

## Epic: Real Estate Sales Management

### Story: Customer and Real Estate Agent Management

As an admin (real estate agent), I want to be able to manage customer and real estate agent information, so that I can effectively handle property sales.

#### Acceptance Criteria:

The system provides an administrative interface for the real estate agent to create and manage customer and real estate agent accounts.

The admin can add new customers and real estate agents, capturing relevant information such as name, contact details, and account balance.

The system validates the input data for required fields and data formats.

The admin can update customer and real estate agent details, including contact information or account balances.

The system allows the admin to delete customer and real estate agent accounts, ensuring data integrity.

Each customer and real estate agent has a unique primary key associated with their account for identification purposes.

### Story: Customer Eligibility and Price Point

As a customer, I want to be able to determine my eligibility for purchasing a property and find properties within my price range.

#### Acceptance Criteria:

The system provides a user-friendly interface for customers to input their financial information, such as income, credit score, and available down payment.

The system performs eligibility calculations based on predefined criteria (e.g., debt-to-income ratio) and determines whether the customer is eligible for a mortgage.

Customers can search for properties within their price range, specifying their desired location, property type, and maximum price.

The system filters property listings based on the customer's criteria and displays relevant properties.

The system validates the customer's input to ensure the accuracy of the search results.

Story: Real Estate Data Management

As an admin (real estate agent), I want to be able to manage real estate property data, including listing, updating, and removing listings.

Acceptance Criteria:

The system allows the admin to enter new real estate property information, including details such as property type, location, price, and features.

The admin can upload photos and provide comprehensive descriptions for each property listing.

The system validates the input data for required fields and appropriate data formats.

The admin can update property information, such as adjusting the listing price, modifying property features, or updating listing dates.

The system provides the option to remove property listings that are no longer available or sold.

Each real estate property has a unique primary key for identification within the system.

ER (Entity-Relationship) diagram based on the table structure you provided, including the 1:1 and 1:M relationships:

+-----+

| Customer |

+-----+

| customer\_id (PK) |

| name |

| contact\_details |

| account\_balance |

+-----+

|

|

1

|

|

+-----+

| RealEstateAgent |

+-----+

| agent\_id (PK) |

| name |

| contact\_details |

+-----+

|

|

1

|

|

+-----+

RealEstate
property_id (PK)
title
location
price
listing_date
sell_date
customer_id (FK)
agent_id (FK)

In this diagram:

The Customer table has a one-to-many relationship with the RealEstate table, indicated by the customer\_id foreign key in the RealEstate table referencing the customer\_id primary key in the Customer table.

The RealEstateAgent table also has a one-to-many relationship with the RealEstate table, indicated by the agent\_id foreign key in the RealEstate table referencing the agent\_id primary key in the RealEstateAgent table.

Both relationships represent a one-to-many relationship because each customer and real estate agent can have multiple real estate properties, but each real estate property can only belong to one customer and one real estate agent.