKAS
DISPENSER	INTERNET
RUANG KAMAR	DAPUR
MEKANIS CUCI	PERALATAN
CUCI GELAS	CCTV
KOLAM RENANG	
PARKIR MOTOR	PARKIR MOBIL
RESTAURANT	WAWET
MALL	APOTEK
BANK	SUPERMARKET
KETERSEDIAAN KAMAR	
CLEAR	CARI

**Gambar 4.49** Implementasi Halaman Pencarian Data Kos

### 14. Halaman Daftar Pencarian Data Kos

Pada halaman ini, pencari dapat melihat daftar kos hasil pencarian dari halaman pencarian data kos. Implementasi halaman daftar pencarian data kos dapat dilihat pada Gambar 4.75.

Cari - Daftar Kos	
<b>A3 - Kost Tegal Mulyosari Baru</b> Jl. Tegal Mulyosari Baru 98	
<b>A4 - Kost Keputih Sukolilo</b> Jl. Keputih 1 No. 10a Sukolilo	
<b>A7 - Kost Tegal Mulyorejo Baru</b> Jl. Tegal Mulyorejo Baru No. 14	
<b>A1 - Kost Wisma Permai Tengah</b> Jl. Wisma Permai Tengah 1a/11	
<b>A5 - Kost-kostan aman berair</b> Berhasil Mengubah Data Fuzzy Kos Jl. Sukadami 2 No. 18 Surabaya	
<b>A6 - Kost Putri Rungkut Mejoyo</b>	

**Gambar 4.50** Implementasi Halaman Daftar Pencarian Data Kos

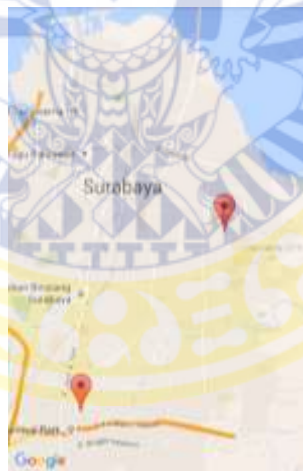


### 15. Halaman Detail Informasi Pencarian Data Kos

Pada halaman ini, pencari dapat melihat detail informasi data kos yang telah dipilih pada halaman daftar pencarian data kos. Halaman detail informasi pencarian data kos dapat dilihat pada Gambar 4.76. Pada halaman ini juga ditampilkan tombol peta yang langsung terhubung dengan *Google Maps* untuk melihat letak kos. Implementasi halaman peta kos pada Gambar 4.77.



**Gambar 4.51** Implementasi Halaman Detail Informasi Pencarian Data Kos

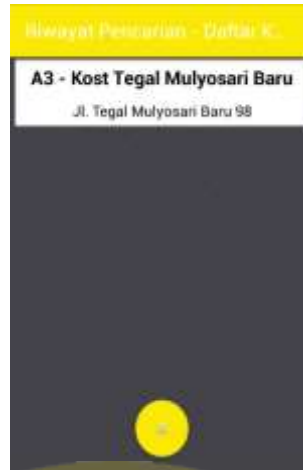


**Gambar 4.52** Implementasi Halaman Peta Kos

### 16. Halaman Daftar Riwayat Pencarian Data Kos

Pada halaman ini, pencari dapat melihat daftar riwayat data kos yang telah disimpan dari daftar pencarian kos. Terdapat tombol hapus untuk menghapus data kos dari daftar riwayat pencarian data kos. Implementasi halaman daftar riwayat pencarian data kos dapat dilihat pada Gambar 4.78.





**Gambar 4.53** Implementasi Halaman Daftar Riwayat Pencarian Data Kos

#### 17. Halaman Detail Informasi Riwayat Pencarian Data Kos

Pada halaman ini, pencari dapat melihat detail informasi data kos yang telah dipilih dimasukan pada riwayat pencarian data kos. Halaman detail informasi riwayat pencarian data kos dapat dilihat pada Gambar 4.79. Pada halaman ini juga ditampilkan tombol peta yang langsung terhubung dengan *Google Maps* untuk melihat letak kos. Implementasi Halaman peta kos dapat dilihat pada Gambar 4.77.



**Gambar 4.54** Implementasi Halaman Detail Informasi Riwayat Pencarian Data Kos

Pengujian Sistem dilakukan dengan metode *black box* testing yang berfokus pada fungsionalitas sistem. Sistem yang diuji adalah fungsi-fungsi yang berkaitan dengan aplikasi Sistem Pendukung Keputusan Pemilihan Kos Daerah Suarabaya.

1. Halaman Awal

Hasil pengujian *black box* testing pada halaman awal ditunjukkan pada Table 4.80.

**Tabel 4.27** Hasil Pengujian Halaman Awal

No.	Test Case	Hasil Harapan	Hasil Keluaran	Hasil Uji
1.	Tekan tombol <i>next</i>	Sistem melakukan tindakan	Sistem membuka halaman awal	Sesuai
2.	Tidak ada input	Sistem tidak melakukan apapun	Sistem tidak melakukan apapun	Sesuai
3.	Tekan disembarang tempat	Sistem tidak melakukan apapun	Sistem tidak melakukan apapun	Sesuai

2. Halaman Masuk

Hasil pengujian *black box* testing pada halaman masuk ditunjukkan pada Table 4.81.

**Tabel 4.28** Hasil Pengujian Halaman Masuk

No	Test Case	Hasil Harapan	Hasil Keluaran	Hasil Uji
1.	Tekan tombol pemilik	Sistem melakukan tindakan	Sistem membuka halaman login pemilik	Sesuai
2.	Tekan tombol pencari	Sistem melakukan tindakan	Sistem membuka halaman login pencari	Sesuai
3.	Tidak ada input	Sistem tidak melakukan apapun	Sistem tidak melakukan apapun	Sesuai
4.	Tekan disembarang tempat	Sistem tidak melakukan apapun	Sistem tidak melakukan apapun	Sesuai

3. Halaman *Login*/Masuk sebagai Pemilik

Hasil pengujian *black box* testing pada halaman *login*/masuk sebagai pemilik ditunjukkan pada Table 4.82.

4. Halaman Signup/Daftar sebagai Pemilik

Hasil pengujian *black box* testing pada halaman *signup*/daftar sebagai pemilik ditunjukkan pada Table 4.83.

**Tabel 4.29** Hasil Pengujian Halaman Masuk Pemilik

No.	Test Case	Hasil Harapan	Hasil Keluaran	Hasil Uji
1.	Input benar	Sistem memproses input	Sistem memproses input dan membuka halaman utama pemilik	Sesuai
2.	Input tidak lengkap	Sistem menolak memproses input	Sistem menolak memproses input dan menampilkan pesan “Data Belum Lengkap”	Sesuai
3.	Input kosong	Sistem menolak memproses input	Sistem menolak memproses input dan menampilkan pesan “Data Kosong”	Sesuai
4.	Input salah	Sistem menolak memproses input	Sistem menolak memproses input dan menampilkan pesan “Data Salah”	Sesuai

**Tabel 4.30** Hasil Pengujian Halaman Daftar Pemilik

No.	Test Case	Hasil Harapan	Hasil Keluaran	Hasil Uji
1.	Input benar	Sistem memproses input	Sistem memproses input dan membuka halaman utama pemilik	Sesuai
2.	Input tidak lengkap	Sistem menolak memproses input	Sistem menolak memproses input dan menampilkan pesan “Data Belum Lengkap”	Sesuai
3.	Input kosong	Sistem menolak memproses input	Sistem menolak memproses input dan menampilkan pesan “Data Kosong”	Sesuai
4.	Input salah	Sistem menolak memproses input	Sistem menolak memproses input dan menampilkan pesan “Data Salah”	Sesuai

5. Halaman Utama Pemilik

Hasil pengujian *black box* testing pada halaman utama pemilik ditunjukkan pada Table 4.84.

**Tabel 4.31** Hasil Pengujian Halaman Utama Pemilik

No.	Test Case	Hasil Harapan	Hasil Keluaran	Hasil Uji
1.	Tekan tombol ubah	Sistem melakukan tindakan	Sistem membuka halaman ubah data diri pemilik	Sesuai
2.	Tekan tombol kelola	Sistem melakukan tindakan	Sistem membuka halaman daftar kelola data kos	Sesuai
3.	Tidak ada input	Sistem tidak melakukan apapun	Sistem tidak melakukan apapun	Sesuai
4.	Tekan disembarang tempat	Sistem tidak melakukan apapun	Sistem tidak melakukan apapun	Sesuai

## 6. Halaman Ubah Data Diri Pemilik

Hasil pengujian *black box* testing pada halaman ubah data diri pemilik ditunjukkan pada Table 4.85.

**Tabel 4.32** Hasil Pengujian Halaman Ubah Data Diri Pemilik

No.	Test Case	Hasil Harapan	Hasil Keluaran	Hasil Uji
1.	Input benar	Sistem memproses input	Sistem memproses input dan membuka halaman utama pemilik	Sesuai
2.	Input tidak lengkap	Sistem menolak memproses input	Sistem menolak memproses input dan menampilkan pesan “Data Belum Lengkap”	Sesuai
3.	Input kosong	Sistem menolak memproses input	Sistem menolak memproses input dan menampilkan pesan “Data Kosong”	Sesuai
4.	Input salah	Sistem menolak memproses input	Sistem menolak memproses input dan menampilkan pesan “Data Salah”	Sesuai

## 7. Halaman Daftar Kelola Data Kos

Hasil pengujian *black box* testing pada halaman daftar kelola data kos ditunjukkan pada Table 4.86.

**Tabel 4.33** Hasil Pengujian Halaman Daftar Kelola Data Kos

No.	Test Case	Hasil Harapan	Hasil Keluaran	Hasil Uji
1.	Tekan tombol ubah	Sistem melakukan tindakan	Sistem membuka halaman detail informasi kelola data kos dan menampilkan data kos yang telah tersimpan di <i>database</i>	Sesuai
2.	Tekan tombol tambah	Sistem melakukan tindakan	Sistem membuka halaman detail informasi kelola data kos	Sesuai
3.	Tekan tombol hapus	Sistem melakukan tindakan	Sistem menghapus data kos dari daftar dan dari <i>database</i>	Sesuai
4.	Tidak ada input	Sistem tidak melakukan apapun	Sistem tidak melakukan apapun	Sesuai
5.	Tekan disembarang tempat	Sistem tidak melakukan apapun	Sistem tidak melakukan apapun	Sesuai



8. Halaman Detail Informasi Kelola Data Kos

Hasil pengujian *black box* testing pada halaman detail informasi kelola data kos ditunjukkan pada Table 4.87.

**Tabel 4.34** Hasil Pengujian Halaman Detail Informasi Kelola Data Kos

No.	Test Case	Hasil Harapan	Hasil Keluaran	Hasil Uji
1.	Input benar	Sistem memproses input	Sistem memproses input dan membuka halaman daftar kelola data kos	Sesuai
2.	Input tidak lengkap	Sistem menolak memproses input	Sistem menolak memproses input dan menampilkan pesan “Data Belum Lengkap”	Sesuai
3.	Input kosong	Sistem menolak memproses input	Sistem menolak memproses input dan menampilkan pesan “Data Kosong”	Sesuai
4.	Input salah	Sistem menolak memproses input	Sistem menolak memproses input dan menampilkan pesan “Data Salah”	Sesuai

9. Halaman Signup/Daftar sebagai Pencari

Hasil pengujian *black box* testing pada halaman *signup*/daftar sebagai pencari ditunjukkan pada Table 4.88.

**Tabel 4.35** Hasil Pengujian Halaman Daftar Pencari

No.	Test Case	Hasil Harapan	Hasil Keluaran	Hasil Uji
1.	Input benar	Sistem memproses input	Sistem memproses input dan membuka halaman utama pencari	Sesuai
2.	Input tidak lengkap	Sistem menolak memproses input	Sistem menolak memproses input dan menampilkan pesan “Data Belum Lengkap”	Sesuai
3.	Input kosong	Sistem menolak memproses input	Sistem menolak memproses input dan menampilkan pesan “Data Kosong”	Sesuai
4.	Input salah	Sistem menolak memproses input	Sistem menolak memproses input dan menampilkan pesan “Data Salah”	Sesuai

10. Halaman *Login*/Masuk sebagai Pencari

Hasil pengujian *black box* testing pada halaman *login*/masuk sebagai pencari ditunjukkan pada Table 4.88.



**Tabel 4.36** Hasil Pengujian Halaman Masuk Pencari

No.	Test Case	Hasil Harapan	Hasil Keluaran	Hasil Uji
1.	Input benar	Sistem memproses input	Sistem memproses input dan membuka halaman utama pencari	Sesuai
2.	Input tidak lengkap	Sistem menolak memproses input	Sistem menolak memproses input dan menampilkan pesan “Data Belum Lengkap”	Sesuai
3.	Input kosong	Sistem menolak memproses input	Sistem menolak memproses input dan menampilkan pesan “Data Kosong”	Sesuai
4.	Input salah	Sistem menolak memproses input	Sistem menolak memproses input dan menampilkan pesan “Data Salah”	Sesuai

#### 11. Halaman Utama Pencari

Hasil pengujian *black box* testing pada halaman utama pencari ditunjukkan pada Table 4.90.

**Tabel 4.37** Hasil Pengujian Halaman Utama Pencari

No.	Test Case	Hasil Harapan	Hasil Keluaran	Hasil Uji
1.	Tekan tombol ubah	Sistem melakukan tindakan	Sistem membuka halaman ubah data diri pencari	Sesuai
2.	Tekan tombol kelola	Sistem melakukan tindakan	Sistem membuka halaman daftar riwayat pencarian data kos	Sesuai
3.	Tekan tombol cari	Sistem melakukan tindakan	Sistem membuka halaman pencarian data kos	Sesuai
4.	Tidak ada input	Sistem tidak melakukan apapun	Sistem tidak melakukan apapun	Sesuai
5.	Tekan disembarang tempat	Sistem tidak melakukan apapun	Sistem tidak melakukan apapun	Sesuai

#### 12. Halaman Ubah Data Diri Pencari

Hasil pengujian *black box* testing pada halaman ubah data diri pencari ditunjukkan pada Table 4.91.

**Tabel 4.38** Hasil Pengujian Halaman Ubah Data Diri Pencari

No.	Test Case	Hasil Harapan	Hasil Keluaran	Hasil Uji
1.	Input benar	Sistem memproses input	Sistem memproses input dan membuka halaman utama pencari	Sesuai
2.	Input tidak lengkap	Sistem menolak memproses input	Sistem menolak memproses input dan menampilkan pesan “Data Belum Lengkap”	Sesuai
3.	Input kosong	Sistem menolak memproses input	Sistem menolak memproses input dan menampilkan pesan “Data Kosong”	Sesuai
4.	Input salah	Sistem menolak memproses input	Sistem menolak memproses input dan menampilkan pesan “Data Salah”	Sesuai

#### 13. Halaman Pencarian Data Kos

Hasil pengujian *black box* testing pada halaman pencarian data kos ditunjukkan pada Table 4.92.

**Tabel 4.39** Hasil Pengujian Halaman Pencarian data Kos

No.	Test Case	Hasil Harapan	Hasil Keluaran	Hasil Uji
1.	Input benar	Sistem memproses input	Sistem memproses input dan menampilkan rekomendasi kos pada halaman daftar pencarian data kos	Sesuai
2.	Input kosong	Sistem menolak memproses input	Sistem menolak memproses input dan menampilkan pesan “Data Kosong”	Sesuai

#### 14. Halaman Daftar Pencarian Data Kos

Hasil pengujian *black box* testing pada halaman daftar pencarian data kos ditunjukkan pada Table 4.93.

#### 15. Halaman Detail Informasi Pencarian Data Kos

Hasil pengujian *black box* testing pada halaman detail informasi pencarian data kos ditunjukkan pada Table 4.94.

**Tabel 4.40** Hasil Pengujian Halaman Daftar Pencarian Data Kos

No.	Test Case	Hasil Harapan	Hasil Keluaran	Hasil Uji
1.	Tekan nama kos	Sistem melakukan tindakan	Sistem membuka halaman detail informasi pencarian data kos dan menampilkan data kos yang telah tersimpan di <i>database</i>	Sesuai
2.	Tidak ada input	Sistem tidak melakukan apapun	Sistem tidak melakukan apapun	Sesuai
3.	Tekan disembarang tempat	Sistem tidak melakukan apapun	Sistem tidak melakukan apapun	Sesuai

**Tabel 4.41** Hasil Pengujian Halaman Detail Informasi Pencarian Data Kos

No.	Test Case	Hasil Harapan	Hasil Keluaran	Hasil Uji
1.	Tekan tombol peta	Sistem melakukan tindakan	Sistem membuka halaman peta kos	Sesuai
2.	Tekan tombol riwayat	Sistem melakukan tindakan	Sistem memproses data kos masuk kedalam <i>database</i> dan daftar riwayat pencarian data kos	Sesuai
3.	Tekan tombol kembali	Sistem melakukan tindakan	Sistem membuka halaman daftar pencarian data kos	Sesuai
4.	Tidak ada input	Sistem tidak melakukan apapun	Sistem tidak melakukan apapun	Sesuai
5.	Tekan disembarang tempat	Sistem tidak melakukan apapun	Sistem tidak melakukan apapun	Sesuai

#### 16. Halaman Daftar Riwayat Pencarian Data Kos

Hasil pengujian *black box* testing pada halaman daftar riwayat pencarian data kos ditunjukkan pada Table 4.95.

#### 17. Halaman Detail Informasi Riwayat Pencarian Data Kos

Hasil pengujian *black box* testing pada halaman detail informasi riwayat pencarian data kos ditunjukkan pada Table 4.96.

**Tabel 4.42** Hasil Pengujian Halaman Daftar Riwayat Pencarian Data Kos

No.	Test Case	Hasil Harapan	Hasil Keluaran	Hasil Uji
1.	Tekan nama kos	Sistem melakukan tindakan	Sistem membuka halaman detail informasi riwayat pencarian data kos dan menampilkan data kos yang telah tersimpan di <i>database</i>	Sesuai
2.	Tekan tombol hapus	Sistem melakukan tindakan	Sistem menghapus data kos dari daftar dan dari <i>database</i>	Sesuai
3.	Tidak ada input	Sistem tidak melakukan apapun	Sistem tidak melakukan apapun	Sesuai
4.	Tekan disembarang tempat	Sistem tidak melakukan apapun	Sistem tidak melakukan apapun	Sesuai

**Tabel 4.43** Hasil Pengujian Halaman Detail Informasi Riwayat Pencarian Data Kos

No.	Test Case	Hasil Harapan	Hasil Keluaran	Hasil Uji
1.	Tekan tombol peta	Sistem melakukan tindakan	Sistem membuka halaman peta kos	Sesuai
2.	Tekan tombol kembali	Sistem melakukan tindakan	Sistem membuka halaman daftar riwayat pencarian data kos	Sesuai
3.	Tidak ada input	Sistem tidak melakukan apapun	Sistem tidak melakukan apapun	Sesuai
4.	Tekan disembarang tempat	Sistem tidak melakukan apapun	Sistem tidak melakukan apapun	Sesuai