

# FINDBRICK

## Smart Real Estate



Your Gateway  
to Smarter  
Property  
Management  
and Discovery



Developed by

**Rumit Solanki, Khushali Mayatra**

# **Saurashtra College of Management**

## **& Computer Science –Atkot**

Report On

**FindBrick Smart Real Estate**

By

**Rumit Solanki & Khushali Mayatra**

Under Guidance

Of

**Prof. Varsha Dabhi**

Submitted to

Department Of Computer Science Shree Saurastra College Of  
Management & Computer Science.

BACHELOR OF COMPUTER APPLICATION (BCA)

IN

5Th SEMESTER

Year of Submission

2025-26

Saurashtra University Rajkot –360001

## Index Of Project

NO	Description	Page No.
1	Declaration	1
2	Acknowledgement	2
3	Preface	3
4	Project Profile	4
5	Feasibility	5
6	Introduction of My website	6
7	Objective	7
8	Functionality	8
9	SDLC	9
10	SDLC Diagram	10
11	Project Time Line	11
12	Admin Activity Diagram	12

13	User Activity Diagram	13
14	DFD Diagram	14
15	Admin Use-Case Diagram	29
16	Users Use-Case Diagram	30
17	ER-Diagram	31
18	Class Diagram	32
19	Data Dictionary	33
20	Client-Side Snapshot, Code Explanation	43
21	Admin Side Snapshot, Code Explanation	77
22	Testing	91
23	Conclusion	96
24	Future Enhancements	97
25	Bibliography	98

## DECLARATION

We, the students of **B.C.A. Semester-5**, declare that the project work written in this report is our own work. This project has been completed under the guidance of all professors of the BCA Department, **Shree College of Management & Computer Science, Atkot (Saurashtra University)**.

The main aim of doing this project is to gain practical knowledge and improve our skills in a specific subject area. This project is also a part of our BCA course study.

We confirm that this report is based on our original work and research done under the guidance of our **project guide, Prof. Varsha Dabhi**. All the information, data, and references used in this report have been properly mentioned and acknowledged.

This project has not been submitted to any other college or university for any purpose. If there are any similarities with other works, they are purely by chance.

We take full responsibility for the contents and information written in this report. If there are any questions about this project, we are ready to give more details or proof of our work.

**Thank You,**

Rumit Solanki & Khushali Mayatra

## ACKNOWLEDGMENT

We, the students of **B.C.A.** studying at **Shree Saurashtra College of Management & Computer Science, Atkot**, would like to express our sincere gratitude to everyone who supported us in completing this project successfully.

First of all, we would like to thank our **project guide, Prof. Varsha Dabhi**, for her constant support, expert guidance, and valuable suggestions throughout the project. Her mentorship played an important role in the development and success of our project.

We are also thankful to all the **faculty members of Shree Saurashtra College of Management & Computer Science** for providing us with a helpful learning environment and all the resources needed for our project.

We express our heartfelt thanks to our **Head of Department** for encouraging us to take up this project work. We also thank all the **teaching and non-teaching staff** of the college for their cooperation and assistance.

This project would not have been possible without the **collective support and contribution** of these individuals and the institution. We are truly grateful for their involvement and encouragement.

## PREFACE

This project is based on the growing need for a **smart and convenient way to find real estate properties** online. The real estate industry is one of the most important sectors in today's world, and technology has made it easier for people to **search, compare, and buy or sell properties** without wasting time and effort.

This **project report** has been prepared as part of our **B.C.A. 5th Semester** coursework at **Shree Saurashtra College of Management & Computer Science, Atkot**, affiliated with **Saurashtra University, Rajkot**.

It gives us great pleasure and pride to present our project titled "**FindBrick – Smart Real Estate**." This project represents the result of our continuous learning, teamwork, and practical application of knowledge gained during our academic journey.

With the fast growth of **online services and digital platforms**, people now expect easier ways to handle real estate transactions. The **FindBrick Web Site** is developed with the aim of providing a **user-friendly, secure, and efficient platform** for users to **find, list, and manage properties** from anywhere at any time.

As students of **Computer Science and Management**, this project gave us a valuable opportunity to **apply our theoretical knowledge to a real-world situation**. We worked on various aspects of **system analysis, database design, web development, and project management** while focusing on creating a useful and practical solution for users.

We would like to express our sincere gratitude to our **project guide, Prof. Varsha Dabhi**, for her continuous support, guidance, and encouragement throughout the project. Her valuable advice helped us in every stage of development and made this project possible.

## PROJECT PROFILE

Developed At	SHREE SAURASHTRA COLLEGE OF MGT &COM.SCE – ATKOT
Developed By	Rumit Solanki & Khushali Mayatra
Project Title:	FindBrick smart Real Estate
Operating System:	Microsoft windows 10 or higher version
Web Server:	Xampp
Web Browser:	Any compatible browser (Crome or Edge)
Editor:	Visual Studio Code
Back End:	MySQLI
Languages:	PHP
Project Guide:	Prof. Varsha Dabhi
Submitted To:	SHREE SAURASHTRA COLLEGE OF MGT &COM.SCE – ATKOT.

## FEASIBILITY

Hardware	Features	Cost
Processor	Intel i3-13100F	12,999
SSD	1tb	7,099
RAM	8gb	3,549
O.S.	Windows 11	6,000
	<b>Total:</b>	<b>29,647</b>

## INTRODUCTION OF MY PROJECT

This project is based on the real estate field named “**FindBrick – Smart Real Estate**”.

The main aim of this project is to make the process of buying, selling, and renting properties easier and faster. The system helps users to find the right property according to their needs and budget.

The project has two main panels:

- 1. Admin Panel** – The admin can manage users, properties, agents, feedback. Admin can also verify and approve property listings.
- 2. User Panel** – The user can create an account, properties, their own properties, contact agents, and send feedback.

This system helps in maintaining all property details, user information, and agent records in one place. It reduces manual work and saves time. The project also includes features like property search, filtering, image display, and secure login and registration with OTP verification.

The project is developed using **PHP, MySQL, HTML, CSS, Bootstrap, and JavaScript**. The main goal of this project is to provide a simple, secure, and smart web-based solution for property management and real estate activities.

Our mission with **FindBrick** is to make property management digital, easy to use, and helpful for both users and administrators.

## OBJECTIVE

The main objective of this project is to manage all real estate-related information in a systematic and organized way. This system helps users to easily buy, sell, or rent properties online.

It allows the admin to add, update, or delete property details, manage user accounts, and verify property listings. Users can create their own accounts, view available properties, contact agents, and post their own properties for sale or rent.

The system aims to save time, reduce manual work, and provide accurate property information to all users. It also helps improve communication between buyers, sellers, and real estate agents.

Overall, this system is designed to be a one-stop solution for managing property-related activities quickly, safely, and efficiently.

## FUNCTIONALITY

### User Side Functionality (Client Side): -

- **Login / Logout:** Users can securely log in and log out of their accounts.
- **Registration:** New users can create an account by entering basic details.
- **Forgot Password:** Users can reset their password using email verification (OTP).
- **View Properties:** Users can view available properties with full details like price, location, images, and property type.
- **Search and Filter:** Users can search properties by location, price range, and category (rent/sale).
- **Add Property:** Registered users can post their own properties for sale or rent.
- **Edit / Delete Property:** Users can edit or delete their own property listings.
- **Contact Agent:** Users can send inquiries or contact agents directly.
- **View Gallery:** Users can see property images and gallery.
- **Feedback:** Users can send feedback or suggestions to the admin.

### Admin Side Functionality: -

- **Login / Logout:** Admin can securely log in and log out of the admin panel.
- **Manage Users:** Admin can view, edit, or delete user accounts.
- **Manage Agents:** Admin can add, edit, or remove agent details.
- **Verify Properties:** Admin can approve or reject property listings before showing them publicly.
- **Manage State or City:** Admin can add or delete State or City.
- **Manage Properties:** Admin can update or delete property details.
- **Manage Categories:** Admin can create or edit property categories like "House," "Flat," "Land," etc.
- **Manage Feedback:** Admin can view and reply to user feedback.
- **Manage About / Contact Pages:** Admin can edit About and Contact information shown on the website.
- **Send Notices / Announcements:** Admin can post notices or announcements for users and agents.

## SYSTEM DEVELOPMENT LIFE CYCLE (SDLC)

The **System Development Life Cycle (SDLC)** is a step-by-step process used to plan, design, develop, test, and implement an information system successfully. It helps developers and users work together to build an efficient and reliable system.

This method includes the following six main steps:

### 1. Requirement Gathering

In this stage, all the requirements of the system are collected. The main goal is to understand what users and admin need in the project. For example, in **FindBrick**, we gathered requirements for property listings, user registration, search filters, and admin management.

### 2. System Analysis

Here, the collected requirements are analysed carefully. The system's functions, data flow, and modules are studied. We identify what is possible, what tools are needed, and how the project will be developed.

### 3. System Design

In this phase, the structure and layout of the system are planned. This includes database design, user interface design, and page navigation. For **FindBrick**, wireframes were created for the home page, property listing, and admin dashboard.

### 4. Coding

This is the development stage. The actual coding is done using programming languages and tools. In our project, we used **PHP, MySQL, HTML, CSS, Bootstrap, and JavaScript** to build both the client and admin panels.

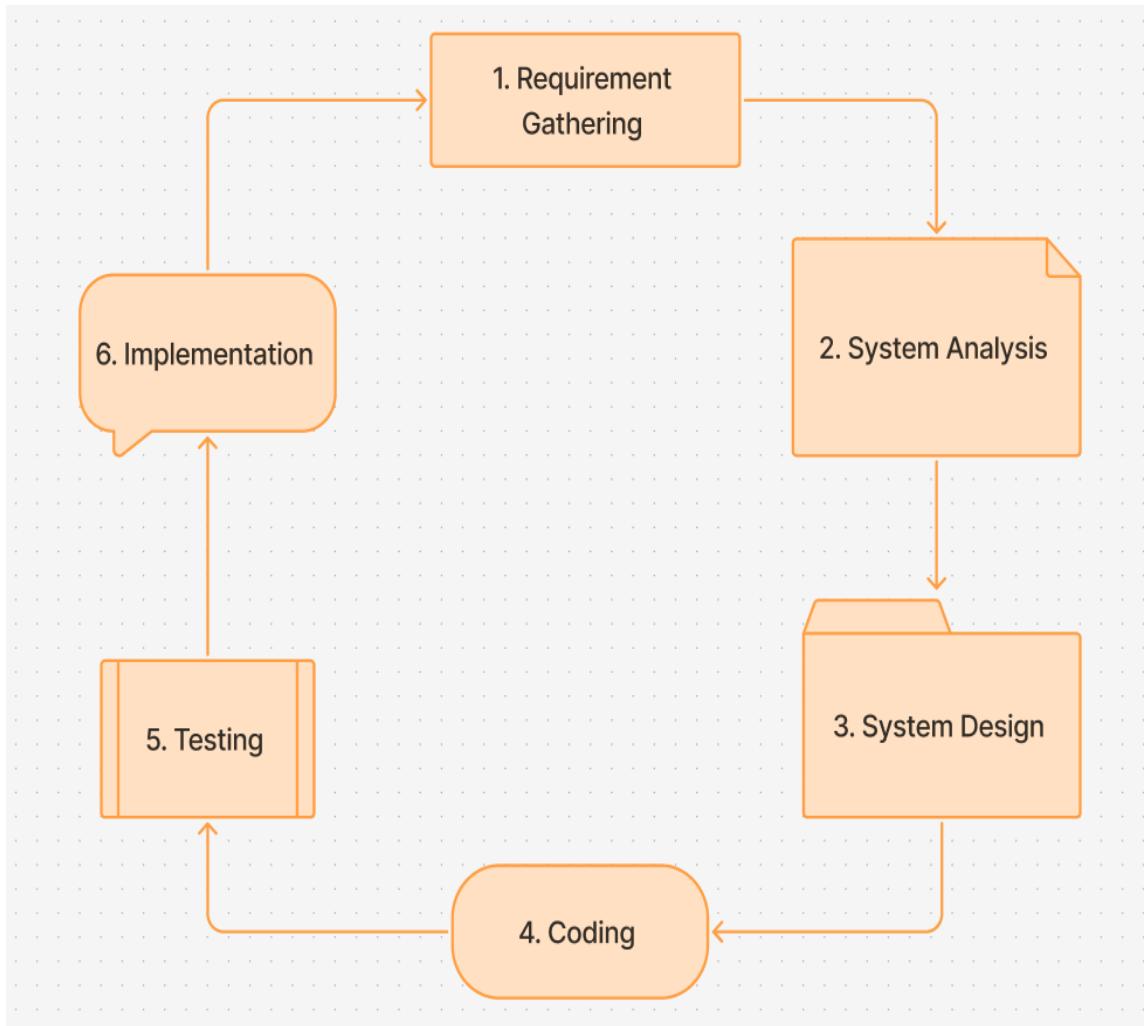
### 5. Testing

After coding, the system is tested to check for any errors or bugs. Each module (like login, registration, and property listing) is tested to make sure it works properly and gives accurate output.

### 6. Implementation

In this stage, the tested system is installed and made ready for real use. The database and files are uploaded to a local or online server so users and admin can use it easily.

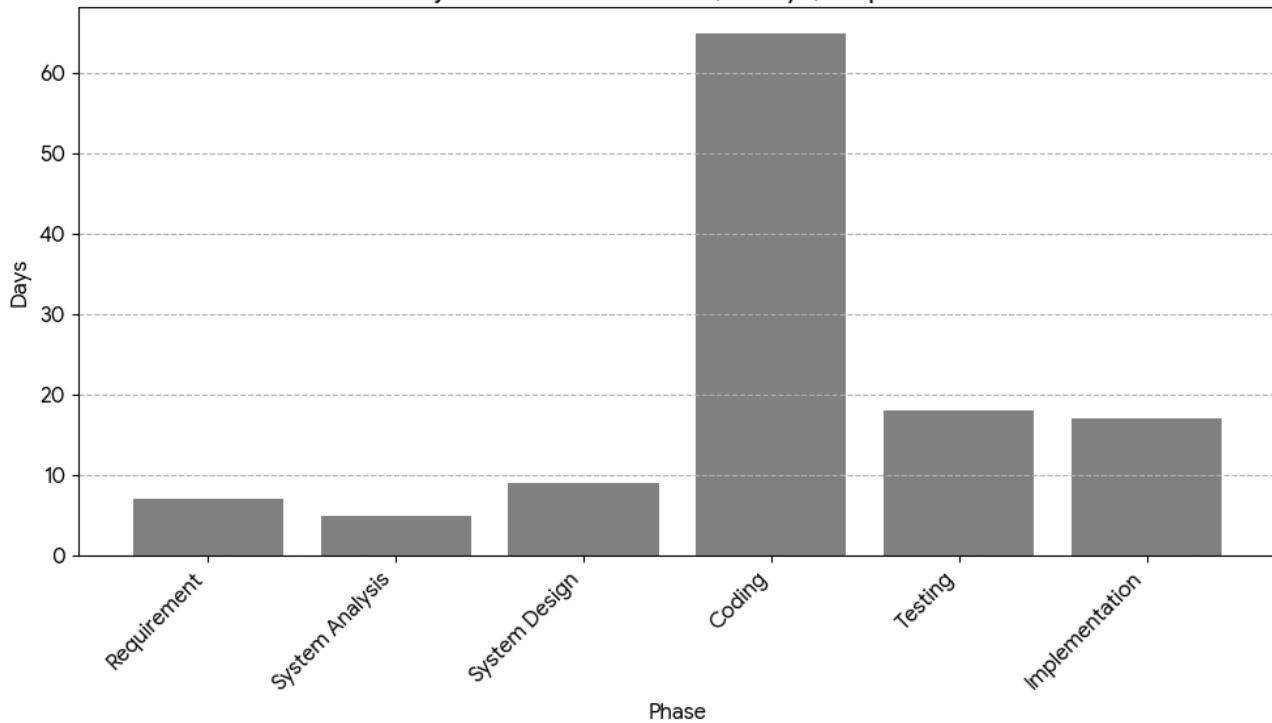
## (SDLC) DIAGRAM



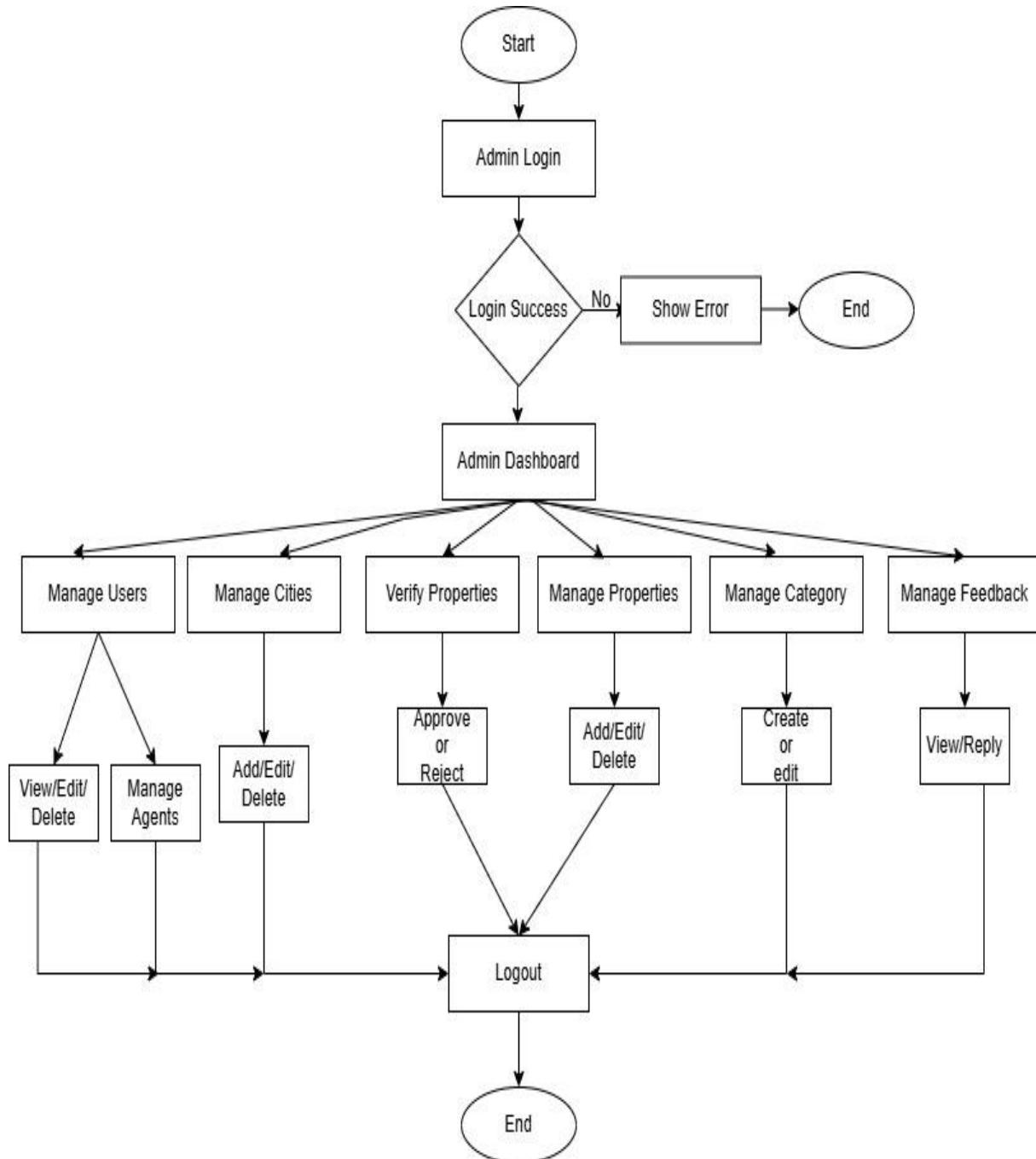
## PROJECT TIMELINE

SR.NO.	WORK NAME	DATE	TOTAL DAYS
1	Requirement Gathering	17/06/2025	07
2	System Analysis	26/06/2025	05
3	System Design	01/07/2025	09
4	Coding	10/07/2025	63
5	Testing	11/09/2025	18
6	Implementation	29/09/2025	17

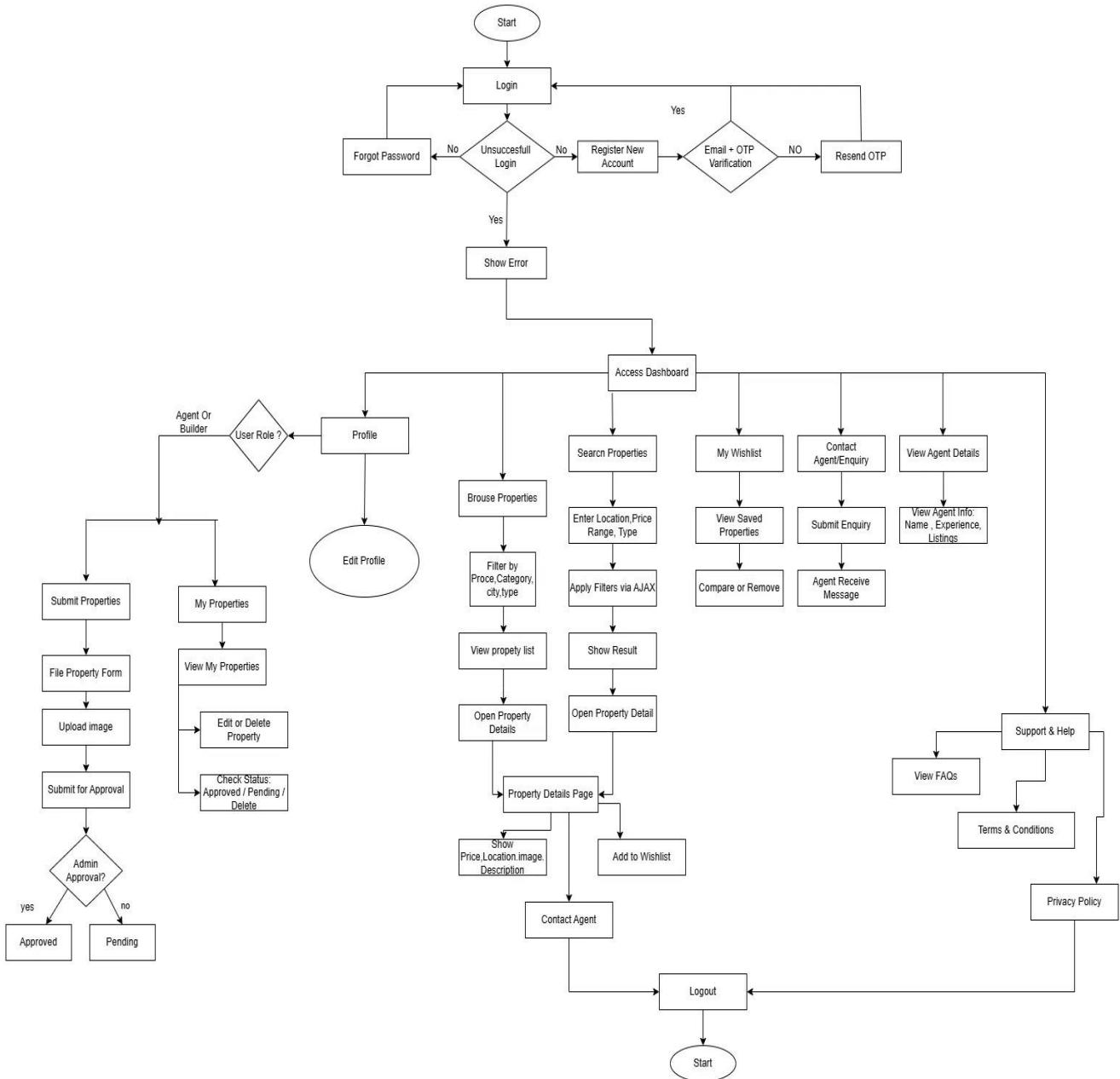
Project Phase Durations (in Days) - Updated



## ADMIN ACTIVITY DIAGRAM:

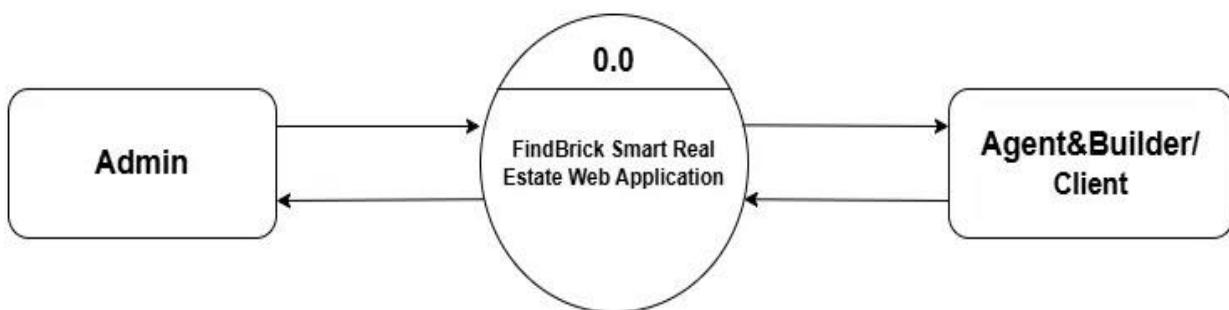


## USER ACTIVITY DIAGRAM:

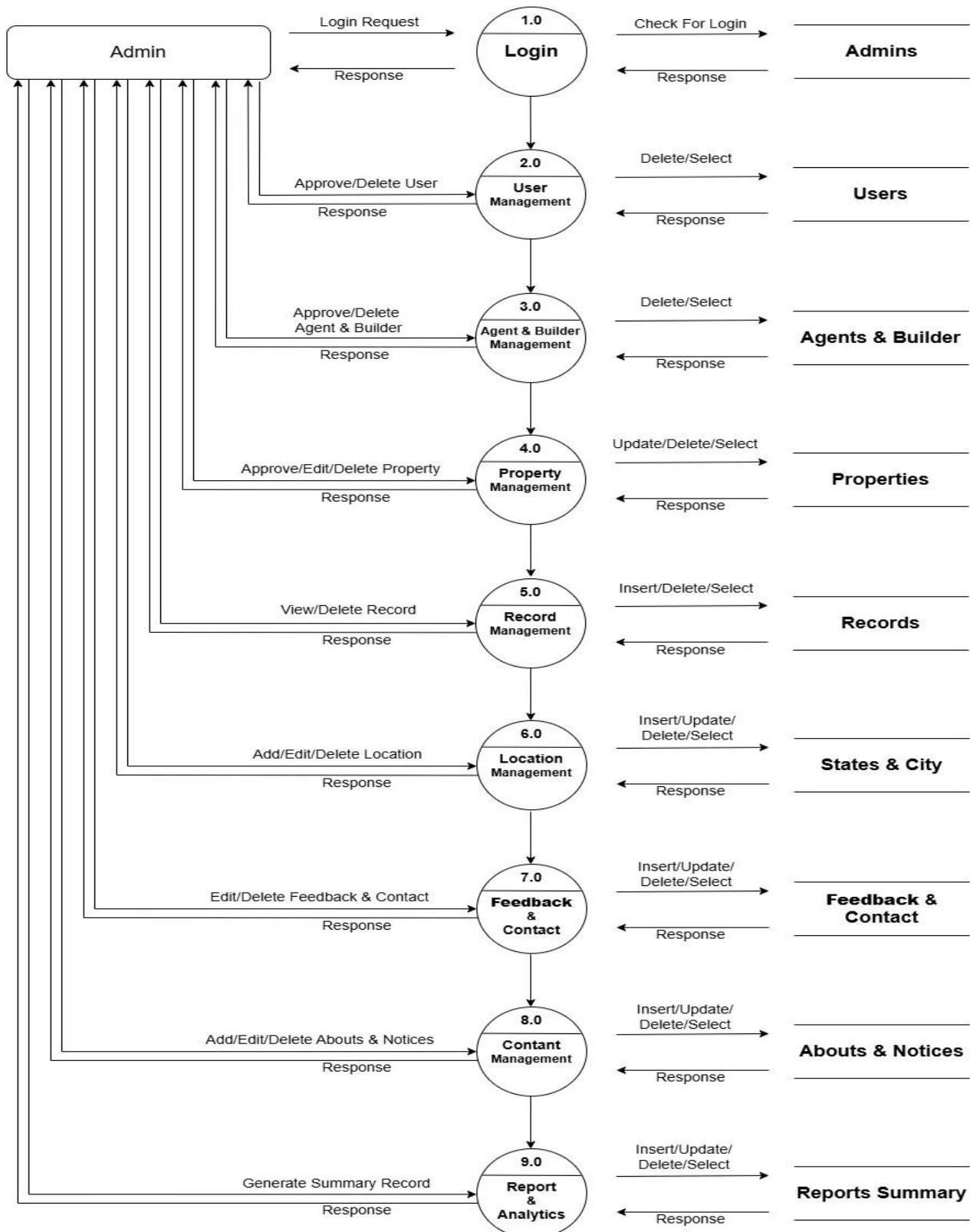


## DATA FLOW DIAGRAM

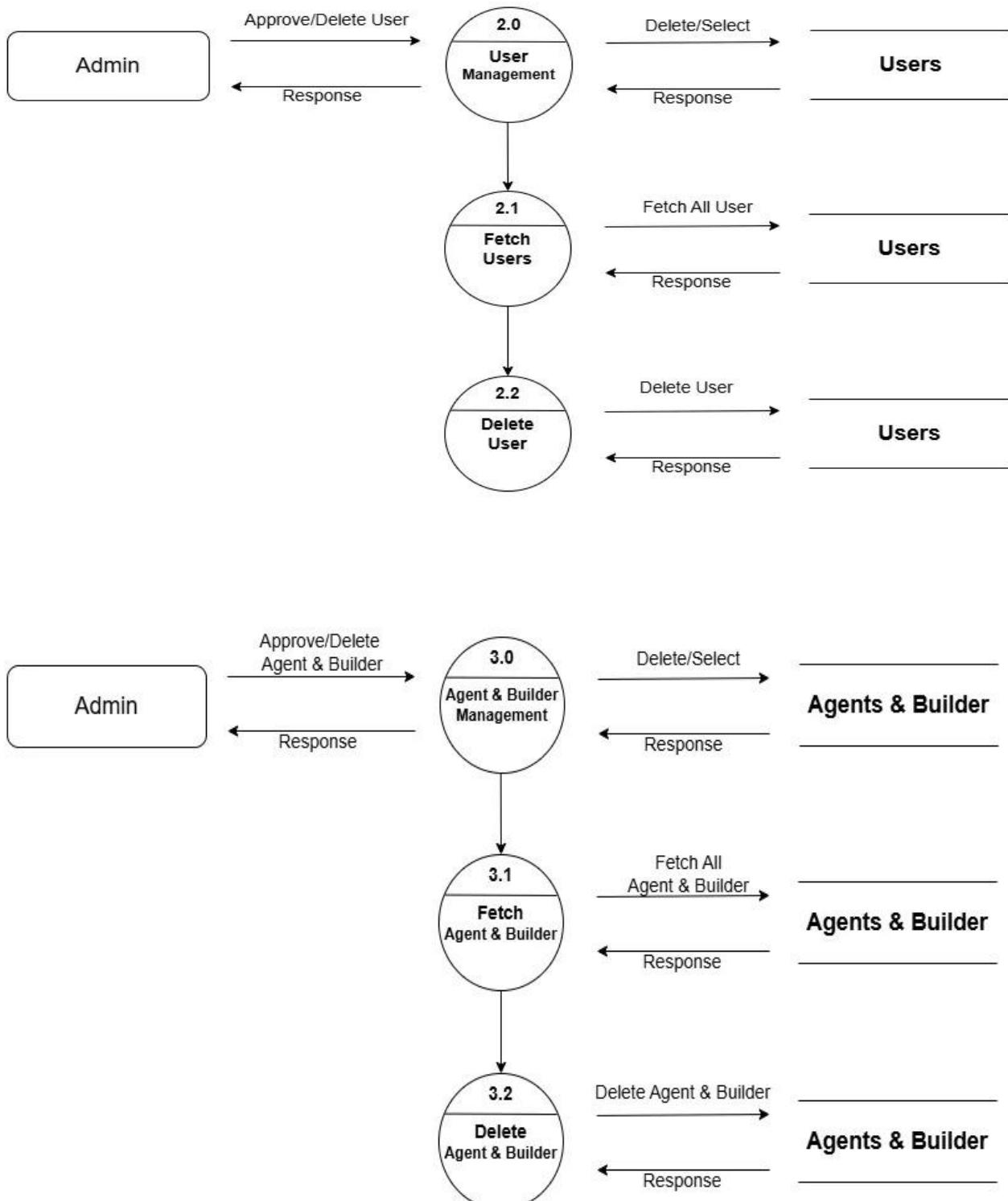
0 level DFD: -

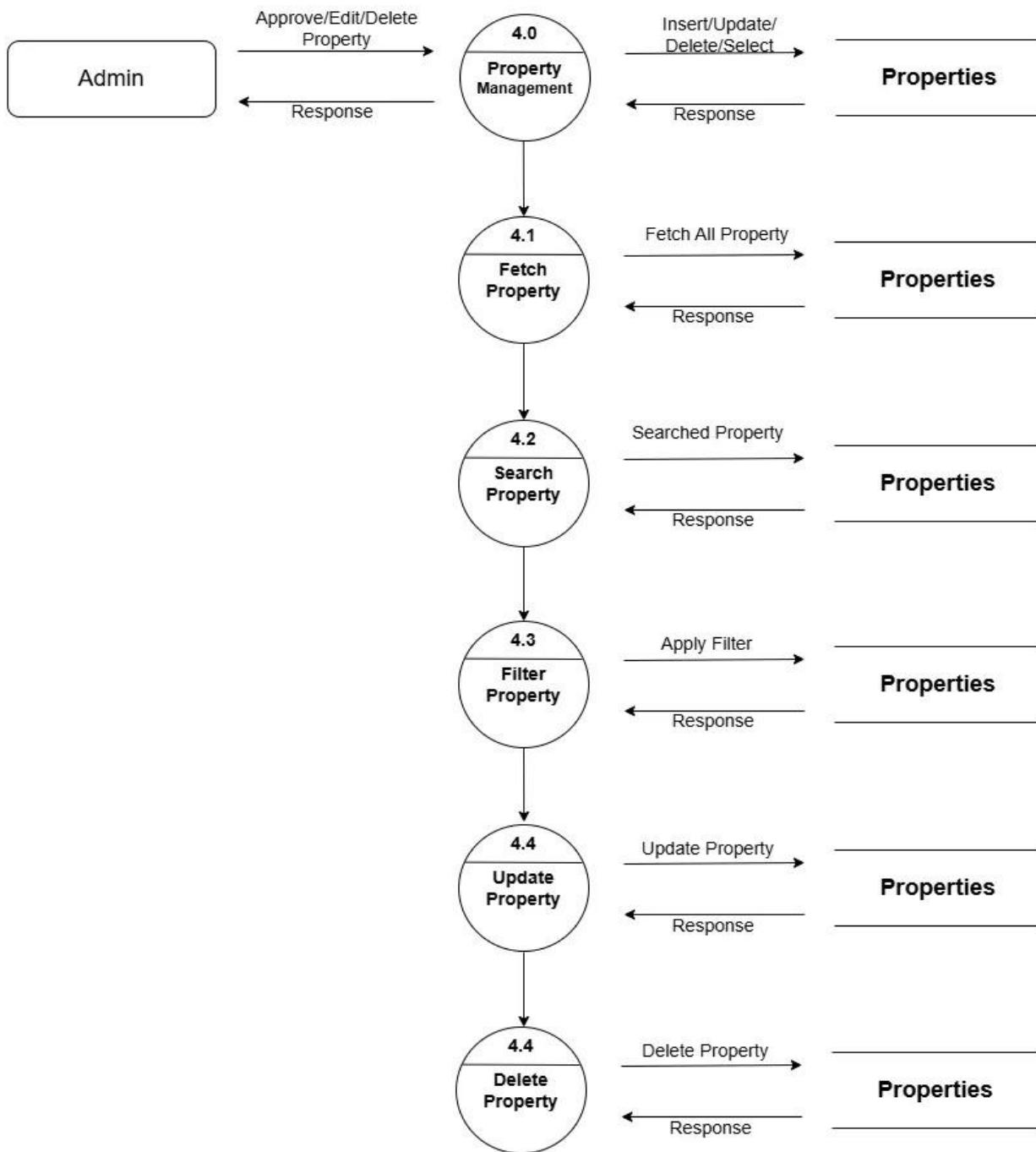


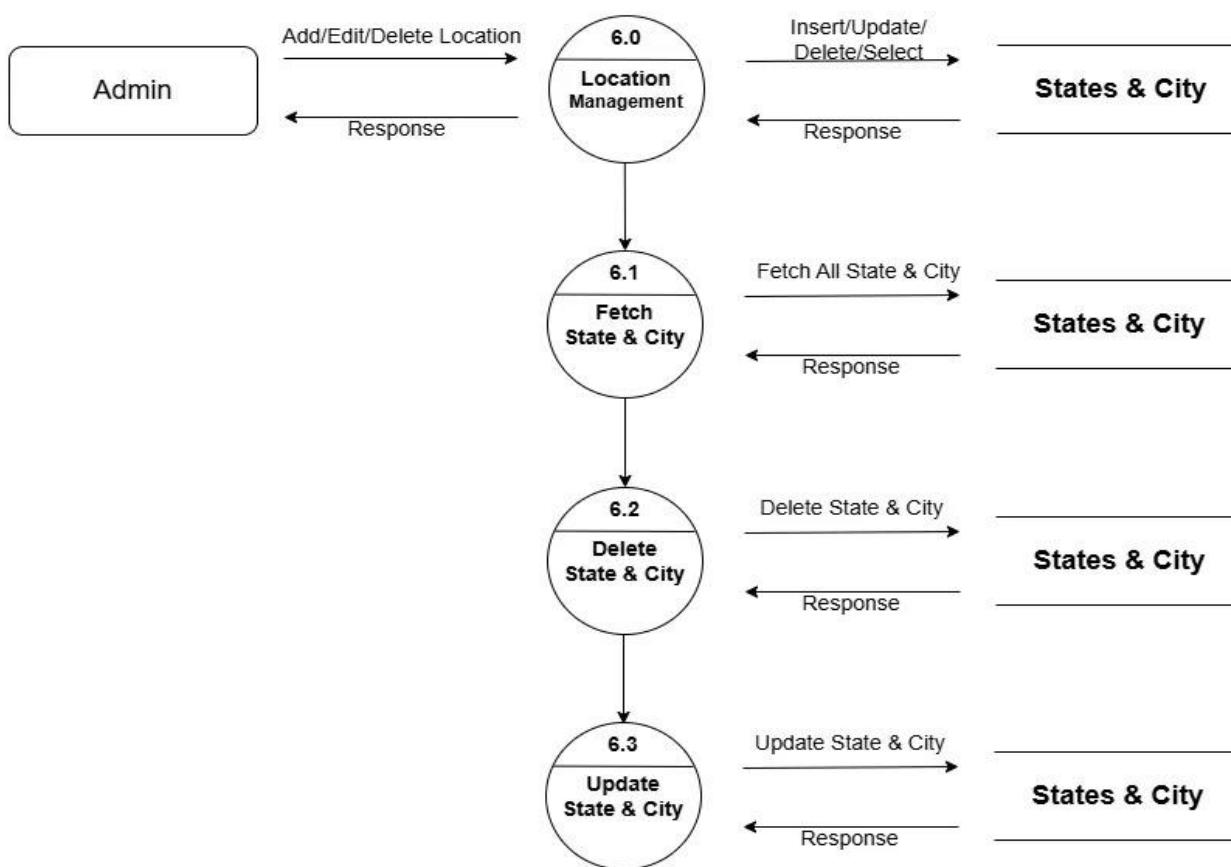
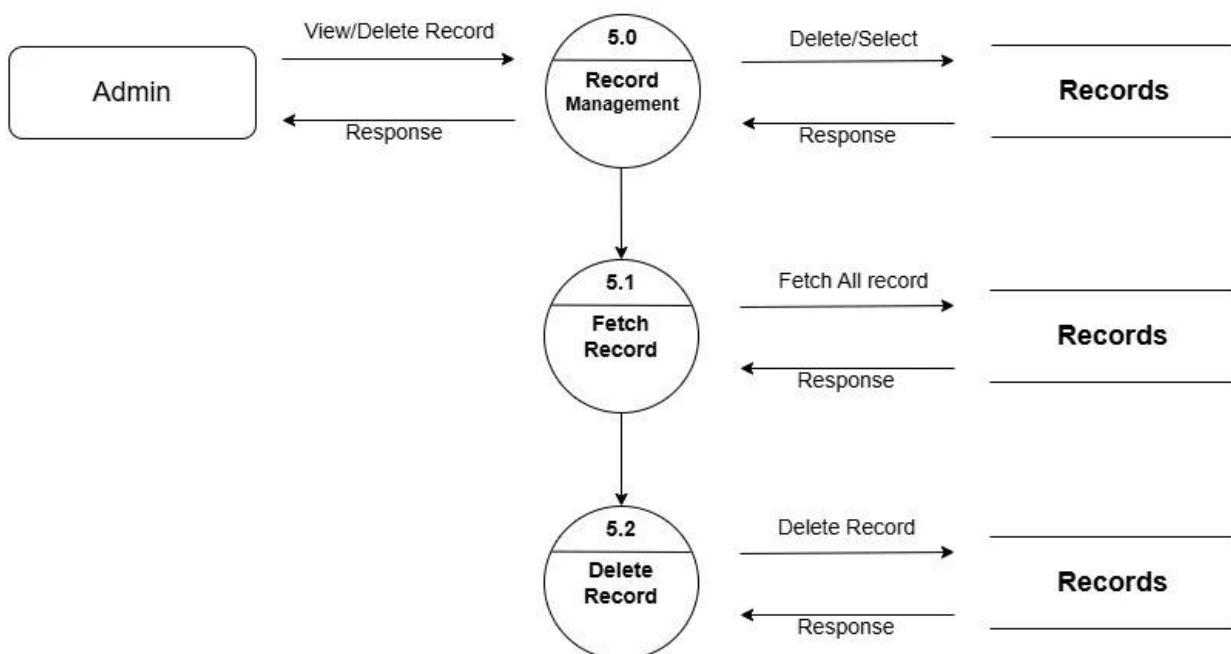
## 1 level Admin DFD: -

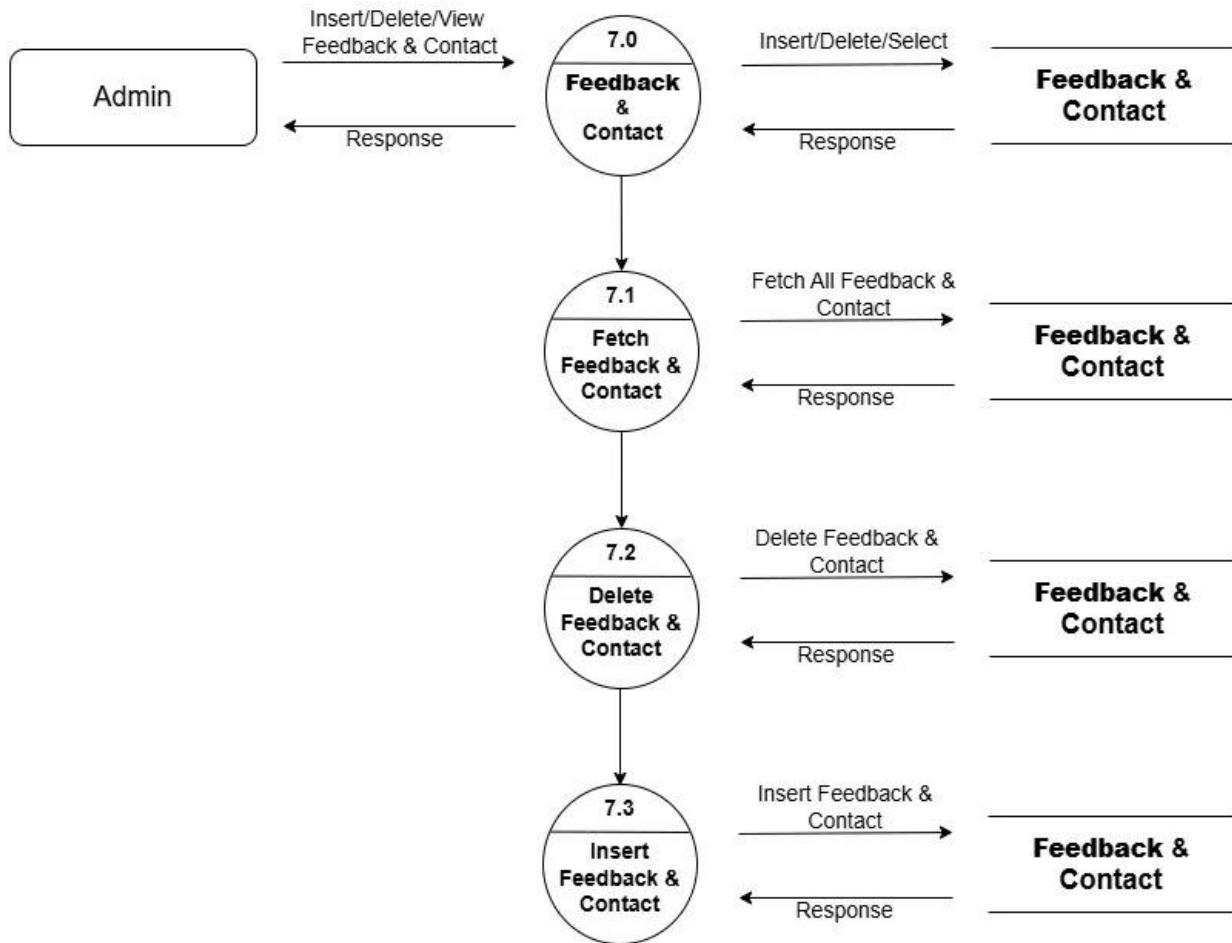


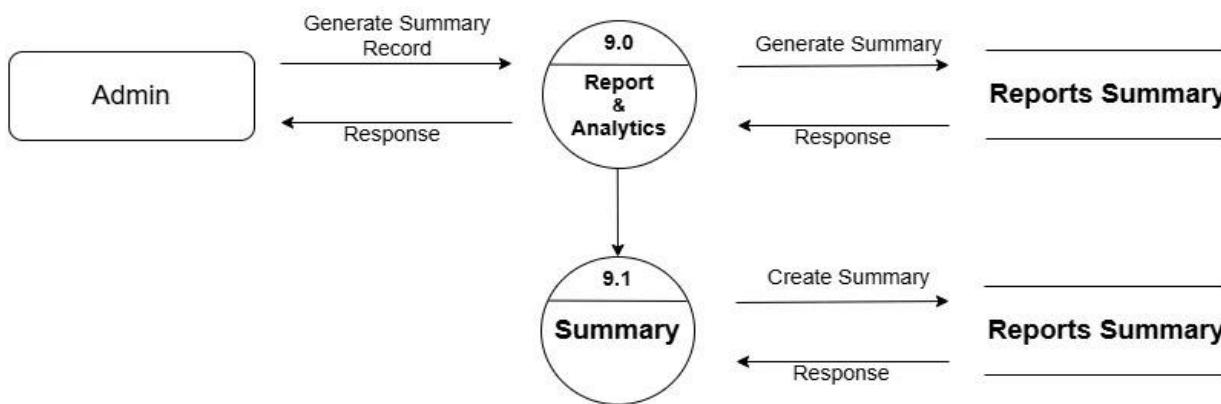
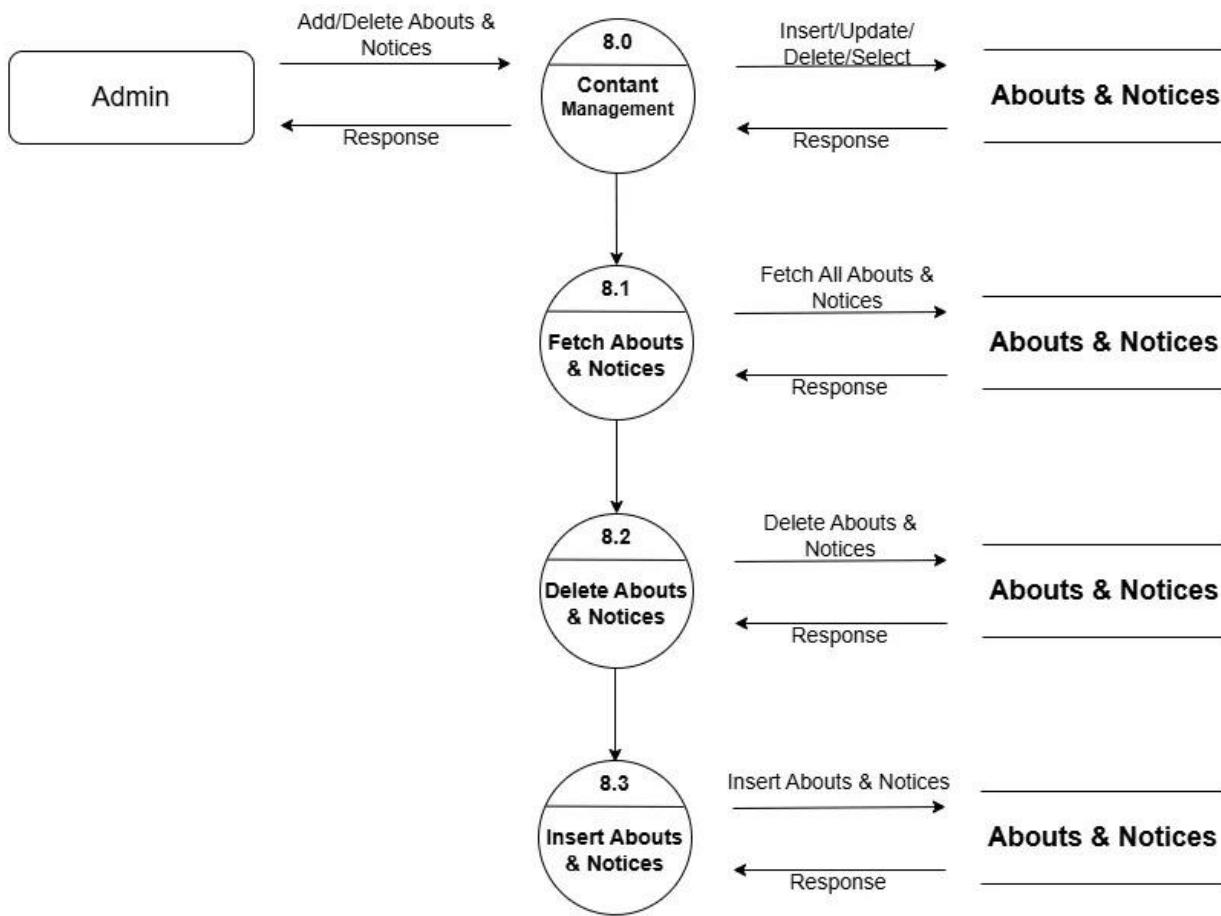
## 2 level Admin DFD: -



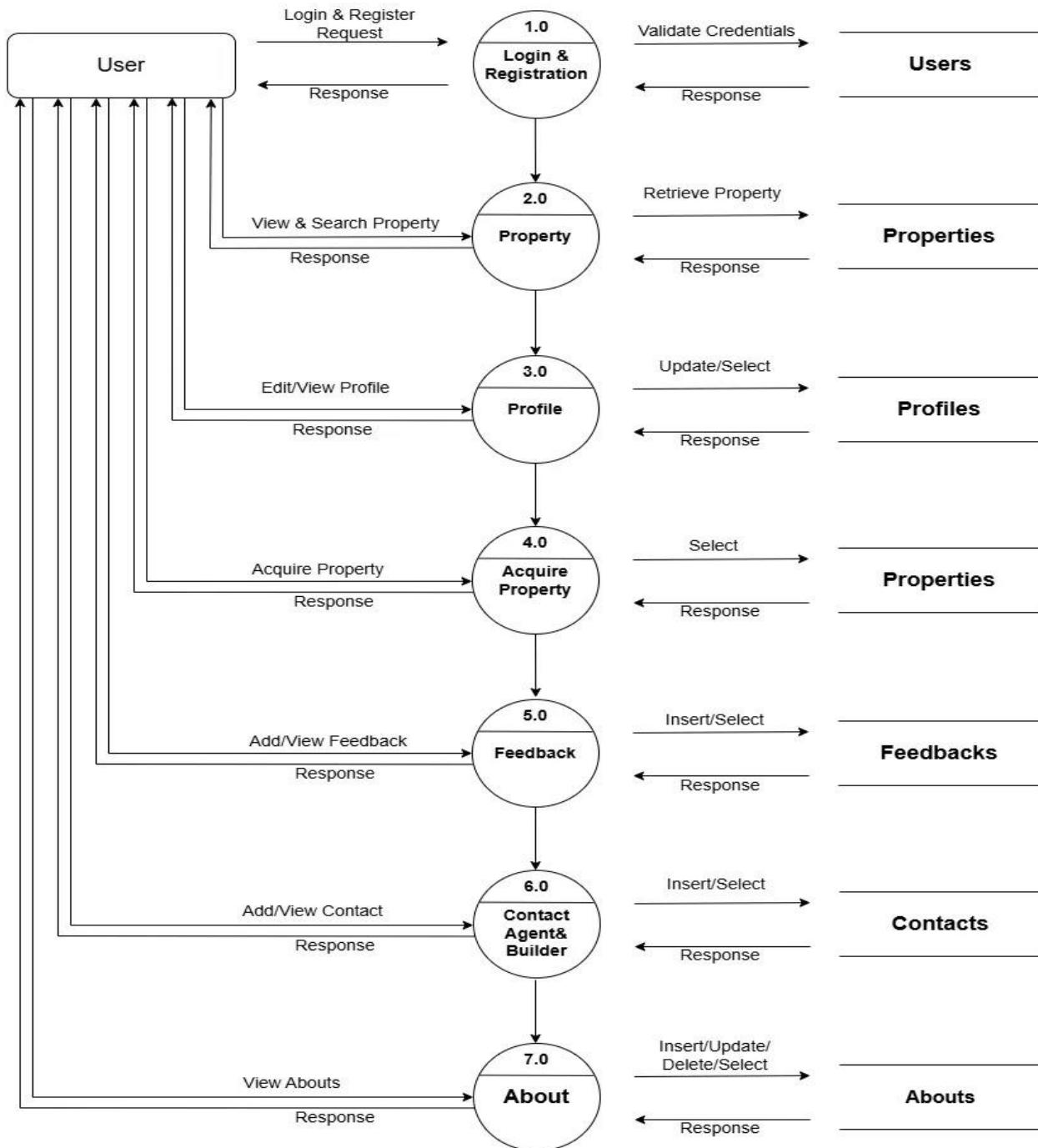




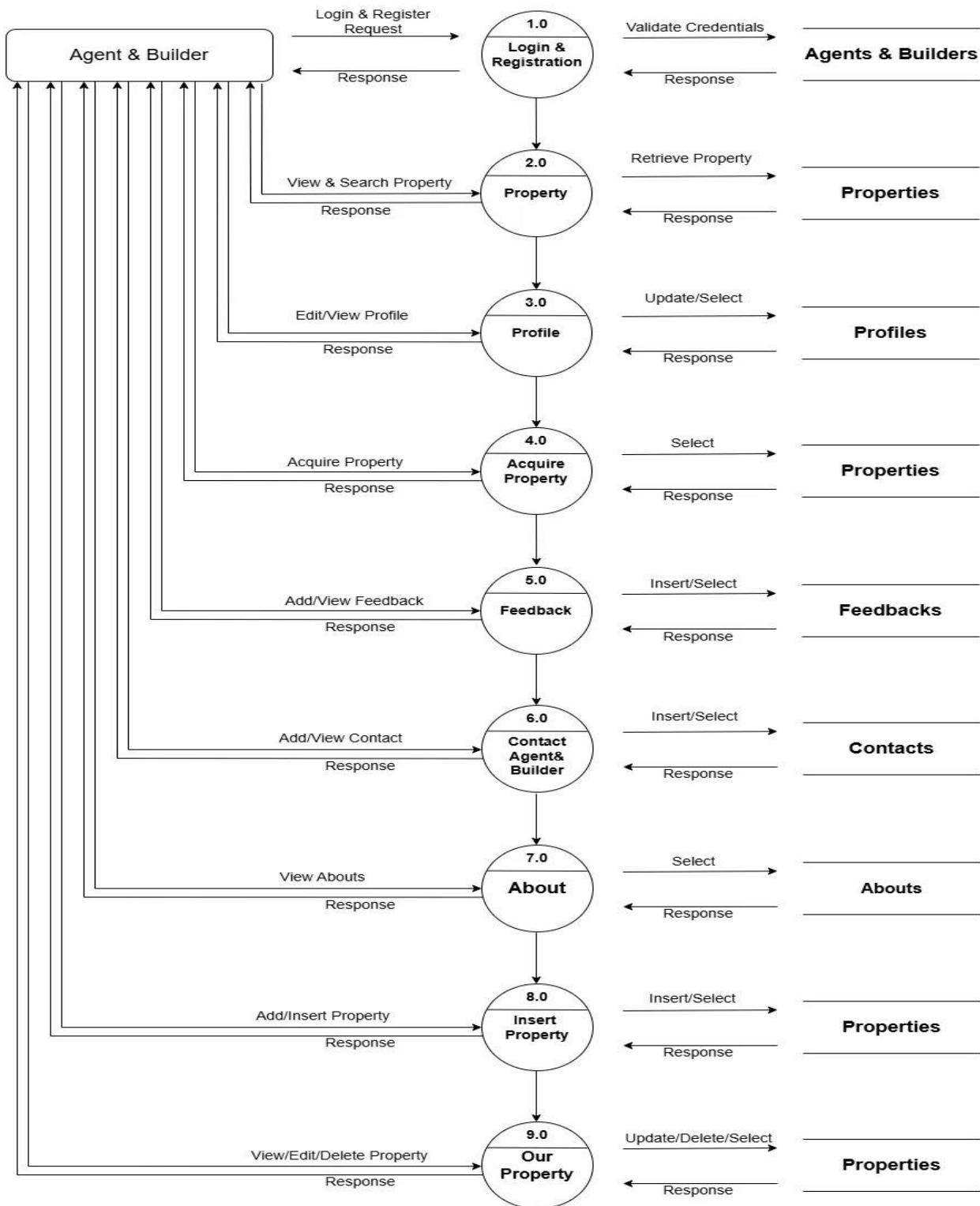




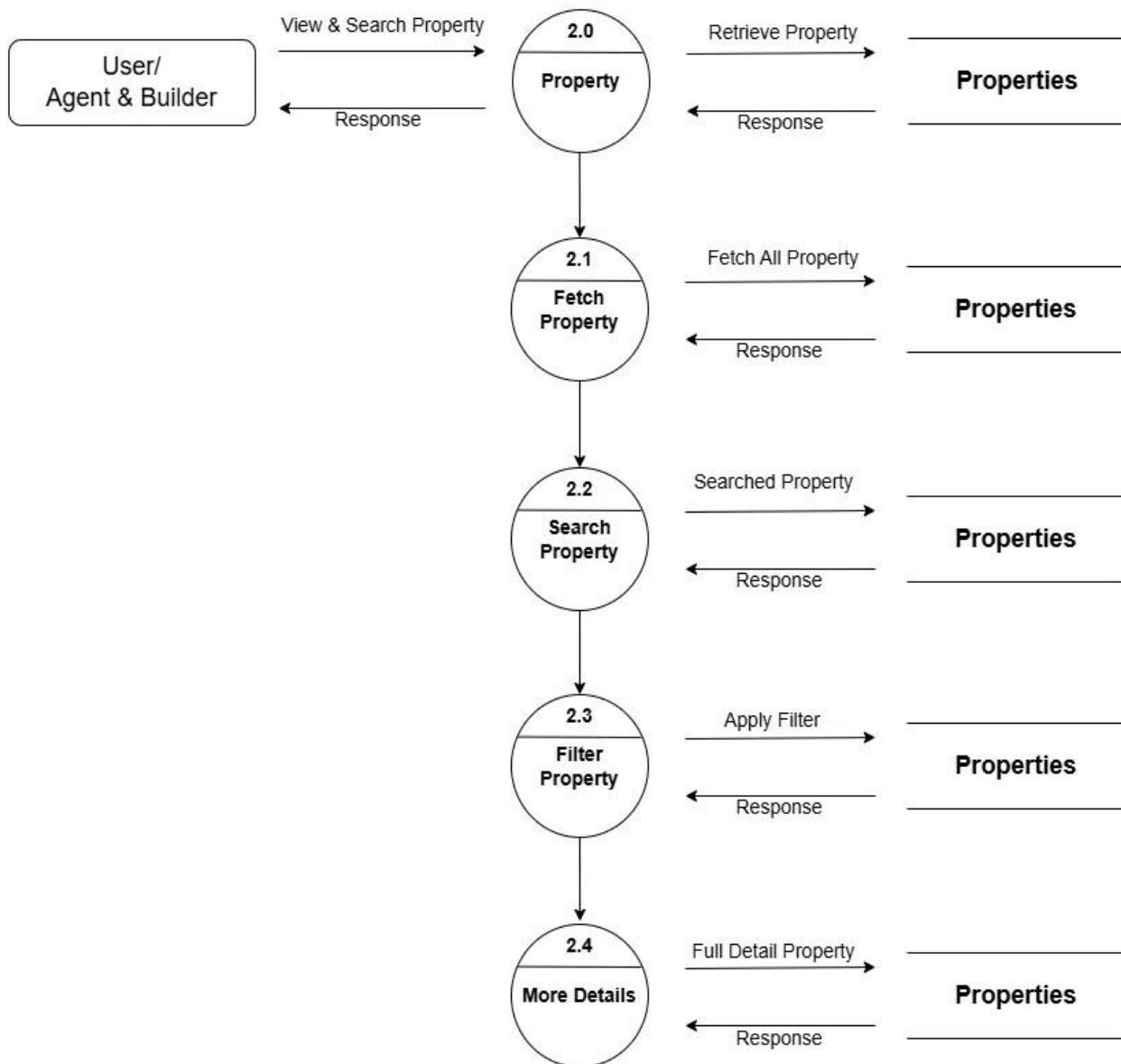
## 1 level User DFD: -

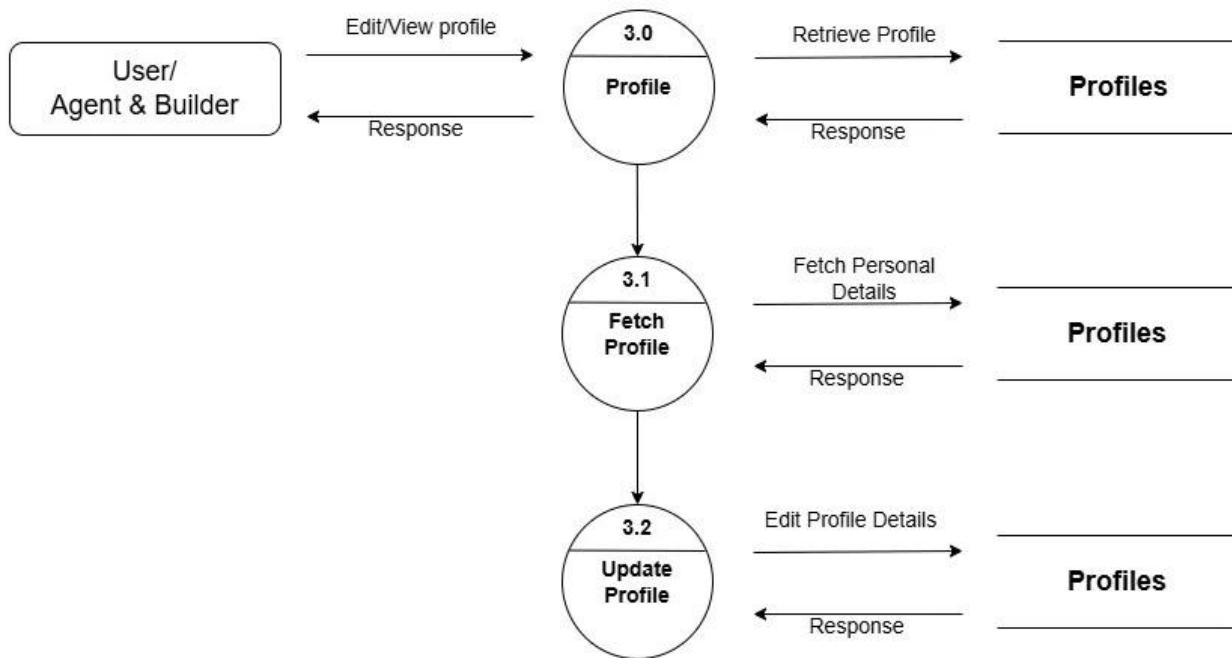


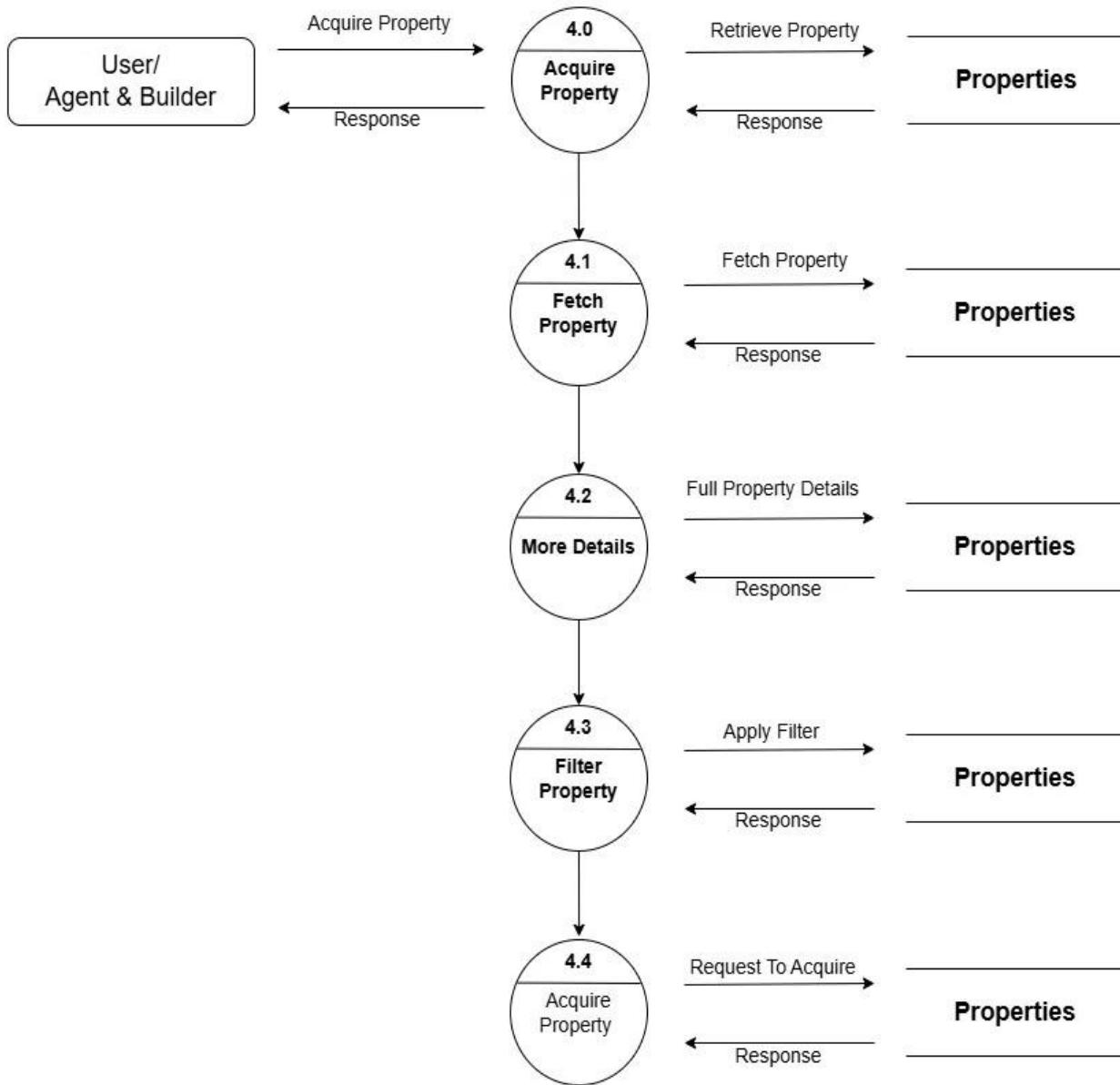
## 1 level Agent & Builder DFD: -

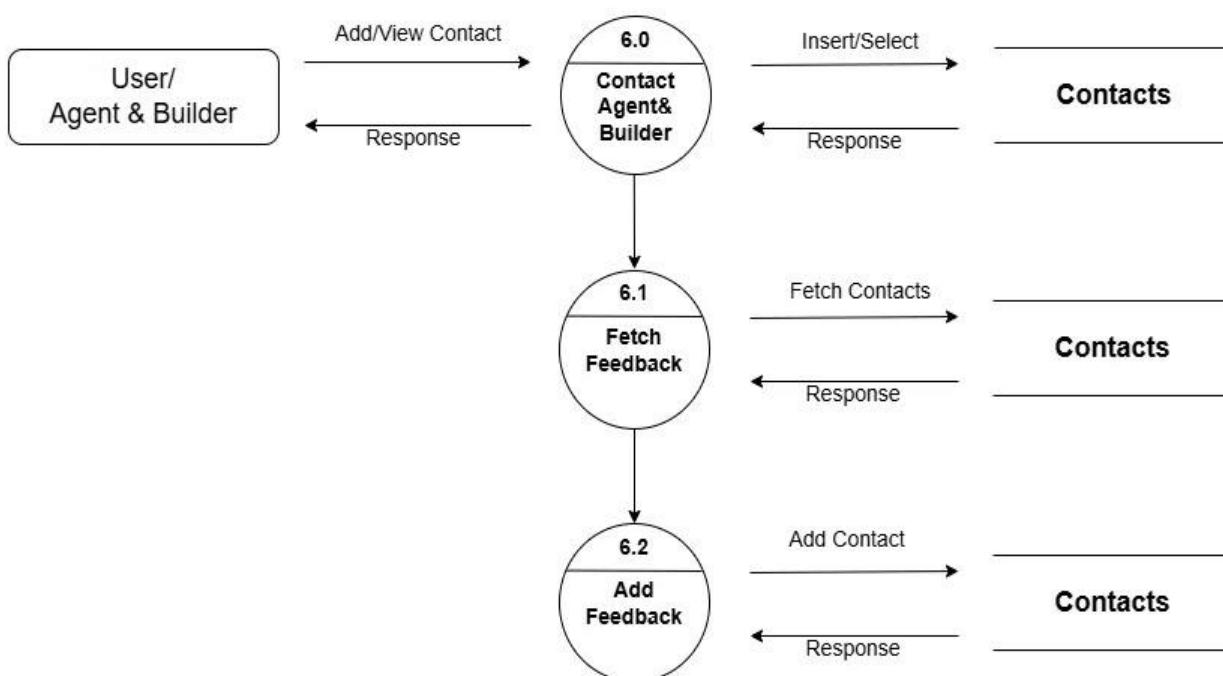
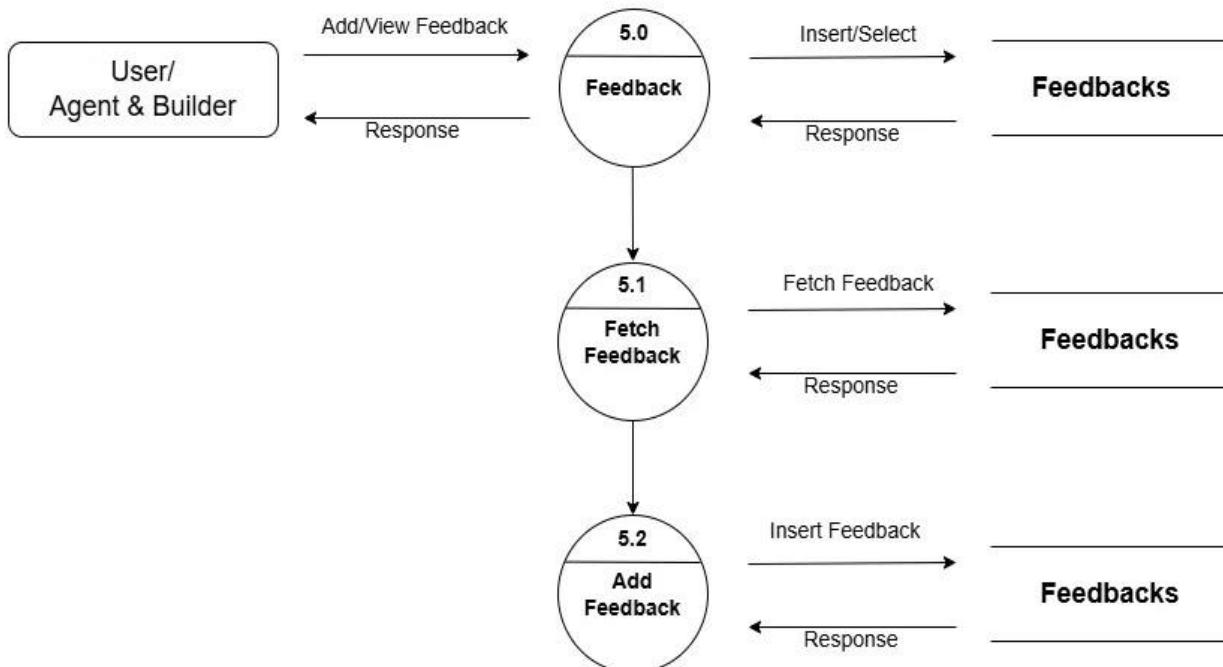


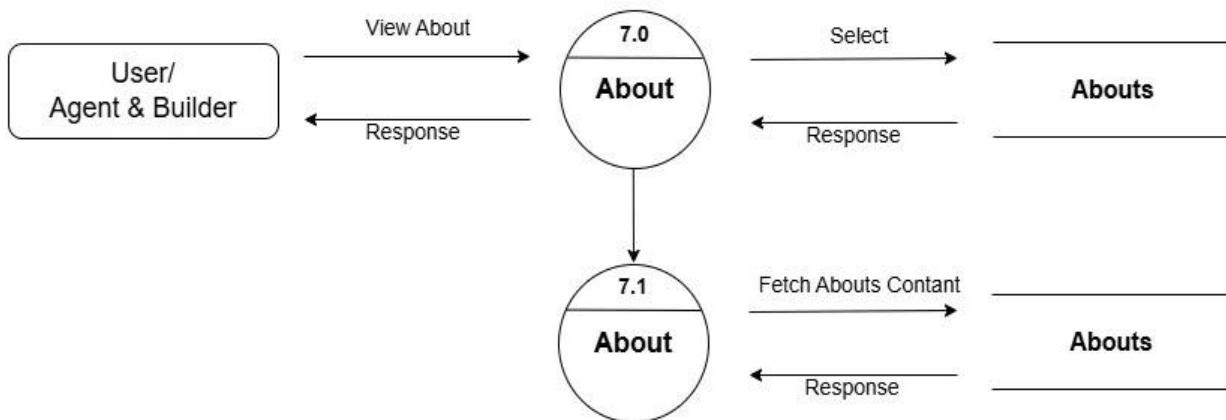
## 1 level User & Agent & Builder DFD: -



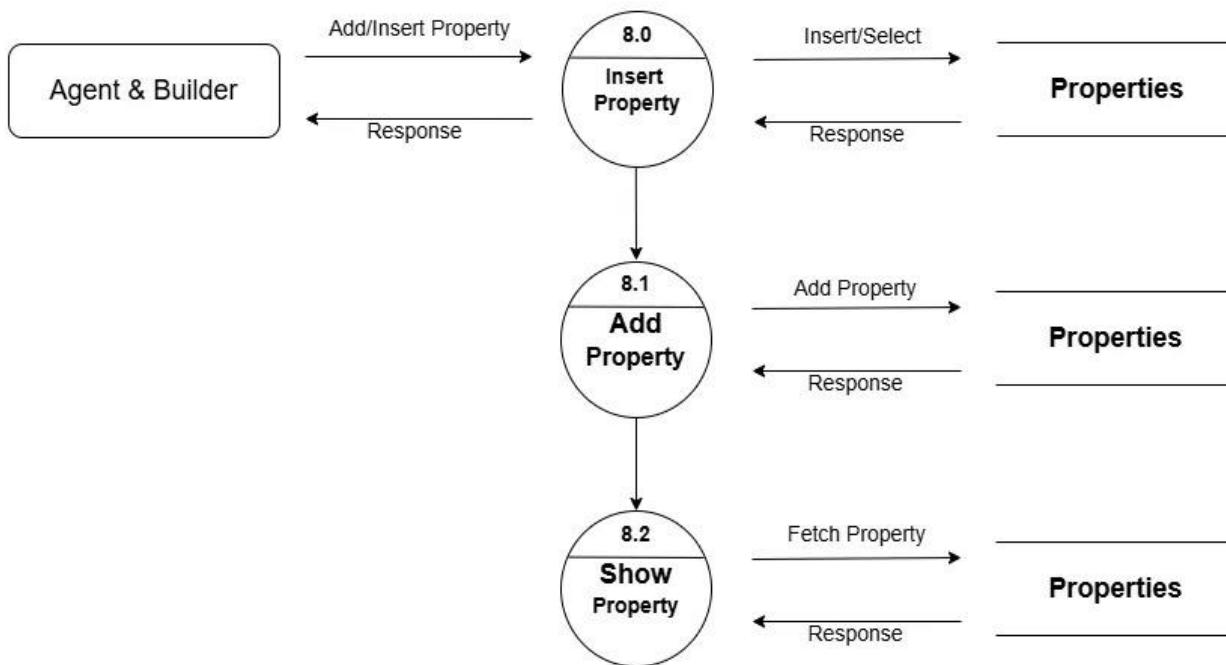


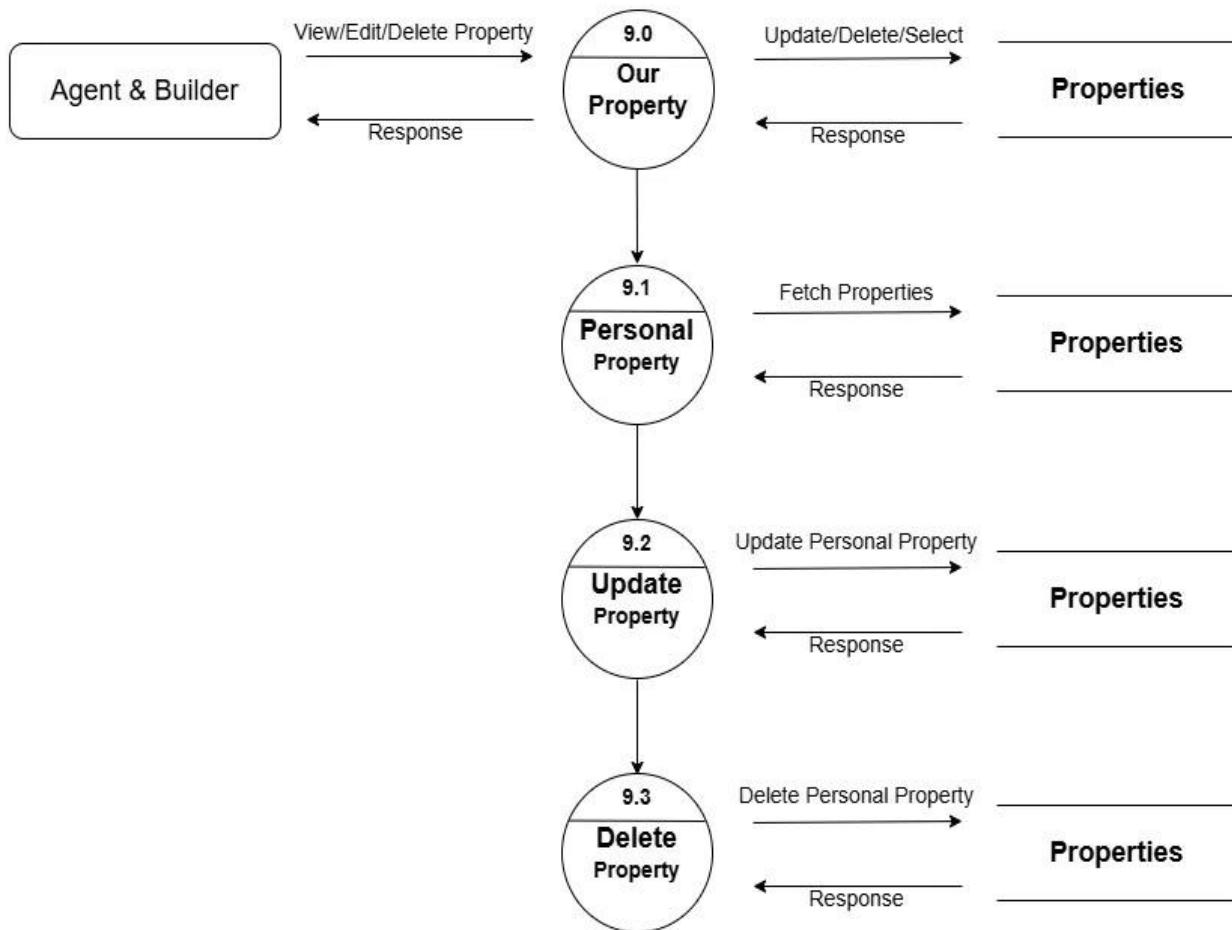




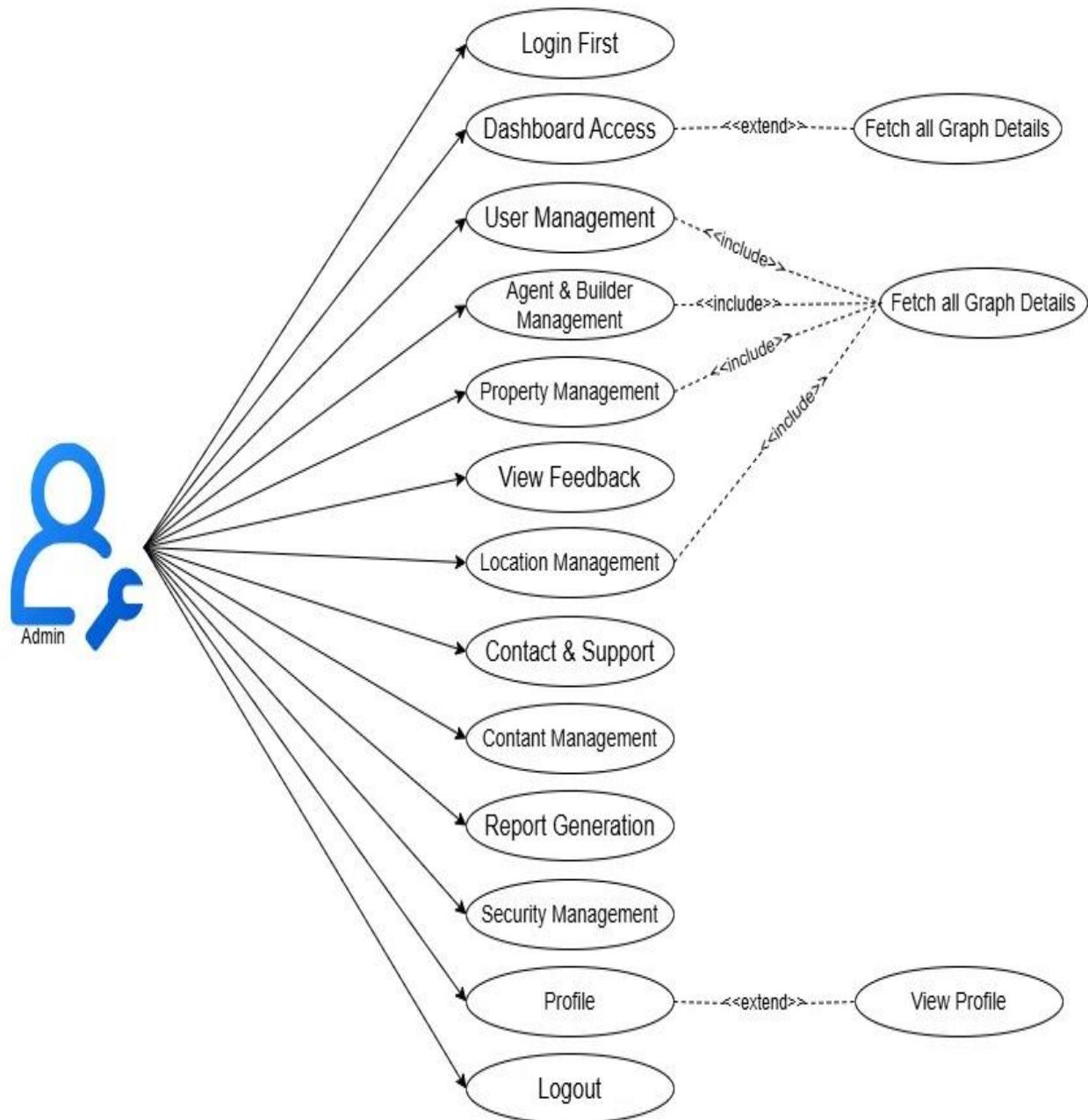


## 2 level Agent & Builder DFD: -

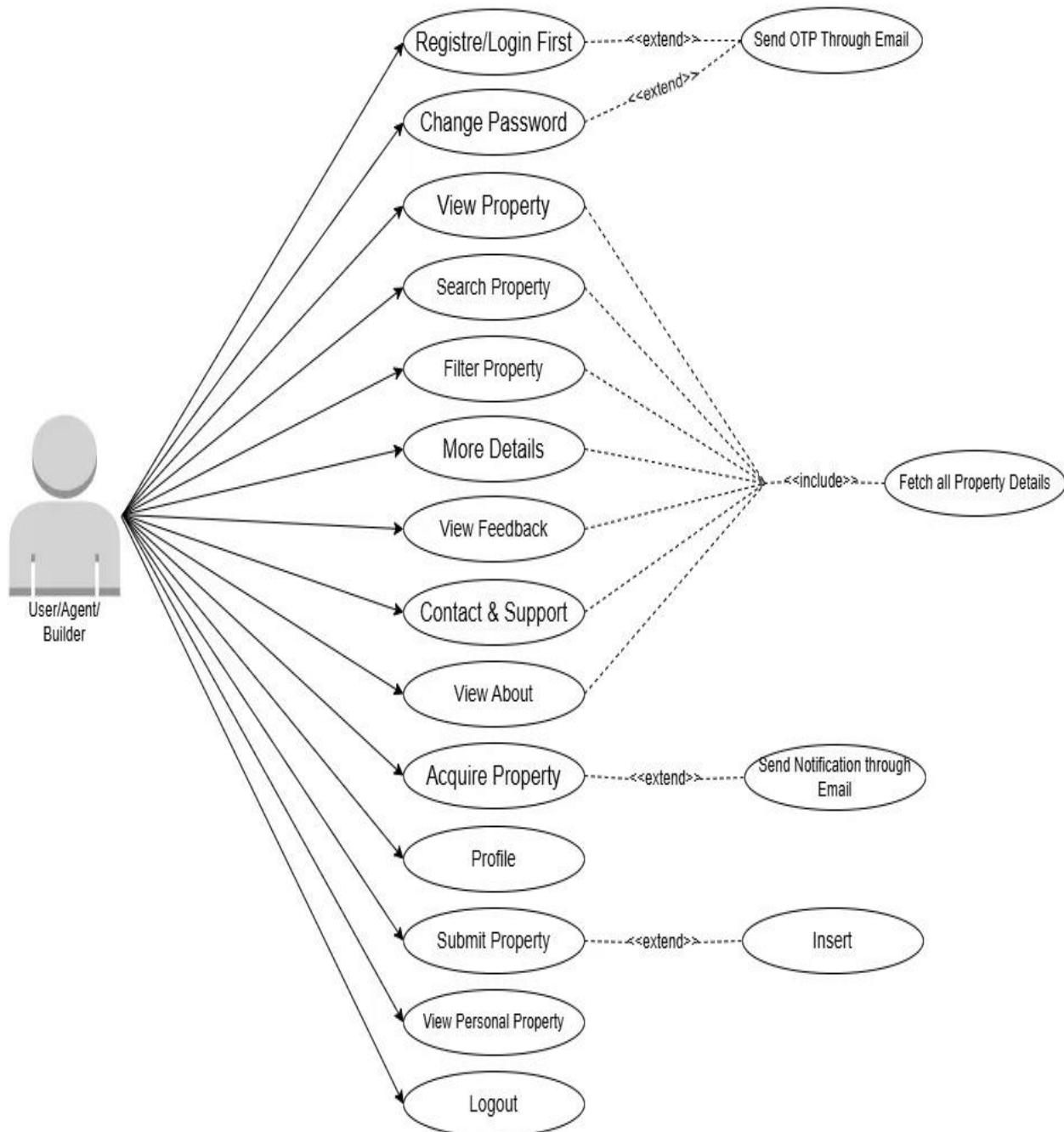




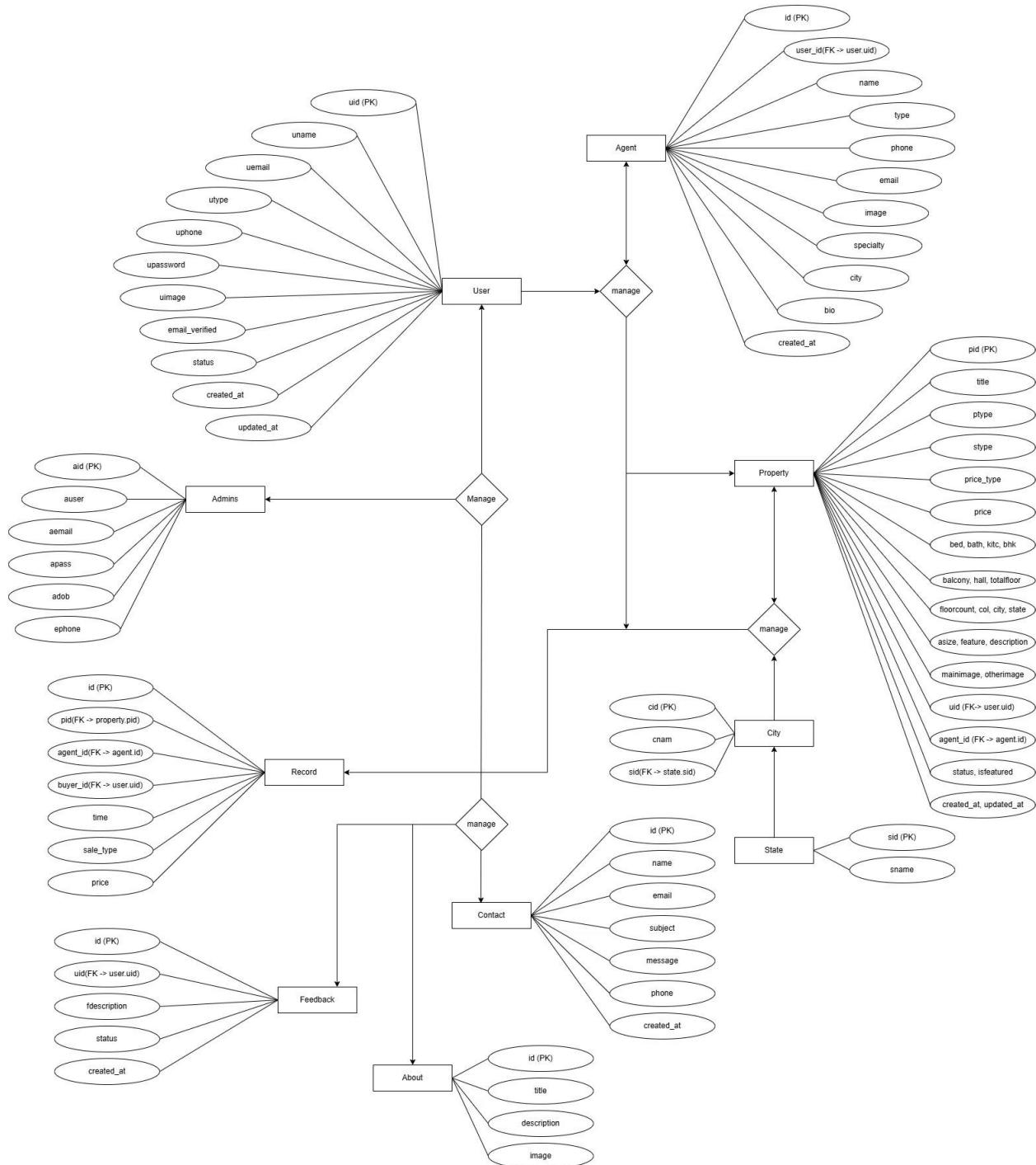
## ADMIN USE CASE DIAGRAM



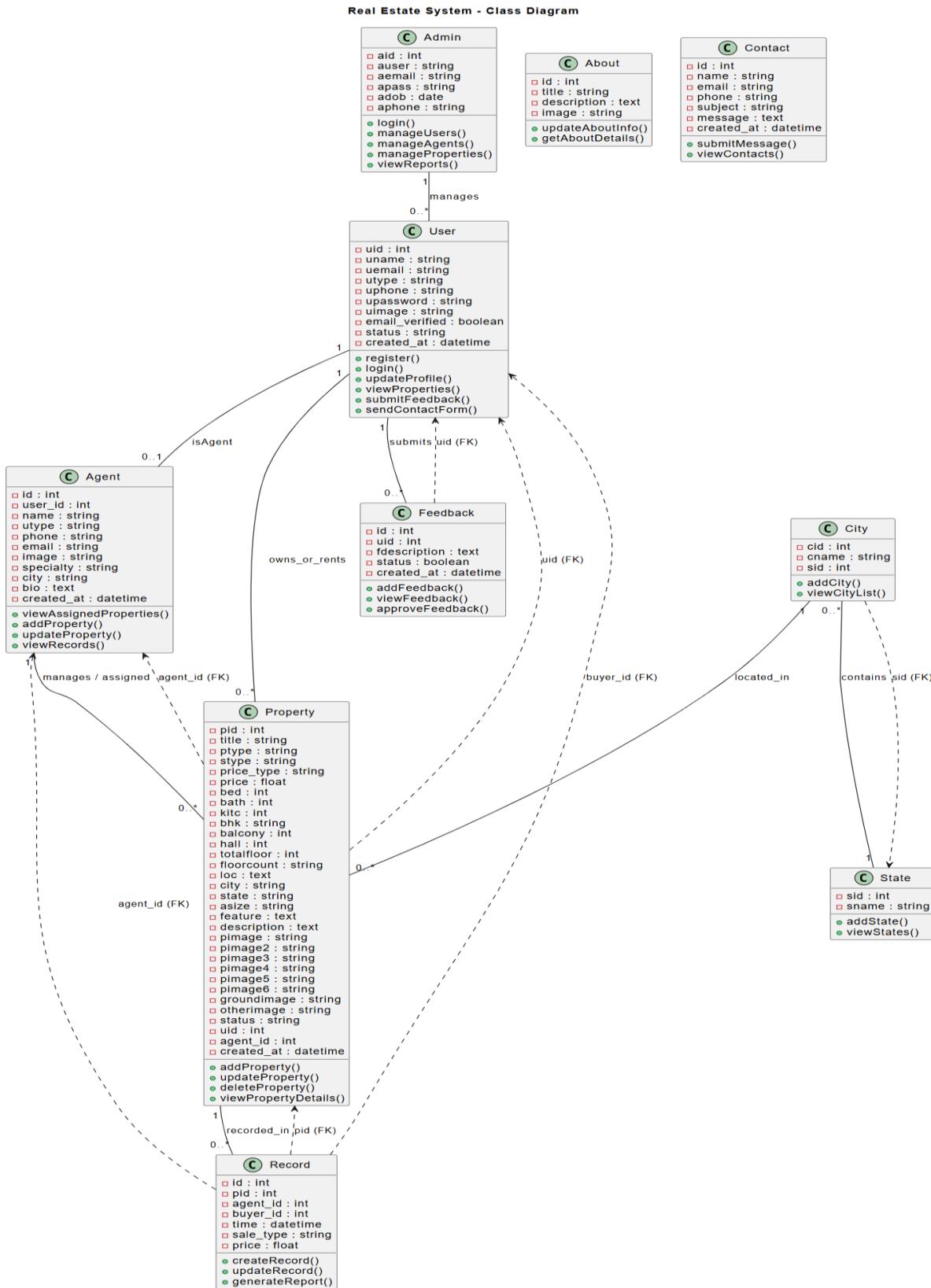
## USER USE CASE DIAGRAM



## ER DIAGRAM



## CLASS DIAGRAM



## DATA DICTIONARY

### Admin:

Stores admin users who manage the platform. Records username, email, password, date of birth, and phone number. Admins have privileged access to manage users, properties, records, and site content. Each admin has a unique id.

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	aid 	int(10)			No	None		AUTO_INCREMENT	 Change  Drop More
<input type="checkbox"/>	2	auser	varchar(50)	utf8mb4_general_ci		No	None			 Change  Drop More
<input type="checkbox"/>	3	aemail 	varchar(50)	utf8mb4_general_ci		No	None			 Change  Drop More
<input type="checkbox"/>	4	apass	varchar(255)	utf8mb4_general_ci		No	None			 Change  Drop More
<input type="checkbox"/>	5	adob	date			No	None			 Change  Drop More
<input type="checkbox"/>	6	aphone	varchar(15)	utf8mb4_general_ci		No	None			 Change  Drop More

## User:

Stores all users of the platform, including buyers, agents, and builders. Records name, email, user type, phone, password, profile image, verification and status info, OTPs, and timestamps. Each user has a unique id.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	<b>uid</b> 	int(10)		UNSIGNED	No	None		AUTO_INCREMENT	 Change  Drop More
2	<b>uname</b>	varchar(100)	utf8mb4_general_ci		No	None			 Change  Drop More
3	<b>uemail</b> 	varchar(100)	utf8mb4_general_ci		No	None			 Change  Drop More
4	<b>utype</b>	enum('user', 'agent', 'builder')	utf8mb4_general_ci		No	None			 Change  Drop More
5	<b>uphone</b>	varchar(20)	utf8mb4_general_ci		No	None			 Change  Drop More
6	<b>upassword</b>	varchar(255)	utf8mb4_general_ci		No	None			 Change  Drop More
7	<b>uimage</b>	varchar(255)	utf8mb4_general_ci		Yes	NULL			 Change  Drop More
8	<b>email_verified</b>	tinyint(1)			No	0			 Change  Drop More
9	<b>status</b>	varchar(20)	utf8mb4_general_ci		No	pending			 Change  Drop More
10	<b>verify_otp</b>	varchar(10)	utf8mb4_general_ci		Yes	NULL			 Change  Drop More
11	<b>otp_expiry</b>	datetime			Yes	NULL			 Change  Drop More
12	<b>created_at</b>	timestamp			No	current_timestamp()			 Change  Drop More
13	<b>updated_at</b>	timestamp			No	current_timestamp()	ON UPDATE CURRENT_TIMESTAMP()		 Change  Drop More
14	<b>reset_password_otp</b>	varchar(6)	utf8mb4_general_ci		Yes	NULL			 Change  Drop More
15	<b>reset_password_otp_expiry</b>	datetime			Yes	NULL			 Change  Drop More

## Agent:

Stores real estate agents registered on the platform. Links to user account, records name, contact info, type (agent/builder), profile image, specialty, city, bio, and social media links. Each agent has a unique id. Agents can list and manage properties.

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	<b>id</b> 	int(11)		UNSIGNED	No	None	AUTO_INCREMENT	 Change  Drop More	
<input type="checkbox"/>	2	<b>user_id</b> 	int(10)		UNSIGNED	No	None		 Change  Drop More	
<input type="checkbox"/>	3	<b>name</b>	varchar(100)	utf8mb4_general_ci		No	None		 Change  Drop More	
<input type="checkbox"/>	4	<b>utype</b>	varchar(20)	utf8mb4_general_ci		Yes	agent		 Change  Drop More	
<input type="checkbox"/>	5	<b>phone</b>	varchar(20)	utf8mb4_general_ci		No	None		 Change  Drop More	
<input type="checkbox"/>	6	<b>email</b>	varchar(100)	utf8mb4_general_ci		No	None		 Change  Drop More	
<input type="checkbox"/>	7	<b>image</b>	varchar(255)	utf8mb4_general_ci		Yes	NULL		 Change  Drop More	
<input type="checkbox"/>	8	<b>specialty</b>	varchar(100)	utf8mb4_general_ci		Yes	Property specialist		 Change  Drop More	
<input type="checkbox"/>	9	<b>city</b>	varchar(100)	utf8mb4_general_ci		Yes	NULL		 Change  Drop More	
<input type="checkbox"/>	10	<b>bio</b>	text	utf8mb4_general_ci		Yes	NULL		 Change  Drop More	
<input type="checkbox"/>	11	<b>facebook</b>	varchar(255)	utf8mb4_general_ci		Yes	https://facebook.com		 Change  Drop More	
<input type="checkbox"/>	12	<b>twitter</b>	varchar(255)	utf8mb4_general_ci		Yes	https://twitter.com		 Change  Drop More	
<input type="checkbox"/>	13	<b>linkedin</b>	varchar(255)	utf8mb4_general_ci		Yes	https://linkedin.com/in		 Change  Drop More	
<input type="checkbox"/>	14	<b>created_at</b>	timestamp			No	current_timestamp()		 Change  Drop More	

## State:

Stores state names available in the system. Each state has a unique id. Used to organize cities and support location-based searches.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	<b>sid</b> 	int(50)		No	None		AUTO_INCREMENT	 Change  Drop More
<input type="checkbox"/>	2	<b>sname</b>	varchar(100)	utf8mb4_general_ci	No	None			 Change  Drop More

## City:

Stores city names available in the system, with a reference to their corresponding state. Each city has a unique id. Used to associate properties and users with a location.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	cid	int(50)			No	None		AUTO_INCREMENT	Change  Drop  More
2	cname	varchar(100)	utf8mb4_general_ci		No	None			Change  Drop  More
3	sid	int(50)			No	None			Change  Drop  More

## Property:

Stores listings for properties on the platform. Contains title, type, sale/rent type, price, number of rooms, amenities, images, location details, status (available/sold out), and references to user and agent. Each property has a unique id.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	pid 📈	int(11)			No	None		AUTO_INCREMENT	Change  Drop  More
2	title	varchar(255)	utf8mb4_general_ci		No	None			Change  Drop  More
3	ptype 🔑	varchar(50)	utf8mb4_general_ci		No	None			Change  Drop  More
4	stype 💳	varchar(50)	utf8mb4_general_ci		No	None			Change  Drop  More
5	price_type	varchar(10)	utf8mb4_general_ci		Yes	NULL			Change  Drop  More
6	price	decimal(12,2)			No	None			Change  Drop  More
7	bed	int(11)			No	None			Change  Drop  More
8	bath	int(11)			No	None			Change  Drop  More
9	kitc	int(11)			No	None			Change  Drop  More
10	bhk	varchar(50)	utf8mb4_general_ci		No	None			Change  Drop  More
11	balcony	int(11)			No	None			Change  Drop  More
12	hall	int(11)			No	None			Change  Drop  More
13	totalfloor	int(11)			No	None			Change  Drop  More
14	floorcount	varchar(50)	utf8mb4_general_ci		No	None			Change  Drop  More
15	loc	text	utf8mb4_general_ci		No	None			Change  Drop  More
16	city	varchar(100)	utf8mb4_general_ci		No	None			Change  Drop  More
17	state	varchar(100)	utf8mb4_general_ci		No	None			Change  Drop  More
18	asize	varchar(50)	utf8mb4_general_ci		No	None			Change  Drop  More
19	feature	longtext	utf8mb4_general_ci		Yes	NULL			Change  Drop  More
20	pimage	varchar(255)	utf8mb4_general_ci		No	None			Change  Drop  More
21	pimage1	varchar(255)	utf8mb4_general_ci		No	None			Change  Drop  More
22	pimage2	varchar(255)	utf8mb4_general_ci		No	None			Change  Drop  More
23	pimage3	varchar(255)	utf8mb4_general_ci		No	None			Change  Drop  More
24	pimage4	varchar(255)	utf8mb4_general_ci		No	None			Change  Drop  More
25	pimage5	varchar(255)	utf8mb4_general_ci		Yes	NULL			Change  Drop  More
26	pimage6	varchar(255)	utf8mb4_general_ci		Yes	NULL			Change  Drop  More
27	groundimage	varchar(255)	utf8mb4_general_ci		Yes	NULL			Change  Drop  More
28	otherimage	varchar(255)	utf8mb4_general_ci		Yes	NULL			Change  Drop  More
29	status	enum('available', 'sold out')	utf8mb4_general_ci		Yes	NULL			Change  Drop  More
30	uid 📈	int(11)			Yes	NULL			Change  Drop  More
31	agent_id	int(11)			Yes	NULL			Change  Drop  More
32	isFeatured	tinyint(1)			Yes	0			Change  Drop  More
33	description	text	utf8mb4_general_ci		Yes	NULL			Change  Drop  More
34	created_at	timestamp			No	current_timestamp()			Change  Drop  More
35	updated_at	timestamp			No	current_timestamp()	ON UPDATE CURRENT_TIMESTAMP()		Change  Drop  More

## Record:

Stores property transaction records. Links property, agent, and buyer, along with transaction time, sale/rent type, and price. Used for tracking sales and rentals. Each record has a unique id.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	<b>id</b> 	int(11)			No	None		AUTO_INCREMENT	 Change  Drop More
2	<b>pid</b> 	int(11)			No	None			 Change  Drop More
3	<b>agent_id</b> 	int(11)		UNSIGNED	No	None			 Change  Drop More
4	<b>time</b>	datetime			No	None			 Change  Drop More
5	<b>sale_type</b>	varchar(50)	utf8mb4_general_ci		Yes	NULL			 Change  Drop More
6	<b>price</b>	decimal(12,2)			No	None			 Change  Drop More
7	<b>buyer_id</b> 	int(10)		UNSIGNED	No	None			 Change  Drop More

## About:

Stores informational content about the company, its mission, or services. Contains a title, description, and optional image for each entry. Used for displaying "About Us" sections on the site. Each entry has a unique id.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	<b>id</b> 	int(11)			No	None		AUTO_INCREMENT	 Change  Drop More
2	<b>title</b>	varchar(255)	utf8mb4_general_ci		No	None			 Change  Drop More
3	<b>description</b>	text	utf8mb4_general_ci		No	None			 Change  Drop More
4	<b>image</b>	varchar(255)	utf8mb4_general_ci		Yes	NULL			 Change  Drop More

## Feedback:

Stores user feedback about the platform or agents. Records user reference, feedback description, status flag (e.g., reviewed), and creation date. Each feedback entry has a unique id.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	<b>id</b> 📄	int(11)			No	None		AUTO_INCREMENT	Change  Drop  More
2	<b>uid</b> 🏠	int(10)		UNSIGNED	No	None			Change  Drop  More
3	<b>fdescription</b>	text	utf8mb4_general_ci		No	None			Change  Drop  More
4	<b>status</b>	tinyint(1)			No	0			Change  Drop  More
5	<b>created_at</b>	timestamp			No	current_timestamp()			Change  Drop  More

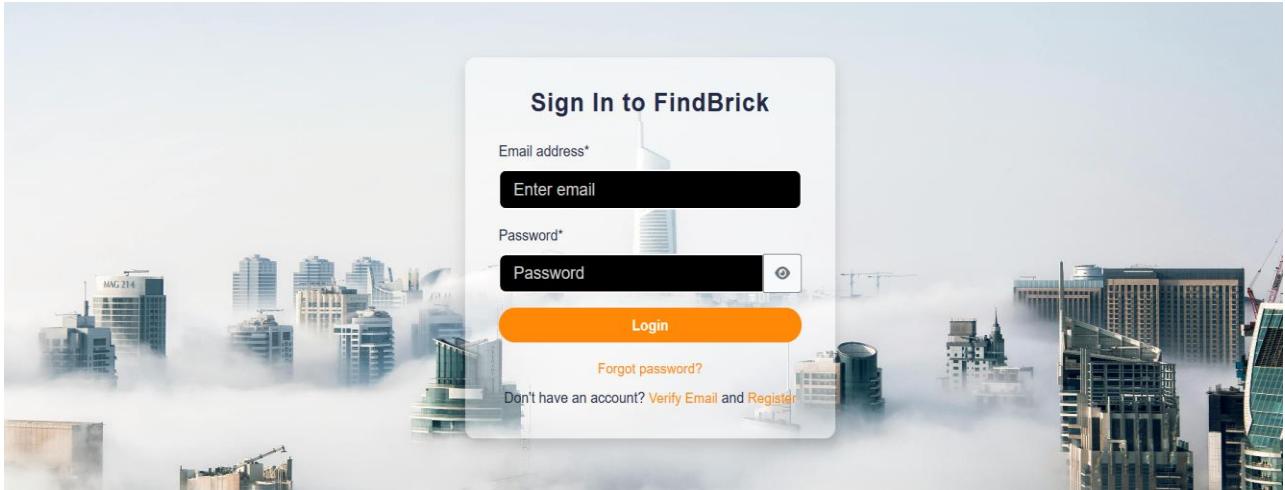
## Contact:

Stores messages sent via the website's contact form. Records sender's name, email, phone number, subject, message body, and creation date. Each message has a unique id.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	<b>id</b> 🚪	int(11)			No	None		AUTO_INCREMENT	Change  Drop  More
2	<b>name</b>	varchar(100)	utf8mb4_general_ci		No	None			Change  Drop  More
3	<b>email</b>	varchar(150)	utf8mb4_general_ci		No	None			Change  Drop  More
4	<b>phone</b>	varchar(20)	utf8mb4_general_ci		No	None			Change  Drop  More
5	<b>subject</b>	varchar(200)	utf8mb4_general_ci		No	None			Change  Drop  More
6	<b>message</b>	text	utf8mb4_general_ci		No	None			Change  Drop  More
7	<b>created_at</b>	timestamp			No	current_timestamp()			Change  Drop  More

## CLIENT-SIDE SNAPSHOT:

### Login Page: -



### Page Description: -

This page allows User to securely log in using their email and password.

After successful login, they are redirected to the Home Page.

### Code Explanation: -

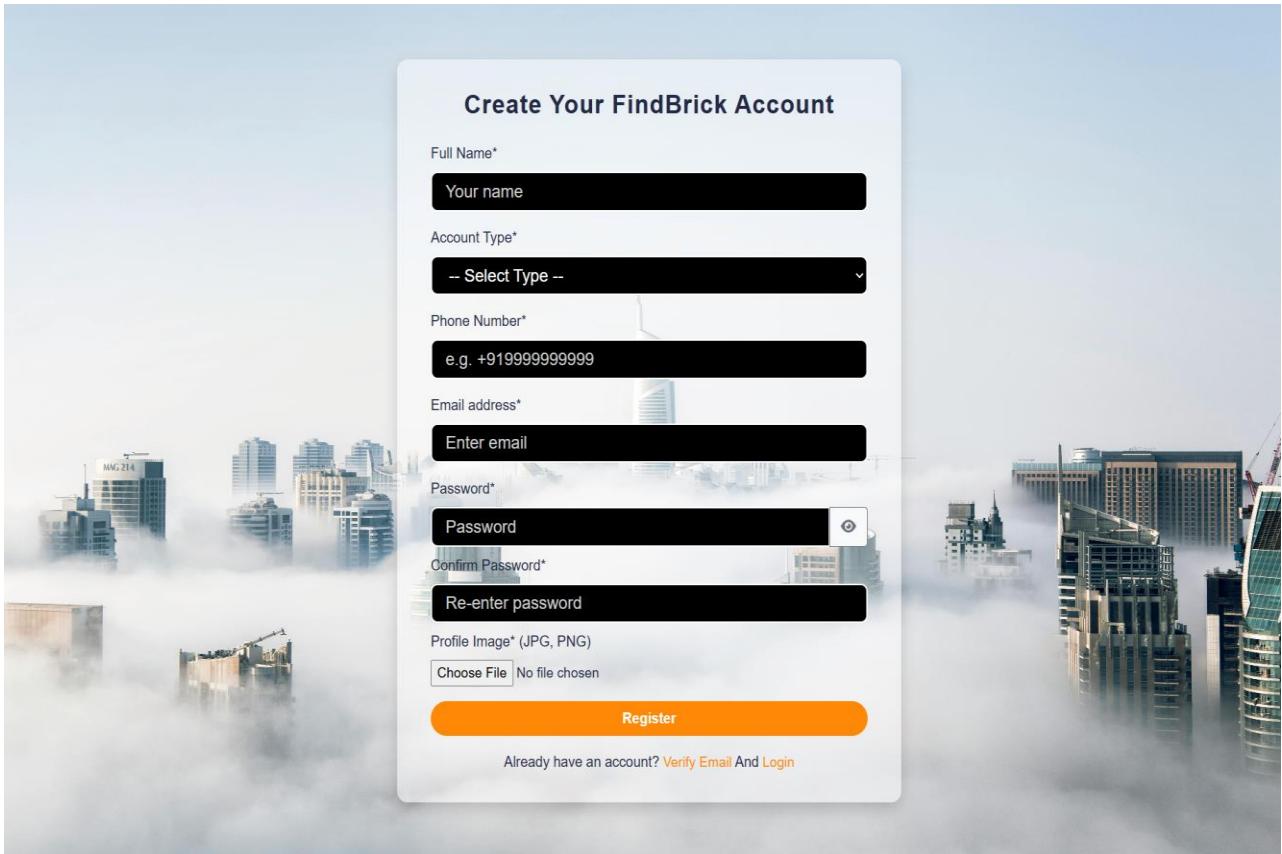
If a session UID already exists, the script redirects to index.php to prevent re-login.

On POST it validates email/password, then uses a prepared SELECT to load the user row (uid, hashed password, email\_verified, status).

It checks the account is verified/active, verifies the submitted password with password\_verify(), regenerates the session ID, and stores uid/uemail in \$\_SESSION then redirects.

Errors and validation messages are shown to the user; output has escaped when printed in the form. Uses prepared statements to avoid SQL injection.

## Register Page: -



## Page Description: -

Allows new users to create an account by providing their personal information, such as name, email, phone number, password, image, and user type (buyer, agent, or builder).

## Code Explanation: -

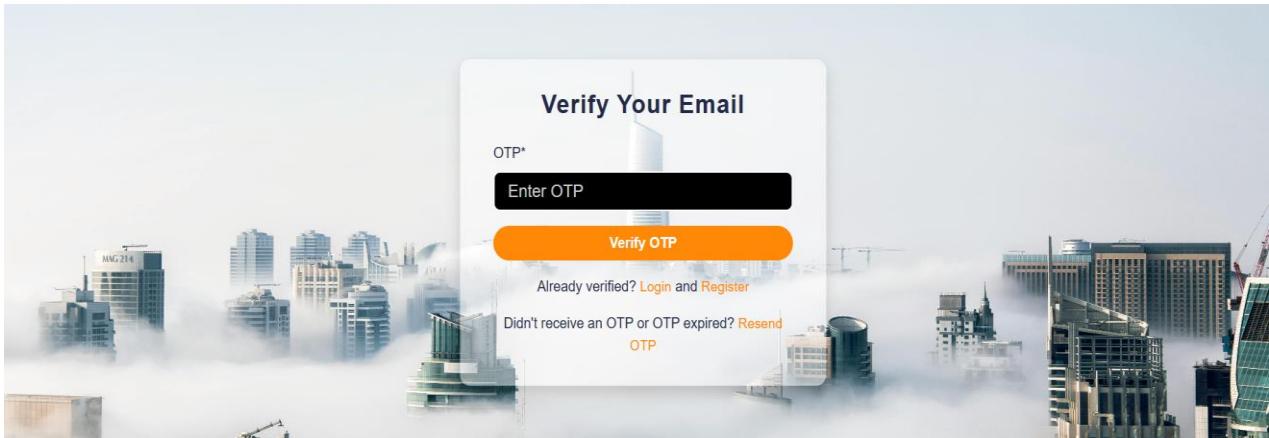
Validates all fields (name, email format, phone digits, password strength and confirmation) and enforces profile image presence, allowed types and max size (2MB).

Saves the uploaded image under admin/ with a unique name after checking/creating the upload directory.

Hashes the password, generates a 6-digit OTP and expiry, inserts a new user row with email\_verified=0 and status='pending', and emails the OTP to the user.

Stores the pending email in session, calls agentornot() to create an agent row for agent/builder accounts, then redirects to verify\_email.php for OTP verification.

## Email Verify Page: -



## Page Description: -

Enables users to verify their email or phone number by entering a One Time Password (OTP) sent to them during registration, ensuring secure account activation.

## Code Explanation: -

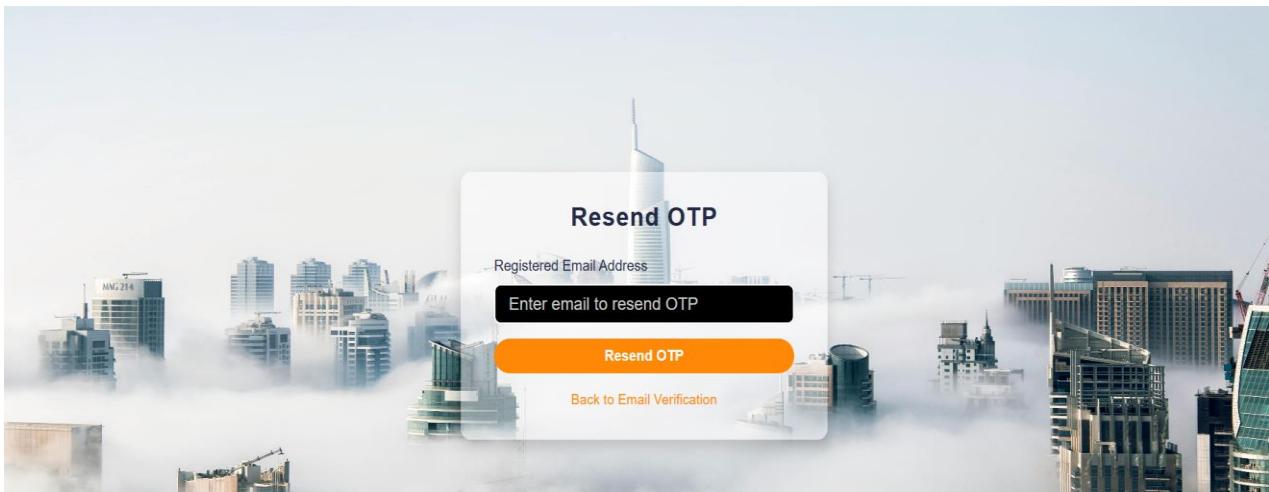
Reads the pending email from session (or GET fallback) and requires an OTP POST to continue.

Uses a prepared SELECT to find a matching user row where verify\_otp matches and email\_verified is 0 and status is 'pending'.

Checks the stored otp\_expiry timestamp against the current time; if valid it updates the row to set email\_verified=1, status='active', clears verify\_otp/otp\_expiry, unsets the session pending email, and redirects to login.

On failure it sets appropriate error/validation messages and offers an OTP resend link.

## Resent OTP Page: -



## Page Description: -

Form to request a fresh verification OTP for a pending (unverified) account — generates a new 6-digit code, updates the DB expiry, sends the email and redirects back to email verification.

## Code Explanation: -

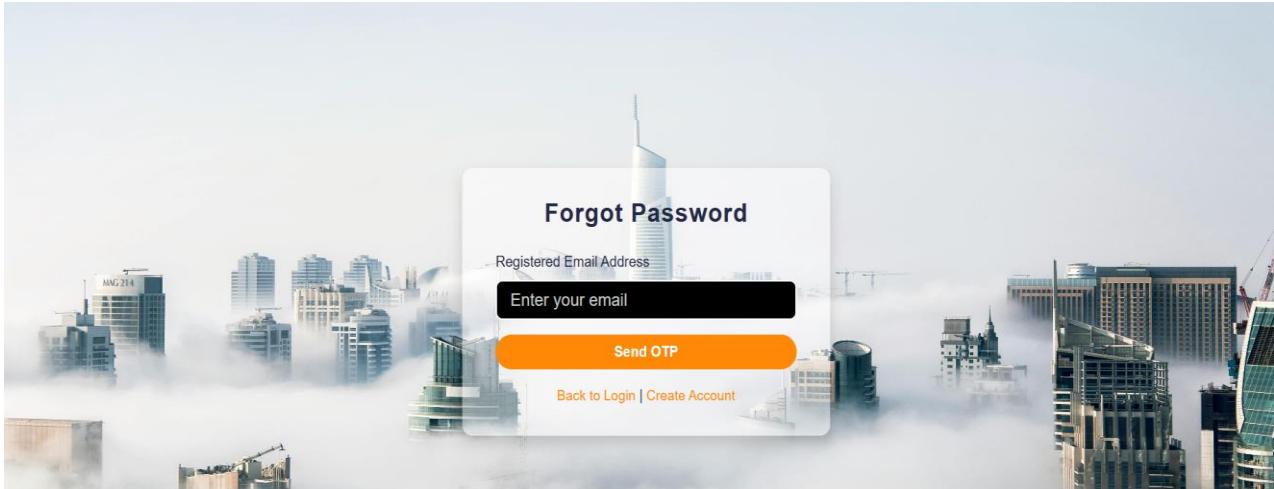
Validates the posted email (presence + format) and checks for a user row where `email_verified = 0` and `status = 'pending'` using a prepared SELECT.

If found, generates a new 6-digit OTP and expiry (+10 minutes), updates the user row with a prepared UPDATE, and sends the OTP via email (PHP mail()).

Stores the pending email in the session and redirects the user to the verification page so they can enter the new OTP.

## FORGOT PASSWORD:

Enter Email: -



Page Description: -

Short description: Request page that accepts a registered email, generates a 6-digit OTP, stores it with a 10-minute expiry for that user, emails the OTP, and redirects to the reset page.

Code Explanation: -

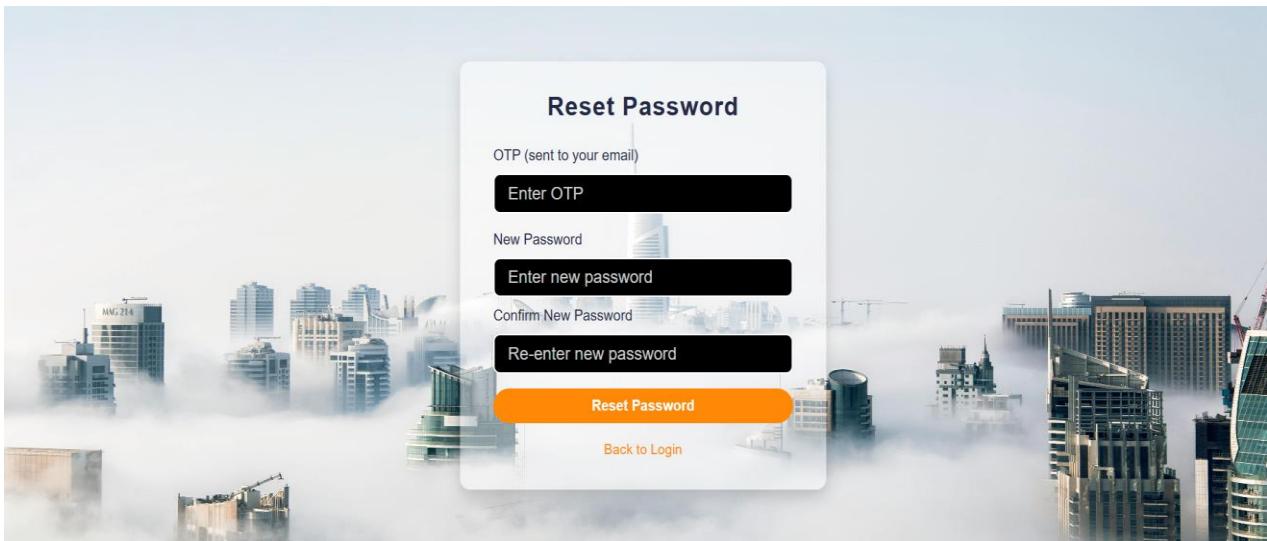
Validates the posted email and verifies the account exists, is active and `email_verified` via a prepared SELECT.

Generates a 6-digit OTP and expiry timestamp (+10 minutes), then updates the user row with `reset_password_otp` and `reset_password_otp_expiry` using a prepared UPDATE.

Sends the OTP email (`mail()`) and stores `reset_email` in session, then redirects user to `password_reset.php` (with email in the URL).

Returns clear error messages if the email is missing, invalid or not found; uses prepared statements for DB safety.

## Verify OTP And Set New and Confirm Password: -



### Page Description: -

Page where a user enters the OTP and a new password; it validates the OTP and password strength, then updates the account password if the OTP is valid and unexpired.

### Code Explanation: -

Loads the target email from session (or GET fallback) and enforces form validation for OTP format and password strength/match.

Uses a prepared SELECT to find a user row matching uemail and reset\_password\_otp; if found it checks reset\_password\_otp\_expiry is still valid.

If OTP is valid, hashes the new password with password\_hash(), updates the user row to set the new password and clear the reset OTP fields, unsets the reset\_email session key, and redirects to login (or shows success).

Errors are shown for expired OTP, invalid OTP/email, validation failures or update failures; uses prepared statements and password hashing for safety.

## Home Page: -

The screenshot displays the homepage of the FindBrick Smart Real Estate website. At the top, there is a navigation bar with links for Home, Property, Contact, Our Agents, and My Account. Below the navigation is a large banner featuring a handshake over a desk with a laptop, with the FindBrick logo and tagline "Your Smart Real Estate Solution". A search bar is located below the banner. The main content area is titled "What we Do" and includes four categories: Buy Property, Sell Property, Rent Property, and Invest Property, each with a brief description. Below this is a section titled "Recently Added Properties" showing a large image of a modern living room and a grid of smaller images for "Popular Properties". A dark callout box titled "Why Choose Us" highlights four benefits: Top Rated, Wide Range of Properties, Trusted by Thousands, and 24/7 Customer Support. The bottom section features a testimonial quote and a footer with links to quick navigation, helpful resources, and contact information.

**FindBrick**

FINDBRICK  
Your Smart Real Estate Solution

Search agent, builder or property...

**What we Do**

**Buy Property**  
Find your dream home with our expert agents.

**Sell Property**  
Get the best value for your property with our expert guidance.

**Rent Property**  
Find the perfect rental property with our extensive listings.

**Invest Property**  
Maximize your returns with our expert investment advice.

**Recently Added Properties**

**test1**  
Surat, Gujarat

**Popular Properties**

Mannat, Hilltop Haven, test1, Townville, Windmere, No Wake Zone, Maple Grove, Rosehill

**Why Choose Us**

**Top Rated**  
Our properties are highly rated by clients.

**Wide Range of Properties**  
We offer a diverse selection of properties to suit every need.

**Trusted by Thousands**  
Join our community of satisfied clients.

**24/7 Customer Support**  
Our team is here to assist you at any time.

Discover the difference with our dedicated team and personalized services.

**Testimonials**

"Absolutely the best property service I've ever used!"  
- Some Quotes

**QUICK LINKS**

- Home
- About
- Properties
- Contact

**HELPFUL**

- FAQs
- Terms & Conditions
- Privacy Policy
- Support

**CONTACT**

- 123 Main Street, Metro City
- findbrick26@gmail.com
- +91 98765 43210
- [Facebook](#) [Twitter](#) [Instagram](#) [LinkedIn](#)

© 2025 FindBrick Smart Real-Estate. All Rights Reserved.

back to top

## Page Description: -

Homepage with hero search, key services (Buy/Sell/Rent/Invest), recently added carousel, popular properties and trust/benefit section.

## Code Explanation: -

This file renders the public homepage with a hero video/banner, search box, carousels and popular property gallery.

It starts/resumes the session, includes header/footer, and queries the database for recent and random properties to populate the carousel and gallery.

The mainSearchInput uses an AJAX handler (ajax/search-handler.php) and requires the user to be logged in before performing searches.

Frontend behaviours (debounce, dropdown, carousel) are handled with jQuery and Bootstrap.

## Property Page:-

**PROPERTY GRID**

Home Property Contact Our Agents My Account ▾

Type: All For: All City: All Bedroom: Any Bathroom: Any  
Min Price: Min Max Price: Max

Apply Filter Reset

No Wake Zone (11700sqft) | Paritosh Building, 101, Usmanpura, Surat | 2 Bed, 2 Bath | Office | View Details

Drake (12000sqft) | Domoria Pul, Domoria Pul, Near Domoria Pul, Ludhiana | 4 Bed, 4 Bath | House | View Details

Mannat (12400sqft) | Paritosh Building, 101, Usmanpura, Surat | 2 Bed, 2 Bath | Apartment | View Details

Eagle's Rest (13000sqft) | Giddangi St, Chirra Bazar, Surat | 4 Bed, 3 Bath | Flat | View Details

Hilltop Haven (14000sqft) | 44/104B, Kaloor, Kochi, kerala | 10 Bed, 10 Bath | Building | View Details

Rosehill (15000sqft) | near by main road, ludhiana | 3 Bed, 3 Bath | Flat | View Details

Windmere (16000sqft) | main road, konni | 2 Bed, 2 Bath | Apartment | View Details

Amberstone (17000sqft) | main road, navi mumbai | 6 Bed, 6 Bath | Villa | View Details

Maple Grove (18000sqft) | main road, Surat | 3 Bed, 3 Bath | House | View Details

test1 (19000sqft) | main road, Surat | 3 Bed, 3 Bath | Flat | View Details

**QUICK LINKS**

- Home
- About
- Properties
- Contact

**HELPFUL**

- FAQs
- Terms & Conditions
- Privacy Policy
- Support

**CONTACT**

- 123 Main Street, Metro City
- findbrick26@gmail.com
- +91 98765 43210

[Facebook](#) [Twitter](#) [Instagram](#) [LinkedIn](#)

© 2025 FindBricks Smart Real-Estate. All Rights Reserved.

[back to top](#)

## Page Description: -

Filterable listing grid showing property cards with image, price/area, quick facts and “View Details” CTAs.

## Code Explanation: -

Shows a filterable property grid builds a dynamic WHERE clause from GET filters (ptype, city, price range, beds, baths).

Uses prepared statements with bound parameters to safely query matching properties and fetches distinct lists for filter dropdowns.

Contains input validation and exception handling to surface filter errors, then renders a responsive card-based grid.

Outputs property details and a "View Details" link for each result with XSS-safe escaping.

## Property Detail Page: -

**FindBrick**

**PROPERTY DETAIL**

[Home](#) / [Property Detail](#)



**Windmere**  
For **Featured**  
**Price: 48.70lakh**  
main road

1000sqft Sqft	2 Bedroom	2 Bathroom
2 Balcony	2 Hall	2 Kitchen

**Property Summary**

BHK	1 BHK	Property Type	apartment
Floor	1 floor	Total Floor	3
City	Konni	State	Kerala

**Description**  
test

**Features**

- Property Age: 10 Years
- Swimming Pool : Yes
- Parking : Yes
- GYM : Yes
- Type : Apartment
- Security : Yes
- Living Capacity : 10 People
- Church/Temple : No
- 3rd Party : No
- Elevator : Yes
- CCTV : Yes
- Water Supply : Ground Water / Tank

**Floor Plans**

Floor Plan



Basement Floor  
Ground Floor

**Instalment Calculator**

**Calculate Instalment** **Buy through this agent**



Your trusted partner for buying, selling, and renting properties. We bring you the best deals from across the city.

© 2025 FindBricks Smart Real-Estate. All Rights Reserved.

**QUICK LINKS**

- Home
- About
- Properties
- Contact

**HELPFUL**

- FAQs
- Terms & Conditions
- Privacy Policy
- Support

**CONTACT**

- 123 Main Street, Metro City
- findbrick26@gmail.com
- +91 98765 43210

[back to top](#)

## Page Description: -

A single-property listing page with large hero photo, price and quick facts (sqft, rooms), features, floor plans, instalment calculator and agent contact to request a visit.

## Code Explanation: -

Loads a single property by pid, displays images, summary, features, floorplans and an instalment form.

Checks availability via the record table, and when a buyer clicks buy it inserts a transaction record, updates property status and owner inside a DB transaction (commit/rollback).

Sends a notification email to the agent and displays recent properties/agent contact info fetched with prepared statements.

Escapes outputs and uses prepared queries; transaction logic ensures consistency on purchase.

## Contact Page: -

The screenshot shows the contact page of the FindBrick website. At the top, there's a dark header bar with the "FindBrick" logo on the left and navigation links for "Home", "Property", "Contact", "Our Agents", and "My Account" on the right. Below the header is a large banner image of a modern house at night with the word "CONTACT" overlaid. To the left of the main content area is a dark sidebar containing "Contact Info" and address details. To the right is a white form titled "Get In Touch" with fields for name, email, phone, subject, and comments, along with a "Send Message" button. The footer is black and contains the company logo, a brief description, quick links, helpful links, contact information, and social media icons.

**Contact Info**

Address: SPS College, Atkot Ring Road, opp. Bhoomi Ginning, Jasdan, Gujarat 360040

Phone: 1010101010

Email: findbrick26@gmail.com

**Get In Touch**

Your Name\* Email Address\*

Phone\* Subject\*

Type Comments...

Send Message

**QUICK LINKS**

Home About Properties Contact

**HELPFUL**

FAQs Terms & Conditions Privacy Policy Support

**CONTACT**

123 Main Street, Metro City  
findbrick26@gmail.com  
+91 98765 43210

f t i n

© 2025 FindBricks Smart Real-Estate. All Rights Reserved.

back to top

### Page Description: -

Admin-protected contact page where visitors submit name, email, phone, subject and message. Shows site contact info and embedded map; saves submissions to the contact table.

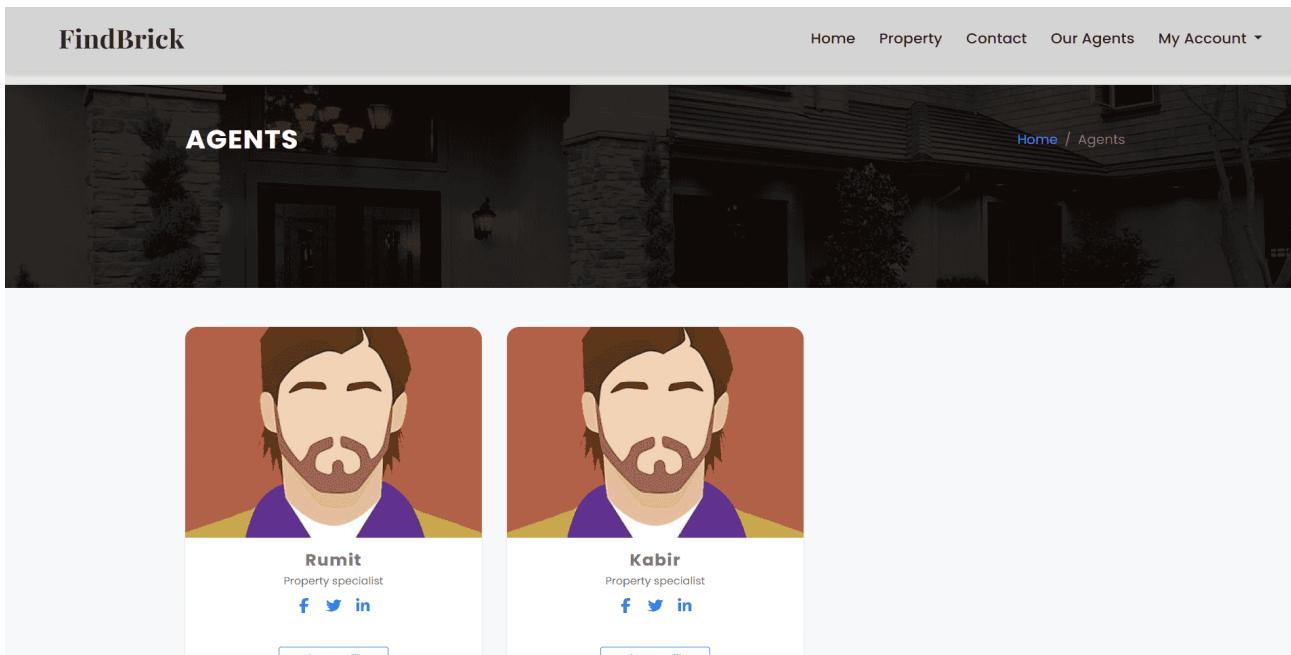
### Code Explanation: -

Checks session (session-check.php) and handles POST when the send button is used. Validates required fields, prepares a parameterized INSERT into contact, and executes it.

On success shows a success alert; on failure shows an error message including DB error when prepare fails.

Uses prepared statements for safety and echoes messages into the form UI.

## Our Agent Page: -



The screenshot shows the 'AGENTS' section of the website. At the top, there's a navigation bar with links for Home, Property, Contact, Our Agents, and My Account. Below the navigation is a banner featuring a house and the word 'AGENTS'. The main content area displays two agent profiles side-by-side:

- Rumit**: Property specialist. Social media links: Facebook, Twitter, LinkedIn. 'View Profile' button.
- Kabir**: Property specialist. Social media links: Facebook, Twitter, LinkedIn. 'View Profile' button.

At the bottom of the page, there's a footer with the FindBrick logo, a tagline 'Your trusted partner for buying, selling, and renting properties. We bring you the best deals from across the city.', quick links (Home, About, Properties, Contact), helpful links (FAQs, Terms & Conditions, Privacy Policy, Support), contact information (Address: 123 Main Street, Metro City, Email: findbrick26@gmail.com, Phone: +91 98765 43210), and social media links.

## Page Description: -

Grid directory of agents with avatar, role and social links plus a "View Profile" CTA for each card to open detailed agent pages.

## Code Explanation: -

Lists agents with pagination counts total agents, computes pages, and fetches a page of agent rows via a prepared LIMIT/OFFSET query.

Renders responsive agent cards with avatar, specialty, city and social links, and provides "View Profile" buttons to agentdetail.php.

Generates pagination links preserving other GET parameters and ensures HTML has escaped to avoid XSS.

## Agent Detail Page: -

**FindBrick**

**AGENT DETAILS**

Home / Agents / Agent Detail



**Rumit**  
Property specialist

Call: 1111111111  
Email: rumitsolanki09@gmail.com

Joined: October 2025  
agent: Rumit

**About**

No any detail about this agent.

**Listed Properties**



**Hilltop Haven**  
guwahati, Assam  
₹8 cr



**Rosehill**  
ludhiana, Punjab  
₹38 k



**Windmere**  
konnai, Kerala  
₹49 lakh



**Amberstone**  
navi mumbai, Maharashtra  
₹3 cr



**Maple Grove**  
Surat, Gujarat  
₹34 lakh

**Customer Feedback**

Aug 31, 2025  
Great experience with the agent, found my dream home!

Aug 31, 2025  
Agent was very helpful and responsive during my search.



Your trusted partner for buying, selling, and renting properties.  
We bring you the best deals from across the city.

**QUICK LINKS**

- [Home](#)
- [About](#)
- [Properties](#)
- [Contact](#)

**HELPFUL**

- [FAQs](#)
- [Terms & Conditions](#)
- [Privacy Policy](#)
- [Support](#)

**CONTACT**

- [123 Main Street, Metro City](#)
- [findbrick26@gmail.com](#)
- [+91 98765 43210](#)

© 2025 FindBricks Smart Real-Estate. All Rights Reserved.

[back to top](#)

## Page Description: -

Agent profile with photo, contact & social links, client feedback/testimonials and a gallery of the agent's listed properties to build trust and drive enquiries.

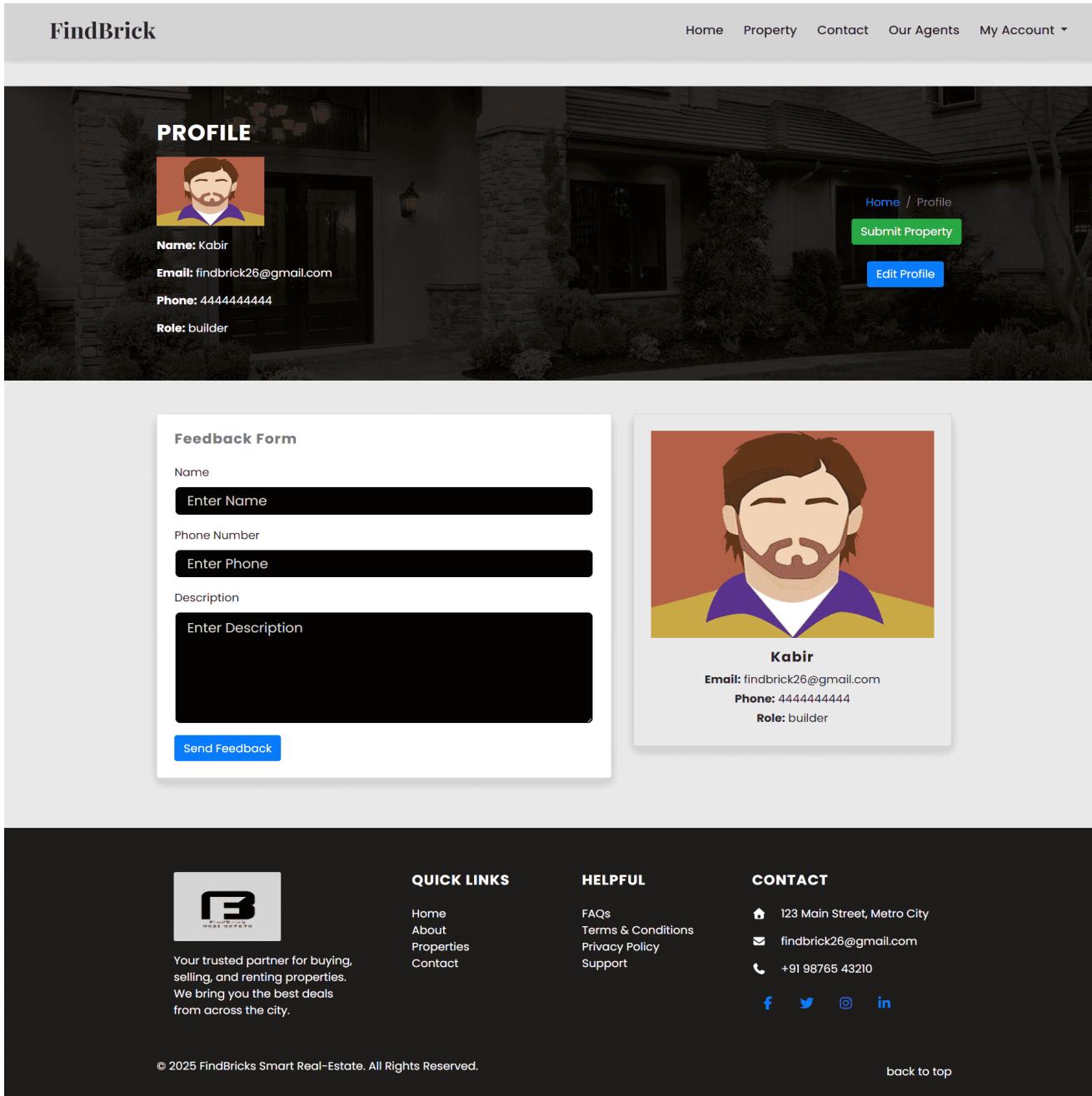
## Code Explanation: -

Displays a single agent's profile (joined with user data), social links, bio, and a feedback snippet.

Fetches and lists up to six properties belonging to that agent and recent approved feedback entries; handles the "agent not found" case.

All DB access uses prepared queries (and reasonable fallbacks), and outputs escaped to prevent XSS.

## Profile Page: -



The screenshot shows the profile page for a user named Kabir. At the top, there's a navigation bar with links for Home, Property, Contact, Our Agents, and My Account. Below the navigation is a header section with a profile picture of Kabir, his name, email, phone number, and role. To the right, there are buttons for Home / Profile, Submit Property, and Edit Profile. The main content area contains a feedback form with fields for Name, Phone Number, and Description, followed by a Send Feedback button. To the right of the form is a card displaying Kabir's profile picture, name (Kabir), email (findbrick26@gmail.com), phone number (4444444444), and role (builder). At the bottom of the page is a footer with the FindBrick logo, a tagline about being a trusted partner for buying, selling, and renting properties, and links to Quick Links (Home, About, Properties, Contact), Helpful (FAQs, Terms & Conditions, Privacy Policy, Support), and Contact information (Address: 123 Main Street, Metro City, Email: findbrick26@gmail.com, Phone: +91 98765 43210). There are also social media icons for Facebook, Twitter, Instagram, and LinkedIn.

## Page Description: -

User account page showing avatar, contact info and role, with quick actions (Submit Property, Edit Profile) and a feedback form to receive messages.

## Code Explanation: -

Displays the logged-in user's profile information, image and role after checking the session.

Fetches user data with a prepared SELECT and shows action buttons (Submit Property, Edit Profile) based on role.

Contains a feedback form that inserts a new feedback row via a prepared INSERT and returns success/error messages.

Renders a sidebar card with recent properties and safe HTML escaping for outputs.

## Submit Property Page: -

**FindBrick**

[Home](#) [Property](#) [Contact](#) [Our Agents](#) [My Account](#)

### Submit New Property

---

**Basic Information**

Property Title *	Selling Type *
<input type="text" value="Enter Title"/>	<input type="text" value="Select Status"/>
Property Type *	Price *
<input type="text" value="Select Type"/>	<input type="text" value="Prk"/> <input type="text" value="Enter Price"/>

**Property Details**

Bedrooms *	Bathrooms *	Kitchen *
<input type="text" value="Enter Bedroom (Only no 1 to 10)"/>	<input type="text" value="Enter Bathroom (Only no 1 to 10)"/>	<input type="text" value="Enter Kitchen (Only no 1 to 10)"/>
BHK *	Balcony *	Hall *
<input type="text" value="Select BHK"/>	<input type="text" value="Enter Balcony (Only no 1 to 10)"/>	<input type="text" value="Enter Hall (Only no 1 to 10)"/>
Total Floor *	<input type="text" value="Enter Floor (Only no 1 to 10)"/>	

---

**Location**

Address *	State *	
<input type="text" value="Enter Address"/>	<input type="text" value="Select state"/>	
Floor *	City *	
<input type="text" value="Select Floor"/>	<input type="text" value="Select city"/>	
Area Size (sqft) *	<input type="text" value="Enter Area Size (in sqft)"/>	

---

**Features**

\* Important Please Do Not Remove Below Content Only Change Yes Or No or Details and Do Not Add More Details

File ~ Edit ~ View ~ Insert ~ Format ~ Tools ~ Table ~

- Property Age : 10 Years
- Indoor Security : Yes
- Parking : Yes
- Gated Community : No
- Type : Apartment
- Security : Yes
- Distance from Market : 10 People
- Church/Temple : No
- 3rd Party : No
- Basement : Yes
- CCTV : Yes
- Water Supply : Ground Water / Tank

34 WORDS POWERED BY TINY

---

**Image & Status**

Main Image *	Image 1 *	Image 2 *
<input type="file" value="Choose File"/> No file chosen	<input type="file" value="Choose File"/> No file chosen	<input type="file" value="Choose File"/> No file chosen
Image 3 *	Image 4 *	Image 5
<input type="file" value="Choose File"/> No file chosen	<input type="file" value="Choose File"/> No file chosen	<input type="file" value="Choose File"/> No file chosen
Floor Plan Image	Basement Floor Plan Image	Ground Floor Plan Image
<input type="file" value="Choose File"/> No file chosen	<input type="file" value="Choose File"/> No file chosen	<input type="file" value="Choose File"/> No file chosen

Floor Plan Image  
Basement Floor Plan Image  
Ground Floor Plan Image

---

**Description**

File ~ Edit ~ View ~ Insert ~ Format ~ Tools ~ Table ~

5 WORDS POWERED BY TINY

---



Your trusted partner for buying, selling, and renting properties. We bring you the best deals from across the city.

**QUICK LINKS**

[Home](#) [About](#) [Properties](#) [Contact](#)

**HELPFUL**

[FAQs](#) [Terms & Conditions](#) [Privacy Policy](#) [Support](#)

**CONTACT**

123 Main Street, Metro City  
[findbrick26@gmail.com](mailto:findbrick26@gmail.com)  
+91 98765 43210

[f](#) [t](#) [g](#) [in](#)

© 2025 FindBricks Smart Real-Estate. All Rights Reserved.

[back to top](#)

## Page Description: -

Agent & Builder-facing submission form to add a listing — title, price/type, rooms, location, features, images/floor plans and description.

## Code Explanation: -

Presents a protected form for agents/builders to submit a new property and handles server-side validation of all fields.

Validates and securely processes multiple image uploads (allowed types, size limits), optimizes images, and stores them under admin/uploads/properties.

Inserts the new property using a prepared INSERT and links the current user/agent; returns success or detailed error messages.

Keeps helper functions for upload/optimization and sanitizes inputs before DB insertion.

## Profile Edit Page: -

FindBrick

Home Property Contact Our Agents My Account ▾

### Edit Profile



Choose File No file chosen

Name \*

Kabir

Phone \*

4444444444

City \*

Bio

Speciality

Save Changes



Your trusted partner for buying, selling, and renting properties. We bring you the best deals from across the city.

© 2025 FindBricks Smart Real-Estate. All Rights Reserved.

**QUICK LINKS**

- Home
- About
- Properties
- Contact

**HELPFUL**

- FAQs
- Terms & Conditions
- Privacy Policy
- Support

**CONTACT**

- 123 Main Street, Metro City
- findbrick26@gmail.com
- +91 98765 43210

[f](#) [t](#) [i](#) [in](#)

[back to top](#)

## Page Description: -

Simple profile edit form to update photo, name, phone, city, bio and specialty with “Save Changes”.

## Code Explanation: -

Provides a profile edit form for users and, if applicable, agent/builder fields; it loads current user/agent rows first.

Handles optional image upload with validation, saves files to uploads/profile, and updates both user and agent tables using prepared statements.

Uses server-side validation and redirects after successful updates; includes client-side preview for uploaded avatar.

Employs htmlspecialchars when printing values to protect against XSS.

## Agent Property Page: -

**FindBrick**

Home   Property   Contact   Our Agents   My Account ▾

**MY PROPERTIES**

Home / My Properties

Search your properties...

 <p><b>Hilltop Haven</b> ₹8.20 / cr</p> <p>guwahati, Assam • Property Age : 10 Yea...</p> <p><a href="#">Delete</a> <a href="#">Update</a></p>	 <p><b>Rosehill</b> ₹38.00 / k</p> <p>ludhiana, Punjab • Property Age : 10 Yea...</p> <p><a href="#">Delete</a> <a href="#">Update</a></p>	 <p><b>Windmere</b> ₹48.70 / lakh</p> <p>konni, Kerala • Property Age : 10 Yea...</p> <p><a href="#">Delete</a> <a href="#">Update</a></p>
 <p><b>Amberstone</b> ₹3.42 / cr</p> <p>navi mumbai, Maharashtra • Property Age : 10 Yea...</p> <p><a href="#">Delete</a> <a href="#">Update</a></p>	 <p><b>Maple Grove</b> ₹34.20 / lakh</p> <p>Surat, Gujarat • Property Age : 10 Yea...</p> <p><a href="#">Delete</a> <a href="#">Update</a></p>	

**QUICK LINKS**

- [Home](#)
- [About](#)
- [Properties](#)
- [Contact](#)

**HELPFUL**

- [FAQs](#)
- [Terms & Conditions](#)
- [Privacy Policy](#)
- [Support](#)

**CONTACT**

- 123 Main Street, Metro City
- findbrick26@gmail.com
- +91 98765 43210

© 2025 FindBricks Smart Real-Estate. All Rights Reserved.

[back to top](#)

## Page Description: -

"My Properties" listing page where a logged-in user can search, view and delete their properties. Content is loaded via AJAX (feature-list.php) and supports client-side search and pagination.

## Code Explanation: -

Requires session, includes header, and provides a search input and container for AJAX-loaded property cards.

JavaScript calls ajax/feature-list.php to fetch JSON with HTML snippets and pagination, handles parsing and error cases.

Binds delete handlers to confirm and navigate to propertydelete.php for removals; debounces search input for UX.

Keeps UI responsive by showing loading spinner and handling invalid or failed AJAX responses.

## Update Property Page: -

**FindBrick**

[Home](#) [Property](#) [Contact](#) [Our Agents](#) [My Account](#)

### Update Property

**Basic Information**

Property Title *	Selling Type *
Hilltop Haven	Sale
Property Type *	Price *
Building	Cr. 8.20

  
**Property Details**

Bedrooms *	Bathrooms *	Kitchen *
10	10	5
BHK *	Balcony *	Hall *
3 BHK	10	3
Total Floor *		
10		

  
**Location**

Address *	State *
main road	Assam
Floor *	City *
5 floor	guwahati
Area Size (sqft) *	
1200sqft	

  
**Features**

\* Important Please Do Not Remove Below Content Only Change Yes Or No or Details and Do Not Add More Details

File ~ Edit ~ View ~ Insert ~ Format ~ Tools ~ Table ~  
 B F I U L M T C S P A G E Z A - =

- Property Age : 10 Years
- Swimming Pool : Yes
- Parking : Yes
- Gym : Yes
- Type : Apartment
- Security : Yes
- Garden / Temple : 10 People
- Church/Temple : No
- 3rd Party : No
- Elevation : Yes
- Connectivity : Yes
- Water Supply : Ground Water / Tank

54 WORDS POWERED BY TINY

  
**Image & Status**

Main Image *	Image 1 *	Image 2 *
<input type="button" value="Choose File"/> No file chosen	<input type="button" value="Choose File"/> No file chosen	<input type="button" value="Choose File"/> No file chosen
Image 3 *	Image 4 *	Image 5
<input type="button" value="Choose File"/> No file chosen	<input type="button" value="Choose File"/> No file chosen	<input type="button" value="Choose File"/> No file chosen
Floor Plan Image	Basement Floor Plan Image	Ground Floor Plan Image
<input type="button" value="Choose File"/> No file chosen	<input type="button" value="Choose File"/> No file chosen	<input type="button" value="Choose File"/> No file chosen
Status *	Is Featured? *	
Available	Yes	

  
**Description**

File ~ Edit ~ View ~ Insert ~ Format ~ Tools ~ Table ~  
 B F I U L M T C S P A G E Z A - =

test

1 WORDS POWERED BY TINY



Your trusted partner for buying, selling, and renting properties. We bring you the best deals from across the city.

**QUICK LINKS**

[Home](#) [About](#) [Properties](#) [Contact](#)

**HELPFUL**

[FAQs](#) [Terms & Conditions](#) [Privacy Policy](#) [Support](#)

**CONTACT**

123 Main Street, Metro City  
 findbrick26@gmail.com  
 +91 98765 43210

[Facebook](#) [Twitter](#) [Instagram](#) [LinkedIn](#)

© 2025 FindBricks Smart Real-Estate. All Rights Reserved.

[back to top](#)

## Page Description: -

Authenticated page for owners/agents to edit an existing property listing and upload images. Preloads property data and provides form sections for basic info, details, location, features and images.

## Code Explanation: -

Loads the property by pid with a prepared SELECT and aborts with a friendly message if not found.

Validates submitted fields (lengths, numeric ranges, required selections) and accumulates errors.

Processes multiple image uploads with processUpload() (type/size checks, optional optimizeImage resizing) and preserves old paths when uploads are skipped.

Uses a prepared UPDATE to save all fields (including image paths) and redirects to your\_property.php on success; errors are presented to the user.

## About Page: -

**FindBrick**

Home Property Contact Our Agents My Account ▾

**ABOUT US**

[Home](#) / [About Us](#)

**Our Company**

FindBricks is committed to providing the best real estate experience for buyers, sellers, and renters. We offer a wide range of properties and dedicated support for all your needs.



**our broker**

Best brokering, and future property for our client




**QUICK LINKS**

- [Home](#)
- [About](#)
- [Properties](#)
- [Contact](#)

**HELPFUL**

- [FAQs](#)
- [Terms & Conditions](#)
- [Privacy Policy](#)
- [Support](#)

**CONTACT**

- 123 Main Street, Metro City
- findbrick26@gmail.com
- +91 98765 43210

© 2025 FindBricks Smart Real-Estate. All Rights Reserved. [back to top](#)

## Page Description: -

Public "About Us" page that displays one or more About entries from the database with title, descriptive text and an image.

Uses the site header/footer and a responsive two-column layout (text left, image right) with breadcrumb navigation.

## Code Explanation: -

Requires session-check, then runs \$conn->query("SELECT \* FROM about") and loops over results with fetch\_assoc() to render each record.

Prints the title using htmlspecialchars and outputs the description via nl2br(\$row['description']) so line breaks are preserved.

Renders the image using src="admin/uploads/about/{image}" with htmlspecialchars for the filename and shows a warning if the query fails.

## FAQs Page: -

FindBrick

Home Property Contact Our Agents My Account ▾



# Frequently Asked Questions

Find answers to common questions about buying, selling, and renting properties on FindBrick.

How do I search for properties on FindBrick?

Can I contact property owners or agents directly?

Is it free to list my property on FindBrick?

How do I get support if I face any issues?

Are my details and data safe on FindBrick?

Need more help? Contact Support



Your trusted partner for buying, selling, and renting properties. We bring you the best deals from across the city.

© 2025 FindBricks Smart Real-Estate. All Rights Reserved.

QUICK LINKS

- Home
- About
- Properties
- Contact

HELPFUL

- FAQs
- Terms & Conditions
- Privacy Policy
- Support

CONTACT

- 123 Main Street, Metro City
- findbrick26@gmail.com
- +91 98765 43210

[f](#) [t](#) [i](#) [in](#)

## Page Description: -

Public FAQ page with an accordion of common questions and answers. Links to support and policy pages for further help.

## Code Explanation: -

Includes shared header/footer and static HTML for an accordion of FAQ entries. Uses Bootstrap collapse components to show/hide answers and client-side scripts for layout.

No database interaction; returns static content and navigation to support/privacy pages. Safe, simple presentation-focused page for end-user reference.

## Terms & Condition Page: -

The screenshot shows the FindBrick website's header with "FindBrick" and a navigation bar with links to Home, Property, Contact, Our Agents, and My Account. Below the header is a large card titled "TERMS & CONDITIONS". Inside the card, there is a section titled "Welcome to FindBrick!" followed by a paragraph of text and a bulleted list of terms and conditions. At the bottom of the card, it says "For full details, please contact our support team." The footer contains the FindBrick logo, a tagline "Your trusted partner for buying, selling, and renting properties. We bring you the best deals from across the city.", quick links (Home, About, Properties, Contact), helpful links (FAQs, Terms & Conditions, Privacy Policy, Support), contact information (address: 123 Main Street, Metro City, email: findbrick26@gmail.com, phone: +91 98765 43210), and social media icons for Facebook, Twitter, Instagram, and LinkedIn.

## Page Description: -

Terms & Conditions page that states rules for using the FindBrick website. Provides summary bullets and links to the privacy policy for legal context.

## Code Explanation: -

Static informational page included inside the site header/footer for consistent layout. No server-side logic or data operations; content is presented in a card for readability.

Useful for compliance and user agreement reference.

## Privacy Policy: -

The screenshot shows the FindBrick website with a navigation bar at the top. The main content area is titled "Privacy Policy". A section titled "Your Privacy Matters" contains a paragraph about data protection and a bulleted list of privacy practices. Below this is a note about support and a "Support page". The footer contains the company logo, a tagline, links to quick links and helpful pages, contact information, and social media icons.

**FindBrick**

Home Property Contact Our Agents My Account ▾

**Privacy Policy**

**Your Privacy Matters**

FindBrick is committed to protecting your privacy. This policy explains how we use your data:

- We collect personal information only when you register, list a property, or contact us.
- Your information is used to provide services and communicate with you regarding your account or property listings.
- We do not share your personal data with third parties except as required by law or to provide our services.
- Cookies may be used to enhance your browsing experience.
- You can request to delete your data by contacting our support team.

If you have any questions, please visit our [Support page](#).

**QUICK LINKS**

Home About Properties Contact

**HELPFUL**

FAQs Terms & Conditions Privacy Policy Support

**CONTACT**

123 Main Street, Metro City  
findbrick26@gmail.com  
+91 98765 43210

[f](#) [t](#) [i](#) [in](#)

© 2025 FindBricks Smart Real-Estate. All Rights Reserved. [back to top](#)

## Page Description: -

Site privacy policy page describing how user data is collected and used. Provides contact link to support for privacy-related requests.

## Code Explanation: -

Includes header/footer and renders static policy text inside a styled card. No DB or form handling; purely informational content meant for legal/compliance display.

Keeps consistent site layout and links to support for data deletion or queries.

## Support Page: -

The screenshot shows a support contact form titled "Support & Contact Us". The form includes fields for Name, Email, and Message, each with a placeholder text ("Your Name", "Your Email", "Your Message..."). A "Send Message" button is located below the message field. Below the form, there is a section titled "Contact Details" with an email address: support@findbrick.com.

The footer contains the company logo, a tagline ("Your trusted partner for buying, selling, and renting properties. We bring you the best deals from across the city."), quick links (Home, About, Properties, Contact), helpful links (FAQs, Terms & Conditions, Privacy Policy, Support), contact information (address: 123 Main Street, Metro City, email: findbrick26@gmail.com, phone: +91 98765 43210), and social media icons for Facebook, Twitter, Instagram, and LinkedIn. Copyright information at the bottom left reads "© 2025 FindBricks Smart Real-Estate. All Rights Reserved.", and a "back to top" link is at the bottom right.

## Page Description: -

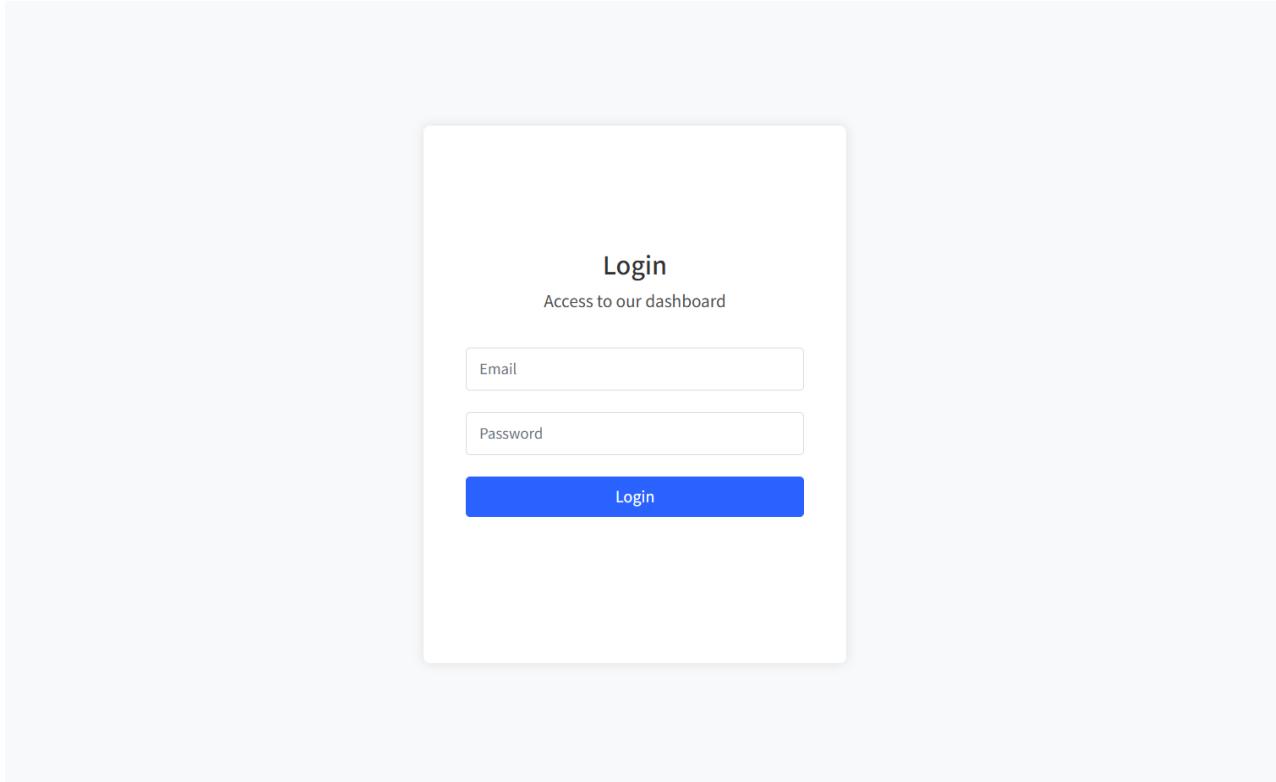
Support contact page with a form that posts to send-support.php and shows support contact details. Encourages users to submit issues or messages for help.

## Code Explanation: -

Renders the support form (name, email, message) and posts to send-support.php. Includes static contact details and uses standard Bootstrap form markup.

## ADMIN-SIDE SNAPSHOT:

### Admin Login Page: -



### Page Description: -

Admin login page that authenticates admin users and starts an admin session on success. Validates credentials against the admin table and redirects to the dashboard when authenticated.

### Code Explanation: -

Accepts POSTED email/password, queries the admin table using a SQL string (no prepared statements), checks row count and sets \$\_SESSION['auser'] on match then redirects to dashboard.

Current code uses plaintext password comparison and direct SQL concatenation — strongly recommend using password hashing and prepared statements to prevent security risks.

## Admin Dashboard Page: -

**Admin Panel**

- MAIN
  - [Dashboard](#)
- PROPERTIES
  - [Properties](#)
- LOCATION
  - [Location](#)
- USERS & AGENTS
  - [Users](#)
- LEADS
  - [Inquiries](#)
- CONTENT
  - [About Us](#)
- ACCOUNT
  - [Authentication](#)

© 2025 FindBrickReal Estate Admin

**Dashboard**

**Total Properties**  
11  
All property listings

**Available**  
10  
Currently for sale/rent

**Sold**  
1  
Already sold

**Property Status**

Status	Count
Available	10
Sold	1

**Monthly Sales**

Month	Sold Properties
Jan	2.0
Feb	3.0
Mar	4.0
Apr	1.0

**Admin Panel**

- MAIN
  - [Dashboard](#)
- PROPERTIES
  - [Properties](#)
- LOCATION
  - [Location](#)
- USERS & AGENTS
  - [Users](#)
- LEADS
  - [Inquiries](#)
- CONTENT
  - [About Us](#)
- ACCOUNT
  - [Authentication](#)

© 2025 FindBrickReal Estate Admin

**Property Status Overview**

Status	Count
Available	10
Sold	1

**Top Agents**

Agent	Email	Phone	Properties
Kabir	findbrick26@gmail.com	4444444444	6 Properties
Rumit	rumitsolanki09@gmail.com	1111111111	5 Properties

**Recent Properties**

ID	Property Name	Type	Status	Price	Action
16	No Wake Zone	office	Available	5.00 lakh	<a href="#">Edit</a>
15	Drake	house	Available	54.90 lakh	<a href="#">Edit</a>
14	Townville	apartment	Available	72.00 lakh	<a href="#">Edit</a>
13	Mannat	villa	Available	5.23 cr	<a href="#">Edit</a>
12	Eagle's Rest	flat	Available	25.40 lakh	<a href="#">Edit</a>

Developed By: Rumit Solanki & Khushali Mayatra

Page 78

## Page Description: -

Admin dashboard summarizing property counts, availability, recent properties, charts and top agents.

Runs aggregated queries for statistics, prepares data for Chart.js, and displays recent property rows and top agents dynamically.

## Code Explanation: -

Requires admin session, queries feedback joined to user, and prints name, email, feedback text and status in the table with Edit/Delete links.

Uses mysqli\_fetch\_array for row access — escape values when outputting and consider filtering or sanitizing long feedback content.

## Admin Property Page: -

**Admin Panel**

- MAIN**
- [Dashboard](#)
- PROPERTIES**
- [Properties](#)
- LOCATION**
- [Location](#)
- USERS & AGENTS**
- [Users](#)
- LEADS**
- [Inquiries](#)
- CONTENT**
- [About Us](#)
- ACCOUNT**

**Property View**

Show  
10 
Search:

ID	Title	Type	Price	City	Status	Images	Actions
6	test1	flat	25.30 lakh	Surat	<span>Available</span>	    	 
7	Maple Grove	house	34.20 lakh	Surat	<span>Available</span>	    	 
8	Amberstone	villa	3.42 cr	navi mumbai	<span>Available</span>	    	 
9	Windmere	apartment	48.70 lakh	konni	<span>Sold</span>	    	 
10	Rosehill	flat	38.00 k	ludhiana	<span>Available</span>	    	 
11	Hilltop Haven	building	8.20 cr	guwahati	<span>Available</span>	    	 
12	Eagle's Rest	flat	25.40 lakh	Surat	<span>Available</span>	    	 
13	Mannat	villa	5.23 cr	guwahati	<span>Available</span>	    	 
14	Townville	apartment	72.00 lakh	konni	<span>Available</span>	    	 
15	Drake	house	54.90 lakh	ludhiana	<span>Available</span>	    	 

Showing 1 to 10 of 11 entries

[<>](#) [1](#) [2](#) [<>>](#)

## Page Description: -

Admin view that lists all properties with thumbnails, status badges and action buttons (edit/delete).

Collects property image fields, displays summaries in a table, and provides management controls per property.

## Code Explanation: -

Checks admin session, selects all properties, assembles available image fields into an array for display, and outputs each property row with status badge and action buttons.

Escapes outputs with htmlspecialchars in some places; ensure consistent escaping and validate image paths; consider lazy loading and pagination for performance.

## State Manage Page: -

ID	State Name	Edit	Delete
5	Assam	<button>Edit</button>	<button>Delete</button>
4	Punjab	<button>Edit</button>	<button>Delete</button>
3	Kerala	<button>Edit</button>	<button>Delete</button>
2	Maharashtra	<button>Edit</button>	<button>Delete</button>
1	Gujrat	<button>Edit</button>	<button>Delete</button>

## Page Description: -

Admin UI to add and list states with edit/delete options. Inserts new state names into the state table and renders the state list for management.

## Code Explanation: -

Verifies admin session, inserts new state via a simple INSERT query on POST and then lists states from the state table.

Show's success/error feedback.

Improve by using prepared statements and form validation to avoid accidental duplicates or SQL injection.

## City Manage Page: -

**Admin Panel**

**MAIN**

- Dashboard**

**PROPERTIES**

- Properties**

**LOCATION**

- Location**
- Manage States**
- Manage Cities** (selected)

**USERS & AGENTS**

- Users**
- Inquiries**

**CONTENT**

**State**

Dashboard / State

**Add City**

State Name: -- Select State --

City Name: Enter city name

**Submit**

**City List**

Show	Search:			
10				
entries				
ID	City Name	State ID	State Name	Actions
1	Surat	1	Gujrat	
2	navi mumbai	2	Maharashtra	
3	konni	3	Kerala	
4	ludhiana	4	Punjab	
5	guwahati	5	Assam	

Showing 1 to 5 of 5 entries

## Page Description: -

Admin interface to add cities (linked to states) and view the city list with edit/delete actions. Inserts new city records and shows a joined city/state table for management.

## Code Explanation: -

Confirms admin session and selects all rows from contact table with mysqli\_query, then renders name, email, phone, subject and message in a table.

Each entry has a Delete link.

Sanitize output to prevent XSS when displaying messages and consider pagination for large datasets.

## Admins Mange Page: -

The screenshot shows a left sidebar with a navigation menu and a main content area titled 'Admin'.

**Left Sidebar (MAIN):**

- Dashboard (selected)
- Properties
- Location
- Users & Agents
  - Users
  - Admins (selected)
  - Clients
  - Agents
  - Builders
- Inquiries
- Content

**Main Content Area:**

**Title:** Admin

**Breadcrumbs:** Dashboard / Admin

**Section:** Admin List

ID	Name	Email	Action
1	admin1	admin1@example.com	Delete
2	admin2	admin2@example.com	Delete

## Page Description: -

Displays a simple list of admin accounts (ID, name, email) for site administrators. Fetches all rows from the admin table and shows delete controls for each entry.

## Code Explanation: -

Verifies admin session, selects all rows from admin table and prints ID, name and email in a table. Each row includes a Delete link.

Uses mysqli\_fetch\_row and direct output — ensure access control and consider pagination and prepared statements for any future operations.

## Clients Manage Page: -

ID	Name	Email	Utype	Phone	Image	Delete
1	Mihir	tivedimihir@gmail.com	user	4444444444		<a href="#">Delete</a>
2	Khushali	kakashisin26@gmail.com	user	1111111111		<a href="#">Delete</a>

## Page Description: -

Admin list of regular users with profile info, phone and image plus delete functionality. Fetches users where utype='user' and displays them for administrative maintenance.

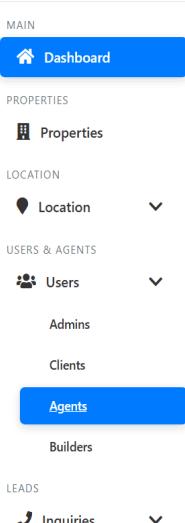
## Code Explanation: -

Checks admin session and selects users where utype='user', printing name, email, phone and profile image with a delete link.

Uses mysqli\_fetch\_row and direct output; for security escape all fields and ensure delete endpoints confirm authorization and properly remove or archive user data.

## Agent Manage Page: -

### Admin Panel



### Manage Agents

#	Name & Photo	Email	User Type	Phone	Action
1		rumitsolanki09@gmail.com	agent	11111111111	

## Page Description: -

Admin list showing users with type 'agent', their contact details and a delete action. Renders agent rows with photo, email and phone and allows admins to remove agent accounts.

## Code Explanation: -

Requires admin session, queries users where utype='agent', renders name, photo, email, phone and a delete action.

Image tag uses on error fallback to a default.

Deleting agents should trigger cascade handling of related properties — ensure delete action cleans related records safely and uses authorization checks.

## Builder Manage Page: -

ID	Name	Email	Utype	Phone	Image	Delete
1	Kabir	findbrick26@gmail.com	builder	4444444444		<a href="#">Delete</a>

## Page Description: -

Admin list showing users with type 'builder', including basic info, image and delete links. Provides a management table for builder accounts to review and remove entries.

## Code Explanation: -

Protects page by session check and fetches users with utype='builder', showing basic columns and image.

Provides delete link for each builder. As with agents, ensure deletes are safe (handle related properties) and outputs escaped.

## Contact Request Manage Page: -

The screenshot shows a left sidebar with a navigation menu and a main content area titled 'Contact'.

**Left Sidebar (MAIN):**

- Dashboard (selected)
- Properties
- Location
- Users & Agents
- Inquiries
- Contact Requests (selected)
- Client Feedback
- About Us
- Account

**Main Content Area:**

### Contact

Dashboard / Contact

#### Contact List

ID	Name	Email	Phone	Subject	Message	Delete
1	John Doe	john.doe@example.com	+1234567890	Inquiry about apartment	I would like to know more about the apartment listed on your site.	<a href="#">Delete</a>
2	Jane Smith	jane.smith@example.com	+1987654321	Site Feedback	The site is very user-friendly. Keep up the good work!	<a href="#">Delete</a>

Show: 10 entries

Search:

Showing 1 to 2 of 2 entries

## Page Description: -

Shows all contact form submissions with name, email, phone, subject and message columns.  
Allows admins to review and delete contact entries stored in the contact table.

## Code Explanation: -

Confirms admin session and selects all rows from contact table with mysqli\_query, then renders name, email, phone, subject and message in a table.

Each entry has a Delete link. Sanitize output to prevent XSS when displaying messages and consider pagination for large datasets.

## Feedback Manage Page: -

The screenshot shows a sidebar navigation on the left with categories like MAIN, PROPERTIES, LOCATION, USERS & AGENTS, LEADS, and CONTENT. Under LEADS, 'Client Feedback' is selected. The main content area is titled 'Feedback' and shows a 'Feedback List'. It includes a search bar, a dropdown for 'Show' (set to 10), and a 'Search:' input field. Below is a table with four rows of feedback data:

ID	Name	Email	Feedback	Status	Edit	Delete
1	Khushali	kakashisin26@gmail.com	test agent	0	<a href="#">Edit</a>	<a href="#">Delete</a>
2	Kabir	findbrick26@gmail.com	Great experience with the agent, found my dream home!	1	<a href="#">Edit</a>	<a href="#">Delete</a>
3	Mihir	tivedimihir@gmail.com	Website is easy to use, but property details could be clearer.	0	<a href="#">Edit</a>	<a href="#">Delete</a>
4	Rumit	rumitsolanki09@gmail.com	Agent was very helpful and responsive during my search.	1	<a href="#">Edit</a>	<a href="#">Delete</a>

At the bottom, it says 'Showing 1 to 4 of 4 entries' with navigation arrows.

## Page Description: -

Admin list of feedback messages joined with user info, showing status and controls to edit/delete.

Selects feedback and user rows and presents them in a management table for review.

## Code Explanation: -

Requires admin session, queries feedback joined to user, and prints name, email, feedback text and status in the table with Edit/Delete links.

Uses mysqli\_fetch\_array for row access — escape values when outputting and consider filtering or sanitizing long feedback content.

## Add About Info Page: -

The screenshot shows the 'About Us' addition page. The left sidebar has a 'Content' section with 'About Us' selected. The main area has a 'Title' field, an 'Image' field with a 'Choose File' button, and a 'Description' field with a rich-text editor toolbar. At the bottom is a 'Submit' button.

## Page Description: -

Admin page to add an "About Us" entry with title, image and rich-text description. Handles file upload (saves to uploads/about) and inserts the record into the about table, then shows success/error feedback.

## Code Explanation: -

Handles a POST form, saves uploaded image to uploads/about (creates folder if missing), and inserts title/description/image into the about table using mysqli\_query.

Show's success or error message after insert.

Note: currently uses direct SQL string concatenation — consider prepared statements and validating/renameing uploaded files to avoid injection and filename issues.

## Manage Abouts Page: -

The screenshot shows a sidebar navigation menu on the left and a main content area on the right. The sidebar includes sections for PROPERTIES, LOCATION, USERS & AGENTS, LEADS, and CONTENT, with 'About Us' currently selected. A 'View Info' button is also present. The main content area is titled 'View About' and shows a 'List Of About' table with two rows:

ID	Title	Description	Image	Edit	Delete
1	Our Company	FindBricks is committed to providing the best real estate experience for buyers, sellers, and renters. We offer a wide range of properties and dedicated support for all your needs.		<a href="#">Edit</a>	<a href="#">Delete</a>
2	our broker	Best brokering, and future property for our client		<a href="#">Edit</a>	<a href="#">Delete</a>

## Page Description: -

Lists all "About" entries in the admin panel with image preview and Edit/Delete links. Queries the about table and renders rows in a responsive table for management.

## Code Explanation: -

Checks admin session, queries the about table with mysqli\_query, then loops rows to render ID, title, description and image from uploads/about.

Displays Edit/Delete links per row.

Outputs are echoed directly — sanitize values when rendering and validate file existence before showing images.

## TESTING

### Test Case: -

ID	Page	Test (short)	Steps (precondition + action)	Expected results & Priority
TC-HOME-01	Home (index.php)	Page loads	Server and DB running; open site root	Home banner, search, carousel and popular properties load. Priority: High
TC-HOME-02	Home	Video fallback image	Browser blocks video or doesn't support autoplay; open page	Fallback image images/home.jpg shows. Priority: Medium
TC-HOME-03	Home search	Prevent search if not logged in	Not logged in; type "abc" in main search	Dropdown shows "pls login first" (or similar). Priority: High
TC-HOME-04	Home search	AJAX search works when logged in	Logged in and data exists; type search term and wait 400ms	Dropdown shows matching agents or properties. Priority: High
TC-HOME-05	Home carousel	Carousel shows DB items	At least one property with created_at < now; open home and click next/prev	Images and captions show and navigation works. Priority: Medium
TC-GRID-01	Property Grid	Grid loads property cards	Properties exist in DB; open property.php	Cards show image, title, location, price. Priority: High
TC-GRID-02	Property Grid	Filters work	DB has varied properties; pick filters (type/city/beds/baths/price)	Only matching properties show; if none, show "No properties". Priority: High
TC-GRID-03	Property Grid	SQL injection attempt	Endpoint reachable; set ptype to "' OR 1=1 --"	No SQL error; app does not break. Priority: High
TC-GRID-04	Property Grid	Pagination works	More properties than a page limit; open ?page=2	Page 2 shows next results; no duplicates. Priority: Medium
TC-PDETA IL-01	Property Detail	Show property details	Valid pid in URL; open propertydetail.php?pid=NN	Images, summary, features, floorplans, instalment form and agent contact show. Priority: High
TC-PDETA IL-02	Property Detail	Image carousel up to 6 images	Property has many pimage fields; open page	Carousel shows images; first image active. Priority: Medium
TC-PDETA IL-03	Instalment calculator	Submit amount	On property page; enter amount and submit to instalment.php	instalment.php receives amount and returns calculation. Priority: Medium

TC-PDETA IL-04	Property Detail	Buy flow success	Logged-in user and property available; click Buy	New record, property status updated, email to agent sent, DB committed. Priority: High
TC-PDETA IL-05	Property Detail	Buy flow rollback on error	Force DB error during buy; attempt buy	Transaction rolls back, property unchanged, user sees failure. Priority: High
TC-PDETA IL-06	Property Detail	Agent contact shows	Agent exists for property; open detail	Agent name, phone and email display correctly. Priority: Medium
TC-AJAX-01	Property Add	Load cities by state (AJAX)	state table has data; change state dropdown	City dropdown fills with expected cities. Priority: Medium
TC-LGIN-01	Login	Session set on login	Valid user exists; login with correct credentials	User redirected and session contains uid & uemail. Priority: High
TC-REGIS TER-01	Register	Basic registration works	Unique email available; fill register form	New user created. Priority: High
TC-PROPA DD-01	Submit Property	Required fields validation	Logged in as agent/builder; submit empty required fields	Show error messages; no DB insert. Priority: High
TC-PROPA DD-02	Submit Property	File upload type/size checks	Logged in; upload .exe or >2MB file	Upload rejected, show error, no DB insert. Priority: High
TC-PROPA DD-03	Submit Property	Image optimization	Upload large JPG/PNG; submit property	Image saved and optimized in admin/uploads/properties. Priority: Medium
TC-PROPA DD-04	Submit Property	DB insertion success	Valid form and images; submit	Property row inserted; uid and agent_id set; pimage paths saved without admin/ prefix. Priority: High
TC-PROPA DD-05	Submit Property	Feature HTML preserved	Provide feature HTML; submit	feature column stores HTML and displays on detail page. Priority: Medium
TC-PROPA DD-06	Submit Property	Remove admin/ prefix	Upload files and submit	DB pimage fields do not include leading "admin/". Priority: Low

TC-PROFIL-E-01	Profile	Access control	Not logged in; open profile.php	Redirect to login.php. Priority: High
TC-PROFIL-E-02	Profile	Show profile info	Logged in; open profile.php	Name, email, phone, role and image match DB. Priority: High
TC-PROFIL-E-03	Profile	Feedback insert	Logged in; submit feedback form	New feedback row inserted with uid, content and status=0. Priority: Medium
TC-PROFIL-EEDIT-01	Edit Profile	Avatar upload works	Logged in; upload jpg/png	Image saved to uploads/profile and user.uimage updated. Priority: High
TC-PROFIL-EEDIT-02	Edit Profile	Update agent table	User is agent/builder; submit agent fields	agent table updated with city, bio, speciality and image. Priority: High
TC-PROFIL-EEDIT-03	Edit Profile	Invalid image blocked	Logged in; upload .gif or .svg	Show validation error; existing image unchanged. Priority: Medium
TC-AGENT-LIST-01	Agents	Agent list pagination	More than 9 agents in DB; open agent.php?page=2	Page 2 shows other agents; pagination links work. Priority: Medium
TC-AGENT-DETAI-L-01	Agent Detail	Agent shows data	Agent exists; open agentdetail.php?id=NN	Profile, socials, listed properties and feedback show. Priority: High
TC-AGENT-DETAI-L-02	Agent Detail	Agent not found handled	No agent with given id; open page	Show "Agent not found" message; no crash. Priority: Medium
TC-FORGO-T-01	Forgot Password	Email validation & existence	Submit blank/invalid/non-existing email	Show proper validation messages; only active verified users can request OTP. Priority: High
TC-FORGO-T-02	Forgot Password	OTP generation & store	Valid user; submit registered email	OTP and expiry stored; email sent; session reset_email set; redirect to password_reset.php. Priority: High
TC-FORGO-T-03	Forgot Password	OTP email delivery	Mail system configured; trigger forgot password	Email received or mail function returns success. Priority: Medium

TC-RESET-01	Reset Password	OTP validation works	reset_email set and valid OTP stored; submit OTP + new passwords	Password hashed and updated; OTP cleared; redirect to login. Priority: High
TC-RESET-02	Reset Password	OTP expiry handled	OTP expired; submit OTP	Show "OTP expired" and require restart. Priority: High
TC-RESET-03	Reset Password	Password complexity check	Any submission with weak password	Show validation error; do not update password. Priority: High
TC-RESET-04	Reset Password	Brute-force detection	Many wrong OTP attempts	System should throttle or log attempts (if implemented). Priority: Medium
TC-SESSION-01	Session	Logout clears session	Logged in; click logout	Session variables removed; user redirected. Priority: High
TC-SEC-01	Security	XSS protection	Malicious payload in DB fields; view pages	Data escaped; no script execution. Priority: High
TC-SEC-02	Security	CSRF check	Forms without CSRF token; attempt CSRF	If no CSRF protection, record as vulnerability and recommend tokens. Priority: High
TC-SEC-03	Security	File upload path traversal	Upload allowed; use filename with ..	Upload sanitized; no file written outside target folder. Priority: High
TC-DB-01	Database	Prepared statements used	Review code or attempt injection; try SQL injection values	No SQL injection; queries safe. Priority: High
TC-ADMIN-01	Admin Dashboard	Stats match DB	DB has known counts; open dashboard	Total/Available/Sold counts match DB. Priority: High
TC-ADMIN-02	Admin Dashboard	Top agents dynamic	Agents and properties exist; open dashboard	Top 2 agents show highest property counts. Priority: Medium
TC-ADMIN-03	Admin Dashboard	Monthly sales fallback	sold_date missing or no sold data; open dashboard	Chart shows fallback data; no JS error. Priority: Low

TC-UX-01	UI Forms	Browser client validation	Any form with required attributes; submit incomplete form	Browser blocks submit and shows validation messages. Priority: Medium
TC-PERF-01	Performance	Large image uploads	Use 5 images around 2MB each; submit property	Server handles or rejects gracefully; images optimized; no crash. Priority: Medium
TC-EMAIL-01	Email	Sender identity correct	Trigger OTP or purchase email	Email From shows "FindBrick <noreply@findbrick.com>". Priority: Low
TC-ACCES-S-01	Access Control	Role-based actions	User is normal (not agent/builder); open profile	"Submit Property" button hidden or disabled. Priority: High

## CONCLUSION

This project is developed to meet the requirements of a comprehensive Real Estate Management System. It provides all the essential features for administrators, agents, buyers, and sellers to efficiently manage property listings, user accounts, transactions, feedback, and communications. The system has been successfully implemented with all functionalities outlined in the system requirement specification. It ensures accurate data management, enhances operational efficiency, and minimizes manual effort. The application is built using PHP and MySQL, adhering closely to the specified requirements. Various diagrams and structured modules have been utilized to clearly illustrate the workflow and processes of the system.

## FUTURE ENHANCEMENTS

### Online Payment and Booking:

Enable secure online payment gateways for booking properties, paying deposits, or service fees through the platform.

### Automated Notifications:

Add email and SMS notification systems to alert users, agents, and admins about new listings, updates, appointments, and inquiries.

### Virtual Tours and Media Galleries:

Allow property owners and agents to upload 360-degree virtual tours, walkthrough videos, and detailed image galleries for each property.

### Mobile Application:

Extend the platform with a mobile app for Android and iOS to make property browsing, communication, and management more accessible.

### Chatbot/Live Chat Support:

Integrate a chatbot or live chat feature to answer user queries instantly and assist in the property search process.

### Document Management:

Allow users to upload and manage legal documents related to property transactions securely within the platform.

### Multi-Language and Localization Support:

Add support for multiple languages and region-specific content to cater to a wider audience.

### Customer Review and Rating System:

Enable customers to rate and review agents and properties, helping maintain transparency and trustworthiness.

### Integration with Third-party Services:

Integrate with external services such as mortgage calculators, property valuation tools, or legal advisory services.

## BIBLIOGRAPHY

We created this project in the early stage of our career. To complete it, we learned many topics from different websites. These websites helped us understand and build the project successfully. The main references are:

### Websites:

- [1] <https://www.tutorialspoint.com/>
- [2] <https://code.visualstudio.com/>
- [3] <https://github.com/>
- [4] <https://www.php.net/>
- [5] <https://www.costadelsol-luxury.com/>
- [6] <https://excalidraw.com/>
- [7] <https://www.php.net/>
- [8] <https://www.mysqltutorial.org/>
- [9] <https://www.phpmyadmin.net/>
- [10] <https://www.w3schools.com/>
- [11] <https://dribbble.com/>
- [12] <https://getbootstrap.com/>
- [13] <https://codepen.io/>
- [14] <https://in.pinterest.com/>
- [15] <https://cdnjs.com/libraries>
- [16] <https://youtube.com>
- [17] <https://www.geeksforgeeks.org>

# Thank You

## PROJECT CD

