

# Project Report

## Aim

To create an application to manage the functions of a Real Estate Agency

## Description

The main idea of the application was to make an application that had functions facilitating the functions of a Real Estate Agency, namely, the transfer of real estate via Selling/Renting of houses.

The application allows users to do two things

1. Buy or rent a house from the database
2. Sell or put up a house for rent on the database

### **1. Buy / Rent a house**

If the user chooses to buy/rent a house, they are prompted to enter the city they want to search in. Then, they are asked whether they want to buy or rent houses.

Then, the user is displayed the list of all houses in the city of the type they asked for, arranged in order of maximum value-for-money. The user can then choose a house to buy/rent and the house is sold. This is updated in the database, and the difference will reflect in the upcoming purchases.

### **2. Sell / Put up a house for rent**

If the user chooses to sell or put up a house for rent, they are prompted to enter various details of the house, including its size, price and location particulars. The city of the house is used to determine the place where it is stored in the database.

All houses in the same city are grouped together and are easily accessible due to this. After the user has entered the particulars of the houses into the form, their house will be displayed to buyers who are interested in the type of contract (to buy or rent) and in the city location of the house.

## Data Structures used

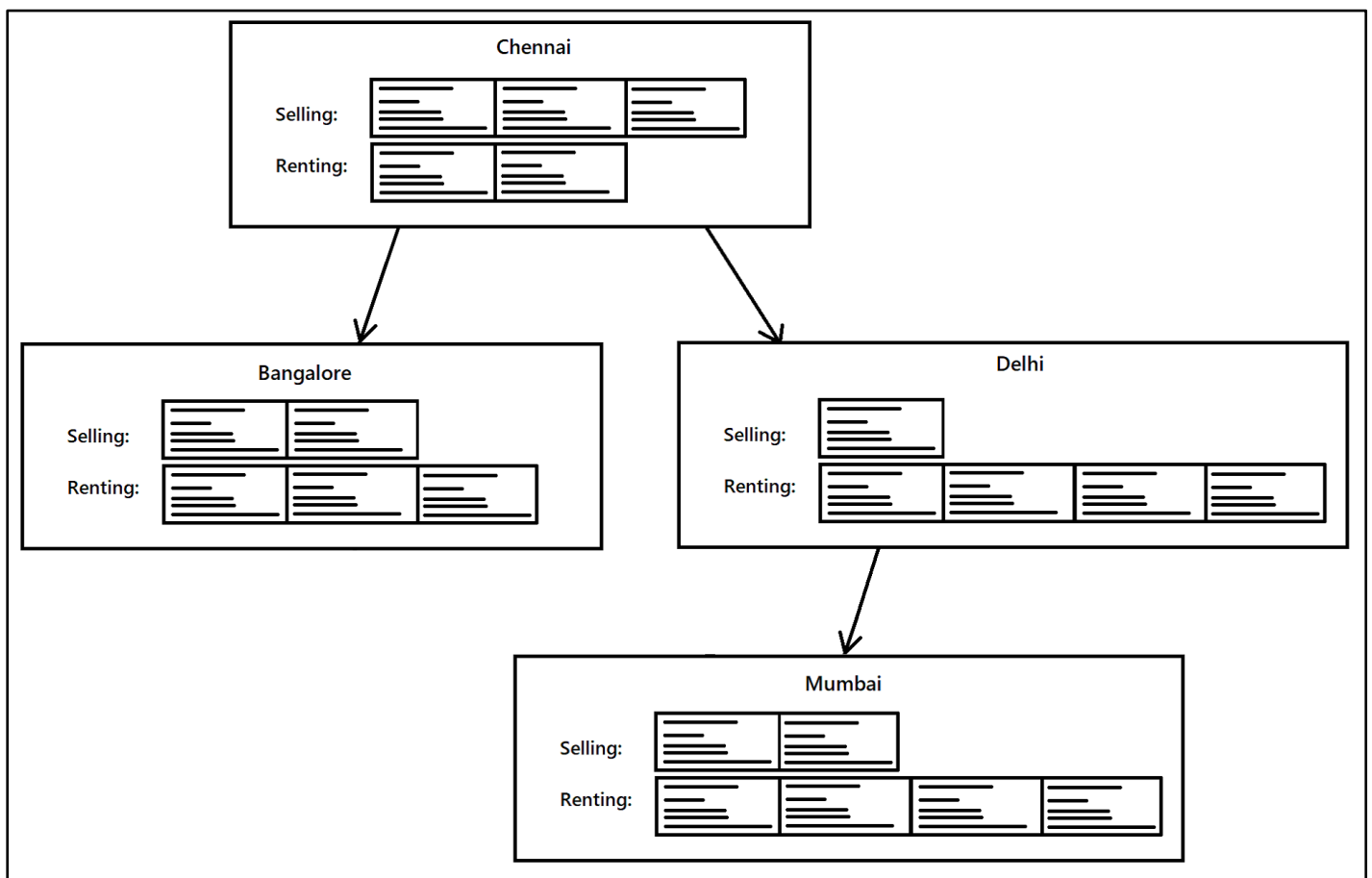
- We used two data structures to achieve the storage of this data
  1. Priority Queue (To store individual house details)
  2. AVL Tree (To store lists of houses in each city)
- The Priority Queue consists of house details, which are
  1. House name
  2. City location
  3. Address of the house
  4. Type of house (Selling or Renting)
  5. Price or Rent per month
  6. Size in square feet
  7. Number of bedrooms
  8. State of furnishing
  9. Availability (Used by the Admin)
  10. Priority (Used by the Admin)
- The priority for each house is calculated as such:

$$Priority = \frac{(Number\ of\ Bedrooms)(State\ of\ Furnishing)(Size\ in\ sq.\ ft.)}{Price}$$

- This is the cost-effectiveness of the house. Greater the priority value, higher is its cost-effectiveness
- Thus, when the priority queue is accessed to display to buyers, the most cost-effective houses will be displayed first, as they will be of most value to the customers.
- Each Node of the AVL tree holds 3 data members
  1. Name of the city
  2. Priority queue of houses to sell
  3. Priority queue of houses to rent
- The main advantage of using an AVL tree is ease of access
- Since most buyers will be looking for houses within a specific city, storing the city-specific lists in an AVL tree will reduce the time taken for accessing and displaying the houses considerably faster.
- The priority queue containing houses of the type (selling or renting) in the city of the customer's request will be the only one that will be traversed, and modified, in case the customer buys the house.

# Design of the Application

The design is, as explained above, an AVL tree where each node is a packet of the City name, the priority queue of houses to be sold, and a priority queue of houses put up on rent. The AVL tree is created based on the alphabetical precedence of the city names.



*Schematic representation of the tree with house details inside their respective queues*

# Learning Experience

1. This application uses two data structures, **Priority Queue** and **AVL tree**. The priority queue is basically a list of any kind of data, but **ordered on the basis of priority given to each element**. The data is organized into priority queues if there has to be a kind of precedence of storage, where the priority given to each element matters in their positioning in the queue. The AVL tree is a data structure in which stored data is in a binary tree format where each node has two or less subtrees. **The left subtree of each node always has lesser precedence than the node itself, while the right subtree has a higher precedence**. This allows the structure to be very handy in terms of time to search any node. **The complexity involved in the search operation is considerably low**.
2. The data organized into this combined data structure is that of houses. The members of the house details are:
  - House name
  - City location
  - Address of the house
  - Type of house (Selling or Renting)
  - Price or Rent per month
  - Size in square feet
  - Number of bedrooms
  - State of furnishing
  - Availability
  - Priority
3. The main operations that are performed on the data stored are **Selling and Buying of houses**. Each of those operations involve several basic operations, namely, **Insertion, Deletion, Searching, Displaying, and Modifying**.
4. Learnt how to design the ADTs for Priority queues and AVL Trees, with any data as the elements. Learnt how to deal with the complications that may arise due to the usage of these new datatypes as elements, and also how to fix them.
5. Learnt how to design an application which can utilize these basic functions offered by the ADTs, to make what we want. Learnt to make the Application to have a simple UI, which is fool-proofed to reduce crashes due to incorrect inputs from the user.

# Screenshots



----- Main Menu -----



----- Selling a house -----

```
Enter the city you want to search in: Chennai
Enter your choice:
1. Buy a house
2. Rent a house
>>2
```

```
-----
| 1 |
-----
```

```
-----
|           The Upper Crest           |
-----
```

```
Semi furnished 1650 sq. feet 3 BHK house
For RENT at Rs. 35000 per month
```

```
-----
Full Address:
128/56, The Upper Crest
Arcot Road, Valasaravakkam
-----
```

```
-----
| 2 |
-----
```

```
-----
|           The Grand Residence       |
-----
```

```
Unfurnished 1800 sq. feet 4 BHK house
For RENT at Rs. 40000 per month
```

```
-----
Full Address:
58/127, The Grand Residence
ECR, Navalur
-----
```

```
Are you interested in buying any of these houses? (Y/N): Y
Enter the number of the house you want to purchase or rent: 3
Please enter a valid choice!
Enter the number of the house you want to purchase or rent: 2
```

```
-----
| Successfully purchased! |
-----
```

```
Press ENTER to go back to Main Menu...
```

## ----- Renting a house -----

```
Enter the city you want to search in: Chennai
Enter your choice:
1. Buy a house
2. Rent a house
>>1
```

```
-----
| 1 |
-----
```

```
-----
|           The Central Park Villas   |
-----
```

```
Semi furnished 1750 sq. feet 3 BHK house
For SALE at Rs. 4500000
```

```
-----
Full Address:
12/7 The Central Park Villas
OMR, Sholinganallur
-----
```

```
Are you interested in buying any of these houses? (Y/N): N
Would you like to search in another city? (Y/N): Y
```

## ----- Buying a house -----