**APLIKASI PEMILIHAN GUDANG PT WARUNG ISLAMI BOGOR MENGGUNAKAN METODE FUZZY C-MEANS DAN TOPSIS**

****

**PROPOSAL**

**Khidir Zahid Muchtadiabillah**

**181106041100**

**Konsentrasi Sistem Informasi**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS TEKNIK DAN SAINS**

**UNIVERSITAS IBN KHALDUN BOGOR**

**BOGOR**

**2022**

# DAFTAR ISI

[DAFTAR ISI ii](#_Toc107672342)

[DAFTAR GAMBAR vi](#_Toc107672343)

[DAFTAR TABEL viii](#_Toc107672344)

[BAB I PENDAHULUAN 1](#_Toc107672345)

[1. 1 Latar Belakang 1](#_Toc107672346)

[1. 2 Rumusan Masalah 3](#_Toc107672347)

[1. 3 Tujuan Penelitian 3](#_Toc107672348)

[1. 4 Batasan Masalah 3](#_Toc107672349)

[1. 5 Manfaat Penelitian 4](#_Toc107672350)

[1. 6 Sistematika Penulisan 4](#_Toc107672351)

[BAB II TINJAUAN PUSTAKA 6](#_Toc107672352)

[2. 1 Studi Literatur 6](#_Toc107672353)

[2. 2 Landasan Teori 9](#_Toc107672354)

[2. 2. 1 Fuzzy Clustering 9](#_Toc107672355)

[2. 2. 2 Fuzzy C-Means (FCM) 9](#_Toc107672356)

[2. 2. 3 Sistem Penunjang Keputusan 11](#_Toc107672357)

[2. 2. 4 Technique for Order Preference by Similarity to Ideal Solution (TOPSIS) 11](#_Toc107672358)

[2. 2. 5 Data Flow Diagram (DFD) 13](#_Toc107672359)

[2. 2. 5. 1 Context Diagram 15](#_Toc107672360)

[2. 2. 6 Unified Modeling Language (UML) 15](#_Toc107672361)

[2. 2. 6. 1 Use Case Diagram 16](#_Toc107672362)

[2. 2. 6. 2 Class Diagram 18](#_Toc107672363)

[2. 2. 6. 3 Component Diagram 20](#_Toc107672364)

[2. 2. 6. 4 Deployment Diagram 21](#_Toc107672365)

[2. 2. 6. 5 Activity Diagram 22](#_Toc107672366)

[2. 2. 6. 6 Sequence Diagram 23](#_Toc107672367)

[2. 2. 7 BlackBox Testing 25](#_Toc107672368)

[BAB III METODE PENELITIAN 26](#_Toc107672369)

[3. 1 Waktu dan Tempat Penelitian 26](#_Toc107672370)

[3. 1. 1 Waktu 26](#_Toc107672371)

[3. 1. 2 Tempat 26](#_Toc107672372)

[3. 2 Alat dan Bahan 26](#_Toc107672373)

[3. 2. 1 Alat 26](#_Toc107672374)

[3. 2. 2 Bahan 27](#_Toc107672375)

[3. 3 Tahapan Penelitian 28](#_Toc107672376)

[3. 4 Model Pengembangan Sistem 29](#_Toc107672377)

[3. 4. 1 Communication 29](#_Toc107672378)

[3. 4. 2 Planning 29](#_Toc107672379)

[3. 4. 3 Modeling 30](#_Toc107672380)

[3. 4. 4 Construction 30](#_Toc107672381)

[3. 4. 5 Deployment 30](#_Toc107672382)

[BAB IV HASIL DAN PEMBAHASAN 31](#_Toc107672383)

[4. 1 Analisa Kebutuhan 31](#_Toc107672384)

[4. 1. 1 Analisa Proses Bisnis Lama 31](#_Toc107672385)

[4. 1. 2 Analisa Proses Bisnis Baru 32](#_Toc107672386)

[4. 2 Proses Perhitungan Fuzzy C-Means dan TOPSIS 33](#_Toc107672387)

[4. 2. 1 Pengelompokan Data Gudang Menggunakan Fuzzy C-Means 33](#_Toc107672388)

[4. 2. 2 Perangkingan Data Gudang Menggunakan TOPSIS 34](#_Toc107672389)

[4. 3 Desain Sistem 35](#_Toc107672390)

[4. 3. 1 Context Diagram 35](#_Toc107672391)

[4. 3. 2 Use Case Diagram 35](#_Toc107672392)

[4. 3. 3 Class Diagram 36](#_Toc107672393)

[4. 3. 4 Component Diagram 36](#_Toc107672394)

[4. 3. 5 Deployment Diagram 36](#_Toc107672395)

[4. 3. 6 Activity Diagram 37](#_Toc107672396)

[4. 2. 6. 1 Activity Diagram Login 37](#_Toc107672397)

[4. 2. 6. 2 Activity Diagram Logout 37](#_Toc107672398)

[4. 2. 6. 3 Activity Diagram Kelola Data Kriteria 38](#_Toc107672399)

[4. 2. 6. 4 Activity Diagram Kelola Data Gudang 38](#_Toc107672400)

[4. 2. 6. 5 Activity Diagram Kelola Data Hasil 39](#_Toc107672401)

[4. 3. 7 Sequence Diagram 39](#_Toc107672402)

[4. 2. 7. 1 Sequence Diagram Login 39](#_Toc107672403)

[4. 2. 7. 2 Sequence Diagram Logout 40](#_Toc107672404)

[4. 2. 7. 3 Sequence Diagram Tambah Data Kriteria 40](#_Toc107672405)

[4. 2. 7. 4 Sequence Diagram Ubah Data Kriteria 41](#_Toc107672406)

[4. 2. 7. 5 Sequence Diagram Hapus Data Kriteria 41](#_Toc107672407)

[4. 2. 7. 6 Sequence Diagram Lihat Data Kriteria 42](#_Toc107672408)

[4. 2. 7. 7 Sequence Diagram Tambah Data Gudang 42](#_Toc107672409)

[4. 2. 7. 8 Sequence Diagram Ubah Data Gudang 43](#_Toc107672410)

[4. 2. 7. 9 Sequence Diagram Hapus Data Gudang 43](#_Toc107672411)

[4. 2. 7. 10 Sequence Diagram Lihat Data Gudang 44](#_Toc107672412)

[4. 2. 7. 11 Sequence Diagram Proses Perhitungan 44](#_Toc107672413)

[4. 2. 7. 12 Sequence Diagram Lihat Hasil Perhitungan 45](#_Toc107672414)

[4. 4 Rancangan Basis Data 45](#_Toc107672415)

[4. 5 Hasil Rancangan User Interface 46](#_Toc107672416)

[4. 5. 1 User Interface Halaman Login 46](#_Toc107672417)

[4. 5. 2 User Interface Halaman Data Kriteria 46](#_Toc107672418)

[4. 5. 3 User Interface Halaman Tambah Kriteria 47](#_Toc107672419)

[4. 5. 4 User Interface Halaman Ubah Kriteria 48](#_Toc107672420)

[4. 5. 5 User Interface Halaman Data Gudang 48](#_Toc107672421)

[4. 5. 6 User Interface Halaman Tambah Gudang 49](#_Toc107672422)

[4. 5. 7 User Interface Halaman Ubah Gudang 50](#_Toc107672423)

[4. 5. 8 User Interface Halaman Data Hasil Perhitungan 50](#_Toc107672424)

[4. 5. 9 User Interface Halaman Proses Perhitungan 51](#_Toc107672425)

[4. 5. 10 User Interface Halaman Detail Hasil Perhitungan 52](#_Toc107672426)

[4. 6 Hasil Pengujian 52](#_Toc107672427)

[4. 6. 1 Hasil Pengujian BlackBox Testing 52](#_Toc107672428)

[DAFTAR PUSTAKA 56](#_Toc107672429)

# DAFTAR GAMBAR

[Gambar 2. 1 Diagram UML 16](#_Toc107047567)

[Gambar 3. 1 Model Waterfall 28](#_Toc107421302)

[Gambar 4. 1 Analisa Proses Bisnis Lama 31](#_Toc107421443)

[Gambar 4. 2 Analisa Proses Bisnis Baru 32](#_Toc107421444)

[Gambar 4. 3 Context Diagram 33](#_Toc107421445)

[Gambar 4. 4 Use Case Diagram 33](#_Toc107421446)

[Gambar 4. 5 Class Diagram 34](#_Toc107421447)

[Gambar 4. 6 Component Diagram 34](#_Toc107421448)

[Gambar 4. 7 Deployment Diagram 34](#_Toc107421449)

[Gambar 4. 8 Activity Diagram Login 35](#_Toc107421450)

[Gambar 4. 9 Activity Diagram Kelola Data Kriteria 35](#_Toc107421451)

[Gambar 4. 10 Activity Diagram Kelola Data Gudang 36](#_Toc107421452)

[Gambar 4. 11 Activity Diagram Proses Perhitungan 36](#_Toc107421453)

[Gambar 4. 12 Sequence Diagram Login 37](#_Toc107421454)

[Gambar 4. 13 Sequence Diagram Tambah Data Kriteria 37](#_Toc107421455)

[Gambar 4. 14 Sequence Diagram Ubah Data Kriteria 38](#_Toc107421456)

[Gambar 4. 15 Sequence Diagram Hapus Data Kriteria 38](#_Toc107421457)

[Gambar 4. 16 Sequence Diagram Lihat Data Kriteria 39](#_Toc107421458)

[Gambar 4. 17 Sequence Diagram Tambah Data Gudang 39](#_Toc107421459)

[Gambar 4. 18 Sequence Diagram Ubah Data Gudang 40](#_Toc107421460)

[Gambar 4. 19 Sequence Diagram Hapus Data Gudang 40](#_Toc107421461)

[Gambar 4. 20 Sequence Diagram Lihat Data Gudang 41](#_Toc107421462)

[Gambar 4. 21 Sequence Diagram Proses Perhitungan 41](#_Toc107421463)

[Gambar 4. 22 Sequence Diagram Lihat Hasil Perhitungan 42](#_Toc107421464)

[Gambar 4. 23 User Interface Halaman Login 43](#_Toc107421465)

[Gambar 4. 24 User Interface Halaman Data Kriteria 44](#_Toc107421466)

[Gambar 4. 25 User Interface Halaman Tambah Kriteria 44](#_Toc107421467)

[Gambar 4. 26 User Interface Halaman Ubah Kriteria 45](#_Toc107421468)

[Gambar 4. 27 User Interface Halaman Data Gudang 45](#_Toc107421469)

[Gambar 4. 28 User Interface Halaman Tambah Gudang 46](#_Toc107421470)

[Gambar 4. 29 User Interface Halaman Ubah Gudang 46](#_Toc107421471)

[Gambar 4. 30 User Interface Halaman Data Hasil Perhitungan 47](#_Toc107421472)

[Gambar 4. 31 User Interface Halaman Proses Perhitungan 47](#_Toc107421473)

[Gambar 4. 32 User Interface Halaman Detail Hasil Perhitungan 48](#_Toc107421474)

# DAFTAR TABEL

[Tabel 2. 1 Studi Literatur 6](#_Toc107421475)

[Tabel 2. 2 Notasi-notasi pada DFD 13](#_Toc107421476)

[Tabel 2. 3 Simbol-simbol pada Use Case Diagram 17](#_Toc107421477)

[Tabel 2. 4 Simbol-simbol pada Class Diagram 19](#_Toc107421478)

[Tabel 2. 5 Simbol-simbol pada Component Diagram 20](#_Toc107421479)

[Tabel 2. 6 Simbol-simbol pada Deployment Diagram 21](#_Toc107421480)

[Tabel 2. 7 Simbol-simbol pada Activity Diagram 22](#_Toc107421481)

[Tabel 2. 8 Simbol-simbol pada Sequence Diagram 23](#_Toc107421482)

[Tabel 3. 1 Data Gudang 27](#_Toc107047605)

[Tabel 4. 1 Matriks awal 34](#_Toc108168422)

[Tabel 4. 2 Pusat Cluster Iterasi ke 1 37](#_Toc108168423)

[Tabel 4. 3 Fungsi Obyektif Iterasi ke 1 38](#_Toc108168424)

[Tabel 4. 4 Matriks Partisi Iterasi ke 1 40](#_Toc108168425)

[Tabel 4. 5 Pusat Cluster Iterasi ke 10 41](#_Toc108168426)

[Tabel 4. 6 Matriks baru 42](#_Toc108168427)

[Tabel 4. 7 Data Gudang Cluster 1 43](#_Toc108168428)

[Tabel 4. 8 Hasil Pengujian BlackBox Testing 62](#_Toc108168429)

# BAB I PENDAHULUAN

## Latar Belakang

PT Warung Islami Bogor adalah salah satu perusahaan berkembang di Indonesia dengan bisnis yang berfokus pada sektor perdagangan produk kemasan industri. Berawal dari sebuah ide untuk membantu rekan-rekan pebisnis/produsen akan sulitnya mendapatkan kemasan produk dengan bahan botol plastik untuk menutupi kebutuhan produksi mereka, maka pada tahun 2012 PT Warung Islami Bogor merintis perusahaan dari warung yang berlokasi di Bogor, hingga menjadi sebuah perusahaan nasional yang terus berkembang. PT Warung Islami Bogor merupakan spesialis dalam mendistribusikan kemasan produk botol plastik dengan jaringan distribusi menyebar keseluruh wilayah Indonesia, baik melalui *channel* tradisional maupun modern berbasis digital dan internet. Dalam pendistribusiannya, PT Warung Islami Bogor membutuhkan gudang untuk menampung barang yang datang dari *supplier* dan barang tersebut akan dikirimkan kembali ke toko-toko milik PT Warung Islami Bogor yang tersebar di berbagai kota yang ada di Indonesia.

Gudang adalah suatu tempat atau bangunan yang dipergunakan untuk menimbun, menyimpan barang, baik berupa bahan baku (*raw material*), barang setengah jadi (*work in process*) atau barang jadi (*finished product*)[1]. Gudang merupakan komponen penting dari rantai pasokan modern. Rantai pasokan melibatkan kegiatan dalam berbagai tahap: *sourcing*, produksi, dan distribusi barang, dari penanganan bahan baku dan barang dalam proses hingga produk jadi. Gudang dapat digambarkan sebagai bagian dari suatu sistem logistik sebuah perusahaan yang berfungsi untuk menyimpan produk dan menyediakan informasi mengenai status serta kondisi material/persediaan yang disimpan di gudang, sehingga informasi tersebut selalu *up-to-date* dan mudah diakses oleh siapa pun yang berkepentingan[2].

Pemilihan gudang saat ini masih bersifat manual yakni staff *general affair* (GA) melakukan survey lapangan dan mencatat data gudang kedalam catatan, lalu memberikan catatan tersebut ke *Supervisor* GA, dan *Supervisor* GA meneruskan ke Manager Operasional untuk dibahas bersama dengan dewan direksi. Pemilihan gudang harus membandingkan satu persatu data gudang yang sudah didata dan pemilihannya bersifat subjektif. Untuk itu, diperlukan sebuah sistem yang dapat mempermudah perusahaan dalam menentukan gudang mana yang akan disewa.

Fuzzy clustering adalah salah satu teknik untuk menentukan cluster optimal dalam suatu ruang vektor yang didasarkan pada bentuk normal *Euclidian* untuk jarak antar vektor. Ada beberapa algoritma *clustering* data, salah satu diantaranya adalah Fuzzy C-Means. Fuzzy C-Means (FCM) adalah suatu teknik pengclusteran data yang mana keberadaan tiap-tiap titik data dalam suatu cluster ditentukan oleh derajat keanggotaan. Konsep dasar FCM, pertama kali adalah menentukan pusat *cluster* yang akan menandai lokasi rata-rata untuk tiap-tiap *cluster*. Dengan metode ini, pusat *cluster* dan derajat keanggotaan selalu diperbaiki secara berulang sehingga dapat dilihat bahwa pusat *cluster* akan bergerak menuju lokasi yang tepat[3].

Setelah data dikelompokkan, maka data tersebut harus diranking. Ada beberapa metode perangkingan data, salah satu diantaranya adalah TOPSIS. TOPSIS adalah konsep dimana alternatif terpilih yang terbaik tidak hanya memiliki jarak terpendek dari solusi ideal positif, namun juga memiliki jarak terpanjang dari solusi ideal negatif[4].

Metode gabungan meminimalkan upaya komputasi yang diperlukan untuk mencapai peringkat karena mengembangkannya dalam dua tahap (mencapai tujuan "efisiensi" dari model MCDA yang baik). Pertama, algoritma FCM digunakan untuk mengklasifikasikan objek data yang serupa ke dalam *cluster*. Kemudian TOPSIS digunakan untuk mengembangkan peringkat untuk *cluster* dan memisahkan peringkat untuk objek data di dalam *cluster*[5].

Setelah data diranking, maka data tersebut akan dipresentasikan oleh manajer kepada jajaran manajemen untuk dijadikan pertimbangan. Untuk mempermudah hal tersebut, diperlukan sebuah sistem yang akurat agar pengambilan keputusan dapat diambil dengan tepat.

## Rumusan Masalah

Berdasarkan latar belakang di atas, maka penulis merumuskan beberapa masalah sebagai berikut :

1. Bagaimana cara mengelompokkan data gudang menggunakan metode Fuzzy C-Means?
2. Bagaimana cara membuat peringkat dari hasil pengelompokan data gudang menggunakan metode TOPSIS?
3. Bagaimana cara membuat aplikasi pemilihan gudang?

## Tujuan Penelitian

Berdasarkan rumusan masalah di atas, maka didapat tujuan dari penelitian ini yaitu sebagai berikut :

1. Mengelompokkan data gudang menggunakan metode Fuzzy C-Means.
2. Membuat peringkat dari hasil pengelompokan data gudang menggunakan metode TOPSIS.
3. Merancang dan membangun aplikasi pemilihan gudang berbasis *website*.

## Batasan Masalah

Pembatasan suatu masalah digunakan untuk menghindari adanya penyimpangan maupun pelebaran pokok masalah supaya penelitian tersebut lebih terarah dan memudahkan dalam pembahasan sehingga tujuan penelitian akan tercapai. Beberapa batasan masalah dalam penelitian ini adalah sebagai berikut :

1. Data Gudang yang akan digunakan periode 2022 dengan data hanya 12 gudang.
2. Data Kriteria yang akan digunakan yaitu harga (juta/tahun), luas bangunan (m2), jarak dari *head office* (HO) (km), luas parkir (m2), dan berapa lantai.

## Manfaat Penelitian

Manfaat yang dapat diperoleh dari penelitian ini adalah :

1. Diharapkan aplikasi ini dapat berguna untuk PT Warung Islami Bogor dalam menentukan gudang mana yang akan dipilih.
2. Diharapkan dapat bermanfaat bagi pihak lain yang akan mengembangkan penelitian ini.

## Sistematika Penulisan

Secara garis besar dalam penulisan skripsi ini, penulis membagi ke dalam lima bab dengan tujuan untuk memudahkan penulis dalam membahasnya. Adapun sistematika penulisannya diuraikan sebagai berikut :

**BAB I PENDAHULUAN**

Dalam bab ini diuraikan tentang latar belakang, rumusan masalah, tujuan penelitian, batasan masalah, manfaat penelitian, dan sistematika penulisan.

**BAB II LANDASAN TEORI**

Dalam bab ini diuraikan tentang teori dasar yang dipergunakan dalam penyusunan skripsi, antara lain: pengertian sistem pendukung keputusan, Fuzzy C-Means, TOPSIS, serta teori pendukung lainnya.

**BAB III METODE PENELITIAN**

Dalam bab ini diuraikan secara rinci metodologi dalam pengembangan sistem.

**BAB IV HASIL DAN PEMBAHASAN**

Dalam bab ini akan diuraikan dan dibahas hasil perancangan sistem yang dibuat untuk diimplementasikan di PT Warung Islami Bogor.

**BAB V KESIMPULAN DAN SARAN**

Bab terakhir ini berisi kesimpulan yang dapat diambil dari inti pembahasan pada bab-bab sebelumnya serta saran-saran yang diharapkan berguna bagi pengembang sistem ini di masa mendatang.

# BAB II TINJAUAN PUSTAKA

## Studi Literatur

Dalam penelitian ini akan digunakan empat penelitian terdahulu yang nantinya mendukung penelitian yang akan dilakukan, dimana studi literatur yang diambil adalah:

Tabel 2. 1 Studi Literatur

| **No** | **Penulis** | **Tahun** | **Judul** | **Pembahasan** |
| --- | --- | --- | --- | --- |
| 1 | Giovan Meidy Susanto, Sandy Kosasi, David, Gat, dan Susanti M. Kuway | 2020 | Sistem Referensi Pemilihan Smartphone Android Dengan Metode Fuzzy C-Means dan TOPSIS | dalam penelitian ini membahas algoritma Fuzzy C-Means dan TOPSIS untuk memberikan referensi pemilihan *smartphone* android. Hasil dari perhitungan Fuzzy C-Means dan TOPSIS mampu melakukan pengelompokan *smartphone* android menjadi tiga *cluster*, lalu dilakukan pengujian menggunakan *White-Box Testing* dan hasilnya seluruh *function* yang ada pada perangkat lunak dapat berjalan dengan baik. Adapun kelemahan dari sistem ini adalah tidak membahas mengenai kebutuhan *lifestyle* dalam memilih android[6]. |
| 2 | Erlita Faridatul Himah dan Raden Sulaiman | 2021 | Implementasi Metode Fuzzy C-Means dan TOPSIS dalam Evaluasi Kinerja Keuangan Perusahaan Perbankan di Indonesia Berdasarkan Rasio Keuangan | dalam penelitian ini membahas kinerja keuangan pada sektor perbankan yang perlu dilakukan evaluasi kinerja perusahaan perbankan di Indonesia berdasarkan rasio keuangan untuk mengetahui keadaan finansial dari suatu perusahaan. Hasil dari pembahasan diatas adalah perangkingan *cluster* menggunakan metode Fuzzy C-Means dan TOPSIS, pada perhitungan evaluasi kinerja perusahaan menggunakan metode Fuzzy C-Means dan TOPSIS diperoleh indeks perpindahan yaitu jumlah rata - rata perusahaan yang berada pada klasifikasi yang tidak sesuai sebesar 1.23 dan tingkat kesalahan sebesar 25.65%[7]. |
| 3 | Deta Oktariani, Desi Andreswari, dan Yudi Setiawan | 2017 | Sistem Pendukung Keputusan dalam Pemetaan Prioritas Perbaikan Jalan dan Jembatan Nasional di Provinsi Bengkulu Menggunakan Metode TOPSIS dan Fuzzy C-Means | dalam penelitian ini membahas bahwa banyak pertimbangan untuk memperbaiki jalan dan jembatan nasional, untuk mendukung pengambilan keputusan prioritas perbaikan jalan dan jembatan nasional dibutuhkan bentuk pemetaan dari kondisi jalan dan jembatan nasional tersebut. Hasil dari pembahasan diatas adalah sistem ini telah berhasil mengelompokan data jalan dan jembatan nasional dengan algoritma Fuzzy C-Means *clustering*, dan perangkingan data dengan algoritma TOPSIS. Berhasil menampilkan kondisi jalan dan jembatan dalam bentuk pemetaan sebagai salah satu pendukung keputusan P2JN[8]. |
| 4 | Christian Sri Kusuma Aditya | 2020 | Pemilihan Kalimat Representatif dengan Pengintegrasian Fuzzy C-Means *Clustering* Dan TOPSIS (FCM-TOPSIS) untuk Peringkasan Dokumen | dalam penelitian ini membahas dokumen teks merupakan sumber informasi yang sangat dibutuhkan namun koleksi dokumen dalam jumlah besar dapat memberikan dampak negatif bagi pengguna yang membutuhkan waktu relatif lama untuk memilah. Ringkasan dibutuhkan untuk inti sari tanpa harus merubah konteks informasi dari sebuah dokumen teks. Hasil dari pembahasan diatas adalah penelitian ini mengintegrasikan antara Fuzzy C-Means dengan metode TOPSIS untuk mendapatkan hasil ringkasan dari dokumen teks[9]. |

Berdasarkan hasil studi literatur, yang menjadi persamaan dari penelitian ini dengan penelitian sebelumnya adalah metode yang digunakan yaitu Fuzzy C-Means dan TOPSIS dan yang membedakan penelitian ini dengan penelitian sebelumnya adalah objek penelitian yaitu pemilihan gudang dengan kriteria harga, luas bangunan, jarak dari HO, luas parkir, dan lantai.

## Landasan Teori

### Fuzzy Clustering

Fuzzy clustering adalah salah satu teknik untuk menentukan cluster optimal dalam suatu ruang vektor yang didasarkan pada bentuk normal *Euclidian* untuk jarak antar vektor. Fuzzy clustering sangat berguna bagi pemodelan fuzzy terutama dalam mengidentifikasi aturan-aturan fuzzy. Ada beberapa algoritma clustering data, salah satu diantaranya adalah Fuzzy C-Means (FCM)[3].

### Fuzzy C-Means (FCM)

Fuzzy C-Means (FCM) adalah suatu teknik pengclusteran data yang mana keberadaan tiap-tiap titik data dalam suatu *cluster* ditentukan oleh derajat keanggotaan. Teknik ini pertama kali diperkenalkan oleh Jim Bezdek pada tahun 1981.

Konsep dasar FCM, pertama kali adalah menentukan pusat cluster yang akan menandai lokasi rata-rata untuk tiap-tiap cluster. Pada kondisi awal, pusat cluster ini masih belum akurat. Tiap-tiap titik data memiliki derajat keanggotaan untuk tiap-tiap cluster. Dengan cara memperbaiki pusat cluster dan derajat keanggotaan tiap-tiap titik data secara berulang, maka akan dapat dilihat bahwa pusat cluster akan bergerak menuju lokasi yang tepat. Perulangan ini didasarkan pada minimisasi fungsi obyektif yang menggambarkan jarak dari titik data yang diberikan ke pusat cluster yang terbobot oleh derajat keanggotaan titik data tersebut. Perhitungan algoritma Fuzzy C-Means adalah sebagai berikut [3]:

1. Input data yang akan dicluster X, berupa matriks berukuran n x m (n = jumlah sampel data, m = atribut setiap data). Xij = data sampel ke-i (i=1,2,..., n), atribut ke-j (j=1,2,..., m).
2. Tentukan:
3. Jumlah *cluster* = ;
4. Pangkat = ;
5. Maksimum iterasi = Maxlter;
6. Error terkecil yang diharapkan = ;
7. Fungsi obyektif awal P0 = ;
8. Iterasi awal = ;
9. Bangkitkan bilangan random , i=1,2,...,n; k=1,2,...,c; sebagai elemen-elemen matriks partisi awal U.

Hitung jumlah setiap kolom (atribut):

(2. 1)

dengan j=1,2,...m.

Hitung:

(2. 2)

1. Hitung pusat *cluster* ke-k: , dengan k=1,2,...,c; dan j=1,2,...,m.

(2. 3)

1. Hitung fungsi obyektif pada iterasi ke-t, Pt

(2. 4)

1. Hitung perubahan matriks partisi:

(2. 5)

dengan: i = 1,2,...,n; dan k = 1,2,...,c.

1. Cek kondisi berhenti:

Jika: atau maka berhenti;

Jika tidak: , ulangi langkah ke-4.

### Sistem Penunjang Keputusan

Sistem Penunjang Keputusan adalah sebuah sistem yang dimaksudkan untuk mendukung para pengambil keputusan manajerial dalam situasi keputusan semiterstruktur. Sistem Penunjang Keputusan dimaksudkan untuk menjadi alat bantu bagi para pengambil keputusan untuk memperluas kapabilitas mereka, namun tidak untuk menggantikan penilaian mereka. Sistem Penunjang Keputusan ditujukan untuk keputusan-keputusan yang memerlukan penilaian atau pada keputusan-keputuan yang sama sekali tidak dapat didukung oleh algoritma[10].

### Technique for Order Preference by Similarity to Ideal Solution (TOPSIS)

*Technique for Order Preference by Similarity to Ideal Solution* (TOPSIS) adalah konsep dimana alternatif terpilih yang terbaik tidak hanya memiliki jarak terpendek dari solusi ideal positif, namun juga memiliki jarak terpanjang dari solusi ideal negatif (Hwang, 1981) (Zeleny, 1982). Konsep ini banyak digunakan pada beberapa model MADM untuk menyelesaikan masalah keputusan secara praktis (Hwang, 1993)(Liang, 1999)(Yeh, 2000). Hal ini disebabkan: konsepnya sederhana dan mudah dipahami; komputasinya efisien; dan memiliki kemampuan untuk mengukur kinerja relatif dari alternatif-alternatif keputusan dalam bentuk matematis yang sederhana.

Secara umum, prosedur TOPSIS mengikuti langkah-langkah sebagai berikut[4]:

1. Membuat matriks keputusan yang ternormalisasi;
2. Membuat matriks keputusan yang ternormalisasi terbobot;
3. Menentukan matriks solusi ideal positif & matriks solusi ideal negatif;
4. Menentukan jarak antara nilai setiap alternatif dengan matriks solusi ideal positif & matriks solusi ideal negatif;
5. Menentukan nilai preferensi untuk setiap alternatif.

TOPSIS membutuhkan rating kinerja setiap alternatif Ai, pada setiap kriteria Cj, yang ternormalisasi, yaitu:

dengan i = 1,2,...,m; dan j=1,2,...,n. (2. 6)

Solusi ideal positif A+ dan solusi ideal negatif A- dapat ditentukan berdasarkan rating bobot ternormalisasi (yij) sebagai:

dengan i=1,2,...,m; dan j=1,2,...,n. (2. 7)

(2. 8)

(2. 9)

dengan

(2. 10)

(2. 11)

j=1,2,...,n.

Jarak antara alternatif Ai dengan solusi ideal positif dirumuskan sebagai:

i = 1,2,...,m. (2. 12)

Jarak antara alternatif Ai dengan solusi ideal negatif dirumuskan sebagai:

i = 1,2,...,m. (2. 13)

Nilai preferensi untuk setiap alternatif (Vi) diberikan sebagai:

i = 1,2,...,m. (2. 14)

Nilai Vi yang lebih besar menunjukkan bahwa alternatif Ai lebih dipilih.

### Data Flow Diagram (DFD)

*Data Flow Diagram* (DFD) adalah representasi grafik untuk perancangan pemgrograman terstruktur yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengalir dari masukan (*input*) dan keluaran (*output*)[11]. Berikut merupakan notasi-notasi pada DFD :

Tabel 2. 2 Notasi-notasi pada DFD

(Sumber: Rosa A. S. dan M. Shalahuddin, 2014)

| **Notasi** | **Keterangan** |
| --- | --- |
|  | Proses atau fungsi atau prosedur; pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harusnya menjadi fungsi atau prosedur di dalam kode program.  Nama yang diberikan pada sebuah proses biasanya berupa kata kerja. |
|  | *File* atau basis data atau penyimpanan (*storage*); pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harusnya dibuat menjadi tabel-tabel basis data yang dibutuhkan, tabel-tabel ini juga harusn sesuai dengan perancangan tabel-tabel pada basis data (*Entity Relationship Diagram* (ERD), *Conceptual Data Model* (CDM), *Physical Data* Model (PDM)).  Nama yang diberikan pada sebuah penyimpanan biasanya kata benda. |
|  | Entitas luar (*external entity*) atau masukan (*input*) atau keluaran (*output*) atau orang yang memakai/berinteraksi dengan perangkat lunak yang dimodelkan atau sistem lain yang terkait dengan aliran data dari sistem yang dimodelkan.  Nama yang digunakan pada masukan (*input*) atau keluaran (*output*) biasanya berupa kata benda. |
|  | Aliran data; merupakan data yang dikirim antar proses, dari penyimpanan ke proses, atau dari proses ke masukan (*input*) atau keluaran (*output*).  Nama yang digunakan pada aliran data biasanya berupa kata benda, dapat diawali dengan kata data misalnya "data gudang" atau tanpa kata data misalnya "gudang". |

#### Context Diagram

*Context Diagram* atau DFD *Level* 0 adalah suatu entitas tunggal yang berinteraksi dengan orang maupun sistem lain. *Context Diagram* digunakan untuk menggambarkan interaksi antara sistem yang akan dikembangkan dengan entitas luar[11].

### Unified Modeling Language (UML)

*Unified Modeling Language* (UML) adalah bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung. UML adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek. UML hanya berfungsi untuk melakukan pemodelan. Jadi penggunaan UML tidak terbatas pada metodologi tertentu, meskipun pada kenyataannya UML paling banyak digunakan pada metodologi berorientasi objek. Pada UML 2.3 terdiri dari 13 macam diagram yang dikelompokkan dalam 3 kategori[11]. Pembagian kategori dan macam-macam diagram tersebut dapat dilihat pada gambar di bawah.

Gambar 2. 1 Diagram UML

(Sumber: Rosa A. S. dan M. Shalahuddin, 2014)

Berikut ini penjelasan singkat dari pembagian kategori tersebut[11].

* *Structure diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang dimodelkan.
* *Behavior diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi pada sebuah sistem.
* *Interaction diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan interaksi sistem dengan sistem lain maupun interaksi antar subsistem pada suatu sistem.

#### Use Case Diagram

*Use Case Diagram* adalah pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu[11]. Berikut merupakan simbol-simbol yang ada pada diagram *use case* :

Tabel 2. 3 Simbol-simbol pada Use Case Diagram

(Sumber: Rosa A. S. dan M. Shalahuddin, 2014)

| **Simbol** | **Deskripsi** |
| --- | --- |
| *Use case* | Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal di awal frase nama *use case*. |
| Aktor (*actor*) | Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri. |
| Asosiasi (*association*) | Komunikasi antara aktor dan *use case* yang berpartisipasi pada *use case* atau *use case* memiliki interaksi dengan aktor. |
| Ekstensi (*Extend*) | Relasi *use case* tambahan ke sebuah *use case* dimana *use case* yang ditambahkan dapat berdiri sendiri walau tanpa *use case* tambahan itu; mirip dengan prinsip *inheritance* pada pemrograman berorientasi objek; biasanya *use case* tambahan memiliki nama depan yang sama dengan *use case* yang ditambahkan. |
| Generalisasi (*generalization*) | Hubungan generalisasi dan spesialisasi (umum - khusus) antara dua buah *use case* dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya. |
| Menggunakan (*include/uses*) | Relasi *use case* tambahan ke sebuah *use case* di mana *use case* yang ditambahkan memerlukan *use case* ini untuk menjalankan fungsinya atau sebagai syarat dijalankan *use case* ini. |

#### Class Diagram

*Class diagram* adalah struktur sistem yang menggambarkan dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi[11].

* Atribut merupakan variabel-variabel yang dimiliki oleh suatu kelas.
* Metode atau operasi adalah fungsi-fungsi yang dimiliki oleh suatu kelas.

Susunan struktur kelas yang baik pada diagram kelas sebaiknya memiliki jenis-jenis kelas berikut[11]:

* Kelas main

Kelas yang memiliki fungsi awal dieksekusi ketika sistem dijalankan.

* Kelas yang menangani tampilan sistem (*view*)

Kelas yang mendefinisikan dan mengatur tampilan ke pengguna.

* Kelas yang diambil dari pendefinisian *use case* (*controller*)

Kelas yang menangani fungsi-fungsi yang harus ada diambil dari pendefinisian *use case*, kelas ini biasanya disebut dengan kelas proses yang menangani proses bisnis pada perangkat lunak.

* Kelas yang diambil dari pendefinisian data (*model*)

Kelas yang digunakan untuk memegang atau membungkus data menjadi sebuah kesatuan yang diambil maupun akan disimpan ke basis data. Semua tabel yang dibuat di basis data dapat dijadikan kelas, namun untuk tabel dari hasil relasi atau atribut *multivalue* pada ERD dapat dijadikan kelas tersendiri dapat juga tidak asalkan pengaksesannya dapat dipertanggungjawabkan atau tetap ada di dalam perancangan kelas.

Berikut merupakan simbol-simbol yang ada pada *class diagram*:

Tabel 2. 4 Simbol-simbol pada Class Diagram

(Sumber: Rosa A. S. dan M. Shalahuddin, 2014)

| **Simbol** | **Deskripsi** |
| --- | --- |
| Kelas (*class*) | Kelas pada struktur sistem. |
| Antarmuka (*interface*) | Sama dengan konsep *interface* dalam pemrograman berorientasi objek. |
| Asosiasi (*association*) | Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan *multiplicity*. |
| Asosiasi berarah (*directed association*) | Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan *multiplicity*. |
| Generalisasi (*generalization*) | Relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus). |
| Kebergantungan (*dependency*) | Relasi antar kelas dengan makna kebergantungan antar kelas. |
| Agregasi (*aggregation*) | Relasi antar kelas dengan makna semua-bagian (*whole-part*). |

#### Component Diagram

*Component Diagram* adalah diagram yang dibuat untuk menunjukkan organisasi dan ketergantuan diantara kumpulan komponen dalam sebuah sistem. Diagram komponen fokus pada komponen sistem yang dibutuhkan dan ada di dalam sistem. Komponen dasar yang biasanya ada dalam suatu sistem adalah sebagai berikut[11]:

* Komponen *user interface* yang menangani tampilan.
* Komponen *business processing* yang menangani fungsi-fungsi proses bisnis.
* Komponen *data* yang menangani manipulasi data.
* Komponen *security* yang manangani keamanan sistem.

Berikut merupakan simbol-simbol yang ada pada *component diagram*:

Tabel 2. 5 Simbol-simbol pada Component Diagram

(Sumber: Rosa A. S. dan M. Shalahuddin, 2014)

| **Simbol** | **Deskripsi** |
| --- | --- |
| *Package* | *Package* merupakan sebuah bungkusan dari satu atau lebih komponen. |
| Komponen (*component*) | Komponen sistem. |
| Kebergantungan (*dependency*) | Kebergantungan antar komponen, arah panah mengarah pada komponen yang dipakai. |
| Antarmuka (*interface*) | Sama dengan konsep *interface* pada pemrograman berorientasi objek, yaitu sebagai antarmuka komponen agar tidak mengakses langsung komponen. |
| *Link* | Relasi antar komponen. |

#### Deployment Diagram

*Deployment Diagram* adalah diagram yang menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi[11]. Berikut merupakan simbol-simbol yang ada pada *deployment diagram*:

Tabel 2. 6 Simbol-simbol pada Deployment Diagram

(Sumber: Rosa A. S. dan M. Shalahuddin, 2014)

| **Simbol** | **Deskripsi** |
| --- | --- |
| *Package* | *Package* merupakan sebuah bungkusan dari satu atau lebih node. |
| *Node* | Biasanya mengacu pada perangkat keras (*hardware*), perangkat lunak yang tidak dibuat sendiri (*software*), jika di dalam node disertakan komponen untuk mengkonsistenkan rancangan maka komponen yang diikutsertakan harus sesuai dengan komponen yang telah didefinisikan sebelumnya pada diagram komponen. |
| Kebergantungan (*dependency*) | Kebergantungan antar *node*, arah panah mengarah pada *node* yang dipakai |
| *Link* | Relasi antar node |

#### Activity Diagram

*Activity Diagram* adalah diagram yang menggambarkan aliran kerja (*workflow*) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak[11]. Berikut merupakan simbol-simbol yang ada pada *activity diagram*:

Tabel 2. 7 Simbol-simbol pada Activity Diagram

(Sumber: Rosa A. S. dan M. Shalahuddin, 2014)

| **Simbol** | **Deskripsi** |
| --- | --- |
| Status awal (*start*) | Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal. |
| Aktivitas (*activity*) | Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja. |
| Percabangan (*decision*) | Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu. |
| Penggabungan (*join*) | Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu. |
| Status akhir (*finish*) | Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuhan status akhir. |
| *Swimlane*  atau | Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi |

#### Sequence Diagram

*Sequence Diagram* adalah diagram yang menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Membuat diagram sekuen juga dibutuhkan untuk melihat skenario yang ada pada *use case*.

Banyaknya diagram sekuen yang harus digambar adalah minimal sebanyak pendefinisian *use case* yang memiliki proses sendiri atau yang penting semua *use case* yang telah didefinisikan interaksi jalannya pesan sudah dicakup pada diagram sekuen[11]. Berikut merupakan simbol-simbol yang ada pada *sequence diagram*:

Tabel 2. 8 Simbol-simbol pada Sequence Diagram

(Sumber: Rosa A. S. dan M. Shalahuddin, 2014)

| **Simbol** | **Deskripsi** |
| --- | --- |
| Aktor (actor*)*    atau | Orang, proses, atau sistem lain yang berinteraksi dengan sistem infomasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri. |
| Garis hidup (*lifeline*) | Menyatakan kehidupan suatu objek. |
| Objek (*object*) | Menyatakan objek yang berinteraksi pesan. |
| Waktu aktif (activation) | Menyaatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya. |
| Pesan tipe *create* | Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat. |
| Pesan tipe *call* | Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri, arah panah mengarah pada objek yang memiliki operasi/metode. |
| Pesan tipe *send* | Menyatakan bahwa suatu objek mengirimkan data/masukan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirimi. |
| Pesan tipe *return* | Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian. |
| Pesan tipe *destroy* | Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri. |

### BlackBox Testing

*BlackBox Testing* adalah pengujian perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Pengujian dimaksudkan untuk mengetahui apakah fungsi-fungsi, masukan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan.

Pengujian *BlackBox* dilakukan dengan membuat kasus uji yang bersifat mencoba semua fungsi dengan memakai perangkat lunak apakah sesuai dengan spesifikasi yang dibutuhkan. Kasus uji yang dibuat untuk melakukan pengujian kotak hitam harus dibuat dengan kasus benar dan kasus salah[11].

# BAB III METODE PENELITIAN

## Waktu dan Tempat Penelitian

### Waktu

Penelitian dimulai dari tanggal 30 Maret 2022 sampai dengan tanggal 30 Juni 2022.

### Tempat

Penelitian dilakukan di PT Warung Islami Bogor yang beralamat di Ruko Galeria No.12, Jl. Raya Pemda Karadenan Pangkalan KM 4, Karadenan, Cibinong, Kab. Bogor, Jawa Barat 16913.

## Alat dan Bahan

### Alat

1. **Perangkat Keras (*Hardware*)**
2. Processor Intel Core i3-6100
3. VGA Intel HD Graphics 530
4. RAM 8 GB
5. SSD 256 GB
6. **Perangkat Lunak (*Software*)**
7. Windows 10
8. Microsoft Office 365
9. XAMPP
10. Visual Studio Code
11. Google Chrome

### Bahan

Bahan yang digunakan pada penelitian ini adalah data gudang di PT Warung Islami Bogor meliputi harga, luas bangunan, jarak dari HO, luas parkir, dan lantai.

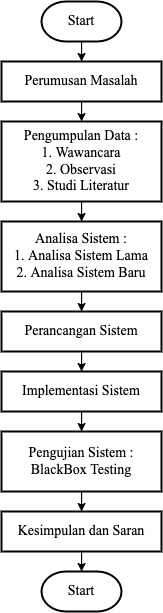
Tabel 3. 1 Data Gudang

(Sumber: PT Warung Islami Bogor, 2022)

|  | **Data Alternatif** | **Data Kriteria** | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| **No** | **Nama** | **Harga (Juta**  **/tahun)** | **Luas Bangunan (m2)** | **Jarak dari HO (KM)** | **Luas Parkir (m2)** | **Lantai** |
| 1 | Depan Pom Bensin Villa Bogor Indah | 200 | 500 | 3,3 | 150 | 2 |
| 2 | Depan Puri Nirwana / Samping Gudang Alfa | 200 | 300 | 1,5 | 150 | 2 |
| 3 | Komplek Pergudangan KM 36 | 355 | 600 | 11,9 | 300 | 2 |
| 4 | Gudang 88 | 460 | 1500 | 2,8 | 500 | 1 |
| 5 | Bengkel LAS | 180 | 300 | 2,8 | 150 | 1 |
| 6 | GUDANG CCC | 279 | 1200 | 3,3 | 500 | 2 |
| 7 | PT. GRASINDO | 653 | 1600 | 3,2 | 150 | 1 |
| 8 | Gudang Kacang Garuda | 921 | 1650 | 0,8 | 400 | 1 |
| 9 | Komplek Pergudangan KEDUNG HALANG | 406 | 720 | 11,1 | 360 | 2 |
| 10 | PT. Nugratama Dayamitra | 868 | 1800 | 10,1 | 500 | 1 |
| 11 | Gudang Nurdhin | 607 | 600 | 12,7 | 700 | 1 |
| 12 | Gudang Hj. Nasrul | 962 | 1400 | 15,2 | 546 | 1 |
| **Keterangan Atribut** | | Cost | Benefit | Cost | Benefit | Benefit |
| **Bobot** | | 4 | 3 | 2 | 2 | 2 |

## Tahapan Penelitian

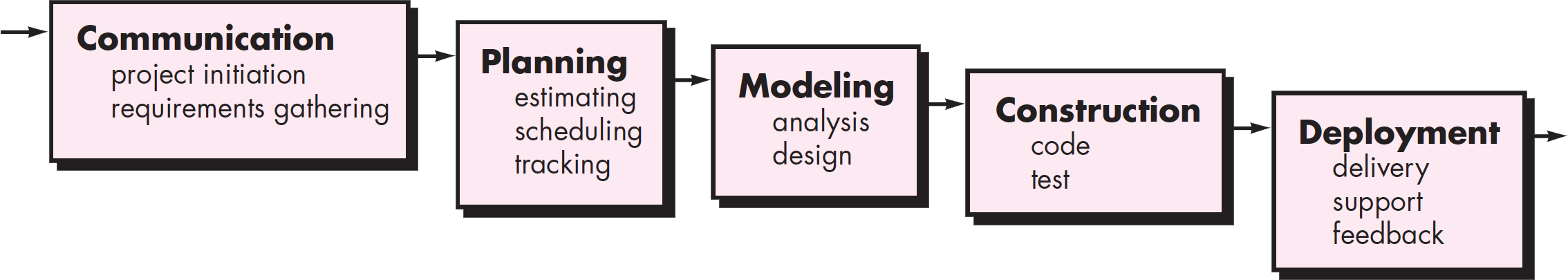
Metode penelitian yang diterapkan pada penelitian ini adalah dengan melakukan wawancara dan observasi langsung ke PT Warung Islami Bogor. Untuk lebih jelasnya dapat dilihat pada Gambar 3.1.



Gambar 3. 1 Tahapan Penelitian

## Model Pengembangan Sistem

Model pengembangan sistem yang digunakan adalah menggunakan model *waterfall*. Model *waterfall* adalah model klasik yang bersifat sistematis dan berurutan untuk pengembangan perangkat lunak yang dimulai dengan spesifikasi kebutuhan *customer* dan berkembang melalui perencanaan, pemodelan, konstruksi, dan penyebaran, yang berpuncak pada dukungan berkelanjutan dari perangkat lunak yang telah selesai[12]. Berikut merupakan gambar model *waterfall*:



Gambar 3. 2 Model Waterfall

(Sumber: R. S. Pressman, 2010)

### Communication

Sebelum memulai pekerjaan yang bersifat teknis, sangat diperlukan adanya komunikasi dengan *customer* demi memahami dan mencapai tujuan yang ingin dicapai. Hasil dari komunikasi tersebut adalah inisialisasi proyek, seperti menganalisis permasalahan yang dihadapi dan mengumpulkan data-data yang diperlukan, serta membantu mendefinisikan fitur dan fungsi perangkat lunak[12].

### Planning

Tahap berikutnya adalah tahapan perencanaan yang menjelaskan tentang estimasi tugas-tugas teknis yang akan dilakukan, resiko-resiko yang dapat terjadi, sumber daya yang diperlukan dalam membuat sistem, produk kerja yang ingin dihasilkan, dan penjadwalan kerja yang akan dilaksanakan[12].

### Modeling

Tahap desain sistem ini menggunakan diagram UML seperti *class diagram*, *component diagram*, *deployment diagram*, *use case diagram*, *activity diagram*, dan *sequence diagram*.

### Construction

Pada tahap ini dilakukan proses implementasi metode Fuzzy C-Means dan TOPSIS ke dalam kode program menggunakan bahasa pemrograman PHP dengan framework Laravel. Metode pengujian yang diterapkan pada penelitian ini menggunakan metode *BlackBox Testing*.

### Deployment

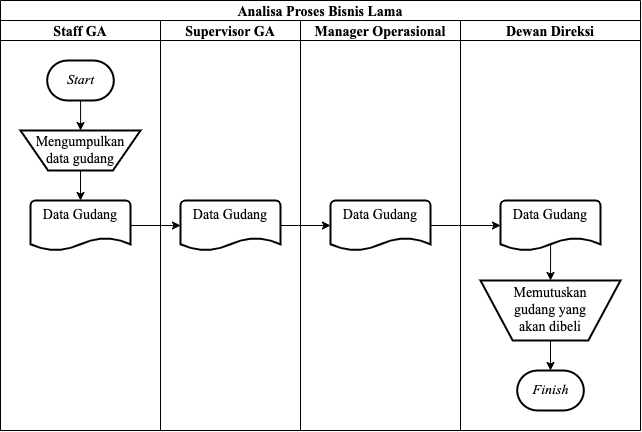
Tahapan penyebaran merupakan tahapan implementasi perangkat lunak ke *customer*, evaluasi perangkat lunak, dan pengembangan perangkat lunak berdasarkan umpan balik yang diberikan agar sistem dapat tetap berjalan dan berkembang sesuai dengan fungsinya[12].

# BAB IV HASIL DAN PEMBAHASAN

## Analisa Kebutuhan

### Analisa Proses Bisnis Lama

Berdasarkan hasil wawancara dan observasi tentang proses bisnis pada PT Warung Islami Bogor dapat diketahui bahwa PT Warung Islami Bogor sampai saat ini untuk memilih gudang yang akan dibeli masih menggunakan cara manual, yaitu dengan cara membandingkan antar alternatif secara subjektif. Alur sistem yang sedang berjalan pada saat ini dapat dilihat pada Gambar 4.1.

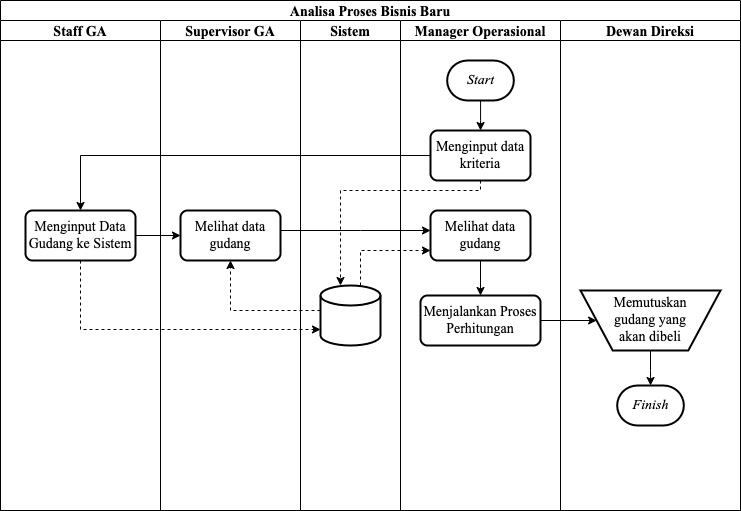


Gambar 4. 1 Analisa Proses Bisnis Lama

Karena penilaian bersifat subjektif sehingga dikhawatirkan mengakibatkan ketidaktepatan perusahaan dalam memutuskan apakah gudang yang dipilih sudah tepat atau belum. Adanya ketidaktepatan dalam mengambil keputusan berdampak pada biaya yang telah dikeluarkan perusahaan, dan juga mempengaruhi arus keluar masuk barang. Kemudian banyaknya alternatif menyebabkan proses pemilihan gudang membutuhkan waktu yang lama sehingga kurang efisien.

### Analisa Proses Bisnis Baru

Berdasarkan proses bisnis yang lama, maka akan diterapkan metode *clustering* Fuzzy C-Means dan metode TOPSIS. Metode Fuzzy C-Means digunakan untuk mengenali ciri-ciri yang mirip dengan yang perusahaan harapkan dan TOPSIS digunakan untuk mendapatkan alternatif terbaik berdasarkan nilai preferensi yang diberikan. Alur sistem yang akan dikembangkan dapat dilihat pada Gambar 4.2.



Gambar 4. 2 Analisa Proses Bisnis Baru

## Proses Perhitungan Fuzzy C-Means dan TOPSIS

### Pengelompokan Data Gudang Menggunakan Fuzzy C-Means

Berikut ini proses pengelompokan data gudang menggunakan metode Fuzzy C-Means :

1. Input data gudang yang akan dicluster berdasarkan Tabel 3.1, berupa matriks xij sebagai berikut :

* i adalah data gudang berjumlah 12 (n=12)
* j adalah data kriteria berjumlah 5 (m=5)

1. Inisialisasi :
2. Jumlah *cluster* = 2;
3. Pangkat = 2;
4. Maksimum iterasi = 100;
5. Error terkecil yang diharapkan = 10-2;
6. Fungsi obyektif awal = 0;
7. Iterasi awal = 1;
8. Bangkitkan bilangan random ()

Cara menghitung matriks awal :

1. Bangkitkan nilai acak matriks partisi

1. Hitung jumlah setiap baris (atribut) berdasarkan persamaan (2.1)

Contoh baris ke 1 :

1. Hitung elemen matriks berdasarkan persamaan (2.2)

adalah jumlah derajat keanggotaan perbaris = 1 :

Sehingga didapat nilai matriks partisi awal baris ke 1 adalah :

Demikian seterusnya untuk baris ke 2 sampai 12, sehingga didapat matriks partisi awal sebagai berikut :

Tabel 4. 1 Matriks awal

|  |  |
| --- | --- |
| 0,292  0,125  0,620  0,514  0,112  0,921  0,205  0,545  0,129  0,410  0,836  0,334 | 0,708  0,875  0,380  0,486  0,888  0,079  0,795  0,455  0,871  0,590  0,164  0,666 |

1. Hitung pusat *cluster* () berdasarkan persamaan (2.3)

Untuk pusat *cluster* ke 1 :

Diketahui , demikian seterusnya sampai gudang ke 12

Sehingga

Hitung nilai gudang ke 1 untuk kriteria ke 1 :

, demikian seterusnya sampai gudang ke 12

Sehingga

Hitung nilai gudang ke 1 untuk kriteria ke 2 :

, demikian seterusnya sampai gudang ke 12

Sehingga

Hitung nilai gudang ke 1 untuk kriteria ke 3 :

, demikian seterusnya sampai gudang ke 12

Sehingga

Hitung nilai gudang ke 1 untuk kriteria ke 4 :

, demikian seterusnya sampai gudang ke 12

Sehingga

Hitung nilai gudang ke 1 untuk kriteria ke 5 :

, demikian seterusnya sampai gudang ke 12

Sehingga

Hitung nilai pusat *cluster* ke 1 :

Demikian seterusnya untuk pusat *cluster* ke 2, sehingga didapat pusat *cluster* sebagai berikut :

Tabel 4. 2 Pusat Cluster Iterasi ke 1

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *Cluster* 1 | 510,301 | 1067,708 | 7,227 | 493,760 | 1,458 |
| *Cluster* 2 | 455,540 | 890,706 | 5,889 | 279,577 | 1,448 |

1. Hitung fungsi obyektif berdasarkan persamaan (2.4)

Untuk cluster ke 1, hitung nilai gudang ke 1 untuk kriteria ke 1 :

, demikian seterusnya sampai gudang ke 12

Hitung nilai gudang ke 1 untuk kriteria ke 2 :

, demikian seterusnya sampai gudang ke 12

Hitung nilai gudang ke 1 untuk kriteria ke 3 :

, demikian seterusnya sampai gudang ke 12

Hitung nilai gudang ke 1 untuk kriteria ke 4 :

, demikian seterusnya sampai gudang ke 12

Hitung nilai gudang ke 1 untuk kriteria ke 5 :

, demikian seterusnya sampai gudang ke 12

Hitung jumlah nilai kriteria berdasarkan persamaan

Gudang 1 : , demikian seterusnya sampai gudang ke 12

Hitung nilai jumlah kriteria dikali nilai matriks U berdasarkan persamaan

Gudang ke 1 : , demikian seterusnya sampai gudang ke 12

Kemudian jumlahkan berdasarkan persamaan

Dengan cara yang sama untuk *cluster* ke 2, sehingga pada *cluster* ke 2 didapat :

Kemudian hitung nilai fungsi obyektif () berdasarkan persamaan (2.4)

Sehingga didapat fungsi obyektif sebagai berikut :

Tabel 4. 3 Fungsi Obyektif Iterasi ke 1

|  |  |
| --- | --- |
| 45715,076 | 117726,384 |
| 12551,624 | 330051,563 |
| 107755,691 | 13737,417 |
| 50103,957 | 99067,354 |
| 10194,818 | 348477,762 |
| 60226,013 | 1104,689 |
| 17662,799 | 353573,876 |
| 153477,066 | 167159,649 |
| 2496,727 | 28886,814 |
| 111846,019 | 363532,067 |
| 189050,165 | 7669,505 |
| 35392,487 | 260336,977 |

1. Hitung matriks partisi () baru berdasarkan persamaan (2.5)

Cara menghitung matriks partisi U baru :

Dari persamaan , dapat dicari nilai matriks sebagai berikut :

Untuk Gudang ke 1 *cluster* ke 1

Demikian seterusnya sampai gudang ke 12

Untuk Gudang ke 1 *cluster* ke 2

Demikian seterusnya sampai gudang ke 12

Hitung jumlah baris untuk gudang ke 1 berdasarkan persamaan

Demikian seterusnya sampai gudang ke 12

Kemudian hitung nilai matriks baru berdasarkan persamaan (2.5)

Demikian seterusnya untuk setiap elemen matriks , sehingga didapat matriks partisi sebagai berikut :

Tabel 4. 4 Matriks Partisi Iterasi ke 1

|  |  |  |
| --- | --- | --- |
| 0,00000186 | 0,00000426 | 0,00000612 |
| 0,00000124 | 0,00000232 | 0,00000356 |
| 0,00000357 | 0,00001052 | 0,00001408 |
| 0,00000528 | 0,00000238 | 0,00000766 |
| 0,00000122 | 0,00000226 | 0,00000349 |
| 0,00001407 | 0,00000570 | 0,00001977 |
| 0,00000237 | 0,00000179 | 0,00000416 |
| 0,00000194 | 0,00000124 | 0,00000317 |
| 0,00000668 | 0,00002625 | 0,00003293 |
| 0,00000151 | 0,00000096 | 0,00000246 |
| 0,00000369 | 0,00000352 | 0,00000721 |
| 0,00000315 | 0,00000170 | 0,00000486 |

1. Cek kondisi berhenti

Pada iterasi ke 1 kondisi belum terpenuhi karena , dan , maka proses dilanjutkan ke iterasi 2 dan sampai kondisi terpenuhi. Pada kasus ini proses berhenti pada iterasi ke 10 karena kondisi sudah terpenuhi. Pada percobaan lain mungkin akan didapat posisi *cluster* yang berbeda dikarenakan inisialisasi awal matriks partisi yang dilakukan secara acak, namun tidak mempengaruhi hasil akhir.

Pada iterasi ke 10 diperoleh pusat *cluster* dan matriks baru sebagai berikut:

Tabel 4. 5 Pusat Cluster Iterasi ke 10

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *Cluster* 1 | 739,094 | 1.554,222 | 6,214 | 427,274 | 1,074 |
| *Cluster* 2 | 307,721 | 521,034 | 6,791 | 284,492 | 1,715 |

Dari Tabel 4.5 didapat informasi sebagai berikut :

1. *Cluster* 1 berisi data gudang yang memiliki harga sekitar Rp 739.094.000; luas bangunan rata-rata 1.554,222 m2; jarak dari HO sekitar 6,214 km; luas parkir rata-rata 427,274 m2; dan lantai 1. *Cluster* ini merupakan *cluster* dengan harga lebih mahal tapi dengan luas bangunan lebih besar.
2. *Cluster* 1 berisi data gudang yang memiliki harga sekitar Rp 307.721.000; luas bangunan rata-rata 521,034 m2; jarak dari HO sekitar 6,791 km; luas parkir rata-rata 284,492 m2; dan lantai 2. *Cluster* ini merupakan *cluster* dengan harga lebih murah tapi dengan luas bangunan lebih kecil.

Tabel 4. 6 Matriks baru

| **No** | **Data Gudang** | **Derajat keanggotaan data pada *cluster*** | | **Derajat keanggotaan terbesar pada *cluster*** |
| --- | --- | --- | --- | --- |
| **Nama** | **1** | **2** |
| 1 | Depan Pom Bensin Villa Bogor Indah | 0,020 | 0,980 | 2 |
| 2 | Depan Puri Nirwana / Samping Gudang Alfa | 0,039 | 0,961 | 2 |
| 3 | Komplek Pergudangan KM 36 | 0,008 | 0,992 | 2 |
| 4 | Gudang 88 | 0,923 | 0,077 | 1 |
| 5 | Bengkel LAS | 0,041 | 0,959 | 2 |
| 6 | GUDANG CCC | 0,597 | 0,403 | 1 |
| 7 | PT. GRASINDO | 0,938 | 0,062 | 1 |
| 8 | Gudang Kacang Garuda | 0,975 | 0,025 | 1 |
| 9 | Komplek Pergudangan KEDUNG HALANG | 0,063 | 0,937 | 2 |
| 10 | PT. Nugratama Dayamitra | 0,960 | 0,040 | 1 |
| 11 | Gudang Nurdhin | 0,211 | 0,789 | 2 |
| 12 | Gudang Hj. Nasrul | 0,935 | 0,065 | 1 |

Tabel matriks baru menunjukkan derajat keanggotaan data gudang pada setiap *cluster*. Derajat keanggotaan terbesar menunjukkan kecenderungan tertinggi data gudang untuk masuk menjadi anggota *cluster*. Dari Tabel 4.6 diperoleh informasi sebagai berikut :

1. Berdasarkan derajat keanggotaan terbesar pada *cluster* 1, maka terdapat 6 data gudang dalam *cluster* 1, yaitu gudang ke : 4, 6, 7, 8, 10, dan 12.
2. Berdasarkan derajat keanggotaan terbesar pada *cluster* 2, maka terdapat 6 data gudang dalam *cluster* 2, yaitu gudang ke : 1, 2, 3, 5, 9, dan 11.

### Perangkingan Data Gudang Menggunakan TOPSIS

Tahap selanjutnya yaitu melakukan perangkingan menggunakan metode TOPSIS. Berikut langkah-langkah yang akan dilakukan :

1. Membuat matriks keputusan yang ternormalisasi

Tabel 4. 7 Data Gudang Cluster 1

| **No** | **Nama Gudang** | **Harga (Juta**  **/tahun)** | **Luas Bangunan (m2)** | **Jarak dari HO (KM)** | **Luas Parkir (m2)** | **Lantai** |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | Gudang 88 | 460 | 1500 | 2,8 | 500 | 1 |
| 2 | GUDANG CCC | 279 | 1200 | 3,3 | 500 | 2 |
| 3 | PT. GRASINDO | 653 | 1600 | 3,2 | 150 | 1 |
| 4 | Gudang Kacang Garuda | 921 | 1650 | 0,8 | 400 | 1 |
| 5 | PT. Nugratama Dayamitra | 868 | 1800 | 10,1 | 500 | 1 |
| 6 | Gudang Hj. Nasrul | 962 | 1400 | 15,2 | 546 | 1 |

Tabel 4. 8 Data Gudang Cluster 2

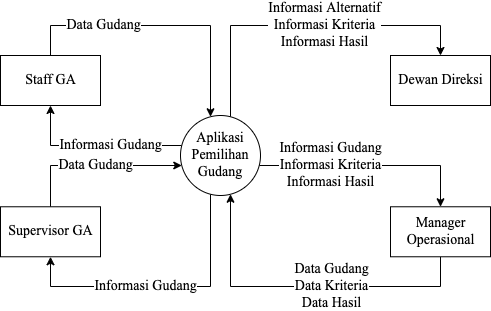
| **No** | **Nama Gudang** | **Harga (Juta**  **/tahun)** | **Luas Bangunan (m2)** | **Jarak dari HO (KM)** | **Luas Parkir (m2)** | **Lantai** |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | Depan Pom Bensin Villa Bogor Indah |  |  |  |  |  |
| 2 | Depan Puri Nirwana / Samping Gudang Alfa |  |  |  |  |  |
| 3 | Komplek Pergudangan KM 36 |  |  |  |  |  |
| 4 | Bengkel LAS |  |  |  |  |  |
| 5 | Komplek Pergudangan KEDUNG HALANG |  |  |  |  |  |
| 6 | Gudang Nurdhin |  |  |  |  |  |

1. Membuat matriks keputusan yang ternormalisasi terbobot;
2. Menentukan matriks solusi ideal positif & matriks solusi ideal negatif;
3. Menentukan jarak antara nilai setiap alternatif dengan matriks solusi ideal positif & matriks solusi ideal negatif;
4. Menentukan nilai preferensi untuk setiap alternatif

## Desain Sistem

### Context Diagram

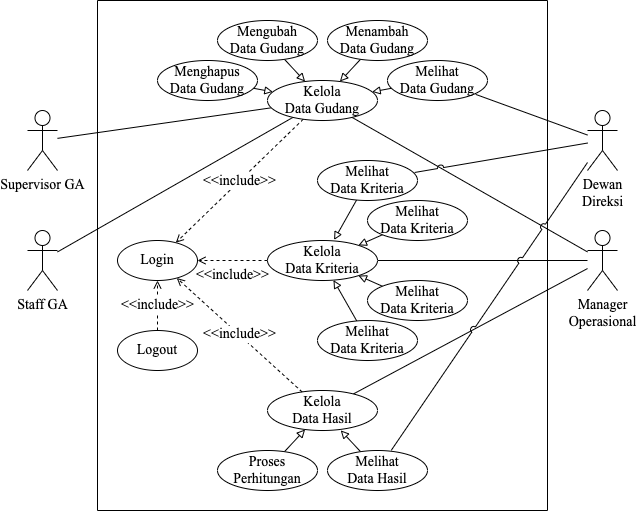
Berikut merupakan *context diagram* dari aplikasi pemilihan gudang :



Gambar 4. 3 Context Diagram

### Use Case Diagram

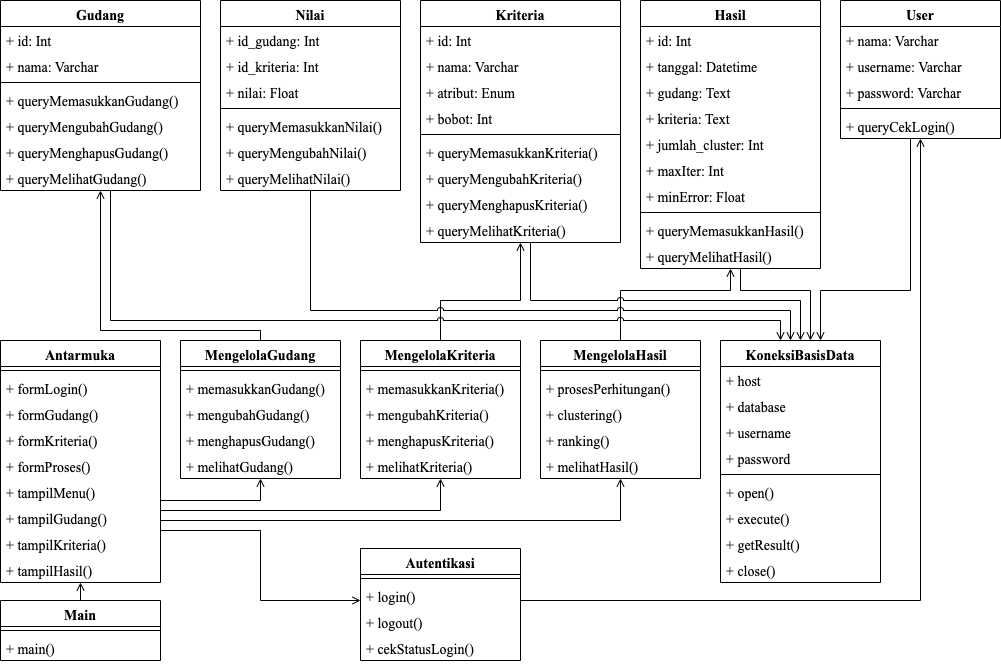
Berikut merupakan *use case diagram* dari aplikasi pemilihan gudang :



Gambar 4. 4 Use Case Diagram

### Class Diagram

Berikut merupakan *class diagram* dari aplikasi pemilihan gudang :



Gambar 4. 5 Class Diagram

### Component Diagram

Berikut merupakan *component diagram* dari aplikasi pemilihan gudang :

Gambar 4. 6 Component Diagram

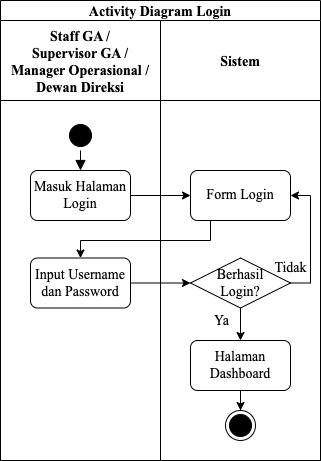
### Deployment Diagram

Berikut merupakan *deployment diagram* dari aplikasi pemilihan gudang :

Gambar 4. 7 Deployment Diagram

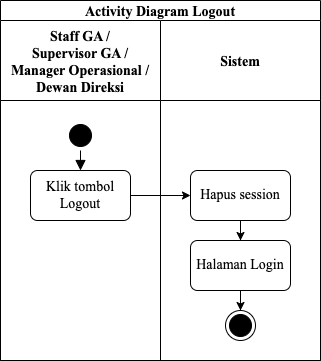
### Activity Diagram

#### Activity Diagram Login



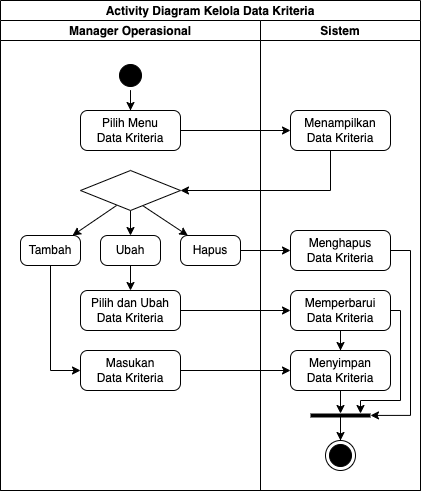
Gambar 4. 8 Activity Diagram Login

#### Activity Diagram Logout



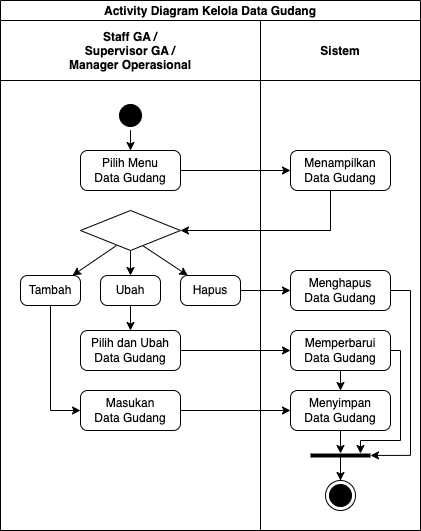
Gambar 4. 9 Activity Diagram Logout

#### Activity Diagram Kelola Data Kriteria



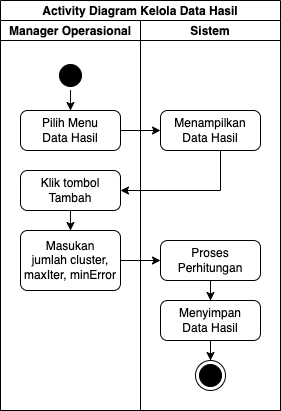
Gambar 4. 10 Activity Diagram Kelola Data Kriteria

#### Activity Diagram Kelola Data Gudang



Gambar 4. 11 Activity Diagram Kelola Data Gudang

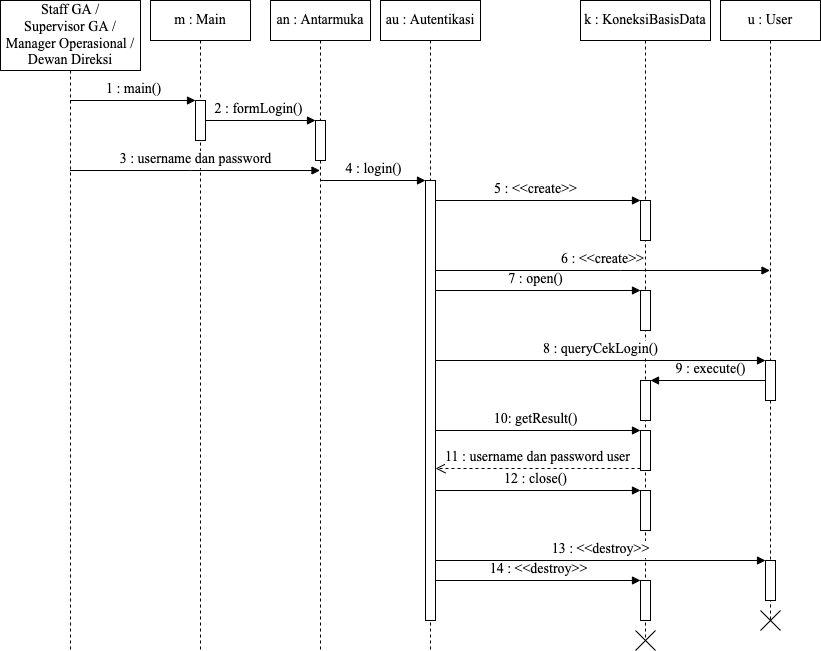
#### Activity Diagram Kelola Data Hasil



Gambar 4. 12 Activity Diagram Kelola Data Hasil

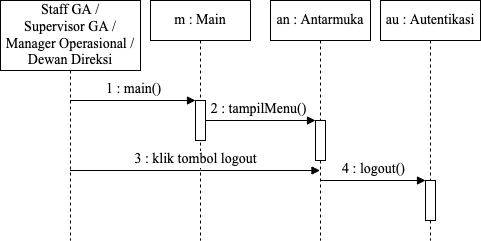
### Sequence Diagram

#### Sequence Diagram Login



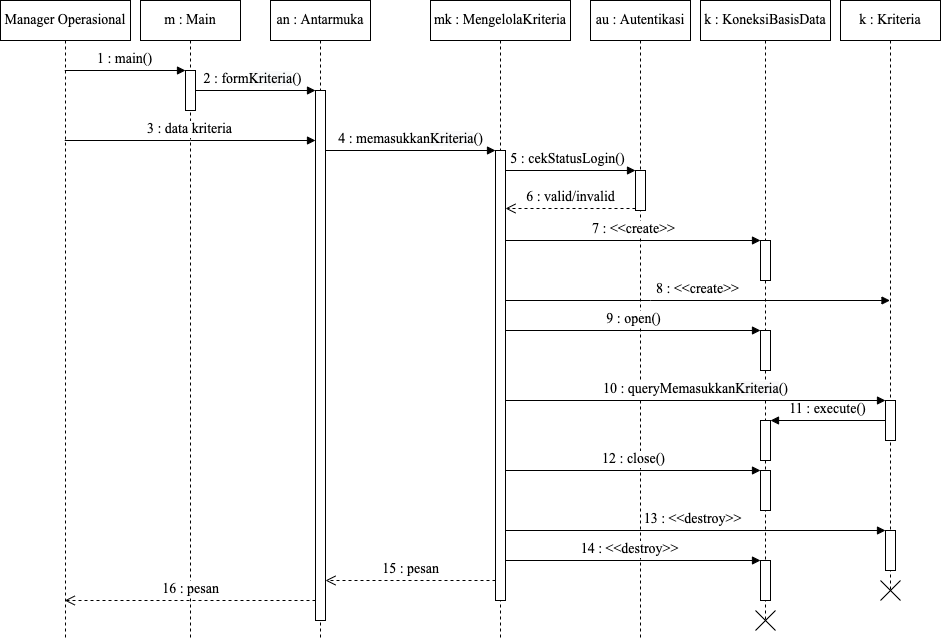
Gambar 4. 13 Sequence Diagram Login

#### Sequence Diagram Logout



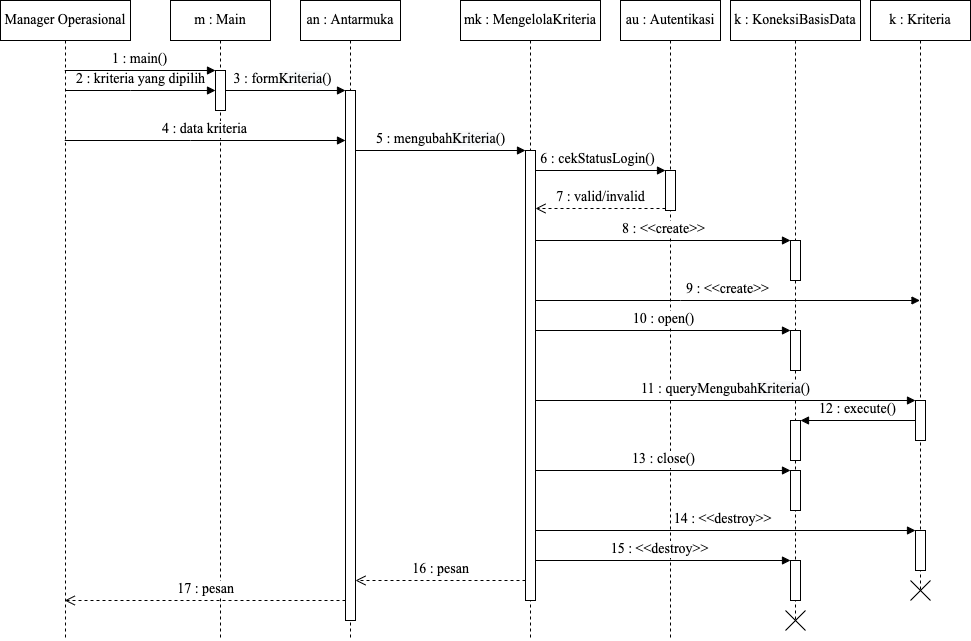
Gambar 4. 14 Sequence Diagram Logout

#### Sequence Diagram Tambah Data Kriteria



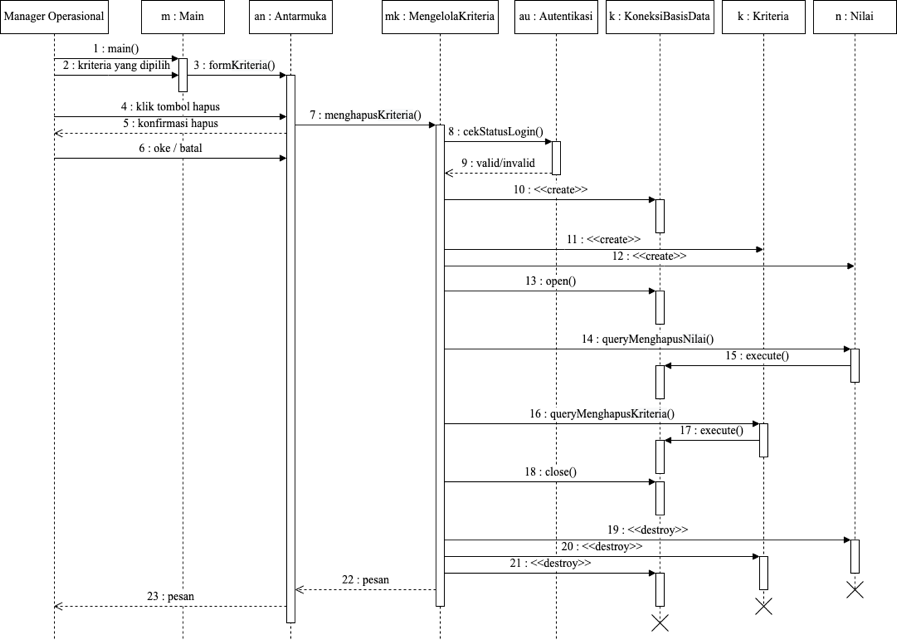
Gambar 4. 15 Sequence Diagram Tambah Data Kriteria

#### Sequence Diagram Ubah Data Kriteria



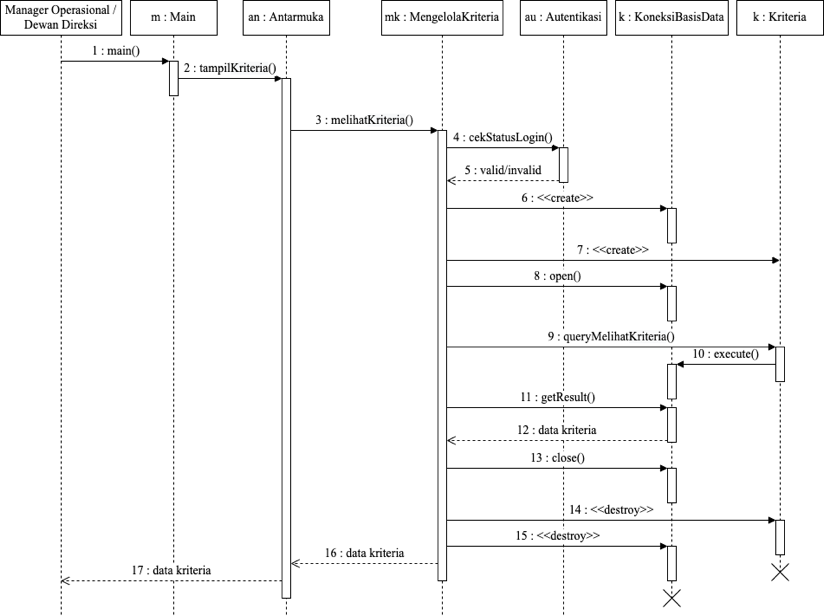
Gambar 4. 16 Sequence Diagram Ubah Data Kriteria

#### Sequence Diagram Hapus Data Kriteria



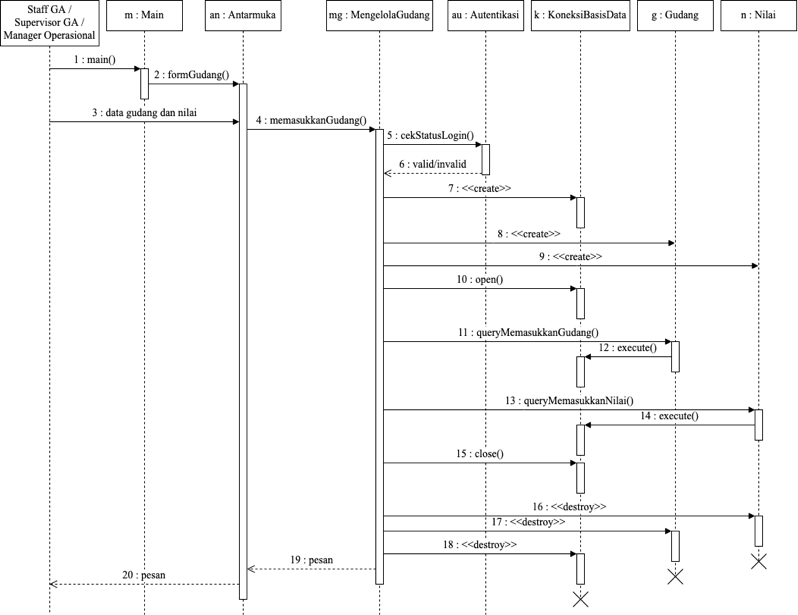
Gambar 4. 17 Sequence Diagram Hapus Data Kriteria

#### Sequence Diagram Lihat Data Kriteria



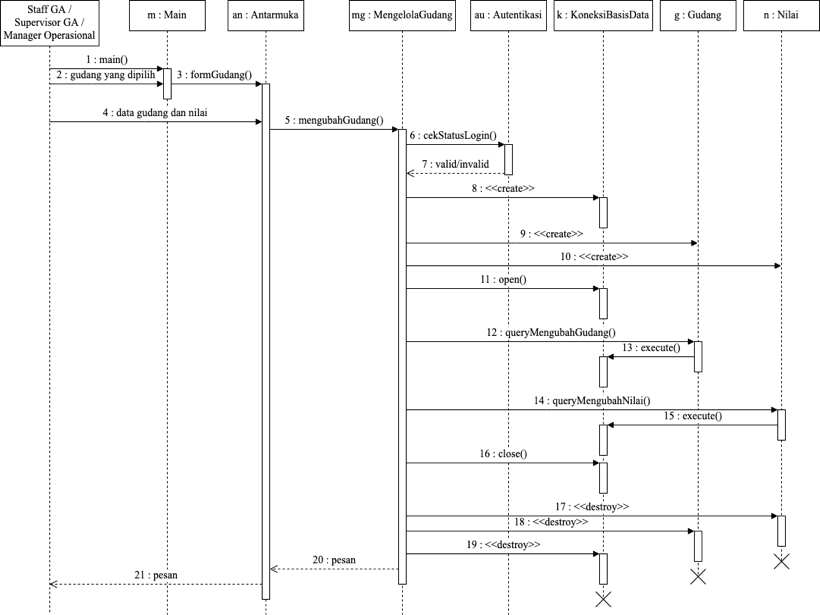
Gambar 4. 18 Sequence Diagram Lihat Data Kriteria

#### Sequence Diagram Tambah Data Gudang



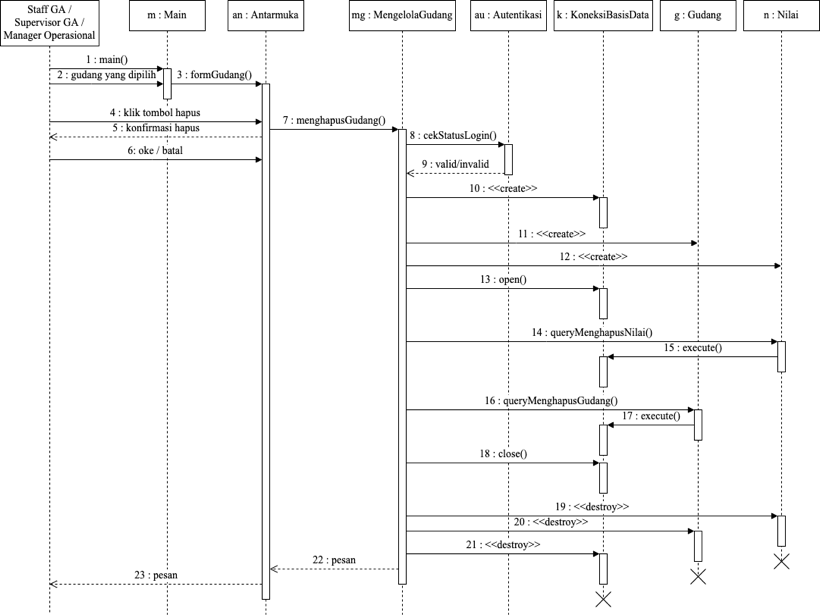
Gambar 4. 19 Sequence Diagram Tambah Data Gudang

#### Sequence Diagram Ubah Data Gudang



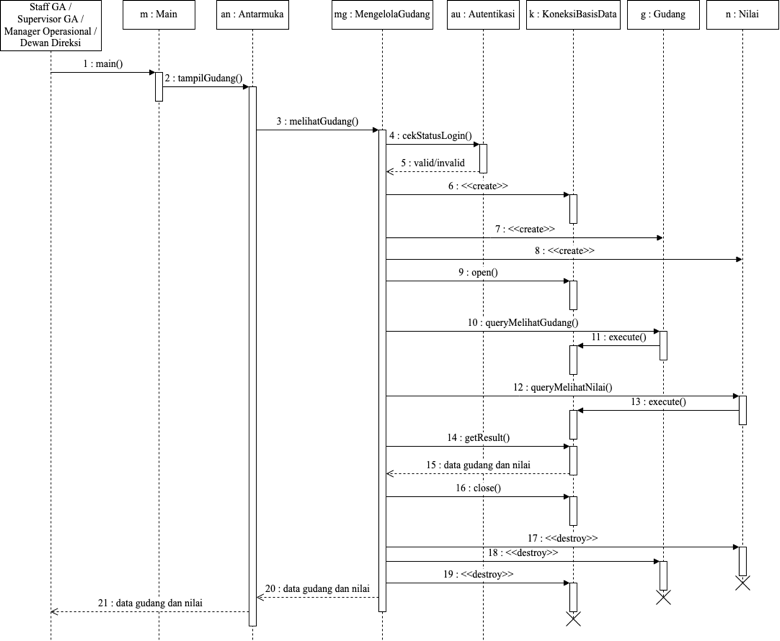
Gambar 4. 20 Sequence Diagram Ubah Data Gudang

#### Sequence Diagram Hapus Data Gudang



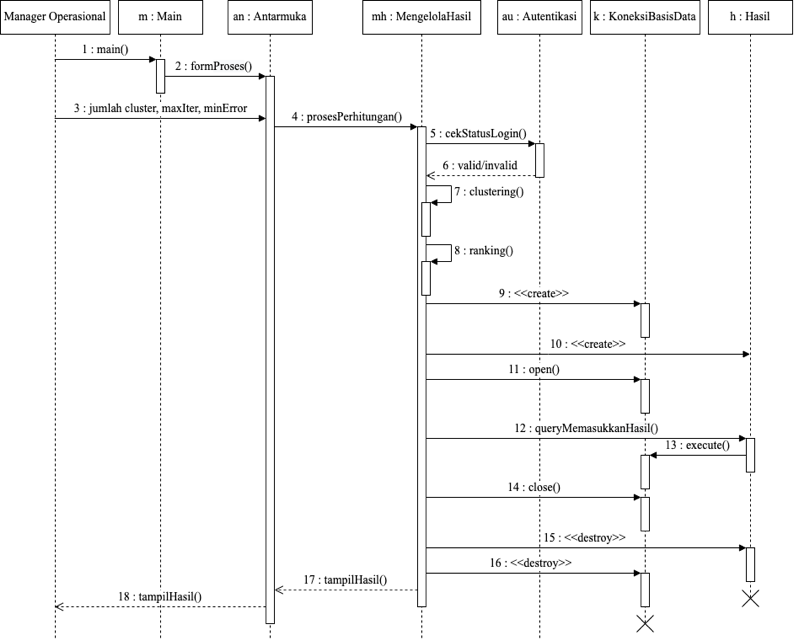
Gambar 4. 21 Sequence Diagram Hapus Data Gudang

#### Sequence Diagram Lihat Data Gudang



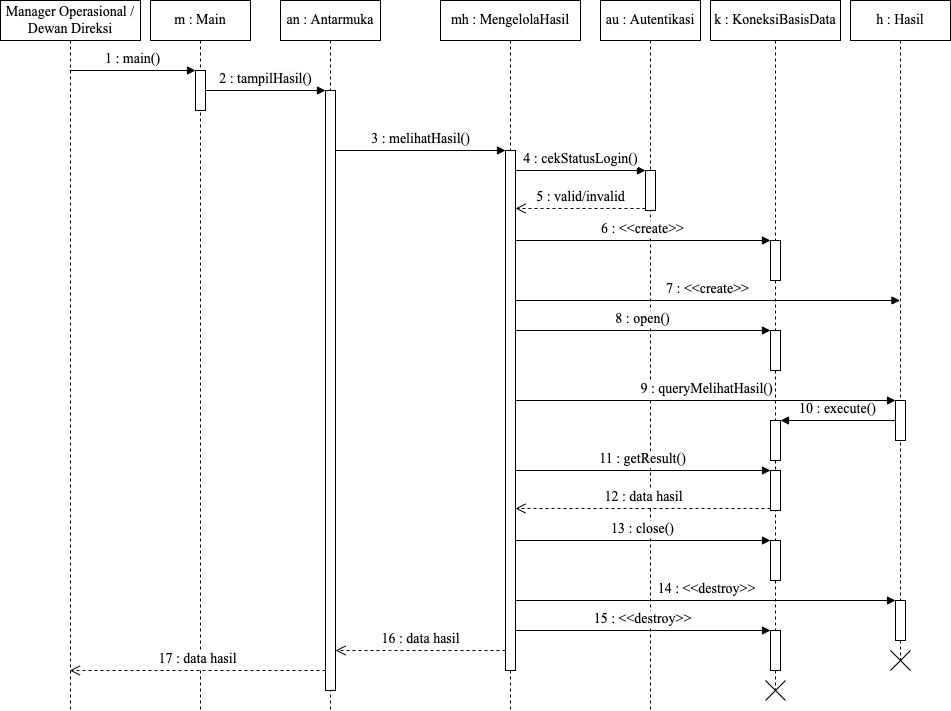
Gambar 4. 22 Sequence Diagram Lihat Data Gudang

#### Sequence Diagram Proses Perhitungan



Gambar 4. 23 Sequence Diagram Proses Perhitungan

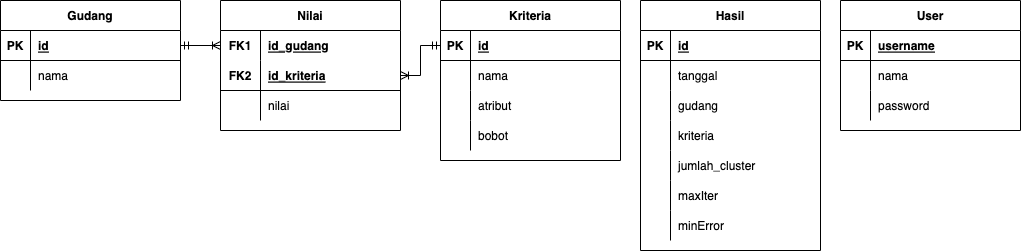
#### Sequence Diagram Lihat Hasil Perhitungan



Gambar 4. 24 Sequence Diagram Lihat Hasil Perhitungan

## Rancangan Basis Data

Berikut merupakan rancangan basis data dari aplikasi pemilihan gudang :

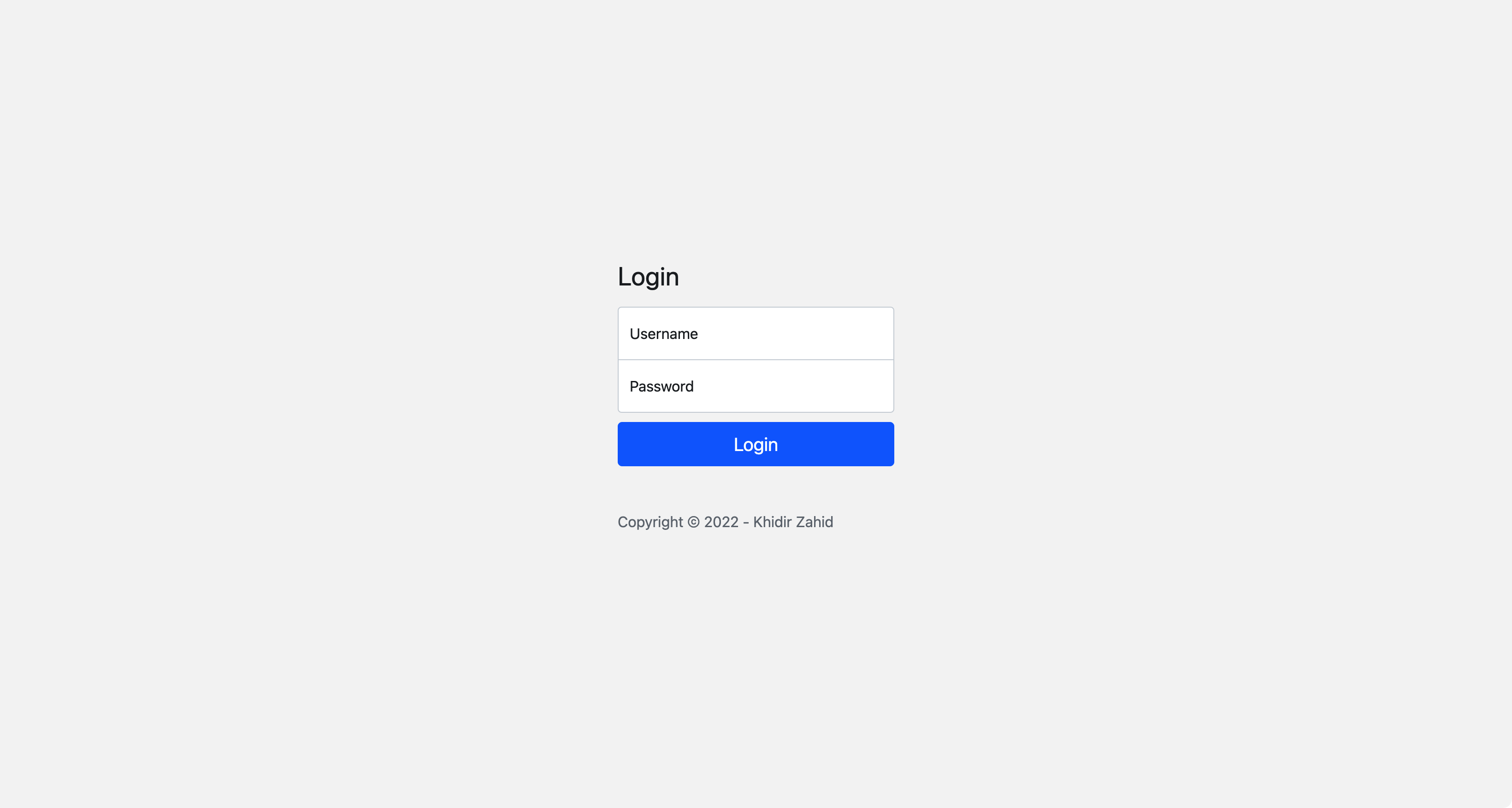


Gambar 4. 25 Rancangan Basis Data

## Hasil Rancangan User Interface

### User Interface Halaman Login

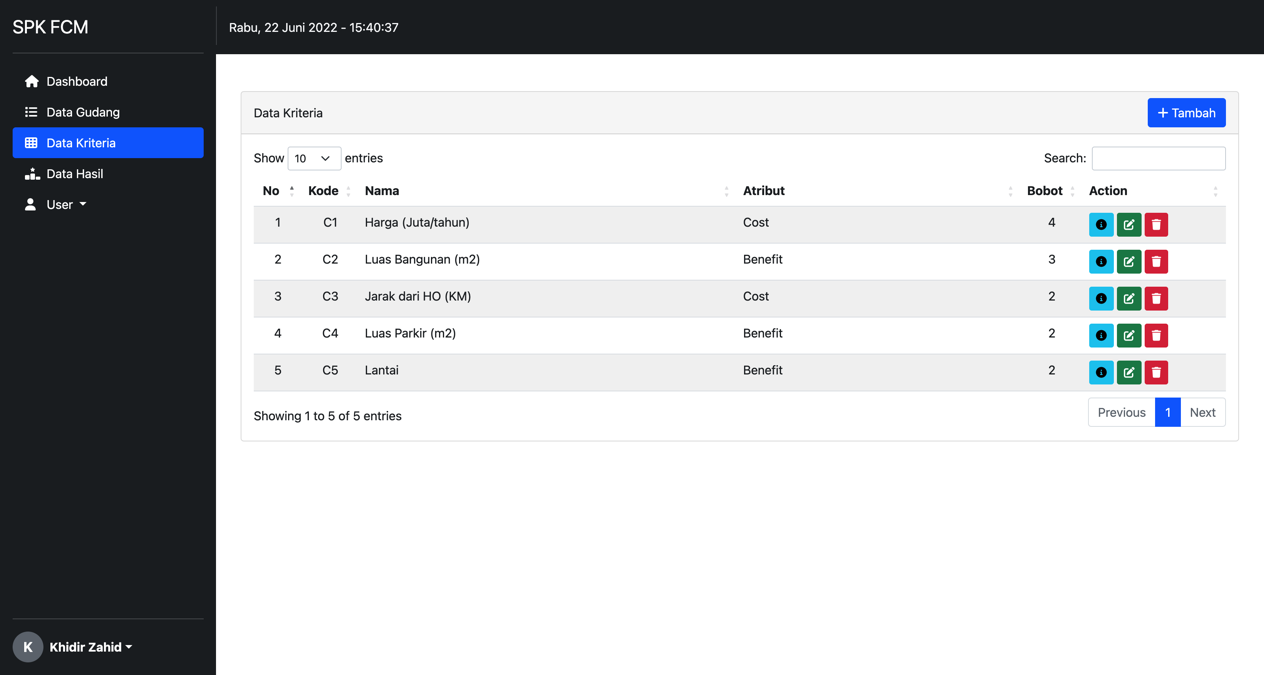
Pengguna diwajibkan *login* dengan mengisi *username* dan *password* yang ada pada halaman tersebut agar dapat mengakses aplikasi pemilihan gudang. Berikut merupakan tampilan antarmuka halaman login :



Gambar 4. 26 User Interface Halaman Login

### User Interface Halaman Data Kriteria

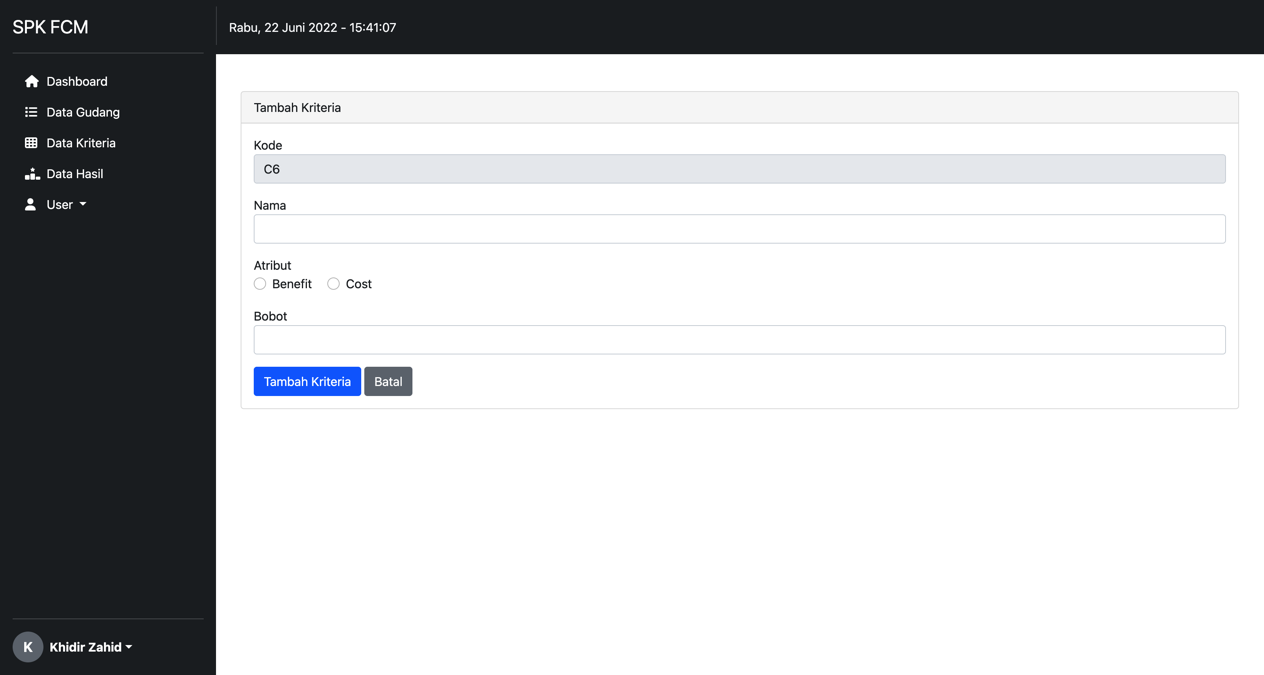
Halaman ini berfungsi untuk menampilkan seluruh data kriteria dan terdapat beberapa aksi yang dapat dilakukan dalam halaman ini yaitu tambah kriteria, ubah kriteria, dan hapus kriteria. Berikut merupakan tampilan antarmuka halaman data kriteria :



Gambar 4. 27 User Interface Halaman Data Kriteria

### User Interface Halaman Tambah Kriteria

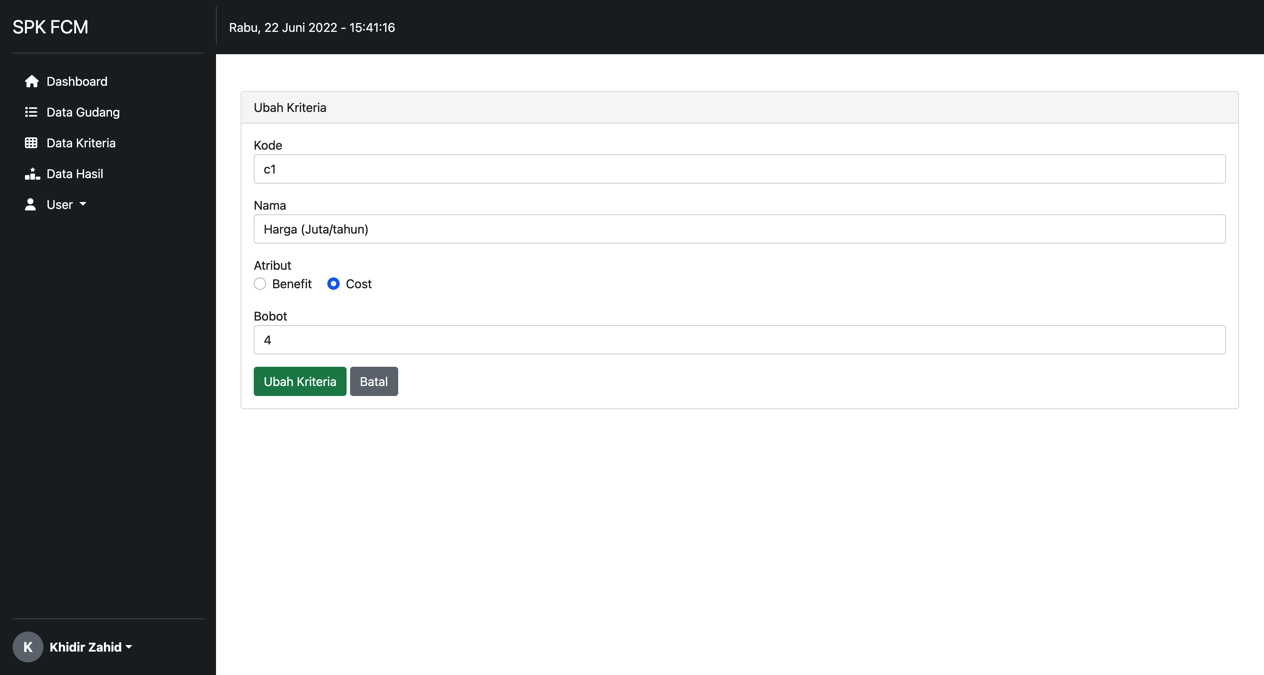
Halaman ini berfungsi untuk menambahkan data kriteria yang baru. Berikut merupakan tampilan antarmuka halaman tambah kriteria :



Gambar 4. 28 User Interface Halaman Tambah Kriteria

### User Interface Halaman Ubah Kriteria

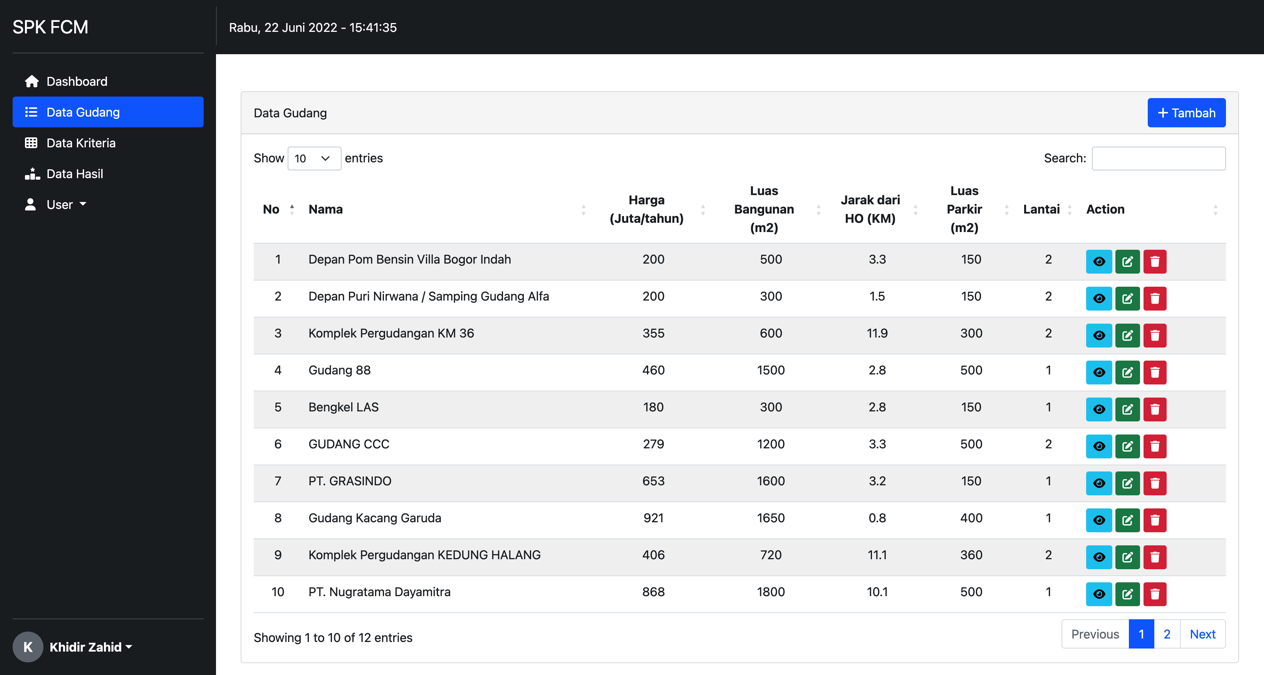
Halaman ini berfungsi untuk mengubah data kriteria jika admin salah dalam menginputkan data atau terdapat perubahan data. Berikut merupakan tampilan antarmuka halaman tambah kriteria :



Gambar 4. 29 User Interface Halaman Ubah Kriteria

### User Interface Halaman Data Gudang

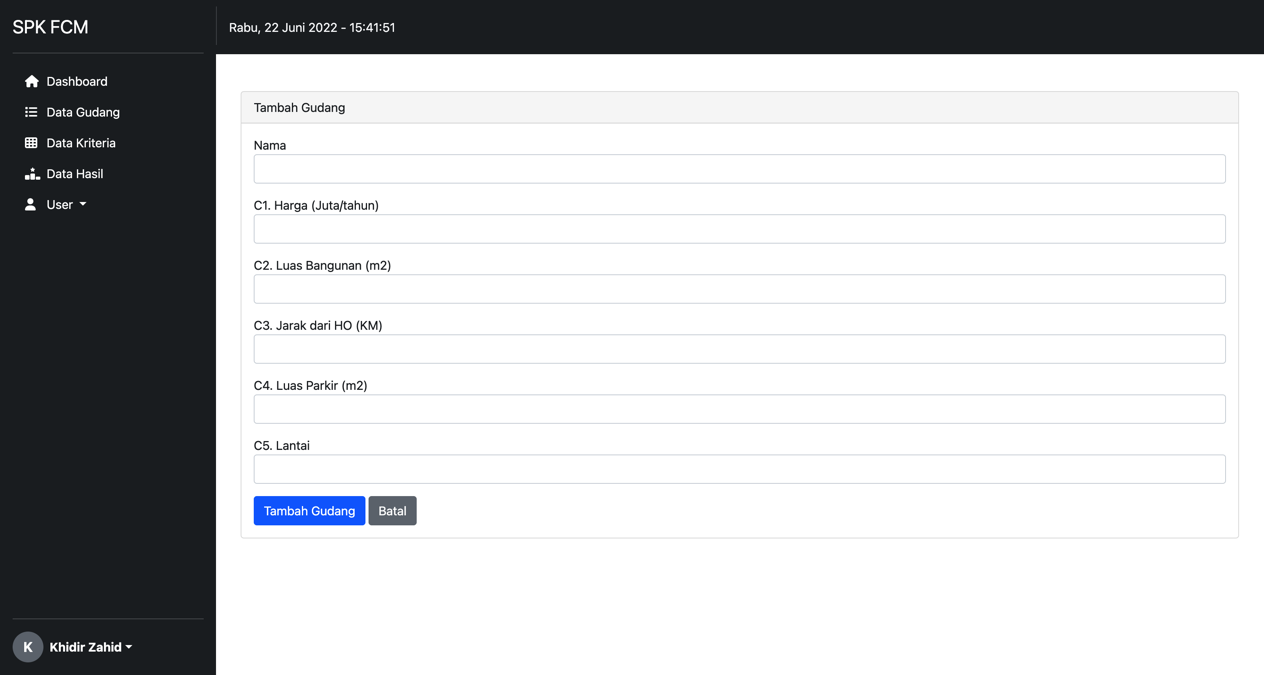
Halaman ini berfungsi untuk menampilkan seluruh data gudang dan terdapat beberapa aksi yang dapat dilakukan dalam halaman ini yaitu tambah gudang, ubah gudang, dan hapus gudang. Berikut merupakan tampilan antarmuka halaman data gudang :



Gambar 4. 30 User Interface Halaman Data Gudang

### User Interface Halaman Tambah Gudang

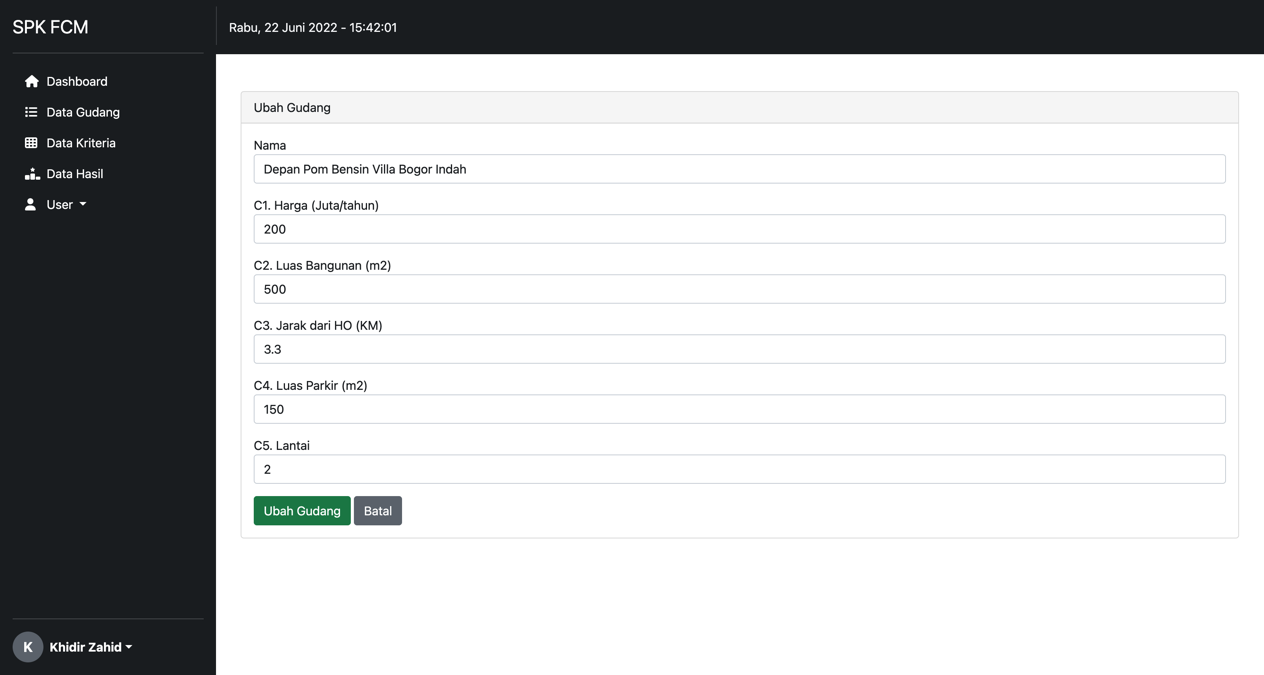
Halaman ini berfungsi untuk menambahkan data gudang yang baru. Berikut merupakan tampilan antarmuka halaman tambah gudang :



Gambar 4. 31 User Interface Halaman Tambah Gudang

### User Interface Halaman Ubah Gudang

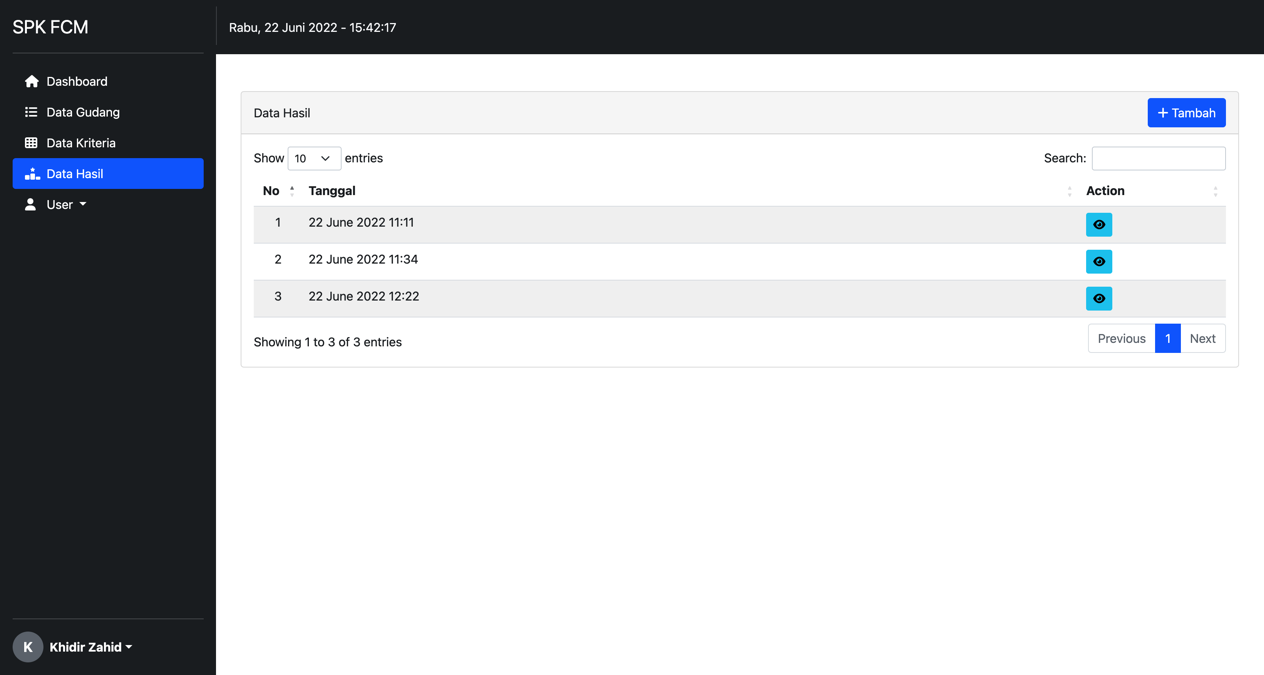
Halaman ini berfungsi untuk mengubah data gudang jika admin salah dalam menginputkan data atau terdapat perubahan data. Berikut merupakan tampilan antarmuka halaman tambah gudang :



Gambar 4. 32 User Interface Halaman Ubah Gudang

### User Interface Halaman Data Hasil Perhitungan

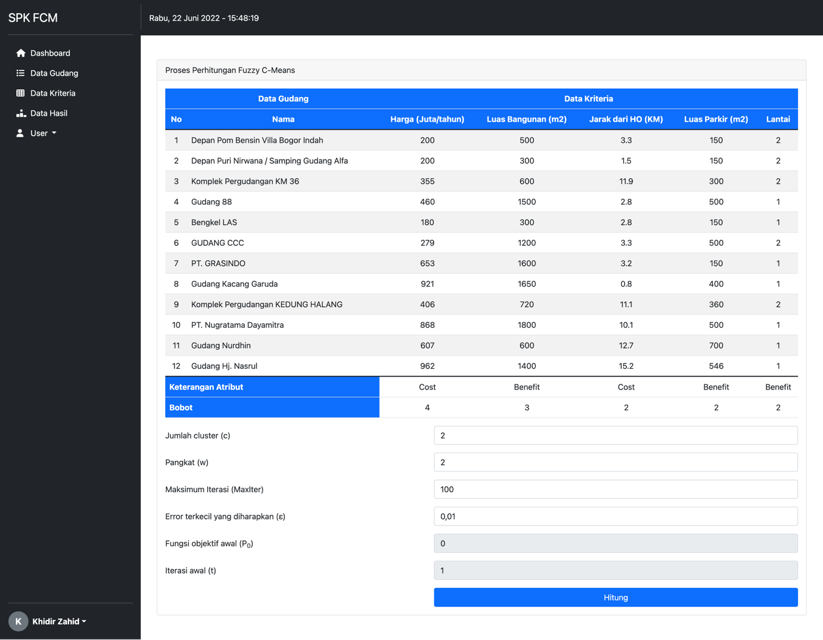
Halaman ini berfungsi untuk menampilkan seluruh data hasil perhitungan dan terdapat aksi untuk menambahkan proses perhitungan baru. Berikut merupakan tampilan antarmuka halaman data hasil perhitungan:



Gambar 4. 33 User Interface Halaman Data Hasil Perhitungan

### User Interface Halaman Proses Perhitungan

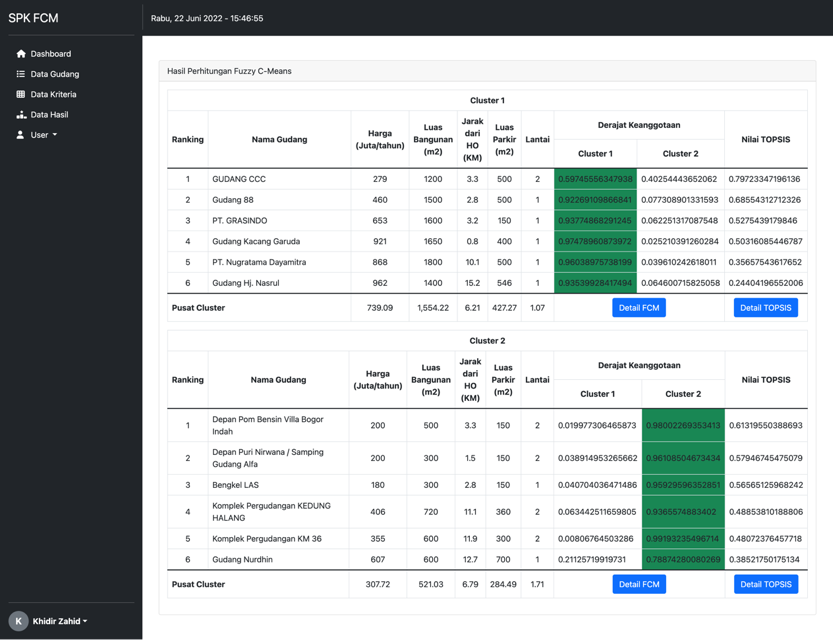
Halaman ini berfungsi untuk menambahkan proses perhitungan yang baru. Berikut merupakan tampilan antarmuka halaman proses perhitungan :



Gambar 4. 34 User Interface Halaman Proses Perhitungan

### User Interface Halaman Detail Hasil Perhitungan

Halaman ini berfungsi untuk menampilkan hasil dari proses perhitungan. Berikut merupakan tampilan antarmuka halaman detail hasil perhitungan :



Gambar 4. 35 User Interface Halaman Detail Hasil Perhitungan

## Hasil Pengujian

### Hasil Pengujian BlackBox Testing

Berikut merupakan tabel hasil pengujian aplikasi pemilihan gudang menggunakan metode *blackbox testing* :

Tabel 4. 9 Hasil Pengujian BlackBox Testing

| **No.** | **Fitur** | **Masukan** | **Keluaran** | **Hasil** |
| --- | --- | --- | --- | --- |
| 1 | *Login* | Data Benar | Tampil pesan "Anda berhasil *login*", *redirect* ke halaman *dashboard* | Sesuai |
| Data Salah | Tampil pesan " Username atau password salah" | Sesuai |
| 2 | *Logout* | - | Tampil pesan "Anda berhasil *logout*", redirect ke halaman *login* | Sesuai |
| 3 | Kelola Data Kriteria   1. Tambah Kriteria | Data Benar | Tampil pesan "Anda berhasil tambah kriteria", *redirect* ke halaman data kriteria | Sesuai |
| Data Salah | Tampil pesan "Anda gagal tambah kriteria" | Sesuai |
| 1. Ubah Kriteria | Data Benar | Tampil pesan "Anda berhasil ubah kriteria", *redirect* ke halaman data kriteria | Sesuai |
| Data Salah | Tampil pesan "Anda gagal ubah kriteria" | Sesuai |
| 1. Hapus Kriteria | - | Tampil pesan "Anda berhasil hapus kriteria", *redirect* ke halaman data kriteria | Sesuai |
| 4 | Kelola Data Gudang  Tambah Gudang | Data Benar | Tampil pesan "Anda berhasil tambah gudang", *redirect* ke halaman data gudang | Sesuai |
| Data Salah | Tampil pesan "Anda gagal tambah gudang" | Sesuai |
| Ubah Gudang | Data Benar | Tampil pesan "Anda berhasil ubah gudang", *redirect* ke halaman data gudang | Sesuai |
| Data Salah | Tampil pesan "Anda gagal ubah gudang" | Sesuai |
| Hapus Gudang | - | Tampil pesan "Anda berhasil hapus gudang", *redirect* ke halaman data gudang | Sesuai |
| 5 | Kelola Data Hasil  Proses Perhitungan | Data Benar | *redirect* ke halaman hasil perhitungan | Sesuai |
| Data Salah | Tampil pesan "Anda gagal proses" | Sesuai |

# DAFTAR PUSTAKA

[1] W. Siahaya, *Sukses Supply Chain Management Akses Demand Chain Management*. Jakarta: In Media, 2013.

[2] Zaroni, *Logistics and Supply Chain Konsep Dasar - Logistik Kontemporer - Praktik Terbaik*. Jakarta: Prasetya Mulya Publishing, 2017.

[3] S. Kusumadewi dan H. Purnomo, *Aplikasi Logika Fuzzy Untuk Pendukung Keputusan*. Yogyakarta: Graha Ilmu, 2004.

[4] S. Kusumadewi, S. Hartati, A. Harjoko, dan R. Wardoyo, *Fuzzy Multi-Attribute Decision Making (Fuzzy MADM)*. Yogyakarta: Graha Ilmu, 2006.

[5] C. Bai, D. Dhavale, dan J. Sarkis, “Integrating Fuzzy C-Means and TOPSIS for performance evaluation: An application and comparative analysis,” *Expert Systems with Applications*, vol. 41, no. 9, hlm. 4186–4196, Jul 2014, doi: 10.1016/j.eswa.2013.12.037.

[6] G. M. Susanto, S. Kosasi, David, Gat, dan S. M. Kuway, “Sistem Referensi Pemilihan Smartphone Android Dengan Metode Fuzzy C-Means dan TOPSIS,” *RESTI*, vol. 4, no. 6, Des 2020, doi: 10.29207/resti.v4i6.2584.

[7] E. F. Himah dan R. Sulaiman, “Implementasi Metode Fuzzy C-Means dan TOPSIS dalam Evaluasi Kinerja Keuangan Perusahaan Perbankan di Indonesia Berdasarkan Rasio Keuangan,” *MU*, vol. 9, no. 1, hlm. 43–53, Jan 2021, doi: 10.26740/mathunesa.v9n1.p43-53.

[8] D. Oktariani, D. Andreswari, dan Y. Setiawan, “Sistem Pendukung Keputusan dalam Pemetaan Prioritas Perbaikan Jalan dan Jembatan Nasional di Provinsi Bengkulu Menggunakan Metode TOPSIS dan Fuzzy C-Means,” *Rekursif: Jurnal Informatika*, vol. 5, no. 2, Art. no. 2, Jul 2017, doi: 10.33369/rekursif.v5i2.1321.

[9] Christian S. K. Aditya, “Pemilihan Kalimat Representatif dengan Pengintegrasian Fuzzy C-Means Clustering dan TOPSIS (FCM-TOPSIS) untuk Peringkasan Dokumen,” *Prosiding SENTRA (Seminar Teknologi dan Rekayasa)*, no. 6, Art. no. 6, Jan 2021, doi: 10.22219/sentra.v0i6.3924.

[10] E. Turban, J. E. Aronson, dan T.-P. Liang, *Decision Support Systems and Intelligent Systems*, Edisi Ketujuh. New Delhi: Prentice-Hall of India, 2005.

[11] Rosa A. S. dan M. Shalahuddin, *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*, Edisi Kedua. Bandung: Informatika, 2014.

[12] R. S. Pressman, *Software engineering: a practitioner’s approach*, Edisi Ketujuh. New York: McGraw-Hill Higher Education, 2010.

[13] E. V. Sandin, N. M. Yassin, dan R. Mohamad, “Comparative Evaluation of Automated Unit Testing Tool for PHP,” *International Journal of Software Engineering and Technology*, vol. 2, no. 2, hlm. 7–11, Des 2016.