

|  |
| --- |
| Business Template  **Real estate agency** |
|  |

**Contents**

1 Business Description 3

1.1 Business background 3

1.2 Problems. Current Situation 3

1.3 The benefits of implementing a database. Project Vision 3

2 Model description 3

2.1 Definitions & Acronyms 3

2.2 Logical Scheme 3

2.3 Objects 3

* Business Description
* Business background

The real estate industry plays a critical role in the economy, involving property transactions such as buying, selling, and renting. Real estate agencies facilitate these transactions by connecting buyers and sellers or tenants and landlords. A key aspect of these transactions involves agents who manage listings, conduct negotiations, and help close deals. Real estate agencies also track property values, client interactions, agent commissions, and related financial activities.

This system is designed for a real estate agency to manage property listings, clients, agents, transactions, and commissions efficiently. It provides tools for agents and staff to oversee transactions, track financials, and ensure smooth operations in a fast-paced environment.

* Problems. Current Situation

The current system used by the real estate agency is manual and inefficient, with significant challenges in managing property listings, client interactions, and transaction details. Agents and staff members are often overwhelmed by the lack of integration between property data, client profiles, and transaction records. The absence of real-time data tracking leads to delays in decision-making and difficulty in identifying trends.

* the Benefits of implementing a database. Project Vision

Implementing a centralized database will streamline property listings, client management, and transactions, improving efficiency and reducing manual work. Real-time data access will enhance decision-making. The system will also scale easily as the agency grows, providing a reliable and transparent platform to manage operations effectively.

* Model description
* Definitions & Acronyms.
* **Agent**: A licensed professional who facilitates property transactions between buyers and sellers or landlords and tenants.
* **Client**: An individual or entity engaging the services of the agency, either as a buyer, seller
* **Property**: A piece of real estate, including land and any structures on it, that is available for sale
* **Transaction**: A completed sale agreement between a buyer and a seller,
* **Commission**: The payment earned by agents for facilitating a property transaction, typically calculated as a percentage of the sale price.
* Logical Scheme

Provided in git named:  
SQL\_*Cuneyd\_Celik*\_Finaltask\_real\_estate\_agency\_ConceptModel.png , SQL\_Cuneyd*\_Celik*\_Finaltask\_ real\_estate\_agency \_LogicalModel.png

* Objects

**Agent Table**

This table stores information about real estate agents, including their personal details, contact information, and hire date.

|  |  |  |
| --- | --- | --- |
| **Field Name** | **Field Description** | **Data Type** |
| agent\_id | Unique identifier for each agent | SERIAL PRIMARY KEY |
| first\_name | First name of the agent | VARCHAR(100) NOT NULL |
| last\_name | Last name of the agent | VARCHAR(100) NOT NULL |
| email | Email address of the agent | VARCHAR(255) UNIQUE NOT NULL |
| phone\_number | Contact phone number of the agent | VARCHAR(20) |
| hire\_date | Date when the agent was hired | DATE NOT NULL DEFAULT CURRENT\_DATE |

Example:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **agent\_id** | **first\_name** | **last\_name** | **email** | **phone\_number** | **hire\_date** |
| 1 | Ahmet | Yilmaz | [ahmet.yilmaz@example.com](mailto:ahmet.yilmaz@example.com) | 501-123-4567 | 2024-01-15 |
| 2 | Ayse | Kaya | [ayse.kaya@example.com](mailto:ayse.kaya@example.com) | 505-123-4567 | 2023-11-10 |
| 3 | Mehmet | Demir | [mehmet.demir@example.com](mailto:mehmet.demir@example.com) | 506-123-4567 | 2023-07-30 |

**Client Table**

This table holds information about clients, including their names, contact details, and whether they are buyers or sellers.

|  |  |  |
| --- | --- | --- |
| **Field Name** | **Field Description** | **Data Type** |
| client\_id | Unique identifier for each client | SERIAL PRIMARY KEY |
| first\_name | First name of the client | VARCHAR(100) NOT NULL |
| last\_name | Last name of the client | VARCHAR(100) NOT NULL |
| email | Email address of the client | VARCHAR(255) UNIQUE NOT NULL |
| phone\_number | Contact phone number of the client | VARCHAR(20) |
| client\_type | Type of client (buyer or seller) | VARCHAR(10) NOT NULL |

Example:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **client\_id** | **first\_name** | **last\_name** | **email** | **phone\_number** | **client\_type** |
| 1 | Fatma | Yildirim | [fatma.yildirim@example.com](mailto:fatma.yildirim@example.com) | 507-123-4567 | buyer |
| 2 | Emre | Tekin | [emre.tekin@example.com](mailto:emre.tekin@example.com) | 508-123-4567 | seller |
| 3 | Elif | Demir | [elif.demir@example.com](mailto:elif.demir@example.com) | 509-123-4567 | buyer |

**Property Table :** This table contains details about properties, including address, size, price, type, and status. It also calculates the price per square foot.

|  |  |  |
| --- | --- | --- |
| **Field Name** | **Field Description** | **Data Type** |
| property\_id | Unique identifier for each property | SERIAL PRIMARY KEY |
| address | Address of the property | TEXT NOT NULL |
| city | City where the property is located | VARCHAR(100) NOT NULL |
| state | State where the property is located | VARCHAR(50) NOT NULL |
| postal\_code | Postal code of the property's location | VARCHAR(20) |
| property\_type | Type of property (e.g., house, apartment) | VARCHAR(50) NOT NULL |
| bedrooms | Number of bedrooms in the property | INTEGER DEFAULT 0 |
| bathrooms | Number of bathrooms in the property | INTEGER DEFAULT 0 |
| square\_feet | Square footage of the property | INTEGER DEFAULT 0 |
| listing\_price | Listing price of the property | NUMERIC(15, 2) NOT NULL |
| listing\_agent\_id | ID of the agent listing the property | INTEGER |
| status | Current status of the property (listed, sold, etc.) | VARCHAR(20) DEFAULT 'listed' |
| listed\_date | Date when the property was listed | DATE NOT NULL |
| price\_per\_sqft | Computed price per square foot | NUMERIC(15, 2) GENERATED ALWAYS AS (listing\_price / NULLIF(square\_feet, 0)) STORED |

Example:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **property\_id** | **address** | **city** | **state** | **postal\_code** |
| 1 | 123 Main St | Istanbul | Istanbul | 34000 |
| 2 | 456 Elm St | Ankara | Ankara | 06000 |
| 3 | 789 Oak St | Izmir | Izmir | 35000 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **property\_type** | **bedrooms** | **bathrooms** | **square\_feet** | **listing\_price** |
| apartment | 3 | 2 | 1200 | 500000 |
| house | 4 | 3 | 2000 | 850000 |
| villa | 5 | 4 | 3000 | 1500000 |

|  |  |  |
| --- | --- | --- |
| **listing\_agent\_id** | **status** | **listed\_date** |
| 1 | listed | 2024-01-10 |
| 2 | pending | 2023-11-25 |
| 3 | sold | 2023-10-05 |

**Transaction Table**

**This table records the details of real estate transactions, including the property involved, buyer, seller, agent, sale price, and transaction date.**

|  |  |  |
| --- | --- | --- |
| **Field Name** | **Field Description** | **Data Type** |
| transaction\_id | Unique identifier for each transaction | SERIAL PRIMARY KEY |
| property\_id | Property involved in the transaction | INTEGER NOT NULL |
| buyer\_id | ID of the buyer involved in the transaction | INTEGER NOT NULL |
| seller\_id | ID of the seller involved in the transaction | INTEGER NOT NULL |
| closing\_agent\_id | ID of the agent closing the transaction | INTEGER NOT NULL |
| sale\_price | Sale price of the property | NUMERIC(15, 2) NOT NULL |
| transaction\_date | Date when the transaction was completed | DATE NOT NULL DEFAULT CURRENT\_DATE |

Example:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **transaction\_id** | **property\_id** | **buyer\_id** | **seller\_id** | **closing\_agent\_id** | **sale\_price** | **transaction\_date** |
| 1 | 1 | 1 | 2 | 1 | 500000 | 2024-01-15 |
| 2 | 2 | 3 | 2 | 2 | 850000 | 2023-11-30 |
| 3 | 3 | 1 | 3 | 3 | 1500000 | 2023-10-10 |

**Commission Table**

**This table tracks the commission payments made to agents based on completed transactions, including commission amounts and paid dates.**

|  |  |  |
| --- | --- | --- |
| **Field Name** | **Field Description** | **Data Type** |
| commission\_id | Unique identifier for each commission | SERIAL PRIMARY KEY |
| transaction\_id | ID of the related transaction | INTEGER |
| agent\_id | ID of the agent receiving the commission | INTEGER |
| commission\_amount | Amount of the commission | NUMERIC(15, 2) |
| paid\_date | Date when the commission was paid | DATE NOT NULL DEFAULT CURRENT\_DATE |

Example:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **commission\_id** | **transaction\_id** | **agent\_id** | **commission\_amount** | **paid\_date** |
| 1 | 1 | 1 | 25000 | 2024-01-20 |
| 2 | 2 | 2 | 42500 | 2023-12-05 |
| 3 | 3 | 3 | 75000 | 2023-10-15 |

**Client\_Property\_View Table**

**This table captures which clients have viewed which properties, along with the date of the view, acting as a bridge between clients and properties.**

|  |  |  |
| --- | --- | --- |
| **Field Name** | **Field Description** | **Data Type** |
| client\_id | ID of the client who viewed the property | INTEGER NOT NULL |
| property\_id | ID of the property viewed by the client | INTEGER NOT NULL |
| view\_date | Date when the property was viewed | DATE NOT NULL DEFAULT CURRENT\_DATE |

Example:

|  |  |  |
| --- | --- | --- |
| **client\_id** | **property\_id** | **view\_date** |
| 1 | 1 | 2024-01-12 |
| 2 | 2 | 2023-11-28 |
| 3 | 3 | 2023-10-02 |

Relations:

**Agent to Property (1:M)**

One agent can list multiple properties, but each property is listed by one agent.

**Agent to Transaction (1:M)**

One agent can close multiple transactions, but each transaction has one closing agent.

**Agent to Commission (1:M)**

One agent can earn multiple commissions, but each commission belongs to one agent.

**Property to Client\_Property\_View (1:M)**

One property can be viewed by many clients, but each view record belongs to one property.

**Property to Transaction (1:1)**

Each property can be sold only once, linked to a single transaction.

**Transaction Buyer\_ID to Client (M:1)**

Many transactions can have the same buyer client, but each transaction has one specific buyer.

**Transaction Seller\_ID to Client (M:1)**

Many transactions can have the same seller client, but each transaction has one specific seller.

**Transaction to Commission (1:1)**

Each transaction results in exactly one commission record.

**Client to Client\_Property\_View (1:M)**

One client can view multiple properties, but each view record belongs to one client.