

**TUGAS AKHIR DATABASE
REAL ESTATE**



Anggota Kelompok:

Adiza Noor Daradjatun -

Muhammad Rizky Febriansyah - 2206830763

Aghna Syahputra Santoso - 2206051481

Muhammad Omar Luthy - 2206829452

Hutari Andini - 2206820680

**DEPARTEMEN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS INDONESIA**

2024

DAFTAR ISI

BAB I

PENDAHULUAN.....	2
1.1 Latar Belakang.....	2
1.2 Tujuan.....	2
1.3 Tabel Variabel.....	3

BAB II

RANCANGAN DATABASE.....	5
2.1 Rancangan Sistem Database.....	5
2.2 ER Diagram.....	6
2.2.1 Entitas.....	8
2.2.2 Relasi.....	9
2.3 Tabel Relasional.....	10

BAB III

IMPLEMENTASI DAN INTERPRETASI.....	12
3.1 Simulasi.....	12
3.1.1 SELECT * FROM TABEL.....	12
3.1.2 INSERT INTO __ VALUES __.....	12
3.1.3 UPDATE __ SET __ WHERE __.....	12
3.1.4 SELECT __ FROM __ JOIN __ WHERE __.....	12
3.2 Analisis.....	12
3.2.1 SELECT __ FROM __ JOIN __ GROUP BY __ ORDER BY __.....	12
3.2.2 SELECT __ FROM __ JOIN __ ON __.....	12
3.3 Implementasi Interaktif (GUI).....	12
3.3.1 Input transaksi baru.....	12
3.3.2 Pencarian informasi.....	12
3.3.3 Pencarian informasi property berdasarkan budget user.....	12
3.4 Testing GUI.....	12
3.4.1 User meng-input Sale_ID yang sudah ada.....	12
3.4.2 User meng-input Region yang tidak ada.....	12

BAB IV

PENUTUP.....	13
4.1 Kesimpulan.....	13
4.2 Saran.....	13

BAB I

PENDAHULUAN

1.1 Latar Belakang

Pasar properti di Melbourne telah menjadi salah satu pasar properti yang terkuat dan paling konsisten selama empat dekade terakhir. Dalam 2 tahun terakhir, Melbourne mengalami kenaikan pada harga properti dan banyak orang ingin menggunakan kesempatan ini untuk terjun ke pasar properti di Melbourne. Tetapi, ada beberapa tantangan yang harus disolusikan. Pertama adalah menjamin ketersediaan lahan termasuk lahan pinjaman dengan suku bunga standar yang dimana akan memberikan dukungan jangka panjang kepada sektor perumahan. Kedua adalah memastikan suplai perumahan yang memadai merupakan hal yang sangat penting. Mengingat relatif langkanya unit perumahan saat ini yang menyebabkan meningkatnya permintaan dan akibatnya harga rumah tetap tinggi.

Data Melbourne Housing Snapshot, yang dikumpulkan dari daftar publik Domain.com.au, memberikan gambaran berharga tentang pasar properti di Melbourne. Dataset ini mencakup beragam informasi penting, seperti alamat properti, jenis properti, kawasan, metode penjualan, jumlah kamar, harga jual, agen properti yang terlibat, tanggal penjualan, dan jarak dari pusat bisnis Melbourne (CBD). Meskipun data ini tidak memiliki kemampuan untuk memprediksi tren masa depan, namun memberikan wawasan yang kuat untuk pemetaan harga perumahan dan pemahaman tentang agen properti yang paling aktif di berbagai area berdasarkan tingkat keberhasilan penjualan mereka.

1.2 Tujuan

1. Untuk pemetaan harga perumahan.
2. Untuk memberikan pengetahuan tentang agen properti yang paling aktif di berbagai area berdasarkan harga jual.
3. Untuk membantu menyimpan informasi lengkap tentang properti, termasuk lokasi, tipe properti, ukuran, fasilitas, dan informasi harga
4. Membantu mencari dan menyaring properti berdasarkan berbagai kriteria, seperti lokasi, tipe properti, dan juga fasilitas untuk memudahkan pencarian oleh pengguna

1.3 Tabel Variabel

Variabel	Tipe data	Keterangan
PropertyID	Integer	Kode indeks properti
RoomID	Varchar	Kode indeks ruangan
LocationID	Varchar	Kode indeks lokasi
SaleID	Varchar	Kode indeks agen penjualan
Suburb	Object	<i>Suburb</i> letak properti
Address	Object	Alamat
Rooms	Integer	Jumlah kamar
Price	Float	Harga dalam Australian Dollar (AUD)
Method	Object	S - properti terjual; SP - properti terjual sebelum lelang; PI - harga properti tidak disetujui oleh vendor; VB - properti ditawarkan oleh vendor; SA - properti terjual setelah lelang.
Type	Object	h - rumah, cottage, villa, semi, terrace; u - unit, duplex; t - townhouse;
SellerG	Object	Agen properti
Date	Float	Tanggal terjual
Distance	Float	Jarak dari <i>central business district</i> (CBD)
Postcode	Float	Kode pos
Regionname	Object	<i>Region</i> letak properti
Propertycount	Float	Jumlah total properti di dalam suburb letak properti
Bedroom	Float	Jumlah kamar tidur
Bathroom	Float	Jumlah kamar mandi
Car	Float	Jumlah parkir mobil
Landsize	Float	Luas tanah

BuildingArea	Float	Luas bangunan
YearBuilt	Float	Tahun pembangunan
CouncilArea	Object	<i>Council</i> letak properti

BAB II RANCANGAN DATABASE

2.1 Rancangan Sistem Database

Dalam merancang sebuah *database*, diperlukan sistem yang tepat untuk menjamin data dapat disimpan dan dikelola secara efisien. Sistem ini mencakup identifikasi entitas, atribut, serta hubungan antar entitas yang sesuai dengan kebutuhan sistem. Maka akan dilihat definisi entitas, dan hubungan antar entitas.

1) Entitas

Entitas adalah objek atau hal yang dapat diidentifikasi secara unik dan yang datanya perlu disimpan dalam database. Entitas adalah komponen utama yang representasinya dalam database memungkinkan pengelolaan dan penyimpanan data yang relevan. Terdapat dua tipe entitas, yaitu *strong entity type* dan *weak entity type*. Entitas yang keberadaannya tidak bergantung pada entitas lain disebut sebagai *strong entity type* sedangkan entitas yang keberadaannya bergantung pada entitas lain adalah *weak entity type*.

2) Relasi

Relasi atau hubungan adalah asosiasi antara dua atau lebih entitas yang menggambarkan bagaimana entitas-entitas tersebut saling berinteraksi. Relasi membantu memahami keterkaitan data dalam entitas yang berbeda dan memungkinkan integrasi data yang efisien. Ada beberapa jenis relasi yang umum dalam database:

- a. *One-to-One* (1-1): relasi ini terjadi ketika setiap entitas A berhubungan dengan hanya satu entitas B.
- b. *One-to-Many* (1-n): relasi ini terjadi ketika setiap entitas A berhubungan dengan lebih dari satu entitas B.
- c. *Many-to-One* (m-1): relasi ini terjadi ketika lebih dari satu entitas A berhubungan dengan hanya satu entitas B.
- d. *Many-to-Many* (m-n): relasi ini terjadi ketika lebih dari satu entitas A berhubungan dengan lebih dari satu entitas B.

Maka, berdasarkan definisi entitas, dapat dibuat 4 tabel yaitu tabel Properti, Agen Penjualan, Lokasi, dan Ruangan. Berdasarkan tabel variabel, maka akan dibuat sistem database sebagai berikut:

Tabel Properti								
Property_ID	Rooms_ID	Location_ID	Type	Distance	Landsize	Building_Area	YearBuilt	Address
0001	R2211	NMYA3067	h	2.5	202	200	1900	85 Turner St
0002	R2210	WMMA3042	h	2.5	156	79	1900	25 Bloomburg St
0003	R3320	SMPA3206	h	2.5	134	150	1992	5 Charles St
0004	R3321	NMDA3078	h	2.5	94	249	2014	40 Federation La
0005	R4312	WMHA3018	h	2.5	120	142	2007	55a Park St
Tabel Agen Penjualan								
Sale_ID	Property_ID	Date	Method	Price	SellerG			
1612BIS	0001	3/12/2016	S	1480000	Biggin			
1602BIS	0002	4/2/2016	S	1035000	Biggin			
1703BIP	0003	4/3/2017	SP	1465000	Biggin			
1703BII	0004	4/3/2017	PI	850000	Biggin			
1606NEV	0005	4/6/2016	VB	1600000	Nelson			
Tabel Lokasi								
Location_ID	Region_name	Council_Area	Suburb	Postcode	Property_Count			
NMYA3067	Northern Metropolitan	Yarra	Abbotsford	3067	4019			
WMMA3042	Western Metropolitan	Moonee Valley	Airport West	3042	3464			
SMPA3206	Southern Metropolitan	Port Phillip	Albert Park	3206	3280			
NMDA3078	Northern Metropolitan	Darebin	Alphington	3078	2211			
WMHA3018	Western Metropolitan	Hobsons Bay	Altona	3018	5301			
Tabel Ruangan								
Rooms_ID	No_Rooms	Bedroom	Bathroom	Car				
R2211	2	2	1	1				
R2210	2	2	1	0				
R3320	3	3	2	0				
R3321	3	3	2	1				
R4312	4	3	1	2				

Setiap tabel memiliki beda representasi. Berikut definisi per tabel:










1. **Properti:** merepresentasikan unit properti yang terjual.
2. **Agen Penjualan:** merepresentasikan agen yang menjual properti tersebut.
3. **Lokasi:** merepresentasikan lokasi dari unit properti yang terjual.
4. **Room:** merepresentasikan ruangan yang tersedia pada unit properti yang terjual.

Berikut adalah rincian *Primary Key* dan *Foreign Key*:

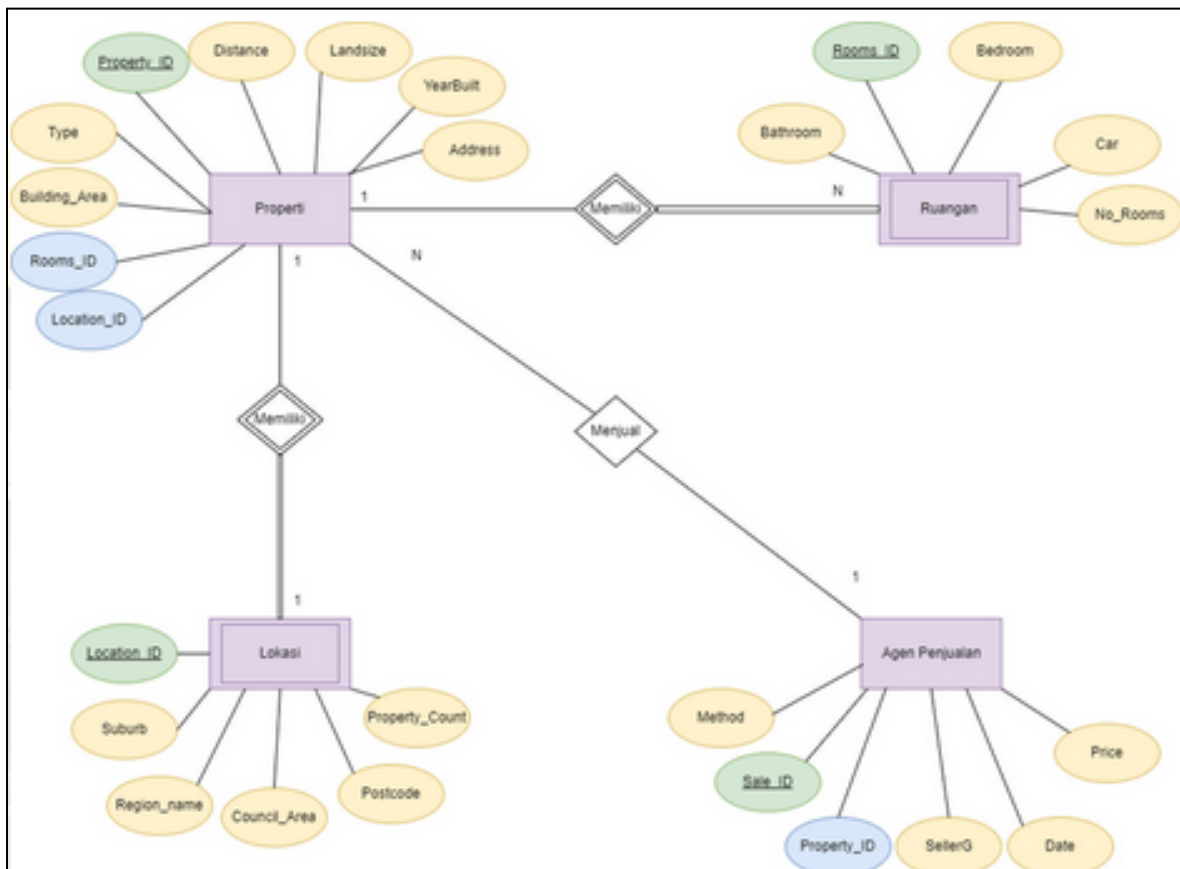
1. Tabel Properti
 - *Primary Key*: Property_ID
 - *Foreign Key*: Rooms_ID, Location_ID
2. Tabel Agen Penjualan
 - *Primary Key*: Sale_ID
 - *Foreign Key*: Property_ID
3. Tabel Lokasi
 - *Primary Key*: Location_ID
4. Tabel Ruangan
 - *Primary Key*: Rooms_ID

2.2 ER Diagram

Setelah dirancang sistem database, maka dapat dibuat sebuah ER Diagram untuk dapat dilihat rincian relasi setiap tabel. Untuk definisi karakteristik diagramnya bisa dilihat sebagai berikut.

Notasi	Arti
1. 	1. Entity
2. 	2. Weak Entity
3. 	3. Relationship
4. 	4. Identifying Relationship
5. 	5. Atribut
6. 	6. Atribut Primary Key
7. 	7. Atribut Multivalue
8. 	8. Atribut Composite
9. 	9. Atribut Derivatif

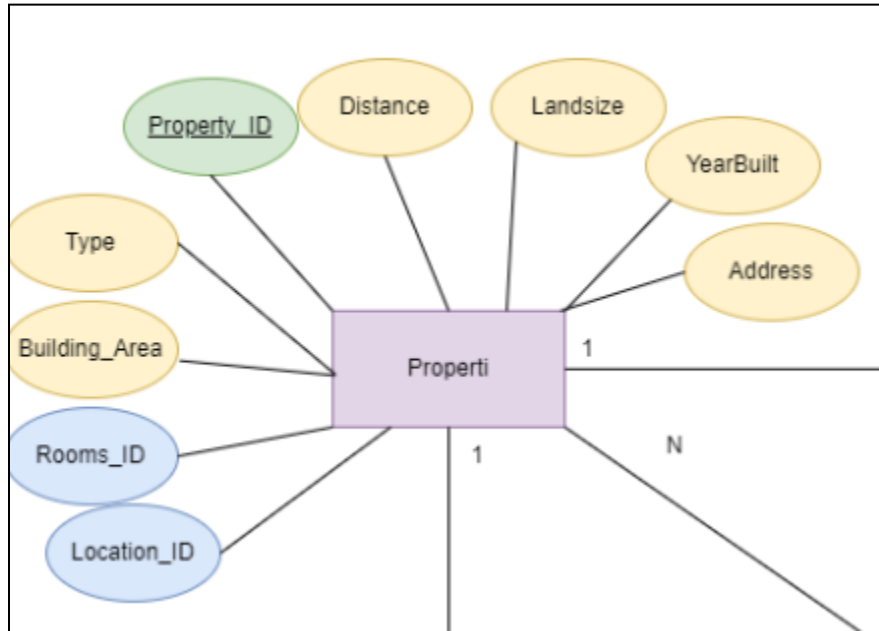
Maka, ER Diagramnya adalah,



Berikut adalah rincian-rincian entitas dan relasi antar entitasnya:

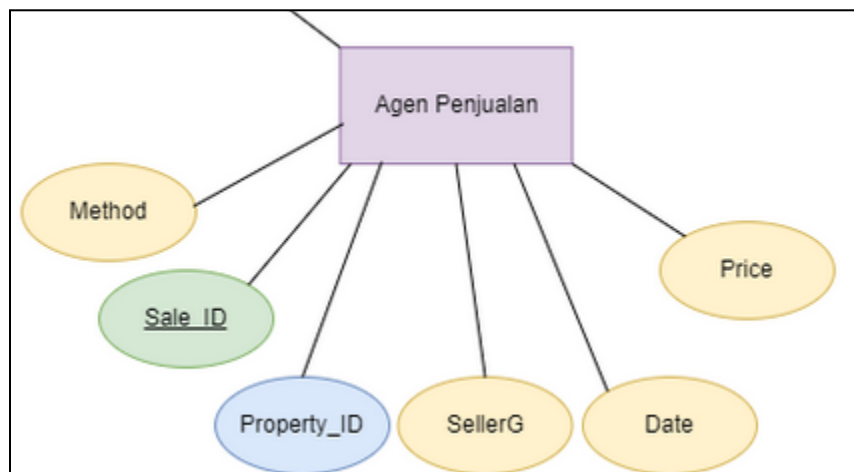
2.2.1 Entitas

a. Tabel Properti



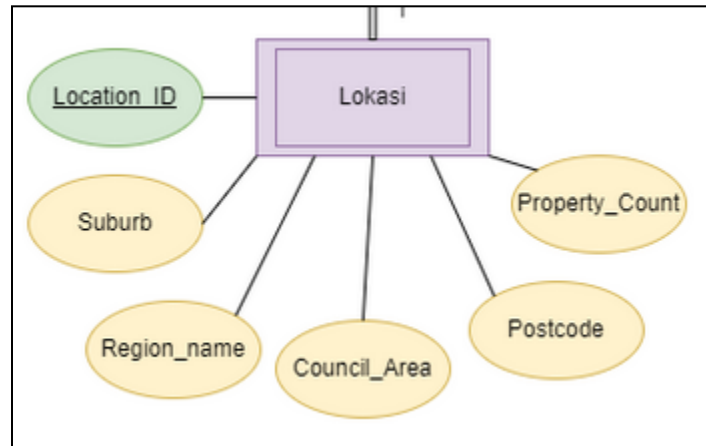
Sebuah properti (entitas) memiliki 9 atribut (*Rooms_ID*, *Location_ID*, *Building_Area*, *Type*, *Property_ID*, *Distance*, *Landsize*, *YearBuilt*, *Address*). *Property_ID* merupakan *Primary Key* sehingga di underline dan warna hijau. *Rooms_ID* dan *Location_ID* merupakan *Foreign Key* sehingga diwarnai biru untuk membedakannya dengan atribut lain.

b. Tabel Agen Penjualan



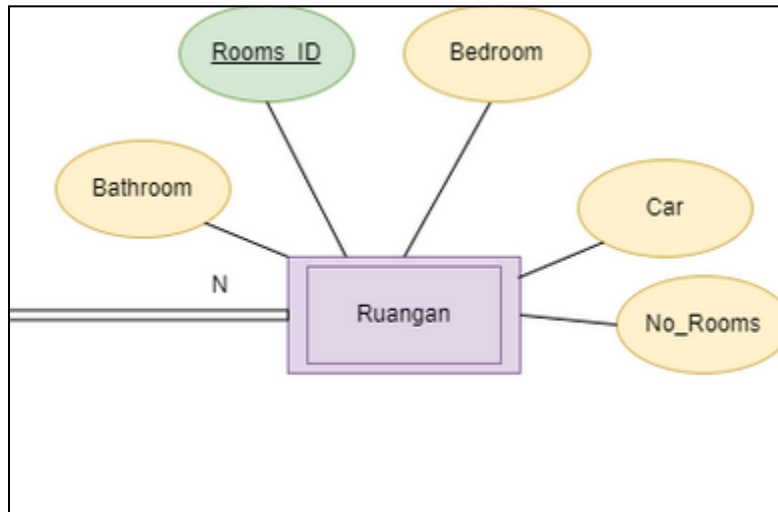
Sebuah agen penjualan (entitas) memiliki 6 atribut (*Method*, *Sale_ID*, *Property_ID*, *SellerG*, *Date*, dan *Price*). *Sale_ID* merupakan *Primary Key* sehingga di underline dan warna hijau. *Property_ID* merupakan *Foreign Key* sehingga diwarnai biru untuk membedakannya dengan atribut lain.

c. Tabel Lokasi



Sebuah Lokasi (entitas lemah) memiliki 6 atribut (*Location_ID*, *Suburb*, *Region_name*, *Council_Area*, *Postcode*, dan *Property_Count*). *Location_ID* merupakan *Primary Key* sehingga di underline dan warna hijau.

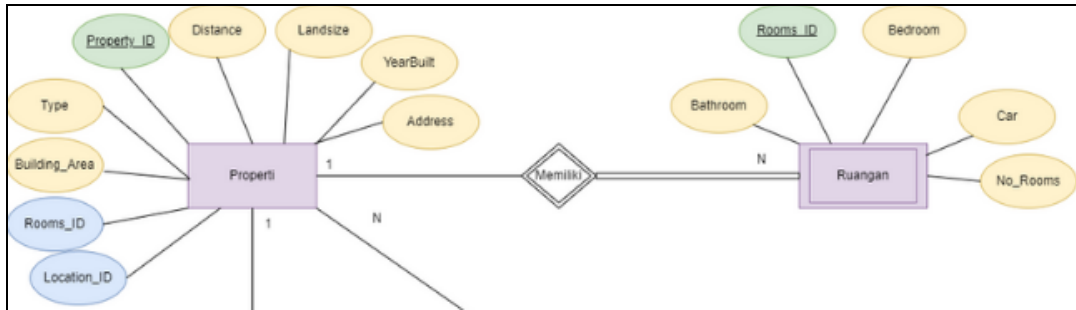
d. Tabel Ruangan



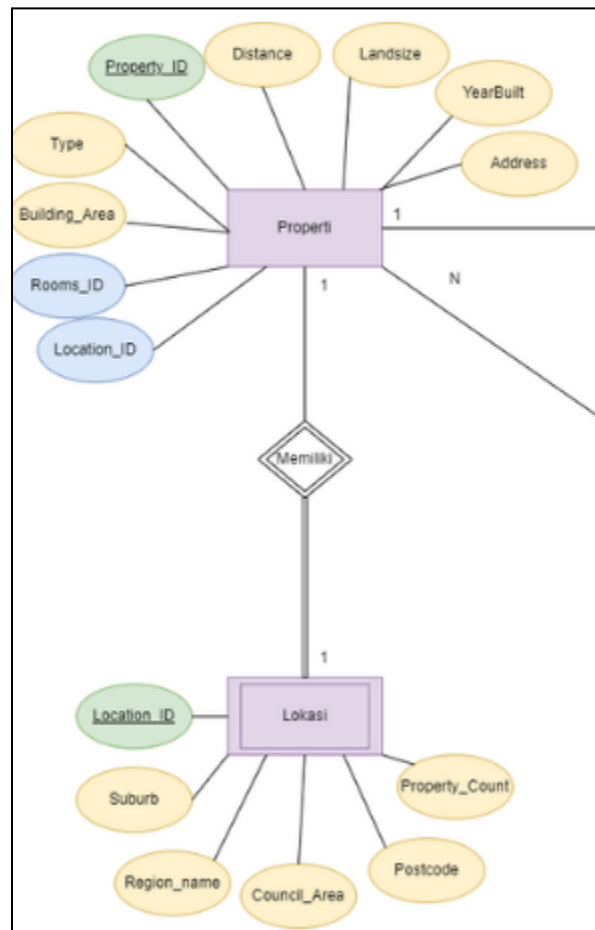
Sebuah ruangan (entitas lemah) memiliki 5 atribut (*Bathroom*, *Rooms_ID*, *Bedroom*, *Car*, dan *No_Rooms*). *Rooms_ID* merupakan *Primary Key* sehingga di underline dan warna hijau.

2.2.2 Relasi

1. Relasi properti dengan ruangan adalah sebuah properti memiliki beberapa ruangan (*One to Many*). sebaliknya, beberapa ruangan dimiliki oleh satu properti.



2. Relasi properti dengan lokasi adalah sebuah properti memiliki sebuah lokasi (*One to One*). Sebaliknya, setiap Lokasi dimiliki oleh satu properti.



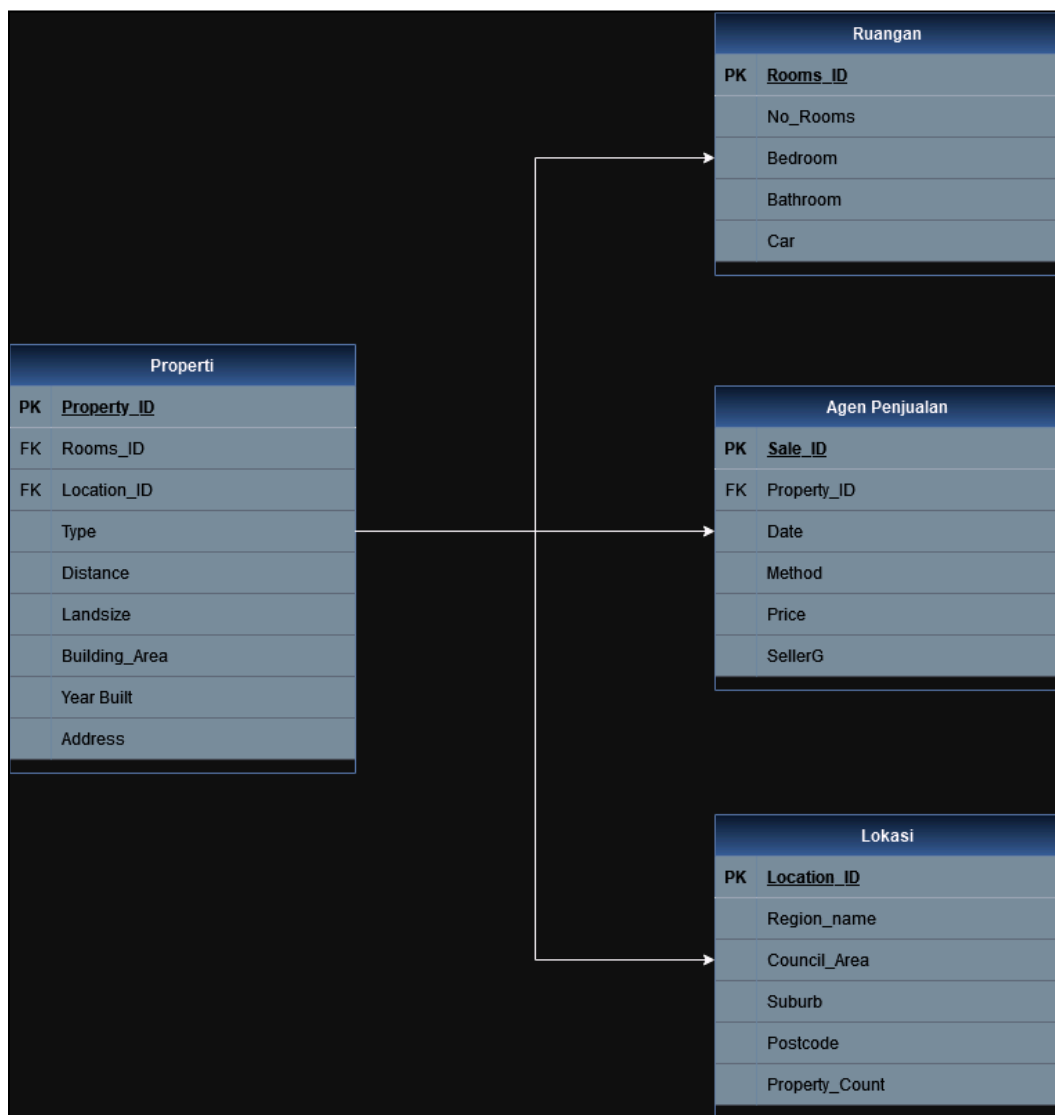
3. Relasi properti dengan agen penjualan adalah sebuah agen penjualan menjual banyak properti (*One to Many*).

2.3 Tabel Relasional

Dalam merancang sebuah database, diperlukan struktur basis data untuk memudahkan pembentukannya. Struktur basis data relasional mengorganisir data dalam bentuk tabel.

Hubungan antar tabel dalam struktur basis data relasional ini menjadi dasar dalam pembuatan database pada aplikasi yang diinginkan sama juga dengan ER Diagram. Langkah untuk menyusun tabel relasional adalah:

1. Tentukan tujuan dari basis data.
2. Kumpulkan data dan susun tabel dengan primary key dan kelas datanya. Dalam tabel, ada satu kolom yang berisi primary key dan kolom lainnya berisi kelas data yang penting untuk pengembangan basis data.
3. Identifikasi hubungan antara tabel-tabel tersebut.
4. Tentukan hubungan satu-ke-satu, satu-ke-banyak, dan banyak-ke-banyak untuk semua tabel, sesuai dengan yang telah didefinisikan dalam ER Diagram.



BAB III IMPLEMENTASI DAN INTERPRETASI

3.1 Simulasi

3.1.1 SELECT * FROM TABEL

Akan dilakukan simulasi untuk menampilkan tabel dengan menggunakan perintah `SELECT * FROM TABEL`. Misalkan akan ditampilkan semua kolom dari tabel `Properti`. Perintah query yang digunakan dan *output* yang dihasilkan adalah sebagai berikut:

```

54  --Simulasi 1
55  SELECT * FROM tabel_properti;
56
57

```

	Property_ID	Rooms_id	Location_id	Type	Distance	Landsize	Building_Area	YearBuilt	Adress
1	1	R2211	NM3067	h	2.5	202	126	1970	85 Turner St
2	2	R2210	NM3067	h	2.5	156	79	1900	25 Bloomburg St
3	3	R3320	NM3067	h	2.5	134	150	1900	5 Charles St
4	4	R3321	NM3067	h	2.5	94	126	1970	40 Federation La

3.1.2 INSERT INTO ____ VALUES ____

Akan dilakukan simulasi *input* data ke dalam tabel dengan menggunakan perintah `INSERT INTO ____ VALUES ____`. Misalkan akan di-*input* data *listing* properti baru ke dalam tabel `Properti`. Perintah query yang digunakan dan *output* yang dihasilkan adalah sebagai berikut:

```

57  --Simulasi 2
58  INSERT INTO tabel_properti(
59  Property_ID, Rooms_id, Location_id, Type, Distance, Landsize, Building_Area, YearBuilt, Adress
60  )VALUES(
61  '13581', 'R3322', 'EM3152', 'h', '14.7', '665', '126', '1970', '51 Amara Cr');
62  SELECT * FROM tabel_properti;
63

```

	Property_ID	Rooms_id	Location_id	Type	Distance	Landsize	Building_Area	YearBuilt	Adress
13578	13578	R3324	WM3016	h	6.8	436	126	1997	83 Power St
13579	13579	R4415	WM3016	h	6.8	866	157	1920	96 Verdon St
13580	13580	R4411	WM3013	h	6.3	362	112	1920	6 Agnes St
13581	13581	R3322	EM3152	h	14.7	665	126	1970	51 Amara Cr

3.1.3 UPDATE ____ SET ____ WHERE ____

Akan dilakukan simulasi memperbarui data pada tabel dengan menggunakan perintah `UPDATE ____ SET ____ WHERE ____`. Misalkan harga properti di tabel `Agen Penjualan` akan diperbarui. Perintah query yang digunakan dan *output* yang dihasilkan adalah sebagai berikut:

```

64 --Simulasi 3
65 UPDATE "tabel_agen_penjualan"
66 SET "Price" = '1064100.0'
67 WHERE "Sale_ID" = 'S04321BA';
68 SELECT*FROM tabel_agen_penjualan

```

	Sale_ID	Property_ID	Date	Method	Price	SellerG
4321	S04321BA	4321	19/11/2016	S	1064100	Barry
4322	S04322BA	4322	22/08/2016	S	1015000	Barry
4323	SP04323NE	4323	22/08/2016	SP	875000	Nelson
4324	S04324BA	4324	25/02/2017	S	1020000	Barry

3.1.4 SELECT __ FROM __ JOIN __ WHERE __

Akan dilakukan simulasi menyaring data dari berbagai tabel dengan menggunakan perintah SELECT __ FROM __ JOIN __ WHERE __. Misalkan akan dicari properti dengan ketentuan tertentu. Perintah query yang digunakan dan *output* yang dihasilkan adalah sebagai berikut:

```

82 ---Simulasi 4 untuk mencari dan menyaring properti
83 SELECT p.Property_ID, p.Rooms_id, r.no_rooms, r.Bedrooms, r.Bathrooms, r.Car,
84 l.Location_id, l.Region_name, l.Council_Area, l.Suburb, l.Postcode,
85 p.Type, p.Distance, p.Landsize, p.Building_Area, p.YearBuilt, p.Address,
86 s.Price, s.Date, s.Method, s.SellerG
87 FROM tabel_property p
88 JOIN tabel_ruangan r ON p.Rooms_id = r.Rooms_id
89 JOIN tabel_lokasi l ON p.Location_id = l.Location_id
90 JOIN tabel_agen_penjualan s ON p.Property_ID = s.Property_ID
91 WHERE s.Price <= 200000;
92

```

	Property_ID	Rooms_id	no_rooms	Bedrooms	Bathrooms	Car	Location_id	Region_name	Council_Area	Suburb	Postcode	Type	Distance	Landsize
1	1806 R4412		4	4	1	2 SM3162		Southern Metropolitan	Glen Eira	Caulfield	3162 h		8.9	499
2	1928 R4311		4	3	1	1 NM3058		Northern Metropolitan	Moreland	Coburg	3058 h		7.8	536
3	2653 R1110		1	1	1	0 WM3011		Western Metropolitan	Maribyrnong	Footscray	3011 u		6.4	0
4	7117 R1111		1	1	1	1 WM3012		Western Metropolitan	Maribyrnong	Maidstone	3012 u		7.8	0
5	7294 R1111		1	1	1	1 WM3020		Western Metropolitan	Brimbank	Sunshine	3020 u		13.9	2347
6	7304 R1211		1	2	1	1 WM3020		Western Metropolitan	Brimbank	Sunshine	3020 u		13.9	36
7	7306 R2211		2	2	1	1 WM3020		Western Metropolitan	Brimbank	Sunshine	3020 u		13.9	0
8	7941 R1110		1	1	1	0 SM3122		Southern Metropolitan	Boroondara	Hawthorn	3122 u		4.6	322
9	8505 R1111		1	1	1	1 WM3012		Western Metropolitan	Maribyrnong	Maidstone	3012 u		8.2	0
10	8564 R1111		1	1	1	1 WM3020		Western Metropolitan	Brimbank	Sunshine	3020 u		13.9	1175
11	8812 R1110		1	1	1	0 WM3011		Western Metropolitan	Maribyrnong	Footscray	3011 u		5.1	30
12	12667 R1110		1	1	1	0 NM3056		Northern Metropolitan	Moreland	Brunswick	3056 u		5.2	1250

3.2 Analisis

3.2.1 SELECT __ FROM __ JOIN __ GROUP BY __ ORDER BY __

Misal seseorang yang berasal dari Northern Metropolitan ingin melihat harga rata-rata (average price) dari suatu region. Maka kami menggunakan SELECT __ FROM __ JOIN __ GROUP BY __ ORDER BY __ untuk melihat harga rata-rata dari region_name dan location_id. Pada tabel terlihat bahwa Western Victoria memiliki harga rata-rata paling murah dibandingkan region lainnya.

```

55 --Simulasi 1 untuk pemetaan harga rumah berdasarkan lokasi
56 SELECT l.Location_id, l.Postcode, l.Region_name,
57        CAST(AVG(s.Price) AS INTEGER) as Average_Price
58 FROM tabel_property p
59 JOIN tabel_agan_penjualan s ON p.Property_ID = s.Property_ID
60 JOIN tabel_lokasi l ON p.Location_id = l.Location_id
61 GROUP BY l.Location_id, l.Postcode, l.Region_name
62 ORDER BY Average_Price ASC;
63

```

	Location_id	Postcode	Region_name	Average_Price
1	WV3340	3340	Western Victoria	285000
2	NV3756	3756	Northern Victoria	366000
3	WV3338	3338	Western Victoria	384697
4	NV3337	3337	Northern Victoria	406916
5	WV3337	3337	Western Victoria	427208
6	NM3061	3061	Northern Metropolitan	479500
7	WM3335	3335	Western Metropolitan	482750
8	WM3024	3024	Western Metropolitan	483250
9	NM3048	3048	Northern Metropolitan	488095
10	NV3754	3754	Northern Victoria	493833
11	EV3810	3810	Eastern Victoria	496333
12	SE3177	3177	South-Eastern ...	496500
13	NM3750	3750	Northern Metropolitan	501811

Misal seseorang ingin membeli properti melalui agen yang dapat membantu transaksinya lancar. Maka kami menggunakan SELECT__FROM__JOIN__GROUP BY__ORDER BY__ untuk melihat agen teraktif dalam jual properti. Pada tabel terlihat bahwa Nelson merupakan agen teraktif.

```

59 --Simulasi 2 untuk agen properti yang paling aktif
60 SELECT Location_id, SellerG, COUNT(*) as Transaction_Count, SUM(Price) as Total_Sales
61 FROM tabel_property
62 JOIN tabel_agan_penjualan ON tabel_property.Property_ID = tabel_agan_penjualan.Property_ID
63 GROUP BY Location_id, SellerG
64 ORDER BY Transaction_Count DESC, Total_Sales DESC;
65
66

```

	Location_id	SellerG	Transaction_Count	Total_Sales
1	WM3040	Nelson	130	145819500
2	NM3121	Biggin	115	117909000
3	NM3056	Nelson	96	102798000
4	NM3073	Barry	90	65040250
5	WM3020	Douglas	90	58482100
6	EM3084	Miles	89	99148500
7	SM3165	Buxton	88	96291000
8	NM3058	Nelson	85	79050000
9	NM3031	Nelson	82	74527500
10	WM3020	Barry	77	46391400
11	SM3122	Jellis	75	104288000
12	NM3121	Jellis	75	92158010

3.2.2 SELECT ____ FROM ____ JOIN ____ ON ____

Misal seseorang ingin melihat informasi lengkap tentang properti, termasuk lokasi, tipe properti, ukuran, fasilitas, dan informasi harga. Maka, kami menggunakan SELECT ____ FROM ____ JOIN ____ ON ____ untuk menampilkan informasi tersebut.

```
--Simulasi 3 untuk membantu menyimpan informasi lengkap tentang properti, termasuk lokasi, tipe properti, ukuran, fasilitas, dan informasi harga
SELECT p.Property_ID, p.Rooms_id, r.no_rooms, r.Bedrooms, r.Bathrooms, r.Car,
       l.Location_id, l.Region_name, l.Council_Area, l.Suburb, l.Postcode,
       p.Type, p.Distance, p.Landsize, p.Building_Area, p.YearBuilt, p.Address,
       s.Price, s.Date, s.Method, s.SellerG
FROM tabel_property p
JOIN tabel_ruangan r ON p.Rooms_id = r.Rooms_id
JOIN tabel_lokasi l ON p.Location_id = l.Location_id
JOIN tabel_agan_penjualan s ON p.Property_ID = s.Property_ID;
```

Property_ID	Rooms_id	no_rooms	Bedrooms	Bathrooms	Car	Location_id	Region_name	Council_Area	Suburb	Postcode	Type
1 R2211		2	2	1	1	NM3067	Northern Metropolitan Yarra	Abbotsford		3067 h	
2 R2210		2	2	1	0	NM3067	Northern Metropolitan Yarra	Abbotsford		3067 h	
3 R3320		3	3	2	0	NM3067	Northern Metropolitan Yarra	Abbotsford		3067 h	
4 R3321		3	3	2	1	NM3067	Northern Metropolitan Yarra	Abbotsford		3067 h	
5 R4312		4	3	1	2	NM3067	Northern Metropolitan Yarra	Abbotsford		3067 h	
6 R2210		2	2	1	0	NM3067	Northern Metropolitan Yarra	Abbotsford		3067 h	
7 R3420		3	4	2	0	NM3067	Northern Metropolitan Yarra	Abbotsford		3067 h	
8 R2212		2	2	1	2	NM3067	Northern Metropolitan Yarra	Abbotsford		3067 h	
9 R1111		1	1	1	1	NM3067	Northern Metropolitan Yarra	Abbotsford		3067 u	
10 R2312		2	3	1	2	NM3067	Northern Metropolitan Yarra	Abbotsford		3067 h	
11 R2221		2	2	2	1	NM3067	Northern Metropolitan Yarra	Abbotsford		3067 u	
12 R3322		3	3	2	2	NM3067	Northern Metropolitan Yarra	Abbotsford		3067 h	
13 R2221		2	2	2	1	NM3067	Northern Metropolitan Yarra	Abbotsford		3067 u	

Misal seseorang ingin melihat properti yang sesuai dengan kriterianya. Pada kasus ini, orang tersebut ingin melihat jumlah kamar, kamar mandi, kamar tidur, dan tipe properti. Maka kami menggunakan SELECT ____ FROM ____ JOIN ____ ON ____ untuk melihat detail properti.

```
66 --Simulasi 4 untuk membantu penyaringan properti berdasarkan berbagai kriteria
67 SELECT
68     tabel_property.Location_id, tabel_property.Property_ID, tabel_property.Type, tabel_ruangan.no_rooms,
69     tabel_ruangan.Bedrooms, tabel_ruangan.Bathrooms, tabel_ruangan.Car
70 FROM tabel_property
71 JOIN tabel_ruangan
72 ON tabel_property.Rooms_id = tabel_ruangan.Rooms_id;
73
```

	Location_id	Property_ID	Type	no_rooms	Bedrooms	Bathrooms	Car
1	NM3067	1 h		2	2	1	1
2	NM3067	2 h		2	2	1	0
3	NM3067	3 h		3	3	2	0
4	NM3067	4 h		3	3	2	1
5	NM3067	5 h		4	3	1	2
6	NM3067	6 h		2	2	1	0
7	NM3067	7 h		3	4	2	0
8	NM3067	8 h		2	2	1	2
9	NM3067	9 u		1	1	1	1
10	NM3067	10 h		2	3	1	2
11	NM3067	11 u		2	2	2	1
12	NM3067	12 h		3	3	2	2

3.3 Implementasi Interaktif (GUI)

3.3.1 Input transaksi baru

Terdapat transaksi penjualan properti baru yang terjadi di wilayah Melbourne. User ingin memasukkan transaksi penjualan tersebut ke dalam database penjualan yang telah dimiliki.

Sale ID: SP27919

ID Property: 13954

Date (YYYY-MM-DD): 05/06/2024

Method: SP

Price: 1980329

Seller: Nelson

Submit

View Data

Data inserted successfully!

```
('SP27919', '13954', '2024-06-05', 'SP', 1980329.0, 'Nelson')
(None, '13851', '2024-06-05', 'SP', 1782393.0, 'Biggin')
(None, '13851', '2024-06-05', 'SP', 17217.0, 'Biggin')
('SP13580VI', '13580', '26/08/2017', 'SP', 1285000.0, 'Village')
('PI13579SW', '13579', '26/08/2017', 'PI', 2500000.0, 'Sweeney')
('S13578RI', '13578', '26/08/2017', 'S', 1170000.0, 'Raine')
('SP13577WI', '13577', '26/08/2017', 'SP', 1031000.0, 'Williams')
('S13576BA', '13576', '26/08/2017', 'S', 1245000.0, 'Barry')
('S13575ED', '13575', '26/08/2017', 'S', 582000.0, 'Red')
('S13574H0', '13574', '26/08/2017', 'S', 635000.0, 'hockingstuart')
```

3.3.2 Pencarian informasi

Terdapat calon pembeli properti di Melbourne yang ingin mengetahui harga pasar terendah di salah satu wilayah Suburb dan juga rekomendasi agen property mana yang paling banyak melakukan penjualan pada lokasi tersebut. Dengan memanfaatkan database yang telah tersedia, maka didapatkan informasi sebagai berikut:

Area: Abbotsford

Simulate

Harga termurah di area Abbotsford: \$300000.0

Agen dengan penjualan terbanyak: Biggin

Rata-rata harga rumah di Melbourne: \$1075684.08

Rumah termurah di Melbourne: Lokasi: 202/51 Gordon St, Room: 1

3.3.3 Pencarian informasi *property* berdasarkan *budget* user

Terdapat User yang ingin mencari informasi mengenai properti di Melbourne yang sudah dijual dengan pertimbangan budget yang ia miliki dan juga tipe properti yang akan ia cari. User ingin mendapatkan informasi berdasarkan jarak terdekat dengan pusat kota karena ia memiliki tempat kerja di CBD.

Price Range: 2905000 - 710

Keterangan: h: rumah, cottage, villa, semi-terrace; u: unit, duplex; t: townhouse.

Type:

	Property_ID	rooms_id	Rooms	Bedroom2	Bathroom	Car	Type	Distance	Landsize	BuildingArea	YearBuilt	Address	Price	Date	Method	Seller6
0	02531	R4442	4	4.0	4.0	2.0	h	1.6	337.0	291.0	1854.0	52 Nicholson St	3310000.0	13/08/2016	S	Caine
1	02554	R4432	4	4.0	3.0	2.0	h	1.6	286.0	275.0	1890.0	17 Bell St	4011000.0	27/11/2016	S	Nelson
2	04337	R4434	4	4.0	3.0	4.0	h	2.3	310.0	276.0	1880.0	10 Canning St	2920000.0	3/09/2016	S	Nelson
3	05267	R4422	4	4.0	2.0	2.0	h	2.6	319.0	122.0	1893.0	131 Lord St	3335000.0	3/12/2016	S	Jellis
4	05273	R5633	5	5.0	3.0	3.0	h	2.6	618.0	260.0	1890.0	78 Davison St	3200000.0	4/03/2017	SP	Jellis
5	05336	R3322	3	3.0	2.0	2.0	h	2.6	389.0	122.0	1965.0	3 Portland St	3020000.0	14/05/2016	S	Jellis
6	05347	R4231	4	2.0	3.0	1.0	h	2.6	544.0	241.0	1920.0	30 Highett St	3001000.0	16/04/2016	S	Biggin
7	01676	R3312	3	3.0	1.0	2.0	h	3.2	499.0	164.0	1910.0	989 Drummond St	2950000.0	7/05/2016	SP	Chambers
8	00103	R3320	3	3.0	2.0	0.0	h	3.3	147.0	146.0	1890.0	97 Page St	3010000.0	18/03/2017	S	Greg
9	00109	R4421	4	4.0	2.0	1.0	h	3.3	330.0	207.0	1910.0	115 Page St	4735000.0	25/02/2017	S	Marshall

3.4 Testing GUI

3.4.1 User meng-*input* Sale_ID yang sudah ada

User memasukkan Sale_ID yang sama ke saat penginputan. Karena Sale_ID merupakan attribute di tabel yang unique maka error message akan ditampilkan.

Error: Sale_ID 'PI06805JE' already exists. Please use a different Sale_ID.

Sale ID:

ID Property:

Date (DD/MM/YYYY):

Method:

Price:

Seller:

Sebagai solusi, Sale_ID akan digenerate berdasarkan value dari str(Method) + str(Price) + str (Seller:[2]). Selain itu, ditambahkan fitur untuk Seller menjadi fitur dropdown sehingga User tidak perlu melakukan input secara manual untuk value Seller.

Sale ID:

ID Property:

Date (YYYY-MM-DD):

Method:

Price:

Seller:

Data inserted successfully!

	Sale_ID	Property_ID	Date	Method	Price	SellerG
0	NM32829BI	32829	2022-06-07	PI	1280981.0	Nelson
1	NM17878MA	17878	04/06/2024	PI	1280122.0	Nelson
2	PI16788HA	16788	2024-05-21	PI	12928023.0	Harcourts
3	SP06808NE	06808	28/05/2016	SP	771000.0	Nelson
4	S06807HO	06807	28/05/2016	S	570000.0	hockingstuart
5	SP06806JE	06806	28/05/2016	SP	400000.0	Jellis
6	PI06805JE	06805	28/05/2016	PI	480000.0	Jellis
7	PI06804JE	06804	27/11/2016	PI	1325000.0	Jellis
8	S06803JE	06803	27/11/2016	S	1600000.0	Jellis
9	S06802NE	06802	27/06/2016	S	970000.0	Nelson

3.4.2 User meng-input Region yang tidak ada

User memasukkan nama area yang tidak terdapat pada Suburb di kota Melbourne. Sehingga muncul error message berupa ‘Tidak ada data untuk area (area).’

Masukkan area berdasarkan Suburb yang terdapat pada kota Melbourne.

Area:

Tidak ada data untuk area Bekasi.

Agar user tidak dapat memasukkan area name yang tidak sesuai dengan Suburb di Melbourne, sebagai solusinya kami menambahkan fitur dropdown pada widget nya. Sehingga, user hanya dapat mendapatkan informasi dari suburb yang sudah tersedia pada database yang telah kami buat.

Pilih area berdasarkan Suburb yang tersedia di Melbourne.

Area: ▼

Harga termurah di area Abbotsford: \$300000.0

Agen dengan penjualan terbanyak: Biggin

Rata-rata harga rumah di Melbourne: \$1075684.08

Rumah termurah di Melbourne: Lokasi: 202/51 Gordon St, Room: 1

BAB IV PENUTUP

4.1 Kesimpulan

Dari data Melbourne Housing Snapshot, telah dibuat empat tabel relasional, yaitu Properti, Ruangan, Agen Penjualan, dan Lokasi. Kemudian, juga dilakukan simulasi beserta analisisnya, dan dibuat implementasi interaktif (GUI) lalu di-testing. Sehingga project ini berhasil memenuhi tujuannya, yakni:

- Untuk pemetaan harga perumahan,
- Untuk memberikan pengetahuan tentang agen properti yang paling aktif di beberapa area berdasarkan harga jual,
- Untuk membantu menyimpan informasi lengkap tentang properti, termasuk lokasi, tipe properti, ukuran, fasilitas, dan informasi harga, dan
- Membantu mencari dan menyaring properti berdasarkan berbagai kriteria, seperti lokasi, tipe properti, dan fasilitas untuk memudahkan pencarian oleh pengguna.

4.2 Saran

Berdasarkan temuan yang ada, berikut adalah beberapa saran untuk pengembangan lebih lanjut:

1. Perluasan Dataset: Mengumpulkan dan menggunakan dataset yang lebih besar dan beragam untuk meningkatkan akurasi simulasi.
2. Pengembangan Fitur: Menambahkan fitur analisis lanjutan seperti prediksi harga dan tren pasar.
3. Pengembangan GUI : Mengembangkan fitur dan desain GUI agar lebih memudahkan pengguna mencari properti.

Dengan demikian, diharapkan sistem ini dapat terus dikembangkan untuk memberikan manfaat yang lebih besar bagi pengguna dalam mengelola dan menganalisis data penjualan *real estate*.

4.3 Lampiran

Seluruh proses dari pembuatan database ini, diproses melalui file lampiran data ipynb, dan juga file database kami. Dapat diakses melalui link:

[Project Akhir Database - Kelompok 3.](#)