Application Name :- Electricity Bill Management

Group No. :- 16

Non- Function Requirement implemented :

1. Usability

- Clear UI Forms:The registration, login, bill viewing, and complaint forms should have an intuitive interface, making it easy for users to interact with the system.
- Error Handling & Feedback: Proper error messages are displayed when there is a failure in login, registration, bill payment, or complaint registration. This improves user experience by providing clear information on what went wrong.
- Confirmation Messages: Upon successful actions (e.g., registration, bill payment), the system gives clear success messages.

2. Performance

- Efficient Data Retrieval: The system should fetch bill history, complaint history, and pending bills efficiently, especially as the database grows.
- Response Time: The system should ensure quick responses for actions like login, bill payment, and data retrieval, ensuring minimal latency for the user.

3. Scalability

- Database Scalability: The system should be designed to handle a growing number of customers, bills, complaints, and transactions. This includes optimizing database queries and indexes for fast access.
- Modular Code Structure: The ability to easily add more modules or functionalities (such as adding new complaint categories or payment methods) ensures the system is scalable for future requirements.

4. Reliability

- Data Integrity: Consistent management of the `Customer`, `Login`, `Bill`, and `Complaint` tables ensures that records are stored accurately, and there is no loss or corruption of data.
- Backup and Restore: Soft delete functionality ensures that customer accounts are not permanently deleted, enabling reliable account recovery and preventing accidental data loss.

5. Security

- Password Encryption: The login table should store passwords securely, ideally in an encrypted format, to prevent unauthorized access.
- Unique Email Validation: Ensuring that each email address is unique in the system prevents conflicts and ensures data integrity.
- Access Control: Administrator accounts have exclusive access to certain functionalities, ensuring proper role-based access control.

6. Maintainability

- Exception Handling: Handling database-related exceptions across various processes ensures that the system is robust and errors are caught gracefully. This allows for easier troubleshooting and maintenance.
- Code Modularity: The system's structure allows for easy updates and maintenance of individual modules (registration, billing, complaints, etc.).

7. Data Consistency

- Atomic Transactions: Operations like bill payment and complaint registration should be atomic to ensure data consistency. For instance, if a bill payment fails halfway through, the system should roll back to maintain consistent data.
- Status Management: The status (active/inactive) fields in the `login` and `bill` tables ensure that records are always consistent and up-to-date.

8. Extensibility

- Easy Integration of New Features: New features like additional payment methods, complaint categories, or even different types of user roles (e.g., super admin) should be easily integrated without major system changes.

9. Error Handling and Robustness

- Graceful Degradation: When the system encounters an error (e.g., database connection issue), it should handle the error gracefully and provide a meaningful message to the user, rather than crashing.
- Input Validation: Ensuring that user inputs (e.g., email format, complaint details) are validated correctly prevents unexpected errors during data processing.

10. Availability

- Minimal Downtime: The system should aim for high availability, ensuring that customers can register, view bills, and submit complaints at any time.
- Administrator Access: Admin accounts should always have access to maintain the system even during peak loads or certain failures.