Society Financial Management

A High level Technical and Solution Architecture (to be explained with UML diagrams/block diagrams)

1. User Authentication Module:

- Handles user authentication and authorization.
- Validates user credentials (username and password) against a database.
- Provides access control based on user roles.

2. Family Management Module:

- Allows administrators to onboard and manage families living in the society.
- Provides functionalities to add, update, and delete family details.
- Stores information such as family members, contact details, and unit/apartment information.

3. Payment Tracking Module:

- Keeps track of monthly maintenance payments received from each family.
- Sends email reminders to families between the 5th and 10th of every month for payment.
- Applies a fine of Rs. 100 for delayed payments.

4. Payment Approval Workflow Module:

- Handles outward payment requests against invoices.
- Requires a checker user to review and approve payment details before processing.
- Ensures proper authorization and control over society expenses.

5. Activity Logging Module:

- Captures all activities within the system.
- Logs user actions, system events, and transaction details.
- Provides an audit trail for tracking changes and monitoring system usage.

6. Reporting Module:

- Generates various reports for financial analysis and decision-making.
- Includes reports such as:
  - Inward Payments for a given date range.
  - Families with frequent delayed payments.
  - Yearly spending reports.
  - Year-over-year spending increase analysis.

7. Data Storage Layer:

◦ Stores all application data including user details, family information, payment records, activity logs, and financial reports.

◦ Utilizes a relational database management system (e.g., MySQL, PostgreSQL) for data storage.

8. User Interface Layer:

◦ Provides a user-friendly interface for interacting with the system.

◦ Includes web-based or mobile application interfaces.

◦ Allows users to access functionalities such as login, family management, payment tracking, reporting, etc.

9. Integration Layer:

◦ Integrates with external systems or services as needed.

◦ Includes integration with email service providers for sending reminders, payment gateways for processing transactions, and other third-party APIs for additional functionalities.

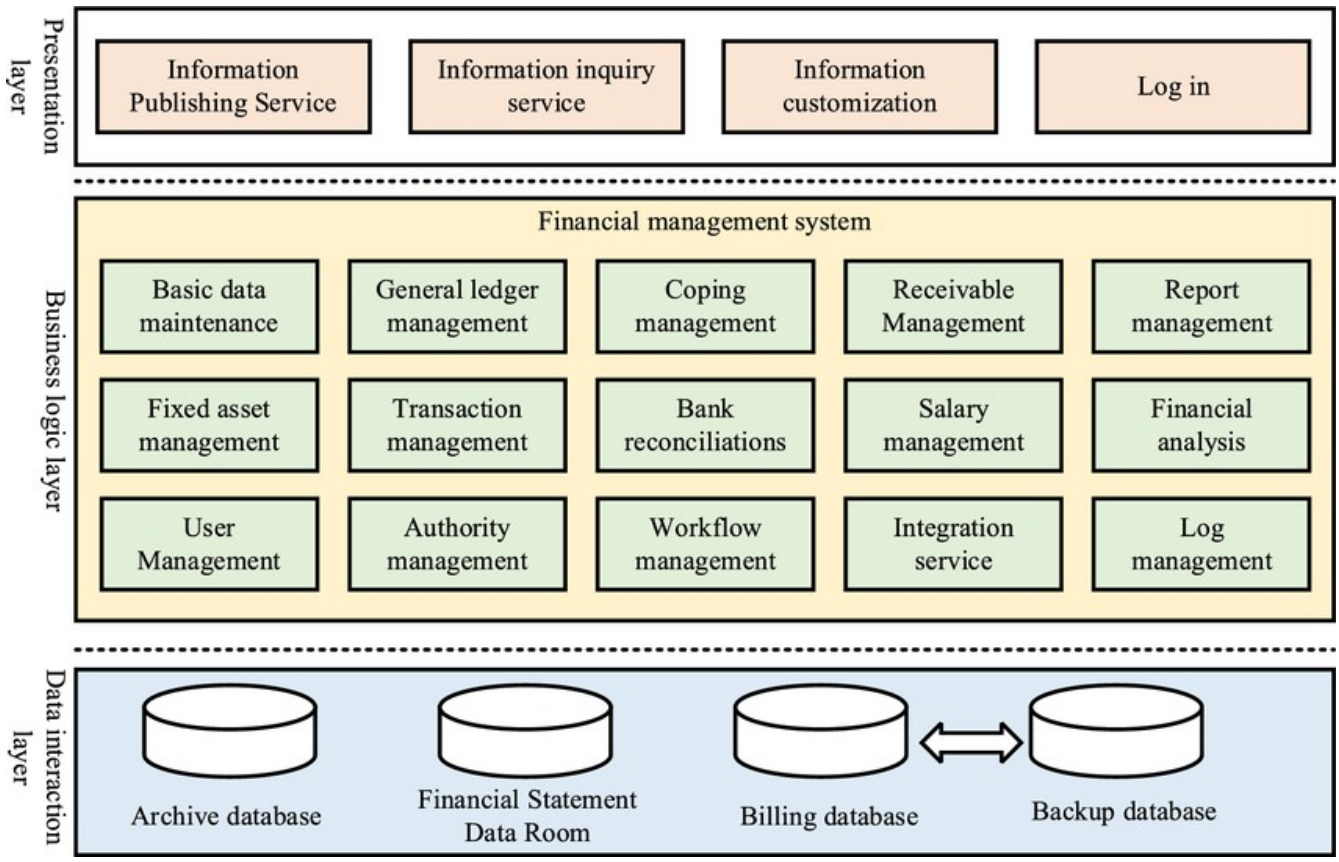
10. Security Layer:

◦ Implements security measures such as encryption, data validation, and access control.

◦ Ensures the confidentiality, integrity, and availability of sensitive data.

This architecture provides a structured framework for developing a robust Society Financial Management system that meets the specified requirements while ensuring scalability, reliability, and security.

Let's break down the high-level technical and solution architecture for the Society Financial Management system using UML diagrams and block diagrams:



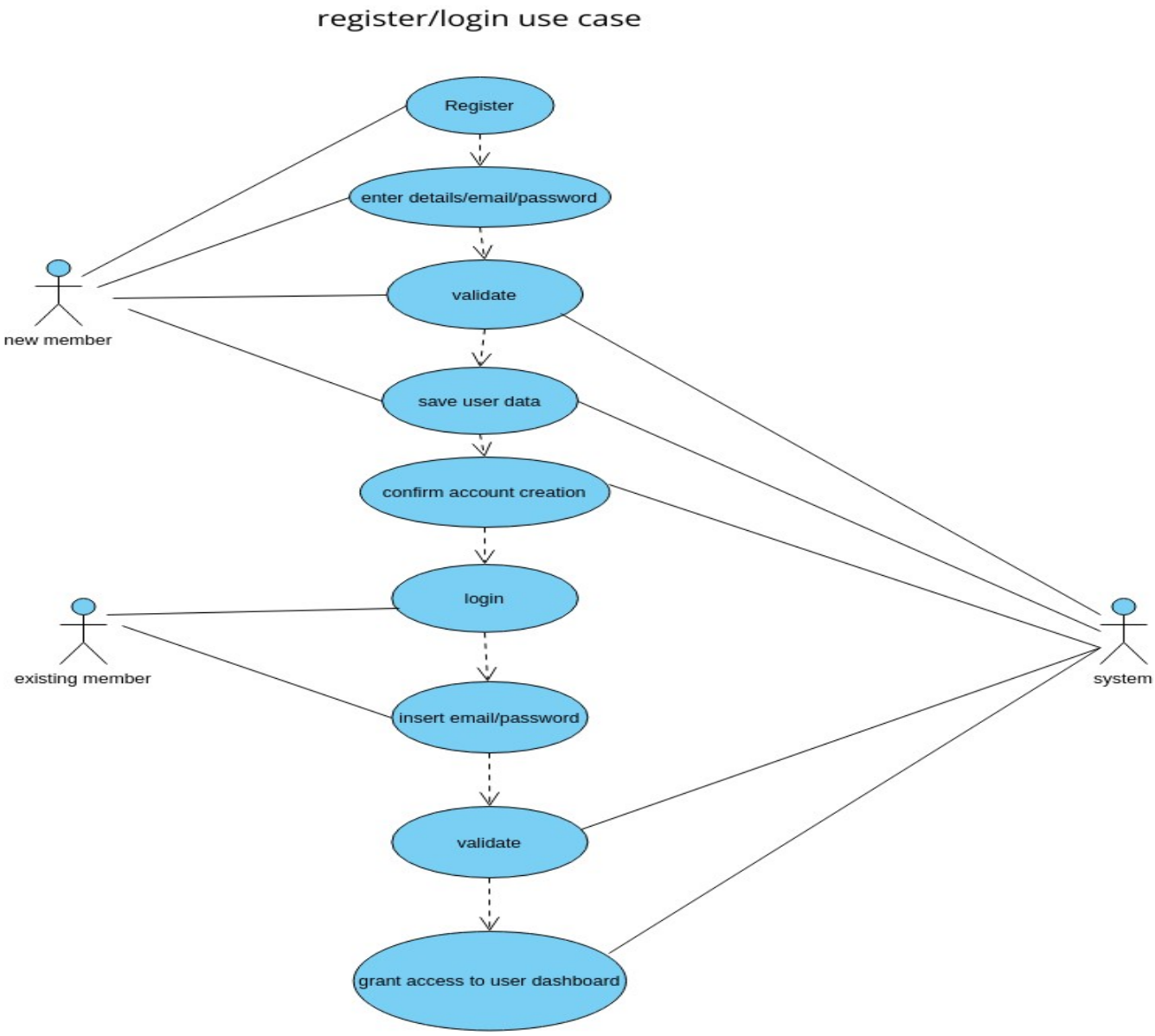
System Architecture Overview:

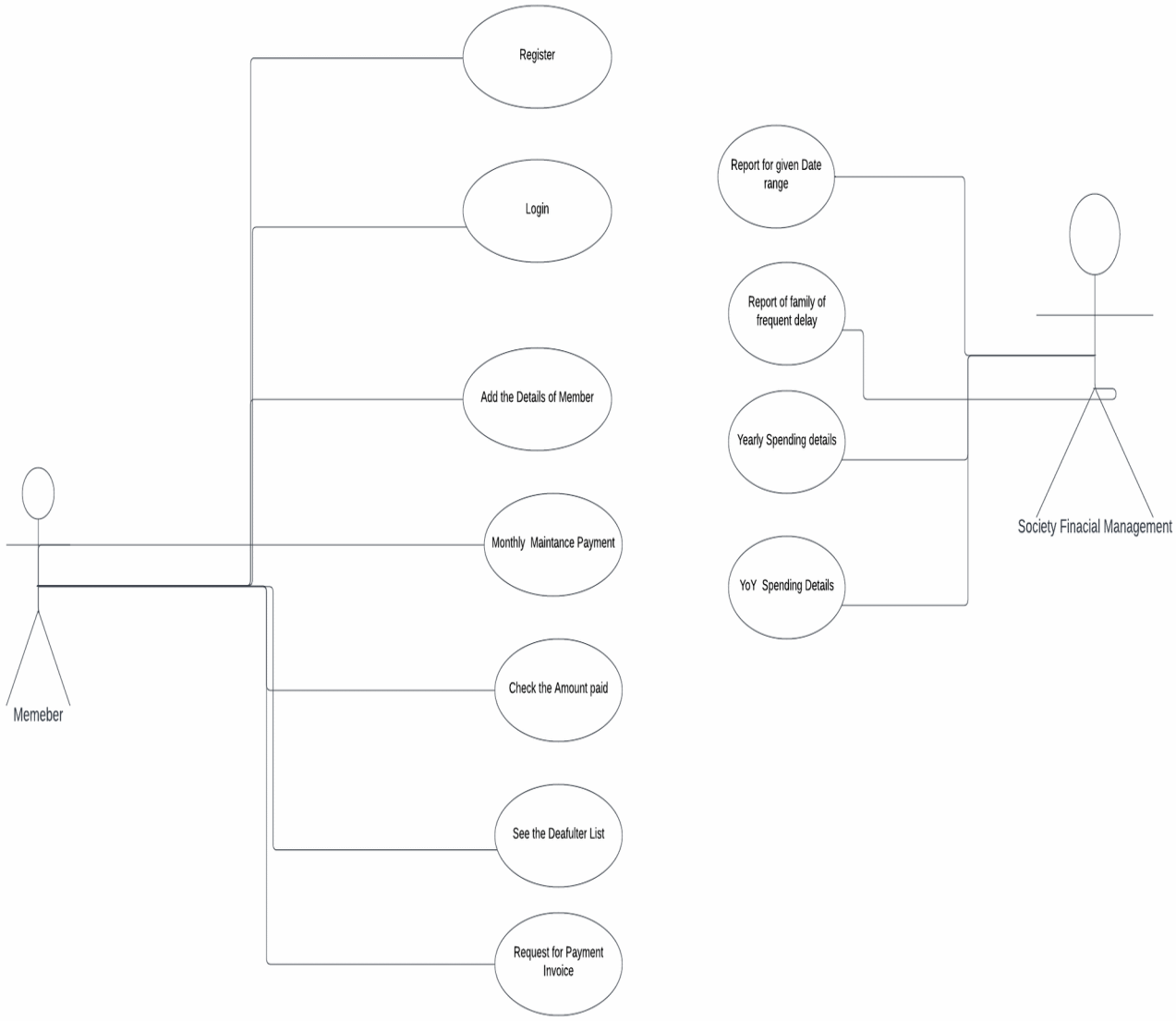
Presentation Layer: User interfaces including web, mobile, or desktop applications.

Application Layer: Business logic and core functionalities like user authentication, family management, payment tracking, reporting, and activity logging.

Data Access Layer: Handles interactions with the database.

External Interfaces: Integration with external services like email providers, payment gateways, and other third-party APIs.\


register/login use case
Register
enter details/email/password
validate
save user data
confirm account creation
login
insert email/password
validate
grant access to user dashboard
new member
existing member
system


Register
Login
Add the Details of Member
Monthly Maintance Payment
Check the Amount paid
See the Deafulter List
Request for Payment Invoice
Memeber
Report for given Date range
Report of family of frequent delay
Yearly Spending details
YoY Spending Details
Society Finacial Management

Class Diagram:

User: Manages user accounts and authentication.

Family: Stores information about families living in the society.

Payment: Tracks monthly maintenance payments and fines for delayed payments.

Expense: Manages society expenses.

ActivityLog: Logs activities within the system for auditing purposes.

Report: Generates various reports as per user requirements.

Sequence Diagram - Monthly Maintenance Payment Reminder:

Shows the interaction between the system and users for generating monthly maintenance payment reminders.

Component Diagram:

User Interface Components: Interfaces for user interaction.

Application Components: Business logic components for managing families, payments, expenses, etc.

Data Access Components: Handles database interactions.

External Interfaces: Integrations with external services.

Security Components: Implements security measures like authentication and access control.

Deployment Diagram:

Illustrates how system components are deployed across different nodes.

Shows the distribution of components on servers, clients, and external services. This high-level architecture provides a structured framework for developing the Society Financial Management system, ensuring scalability, maintainability, and reliability.