**Property Maintenance System**

**Actors :**

1. Admin
2. Member
3. Watchmen
4. **Admin :**

* Login in system
* Update Profile
* Maintain member,watchmen detail

(Provide Credential for Login)

* Maintain maintenance agreement related detail
* Maintain member maintenace record
* Maintain expenses record according to income of maintenance
* Maintain settings of website

(like , notification about due date of maintence ,event etc.)

* Maintain comity member and meeting
* Maintain member complain according complain type

(Give response about complain status)

* Maintain event according to types

(View member suggestion for event)

* Maintain Gallery of event photo
* Maintain Digital Dashboard

(like, urgent meeting etc.)

* Maintain emergency contact record

(Also alert system - if anyone press alert all member and admin notify about emergency)

* View vistior detail

1. **Member :**

* Login in system
* Update Profile
* View maintenance related detail
* Pay maintence , get receipt
* View expenses of all property
* Get notification related maintenance,event etc.
* Maintain complain detail

(view response from admin)

* View event detail

(Add some suggestion for event)

* Maintain gallery of event photo
* View Digital dashboard
* View emergency contact information with alert system
* View visitor detail

1. **Watchmen :**

* Login in system
* Update profile
* Maintain visitor detail

(like name,phone-no,which person to meet,entry-exit timing)

[-receipt upload compulsory

-monthwise prices]

[

gmail:sociosphere25@gmail.com

Password:Asti@25092002

App Password:lojonxbjgecndbgd

]

[email:paladiyavimala@gmail.com

Password:**RhenU3f**

Group:member

]

**SocioSphere** is a combination of two words:

1. **Socio-**: Derived from "society" or "social," representing **community, people, and organized living**.
2. **Sphere**: Represents **wholeness, completeness, and a well-structured system**.

### ****Meaning of SocioSphere****

It symbolizes a **connected, organized, and well-managed society** where residents, management, and services function smoothly within a unified system. The name gives a sense of **inclusivity, harmony, and digital transformation** in society management.

SocioSphere - Where Community Meets Convenience.

### 💡 Summary of Button Names Based on Status

| **Complaint Status** | **Button(s) to Show** |
| --- | --- |
| Pending | ✅ Accept, ❌ Reject |
| In Progress | ✔️ Mark as Completed |
| Completed | — (No button, just status) |
| Rejected | — (No button, just status) |

For entity creation:

Scaffold-DbContext "Data Source=(localdb)\ProjectModels;Initial Catalog=sociosphere;Integrated Security=True;Encrypt=True;Trust Server Certificate=True" Microsoft.EntityFrameworkCore.SqlServer -OutputDir Models/Entity

[notification]

[jwtvalidation]

[-email]

[-errormessage]

[-validation(client,user)]

[-status update na time pr confirmation message or koi pn note send karvani hoy evu mukvanu]

[update ma type sathe ena record block nathi karva khali e block valu dropdown ma nai batavanu]

* This Link for get data from api in MVC

https://youtu.be/pIlG3JVZy1Q?si=O5Kakbw2HMhL1XxN

Pending work

-committemember controller ma=active user dropdown baki

-committemember controller ma=addcommitte na time pr last e member nu je kyo committee member che e add karvanu e baki

society-management react file sturcture:

│── src

│ │── admin

│ │ ├── components

│ │ │ ├── AdminNavbar.js

│ │ │ ├── AdminSidebar.js

│ │ ├── pages

│ │ │ ├── AdminDashboard.js

│ │ │ ├── SocietyCommittee.js

│ │ ├── AdminPanel.js

│ │── member

│ │ ├── components

│ │ │ ├── MemberNavbar.js

│ │ │ ├── MemberSidebar.js

│ │ ├── pages

│ │ │ ├── MemberDashboard.js

│ │ ├── MemberPanel.js

│ │── pages

│ │ ├── Login.js <-- Shared Login Page

│ │── auth

│ │ ├── AuthContext.js

│ │── App.js

│ │── index.js

│── package.json

**Property Maintenance System**

* **Database :**

1. **Group\_master :**

|  |  |
| --- | --- |
| Id(P.K.) | Int |
| Group\_name | Varchar |

1. **User\_master**

|  |  |
| --- | --- |
| Id(P.K.) | Int |
| Group\_id(F.K.) | Int |
| Commite\_member\_id(F.K.) | Int |
| First\_name | Varchar |
| Middle\_name | Varchar |
| Last\_name | Varchar |
| Profile\_photo | Varchar |
| Email\_address(unique) | Varchar |
| Phone\_no | Bigint |
| password | Text |
| gender | Varchar |
| address | Text |
| Total\_familymember | Int |
| Adharcard\_no | int |
| Square\_footsize | float |
| Living\_date(move\_in\_date) | Date |
| Status(Active/Block) | Varchar |
| Created\_at | datetime |
| Updated\_at | datetime |

* group\_id(foreign\_key)->Group\_master(id)
* Commite\_member\_id(foreign\_key)->Commitie\_member\_record\_(id)

1. **User\_personal\_detail:**

|  |  |
| --- | --- |
| Id(P.K.) | Int |
| User\_id(F.K.) | Int |
| Flat\_no | Int |
| Father\_name | Varchar |
| Mother\_name | Varchar |
| Relationship\_status | Varchar |
| Spouse\_name | Varchar |
| Spouse\_occupation | Varchar |
| No\_of\_child | Int |
| Date\_of\_birth | Date |
| Occupation | Varchar |
| Another\_phone\_no | Bigint |
| Shift\_start\_time | Time |
| Shift\_end\_time | Time |
| Work\_experience | Varchar |
| Another\_id\_proof | Varchar |
| Joining\_date | Date |
| Salary | float |

* user\_id(foreign\_key)->User\_master(id)

1. **Society\_Committe\_master:**

|  |  |
| --- | --- |
| Id(P.K.) | Int |
| Committe\_name | Varchar |
| description | Text |
| Status(Active/Block) | Varchar |
| Created\_at | datetime |
| Updated\_at | datetime |

1. **Committe\_member\_record:**

|  |  |
| --- | --- |
| Id(P.K.) | Int |
| Commitie\_type\_id(F.K.) | Int |
| User\_id(F.K.) | Int |
| Start\_date | Date |
| End\_date | Date |
| Status(Active/Block) | Varchar |
| Created\_at | datetime |
| Upadted\_at | datetime |

* Commitie\_type\_id(foreign\_key)->Group\_master(id)
* user\_id(foreign\_key)->User\_master(id)

1. **Maintenance\_charge\_master:**

|  |  |
| --- | --- |
| Id(P.K.) | Int |
| Maintenace\_charge(per month wise) | Double |
| description | Text |
| Starting\_month\_year | Date |
| Due\_month\_year | Date |
| End\_month\_year | Date |
| Maintennace\_charge(Per Month) | float |
| Maintenance\_type(Flatt-1/square\_foot-1) | Varchar |
| Created\_at | datetime |
| Updated\_at | datetime |

1. **Maintenance\_record :**

|  |  |
| --- | --- |
| Id(P.K.) | Int |
| Maintenance\_id(F.K.) | Int |
| User\_id(F.K.) | Int |
| Receipt\_no | Int |
| Total\_maintenance | Double |
| Paid\_date | Date |
| From\_month | Date |
| To\_month | Date |
| Payment\_image | Varchar |
| Late\_payment\_charge | float |
| Late\_payment\_reason | Varchar |
| IsLatePayment | boolean |
| Status(Pending/Complete) | Varchar |
| Created\_at | datetime |
| Updated\_at | datetime |

* User\_id(foreign\_key)->User\_master(id)
* Maintenance\_id(foreign\_key)->Maintenance\_charge\_master(id)

1. **Expenses\_master**

|  |  |
| --- | --- |
| Id(P.K.) | Int |
| Expenses\_name | Varchar |
| Status(Active/Block) | Varchar |
| Created\_at | datetime |
| Updated\_at | datetime |

1. **Expenses\_record**

|  |  |
| --- | --- |
| Id(P.K.) | Int |
| Expenses\_type\_id(F.K.) | Int |
| Expenses\_title | Varchar |
| description | Text |
| Invoice(Bill) | Varchar |
| image | varchar |
| Price | Double |
| Expenses\_Date | Date |
| Status(Pending/Completed) | Varchar |
| Created\_at | datetime |
| Updated\_at | datetime |

* Expenses\_type\_id(foreign\_key)->Expenses\_master(id)

1. **Complain\_type:**

|  |  |
| --- | --- |
| Id(P.K.) | Int |
| Complain\_name | Varchar |
| Status(Active/Block) | Varchar |
| Created\_at | datetime |
| Updated\_at | datetime |

1. **Complain\_suggestion\_master:**

|  |  |
| --- | --- |
| Id(P.K) | Int |
| User\_id(F.K.) | Int |
| category(Suggestion/Complain) | varchar |
| Complain\_type\_id(F.K.) | Int |
| Priority(High/Medium/Low) | varchar |
| Complain\_title | Varchar |
| description | Text |
| photo | Varchar |
| Admin\_action\_taken\_note | Varchar |
| Action\_taken\_due\_date | Date |
| Votes\_count(No. of members supporting this suggestion) | Int |
| Status(Pending/InProgress/Completed/Rejected/Accepted) | Varchar |
| Created\_at | datetime |
| Updated\_at | datetime |

* User\_id(foreign\_key)->User\_master(id)
* Complain\_type\_id(foreign\_key)->Complain\_type(id)

1. **Event\_master :**

|  |  |
| --- | --- |
| Id(P.K.) | Int |
| Event\_type(Paid/Free) | Varchar |
| Event\_name | Varchar |
| description | Text |
| Event\_date | Date |
| Event\_time | Time |
| destination | Varchar |
| Price\_type(per\_person/all) | Varchar |
| Price | float |
| Reason | varchar |
| Status(Upcoming/completed/canceled) | Varchar |
| Created\_at | datetime |
| Updated\_at | datetime |

1. **Paid\_event\_record :**

|  |  |
| --- | --- |
| Id(P.K.) | Int |
| Event\_id(F.K.) | Int |
| User\_id(F.K.) | Int |
| Total\_member | Int |
| Total\_price | Float |
| Payment\_image | Varchar |
| Status(Pending/complete) | Varchar |
| Created\_at | datetime |
| Updated\_at | datetime |

* User\_id(foreign\_key)->User\_master(id)
* Event\_id(foreign\_key)->Event\_master(id)

1. **Event\_gallery:**

|  |  |
| --- | --- |
| Id(P.K.) | Int |
| User\_id(F.K.) | Int |
| Event\_id(F.K) | Int |
| photo | Varchar |
| description | Text |
| Created\_at | datetime |
| Updated\_at | datetime |

* Event\_id(foreign\_key)->Event\_master(id)
* User\_id(foreign\_key)->User\_master(id)

1. **Society\_meeting :**

|  |  |
| --- | --- |
| Id(P.K.) | Int |
| Meeting\_name | Varchar |
| notice | Text |
| Date | Date |
| time | Time |
| photo | Varchar |
| location | Varchar |
| End\_date(notification) | Date |
| Status(Scheduled/completed/canceled) | Varchar |
| Created\_at | datetime |
| Updated\_at | datetime |

1. **Agenda:**

|  |  |
| --- | --- |
| Id(P.K.) | Int |
| Society\_meeting\_id(f.k.) | Int |
| User\_agenda\_id(F.K.)(who ‘s given agenda) | Int |
| Agenda\_Description | Text |
| Priority(High/Medium/Low) | Varchar |
| Action\_taken\_note | Varchar |
| Status(Complete/Pending) | Varchar |
| Created\_at | datetime |
| Updated\_at | datetime |

* User\_id(foreign\_key)->User\_master(id)
* Society\_meeting\_id(foreign\_key)->Society\_meeting(id)

1. **Agency\_master:**

|  |  |
| --- | --- |
| Id(P.K.) | Int |
| Agency\_type\_name | Varchar |
| Status(Active/Block) | Varchar |
| Created\_at | Timestamp |
| Updated\_at | Timestamp |

1. **Agency\_contact:**

|  |  |
| --- | --- |
| Id(P.K.) | Int |
| Agency\_type\_id(F.K.) | Int |
| Contact\_person\_name | Varchar |
| location | Varchar |
| Email\_id | Varchar |
| Contact\_no | Bigint |
| Alternate\_contact\_no | Bigint |
| Created\_at | datetime |
| Updated\_at | datetime |

* Agency\_type\_id(foreign\_key)->Agency\_master(id)

1. **Visiter:**

|  |  |
| --- | --- |
| Id(P.K.) | Int |
| Visiting\_user\_id(F.K.) | Int |
| Watchmen\_id(F.K.) | Int |
| name | Varchar |
| Phone\_no | BigInt |
| photo | Varchar |
| Entry\_date | Date |
| Entry\_time | Time |
| Exit\_date | Date |
| Exit\_time | Time |
| Created\_at | datetime |
| Updated\_at | datetime |

* Visiting\_user\_id(foreign\_key)->User\_master(id)
* Watchmen\_id(Foreign\_key)->User\_master(id)

1. **Setting\_menu**

|  |  |
| --- | --- |
| Id(P.K.) | Int |
| Key | Varchar |
| Value | Varchar |