

PERANCANGAN APLIKASI BANK SAMPAH PONTIANAK DENGAN MENGGUNAKAN SDK MAPBOX DAN LOGIKA FUZZY

Giovanni De Leandro¹, Tony Darmanto², Susana³

^{1,2,3}Informatika, Fakultas Teknologi Informasi Universitas Widya Dharma Pontianak
e-mail: ¹18421161_giovanni_d_l@widyadharm.ac.id, ²tony.darmanto@yahoo.com,
³susana@widyadharm.ac.id

Abstract

The lack of public knowledge about waste banks causes waste to pile up in temporary disposal sites. Therefore, the government started building waste banks in 2008. To help provide information about waste banks in Pontianak City, this research aims to design an Android-based application named Bank Sampah Pontianak using the Mapbox SDK and fuzzy logic. The Bank Sampah Pontianak application was built using Android Studio. This application uses Firebase Firestore as the database provider and Mapbox as the map provider. In fuzzy logic calculations, the distance between the user and the waste bank and/or the price of the waste is used. The result of this calculation is the recommendation value for each waste bank which can be sorted in the application. The higher the recommendation value of a waste bank, the closer the user is to the waste bank and the higher the price of the waste. Application testing was carried out on four devices and showed that the application could run well. The conclusion of this research is that the application can provide waste bank recommendations to users based on fuzzy logic calculations and determine the estimated funds that will be received after handing over the waste to the waste bank. Users can also see the location of waste banks on the map. Apart from that, waste bank admins can also update waste data easily.

Keywords—Waste Bank, Pontianak, Android, Mapbox, Fuzzy Logic

Abstrak

Kurangnya pengetahuan masyarakat mengenai bank sampah menyebabkan sampah yang dibuang menjadi menumpuk di tempat pembuangan sementara. Oleh karena itu, pemerintah mulai membangun bank sampah pada tahun 2008. Untuk membantu memberikan informasi tentang bank sampah di Kota Pontianak, penelitian ini bertujuan untuk merancang aplikasi Bank Sampah Pontianak berbasis Android menggunakan SDK Mapbox dan logika fuzzy. Aplikasi Bank Sampah Pontianak dibangun menggunakan Android Studio. Aplikasi ini menggunakan Firebase Firestore sebagai penyedia basis data dan Mapbox sebagai penyedia peta. Pada perhitungan logika fuzzy digunakan jarak pengguna terhadap bank sampah dan/atau harga sampah. Hasil dari perhitungan tersebut adalah nilai rekomendasi pada setiap bank sampah yang dapat disortir di dalam aplikasi. Semakin tinggi nilai rekomendasi sebuah bank sampah, maka jarak pengguna terhadap bank sampah semakin dekat dan harga sampah semakin tinggi. Pengujian aplikasi dilakukan pada empat perangkat dan menunjukkan bahwa aplikasi dapat berjalan dengan baik. Kesimpulan dari penelitian ini adalah aplikasi dapat memberikan rekomendasi bank sampah kepada pengguna berdasarkan perhitungan logika fuzzy dan mengetahui perkiraan dana yang akan diterima setelah menyerahkan sampah kepada bank sampah. Pengguna juga dapat melihat lokasi bank sampah pada peta. Selain itu, admin bank sampah juga dapat memperbarui data sampah dengan mudah.

Kata kunci—Bank Sampah, Pontianak, Android, Mapbox, Logika Fuzzy

1. PENDAHULUAN

Peningkatan jumlah penduduk yang terjadi di Indonesia mengalami kenaikan yang sangat pesat. Hal ini juga terjadi di Kota Pontianak. Berdasarkan data Dinas Kependudukan dan Pencatatan Sipil Kota Pontianak, penambahan penduduk Kota Pontianak dapat bertambah dari sekitar 200 hingga 2000 orang setiap semester. Pertambahan jumlah penduduk ini berbanding lurus dengan meningkatnya jumlah sampah yang terbentuk dari kegiatan sehari-hari manusia yaitu produksi, distribusi, dan konsumsi.

Jumlah sampah yang berlebih apabila tidak terurus dengan baik dapat merusak lingkungan tempat tinggal masyarakat. Membuang sampah tidak pada tempatnya atau membakar sampah secara tidak bertanggung jawab adalah contoh dari penanganan sampah yang salah. Dampak dari penanganan sampah yang salah tersebut adalah kemungkinan terjadinya banjir yang disebabkan oleh sampah yang menyumbat saluran air, polusi udara yang disebabkan oleh asap pembakaran sampah, dan yang lainnya. Apabila hal ini terus dibiarkan, maka lingkungan tempat tinggal masyarakat akan menjadi tidak layak untuk dihuni.

Untuk mencegah terjadinya hal tersebut, pemerintah mulai membangun bank sampah pada tahun 2008. Sesuai dengan Peraturan Pemerintah Republik Indonesia Nomor 27 Tahun 2020 tentang Pengelolaan Sampah Spesifik, bank sampah adalah tempat pemilahan dan pengumpulan sampah yang dapat didaur ulang dan/atau diguna ulang sehingga memiliki nilai ekonomis. Sampah anorganik yang telah dipisahkan dari sampah organik diberikan kepada bank sampah, kemudian akan didaur ulang menjadi barang baru yang dapat dijual. Bagi masyarakat yang menyerahkan sampah kepada bank sampah dapat diberikan dana berdasarkan jenis, kualitas, dan kuantitas sampah yang telah dikumpulkan.

Kurangnya pengetahuan masyarakat mengenai bank sampah menyebabkan sampah yang dibuang menjadi menumpuk di Tempat Pembuangan Sementara (TPS). Sampah yang dibuang juga bercampur sehingga proses daur ulang menjadi lambat. Agar dapat membantu memberikan informasi mengenai bank sampah di Kota Pontianak, maka peneliti merancang aplikasi Bank Sampah Pontianak berbasis Android menggunakan SDK Mapbox dan logika Fuzzy, dengan tujuan untuk menginformasikan tentang lokasi bank sampah di Kota Pontianak pada peta, estimasi dana yang akan didapatkan setelah menyerahkan sampah, serta merekomendasikan bank sampah terbaik untuk pengguna berdasarkan logika Fuzzy.

2. METODE PENELITIAN

2.1 Perancangan Aplikasi, Pembangunan Aplikasi, Pengujian Aplikasi

2.1.1 Perancangan Aplikasi

Pada proses perancangan aplikasi, peneliti menggunakan pemodelan *Unified Modeling Language* (UML) untuk mengilustrasikan proses interaksi pengguna dengan aplikasi.

2.1.2 Pembangunan Aplikasi

Pada proses pembangunan aplikasi, peneliti menggunakan Android Studio untuk pengetikan kode aplikasi, Firebase Firestore sebagai penyedia basis data, dan SDK Mapbox sebagai penyedia peta.

2.1.3 Pengujian Aplikasi

Pada proses pengujian aplikasi, peneliti menggunakan metode *black box testing* terhadap empat buah *smartphone* dengan versi Android dan spesifikasi yang berbeda untuk mengetahui apakah aplikasi dapat berjalan sesuai dengan perancangan.

2.2 Landasan Teori

2.2.1 Bank Sampah

Bank sampah adalah tempat yang digunakan untuk memilah dan mengumpulkan sampah daur/guna ulang yang sudah dipilah, yang akan disetorkan ke tempat pembuat kerajinan dari sampah atau ke tempat pengepul sampah^[1,2].

2.2.2 Android

Android adalah sistem operasi berbasis Linux yang bersifat *open source* pada *smartphone* atau tablet^[3].

2.2.3 Kotlin

Kotlin adalah bahasa pemrograman yang dikembangkan oleh JetBrains dan dapat digunakan untuk pengembangan aplikasi Android^[4,5].

2.2.4 Mapbox

Mapbox adalah platform data lokasi untuk aplikasi *mobile* dan web yang tampilan petanya dapat diubah, serta dapat menambahkan penanda pada lokasi yang diinginkan. Data yang digunakan Mapbox diambil dari sumber-sumber terbuka maupun berbayar^[6,7].

2.2.5 Software Development Kit (SDK)

Software Development Kit atau SDK adalah sekumpulan alat yang berfungsi untuk membuat aplikasi agar bisa dijalankan pada berbagai platform termasuk Android^[8,9].

2.2.6 Firebase

Firebase adalah platform untuk pengembangan aplikasi Android dan web yang menawarkan layanan berupa 18 buah produk yang salah satunya adalah *realtime database* untuk mempermudah para developer dalam mengembangkan aplikasinya^[10,11].

2.2.7 Database

Database adalah kumpulan data yang saling berelasi dan terorganisir, yang dapat diolah atau dimanipulasi untuk menghasilkan informasi yang bermanfaat secara elektronik menggunakan suatu perangkat lunak (*software*)^[12,13].

2.2.8 Logika Fuzzy

Logika *fuzzy* adalah suatu bentuk logika dimana nilai kebenaran variabel dapat berupa bilangan real dari 0 sampai dengan 1^[14]. Dalam bidang ilmu kecerdasan buatan, logika *fuzzy* merupakan bagian dari salah satu teknik penyelesaian masalah *reasoning*. Teknik penyelesaian masalah *reasoning* atau dikenal dengan teknik penalaran yaitu teknik penyelesaian masalah yang merepresentasikan masalah ke dalam basis pengetahuan menggunakan logika (*logic*). Terdapat tiga tahapan dalam penyelesaian sistem kendali berbasis aturan *fuzzy*, yaitu:

a. Fuzzification

Fuzzifikasi adalah proses untuk mengubah suatu masukan dari bentuk tegas (*crisp input*) ke dalam bentuk linguistik (*fuzzy input*). Fuzzifikasi disajikan dalam bentuk himpunan-himpunan *fuzzy* menggunakan suatu

fungsi keanggotaan (*membership function*). Fungsi keanggotaan digunakan untuk menentukan derajat keanggotaan (*degree of membership function*) suatu item x dalam himpunan *fuzzy*.

- b. Inference
Proses inferensi (*inference*) adalah proses penalaran terhadap *fuzzy input* yang didapatkan dari hasil fuzzifikasi dan dikirim ke dalam basis pengetahuan yang berisi aturan-aturan *fuzzy* untuk menghasilkan *fuzzy output*.
- c. Defuzzification
Proses perubahan *fuzzy output* menjadi *crisp output* (numerik) berdasarkan fungsi keanggotaan yang telah ditentukan disebut dengan proses defuzzifikasi.^[15]

2.2.9 Unified Modeling Language (UML)

UML adalah bahasa untuk menspesifikasi, memvisualisasi, membangun, dan mendokumentasikan *artifacts* (bagian dari informasi yang digunakan untuk dihasilkan oleh proses pembuatan perangkat lunak, *artifact* tersebut dapat berupa model, deskripsi atau perangkat lunak) dari sistem perangkat lunak, seperti pada pemodelan bisnis dan sistem non perangkat lunak lainnya^[16]. Jenis-jenis diagram dalam UML:

- a. Diagram Class
Diagram *class* merupakan struktur statis/tetap *class* di dalam sebuah sistem. *Class* mencitrakan sesuatu yang dapat diselesaikan oleh sistem. *Class* memiliki tiga area pokok yang terdiri dari nama dan *stereotype*, atribut, dan metode. Atribut dan metode dapat bersifat *private* (sifat yang tidak dapat dipanggil dari luar *class* yang bersangkutan), *protected* (sifat yang hanya dapat dipanggil oleh *class* yang bersangkutan dan anak-anak yang mewarisinya), atau *public* (sifat yang dapat dipanggil oleh siapa saja dari kelas manapun).
- b. Diagram Use Case
Diagram *use case* menggambarkan sejumlah *external actors* dan hubungannya kepada *use case* yang diberikan oleh sistem. *Use case* dapat digambarkan mengenai apa yang hanya dilihat dari luar oleh *actor* (keadaan lingkungan luar dari sistem yang dilihat *user*) dan juga bukan dari bagaimana fungsi yang terdapat di dalam sistem tersebut.
- c. Diagram Sequence
Diagram *sequence* merupakan diagram yang menggambarkan kolaborasi dinamis (kolaborasi yang berubah-ubah) antara beberapa objek. Kegunaan dari diagram *sequence* adalah untuk menunjukkan rangkaian pesan atau *message* yang dikirim antar objek dan juga interaksi antar objek, serta sesuatu yang terjadi pada titik tertentu dalam pelaksanaan sebuah sistem.
- d. Diagram Activity
Diagram *activity* merupakan diagram yang menggambarkan rangkaian aliran dari aktivitas dalam artian urutan aktivitas. Diagram *activity* ini dapat digunakan untuk menjelaskan mengenai aktivitas yang dibentuk dalam suatu operasi sehingga dapat dipergunakan untuk aktivitas lainnya seperti *use case* ataupun interaksi.^[17]

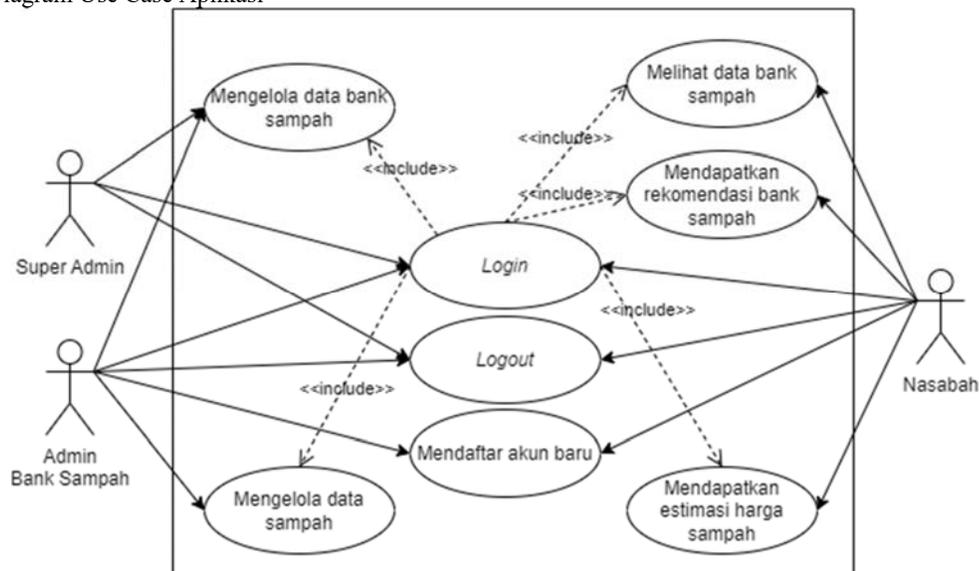
2.2.10 Black Box Testing

Pengujian *black box* adalah pengujian yang dilakukan terhadap spesifikasi fungsional yang terdapat pada aplikasi, dimana setiap *user interface* akan dilakukan pengecekan agar setiap fungsi yang ada sesuai dengan kebutuhan^[18].

3. HASIL DAN PEMBAHASAN

3.1 Perancangan Aplikasi

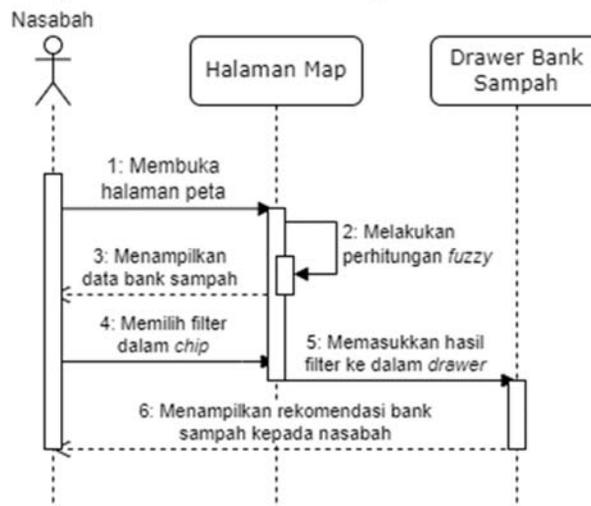
3.1.1 Diagram Use Case Aplikasi



Gambar 1. Diagram Use Case Aplikasi

Diagram *use case* aplikasi Bank Sampah Pontianak pada Gambar 1 menjelaskan bahwa pada aplikasi yang dikembangkan terdapat tiga jenis pengguna yaitu nasabah, admin bank sampah, dan super admin. Pengguna aplikasi akan dapat membuat dan menggunakan akun dengan jenis pengguna sebagai nasabah atau admin bank sampah. Untuk jenis pengguna berupa super admin hanya dapat digunakan oleh peneliti. Setiap jenis pengguna akan memiliki akses untuk fitur yang berbeda pada aplikasi. Apabila pengguna masuk ke aplikasi menggunakan akun nasabah, pengguna dapat melihat data bank sampah seperti nama, alamat, dan lokasi bank sampah pada peta; mendapatkan rekomendasi bank sampah menurut logika *fuzzy*; dan mendapatkan estimasi harga sampah. Sebagai admin bank sampah, pengguna dapat mengolah data bank sampah yang diurusnya, serta sampah yang dapat diterima. Sebagai super admin, pengguna dapat membantu semua admin bank sampah dalam mengolah data bank sampah.

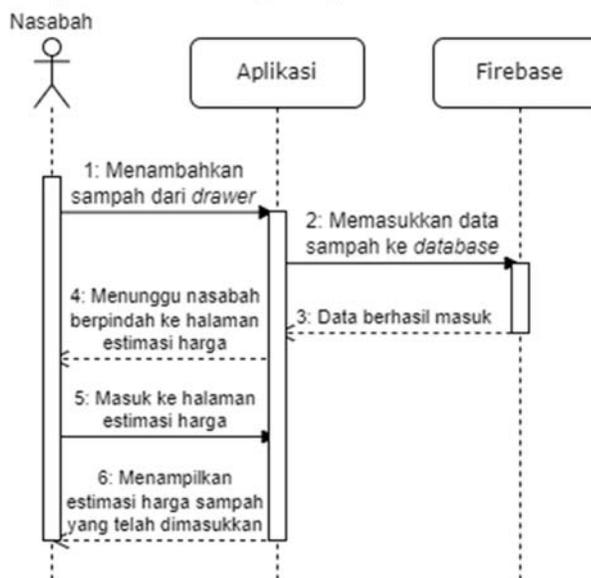
3.1.2 Diagram Sequence Mendapatkan Rekomendasi Bank Sampah



Gambar 2. Diagram Sequence Mendapatkan Rekomendasi Bank Sampah

Diagram *sequence* mendapatkan rekomendasi bank sampah pada aplikasi Bank Sampah Pontianak dalam Gambar 2 dimulai saat nasabah membuka halaman Peta. Pada halaman tersebut, aplikasi akan melakukan perhitungan logika *fuzzy*. Di dalam map akan terdapat *chip group* yang berisi tiga buah *chip* untuk memfilter data bank sampah sesuai filter pilihan nasabah. Setelah nasabah memilih filter yang diinginkan, aplikasi akan menampilkan urutan bank sampah yang paling direkomendasikan berdasarkan filter yang dipilih sesuai dengan perhitungan logika *fuzzy* di dalam *drawer*.

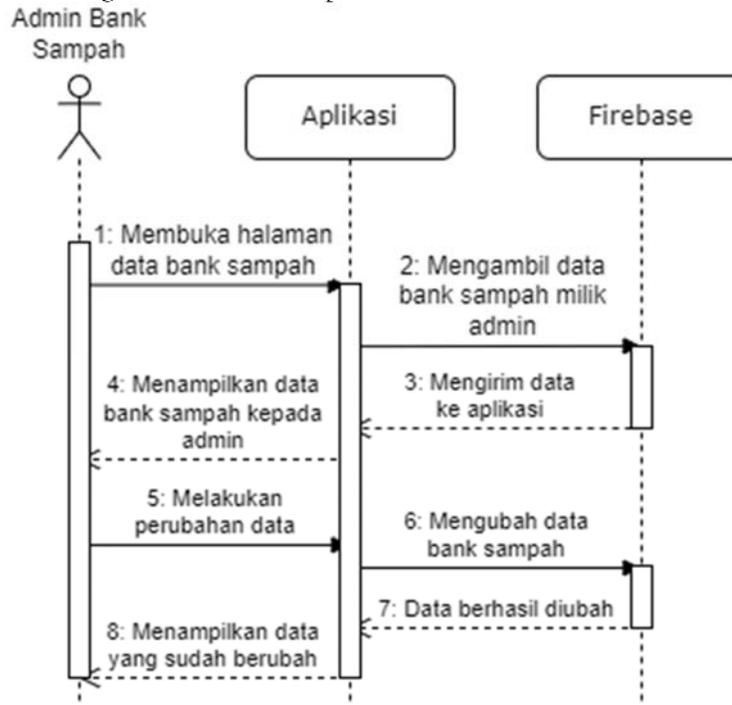
3.1.3 Diagram Sequence Mendapatkan Estimasi Harga Sampah



Gambar 3. Diagram Sequence Mendapatkan Estimasi Harga Sampah

Diagram *sequence* mendapatkan estimasi harga sampah pada aplikasi Bank Sampah Pontianak dalam Gambar 3 dimulai pada saat nasabah menambahkan sampah dari *drawer* pada halaman Peta. Aplikasi akan memasukkan data sampah yang sudah dipilih nasabah ke dalam *database* dan akan menampilkannya kepada nasabah di halaman Estimasi Harga. Pada halaman ini nasabah dapat mengubah jumlah sampah yang hendak diserahkan.

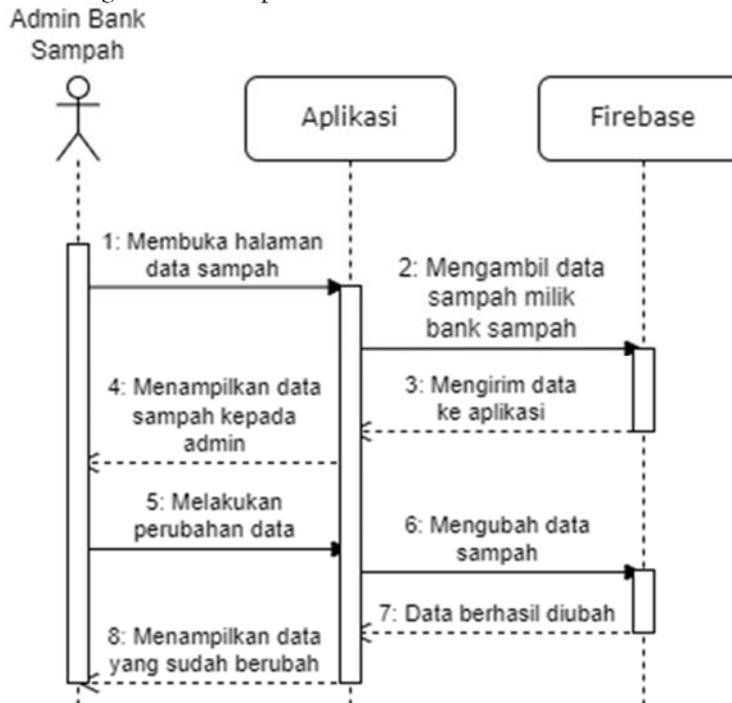
3.1.4 Diagram Sequence Mengelola Data Bank Sampah



Gambar 4. Diagram Sequence Mengelola Data Bank Sampah

Diagram *sequence* mengolah data bank sampah pada aplikasi Bank Sampah Pontianak dalam Gambar 4 dimulai pada saat admin bank sampah membuka halaman Data Bank Sampah. Aplikasi akan mendapatkan data bank sampah dari *database* kemudian menampilkannya kepada admin. Pada halaman ini admin dapat melakukan perubahan terhadap data bank sampah kemudian aplikasi akan mengubah data yang sudah ada di *database*, kemudian menampilkannya kepada admin bank sampah.

3.1.5 Diagram Sequence Mengolah Data Sampah



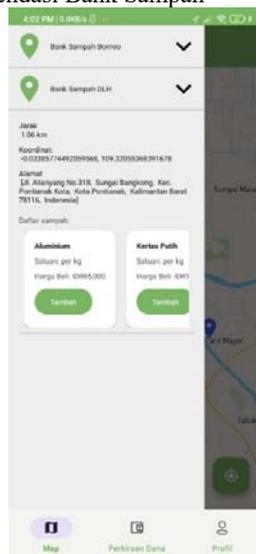
Gambar 5. Diagram Sequence Mengolah Data Sampah

Diagram *sequence* mengolah data sampah pada aplikasi Bank Sampah Pontianak dalam Gambar 5 dimulai pada saat admin bank sampah membuka halaman Data Sampah. Aplikasi akan mendapatkan data sampah dari *database* kemudian menampilkannya kepada admin. Pada halaman ini admin dapat melakukan perubahan terhadap data sampah yang dapat diterima oleh bank sampah yang telah diolah oleh admin, kemudian aplikasi akan mengubah data yang sudah ada di *database* dan menampilkannya kepada admin bank sampah.

3.2 Tampilan Antarmuka Hasil Pembangunan Aplikasi

Aplikasi Bank Sampah Pontianak memiliki tampilan antarmuka yang dapat berubah sesuai dengan mode terang atau gelap sesuai dengan pengaturan yang diterapkan pada *smartphone* pengguna. Berikut adalah tampilan antarmuka aplikasi Bank Sampah Pontianak yang dibuat berdasarkan rancangan:

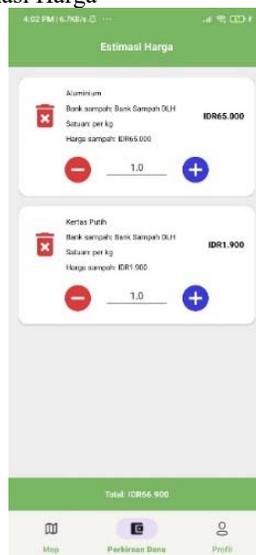
3.2.1 Tampilan Antarmuka Urutan Rekomendasi Bank Sampah



Gambar 6. Tampilan Antarmuka Urutan Rekomendasi Bank Sampah

Ketika pengguna memilih tombol Membuka *Drawer* yang terdapat di bagian kiri atas halaman Peta Bank Sampah, maka pengguna dapat melihat daftar bank sampah yang sudah terdaftar di aplikasi seperti pada Gambar 6. Pada *drawer* tersebut, terdapat beberapa tombol yang dapat dipilih pengguna yaitu tombol berbentuk pin *marker* hijau, tombol *Expand*, dan tombol Tambah Sampah. Ketika pengguna memilih tombol berbentuk pin *marker* hijau, maka peta akan menampilkan lokasi bank sampah yang dipilih. Selain menggunakan tombol *Expand*, pengguna juga dapat memilih nama bank sampah untuk melihat keterangan bank sampah tersebut. Untuk menutup *drawer*, pengguna dapat memilih area gelap pada sebelah kanan *drawer*.

3.2.2 Tampilan Antarmuka Halaman Estimasi Harga



Gambar 7. Tampilan Antarmuka Halaman Estimasi Harga

Halaman Estimasi Harga pada Gambar 7 merupakan halaman yang akan muncul saat pengguna memilih menu estimasi harga pada bagian bawah aplikasi. Pada halaman ini ditampilkan data sampah yang sudah pengguna tambahkan dari *drawer* pada halaman Peta Bank Sampah. Di halaman ini, pengguna dapat melihat estimasi harga sampah yang hendak diserahkan berdasarkan jumlah sampah per satuan. Terdapat tiga buah tombol yaitu tombol Hapus Sampah, Kurang dan Tambah. Ketika pengguna memilih tombol Hapus Sampah, maka akan tampil sebuah jendela konfirmasi untuk menghapus sampah dari halaman Estimasi Harga. Ketika pengguna memilih tombol Kurang atau Tambah, maka kuantitas sampah akan berkurang atau bertambah satu. Terdapat juga *input* teks jumlah sampah yang dapat diketikkan oleh pengguna. Pada bagian bawah aplikasi terdapat sebuah teks yaitu total estimasi harga sesuai dengan jumlah dan sampah yang telah ditambahkan ke halaman Estimasi Harga.

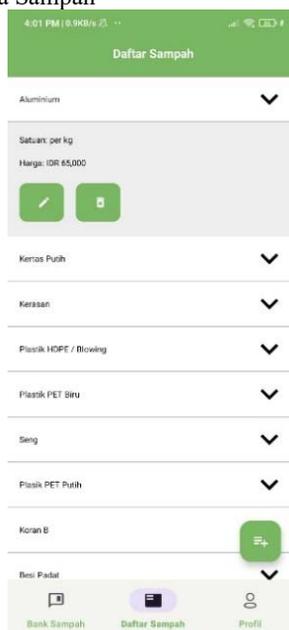
3.2.3 Tampilan Antarmuka Halaman Data Bank Sampah



Gambar 8. Tampilan Antarmuka Halaman Data Bank Sampah

Halaman Data Bank Sampah seperti pada Gambar 8 merupakan halaman yang akan muncul saat pengguna menyelesaikan proses *login* menggunakan akun dengan jenis pengguna admin bank sampah atau ketika pengguna memilih menu data bank sampah. Pada halaman ini terdapat data bank sampah yaitu nama, titik koordinat, dan alamat bank sampah, serta sebuah tombol untuk mengubah data bank sampah.

3.2.4 Tampilan Antarmuka Halaman Data Sampah



Gambar 9. Tampilan Antarmuka Halaman Data Sampah

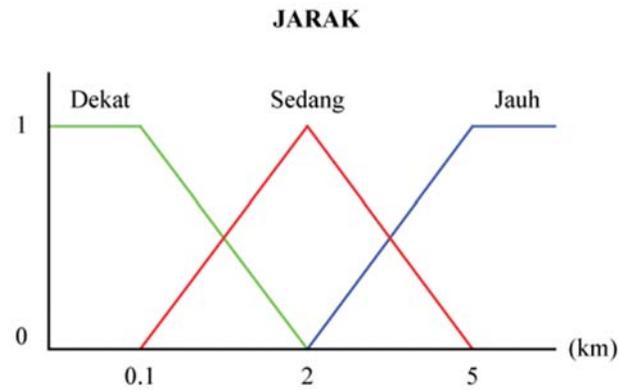
Halaman Data Sampah seperti pada Gambar 9 merupakan halaman yang akan muncul saat pengguna memilih menu data sampah. Pada halaman ini terdapat data sampah yang dikelola oleh bank sampah yaitu nama, satuan, dan harga sampah. Pengguna dapat memilih tombol *Expand* atau memilih nama sampah yang ada untuk melihat data sampah tersebut. Terdapat dua buah tombol pada tiap sampah yaitu tombol Ubah Data Sampah dengan *icon* pensil dan tombol Hapus Sampah dengan *icon* tong sampah. Terdapat juga tombol Tambah Sampah berupa *floating button* di bagian bawah kanan halaman dan berwarna hijau.

3.3 Perhitungan Logika Fuzzy

Proses perhitungan nilai rekomendasi bank sampah pada aplikasi berdasarkan logika *fuzzy* adalah sebagai berikut:

3.3.1 Fuzzifikasi

Pada proses fuzzifikasi, digunakan fungsi bentuk "bahu" untuk menentukan nilai keanggotaan sebuah variabel. Berikut adalah grafik yang digunakan untuk menentukan nilai keanggotaan:



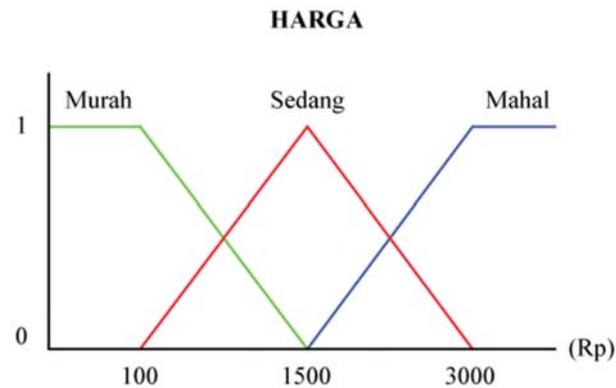
Gambar 10. Fungsi Keanggotaan Variabel Jarak

Variabel *input* jarak pengguna ke bank sampah dalam satuan kilometer dibagi menjadi tiga himpunan *fuzzy* yaitu dekat (nol hingga dua kilometer), sedang (0,1 hingga lima kilometer), dan jauh (lebih dari dua kilometer). Jarak tersebut digunakan untuk mendapatkan derajat keanggotaan variabel jarak dengan persamaan sebagai berikut:

$$\mu_{dekat} = \begin{cases} 1, & x \leq 0.1 \\ \frac{2-x}{2-0.1}, & 0.1 < x < 2 \\ 0, & x \geq 2 \end{cases} \quad (1)$$

$$\mu_{sedang} = \begin{cases} 1, & x = 2 \\ \frac{x-0.1}{2-0.1}, & 0.1 < x < 2 \\ \frac{5-x}{5-2}, & 2 < x < 5 \\ 0, & x \leq 0.1 \text{ atau } x \geq 5 \end{cases} \quad (2)$$

$$\mu_{jauh} = \begin{cases} 1, & x \geq 5 \\ \frac{x-2}{5-2}, & 2 < x < 5 \\ 0, & x \leq 2 \end{cases} \quad (3)$$



Gambar 11. Fungsi Keanggotaan Variabel Harga

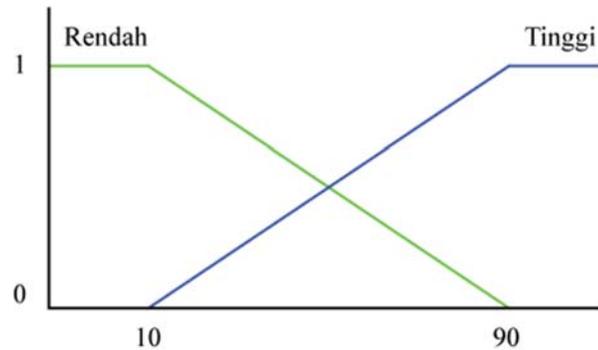
Variabel *input* harga sampah dibagi menjadi tiga himpunan *fuzzy* yaitu murah (Rp. 0 hingga Rp. 1.500), sedang (Rp. 100 hingga Rp. 3000), dan mahal (lebih dari Rp. 1.500). Harga tersebut digunakan untuk mendapatkan derajat keanggotaan variabel harga dengan persamaan sebagai berikut:

$$\mu_{murah} = \begin{cases} 1, & x \leq 100 \\ \frac{1500-x}{1500-100}, & 100 < x < 1500 \\ 0, & x \geq 1500 \end{cases} \quad (4)$$

$$\mu_{sedang} = \begin{cases} 1, & x = 1500 \\ \frac{x-100}{1500-100}, & 100 < x < 1500 \\ \frac{3000-x}{3000-1500}, & 1500 < x < 3000 \\ 0, & x \leq 100 \text{ atau } x \geq 3000 \end{cases} \quad (5)$$

$$\mu_{mahal} = \begin{cases} 1, & x \geq 3000 \\ \frac{x-1500}{3000-1500}, & 1500 < x < 3000 \\ 0, & x \leq 1500 \end{cases} \quad (6)$$

NILAI REKOMENDASI



Gambar 12. Fungsi Keanggotaan Nilai Rekomendasi

Variabel *output* nilai rekomendasi dibagi menjadi dua himpunan *fuzzy*, yaitu rendah dan tinggi. Fungsi keanggotaan *output* yang digunakan untuk mendapatkan nilai rekomendasi masing-masing bank sampah didapatkan menggunakan persamaan sebagai berikut:

$$\mu_{rendah} = \begin{cases} 1, & z \leq 10 \\ \frac{90-z}{90-10}, & 10 < z < 90 \\ 0, & z \geq 90 \end{cases} \quad (7)$$

$$\mu_{tinggi} = \begin{cases} 0, & z \leq 10 \\ \frac{z-10}{90-10}, & 10 < z < 90 \\ 1, & z \geq 90 \end{cases} \quad (8)$$

3.3.2 Inferensi Model Tsukamoto

Berikut adalah aturan-aturan *fuzzy* yang diterapkan untuk perhitungan nilai rekomendasi bank sampah:

Tabel 1. Aturan Fuzzy Jarak

Jarak	Nilai Rekomendasi
Dekat	Tinggi
Sedang	Tinggi
Jauh	Rendah

Tabel 2. Aturan Fuzzy Harga

Jarak	Nilai Rekomendasi
Murah	Rendah
Sedang	Tinggi
Mahal	Tinggi

Tabel 3. Aturan Fuzzy Jarak dan Harga

Jarak	Harga	Nilai Rekomendasi
Dekat	Murah	Rendah
Dekat	Sedang	Tinggi
Dekat	Mahal	Tinggi
Sedang	Murah	Rendah
Sedang	Sedang	Rendah
Sedang	Mahal	Tinggi
Jauh	Murah	Rendah
Jauh	Sedang	Rendah
Jauh	Mahal	Tinggi

Jika jarak pengguna ke bank sampah adalah satu kilometer, maka variabel jarak memiliki derajat keanggotaan sebagai berikut:

$$\mu_{dekat} = \frac{2-1}{2-0.1} = 0.526315789$$

$$\mu_{sedang} = \frac{1-0.1}{2-0.1} = 0.473684211$$

Jika harga sampah adalah 450, maka variabel harga memiliki derajat keanggotaan sebagai berikut:

$$\mu_{murah} = \frac{1500-450}{1500-100} = 0.75$$

$$\mu_{sedang} = \frac{450-100}{1500-100} = 0.25$$

Berdasarkan empat data *fuzzy input* tersebut, maka didapatkan empat aturan dari Tabel 3 yang dapat diaplikasikan, yaitu:

- Jika jarak dekat (0.526315789) dan harga murah (0.75), maka nilai rekomendasi rendah.
- Jika jarak dekat (0.526315789) dan harga sedang (0.25), maka nilai rekomendasi tinggi.
- Jika jarak sedang (0.473684211) dan harga murah (0.75), maka nilai rekomendasi rendah.
- Jika jarak sedang (0.473684211) dan harga sedang (0.25), maka nilai rekomendasi rendah.

Untuk mendapatkan nilai rekomendasi, dibutuhkan derajat keanggotaan nilai rekomendasi berdasarkan derajat keanggotaan paling kecil dari jarak dan harga pada tiap aturan yang berlaku seperti pada persamaan berikut:

$$\mu(z) = \text{Min}(\mu_{\text{jarak}}; \mu_{\text{harga}}) \quad (9)$$

Dari keempat aturan tersebut, dihitung nilai rekomendasi dengan Persamaan 7 dan Persamaan 8 berdasarkan derajat keanggotaan menurut Persamaan 9:

- $\mu(z) = \text{Min}(0.526315789; 0.75) = 0.52631578$
 $z = -(0.526315789 * 80) + 90 = 47.8947369$
- $\mu(z) = \text{Min}(0.526315789; 0.25) = 0.25$
 $z = 0.25 * 80 + 10 = 30$
- $\mu(z) = \text{Min}(0.473684211; 0.75) = 0.473684211$
 $z = -(0.473684211 * 80) + 90 = 52.1052631$
- $\mu(z) = \text{Min}(0.473684211; 0.25) = 0.25$
 $z = -(0.25 * 80) + 90 = 70$

3.3.3 Defuzzifikasi Model Tsukamoto

Proses defuzzifikasi model Tsukamoto menggunakan metode *weighted average*, maka nilai rekomendasi bank sampah tersebut adalah sebagai berikut:

$$\text{Nilai rekomendasi} = \frac{\sum \mu(z) * z}{\sum \mu(z)} \quad (10)$$

$$\text{Nilai rekomendasi} = 49.9261311$$

Nilai rekomendasi tersebut merupakan nilai rekomendasi sebuah bank sampah berdasarkan jarak pengguna terhadap bank sampah dan dengan salah satu sampah yang ada di bank sampah tersebut. Apabila bank sampah tersebut memiliki lebih dari satu sampah, maka aplikasi akan menghitung rata-rata semua nilai rekomendasi berdasarkan jumlah jenis sampah yang ada pada bank sampah tersebut. Untuk penghitungan nilai rekomendasi hanya berdasarkan jarak atau harga sampah, maka proses inferensi menggunakan Tabel 2 atau Tabel 3. Setelah proses perhitungan nilai rekomendasi selesai, maka pengguna dapat memilih filter *chip* yang dibutuhkan dan aplikasi akan menyortir nilai rekomendasi tersebut.

4. KESIMPULAN

Berdasarkan hasil analisis perancangan, pembangunan dan pengujian aplikasi Bank Sampah Pontianak, dapat disimpulkan bahwa:

- Aplikasi dapat beroperasi dengan baik terhadap semua *smartphone* dengan sistem operasi minimal Android 7.0, memiliki izin akses lokasi, fitur lokasi yang aktif, dan memiliki akses internet.
- Jika admin bank sampah sudah mendaftarkan bank sampah yang dikelolanya ke dalam aplikasi Bank Sampah Pontianak, maka aplikasi dapat membantu nasabah dalam menginformasikan daftar bank sampah yang sudah terdaftar di aplikasi.
- Aplikasi dapat memberikan rekomendasi bank sampah kepada nasabah menggunakan logika *fuzzy* dengan sistem inferensi Tsukamoto berdasarkan jarak bank sampah, harga sampah, ataupun keduanya.

5. SARAN

Selama proses pembangunan aplikasi berlangsung, peneliti menyadari bahwa terdapat beberapa hal yang dapat dikembangkan dari aplikasi Bank Sampah Pontianak, yaitu:

- Membangun aplikasi serupa yang dapat berjalan tidak hanya pada Android, melainkan pada iOS dan *website* juga, sehingga jangkauan pengguna menjadi lebih luas dan informasi bank sampah dapat menyebar lebih cepat.
- Menambahkan fitur lain seperti penjemputan sampah agar aplikasi yang dirancang dapat digunakan dalam jangka waktu yang panjang dan meningkatkan interaksi pengguna dengan aplikasi.

UCAPAN TERIMA KASIH

Peneliti mengucapkan terima kasih kepada Bapak dan Ibu dosen, serta Staf Sekretariat Fakultas Teknologi Informasi Universitas Widya Dharma Pontianak yang telah memberi bimbingan dan dukungan terhadap penelitian ini.

DAFTAR PUSTAKA

- [1] Hartono, Yadi, Dwi Mardhia, Ieke Wulayan Ayu, dan Rudi Masniadi, (2020), *Pengelolaan dan Pemanfaatan Sampah Berbasis Rumah Tangga*, CV. Literasi Nusantara Abadi, Malang.
- [2] Masri, Rina Marina, dan Iskandar Muda Purwaamijaya, (2021), *Rekayasa Lingkungan*, Deepublish, Sleman.
- [3] Isnardi, Ikhsan, dan Rini Asmara, (2021), *Membangun Restfull API Menggunakan Codeigniter 4 dan Client Android dengan Bahasa Pemrograman Kotlin*, Pustaka Galeri Mandiri, Padang.
- [4] Aljundi, Muhammad Ihsan, dan Muhammad Azriel Akbar, (2018), *Kotlin Zero to Hero*, Udacoding, Tangerang Selatan.
- [5] Jubilee Enterprise, (2021), *Pengantar Pemrograman Kotlin*, PT. Elex Media Komputindo, Jakarta Pusat.
- [6] Azis, Haydal Abdul, (2019), Penerapan Algoritma Floyd-Warshall Untuk Menentukan Rute Terdekat Lokasi Service Handphone di Kota Yogyakarta, *Skripsi*, Program Pasca Sarjana Teknik Informatika, STMIK AKAKOM Yogyakarta, Yogyakarta.
- [7] Pratama, Muhammad Rafi Ulwa, (2020), *Membuat Aplikasi dengan Mapbox pada Android*, Udacoding, Tangerang Selatan.
- [8] Firly, Nadia, (2018), *Create Your Own Android Application*, PT. Elex Media Komputindo, Jakarta Pusat.
- [9] Hartini, Yustina Sri, Antonia Brigita Putri Lefanska, Amellya Anastasya Ursia, Dominikus Arif Budi Prasetyo, dan Budi Sugiharto, (2022), *Prosiding Seminar Nasional Sanata Dharma Berbagi "Pengembangan, Penerapan dan Pendidikan 'Sains dan Teknologi' Pasca Pandemi"*, Sanata Dharma University Press, Yogyakarta.
- [10] Husni, Nando Septian, dan Muhamad Alam Syah, (2019), *Membangun Ojek Online Menggunakan Firebase*, Udacoding, Tangerang Selatan.
- [11] Purnomo, Rosyana Fitria, Onno W. Purbo, dan RZ. Abd. Aziz, (2020), *Firestore: Membangun Aplikasi Berbasis Android*, Penerbit Andi, Sleman.
- [12] Rerung, Rintho Rante, (2020), *Database dengan Aplikasi Microsoft Access*, CV. Media Sains Indonesia, Bandung.
- [13] Fitri, Rahimi, (2020), *Pemrograman Basis Data Menggunakan MySQL*, Poliban Press, Banjarmasin.
- [14] Hakim, Galang Persada Nurani, Diah Septiyana, Ahmad Firdausi, Fajar Rahayu Ikhwannul Mariati, dan Setiyo Budiyanto, (2021), *Sistem Fuzzy: Panduan Lengkap Aplikatif*, Penerbit Andi, Sleman.
- [15] Handayani, Anik Nur, Hakkun Elmunsyah, dan Dyah Rosita, (2021), *Modul Ajar Fuzzy*, Ahlimedia Press, Malang.
- [16] Destriana, Rachmat, Syepri Maulana Husain, Nurdiana Handayani, Aditya Tegar Prahara Siswanto, (2021), *Diagram UML dalam Membuat Aplikasi Android Firebase "Studi Kasus Aplikasi Bank Sampah"*, Deepublish, Sleman.
- [17] Habibi, Roni, Dinda Anik Masruro, dan Nuha Hanifatul Khonsa, (2020), *Aplikasi Inventory Barang Menggunakan QR Code*, CV. Kreatif Industri Nusantara, Bandung.
- [18] Ardiansah, Irfan, dan Faisal Ahmad, (2021), *Perancangan Sistem Informasi Penjualan Kopi Berbasis Object Oriented Programming (OOP)*, CV. Cendekia Press, Bandung.