MAKALAH

APLIKASI PENGIMPUTAN PEMBELIAN GAS LPG 3 KG

Disusun Guna Memenuhi Tugas UPK Mata Kuliah Pemrograman Berorientasi Objek

Dosen Pengampu : Mardiyyah Hasnawi, S.Kom., M.T.,MTA

DISUSUN OLEH:

Nama: Aisyah Safira Rachman

Nim: 13020210004

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS ILMU KOMPUTER

UNIVERSITAS MUSLIM INDONESIA

2023

DAFTAR ISI

Abstrak v

BAB I PENDAHULUAN 1

1.1 Latar Belakang 1
1.2 Rumusan Masalah 1
1.3 Tujuan 1
1.4 Manfaat 1

BAB II TINJAUAN PUSTAKA 3

2.1 Konsep Dasar OOP 3

2.1.1 Pengertian OOP 3

2.1.2 Prinsip-Prinsip OOP 3

2.2 Pentingnya OOP dalam Pengembangan Perangkat Lunak 3

2.2.1 Struktur Terstruktur dan Modular 3

2.2.2 Reusabilitas dan Keterbacaan Kode 3

2.2.3 Mempermudah Pemeliharaan dan Perubahan 4
2.2.4 Peningkatan Produktivitas Pengembangan 4

2.3 Konsep Dasar GUI 4

2.3.1 Pengertian GUI 4
2.3.2 Komponen-Komponen GUI 4
2.3.3 Interaksi Pengguna dan Respons GUI 5

2.4 Kebutuhan dan Pentingnya GUI dalam Pengembangan Perangkat Lunak 5

2.4.1 Meningkatkan User Experience 5

2.4.2 Mempermudah Navigasi dan Interaksi Pengguna 5

2.5 Konsep Dasar Arsitektur MVC.....5

2.5.1 Pengertian Arsitektur MVC.....5

2.5.2 Komponen-Komponen Arsitektur MVC (Model, View, Controller).....5

2.5.3 Cara Kerja Arsitektur MVC dalam Pengembangan Perangkat Lunak dengan GUI.....5

BAB III ANALISIS DAN DESAIN.....8

3.1 Identifikasi Pengguna dan Analisis Kebutuhan.....8

3.1.1 Identifikasi Pengguna Sistem.....8

3.1.1.1 Admin Gas LPG 3 KG.....8

3.1.2 Kebutuhan Fungsional.....8

3.1.2.1 Pengelolaan Data Pelanggan.....8

3.1.3 Kebutuhan Non-Fungsional.....8

3.1.3.1 Antarmuka Pengguna yang Intuitif dan User-Friendly.....8

3.1.3.2 Keamanan Data dan Akses Pengguna yang Terkontrol.....8

3.2 Metode Pengembangan.....8

3.2.1 Metode Waterfall.....8

3.3 Flowchart.....9

BAB IV IMPLEMENTASI DAN PENGUJIAN.....10

4.1 Output Aplikasi Data Pembelian Gas LPG 3KG.....10

4.2 Database Aplikasi Data Pembelian Gas LPG 3KG.....10

4.3 Pengujian Program.....11

4.3.1 Create.....11

4.3.2 Pengujian Update 11

4.3.3 Pengujian Delete 12

4.3.4 Mencari Data 13

BAB V PENUTUP 14

5.1 Kesimpulan 14

DAFTAR PUSTAKA 15

ABSTRAK

APLIKASI PENGIMPUTAN PEMBELIAN GAS LPG 3 KG

Aisyah Safira Rachman

NIM: 13020210004

Makalah ini mengimplementasikan sebuah aplikasi java GUI berbasis MVC yang bertujuan untuk memudahkan admin untuk memgelolah data pembelian gas LPG 3 kg. Aplikasi ini menyediakan fungsi operasi CRUD (Create, Retrieve, Update, Delete) untuk membuat, mengambil, memperbaharui, dan menghapus data pembeli, dan riwayat transaksi dengan mudah.

Aplikasi ini dirancang dengan tujuan untuk mengatasi beberapa kendala yang sering dihadapi oleh admin dalam memperoleh data pembeli gas LPG 3 kg, seperti mencari manual daftar nama pelanggan, alamat pelanggan, dan jumlah pembelian.

Metode pengembangan yang digunakan meliputi analisis kebutuhan pengguna, perancangan antarmuka pengguna yang menarik dan responsif, pengembangan aplikasi dengan teknologi GUI, pengujian fungsionalitas dan keamanan aplikasi, serta implementasi fitur CRUD untuk manajemen data.

Hasil pengujian aplikasi menunjukkan bahwa aplikasi ini mampu mempercepat proses pengelolaan data pembeli yang memungkinkan penjual untuk dengan mudah melihat dan menganalisis data pembeli.

Makalah ini diharapkan dapat memberikan kontribusi dalam meningkatkan efisiensi dalam proses mengelolah data pembelian gas LPG 3 kg, dan menyediakan sistem pengelolaan data yang terintegrasi untuk pemantauan dan analisis data pembeli. sekaligus memperkuat keamanan dan kerahasiaan data pembeli.

BAB I PENDAHULUAN

1.1 Latar Belakang

Dalam era perkembangan teknologi informasi yang terus maju, pengembangan perangkat lunak dengan antarmuka grafis pengguna (Graphical User Interface/GUI) dan penerapan pemrograman berorientasi objek (Object-Oriented Programming/OOP) menjadi semakin penting. GUI memungkinkan pengguna berinteraksi dengan perangkat lunak melalui elemen-elemen visual, sementara OOP memungkinkan pengembangan perangkat lunak yang terstruktur dan modular. Selain itu, untuk mengelola kompleksitas perangkat lunak dengan GUI, penggunaan arsitektur Model-View-Controller (MVC) dapat menjadi pendekatan yang efektif.

1.2 Rumusan Masalah

a. Apa itu OOP dan mengapa OOP penting dalam pengembangan perangkat lunak?

b. Apa itu GUI dan mengapa GUI diperlukan dalam pengembangan perangkat lunak?

c. Apa itu arsitektur MVC dan bagaimana cara kerjanya dalam pengembangan perangkat lunak dengan GUI?

1.3 Tujuan

a. Memperkenalkan konsep dasar OOP dan GUI kepada pembaca.

b. Menggambarkan konsep arsitektur MVC dan cara kerjanya dalam pengembangan perangkat lunak dengan GUI.

c. Menjelaskan kebutuhan dan pentingnya OOP, GUI, dan arsitektur MVC dalam pengembangan perangkat lunak yang kompleks.

1.4 Manfaat

a. Memahami OOP memungkinkan pembaca untuk merancang perangkat lunak yang terstruktur, modular, dan mudah diorganisasi.

b. Memahami GUI memungkinkan pembaca untuk merancang antarmuka pengguna yang intuitif dan responsif.
c. Memahami arsitektur MVC memungkinkan pembaca untuk memisahkan logika bisnis, tampilan, dan pengendalian dalam pengembangan perangkat lunak dengan GUI.

BAB II TINJAUAN PUSTAKA

2.1 Konsep Dasar OOP

2.1.1 Pengertian OOP

Pemrograman Berorientasi Objek (Object-Oriented Programming/OOP) adalah paradigma pemrograman yang berfokus pada konsep objek. Objek adalah entitas yang memiliki atribut (data) dan perilaku (metode/fungsi) yang terkait. OOP memungkinkan pengembangan perangkat lunak dengan pendekatan yang lebih terstruktur, modular, dan lebih mudah dipahami.

2.1.2 Prinsip-Prinsip OOP

Ada 4 prinsip utama dari OOP:

1. Abstraksi: Memodelkan objek dunia nyata menjadi abstraksi yang dapat diimplementasikan dalam perangkat lunak.
2. Pewarisan: Mewarisi sifat dan perilaku dari objek yang sudah ada untuk membuat objek baru yang lebih spesifik.
3. Enkapsulasi: Menggabungkan data dan metode terkait dalam sebuah objek, dan menyembunyikan akses langsung ke data internal objek tersebut.
4. Polimorfisme: Menggunakan konsep yang sama dengan cara yang berbeda, sehingga objek dapat memiliki banyak bentuk (tipe).

2.2 Pentingnya OOP dalam Pengembangan Perangkat Lunak

2.2.1 Struktur Terstruktur dan Modular

OOP memungkinkan struktur yang terstruktur dan modular dalam pengembangan perangkat lunak. Dengan membagi perangkat lunak menjadi kelas-kelas yang terpisah, kita dapat mengatur kode dengan lebih baik, membaginya menjadi bagian-bagian yang lebih kecil, dan mempermudah pemeliharaan.

2.2.2 Reusabilitas dan Keterbacaan Kode

OOP memungkinkan reusabilitas kode yang tinggi. Kita dapat menggunakan kembali kelas-kelas yang sudah ada untuk membangun perangkat lunak baru, menghemat waktu dan usaha dalam pengembangan. Selain itu, OOP juga mendorong penulisan kode yang lebih keterbaca dan mudah dipahami oleh orang lain, sehingga memudahkan kolaborasi dalam tim pengembangan.

2.2.3 Mempermudah Pemeliharaan dan Perubahan

Dalam OOP, pemeliharaan perangkat lunak menjadi lebih mudah karena kode terorganisir dalam kelas-kelas yang terpisah. Jika terjadi perubahan pada suatu bagian, kita dapat fokus pada kelas terkait tanpa mempengaruhi bagian lain. Hal ini mengurangi dampak perubahan dan meminimalkan risiko kesalahan.

2.2.4 Peningkatan Produktivitas Pengembangan

Dengan menggunakan OOP, pengembang dapat bekerja lebih efisien dan produktif. OOP memungkinkan pemrograman yang lebih terstruktur, modular, dan keterbacaan kode yang baik. Ini memudahkan pengembang dalam merancang, mengimplementasikan, dan menguji komponen-komponen perangkat lunak. Selain itu, reusabilitas kode juga mengurangi upaya pengembangan yang berulang-ulang.

2.3 Konsep Dasar GUI

2.3.1 Pengertian GUI

Graphical User Interface (GUI) adalah antarmuka pengguna berbasis grafis yang memungkinkan pengguna berinteraksi dengan perangkat lunak melalui elemen-elemen visual seperti tombol, menu, dan jendela. GUI memanfaatkan grafika dan ikon untuk memvisualisasikan informasi dan memudahkan interaksi pengguna.

2.3.2 Komponen-Komponen GUI

Komponen-komponen GUI mencakup elemen-elemen seperti tombol, teks, gambar, jendela, dan menu. Komponen-komponen ini digunakan untuk membangun antarmuka pengguna yang intuitif dan responsif.

2.3.3 Interaksi Pengguna dan Respons GUI

GUI memungkinkan pengguna untuk berinteraksi dengan perangkat lunak melalui tindakan seperti mengklik tombol, memasukkan teks, atau memilih opsi dari menu. Selanjutnya, GUI memberikan respons dengan menampilkan informasi, memperbarui tampilan, atau menjalankan perintah sesuai dengan aksi pengguna.

2.4 Kebutuhan dan Pentingnya GUI dalam Pengembangan Perangkat Lunak

2.4.1 Meningkatkan User Experience

GUI meningkatkan pengalaman pengguna dengan menyediakan antarmuka yang intuitif dan mudah digunakan. Pengguna dapat berinteraksi dengan perangkat lunak secara visual, mempercepat pemahaman, dan mengurangi kesalahan dalam penggunaan.

2.4.2 Mempermudah Navigasi dan Interaksi Pengguna

GUI memungkinkan navigasi yang mudah melalui elemen-elemen visual seperti tombol, menu, dan tautan. Pengguna dapat dengan cepat berpindah antar elemen dan berinteraksi dengan perangkat lunak secara efisien. Hal ini mempercepat akses ke fitur-fitur yang diinginkan dan meningkatkan efektivitas penggunaan.

2.4.3 Memvisualisasikan Informasi dengan Jelas

GUI memanfaatkan elemen visual seperti grafik, ikon, dan tata letak yang dapat membantu pengguna memahami informasi dengan lebih baik. Data dan hasil perhitungan dapat ditampilkan secara visual melalui diagram, grafik, atau tabel, mempermudah interpretasi dan pengambilan keputusan.

2.4.4 Meningkatkan Produktivitas dan Efisiensi

Dengan antarmuka yang intuitif, GUI memungkinkan pengguna untuk bekerja lebih efisien dan produktif. Pengguna dapat dengan mudah menavigasi, mengakses fitur, dan melakukan tugas-tugas secara visual, mengurangi waktu dan usaha yang dibutuhkan dalam penggunaan perangkat lunak.

2.5 Konsep Dasar Arsitektur MVC

2.5.1 Pengertian Arsitektur MVC

Model-View-Controller (MVC) adalah sebuah pola arsitektur perangkat lunak yang memisahkan logika bisnis (Model), tampilan antarmuka pengguna (View), dan pengendalian (Controller) ke dalam komponen-komponen terpisah. MVC memungkinkan pemisahan tanggung jawab dan meningkatkan pemeliharaan dan pengembangan perangkat lunak.

2.5.2 Komponen-Komponen Arsitektur MVC (Model, View, Controller)

- Model: Mewakili logika bisnis dan data dalam perangkat lunak. Model mengelola pengolahan data, perhitungan, validasi, dan interaksi dengan penyimpanan data.
- View: Menampilkan antarmuka pengguna kepada pengguna. View bertanggung jawab untuk mempresentasikan data dari Model dengan cara yang sesuai, seperti tampilan grafis atau tampilan teks.
- Controller: Mengendalikan interaksi antara pengguna, View, dan Model. Controller menangkap aksi pengguna, mengupdate Model, dan memperbarui tampilan View sesuai dengan perubahan data.

2.5.3 Cara Kerja Arsitektur MVC dalam Pengembangan Perangkat Lunak dengan GUI

Dalam pengembangan perangkat lunak dengan GUI, arsitektur MVC bekerja sebagai berikut:

- Pengguna berinteraksi dengan View melalui antarmuka pengguna.
- View menangkap aksi pengguna dan meneruskannya ke Controller.
- Controller menginterpretasikan aksi pengguna, memperbarui Model sesuai, dan memperbarui View dengan data yang baru.
- View menampilkan data dari Model yang telah diperbarui kepada pengguna.

- Proses ini terus berulang saat pengguna melakukan interaksi, sehingga Model, View, dan Controller saling berkomunikasi dan bekerja bersama untuk mengelola perangkat lunak.

BAB III ANALISIS DAN DESAIN

3.1 Identifikasi Pengguna dan Analisis Kebutuhan

3.1.1 Identifikasi Pengguna Sistem

3.1.1.1 Admin Gas LPG 3 KG

- Memiliki akses untuk melihat data pembelian, nama pelanggan, dan alamat pelanggan gas LPG 3 KG.
- Memiliki akses penuh terhadap data dan pengaturan aplikasi.

3.1.2 Kebutuhan Fungsional

3.1.2.1 Pengelolaan Data Pelanggan

- Sistem dapat menyimpan dan mengelola informasi pelanggan seperti nama, alamat, dan nomor telepon.
- Administrator sistem dapat mengelola data petugas penjualan, termasuk pengaturan dan perubahan informasi mereka.

3.1.3 Kebutuhan Non-Fungsional

3.1.3.1 Antarmuka Pengguna yang Intuitif dan User-Friendly

- Antarmuka aplikasi harus dirancang dengan baik agar mudah digunakan oleh pengguna.
- Navigasi yang jelas dan tampilan yang menarik harus dipertimbangkan.

3.1.3.2 Keamanan Data dan Akses Pengguna yang Terkontrol

- Data pelanggan dan transaksi harus diamankan dan hanya dapat diakses oleh pihak yang berwenang.

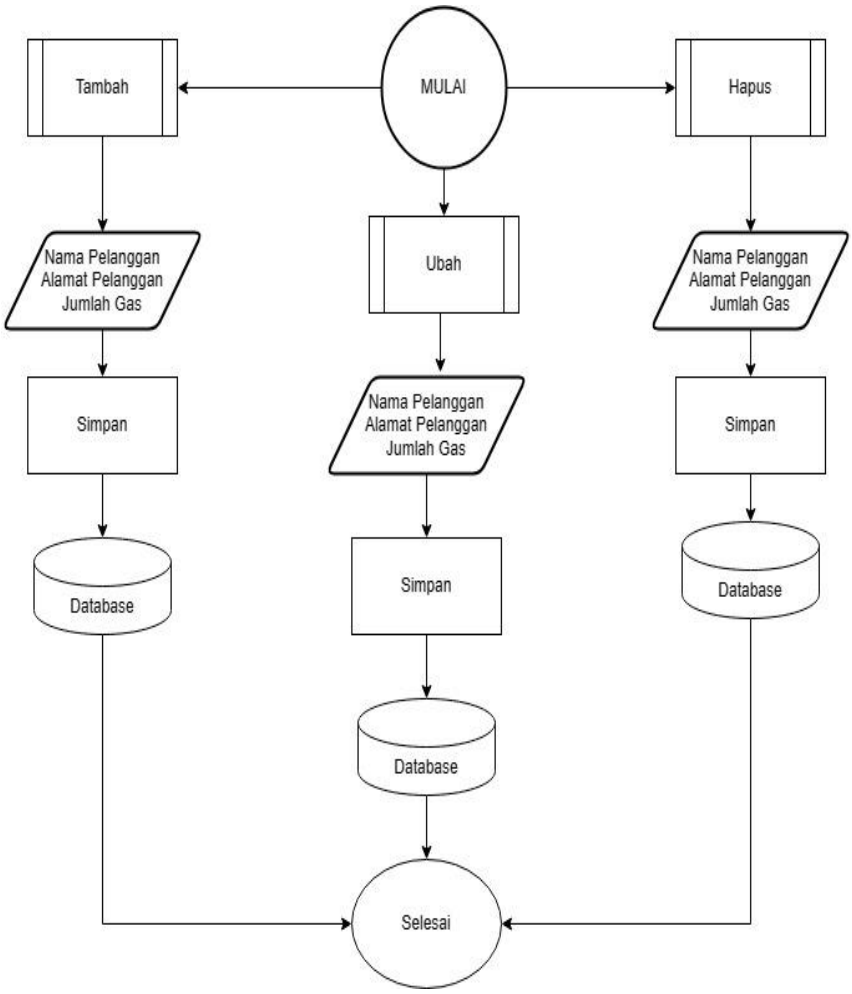
3.2 Metode Pengembangan

3.2.1 Metode Waterfall

Metode waterfall dipilih karena kebutuhan aplikasi telah ditentukan dengan jelas sejak awal dan tidak diharapkan terjadi perubahan yang signifikan selama proses pengembangan. Salah satu alasan penggunaan

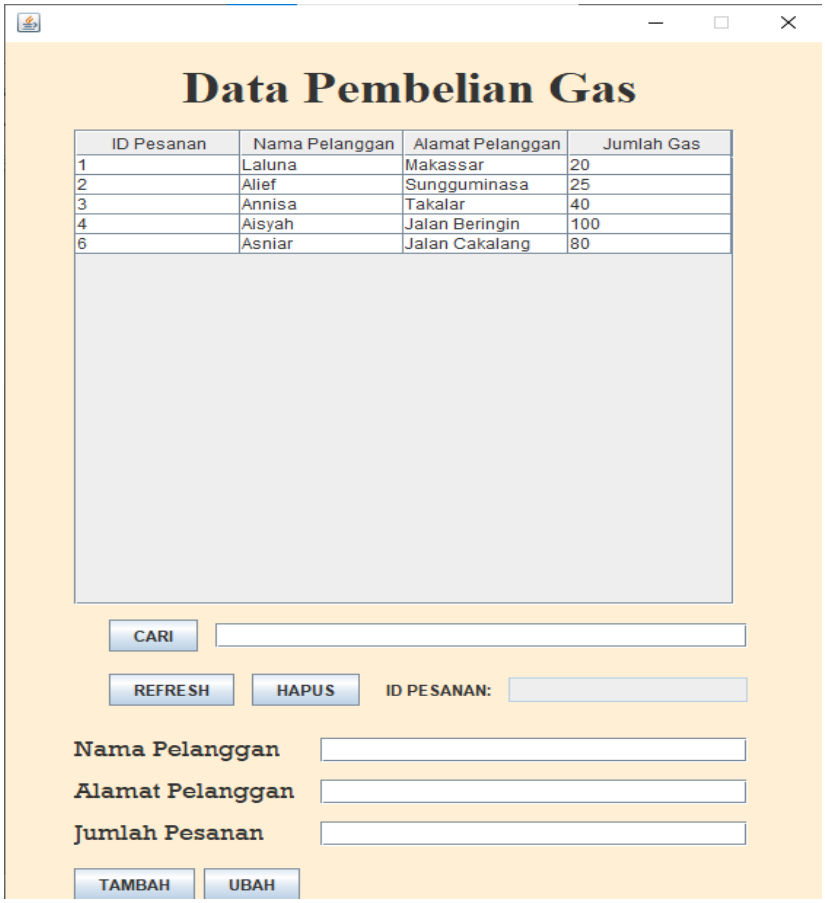
metode waterfall adalah kebutuhan yang telah ditetapkan dengan jelas. Sebelum memulai pengembangan aplikasi, kebutuhan-kebutuhan dasar, seperti ID pesanan, nama pelanggan, alamat pelanggan, dan jumlah gas, telah diidentifikasi dengan jelas. Oleh karena itu, langkah-langkah yang harus diambil dalam pengembangan aplikasi dapat ditentukan dengan baik.

3.3 Flowchart

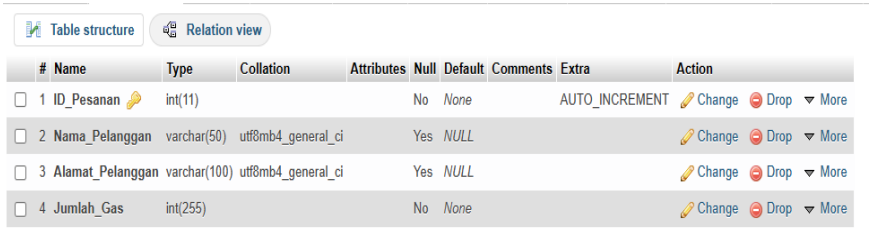


BAB IV IMPLEMENTASI DAN PENGUJIAN

4.1 Output Aplikasi Data Pembelian Gas LPG 3KG



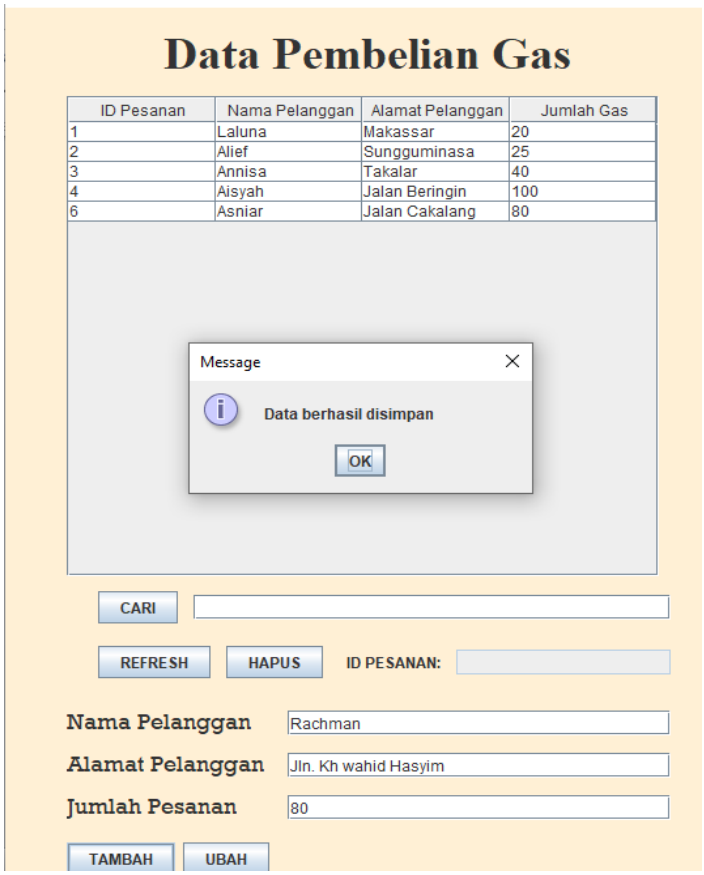
4.2 Database Aplikasi Data Pembelian Gas LPG 3KG



#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	ID_Pesanan	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	Nama_Pelanggan	varchar(50)	utf8mb4_general_ci		Yes	NULL			Change Drop More
3	Alamat_Pelanggan	varchar(100)	utf8mb4_general_ci		Yes	NULL			Change Drop More
4	Jumlah_Gas	int(255)			No	None			Change Drop More

4.3 Pengujian Program

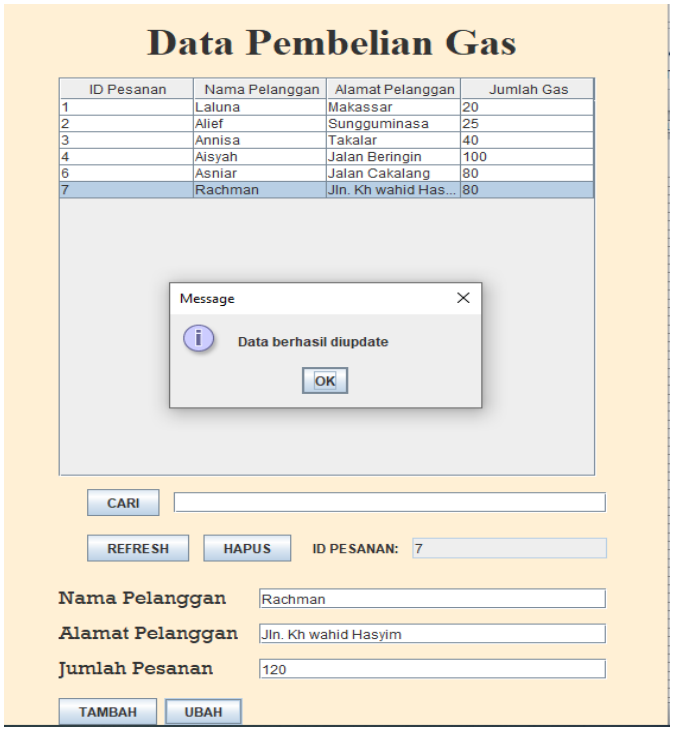
4.3.1 Create



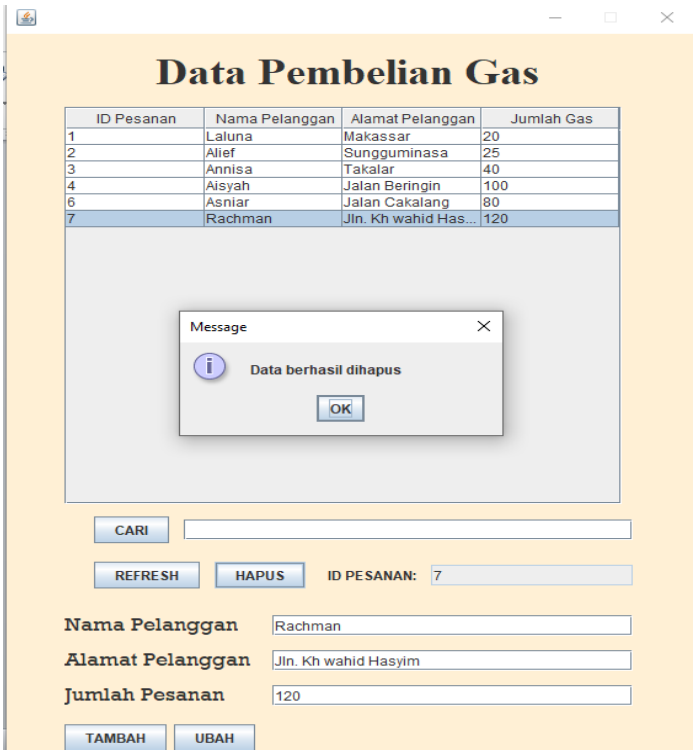
ID Pesanan	Nama Pelanggan	Alamat Pelanggan	Jumlah Gas
1	Laluna	Makassar	20
2	Alief	Sungguminasa	25
3	Annisa	Takalar	40
4	Aisyah	Jalan Beringin	100
6	Asniar	Jalan Cakalang	80
7	Rachman	Jln. Kh wahid Has...	80

4.3.2 Pengujian Update

ID Pesanan	Nama Pelanggan	Alamat Pelanggan	Jumlah Gas
1	Laluna	Makassar	20
2	Alief	Sungguminasa	25
3	Annisa	Takalar	40
4	Aisyah	Jalan Beringin	100
6	Asniar	Jalan Cakalang	80
7	Rachman	Jln. Kh wahid Has...	120

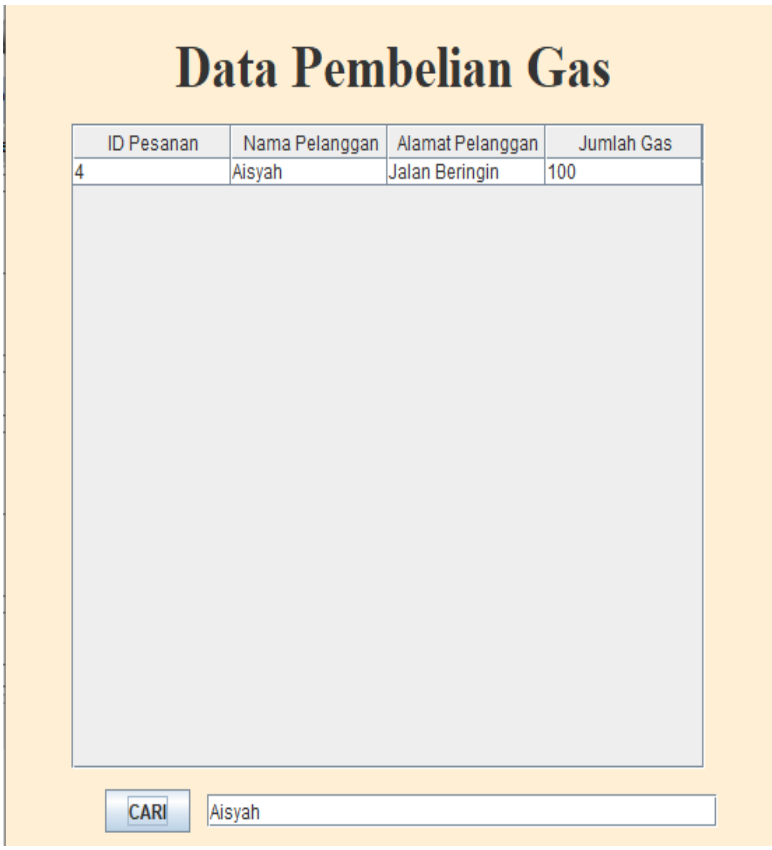


4.3.3 Pengujian Delete



ID Pesanan	Nama Pelanggan	Alamat Pelanggan	Jumlah Gas
1	Laluna	Makassar	20
2	Alief	Sungguminasa	25
3	Annisa	Takalar	40
4	Aisyah	Jalan Beringin	100
6	Asniar	Jalan Cakalang	80

4.3.4 Mencari Data



BAB V PENUTUP

5.1 Kesimpulan

Dalam dunia yang semakin digital ini, pengelolaan pembelian gas LPG 3 KG menjadi semakin penting. Dalam konteks ini, aplikasi dengan operasi CRUD berbasis MVC adalah solusi yang efektif. Melalui pemahaman konsep OOP dan desain antarmuka yang baik, aplikasi ini memberikan manfaat yang signifikan dalam pengelolaan data dengan mudah. Aplikasi ini menggunakan operasi CRUD yang memudahkan kita dalam menambahkan data pembelian, melihat data yang ada, mengedit data yang sudah ada, dan menghapus data yang tidak diperlukan. Dengan demikian, pengguna dapat dengan mudah mengelola data pembelian gas LPG 3 KG.

DAFTAR PUSTAKA

Alamsyah, A. (2020). Dasar OOP Java (Inhertance). Abengz Media.

Mustakim, M., Fitrianingsih, N., & Fitriati, I. (2019). Pengembangan Aplikasi E-Raport Berbasis Graphical User Interface (GUI) dengan Menggunakan VB. Net 2010 di SMKN 10 Bima. Jurnal Pendidikan Mipa, 9(1), 67-75.

Wisnu Wirawan, P. (2010, August). Model-View-Controller (MVC) Design Pattern Untuk Aplikasi Perangkat Bergerak Berbasis Java. In PROSIDING SEMINAR NASIONAL ILMU KOMPUTER UNIVERSITAS DIPONEGORO 2010.

Maulani, J. (2020). Penerapan metode waterfall pada pengembangan aplikasi sistem informasi jasa dan penjualan dengan pemodelan berorientasi objek. Technologia: Jurnal Ilmiah, 11(2), 64-70.