Society Management System – Feature Blueprint

# . Technical Stack

- Frontend: React.js / html css Javascript   
- Backend: Django (Python) + REST API  
- Database: MySQL / PostgreSQL (for structured data).  
- Authentication: JWT / OAuth

# User Roles & Permissions

## Super Admin

- Create, update, and delete societies.  
- Assign Admins to specific societies.  
- Monitor activities of all societies (billing, complaints, reports).  
- Manage subscription plans/maintenance models.  
- Manage integrations ( SMS/email services).

## Admin (Per Society)

- Manage residents (add/update/remove members).  
- Manage flats/units and parking slots.  
- Create and track maintenance bills.  
- Approve/reject facility bookings.  
- Post announcements/events/notices.  
- Handle complaints/service requests.  
- Generate society-specific reports (finance, complaints, usage).  
- Vendor/Staff Management (assign work, track payments).  
- Emergency Management (fire, medical, security alerts).

## Resident (Society Member)

- Raise complaints/service requests.  
- Book facilities (gym, guest room, clubhouse).  
- Receive notifications for announcements/events.  
- Update personal profile.  
- Access digital documents (meeting minutes, society rules).  
- Participate in polls/surveys (decision-making).

# 2. Core Modules

## Authentication & User Management

- Role-based access control (Super Admin / Admin / Resident).

- Secure login with JWT/OAuth.

- Password reset & profile management.

- Biometric/OTP-based login (for mobile app).

## Society Management (Super Admin)

- CRUD operations for societies.

- Assign Admins to societies.

- Manage society-level settings.

- Subscription/plan activation.

## Resident & Flat/Unit Management (Admin)

- Maintain resident database.

- Assign residents to flats/units.

- Track occupancy/vacancy.

- Parking slot management.

## Complaints & Requests

- Residents raise complaints (Plumbing, Security, Electricity, etc.).

- Status workflow: Open → In Progress → Resolved → Closed.

- Admin assigns complaints to staff/vendors.

- SLA (time-bound resolution tracking).

- Feedback option after complaint closure.

## Announcements & Events

- Admin can post notices/events (with attachments).

- Residents get instant notifications.

- Event RSVP/acknowledgment option.

- Digital notice board (auto-expiry after event).

## Facility Booking

- Calendar-based booking for amenities.

- Admin approval/rejection.

- Track usage and revenue from bookings.

- Restrictions (max hours per resident, penalty for no-show).

## Reports & Analytics

- Finance Reports: Income vs expenses, dues collected.

- Complaint Reports: Resolved vs pending complaints.

- Facility Reports: Usage & revenue from bookings.

- Export options: PDF/Excel.

- AI Insights (predict dues collection trends, high-complaint areas).

## Communication & Notifications

- Email/SMS/App notifications.

- In-app messaging between Admin & Residents.

- Automated reminders (booking confirmation, complaint update).

- Emergency Broadcast (fire, security, medical alerts).

## Marketplace (Optional – Real-Time Value Add)

- Residents can buy/sell/rent items (internal classifieds).

- Services listing (electrician, plumber, tutor).

- Integration with local vendors for deliveries.

# 3. Technical Stack

- Frontend: React.js / html css Javascript   
- Backend: Django (Python) + REST API  
- Database: MySQL / PostgreSQL (for structured data).  
- Authentication: JWT / OAuth

# 6. Extra Real-Time Features

- Digital Voting (for society decisions, elections).  
- Panic Button in mobile app (sends SOS to Admin/Security).  
- AI Chatbot for residents (quick queries like due amount, booking status).