1. INTRODUCTION TO PROJECT

The web-based "Society Management System" project is an attempt to digitize and streamline the management operations of residential societies. This system enables residents and society administrators to handle tasks such as visitor management, maintenance requests, resident communication efficiently and transparently.

The system provides a Quick Search feature, allowing users to look up information such as contact details of society members, ongoing maintenance activities, and financial reports without requiring login. However, for more detailed actions like submitting maintenance requests or viewing personal account statements, users must log in to their accounts.

The system allows society residents to access a host of features, including but not limited to updating personal information, viewing the society's event calendar, booking amenities, and receiving important notices from the management. Administrators have additional capabilities such as managing resident information, scheduling maintenance tasks, and overseeing financial transactions. The system ensures data integrity and enhances communication between residents and the management committee.

---

2. REQUIREMENTS

2.1 FUNCTIONAL REQUIREMENTS

2.1.1 Homepage

Displays a Homepage of Application with a Login Option for Different Users.

2.1.2 Login Page

Allow residents, secretaries, and admins to log in to access their personalized dashboards.

2.1.3 Admin Dashboard

Option to add a new society with fields for name, address, and basic details and Manage existing societies, including editing information and overseeing society operations.

2.1.4 Resident Dashboard

- View personal details like flat number, contact information.
- Submit maintenance requests for their flat or building.
- Book amenities available in their society, such as the gym, pool, or community hall.

2.1.5 Secretary Dashboard

- Manage residents' requests and amenity bookings.
- Post announcements and update event schedules.
- Access and update society information.

2.1.16User Account

1. Residents, referred to as 'users', are presented with various options when interacting with the system. Registered users can access all features can view public information like event schedules and notices.

2.1.7 Registration and Creation of User Profile

Users must register to perform most actions within the system. Registration requires details such as name, address, contact number, and email. A user ID and password are generated for future logins.

2.1.38Add Society

The system's Add Society allows Admin to Add New Societies is the System and Individual amenities for each Society.

2.1.9 Add Flat

Admin can Add Flats into the Respected Different Societies.

2.2 NON-FUNCTIONAL REQUIREMENTS

2.2.1 Interface

Refer to Appendix B for user interface designs.

2.2.2 Performance

- The system should handle up to 500 concurrent users.
- It must provide real-time updates for bookings and Announcements.

2.2.3 Security and Privacy

The system must ensure secure storage of user data and adherence to data protection regulations. Access controls must be in place to prevent unauthorized access to sensitive information.

2.2.4 Other Requirements:

Hardware Interfaces:
The SPMS is expected to function on Intel PIII 900 MHz Processor equivalent or above, 128 MB RAM, 20 GB HDD.

Software Interfaces:

The SPMS shall work on MS Windows operating systems family (MS Windows 98, MS Windows NT Workstation, MS Windows 2000, MS Windows XP). It configures to work with Oracle database. This System works on Apache Tomcat server. It uses browser IE 5.0 & above. It uses IIS 5.0 server.

---

3. DESIGN

3.1 Database Design

1. Admin Table

Key Type / Constraint	Column Name	Data Type	Length	Allow Null (1=Yes; 0=No)
Primary	id	int	-	0
	name	varchar	45	1
	email	varchar	45	0
	mobile_no	varchar	15	1
	password	varchar	255	0
	status	tinyint	1	1
	last_login_at	datetime	-	1

2. Society Table

Key Type / Constraint	Column Name	Data Type	Length	Allow Null (1=Yes; 0=No)
Primary	id	int	-	0
	name	varchar	255	1
	address	varchar	255	1

3. Secretary Table

Key Type / Constraint	Column Name	Data Type	Length	Allow Null (1=Yes; 0=No)
Primary	id	int	-	0
Foreign (Society)	society_id	int	-	1
	name	varchar	45	1
	email	varchar	45	1
	mobile_no	varchar	15	1
	password	varchar	255	1
	last_login_at	datetime	-	1
	status	tinyint	1	1

4. Owner Table

Key Type / Constraint	Column Name	Data Type	Length	Allow Null (1=Yes; 0=No)
Primary	id	int	-	0
Foreign (Secretary)	secretary_id	int	-	1

Key Type / Constraint	Column Name	Data Type	Length	Allow Null (1=Yes; 0=No)
	flat_no	int	-	1
	name	varchar	255	1
	address	varchar	255	1
	mobile_no	varchar	15	1
	email	varchar	45	1
	password	varchar	255	1
	status	tinyint	1	1

5. Announcement Table

Key Type / Constraint	Column Name	Data Type	Length	Allow Null (1=Yes; 0=No)
Primary	id	int	-	0
Foreign (Secretary)	secretary_id	int	-	1
	announcement	varchar	255	1
	created_at	timestamp	-	1
	updated_at	datetime	-	1

6. Services Table

Key Type / Constraint	Column Name	Data Type	Length	Allow Null (1=Yes; 0=No)
Primary	id	int	-	0
	name	varchar	45	1
	service_person	varchar	45	1
Foreign (Secretary)	secretary_id	int	-	1

7. Amenities Table

Key Type / Constraint	Column Name	Data Type	Length	Allow Null (1=Yes; 0=No)
Primary	id	int	-	0
Foreign (Secretary)	secretary_id	int	-	1
	name	varchar	45	1

---

4. CODING STANDARDS IMPLEMENTED

- **Naming and Capitalization:** Pascal casing for class names, camel casing for methods and parameters.
- **Comments:** Each method and significant code block are well-documented. Comments explain logic rather than restating code.

5. TEST REPORT

General Testing:

SR-NO	TEST CASE	EXPECTED RESULT	ACTUAL RESULT	ERROR MESSAGE
1	User Registration	User account created successfully	OK	Nothing
2	Login	Redirect to dashboard on successful login	OK	Please enter correct username/password.
3	Add Society	Submitted and confirmation shown	OK	Nothing
4	Add Flat	Submitted and confirmation shown	OK	Nothing

(Additional test cases will be documented here.)

---

6. PROJECT MANAGEMENT RELATED STATISTICS

DATE	WORK PERFORMED	SLC Phase	Additional Notes
Feb 1, 2024	Project Allotment and User Requirements Gathering	Feasibility Study	Initial meeting with client to gather requirements.
Feb 5, 2024	Initial SRS Document Validation	Requirement Analysis	Presentation to client for requirement clarification.
...	...	...	...

(Detailed timeline with project milestones and phases.)

Appendix A

(Diagrams and additional design details.)

Appendix B

(User interface screenshots and descriptions.)

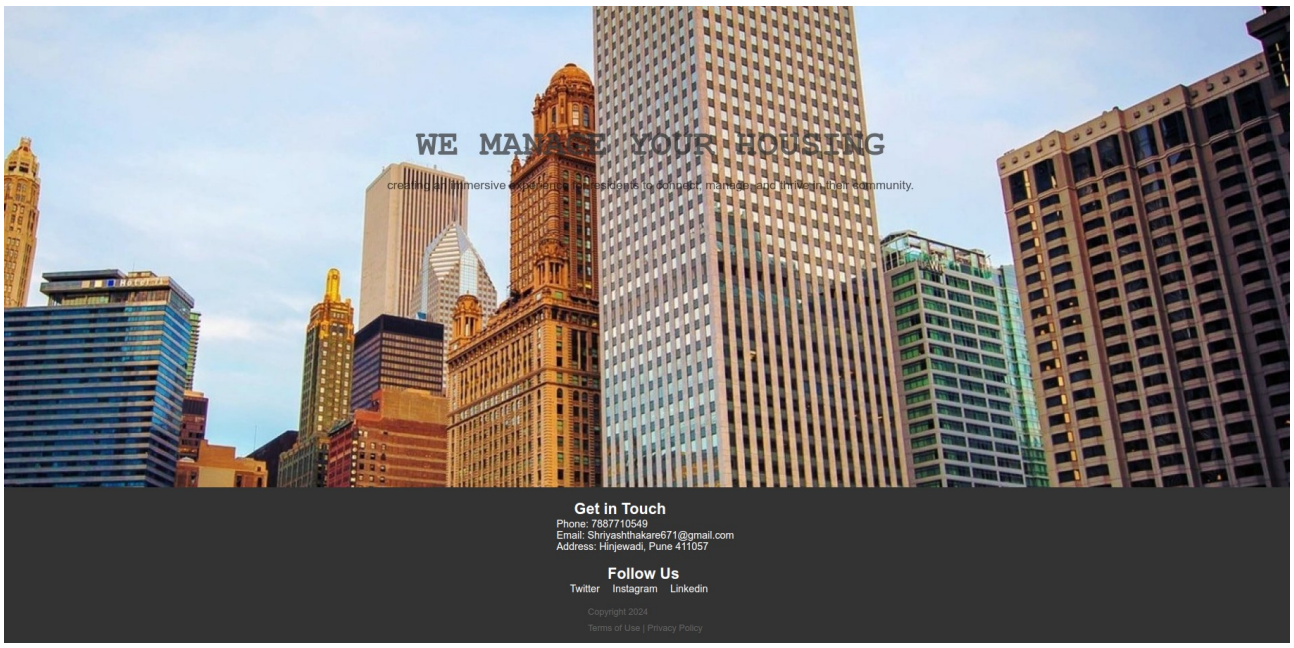
---

7. REFERENCES

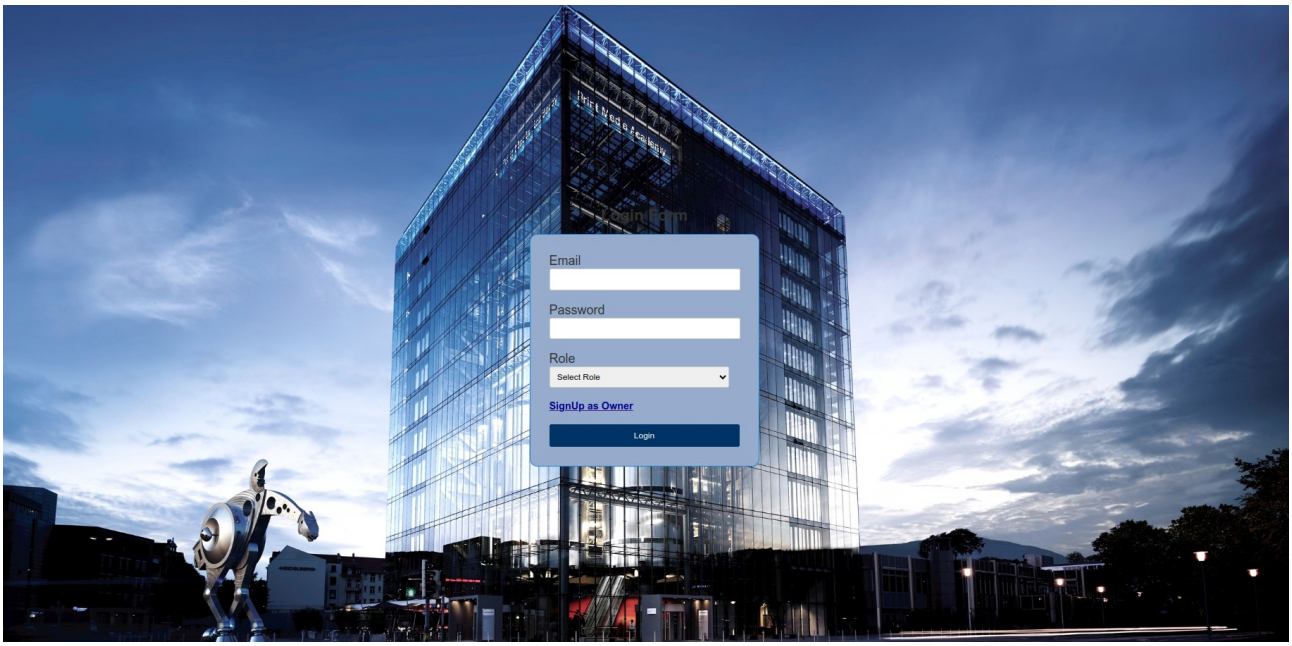
- Google
- W3Schools
- geeksgorgeeks
- Javatpoint

8. Output Snaps :

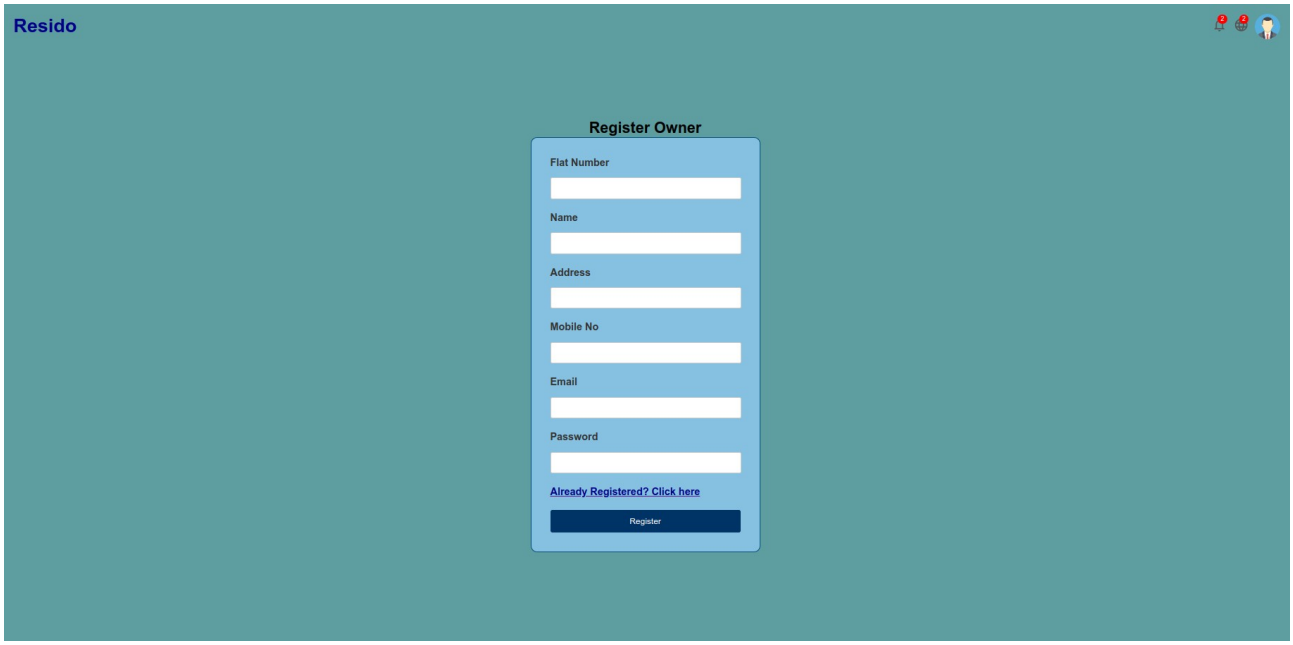
Home Page :



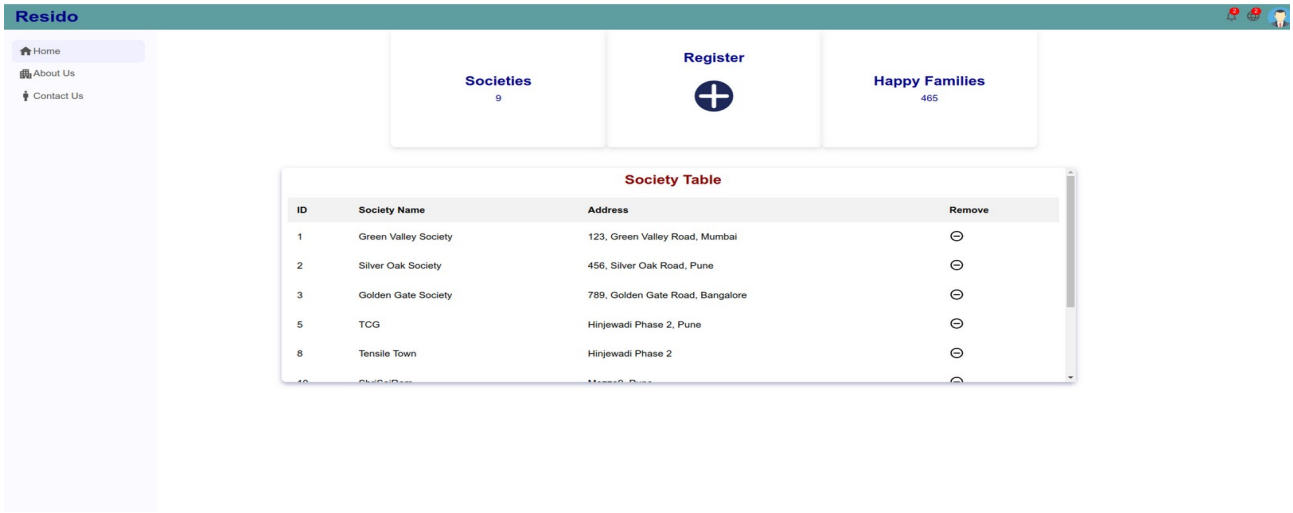
LogIn Page :

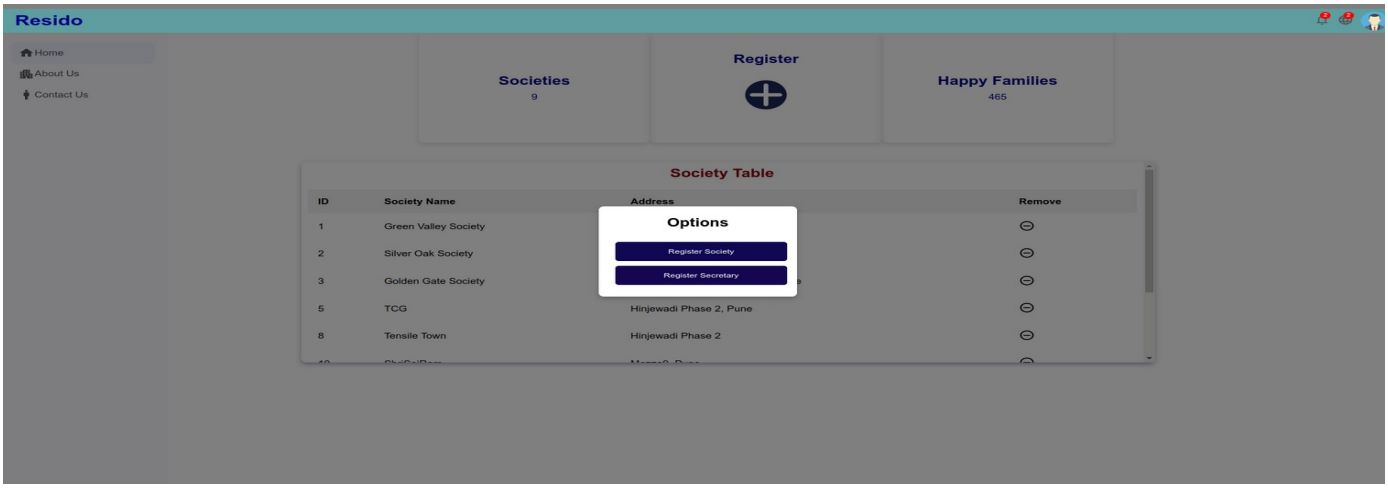


Registration Page:

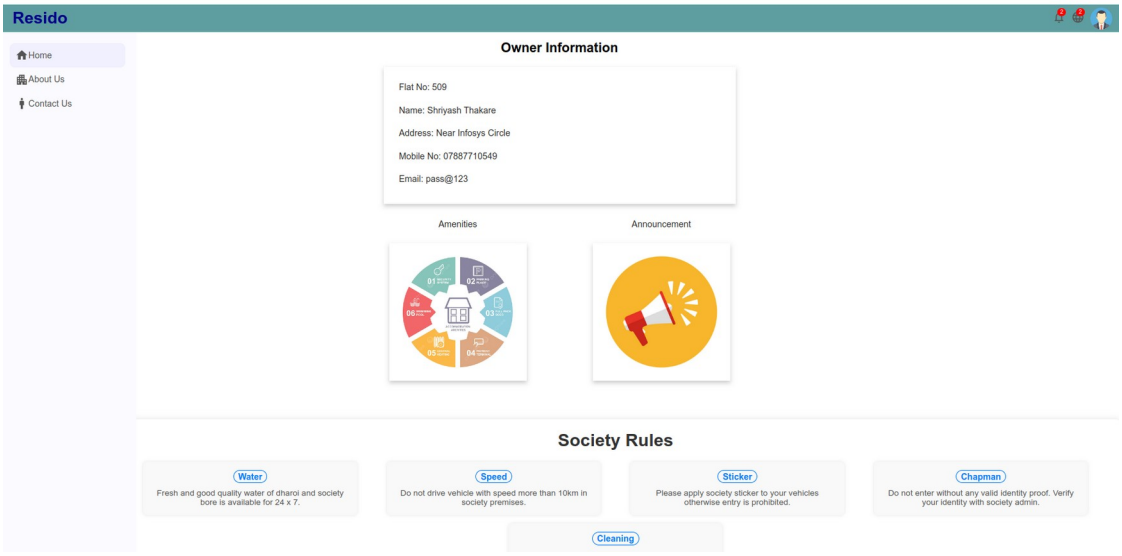


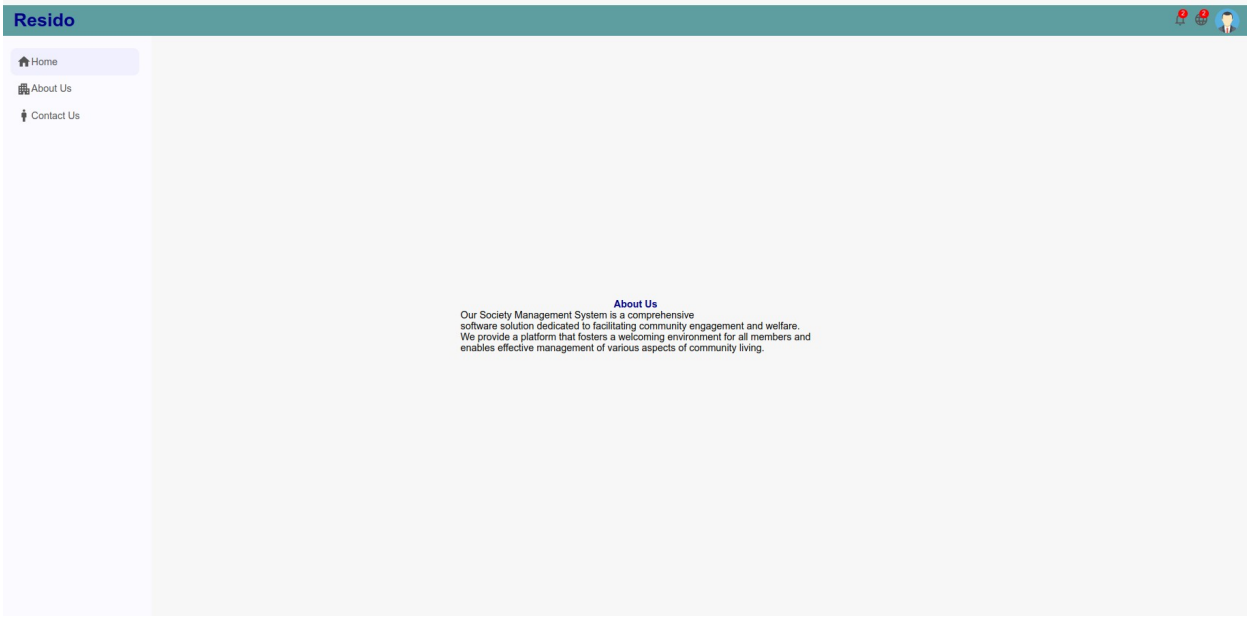
Admin Dashboard :

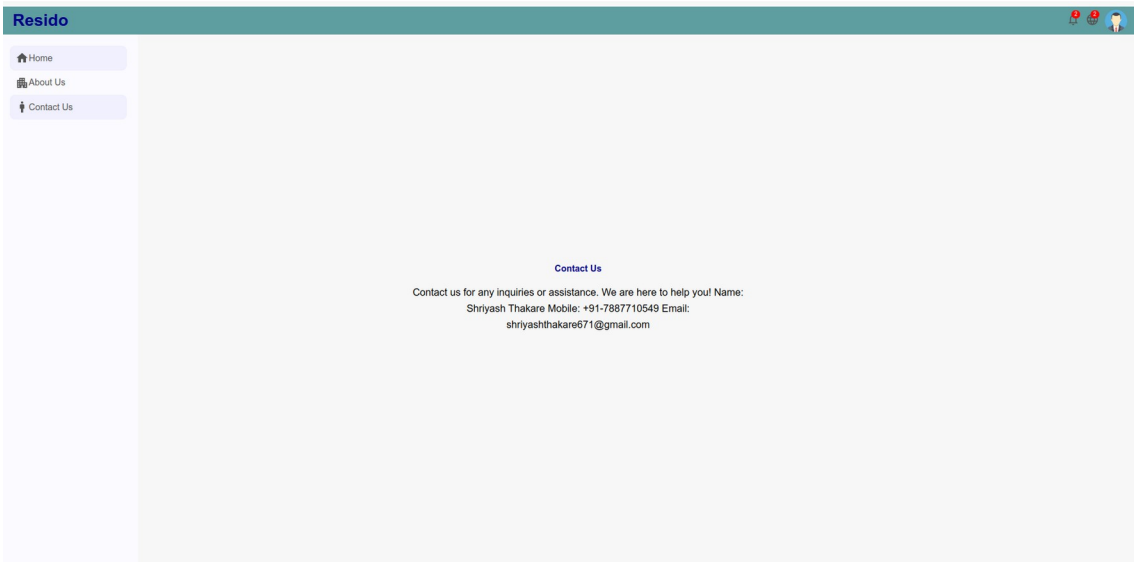




Owner dashboard :







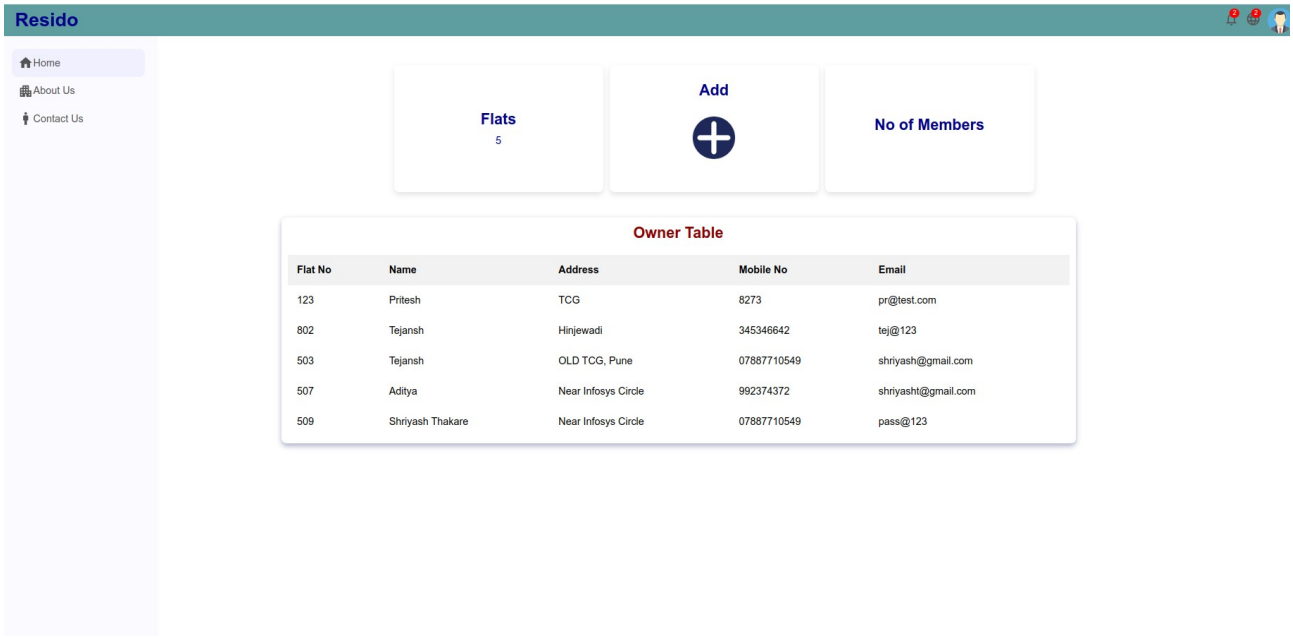
Amenities Table

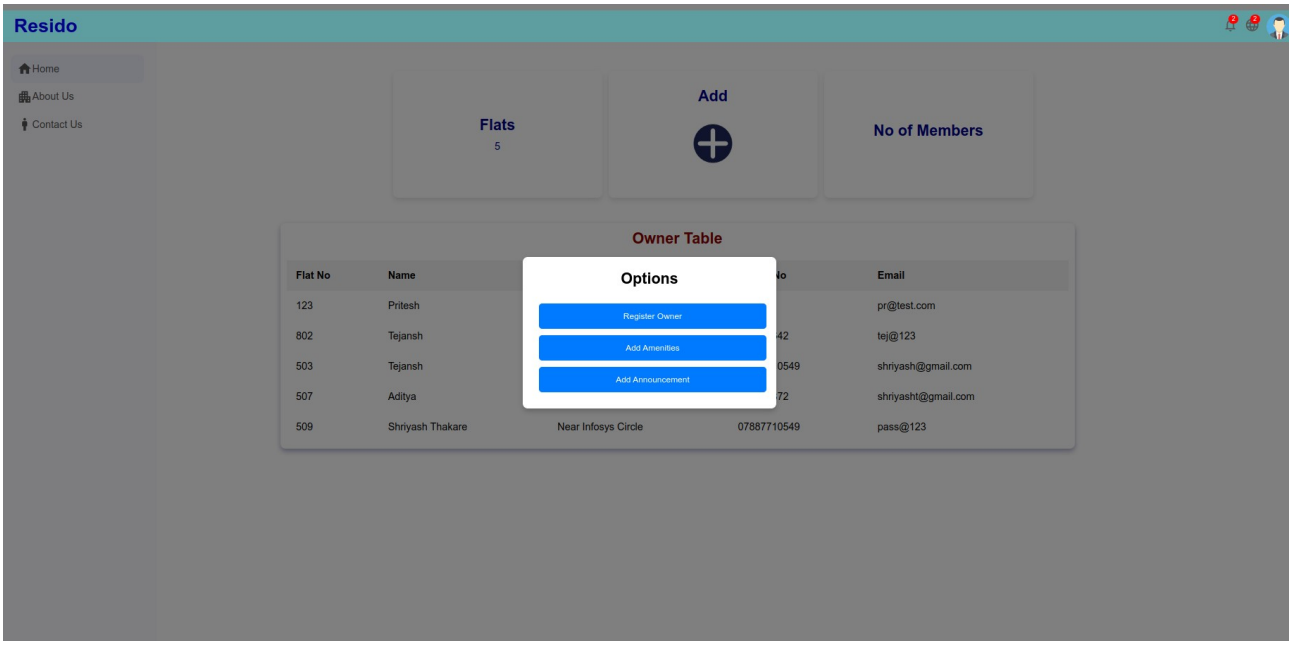
Amenity ID	Amenity Name
5	Swimming Pool
6	Lift
7	Gas pipeLine
8	AC
9	Gym
10	parking

Announcements

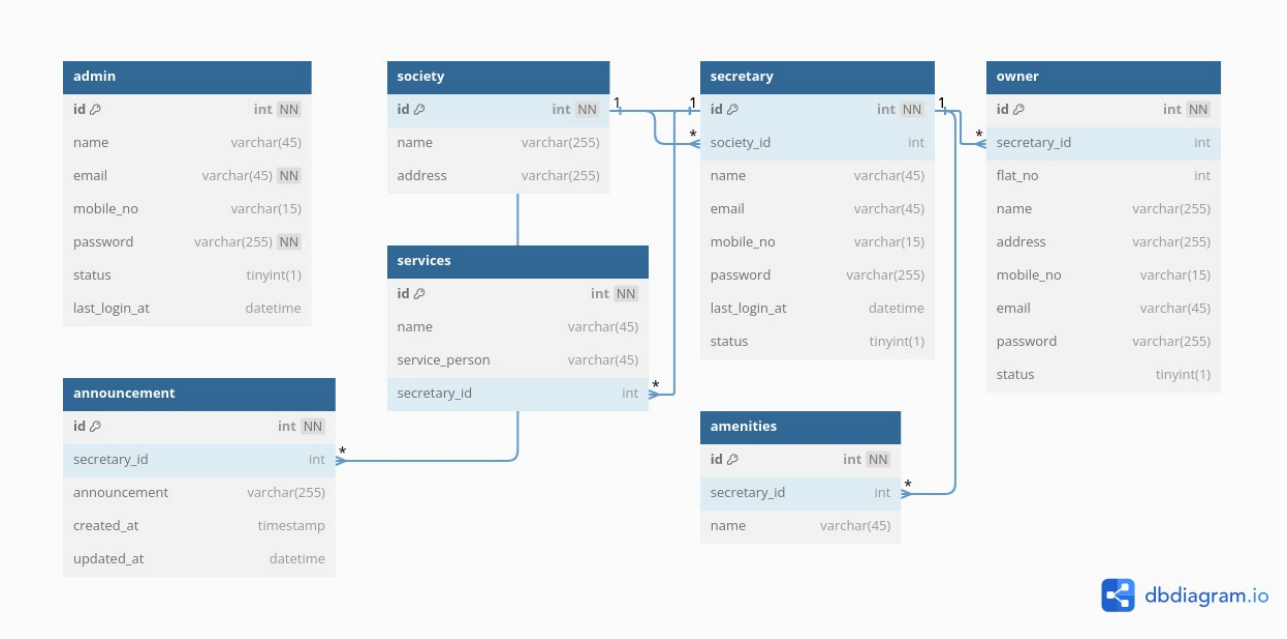
Announcement	Created At
Birthday Party	Fri Aug 09 2024
Pani Nahi Ayegaaa	Fri Aug 09 2024
Lite Nahi Ayegi	Sat Aug 10 2024
light nhi hai	Mon Aug 12 2024

Secretary Dashboard :

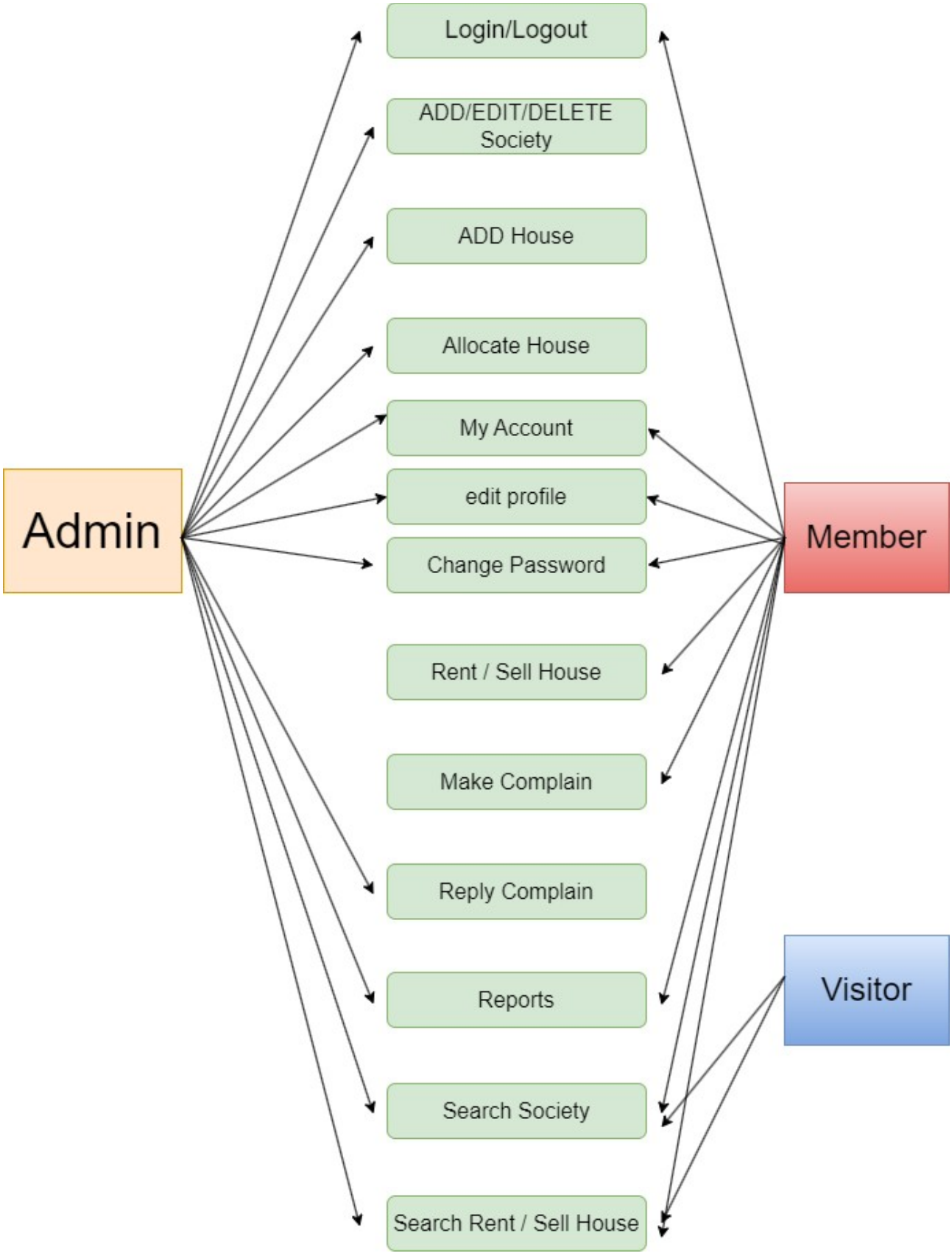
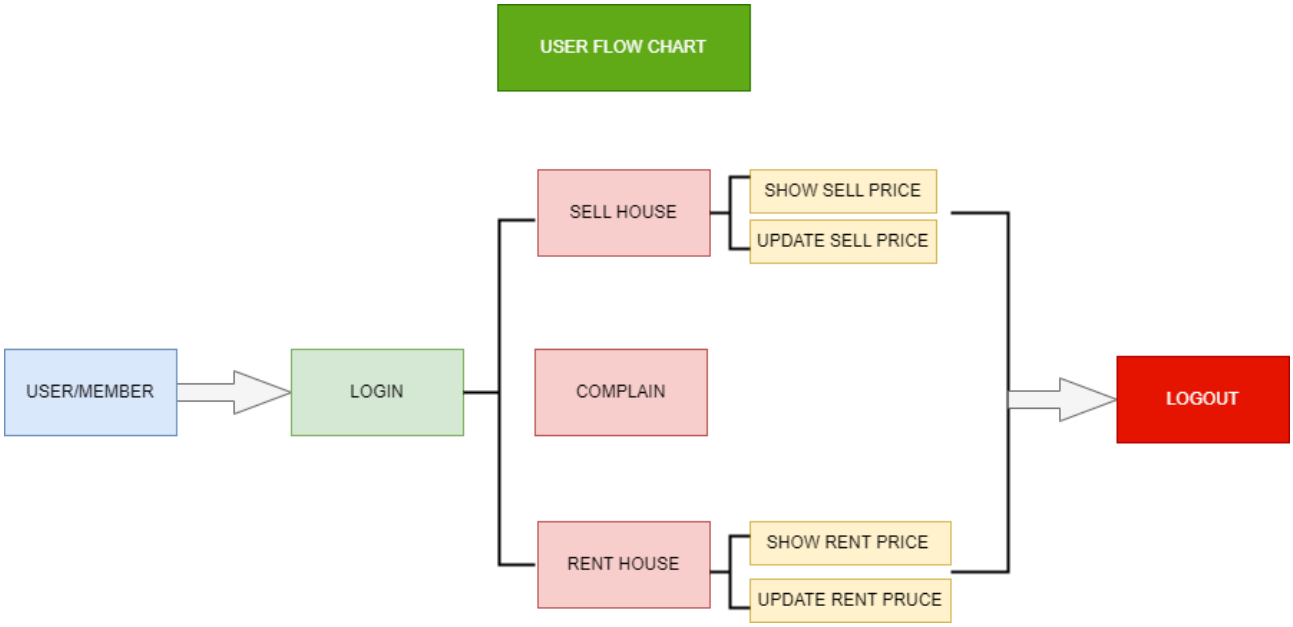




ER Diagrams :


admin
id int NN
name varchar(45)
email varchar(45) NN
mobile_no varchar(15)
password varchar(255) NN
status tinyint(1)
last_login_at datetime
society
id int NN
name varchar(255)
address varchar(255)
secretary
id int NN
society_id int
name varchar(45)
email varchar(45)
mobile_no varchar(15)
password varchar(255)
last_login_at datetime
status tinyint(1)
owner
id int NN
secretary_id int
flat_no int
name varchar(255)
address varchar(255)
mobile_no varchar(15)
email varchar(45)
password varchar(255)
status tinyint(1)
services
id int NN
name varchar(45)
service_person varchar(45)
secretary_id int
announcement
id int NN
secretary_id int
announcement varchar(255)
created_at timestamp
updated_at datetime
amenities
id int NN
secretary_id int
name varchar(45)
1
*
dbdiagram.io

USE Case Diagram :



User Flow Diagram :