**LAPORAN**

**TUGAS BESAR PEMROGRAMAN BERBASIS OBJEK**



Disusun Oleh:

Stephanie Debora Assa – NIM 1204220134

Refangga Lintar Prayoga – NIM 1204220137

**PROGRAM STUDI SISTEM INFORMASI**

**FAKULTAS TEKNOLOGI INFORMASI DAN BISNIS**

**INSTITUT TEKNOLOGI TELKOM SURABAYA**

**2023**

# Daftar Isi

[Daftar Isi 2](#_Toc142033063)

[Daftar Gambar 3](#_Toc142033064)

[BAB 1 PENDAHULUAN 4](#_Toc142033065)

[1.1 Latar Belakang 4](#_Toc142033066)

[BAB 2 PEMBAHASAN 5](#_Toc142033067)

[2.1 Login 5](#_Toc142033068)

[2.2.1 Property Menu 8](#_Toc142033069)

[2.2.3 Property Images Menu 12](#_Toc142033070)

[2.2.4 Owner Type Menu 14](#_Toc142033071)

[2.2.5 Client Menu 15](#_Toc142033072)

[2.2.6 Sale Menu 16](#_Toc142033073)

[3. Penjelasan Syntax Java 18](#_Toc142033074)

# Daftar Gambar

[Gambar 1. 1 Login 5](#_Toc142032950)

[Gambar 2. 1 Tampilan Dashboard 7](#_Toc142032964)

[Gambar 2. 2 Menu Property 8](#_Toc142032965)

[Gambar 2. 3 Remove Property 9](#_Toc142032966)

[Gambar 2. 4 Edit Property 9](#_Toc142032967)

[Gambar 2. 5 Master Properties 10](#_Toc142032968)

[Gambar 2. 6 Add Property Type 11](#_Toc142032969)

[Gambar 2. 7 Add Property Images 12](#_Toc142032970)

[Gambar 2. 8 Preview Property Image 12](#_Toc142032971)

[Gambar 2. 9 Delete Property Images 13](#_Toc142032972)

[Gambar 2. 10 Master Owner 14](#_Toc142032973)

[Gambar 2. 11 Master Client 15](#_Toc142032974)

[Gambar 2. 12 Property Sale Page 16](#_Toc142032975)

# BAB 1 PENDAHULUAN

## Latar Belakang

Pada zaman sekarang bisnis yang bergerak dalam dunia selling properti sangat aktif dalam perkembangan dan pertumbuhannya. Semakin banyak masyarakat yang menganggap properti bisa sebagai jalan pintas atau opsi untuk menambah dan menyimpan aset kekayaan mereka sebagaimana halnya crypto, reksadana atau pembelian aset-aset perusahaan lainnya. Semakin tingginya permintaan dan persaingan di bidang ini menuntut para agen-agen properti atau *owner selling* untuk menjadi lebih efisien lagi dalam mencatat dan mengelola data penjualan properti.

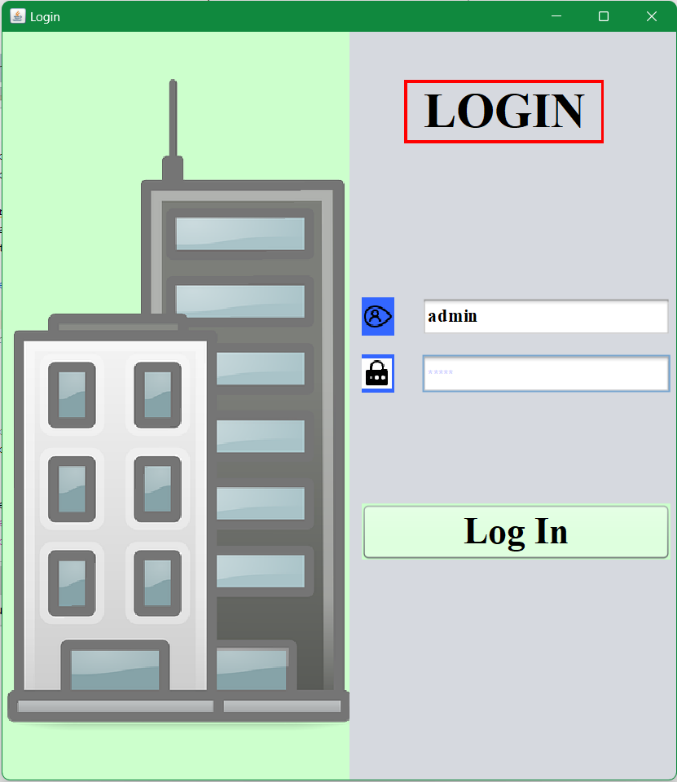
Maka dari itu melihat peluang-peluang dalam bidang ini, kelompok 5 (lima) mendapatkan ide untuk membuat sebuah *App* dengan menggunakan GUI (*Graphical User Interface*) di Java untuk pencatatan *selling owner* sehingga dapat membantu *owner* atauagen dalam pembukuannya menjadi lebih relevan. *App* ini bertujuan untuk menyediakan alat yang mudah dan efisien dalam penggunaannya bagi para *selling owner* atau agen properti untuk mengelola dan menyimpan data penjualan properti *client*, termasuk informasi tentang properti yang dijual, data pelanggan, harga, dan status penjualan.

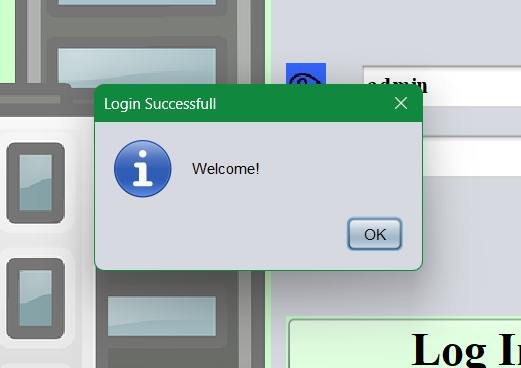
Dari pada menggunakan metode pencatatan manual dalam mencatat transaksi penjualan properti yang dapat menyebabkan terjadinya kebingungan dan kesalahan. Maka dengan adanya *App* ini data penjualan dapat diatur secara terstruktur, meminimalkan kesalahan, dan meningkatkan efisiensi dalam pencatatan data. *App* ini juga dapat membantu mengawasi progres dan perkembangan dari berbagai transaksi penjualan properti yang sedang berlangsung. Hal ini membantu *owner* atau agen properti untuk mengetahui status penjualan dan mengidentifikasi prospek penjualan.

Dalam *App* ini setiap *user* dapat dengan mudah mengakses dan mengubah data penjualan properti. *App* ini juga masih dapat ditingkatkan lagi fitur-fiturnya untuk dapat lebih berkembang sesuai kebutuhan *owner* atau agen properti dalam meningkatkan pertumbuhan bisnisnya dengan melihat kebutuhan tambahan dari bisnis ini.

# BAB 2 PEMBAHASAN

## 2.1 Login





Gambar 1. 1 Login

Pada tampilan awal dalam App Joy House, *user* akan pertama kali mendapati tampilan menu untuk login sesuai dengan user dan password yang sudah di tentukan. Kemudian setelah mengisi user dan password maka *App* akan langsung menampilkan notifikasi bahwa login berhasil dengan notif “Login Successful” serta “Welcome”, seperti pada gambar diatas.

2.2 FITUR-FITUR JOY HOUSE APP



Gambar 2. 1 Tampilan Dashboard

Setelah berhasil login, maka app akan langsung membawa user masuk pada tampilan dashboard dari Joy House app, seperti pada gambar diatas. Dalam App terdapat 6 (enam) menu yang dapat membantu dalam input serta penyimpanan data dari selling property oleh owner atau agen seperti *property, property type, property images, owner, client,* dan *sale*. Keenam menu tersebut dibuat untuk memudahkan dalam proses input dan pembaruan data penjualan properti. Pengguna dapat menggunakan menu-menu tersebut untuk mengatur dan mengelola informasi tentang properti yang sedang dijual, termasuk mengunggah foto-foto properti, memasukkan deskripsi dan detail properti, serta memperbarui informasi jika ada perubahan.

Dengan adanya enam menu tersebut, pengguna akan dapat dengan mudah mengakses dan melihat status terkini dari setiap properti yang dijual. Menu-menu ini juga memberikan opsi untuk memantau kinerja penjualan secara keseluruhan, sehingga *user* dapat menilai performa penjualan dan mengidentifikasi area yang perlu diperbaiki atau ditingkatkan.

Dalam hal ini, Joy House app berperan sebagai alat yang efisien dan handal untuk mendukung proses penjualan properti. Dengan menggunakan menu-menu yang ada, pengguna dapat meningkatkan produktivitas mereka dan lebih fokus pada strategi pemasaran yang tepat untuk mencapai keberhasilan dalam bisnis penjualan properti.

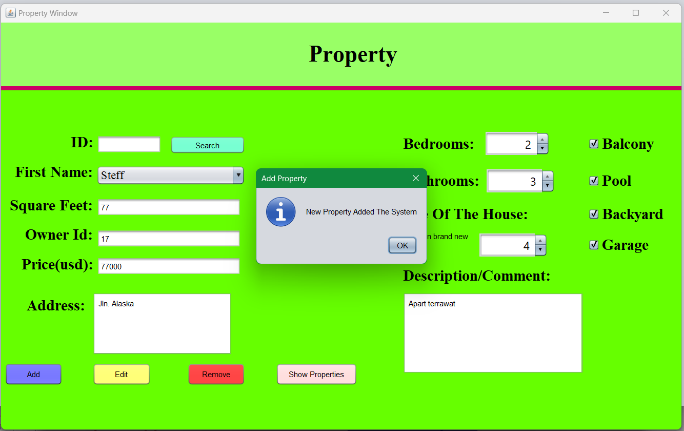
Dalam Joy House *app* juga terdapat fitur-fitur *add,edit,remove*, serta *refresh* yang membantu *user* dalam proses *input* dan *update* data penjualan properti. Setelah memahami garis besar fitur-fitur utama ini, mari lihat lebih detail lagi tentang penjelasan fitur-fitur setiap menu dalam subbab-subbab tersesbut.

### 2.2.1 Property Menu

*Property* menu Joy House app berfungsi sebagai pusat kontrol bagi *user* untuk mengelola seluruh informasi terkait properti yang sedang atau akan dijual. Dengan mengakses menu ini, pengguna dapat dengan mudah melihat daftar lengkap semua properti yang telah *user* tambahkan sebelumnya ke dalam sistem.

* Menambah Properti (Add):

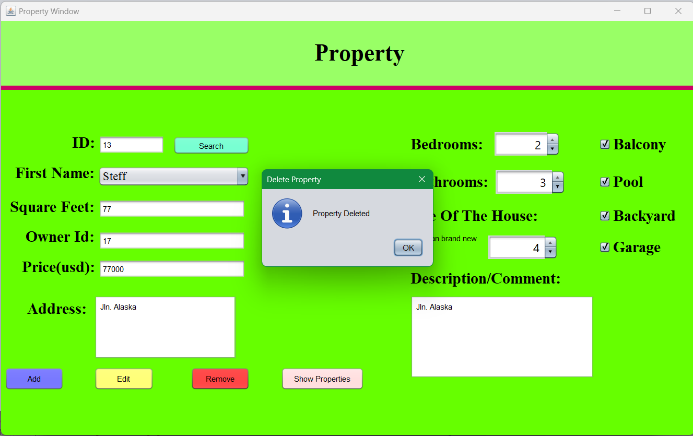
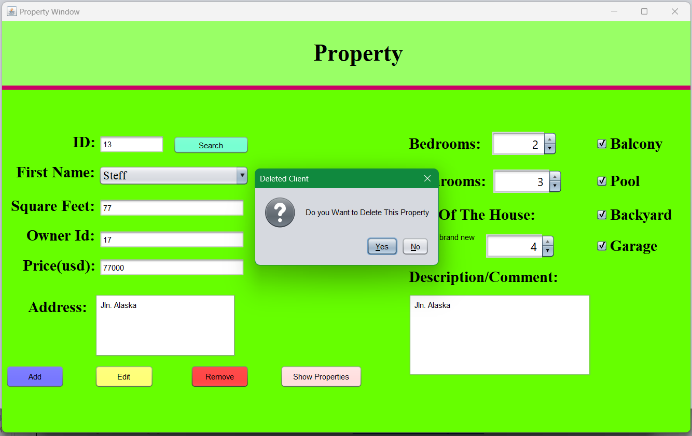
Pengguna dapat dengan cepat menambahkan properti baru ke dalam daftar dengan mengisi formulir atau melengkapi informasi yang relevan, seperti alamat, deskripsi, ukuran, harga, dan foto-foto properti. Setelah melengkapi data tersebut, pengguna dapat menyimpannya dan properti baru tersebut akan secara otomatis dimasukkan ke dalam daftar properti yang ada.



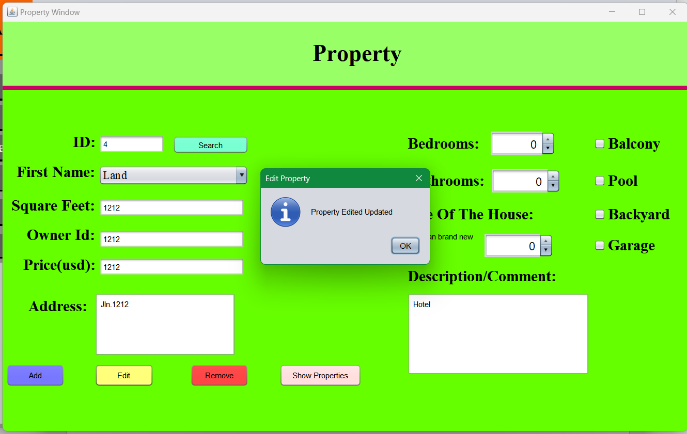
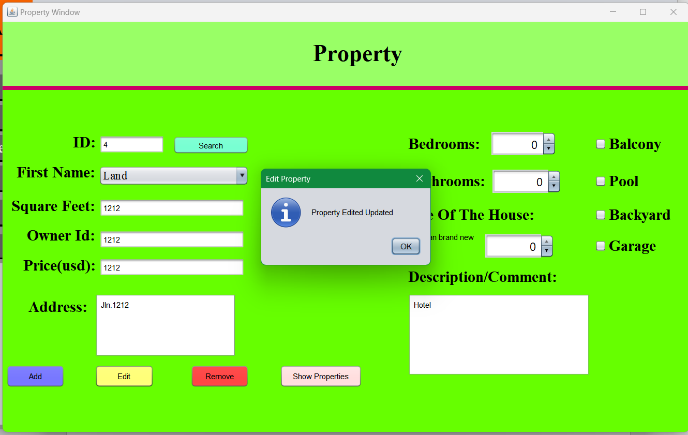
Gambar 2. 2 Menu Property

* Menghapus Properti (Remove):

Jika suatu properti sudah tidak lagi dijual atau tidak relevan untuk ditampilkan dalam daftar, pengguna dapat dengan mudah menghapusnya melalui menu "Remove". Tindakan ini akan menghapus properti dari daftar sehingga tidak lagi ditampilkan di tampilan dashboard atau menu lainnya.



Gambar 2. 3 Remove Property

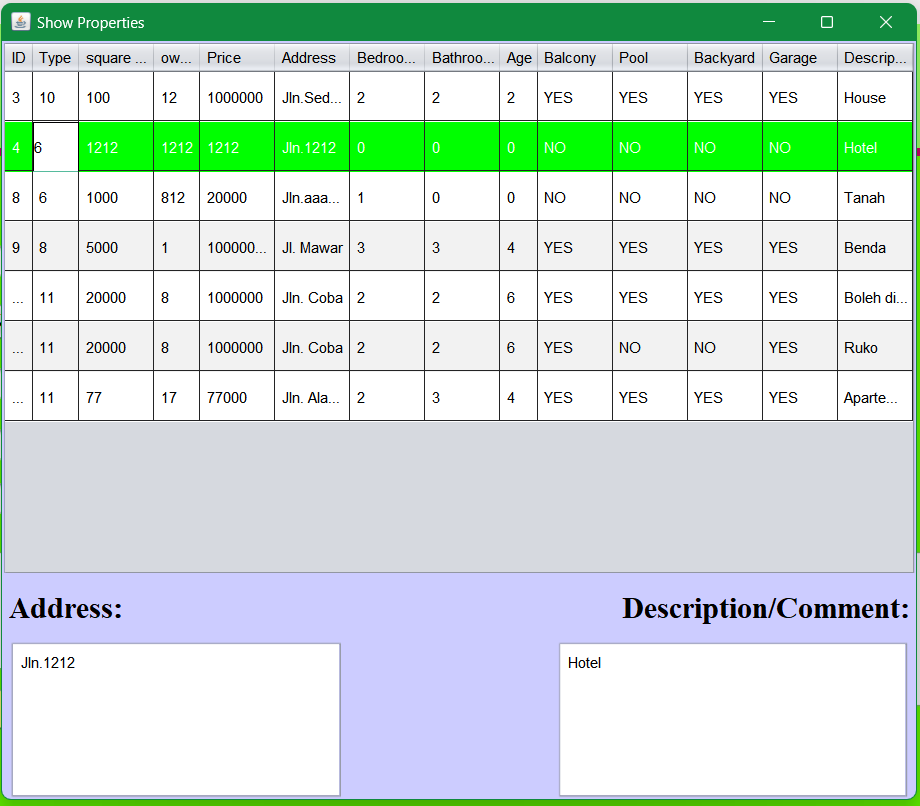


Gambar 2. 4 Edit Property

* Mengedit Informasi Properti (Edit):

Jika ada perubahan atau pembaruan pada properti yang sudah ada, pengguna dapat mengakses menu "Edit" untuk mengubah informasi yang terkait. Misalnya, jika ada penyesuaian harga, perubahan deskripsi, atau perubahan kondisi properti, pengguna dapat memperbarui data tersebut melalui menu ini. Hal ini memastikan bahwa data properti selalu akurat dan terbaru.

Refresh atau memperbarui tampilan daftar properti memungkinkan *user* untuk memperbarui tampilan daftar properti. Misalnya jika *user* baru saja menambahkan properti baru atau melakukan perubahan pada data properti yang sudah ada, mereka dapat menggunakan opsi "Refresh" untuk menyegarkan daftar sehingga perubahan terbaru dapat terlihat. Dengan adanya fitur-fitur yang disediakan *user* dapat secara efisien mengelola seluruh inventaris properti mereka dengan mudah. Hal ini membantu meningkatkan efisiensi dalam proses penjualan properti dan memastikan data yang ditampilkan dalam Joy House app selalu akurat dan terkini seperti pada tampilan gambar dibawa.

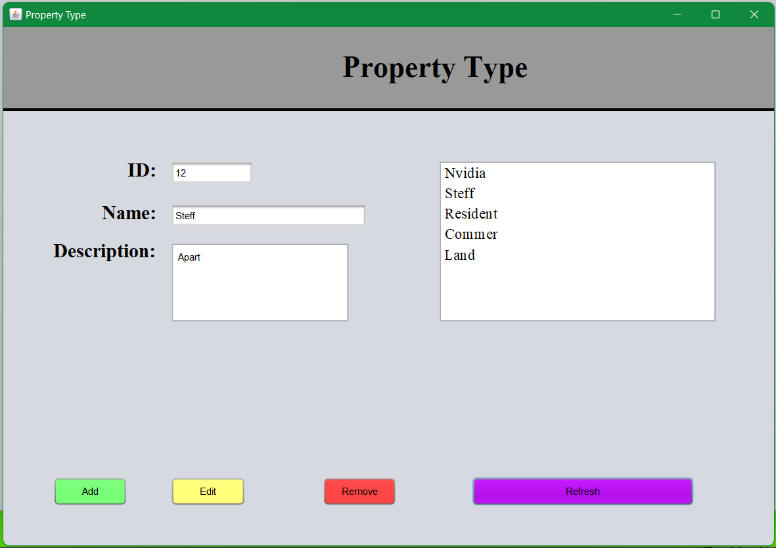
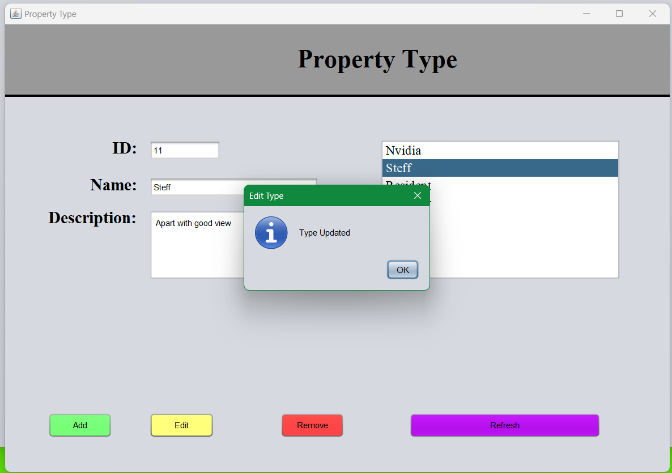
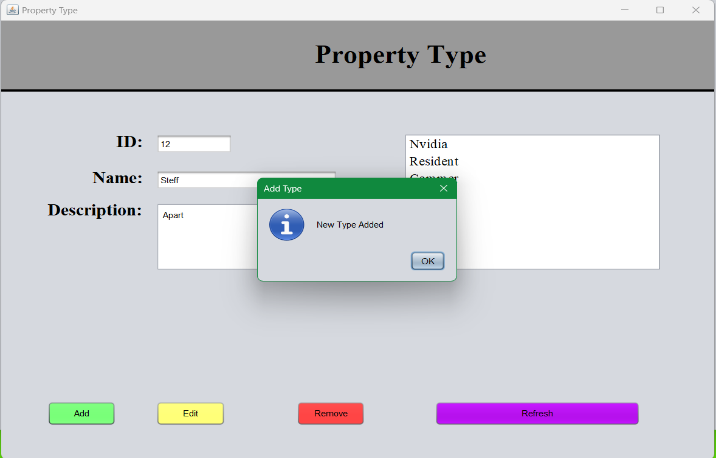


Gambar 2. 5 Master Properties

Selain itu, menu ini juga memudahkan *user* untuk memiliki kontrol penuh atas portofolio properti mereka dalam satu tempat yang terorganisir dengan baik.

2.2.2 Property Type Menu

Dalam pengelolaan properti, *user* memiliki akses ke berbagai jenis properti yang berbeda. Fitur yang tersedia pada menu ini mirip dengan yang ada di menu Properti, dimana memungkinkan *user* untuk melakukan berbagai aktivitas seperti pada salah satu contoh gambar berikut.



Gambar 2. 6 Add Property Type

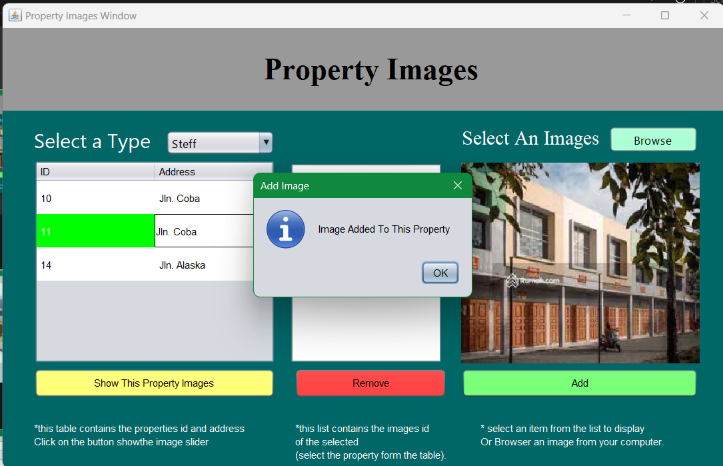
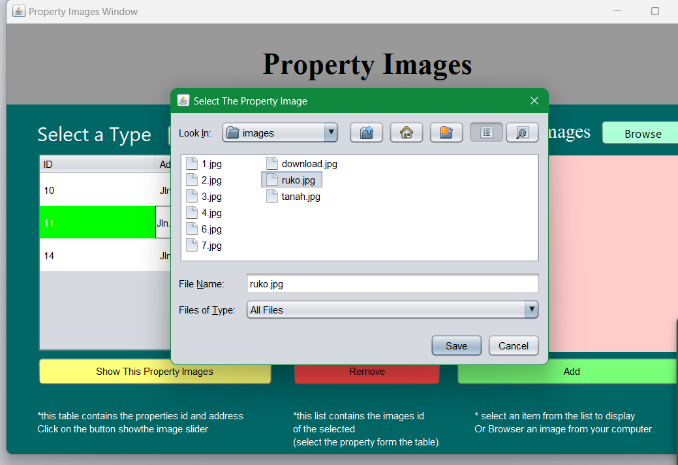
* Menambahkan Jenis Properti Baru (*Add*):

*User* dapat dengan mudah menambahkan jenis properti baru ke dalam sistem. Hal ini memungkinkan mereka untuk mengakomodasi berbagai jenis properti yang mungkin belum ada sebelumnya. Saat menambahkan jenis properti baru, *user* akan diminta untuk mengisi informasi penting terkait properti seperti pada gambar proses *add type* dibawa. Kemudian setelah mengisi data yang dibutuhkan maka *user* selanjutnya menekan *button* *add*, maka setelah itu akan muncul notifikasi menyetujui *add type*, setelah di “Ok” maka *list* dalam *property type* akan langsung terbaharui setelah menekan *button* *refresh*.

Sama halnya pada fitur-fitur lainnya dalam menu dapat dilakukan sama seperti pada fitur *add* diatas.

### 2.2.3 Property Images Menu

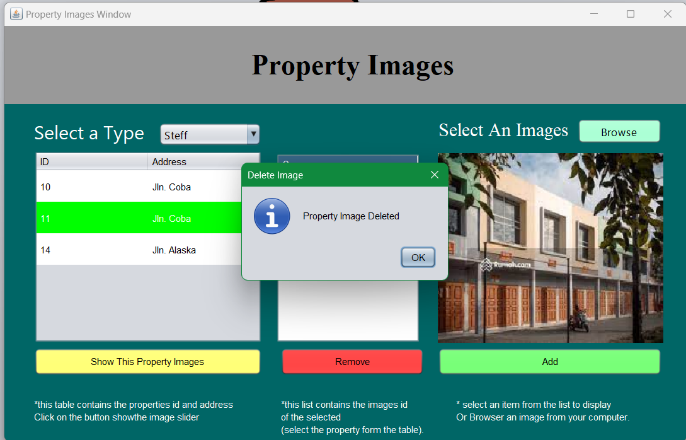
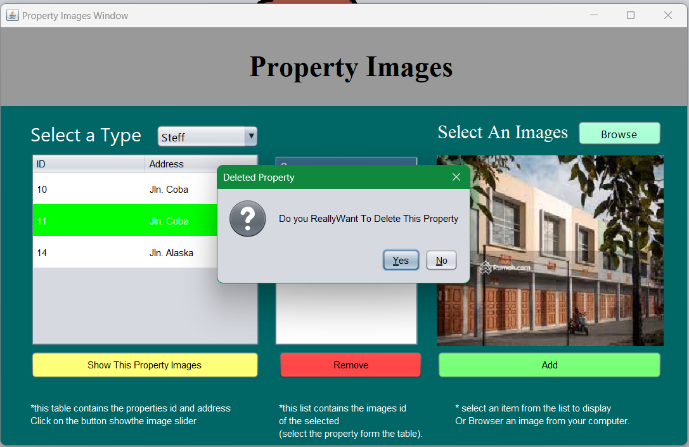
Menu ini merupakan salah satu fitur yang sangat berguna dalam mengelola gambar-gambar properti. Dengan menu ini, *user* dapat melakukan berbagai tindakan yang berhubungan dengan gambar properti secara efisien dan mudah. Dengan adanya menu ini, mengelola gambar-gambar properti menjadi lebih efisien dan tertata. Dengan fitur add, browse, dan remove gambar properti, memberikan kendali penuh kepada *user* untuk memastikan visualisasi properti yang tepat serta sesuai dengan kebutuhan. Selain itu, fitur-fitur ini juga membantu meningkatkan pengalaman *user* dalam mengelola properti mereka dengan lebih baik.



Gambar 2. 7 Add Property Images



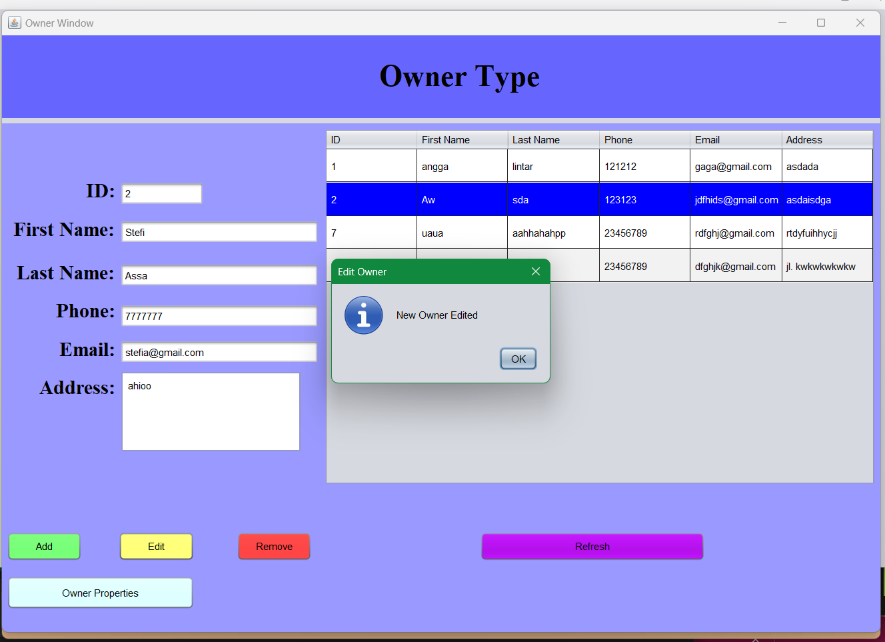
Gambar 2. 8 Preview Property Image

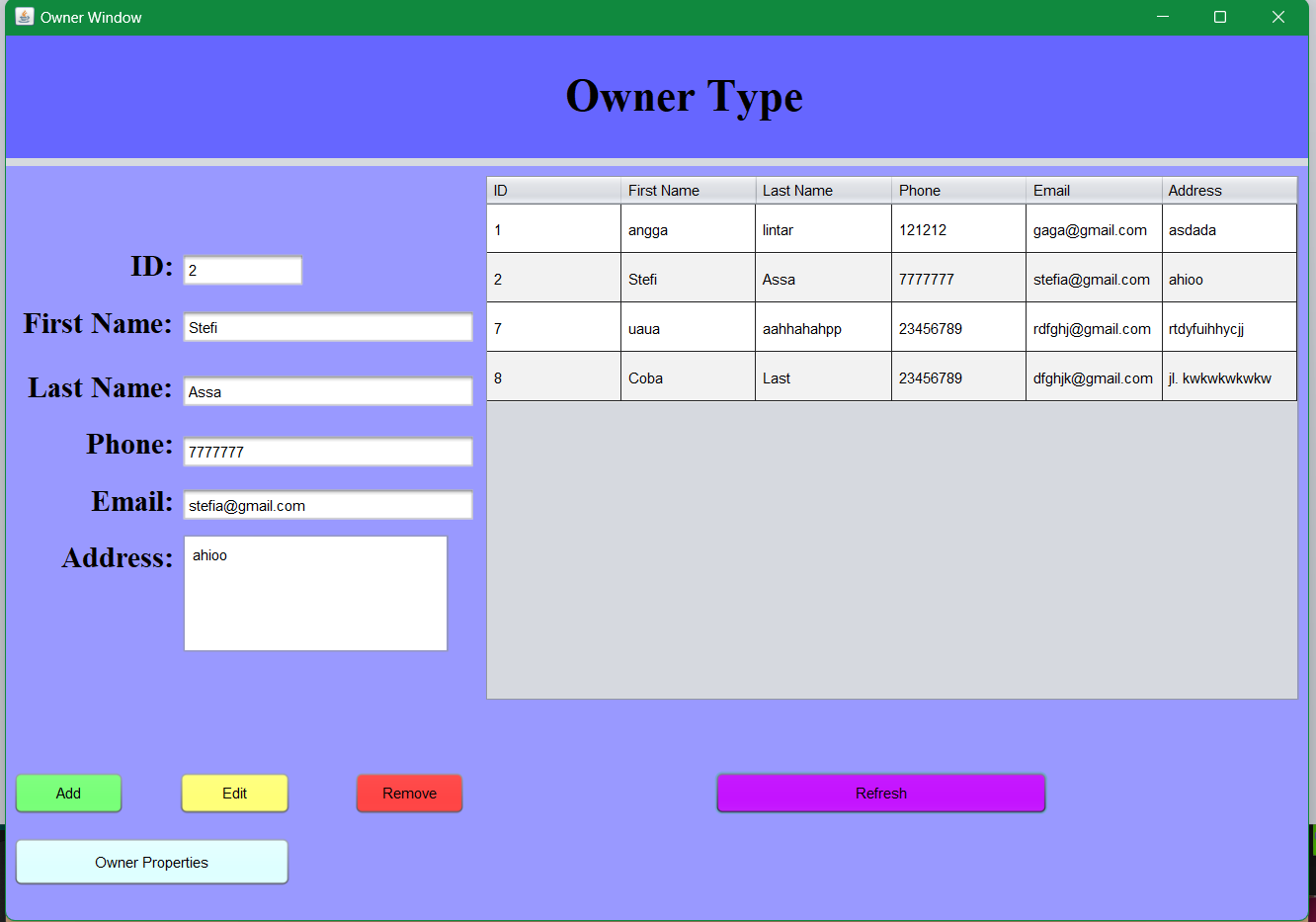


Gambar 2. 9 Delete Property Images

### 2.2.4 Owner Type Menu

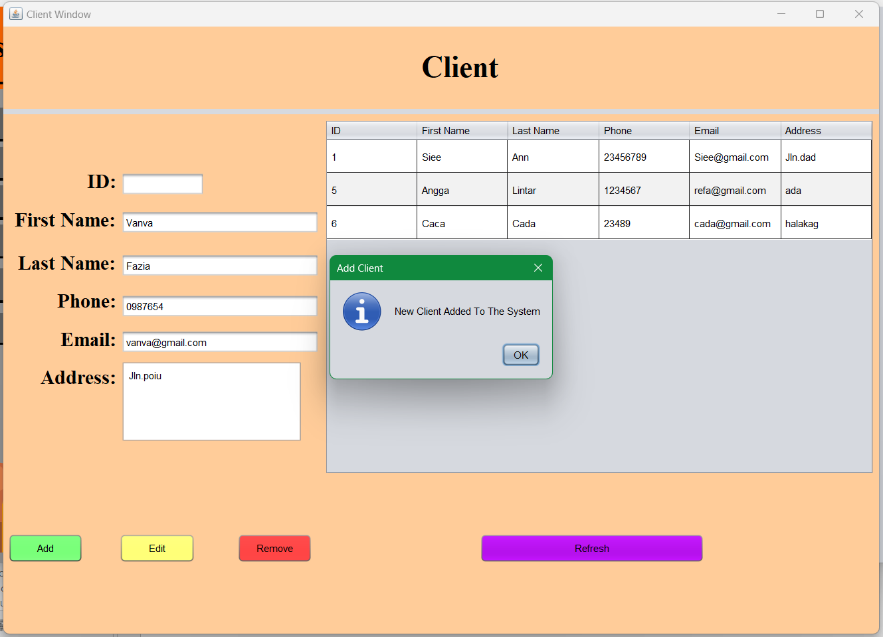
Pada menu ini *user* dapat mengelola data *owner* properti. *User* dapat menambahkan data *owner* baru, mengedit informasi *owner* yang sudah ada, menghapus data *owner*, dan memperbarui tampilan daftar data *owner* dalam tabel, tentunya fitur-fitur tersebut dapat dilakukan setelah mengisi beberapa informasi yang dibutuhkan dalam menu *owner type* ini. Untuk praktik dari fitur-fitur diatas dapat dilihat pada gambar-gambar dibawa ini.

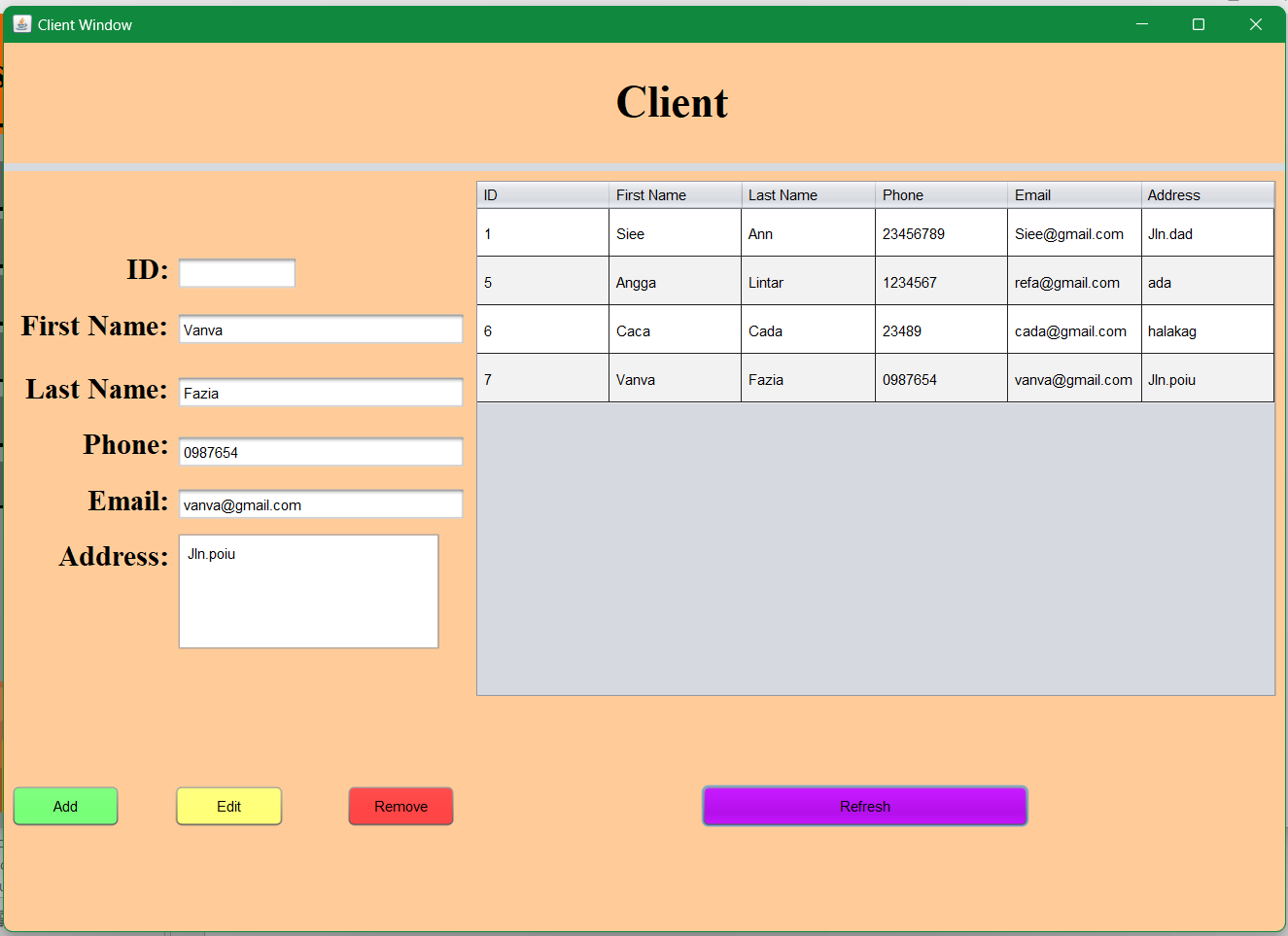




Gambar 2. 10 Master Owner

### 2.2.5 Client Menu

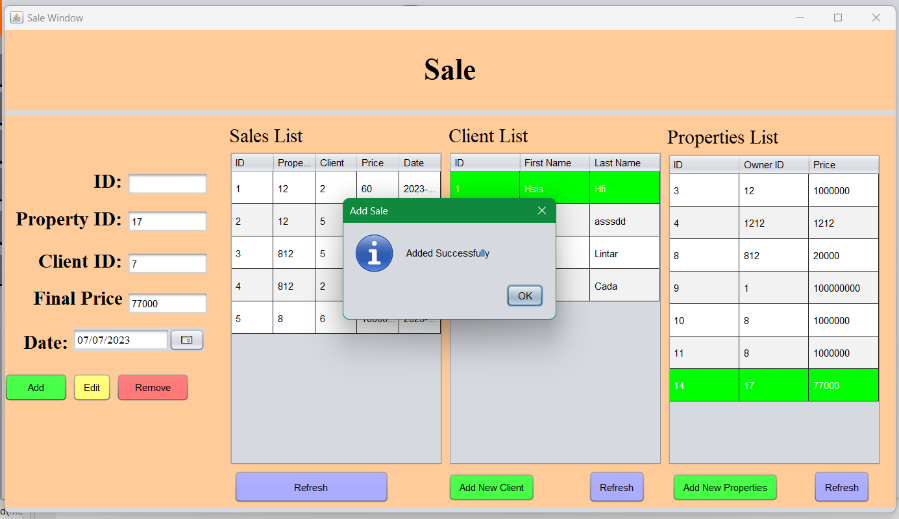


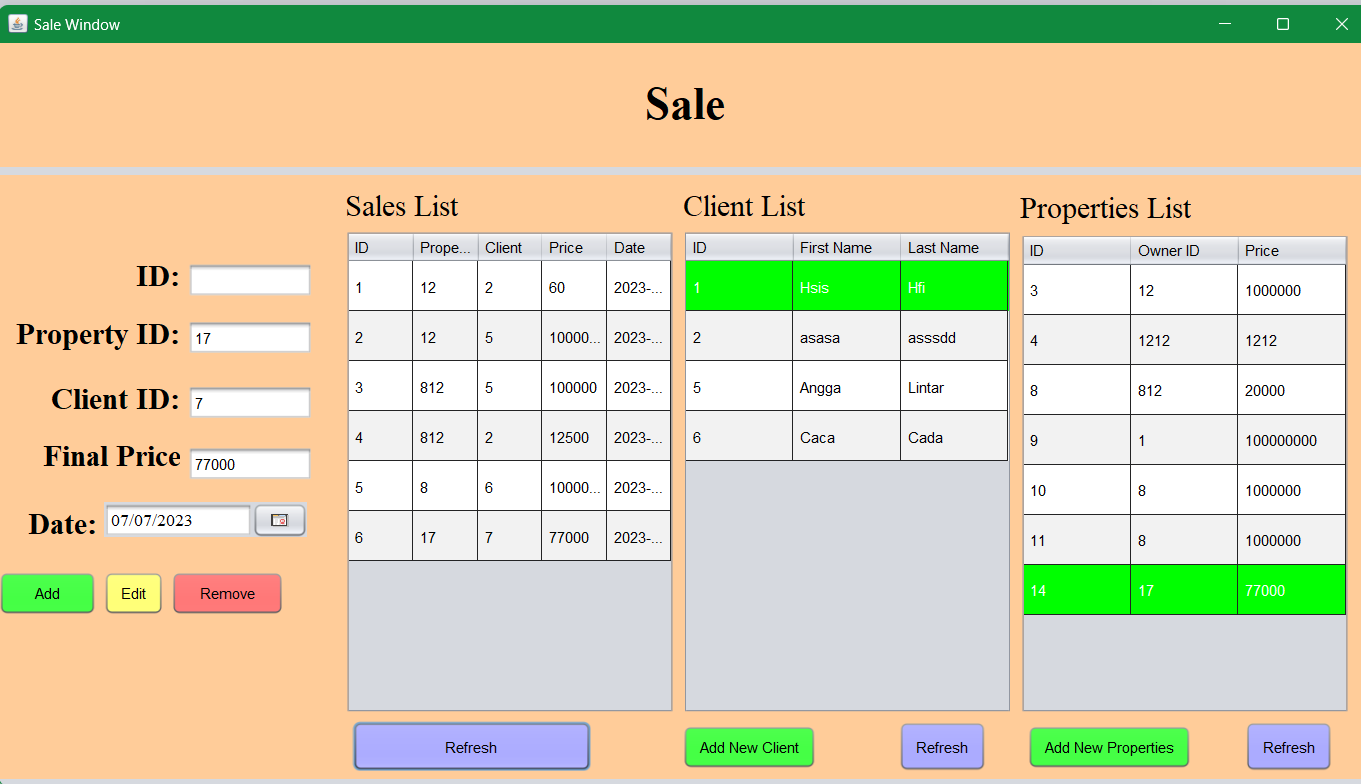


Gambar 2. 11 Master Client

Menu ini memungkinkan *user* untuk mengelola data klien atau calon pembeli properti. Sama halnya seperti pada menu-menu sebelumnya, pengguna dapat menambahkan data klien baru, mengedit informasi klien yang sudah ada, menghapus data klien, serta memperbarui tampilan daftar klien dengan mengisi informasi yang dibutuhkan dalam menu. Setelah memasukkan informasi yang dibutuhkan, maka selanjutnya *user* dapat menekan *button* *add,edit* atau *remove* pada menu sesuai dengan kebutuhan *user.* Maka langkah terakhir, *user* dapat menekan *button refresh* agar data dalam tabel dapat terbaharui.

### 2.2.6 Sale Menu





Gambar 2. 12 Property Sale Page

Menu Sale merupakan fitur yang sangat penting dalam *app* manajemen penjualan properti ini. Melalui menu ini, *user* diberikan kemudahan dalam melihat dan mengelola seluruh transaksi penjualan properti yang terjadi. Berbagai fungsi yang disediakan dalam menu Sale membantu *user* untuk melakukan beberapa tindakan krusial, di antaranya seperti menambah data ransaksi baru, mengedit informasi transaksi yang sudah ada, menghapus data transaksi, dan memperbarui tampilan daftar transaksi penjualan

Dengan adanya Menu Sale dalam app manajemen penjuallan properti ini, proses pengelolaan transaksi penjualan menjadi lebih terstruktur, aman, dan efisien. Para *owner* atau agen dapat dengan mudah mengakses dan mengelola data transaksi, memantau tingkat naik turun penjualan, serta memberikan layanan yang lebih baik kepada klien. Selain itu, *histori* transaksi yang tersimpan dengan rapi dalam basis data di *app* ini akan menjadi sumber informasi berharga untuk analisis bisnis ini dan pengambilan keputusan strategis di masa depan oleh *owner* atau agen.

## 3. Penjelasan Syntax Java

A screenshot of a computer code

Description automatically generated**3.1 P\_Client,P\_Owner, P\_Sale**

Dapat kita lihat pada P\_Client yang sama dengan P\_Owner dan P\_Sale type data dan getter setter

A screen shot of a computer program

Description automatically generatedBerisi Constructor yang isinya dapat mengakses semua type data menggunakan this pada P\_Client yang sama dengan P\_Owner dan P\_Sale

A screen shot of a computer screen

Description automatically generatedBerisi boolean public yang type add atau menambahkan Client. Menggunakan validasi database mysql menggunakan insert into dan diisi sesuai value nya dan hampir sama dengan P\_Client yang sama dengan P\_Owner dan P\_Sale

A screenshot of a computer screen

Description automatically generatedBerisi edit Client yang menggunakan parameter P\_Client client dan menggunakan koneksi mysql. Dengan menggunakan update property set dan diisi oleh value yang diisi dan hampir sama dengan P\_Client yang sama dengan P\_Owner dan P\_Sale

A computer screen shot of a program code

Description automatically generatedBerisi perintah delete yang menggunakan parameter int clientId. Menggunakan fungsi delete sql yang isinya adalah delete from where id yang artinya menggunakan id sebagai penghapus data. Jika penghapus data berupa id , maka data yang lain akan ikut terhapus dan hampir sama dengan P\_Client yang sama dengan P\_Owner dan P\_Sale

A screen shot of a computer screen

Description automatically generatedBerisi arrayList yang isinya menampilkan data Client dari semua aspek. Menggunakan data select \* form property\_client maka data tersebut akan tampil sebagai bentuk array list. Dan hampir sama dengan P\_Client yang sama dengan P\_Owner dan P\_Sale

3.2 Koneksi

A screen shot of a computer program

Description automatically generatedBerisi koneksi penghubung antara mysql dan netbeans. Menggunakan mysql xampp dan mysql connector 8.033 jar sebagai jalur pihak ketiga. Dapat kita lihat nama database nya adalah realrestate1.

==================================================================================

**Di Java Class seperti P\_Client,P\_Owner, P\_Sale hampir memiliki syntax yang sama dengan konsep Create, Read, Update , Delete. Yang membedakan hanya diperuntukkan untuk kelas dan fungsi masing-masing.**

3.3 P\_TypeA screen shot of a computer code

Description automatically generated

Berisi type data dari P\_Type dan ada juga getter setter untuk memberi dan mengatur value.

A screen shot of a computer code

Description automatically generated

Berisi Constructor yang berisi parameter dari type data yang berupa this.

A screen shot of a computer program

Description automatically generated

Berisi semua alur dari Type yang mengadung add, update, delete. Semua kendali dari Type dapat ita lihat di method execTypeQuery Ini

A screen shot of a computer program

Description automatically generatedPada Hashmap ialah mengumpulkan data sesuai format aturan select \* from property\_type

A screen shot of a computer code

Description automatically generated

Pada Public GetType id adalah menampilkan type bersarakan id menggunakan integer id dan divalidasi juga oleh id

========================================================================

3.4 P\_Property

A screen shot of a computer screen

Description automatically generated

Berisi add image yang menambahkan image kedalam property.menggunakan file input output stream

A screen shot of a computer screen

Description automatically generated

Berisi hasmap untuk menampilkan image menggunakan byte bertipe array

A computer screen shot of a program code

Description automatically generatedGetImagebyid. Dalam fungsi tersebut terdapat method untuk menampilkan images berdasarkan id yang diinginkan dengan validasi select \* the images from ‘property\_images’ where id yang divalidasikan juga oleh id

A screen shot of a computer program

Description automatically generated

Berisi method remove image yang berisi menghapus image yang dipilih berdasarkan id