Software Design Specification

for

Electricity Billing System

Version <2.0>

Group No.: <2>

Mohammed Yousef Mohammed Abdulkarem	**1221305727**
Mohammed Aamena Mohammed Abdulkarem	**1221305728**
Muhammad Faiz bin Ilyasa	**1221309435**
Chua Cheng Zong	**1221309173**

Date: **14 JAN 2025**

Table of Content

Table of Figures

Table of Tables

Revisions

Version	Primary Author(s)	Description of Version	Date Completed
1.0	Mohammed Yousef Mohammed Abdulkarem Mohammed Amena Mohammed Abdulkarem Muhammad Faiz bin Ilyasa Chua Cheng Zong	Generated the Project Group & Plan, System Overview, Scenario-Based Models, Class Models, and Behavioural & Flow Models.	08/12/2024
2.0	Mohammed Yousef Mohammed Abdulkarem Mohammed Amena Mohammed Abdulkarem Muhammad Faiz bin Ilyasa Chua Cheng Zong	Created Data Design Diagrams, Architecture Design Diagrams, Interface Design Diagrams, Component Design Diagrams and Deployment Design Diagrams.	12/01/2024

1 System Overview

1.1 Description

The Electricity Billing System is a comprehensive solution aimed at improving the management of electricity services. It facilitates effective communication and coordination among key stakeholders, including customers, utility providers, support administrators, and staff. The system ensures secure user registration and authentication, accurate billing calculations, smooth payment processing, and comprehensive customer support. Additionally, it features a robust feedback mechanism to continually improve service quality and ensure customer satisfaction.
The system prioritizes data integrity, security, and operational efficiency, providing an intuitive interface that simplifies service management, minimizes manual efforts, and ensures reliable performance. Utilizing advanced technology, the Electricity Billing System sets a new standard for operational excellence, offering a flexible solution that evolves to meet the dynamic needs of both utility providers and their customers.

1.2 Actors

The system integrates multiple roles, each with specific responsibilities to ensure smooth operation. Customers can create accounts, log in securely, reset passwords, and update their personal information. They can view detailed billing information, make payments using their preferred methods, receive receipts, provide feedback, and report any issues or concerns.

Utility providers manage critical operational tasks, such as logging into the system to update meter readings, adjusting tariff rates in response to changes in tax or regulatory policies, generating accurate monthly bills, and tracking overdue payments. When overdue payments exceed the given grace period, utility providers will set penalty fee for the associated bill and coordinate with the staff to notify the customer of the final payment deadline and initiate service disconnection if necessary.

Support administrators prioritize customer satisfaction by resolving issues related to billing, payments, and service concerns. They review billing data to identify discrepancies, analyse customer feedback to identify areas for system improvements, and respond promptly to customer inquiries, ensuring efficient issue resolution and maintaining a high level of service quality.

Staff members manage administrative and monitoring functions, such as tracking overdue bills for verification and timely payment, analysing electricity usage trends to assess consumption patterns, and maintaining customer account records to ensure accuracy and operational efficiency.

This well-structured system ensures that all participants fulfil their roles effectively, creating a seamless and transparent process for electricity service delivery. By consolidating all operations into one platform, the Electricity Billing System fosters efficient communication, accurate billing, and improved service delivery, making it an essential tool for the electricity sector.

Table 1: Actors and respective use cases

Actor	Use Cases
Customer	System Registration, Log in to the System, Reset Password, View Bill Details, Make a Payment and Receive Payment Receipt, Submit Feedback, and Submit Issues.
Utility Providers	Log in to the System, Update Meter Readings and Tariffs, Generate Monthly Bills, and Track Overdue Bills.
Support Admin	Log in to the System, View Bill Details, Resolve Customer Issues, View and Analysis Feedback
Staff	Log in to the System, Track Overdue Bills, Usage Monitoring, and Manage Customer Accounts.

1.3 Assumptions and Dependencies

Assumptions:

User Access and Connectivity

- Users will have access to devices capable of running modern web browsers and will have stable internet connectivity.
- Customers will provide valid email addresses, meter number and mobile numbers for notifications and account management.

Data Availability and Accuracy

- Utility providers will provide accurate and up-to-date meter readings and tariffs for accurate billing.
- Customers are expected to provide valid personal information.
- The system will automatically check the input data for validation, such as meter readings and customer information, to minimize errors in billing.

Scalability Requirements

The system will support up to 1000 users initially. As the user base grows, scaling the system will require infrastructure upgrades, such as database optimization and increased server capacity, to ensure continued performance and reliability.

Compliance with Standards

The system will comply with relevant legal and industry regulations, including:

- Data Privacy Laws: The system will follow GDPR to ensures that the system is in compliance with data protection regulations, protecting personal data and the privacy rights of users. This includes secure data storage, processing, and consent management to avoid legal violations.
- Payment Security Standards: Ensures the realization of compliance with the Payment Card Industry Data Security Standard (PCI DSS) for the safe handling of credit card information, helping to protect payment data from fraud or breaches during transactions.
- Utility Service Requirements: Ensures that the system meets regional standards and regulations of electricity services provided, regarding billing accuracy, service reliability, and customer protection.

Dependencies:

Integration with Payment Gateways

The system payment functionality relies on secure and reliable third-party payment services to process transactions efficiently while maintaining industry standards for data protection.

Accurate Tariff and Meter Data

The system's billing accuracy is dependent on regular updates of tariff rates and meter readings provided by utility providers. Any delays or inaccuracies in this data will directly affect billing and payment processes.

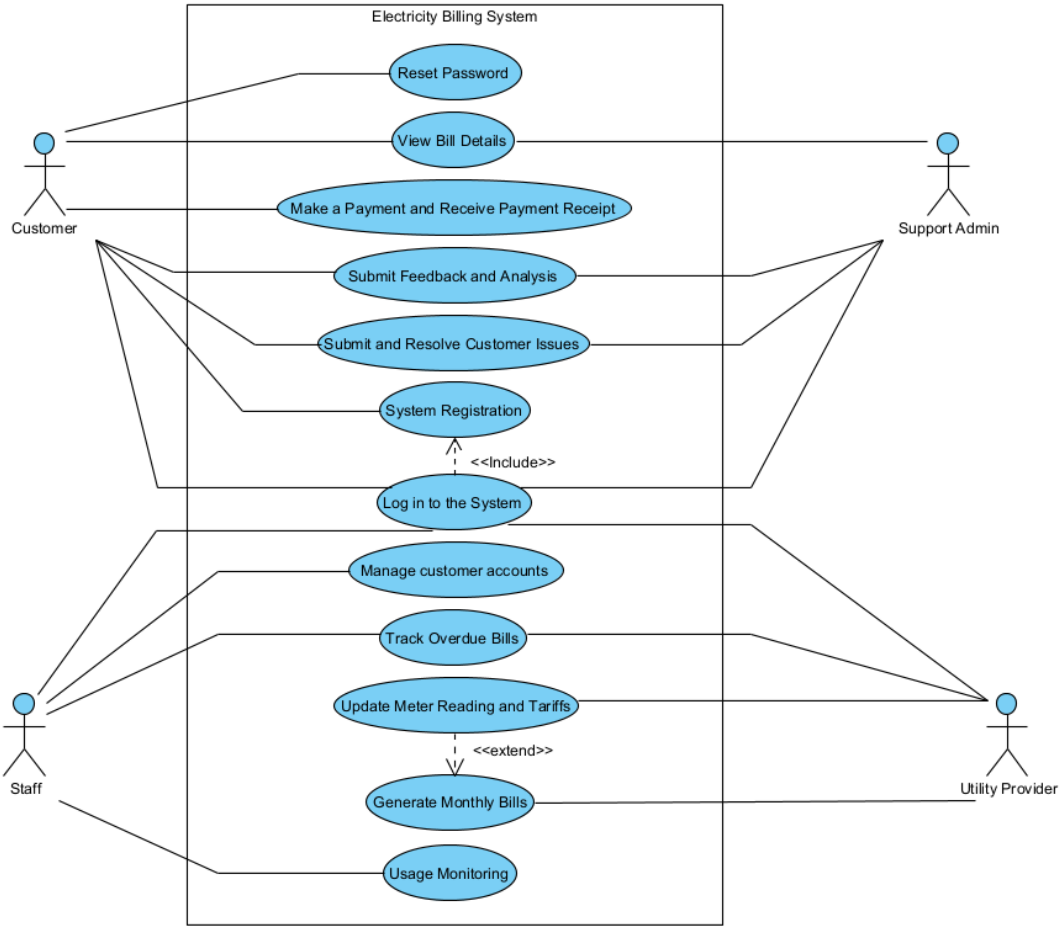
Database Management System

Microsoft Access, a robust and scalable relational database system, will manage all system data effectively. It ensures secure storage, fast retrieval, and the capability to scale as the system grows, supporting seamless operations and reliability.

Ongoing Maintenance and Support

Continuous system updates, performance monitoring, and regular maintenance will be essential to address emerging needs, improve functionality and support best system performance.

1.4 Use Case Diagram



2 Use Cases

2.1 Use Case Diagram

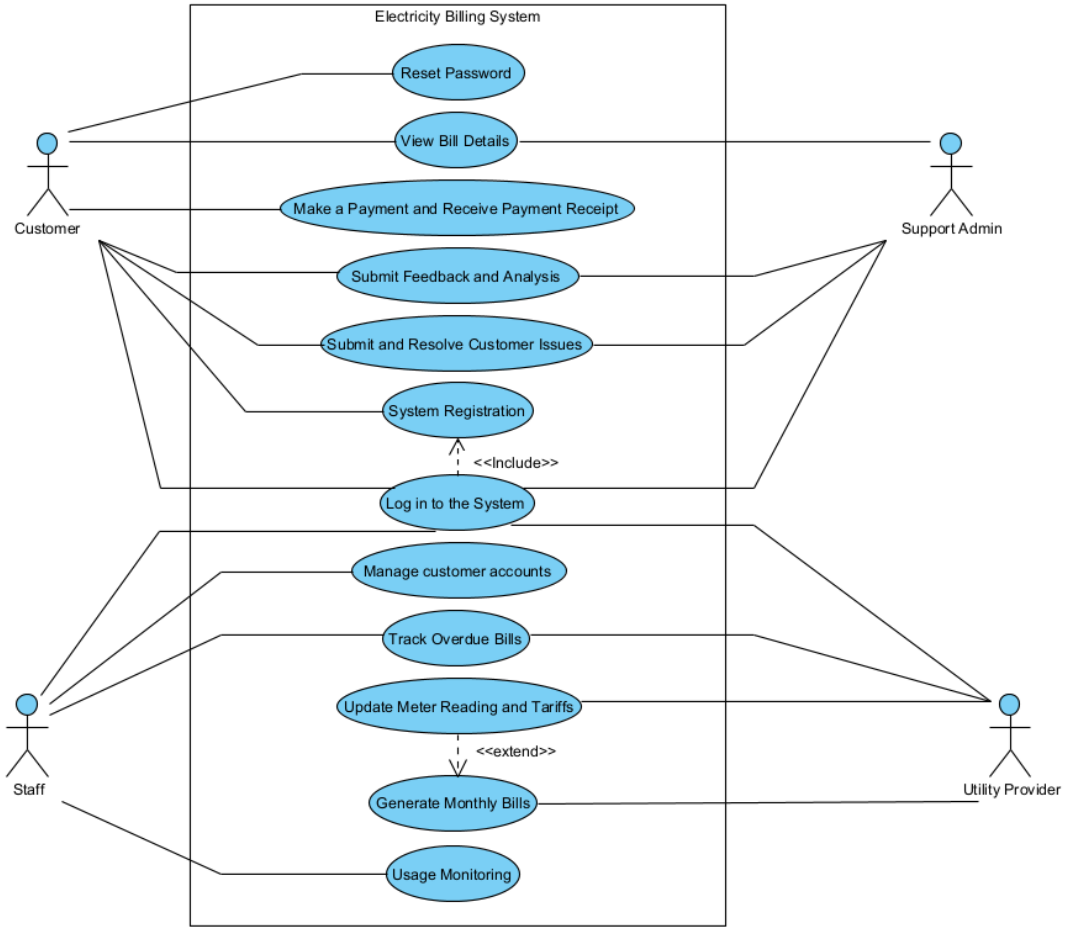


Figure 1: Use case diagram

2.2 Actor 1: Customer

2.2.1 Use Case 1: System Registration

Actors: Customer
Description:

The system allows new customers to register by providing their meter ID to connect the account with and their personal information. Then the system will verify the provided information by sending a code to the customer's email it also checks with the system database for any duplication (e.g., email, meter number). The Customer account is created and stored in the system database if no duplication has been found. Upon successful registration, the user can easily access the system by logging in to the system.

Preconditions: N/A
Main Flow:

1. The customer navigates to the homepage and selects the "New Customer Registration" option.
2. The customer enters their meter ID.
3. The customer clicks the "Next" button, so the system validates the provided ID.
4. The customer enters the required details, including:
    - Name
    - Email
    - Phone number
    - Address
    - Username
    - Password
5. The customer clicks the "Register" button, which submits the registration form.
6. The system validates the provided information to ensure they are complete and correctly formatted.
7. The system checks for any duplicate accounts based on the email or meter number.
8. If no duplicate records are found, the system creates a user account and stores the customer information in the database.

Postconditions:

- A new customer account is successfully created and securely stored in the system database.

Alternative Flows:

- **Incomplete or Invalid Information:**
    - If required fields are incomplete or contain invalid data, the system displays an error message "Please complete all required fields or correct invalid entries".
    - The system marked the fields requiring correction for the customer's attention.
    - The customer updates the information and resubmits the registration form.

- **Duplicate Account Detection:**
    - If a duplicate account is detected based on the provided email or meter number, the system displays a message such as "An account with this email or meter ID already exists".
    - The customer is prompted to provide unique information again or use a different email or meter number to proceed.

Assumptions:

- Users have a stable internet connection while attempting to log in.
- The system is operational during the customer's registration attempt.
- The customer provides accurate and verifiable personal or organizational details for registration.

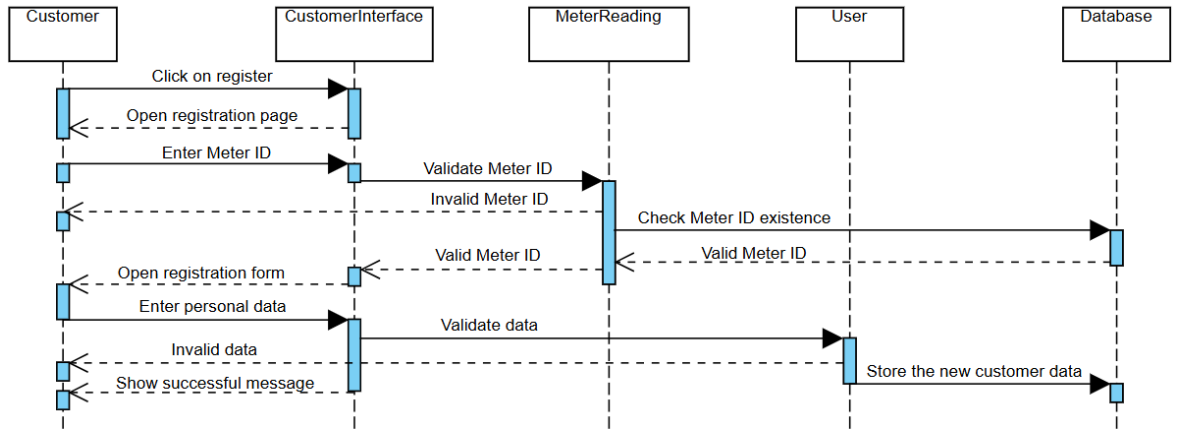


Figure 2: System registration sequence diagram

2.2.2 Use Case 2: Log in to the System

Actors: Customer, Staff, Support Admin, and Utility Provider.

Description:

This use case enables the user to log into the system using valid credentials. The system will open the respective System dashboard based on the user role, ensuring that users interact only with the needed functionalities based on their role.

Preconditions:

- The system and its database must be operational.
- The user must possess a registered account.

Main Flow:

1. The user accesses the system's login page.
2. The user inputs their registered username and password into the corresponding fields.
3. The user submits the login form by clicking the "Login" button.
4. The system verifies the provided credentials against the database.
5. If the credentials are correct. The user is redirected to their personalized dashboard or homepage, where features and options are tailored to their role.

Postconditions:

- The user successfully accesses the system with functionality appropriate to their role.
- A record of the login is stored in the system for tracking purposes.

Alternative Flows:

- **Invalid Credentials:**
  If the entered username or password is incorrect:
  - The system returns an error "Invalid username or password."
  - The user must re-enter their credentials.
- **Forgotten Password:**
  - Click the "Forgot Password" option.
  - Provide their registered email to receive password reset instructions.
  - Follow the reset process and attempt to log in again.

Assumptions:

- Users have a stable internet connection while attempting to log in.
- The authentication system and database are functioning correctly.
- Role-based permissions for users are accurately configured within the system.

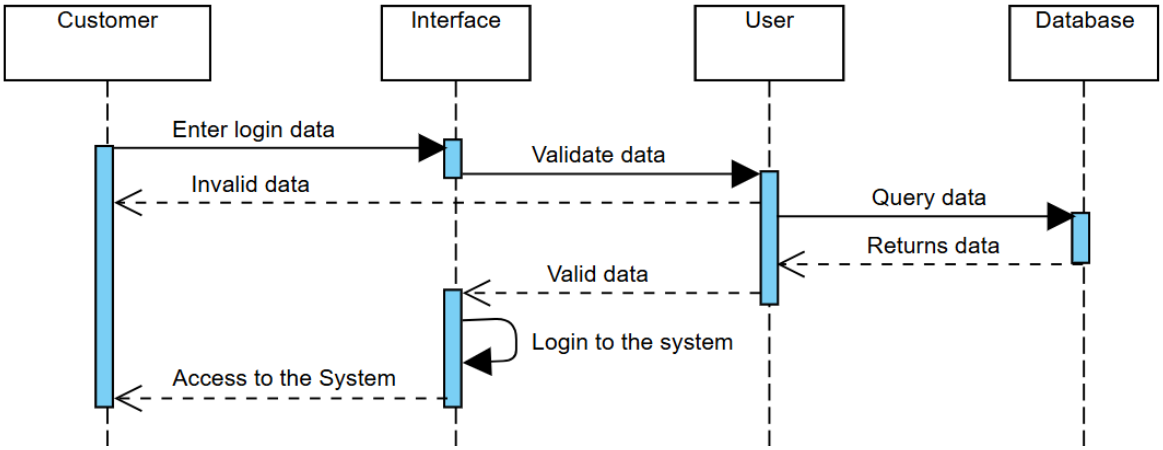


Figure 3: Log in to the system sequence diagram

2.2.3 Use Case 3: Reset Password

Actors: Customer

Description:

This use case enables the customer to reset their password either if they forget it or prefer to. The system will ask for the customer's email, so it sends the reset instructions to it. Further on the database will be updated so the customer can use their new password.

Preconditions:

- The system and its database must be operational.
- The user must possess a registered account.

Main Flow:

1. The customer selects the forget password option or navigates to the settings and selects the reset password option.
2. The system will ask the user to provide their email address, so it can send the instructions to it.
3. The customer will follow the instructions and set their new password.
4. The system will update the old password and store the new one in the database.

Postconditions:

- The user successfully updated their password.
- A record of the operation is stored in the system for tracking purposes.

Alternative Flows:

- **If the new password matches the old one:**
  - The system will display an error message “Your new password can’t be the same as the old one”.
  - The customer must enter another password.

Assumptions:

- Customers have a stable internet connection while resetting their password.
- The customer provides a different password than their old one.

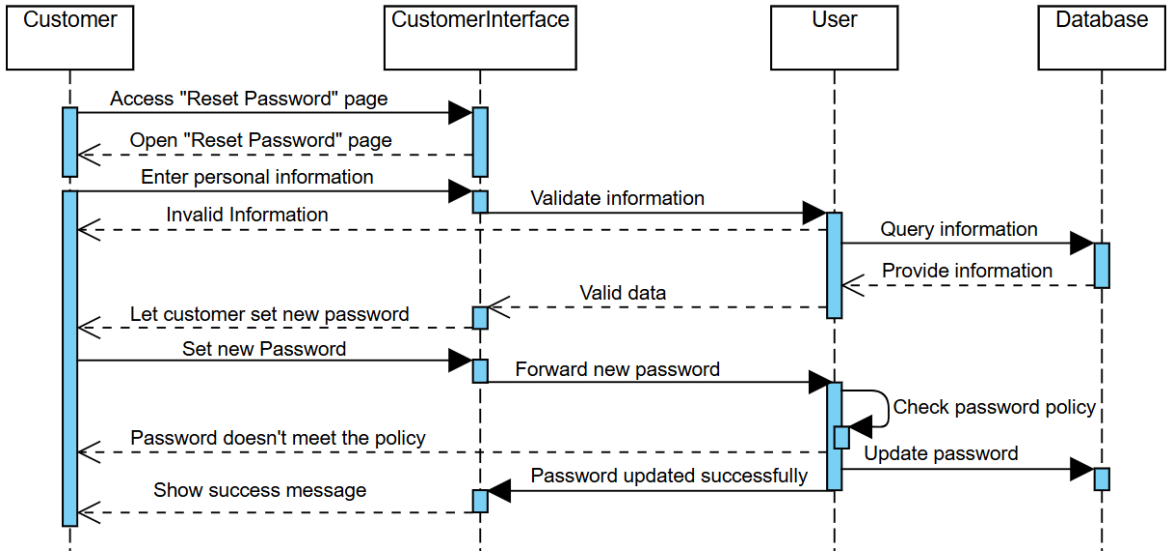


Figure 4: Reset password sequence diagram

2.2.4 Use Case 4: Make a Payment and Receive Payment Receipt

Actors: Customer

Description:

This use case describes the process of making a payment for electricity bills and receiving a digital receipt.

Preconditions:

- The customer must be registered and logged into the system.
- The customer must have at least one active bill.

Main Flow:

1. The customer navigates to “Make a Payment” section.
2. The system retrieves a list of all the customer's active bills and displays them to the customer.
3. The customer selects specific bills to pay.
4. The customer selects the preferred payment method (e.g., credit card, bank transfer).
5. The customer clicks the "Make Payment" button to proceed.
6. The system redirects the customer to the selected payment gateway.
7. The customer completes the payment transaction through the payment gateway.
8. Upon successful payment confirmation from the gateway, the system updates the status of the selected bills to "Paid".
9. The system generates and displays a digital receipt to the customer, confirming the successful payment

Postcondition:

- The selected bills are marked as "Paid" in the system.
- The customer receives a digital receipt confirming the payment.

Alternative Flow:

Payment Failure:

- If the payment fails at the gateway, the system displays an error message to the customer.
- The bill status remains unchanged.

Assumptions:

- The customer is able to access the system with a stable internet connection.
- The payment gateway integration is stable and reliable.
- The system can handle various payment methods and currencies.
- The system ensures the security and confidentiality of customer payment information.

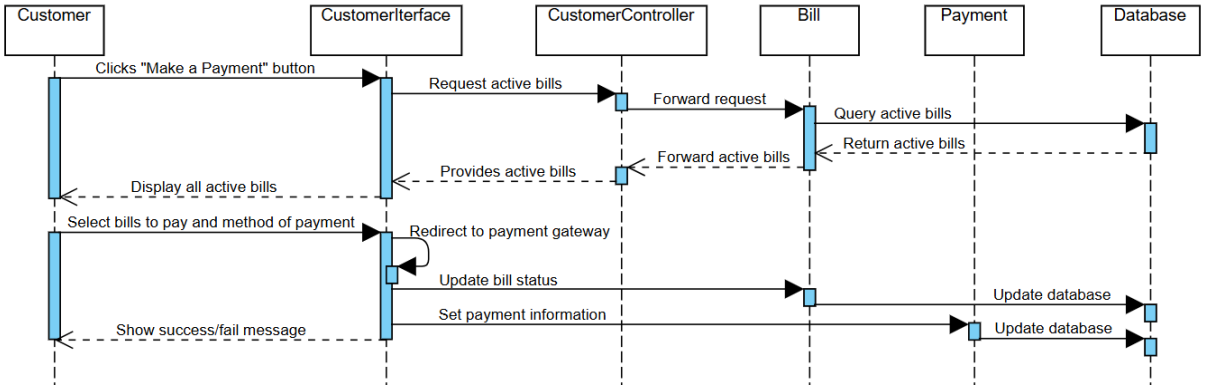


Figure 5: Make a payment and receive payment receipt sequence diagram

2.2.5 Use Case 4: View Bill Details

Actors: Customer

Description:

This use case allows the customer to view detailed electricity billing information, such as outstanding balances, consumption records, and payment history, through a user-friendly interface.

Preconditions:

- The customer must have a registered account.
- The customer must be logged into the system.
- Relevant billing data must exist and be accessible in the system.

Main Flow:

1. The customer navigates to the "View Bills" section in the system interface.
2. The customer clicks on the specific bill for which they want to view details.
3. The system retrieves billing details based on the selected bill.
4. The customer reviews the displayed billing information:
    - Outstanding balance.
    - Total electricity consumption.
    - Payment due date.

Postconditions:

The customer successfully views their bill.

Alternative Flows:

No bill data found

The system sending the customer a message, "No billing information available." in case of no billing data available at its end.

Error Retrieving Bill from System

If there is an error during bill retrieve the system displays a message (e.g., "Unable to retrieve bill details. Please try again later.").

The error is logged, and the customer arise issue to the support administrator for resolution.

Assumptions:

- The customer is able to access the system with a stable internet connection.
- Billing data is updated in real-time or at regular intervals for accuracy.
- The customer account is active, authorized, and not subject to any restrictions.

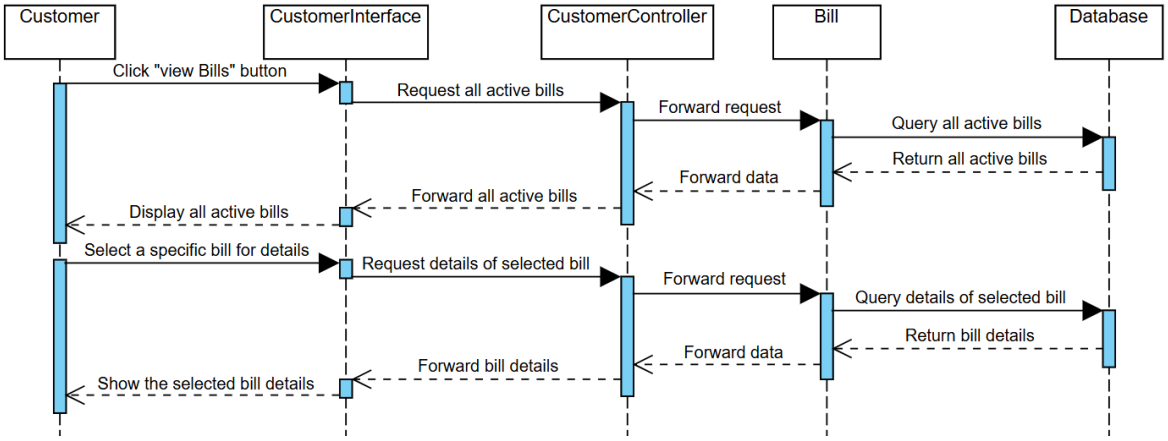


Figure 6: View bill details sequence diagram

2.3 Actor 2: Utility Provider

2.3.1 Use Case 5: Track Overdue Bills

Actors: Utility Provider and Staff.
Description:
This use case allows the utility provider to track users with overdue bills marked by the staff in the system. The utility provider will take necessary actions such as penalizing the users with overdue bills and disconnecting their electricity.
Preconditions:

- Relevant billing data must exist and be accessible in the system.
- The user must be logged in to a utility provider account.
- The system and its database must be operational.

Main Flow:

1. The utility provider accesses the “Track Bills” page.
2. The utility provider can generate a list of overdue bills.
3. The utility provider calculates the penalty and disconnects the penalized customer’s electricity.
4. The utility provider will update the system’s database.
5. The customer will receive a notification on the update.

Postconditions:

- The status of the bill is updated to penalized.
- Customer is notified about the update and the penalty details.

Alternative Flows:

No overdue bills found:

- If the customer account does not have any overdue bills, the system displays, "No overdue bills found".

Assumptions:

- The system is up and running, and capable of generating a list of customer accounts.
- The system can automatically identify overdue bills based on the pay date.
- The system's database is functioning properly and can update new data.

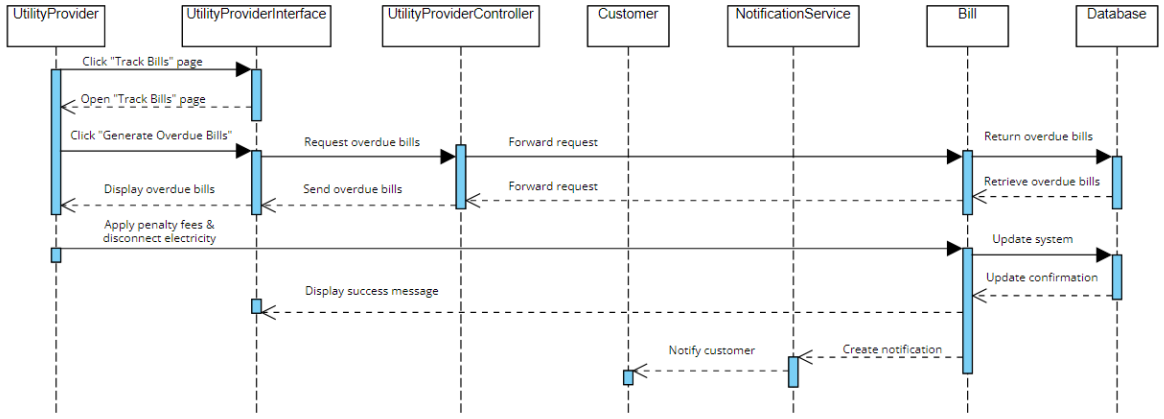


Figure 7: Track overdue sequence diagram

2.3.2 Use Case 6: Update Meter Readings and Tariffs

Actors: Utility Provider

Description:

This use case enables the utility provider to update meter readings and tariffs. The system then validates the data, ensuring that the data given applies with the rules, then updates the database with the new readings and tariff rates.

Preconditions:

- The user must possess a utility provider account.
- The system and its database must be operational.

Main Flow:

1. The user accesses the “Meter Readings and Tariffs” page section.
2. The user accesses the “Update Rates” section.
3. The user enters updated readings or tariff rates for the specified period.
4. The user submits the updated data.
5. The system verifies the provided data.
6. If the credentials are correct:
    - The system updates database with the new provided data.
    - A success message will be displayed.

Postconditions:

- The user successfully updated the database with new reading or rates.
- A record of the update is stored in the system for tracking purposes.

Alternative Flows:

- **Invalid Credentials:**
  If the entered data is invalid:
    - The system returns an error message “Invalid data."
    - The user must enter the updated data again.

Assumptions:

- Users have a stable internet connection while attempting to log in.
- The authentication system and database are functioning correctly.
- The user possesses a utility provider account.

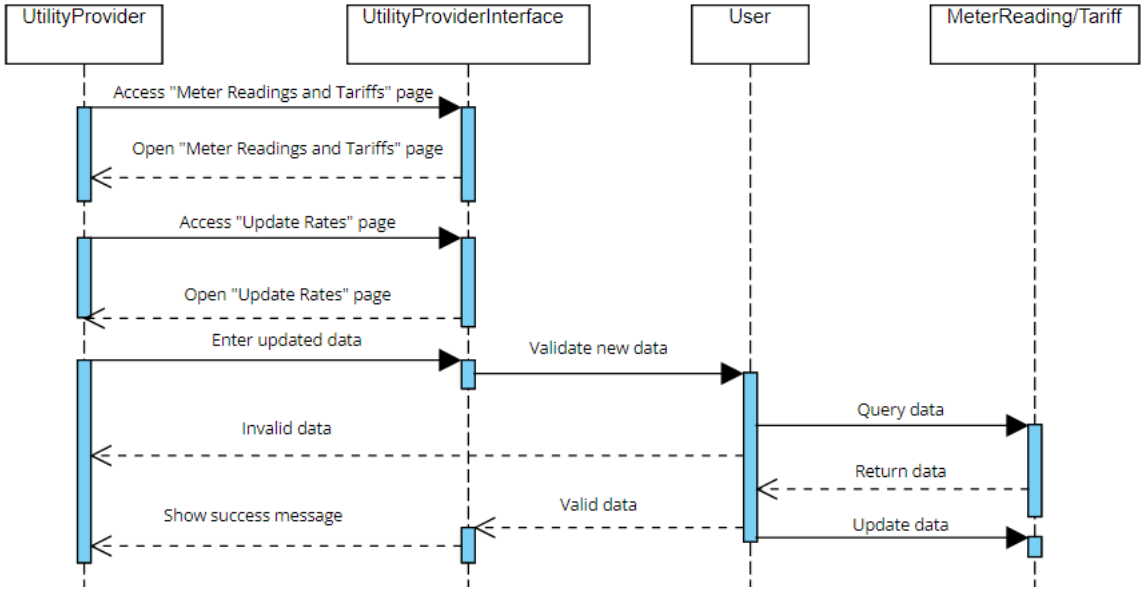


Figure 8: Update meter reading and tariffs sequence diagram

2.3.3 Use Case 7: Generate Monthly Bills

Actors: Utility Provider

Description:

This use case generates customer's monthly bill. This system is automatically scheduled, it will generate a digital copy of the bill and attaches it to the customer's account, the system then notifies customers via email or SMS.

Preconditions:

- The user must possess a valid customer account.
- The system and its database must be operational.

Main Flow:

1. On a prescheduled day each month, the system will process each customer's monthly bill generation.
2. The system will retrieve the customer's data on meter readings, tariffs, and payment history.
3. The system then calculates the bill amount based on usage and current tariff rates.
4. The system generates a detailed bill for each customer for the current billing period and saves it in database for record purposes
5. The system generates a digital copy of the bill and attaches it to the customer's account.

Postconditions:

- The customers will be notified via email or SMS about their monthly bills.
- The bill will be accessible by customers through the system.

Assumptions:

- The system and database are functioning correctly.

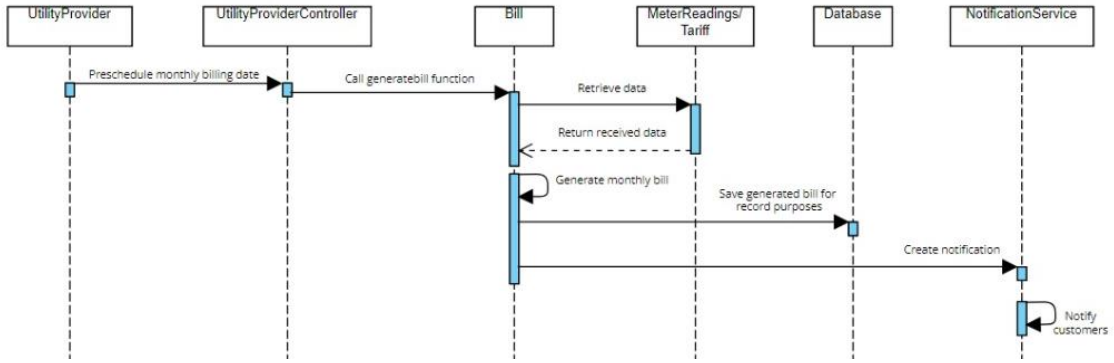


Figure 9: Generate monthly bills sequence diagram

2.4 Actor 3: Support Admin

2.4.1 Use Case 8: View Bill Details

Actors: Support Admin

Description:

This use case allows the Support Admin to view detailed electricity billing information for a customer, such as outstanding balances, consumption records, and payment history, through a system interface. The Support Admin can assist the customer in resolving billing-related issues or inquiries.

Preconditions:

- The customer must have a registered account.
- The Support Admin must have login to the system and have the necessary permissions to access customer billing data.
- Relevant billing data must exist and be accessible in the system.

Main Flow:

1. The Support Admin logs into the system with appropriate credentials.
2. The Support Admin navigates to the "View Customer Bill" section in the system interface.
3. The System will retrieve the list of customer bills.
4. The Support Admin searches for the customer's account using unique identifiers (e.g., bill ID, customer name, meter number).
5. The system retrieves the customer's billing details.
6. The Support Admin reviews the displayed billing information:
   - Outstanding balance.
   - Total electricity consumption.
   - Payment due date.
   - Payment history.
7. The Support Admin assists the customer with their billing inquiry or issue.

Postconditions:

- The Support Admin successfully views the customer's bill details.
- The Support Admin resolves the customer's billing-related issue or provides necessary information to the requested concerned.

Alternative Flows:

No bill data found

If no billing data is found for the customer, the system displays a message: "No billing information available for this customer."

Error Retrieving Bill from System

If there is an error during bill retrieve the system displays a message (e.g., "Unable to retrieve bill details. Please try again later.").

Assumptions:

- The Support Admin has access to the system with appropriate permissions.
- The system is operational, and billing data is updated in real-time or at regular intervals for accuracy.
- The customer account is active, authorized, and not subject to any restrictions.

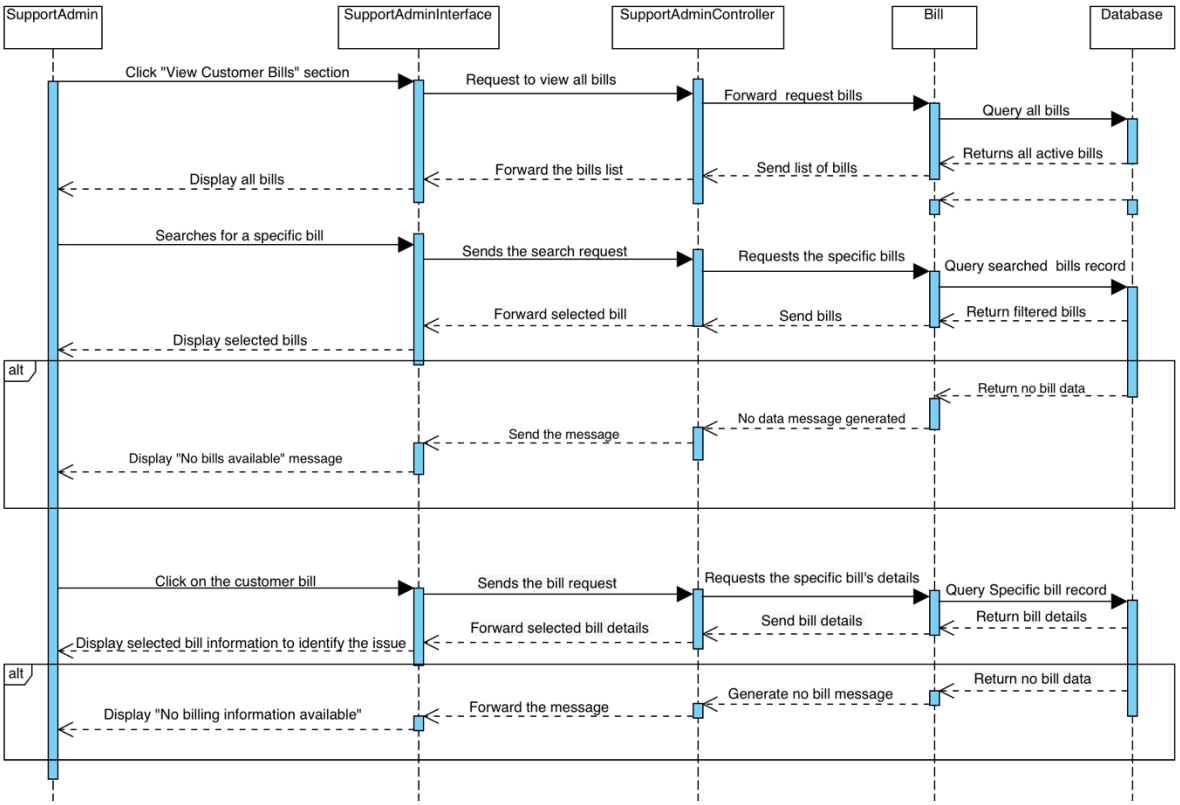


Figure 10: View customer bills details sequence diagram

2.4.2 Use Case 9: Submit and Resolve Customer Issues

Actors: Customer and Support Admin

Description:

Allows customers to report problems related to electricity services and track their resolution progress. Support members review, manage, and resolve these issues while providing timely updates to customers. The system assigns a unique ticket ID to each issue, ensuring efficient tracking and resolution of all reported concerns.

Preconditions:

- The customer must have a registered account.
- The customer must be logged into the system.
- The Support Admin must have login to the system to get necessary permissions to access and resolve customer issues.

Main Flow:

1. The customer navigates to the "Support" section.
2. The customer completes the form by describing the details of issue with Attachments (if applicable).
3. The customer clicks the “Submit” button to submit the form.

4. The system generates a unique ticket ID for the issue.
5. The system sends a confirmation message to the customer, including the ticket ID and assign the issue status “In progress”.
6. The system assigns the issue to the support admin.
7. The Support Admin reviews the customer issue.
8. The Support Admin takes the appropriate action to resolve the issue.
9. The Support Admin updates the status of the issue and add comments regarding to the issue.
10. The customer receives email notifications regarding the issue status when the issue has been resolved.

Postconditions:

- A unique ticket ID is generated and assigned to each submitted issue.
- The system continuously tracks and updates the status of the issue by support Admin.
- The customer is notified via email about any updates or resolutions related to their issue.
- The issue is marked as "resolved" once the Support Admin has successfully addressed it.

Alternative Flows

- **Incomplete or Invalid Issue Form:**
  If the customer submits an incomplete issue form with some required fields not filled in, the system generates an error message requesting the customer to fill in the required fields. The customer corrects the form and resubmits it.
- **Submission Failure:**
  If a system error prevents the issue from being submitted, the system displays an error message "Unable to submit the issue. Please try again later".

Assumptions:

- The customer and support admin has internet access and is able to log into the system successfully using valid credentials.
- The system is online and functioning during the time the customer issue submission and support admin resolved it.
- Support admins have appropriate permissions and sufficient training to handle and resolve the issues reported by their customers.
- Email services are working, and notifications can be sent successfully to the customer.

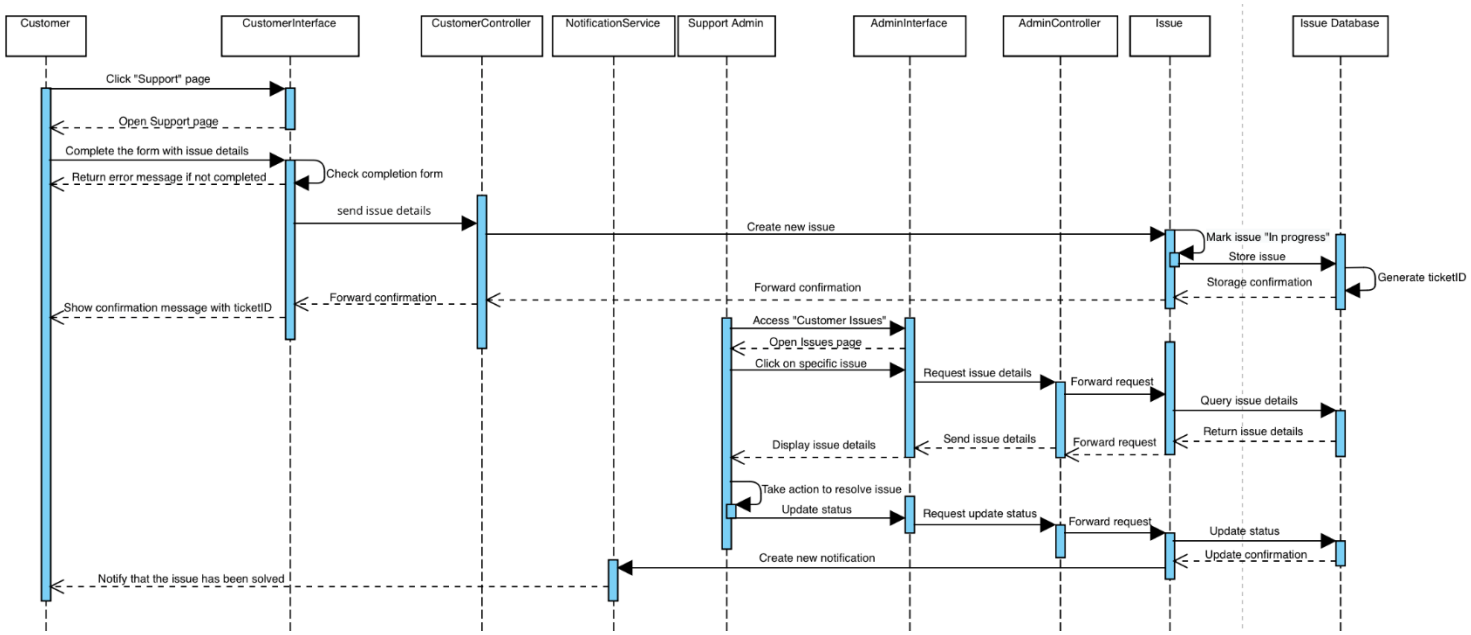


Figure 11: Submit and resolve customer issue

2.4.3 Use Case 10: View and Analysis Feedback

Actors: Support Admin

Description:

This use case focuses on enabling the Support Admin to access and analyse customer feedback about their experiences with the system. The system validates and stores feedback submitted by customers, allowing the Support Admin to filter and analyse the data to identify trends, issues, or areas for improvement.

Preconditions:

- The Support Admin must be logged into the system with necessary permissions to access and analyse customer feedback.
- The system is operational and able to receive feedback.
- The Support Admin must have necessary permission to access and analysis customer feedback.

Main Flow:

1. The Support Admin logs into the system.
2. The Support Admin clicks on “View Customer Feedback” to access submitted feedback.
3. Support Admin uses the search feature to filter feedback such as rating (e.g., 1-5 star).
4. The Support Admin analyses the feedback to identify trends, issues, or areas for improvement.
5. Support Admin takes actions based on the insights derived from the analysis.

Postconditions:

- The feedback is successfully stored in the database for analysis.
- The Support Admin can efficiently search, filter, and analyse feedback using the search feature.
- Insights gained from the feedback analysis can guide system improvements or other relevant actions.

Alternative Flows:

No feedback matches search filters

If no feedback matches the criteria, the system displays: "No feedback found for the selected filters.". Support Admin adjusts the filters and retries the search.

Assumptions:

- The Support Admin has internet access and can log into the system successfully.
- The system is up and running, and capable of processing feedback submissions.
- The system's database is functioning properly and can store feedback data.
- The Support Admin has the necessary permissions and training to review and act on feedback.

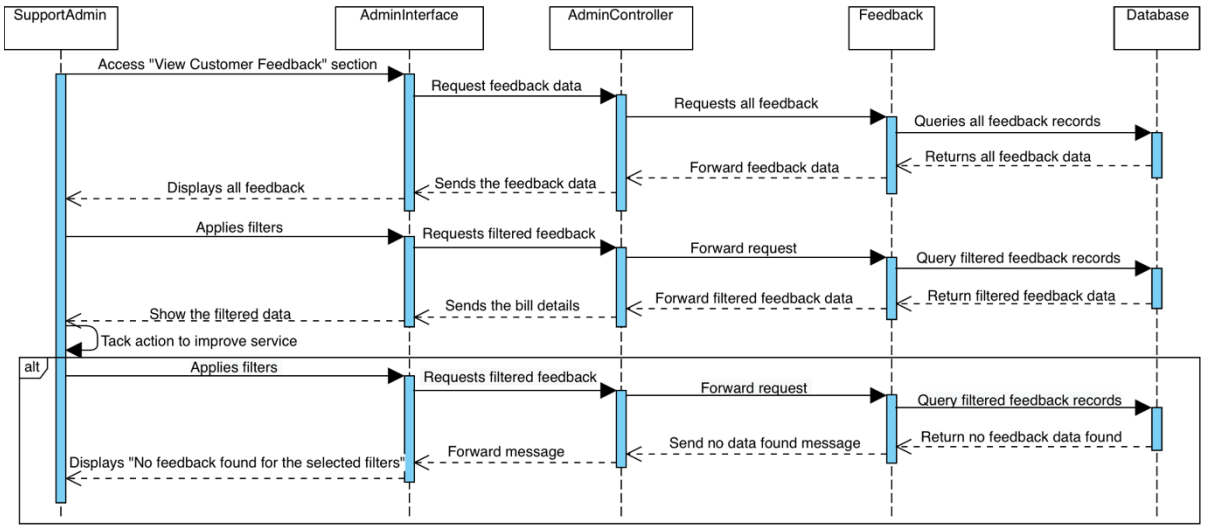


Figure 12: View and analysis feedback sequence diagram

2.5 Actor 4: Staff

2.5.1 Use Case 10: Manage Customer Accounts

Actors: Staff

Description:

This use case allows staff to monitor customer accounts and monitor each account details. The staff can go through a list of customer accounts or search for a specific customer to view each one of them. The staff can also update the customer's account details. The staff will update the system, and a notification will be sent to the customer.

Preconditions:

- The system must have an up-to-date database of accounts.
- The staff must be logged in to access customers account details.

Main Flow:

1. The staff accesses the "Manage Accounts" page.
2. The system displays a list of customer accounts.
3. The staff selects an account to view or update its details.
4. The staff will update the system's database.
5. The customer will receive a notification on the update.

Postconditions:

1. The customer details are updated.
2. Customer is notified about the update.

Alternative Flows:

- **Update details failed:**
  If an error occurred when updating the customer's account details, the system displays, "Update failed. Please try again.".
- **Invalid customer search:**
  If no accounts match the search, the system displays, "No accounts found.".

Assumptions:

- The system is up and running, and capable of generating a list of customer accounts.
- The system's database is functioning properly and can update new data.

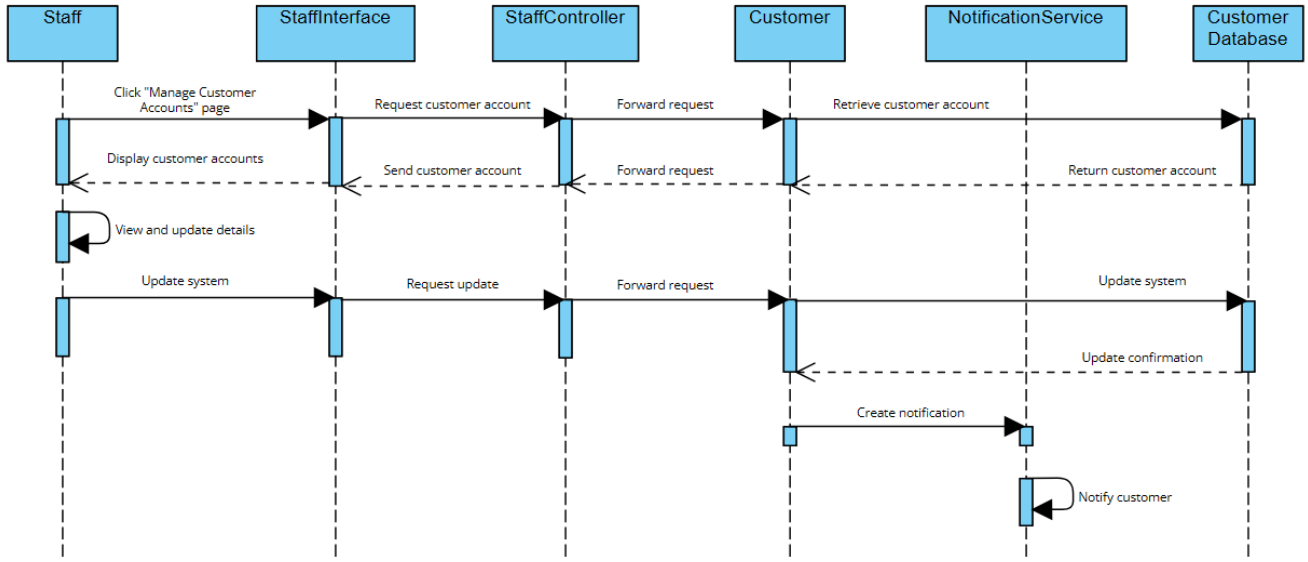


Figure 13: Manage customer accounts sequence diagram

2.5.2 Use Case 11: Track Overdue Bills

Actors: Staff, Utility Provider

Description:

This use case allows the staff to track overdue bills notified by the system based on the pay date. The staff will review the account details and payment histories to verify the overdue bill and will take necessary actions such as setting a deadline for the overdue payment. The system will be updated by the staff.

Preconditions:

- The system must have an up-to-date database of accounts and bill details.
- The staff must be logged in to access customers account details.

Main Flow:

1. The staff accesses the “Track Bills” page.
2. The system generates a list of customer accounts with overdue bills.
3. The staff selects an account to view its details.
4. The staff can review and verify account details and payment histories.
5. The staff sets the due date for the overdue bill.
6. The staff will update the system’s database.
7. The customer will receive a notification on the update.

Postconditions:

1. The status of the bill due date is updated.
2. Customer is notified about the update and the overdue bill details.

Alternative Flows:

- **No overdue bills found:**
  If the customer account does not have any overdue bills, the system displays, "No overdue bills found".
- **Invalid date:**
  If the date entered is invalid or empty, the system will display an error message such as “Please enter a valid date”.

Assumptions:

- The system is up and running, and capable of generating a list of customer accounts.
- The system can automatically identify overdue bills based on the pay date.
- The system's database is functioning properly and can update new data.

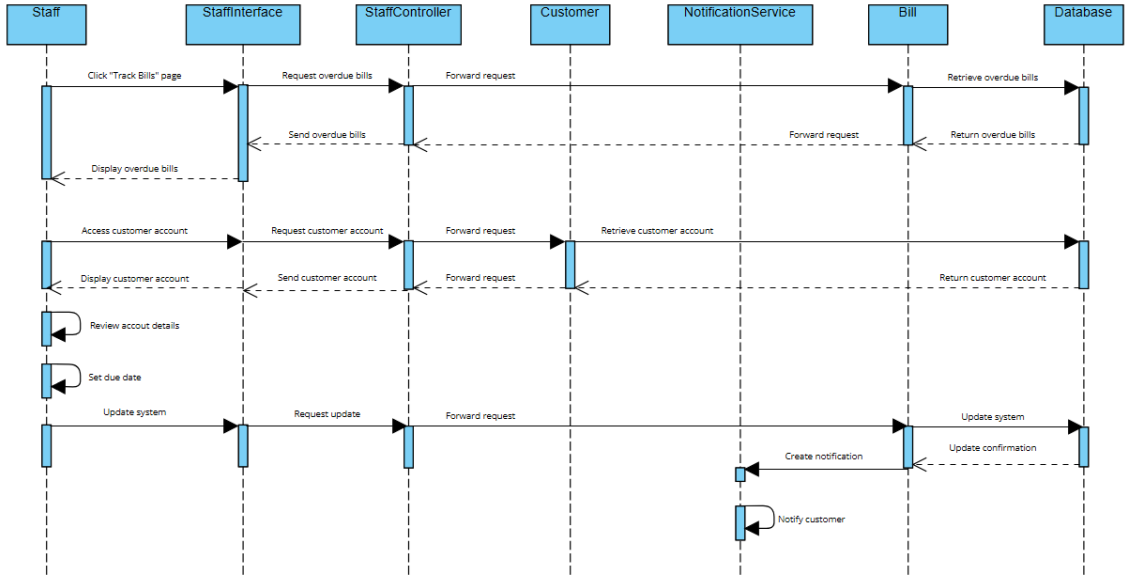


Figure 14: Track overdue bills sequence diagram

2.5.3 Use Case 12: Usage Monitoring

Actors: Staff

Description:

This use case allows the staff to monitor key-metrics related to the system usage such as total number of customers, total number of bills generated, and total number of bills paid. It also allows the staff to view the total electricity usage and average electricity usage. The system will display the trends for staff reference.

Preconditions:

- The system must have an up-to-date database with relevant metrics logged.
- The staff must be logged in to access the “Usage Monitoring” page.

Main Flow:

1. The staff accesses the “Usage Monitoring” page.
2. The staff selects the day and month to filter the metrics.
3. The system will display the usage metrics based on the filter.
4. The staff can view and monitor the analytics to see trends.

Postconditions:

- The staff receives a clear summary of required analytics.
- Data is display accurately and updated.

Alternative Flows:

- **No data available**
  If the data requested by the staff is not available, the system will display an error message, such as “No data is available”.

Assumptions:

- The system is up and running, and capable of generating key-metrics for the system’s usage.
- The system's database is functioning properly and is up to date.

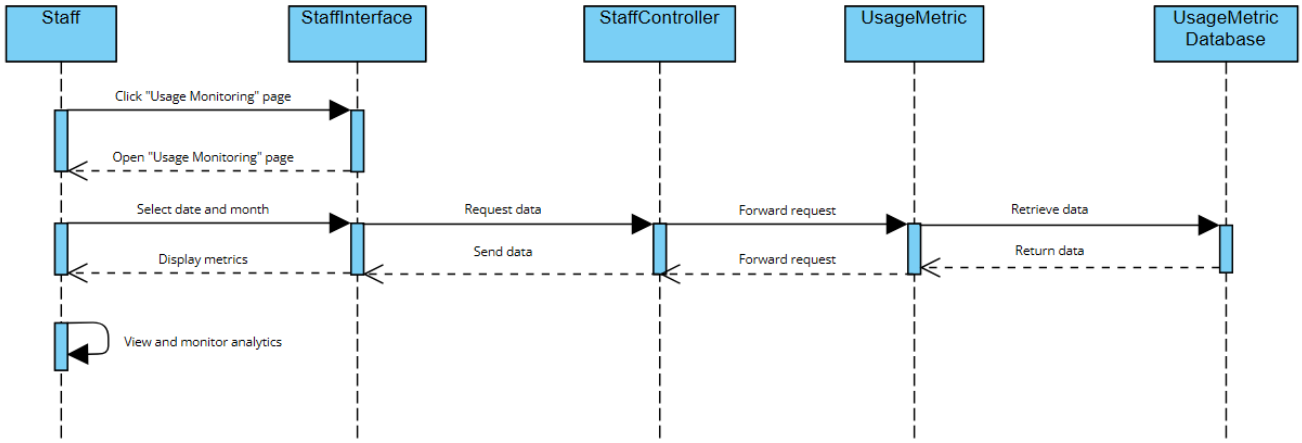


Figure 15: Usage monitoring sequence diagram

3 Data Design

3.1 Design Class Diagram

The design class diagram acts as a blueprint for the Electricity Billing System, illustrating the structure and interactions between the system's various components. It provides a clear visualization of how the different entities work together to achieve the desired functionality. By specifying the classes, their attributes, methods, and relationships, the diagram ensures an organized representation of the system's architecture. This approach supports modularity, scalability, and maintainability, facilitating efficient development and future updates. This section will detail each class, explaining its purpose, attributes, and methods, and its contribution to the system's overall functionality. This thorough explanation forms the basis for understanding the system design and implementation.

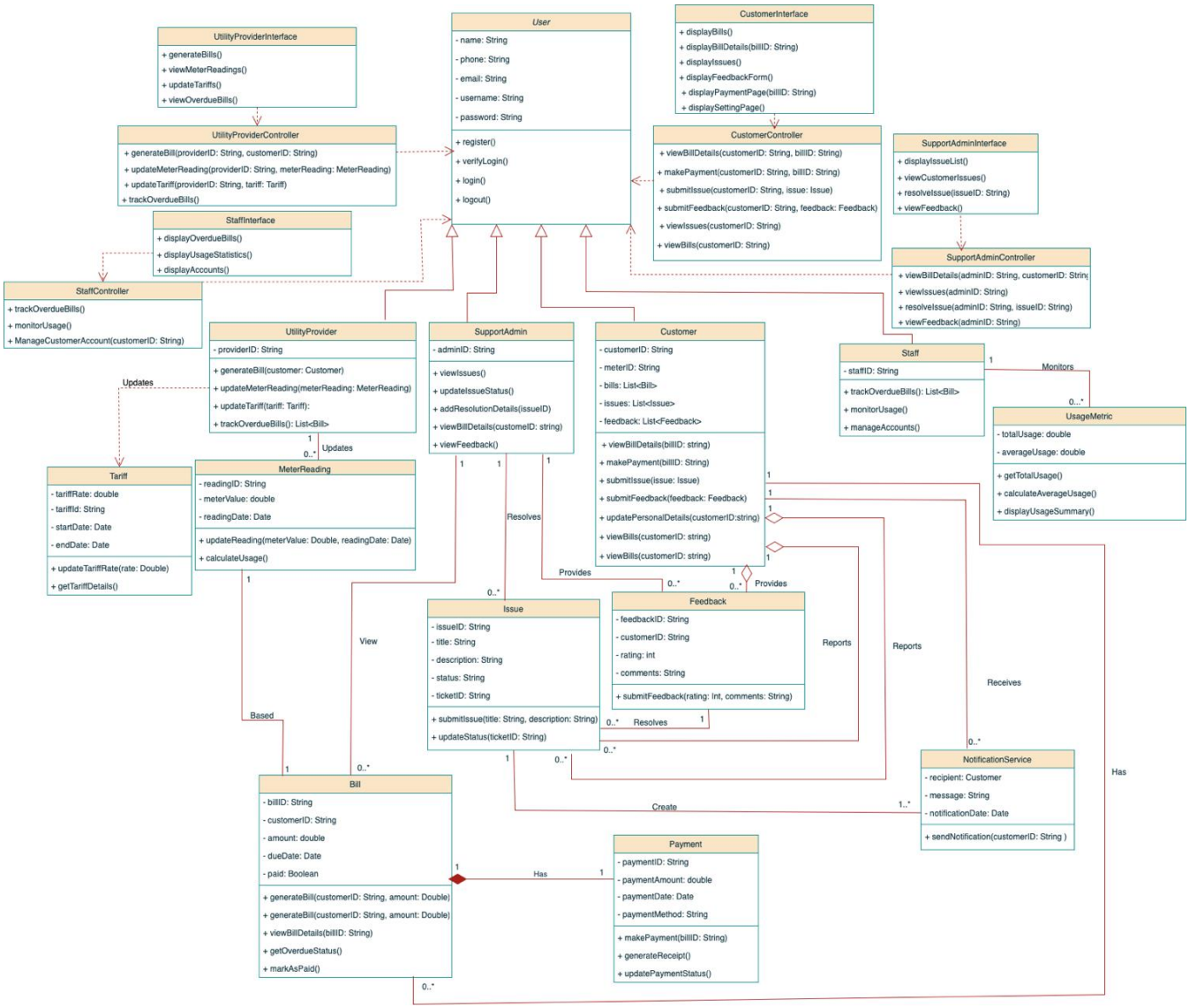


Figure 16: Class diagram

Table 2: Classes

Class / Entity	Description
User	The User class defines the general attributes and functionalities for all user types in the system. It contains key properties like name, phone, email, username, and password, which are important features for identifying and authenticating users. The class also defines core methods such as verifyLogin(), login(), and logout() that handle user session management. This class will be the base class for all other specific user types (classes), such as Staff, Supporting Admin, Utility Provider classes and Customer, which inherit from it and thus can implement similar functionality while allowing customization for each role.
Customer	The Customer class inherits from the User class, sharing common attributes and methods while adding specific features to customers. It includes attributes such as meterID, issues-a list of reported issues, and submitted feedback. It has methods like viewing bills through viewBills(), tracking the issues in viewIssues(), making payments through makePayment(), reporting problems via submitIssue(), and submitting feedback via submitFeedback(). This is a specialized class to gives customers a good experience.
CustomerInterface	The CustomerInterface class provides a user-friendly interface for customers to interact with the system. Through this interface, customers can perform their tasks with the help of methods such as displayBills(), displayBillDetails(), displayIssues(), displayFeedbackForm(), displayPaymentPage(), and displaySettingPage().
CustomerController	The CustomerController class serves as a bridge for the CustomerInterface with the system. It processes actions from the interface and ensures these actions are correctly executed in the system. Methods like viewBillDetails(), makePayment(), submitIssues(), submitFeedback(), viewIssues(), and viewBills() ensures efficient communication between customers and the system.
SupportAdmin	The SupportAdmin class is specialized for support personnel who oversee customer issues and their resolutions. Inheriting from User, it introduces unique functionality tailored to issue management. Methods such as viewIssues(), updateIssueStatus(), and addResolutionDetails() help the support admin track and resolve customer-reported problems. The support admin also has the responsibility to view customer bills and manage system-wide issues, making this class essential for ensuring that customer concerns are addressed and resolved promptly.
SupportAdminInterface	The SupportAdminInterface class offers an interface for support admins to perform their tasks. It includes methods such as displayIssueList(), viewCustomerIssues(), resolveIssue(), and viewFeedback() to support them in efficiently address customer concerns.
SupportAdminController	The SupportAdminController class serves as the connection between the interface and the system. It processes actions from support admins

	and communicates with the database accordingly. Methods like viewBillDetails(), viewIssues(), resolveIssues(), and viewFeedback() offers support in executing the tasks.
UtilityProvider	The UtilityProvider class represents the service provider in the system, responsible for managing the utility services provided to customers. It includes providerID, which identifies the provider, and methods for managing essential data like meter readings and tariffs. The class facilitates generateBill() and updateMeterReading() to calculate charges based on usage and to set tariff rates for services provided. By managing these core operations, the UtilityProvider class ensures that customers receive accurate bills and that the service's pricing structure is updated as needed.
UtilityProviderInterface	The UtilityProviderInterface class provides an interface for utility providers to manage the system. It includes methods like generateBills(), viewMeterReadings(), updateTariffs(), and viewOverdueBills() to help them oversee the system's operations and maintain control over the utility system effectively.
UtilityProviderController	The UtilityProviderController class acts as the bridge between the UtilityProviderInterface and the system. It processes actions by providing methods such as generateBill(), updateMeterReading(), updateTariff(), and trackOverdueBills().
Staff	The Staff class inherits from User and defines the roles and responsibilities of employee managing the system. With attributes like staffID, this class helps define who can access various system functionalities. Staff member can monitor customer UsageMetrics and track overdue bills. The ability to monitor usage and track overdue bills ensures that staff can efficiently oversee customer accounts, respond to the payment issues, generate trends reports and maintain operational efficiency within the system.
StaffInterface	The StaffInterface class serves as the graphical interface where the staff can interact with the system. It provides user-friendly screens for tasks such as Manage Customer Accounts, Tracking Overdue Bills, and Usage Monitoring. The class includes methods such as displayOverdueBills(), displayUsageStatistics(), and displayAccounts() to ensure those tasks can be done efficiently.
StaffController	The StaffController class acts as the intermediary between StaffInterface and the system's functionality. It translates the input from the interface and communicates with the system's database to execute corresponding actions. It includes methods such as trackOverdueBills(), monitorUsage(), and manageCustomerAccounts() to enable staff to effectively perform their roles.
Bill	The Bill class represents a utility bill issued to a customer. It includes attributes such as billID, customerID, amount, dueDate, and paidStatus. Methods like generateBill(), viewBillDetails(), and getOverdueStatus() are used to generate bills, view bill details, and check if the bill is overdue, allowing customers and staff to track payments and statuses and markAsPaid() updates the payment status.

Payment	The Payment class records payments made by customers. It holds attributes such as paymentID, paymentAmount, paymentDate, and paymentMethod. Customers can use methods like makePayment() and updatePaymentStatus() to process payments and track the customer payment status by the staff.
Receipt	The Receipt class provides customers with proof of payment. It holds information like receiptID, paymentID, and amountPaid, and offers methods like generateReceipt() and getReceiptDetails() to generate and retrieve detailed receipt information. The Receipt class is critical for confirming the completion of payments, ensuring that customers have an official record of their transactions, which they can use for future reference or dispute resolution.
MeterReading	The MeterReading class holds data about the readings taken from customer meters, with attributes such as readingID, meterValue, and readingDate. It provides the updateReading() method to update meter readings and calculateUsage() to calculate the usage based on the recorded values.
Feedback	The Feedback class enables customers to share their experiences with the service by submitting ratings and comments. It includes attributes like feedbackID, customerID, rating, and comments, and the submitFeedback() method allows customers to provide feedback on their service experience. This class helping the support admin identify areas for improvement and allowing them to assess customer satisfaction.
Tariff	The Tariff class helps calculate electricity pricing with attributes such as tariffID, tariffName, and rating. It categorizes customers under different plans based on their usage. The applyTariff() method calculates charges for consumption, while updateTariff() allows for modifying rates and conditions. This class ensures transparent and adaptable billing for all customers.
UsageMetric	The UsageMetric class allows for managing and analyzing key metrics of the system efficiently with two key attributes which are totalUsage and averageUsage. It provides functionalities to retrieve, calculate and display the usage with methods such as getTotalUsage(), calculateAverageUsage() and displayUsageSummary().
NotificationService	The Notification class control the messages send to customer. Attributes include notificationID to uniquely identify each message, recipient, which specifies the intended customer, message containing the notification content, and status, which tracks whether the notification has been sent or not. Methods such as sendNotification() and markAsSend() to ensure that customers remain up-to-date with all critical information.
Issue	The issue class tracks problems reported by customers regarding utility services. It contains attributes like issueID, description, status, and resolutionDate. Methods such as submitIssue() and updateStatus() allow customer to report issues and track their resolution.

3.2 Data Dictionary

The Data Dictionary outlines the structure of the database that supports the Electricity Billing System. Each table within the dictionary corresponds to a key component of the system, detailing the relationships, attributes, and constraints necessary for seamless data management. It provides a comprehensive overview of fields, data types, and constrains, ensuring data integrity, scalability, and optimal system performance. The following tables are described in detail:

- ***Customer Table***

Table 3: Customer database

Field Name	Data Type	Length	PK/FK	Required?	Null/Not Null	Description
customer_id	Varchar	5	PK	Yes	Not Null	Unique identifier for each customer.
customer_name	Varchar	30		Yes	Not Null	Full name of the customer.
customer_address	Varchar	100		Yes	Not Null	Residential address of the customer.
customer_phone	Char	13		Yes	Not Null	Phone number of the customer.
customer_email	Varchar	120		Yes	Not Null	Email address of the customer.
meter_id	Varchar	5	FK	Yes	Not Null	Reference to the meter connected to the customer for billing purposes.

- ***Utility Provider***

Table 4: Utility provider database

Field Name	Data Type	Length	PK/FK	Required?	Null/Not Null	Description
utility_id	Varchar	5	PK	Yes	Not Null	Unique identifier for each utility provider.

username	Varchar	10		Yes	Not Null	Unique username for each utility provider.
password	Char	8		Yes	Not Null	Encrypted utility provider password to secure access to the system.
up_name	Varchar	30		Yes	Not Null	Full name of the utility provider.
up_phone	Char	12		Yes	Not Null	Utility provider phone number.
up_email	Varchar	120		Yes	Not Null	Email address of the utility provider.

- ***Support Admin***

Table 5: Support admin database

Field Name	Data Type	Length	PK/FK	Required?	Null/Not Null	Description
admin_id	Varchar	5	PK	Yes	Not Null	Unique identifier for each admin.
username	Varchar	10		Yes	Not Null	Unique username for each admin.
password	Char	8		Yes	Not Null	Encrypted admin password to secure access to the system.
name	Varchar	30		Yes	Not Null	Full name of the admin.
phone	Char	13		Yes	Not Null	Admin phone number.
email	Varchar	120		Yes	Not Null	Email address of the admin.

- ***Staff***

Table 6: Staff database

Field Name	Data Type	Length	PK/FK	Required?	Null/Not Null	Description
staff_id	Varchar	5	PK	Yes	Not Null	Unique identifier for each staff.
username	Varchar	10		Yes	Not Null	Unique username for each staff.
password	Char	8		Yes	Not Null	Encrypted staff password to secure access to the system.

name	Varchar	30		Yes	Not Null	Full name of the staff.
phone	Char	13		Yes	Not Null	Staff phone number.
email	Varchar	120		Yes	Not Null	Email address of the staff.

- ***Bill Table***

Table 7: Bill database

Field Name	Data Type	Length	PK/FK	Required?	Null/Not Null	Description
bill_id	Varchar	5	PK	Yes	Not Null	Unique identifier for each bill.
customer_id	Varchar	5	FK	Yes	Not Null	Reference to the customer responsible for the bill.
amount	Double	8		Yes	Not Null	The bill amount that must be paid.
due_date	Date	3		Yes	Not Null	Due date for bill payment
paid	Boolean	1		Yes	Not Null	Status indicates if bill has been paid or not.
creation_date	Date			Yes	Not Null	Date when the bill was generated.
penalty_fee	Decimal	10,2		Yes	Not Null	A calculated fee for each late day.

- ***Issue Table***

Table 8: Issue database

Field Name	Data Type	Length	PK/FK	Required?	Null/Not Null	Description
issue_id	Varchar	5	PK	Yes	Not Null	Unique identifier for each reported issue.
customer_id	Varchar	5	FK	Yes	Not Null	Reference to the customer reporting the issue.
title	Varchar	15		Yes	Not Null	Short title of the issue.

description	Varchar	300		Yes	Not Null	The detailed description of the reported issue
status	Varchar	15		Yes	Not Null	Current status of the issue (e.g., "In Progress", "Resolved").
ticket_id	Varchar	5		Yes	Not Null	Unique ticket number for tracking the issue.

- ***Payment Table***

Table 9: Payment database

Field Name	Data Type	Length	PK/FK	Required?	Null/Not Null	Description
payment_id	Varchar	5	PK	Yes	Not Null	Unique identifier for each payment.
customer_id	Varchar	5	FK	Yes	Not Null	Reference to customer who made the payment.
payment_date	Date			Yes	Not Null	Date the payment was made.
payment_method	Varchar	10		Yes	Not Null	Payment method used (e.g., credit card, bank transfer).
amount	Decimal	10, 2		Yes	Not Null	The amount paid.

- ***Feedback Table***

Table 10: Feedback database

Field Name	Data Type	Length	PK/FK	Required?	Null/Not Null	Description
feedback_id	Varchar	5	PK	Yes	Not Null	Unique identifier for each

						feedback entry.
customer_id	Varchar	5	FK	Yes	Not Null	Reference to the customer providing the feedback.
rating	Int	1		Yes	Not Null	Customer rating out of 5 stars.
comment	Varchar	200		Yes	Not Null	Feedback comment or suggestions from the customer.
Feedback_date	Date			Yes	Not Null	Date the feedback was submitted.

- ***Meter Table***

Table 11: Meter database

Field Name	Data Type	Length	PK/FK	Required?	Null/Not Null	Description
meter_id	Varchar	5	PK	Yes	Not Null	Unique identifier for each meter.
customer_id	Varchar	5	FK	Yes	Not Null	Reference to the customer associated with the meter.
meter_reading	Int	8		Yes	Not Null	Latest recorded meter reading.
tariff_rate	Decimal	10,2		Yes	Not Null	Rate per unit of electricity.

3.3 Data Structures

The Data Structures section defines the organization and format of data used in the Electricity Billing System. It outlines arrays and their attributes, ensuring data integrity, scalability, and efficient retrieval. These structures support key functionalities like user management, billing, payments, and feedback, enabling seamless interaction between system actors (customers, utility providers, support admins, and staff). The following subsections detail each data structure, including attributes and data types, to ensure consistent and accurate data storage.

3.3.1 Customer Array

The Customer Array stores specific details about the customers, which are linked to other related arrays such as bill, meter, issue, payment, and feedback.

- **customer_id (varchar):** The unique identifier for each customer.
- **username (varchar):** Unique username for each customer.
- **password (Char):** Encrypted customer password to secure access to the system.
- **customer_name (varchar):** Full name of the customer.
- **customer_address (varchar):** Residential address of the customer.
- **customer_phone (char):** Phone number of the customer.
- **customer_email (varchar):** Email address of the customer.
- **ceter_id (integer):** Reference to the meter connected to the customer for billing purposes.

Table 12: Customer data structure

Attribute	Data Type
customer_id	Varchar
Username	Varchar
password	Char
customer_name	Varchar
customer_address	Varchar
customer_phone	Char
customer_email	Varchar
meter_id	Integer

3.3.2 Utility Provider Array

The Utility Provider Array serves as a centralized repository that stores comprehensive information about each utility provider. It enables management and efficient utilization of the services offered by these providers.

- **utility_id (varchar):** Unique identifier for each utility provider.
- **username (varchar):** Unique username for each utility provider.
- **password (char):** Encrypted utility provider password to secure access to the system.
- **up_name (varchar):** Full name of the utility provider.
- **up_phone (char):** Phone number of the utility provider.
- **up_email (varchar):** Email address of the utility provider.

Table 13: Utility provider data structure

Attribute	Data Type
utility_id	Varchar
username	Varchar
password	Char
up_name	Varchar
up_address	Varchar
up_phone	Char

up_email	Varchar

3.3.3 Support Admin Array

The Support Admin Array stores comprehensive information for each admin in the system, to provide their support in the most efficient ways.

- **admin_id (varchar):** Unique identifier for each admin.
- **username (varchar):** Unique username for each admin.
- **password (char):** Encrypted admin password to secure access to the system.
- **admin_name (varchar):** Full name of the admin.
- **admin_phone (char):** Phone number of the admin.
- **admin_email (varchar):** Email address of the admin.

Table 14: Support admin data structure

Attribute	Data Type
admin_id	Varchar
username	Varchar
password	Char
admin_name	Varchar
admin_address	Varchar
admin_phone	Char
admin_email	Varchar

3.3.4 Staff Array

The Staff Array stores information for each staff in the system, enabling their login to the system and better tracking.

- **staff_id (varchar):** Unique identifier for each admin.
- **username (varchar):** Unique username for each admin.
- **password (char):** Encrypted admin password to secure access to the system.
- **staff_name (varchar):** Full name of the admin.
- **staff_phone (char):** Phone number of the admin.
- **staff_email (varchar):** Email address of the admin.

Table 15: Staff data structure

Attribute	Data Type
staff_id	Varchar
username	Varchar
password	Char
staff_name	Varchar
staff_address	Varchar
staff_phone	Char

staff_email	Varchar

3.3.5 Bill Array

The Bill Array tracks billing information, including amounts, due dates, and payment statuses for customers.

- **bill_id(varchar):** Unique identifier for each bill.
- **customer_id (varchar):** Reference to the customer responsible for the bill.
- **tariff_ id (varchar):** Reference to the applicable tariff in the Tariff Array.
- **amount (double):** The bill amount that must be paid.
- **due_date (date):** Due date for bill payment.
- **paid (boolean):** Status indicating whether the bill has been paid.
- **created_date (date):** Date when the bill was generated.
- **penalty_fee (double):** A calculated fee for each late day.

Table 16: Bill data structure

Attribute	Data Type
bill_id	Varchar
customer_id	Varchar
tariff_id	Varchar
amount	Double
due_date	Date
paid	Boolean
created_date	Date
penalty_fee	Double

3.3.6 Issue Array

The Issue Array logs and monitors customer-reported issues for effective resolution and tracking.

- **issue_id (varchar):** Unique identifier for each reported issue.
- **customer_id (varchar):** Reference to the customer reporting the issue.
- **title (varchar):** Short title of the issue.
- **description (varchar):** Detailed description of the reported issue.
- **status (varchar):** Current status of the issue (e.g., "In Progress," "Resolved").
- **ticket_id (varchar):** Unique ticket number for tracking the issue.

Table 17: Issue data structure

Attribute	Data Type
issue_id	Varchar
customer_id	Varchar
title	Varchar
description	Varchar
status	Varchar

ticket_id	Varchar

3.3.7 Payment Array

The Payment Array records payment transactions made by customers, including methods and amounts.

- **payment_id (varchar):** Unique identifier for each payment.
- **customer_id (varchar):** Reference to the customer who made the payment.
- **payment_date (date):** Date the payment was made.
- **payment_method (varchar):** Payment method used (e.g., credit card, bank transfer).
- **amount (decimal):** The amount paid.

Table 18: Payment data structure

Attribute	Data Type
payment_id	Varchar
customer_id	Varchar
payment_date	Date
payment_method	Varchar
amount	Decimal

3.3.8 Feedback Array

The Feedback Array captures customer feedback, including ratings and comments, to assess service quality.

- **feedback_id (varchar):** Unique identifier for each feedback entry.
- **customer_id (varchar):** Reference to the customer providing the feedback.
- **rating (integer):** Customer rating out of 5 stars.
- **comment (varchar):** Feedback comment or suggestions from the customer.
- **feedback_date (date):** Date the feedback was submitted.

Table 19: Feedback data structure

Attribute	Data Type
feedback_id	Varchar
customer_id	Varchar
rating	Integer
comment	Varchar
feedback_date	Date

3.3.9 Meter Array

The Meter Array stores information about meters, including readings and customer associations, for accurate billing.

- **meter_id (varchar):** Unique identifier for each meter.
- **customer_id (varchar):** Reference to the customer associated with the meter.
- **meter_reading (integer):** Latest recorded meter reading.

Table 20: Meter data structure

Attribute	Data Type
meter_id	Varchar
customer_id	Varchar
meter_reading	Integer

3.3.10 Tariff Array

The Tariff Array defines electricity tariff rates and categories for billing calculations.

- **Tariff_id (varchar):** Unique identifier for each tariff.
- **tariff_name (varchar):** Name of the tariff (e.g., "Residential," "Commercial").
- **rate (decimal):** Rate per unit of electricity.

Table 21: Tariff data structure

Attribute	Data Type
Tariff_id	Varchar
Tariff_name	Varchar
rate	Decimal

4 Architecture Design

4.1 Software Architecture

The architecture of the Electricity Billing System is developed to ensure modularity, scalability, and security, enabling efficient management of electricity services. It comprises four key subsystems: Customer, Utility Provider, Support Admin, and Staff. Each subsystem is dedicated to specific functionalities and communicates with the others via clearly defined APIs and interfaces. The system adopts a client-server model, where the front-end interface serves users, and the back-end server manages business logic, data processing, and storage. This architecture facilitates smooth interaction among all stakeholders, including customers, utility providers, support teams, and administrative staff.

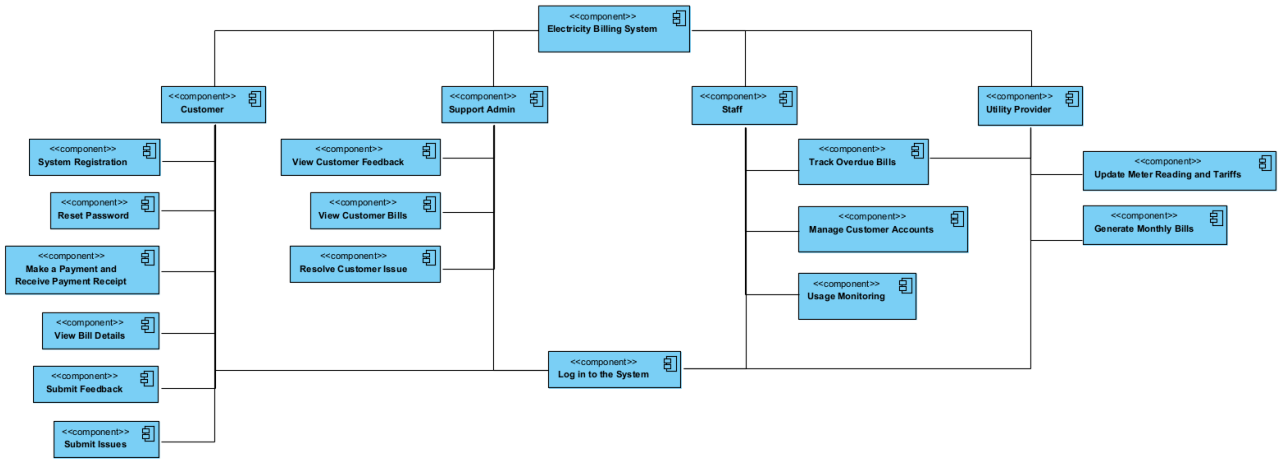


Figure 17: Software architecture

4.1.1 Subsystem 1: Customer

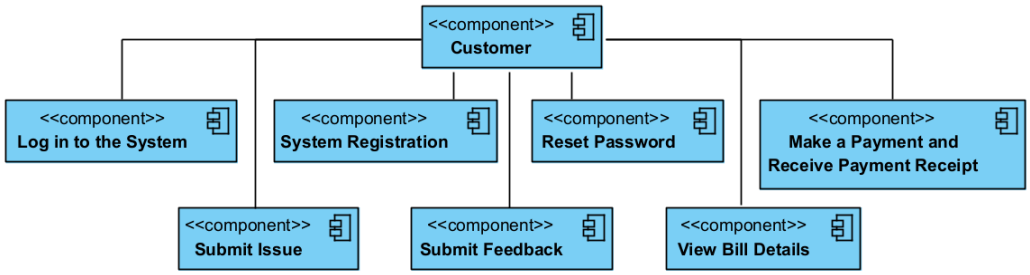


Figure 18: Customer software architecture

The Customer subsystem acts as the main interface through which end-users engage with the system. It is crafted to deliver a seamless and user-friendly experience while maintaining secure and efficient access to electricity services. The subsystem is organized into the following modules:

- **System Registration:** Enables new users to create accounts by providing necessary information.
- **Log in to the System:** Allows registered users to securely access their accounts using their credentials.

- **Reset Password:** Provides a mechanism for users to reset their forgotten passwords.
- **Make a Payment and Receive Payment Receipt:** Facilitates online bill payments and generates digital receipts for successful transactions.
- **Submit Feedback:** Enables users to provide feedback and suggestions to the utility provider.
- **Submit Issues:** Allows users to report issues or problems related to their electricity service.
- **View Bill Details:** Provides users with access to detailed information about their electricity bills, including consumption history, due dates, and payment history.

4.1.2 Subsystem 2: Utility Provider

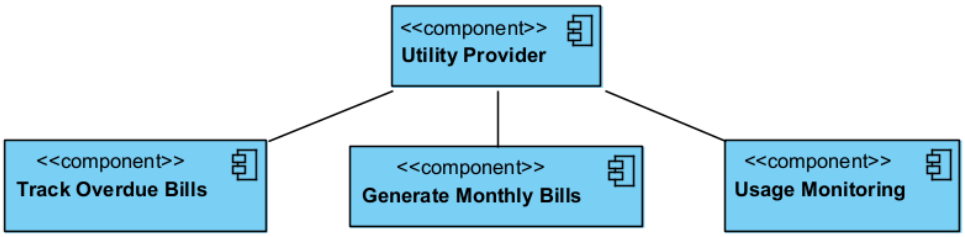


Figure 19: Utility provider software architecture

The Utility Provider subsystem is used to optimize billing processes and monitor electricity usage. It is designed to manage billing functions and maintaining efficient communication with customers and the staff. It has the following three main components:

1. **Track Overdue Bills**
   This component enables the utility provider to identify, verify and handle overdue bills within the system. It provides them with a list of overdue users after the staff has updated the system. Utility provider can then take necessary actions such as notifying the customer, setting a penalty or even cutting their electricity, depending on the severity.
2. **Generate Monthly Bills**
   This component helps the utility provider automate the monthly billing process, ensuring timely and accurate bill generation for all customers. This helps the customers to track their usage accurately, it also stores the digital bill info in the database for future reference.
3. **Meter Reading and Tariff**
   This component lets the utility provider check data of the meter readings and tariffs and also change them when needed. This helps the bill generation process to be more accurate.

The Utility Provider subsystem ensures efficient management of financial billing and monitoring. Each components work together to enhance operational efficiency, maintain financial stability, and improve customer satisfaction.

4.1.3 Subsystem 3: Support Admin

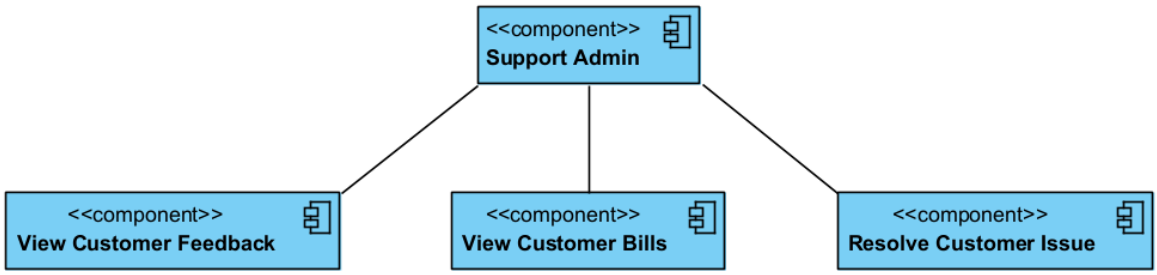


Figure 20: Support admin software architecture

This subsystem is responsible for managing interactions between the support administrator and the various system components related to customer support. The diagram shows three key components under the "Support Admin" subsystem:

1. **View Customer Feedback**:
   This component allows the support admin to access and review feedback provided by customers. It helps the administrator to analyze and address customer satisfaction and concerns.
2. **View Customer Bills**:
   This component enables the support admin to view and manage billing details for customers. It ensures that the admin has access to customer billing history for handling inquiries or disputes.
3. **Resolve Customer Issue**:
   This component provides the functionality for the support admin to address and resolve customer-related problems. It focuses on troubleshooting and ensuring customer satisfaction.

The central "Support Admin" component acts as the parent node, coordinating the activities of the three sub-components. Each of these components plays a crucial role in ensuring effective customer service and maintaining the overall functionality of the system.

4.1.4 Subsystem 4: Staff

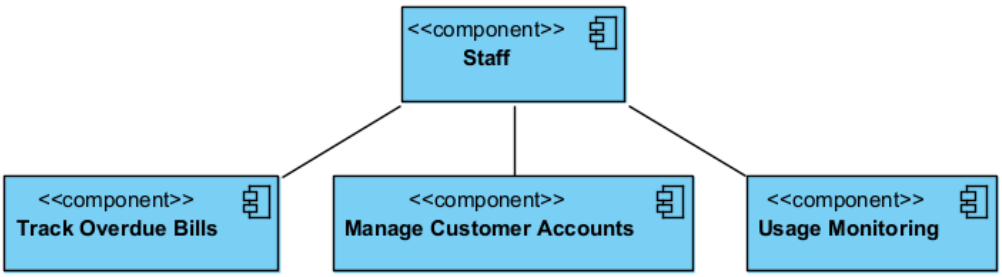


Figure 21: Staff software architecture

The staff subsystem allows the staff to efficiently handle and manage customer-related activities and ensure smooth system operations. This subsystem includes features to manage customer accounts, track overdue bills and monitor usage. Below is an overview of the subsystem's components:

1. **Manage Customer Accounts**:
   This component allows staff to manage customer accounts by enabling them to view account details and update its information. Staff can search for accounts for easier and efficient access to customer accounts.
2. **Track Overdue Bills**:
   This component enables staff to identify, verify and handle overdue bills within the system. It generates a list of accounts with overdue bills updated by the system based on the pay date. Staff can review the customer's account and take necessary actions such as setting a new due date for the overdue bill and notify the customer.
3. **Usage Monitoring**:
   This component allows staff to review and analyze customer usage data, such as total number of customers, total number of bills generated, and total number of bills paid. It helps staff to identify trends and irregularities based on the data received.

Overall, the staff subsystem enhances operational efficiencies and customer service by providing staff with capabilities to manage customer-related process. These three components work seamlessly to allow staff to perform their tasks with ease and precision, improving customer satisfaction and supporting the overall success of the program.

4.2 State Diagram

4.2.1 Customer

The customer state diagram models the customer workflow within an electricity billing system. The central element is the "Customer Main Page," which acts as a hub for various functionalities. From this page, customer can access different options, they can view their active bills and drill down to view detailed information about each bill. Additionally, they can access the "Setting" section to update their personal information or reset their password. The system also allows customers to provide feedback, make payments for their bills, or report any issues they may be experiencing. After performing any of these actions, users can return to the "Customer Main Page" to access other functionalities. This diagram illustrates the dynamic nature of the system, enabling users to navigate seamlessly between different functionalities within the context of the electricity billing system.

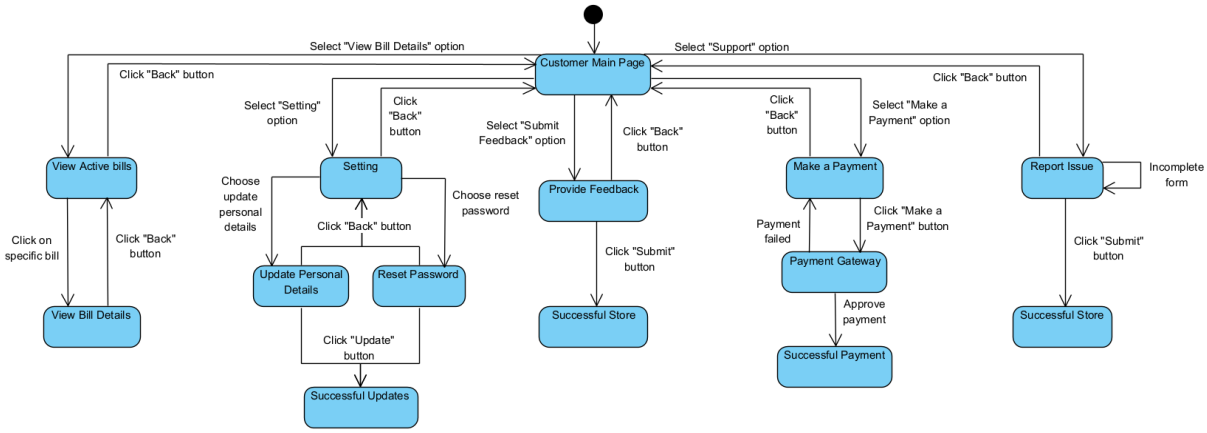


Figure 22: Customer state diagram

4.2.2 Utility Provider

The Utility Provider state diagram shows the states the system can be inside the Utility Provider subsystem. From the main page, the utility provider can access three pages which are “Track Overdue Bills”, “Update Meter Readings and Tariffs”, and “Generate Monthly Bills”. The actions of the utility provider will trigger the transitions between states with the help of buttons like “Set Penalty Amount”, “View Details”, and “Update Rates”. Each state includes a “Back” button that allows the staff to return to the previous state.

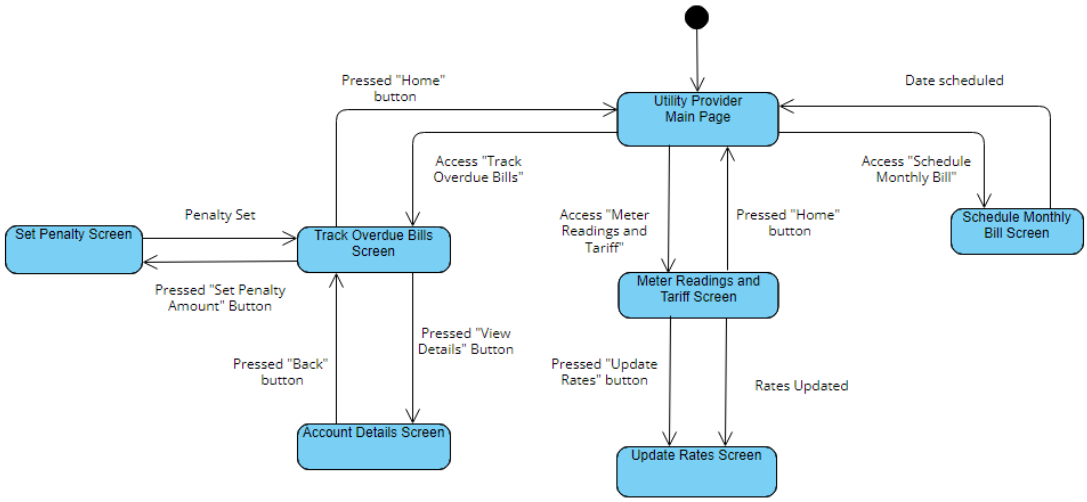


Figure 23: Utility provider state diagram

4.2.3 Support Admin

The support admin state diagram models the workflow of the Support Admin interface in an Electricity Billing System. It consists of multiple states representing different functional areas, such as "View Customer Bill," "View Customer Feedback," and "Resolve Customer Issue." Transitions between states are triggered by admin actions, such as clicking buttons like "Search," "View Details," "Update Status," and "Home." The "Home" state acts as a central hub, allowing the administrator to return to the main page from any other state. This hierarchical structure enables efficient navigation and task management within the system.

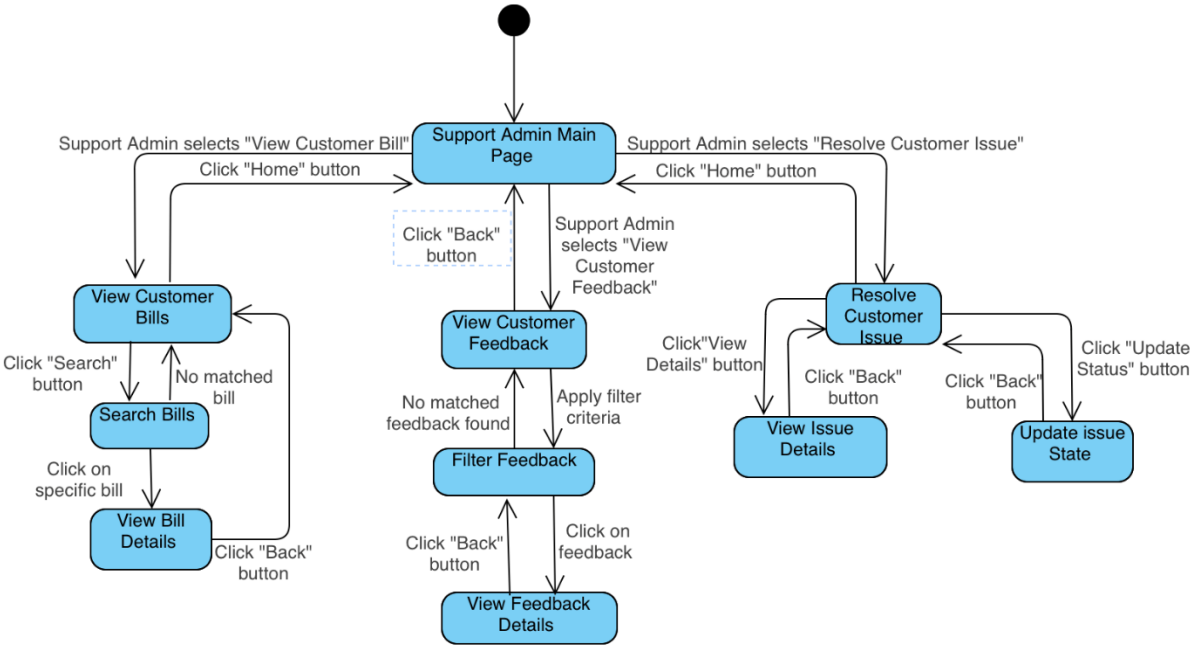


Figure 24: Support admin state diagram

4.2.4 Staff

The staff state diagram shows the various states the system can be in inside the staff subsystem starting from the staff main page where staff can navigate to three primary functionalities which are “Manage Customer Accounts”, “Track Overdue Bills”, and “Usage Monitoring”. In each state, the transitions are caused by the staff’s action such as clicking buttons like “View Details”, “Update Details”, “Set Final Due Date”, and filtering month and year. Each state includes a “Back” button that allows the staff to return to the previous state.

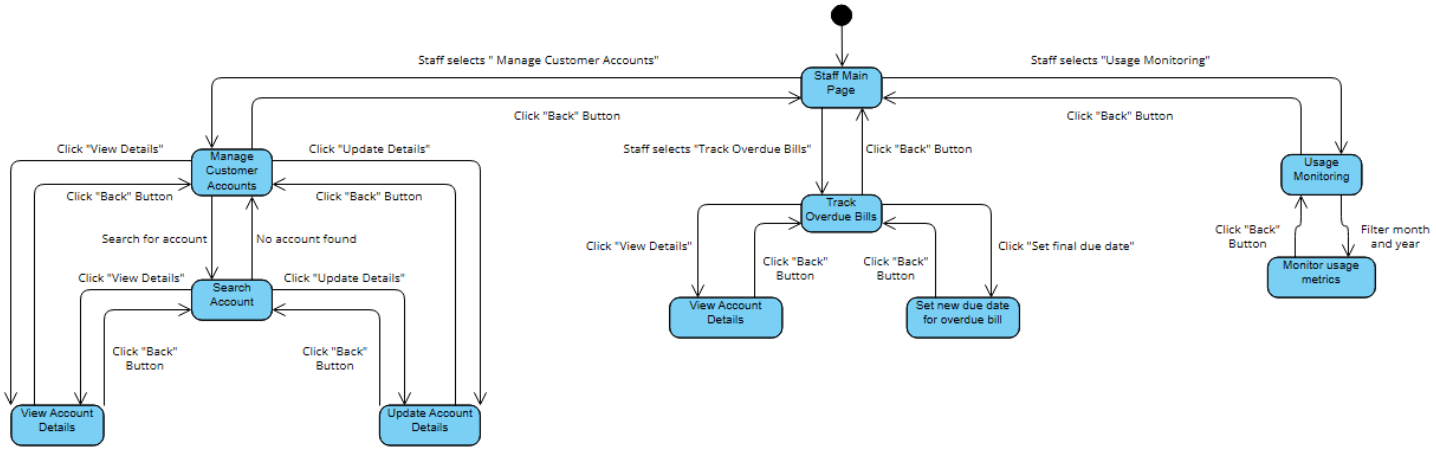


Figure 25: Staff state diagram

5 Interface Design

5.1 Main Screens

The Login Screen for the Electricity Billing System serves as the entry point for all users, including customers, support administrators, staff, and utility providers, to access the application. It ensures secure and authorized access while maintaining a user-friendly interface. After successful authentication, users are directed to their respective interfaces, tailored to their roles and permissions.

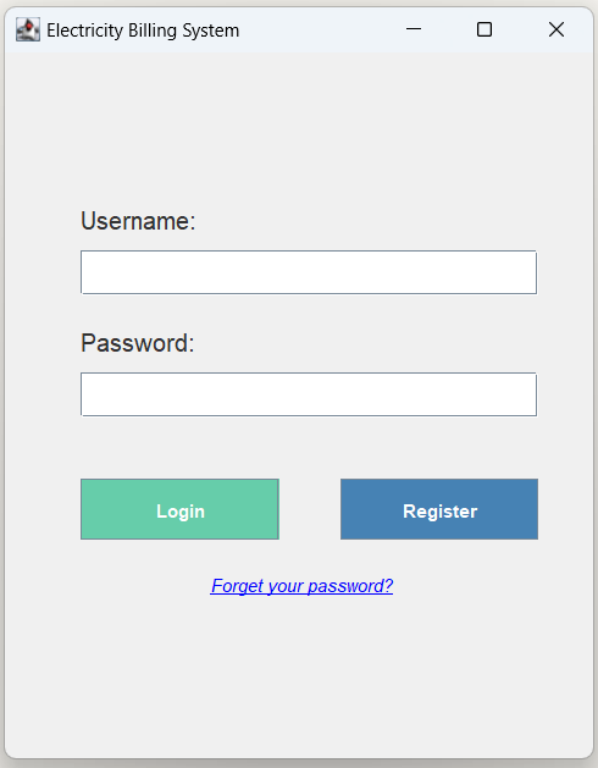


Figure 26: System login screen

5.2 Subsystem 1: Customer Screens

The main screen will be a one-stop dashboard where customers can manage all the tasks related to electricity. It will have easy access to the most used features like payment, bill display, profile update, feedback, and issue reporting. The design has focused on user convenience and operational efficiency.

Main screen

This is the first screen customers encounter upon logging into their accounts. It features multiple buttons, each providing access to a specific service, ensuring a user-friendly experience. The displayed name, such as "Yousef," dynamically updates to reflect the logged-in customer's identity, adding a personalized touch to the interface.

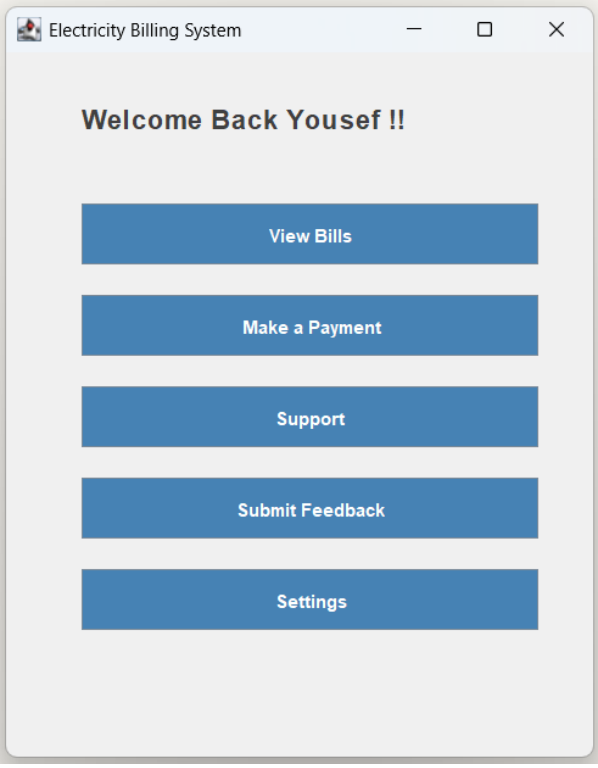


Figure 227: Customer main screen

- **Customer-Specific Welcome Message**
  The screen displays a personalized message, such as "Welcome Back Yousef!". This feature confirms successful login and enhances the user experience by providing a warm and tailored greeting.
- **View Bills**
  The "View Bills" button allows customers to view a comprehensive list of all their electricity bills. The list includes essential details such as the billing balance, due date, and total amount due.
- **Make a Payment**
  The "Make a Payment" button allows customers to pay their electricity bills easily. Upon clicking the button, it redirects the user to a payment interface where they can see the outstanding balance of their bills, select a mode of payment, and make the payment seamlessly.
- **Feedback**
  The "Submit Feedback" button helps customers to provide feedback regarding the experience of using the electricity service system. Upon clicking, users are taken to a feedback form where they rate and make comments or give suggestions about their experiences. This will help the Support Admin improve on the quality of service based on the input provided by the customers.
- **Support**
  The "Submit Issues" button allows the customer to report a problem or an issue regarding the electricity service, such as errors in billing or malfunctioning of the meter. It redirects the users to the form for submitting an issue, where they can explain the problem with attachments. This helps in timely resolution of the reported issue.

Registration screen

The **"Enter Meter ID" Screen** is a crucial step in linking the user's account to their specific electricity meter. Users are prompted to input their unique Meter ID and click "Next." The system validates the entered Meter ID to ensure accuracy:

- **If Valid:** The user proceeds to the next registration step, such as entering personal details.
- **If Invalid or Already Registered:** An error message is displayed, preventing further progress.

This step ensures accurate billing and proper association of accounts with the correct meter, forming the foundation for reliable service.

Once the user clicks "Next" with a valid Meter ID, they are redirected to the **Electricity Billing System Registration Form**. This form is designed to onboard new users by collecting mandatory details, including:

- Name
- Email
- Phone Number
- Address
- Username
- Password

After completing the required fields, users click the "Submit" button to finalize account creation. This registration process ensures security and proper access control, allowing only authorized users to manage electricity bill processing and account-related tasks efficiently.

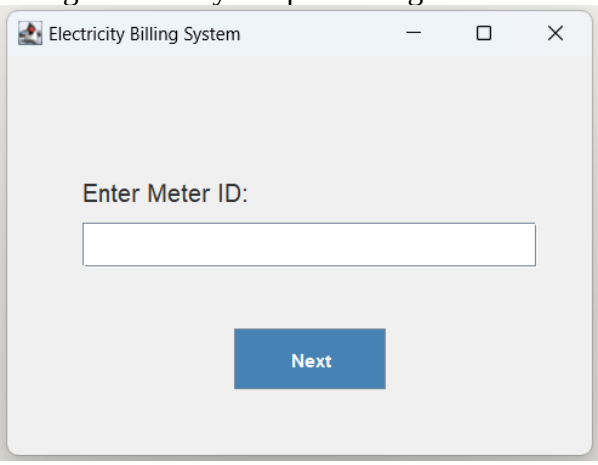


Figure 28: Registration screen (1)

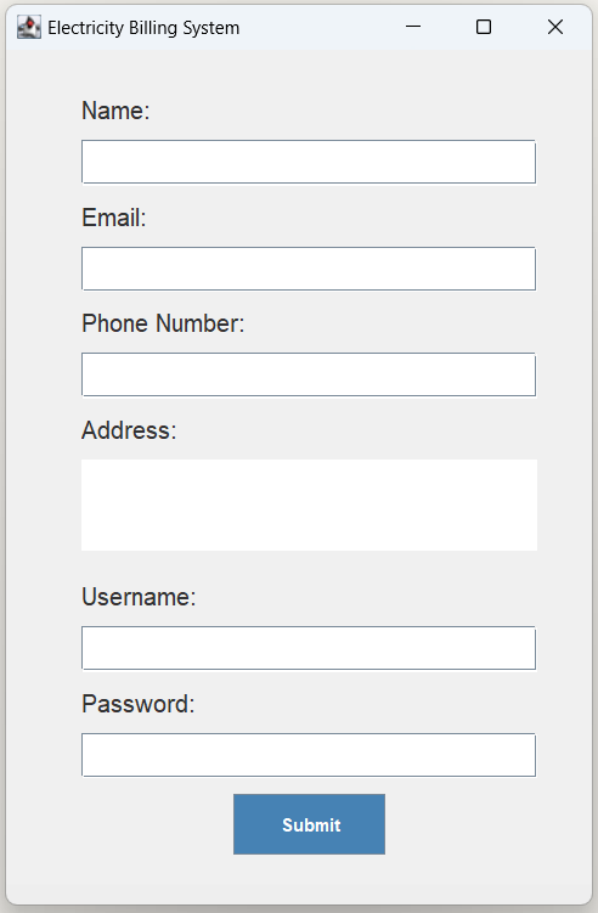


Figure 29: Registration screen (2)

Reset password screen

The **"Reset Your Password"** feature of the Electricity Billing System offers a straightforward and secure mechanism for users to regain access to their accounts in case of forgotten passwords.

- **User Input:** Users enter the email address linked to their registered account in a simple and intuitive interface.
- **Action:** Upon clicking the "Submit" button, the system triggers the password reset process.
- **System Response:** An email is sent to the provided address containing detailed instructions, typically including a reset link or a verification code, to facilitate password recovery.

This feature ensures user convenience by enabling seamless account recovery while maintaining robust security measures to protect sensitive account information.

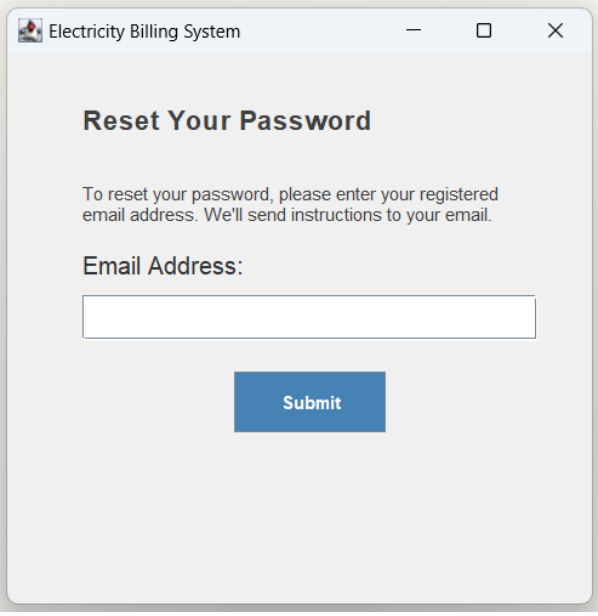


Figure 30: Reset password screen

Report issue screen

This interface allows customers to easily report problems or issues related to their electricity service.

Navigate Home Button: This button takes the customer back to the main home screen of the system.

Issue Title: A text field where the customer can enter a brief title or subject for their issue.

Issue Description: A text area where the customers can provide a detailed description of the problem they are facing. This could include information like the nature of the problem, when it started, and any relevant symptoms.

Attachments: This section allows the customer to upload any supporting documents or images related to their issue. Upload file button triggers the file upload dialog, allowing to select files from the computer with message indicates if no file has been selected.

Submit Button: Once the user has entered the necessary information, they can click this button to submit the issue report.

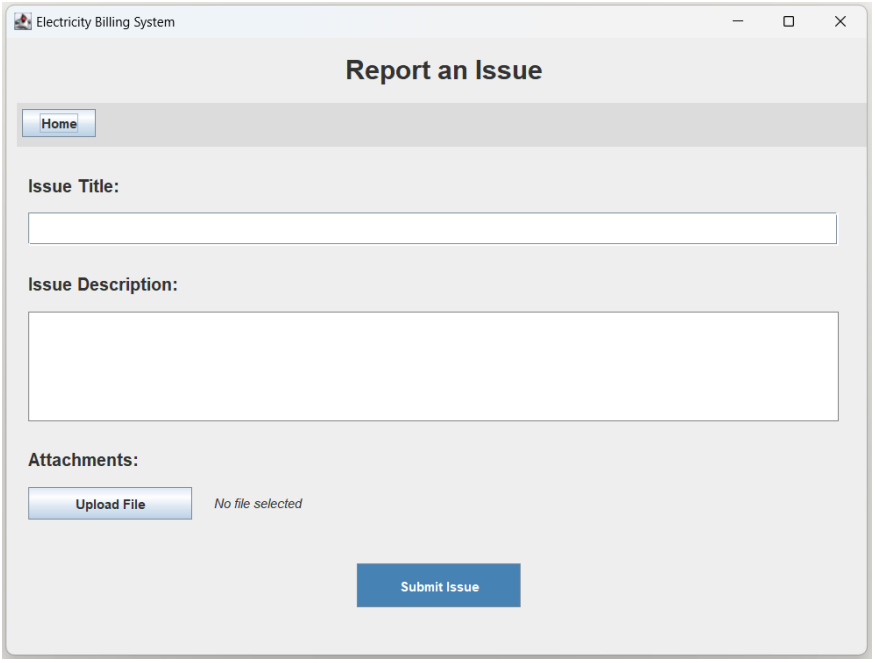


Figure 31: Report an issue screen

View bill details screen

This screen displays a list of the user's active electricity bills and provides detailed information about the selected bill.

View Bills: A list displaying all active bills associated with the user's account. Each bill is represented by a numbered entry (e.g., "Bill 1," "Bill 2").

Bill Details: A section that displays detailed information about the selected bill from the "View Bills" list. This includes:

- Outstanding Balance: The amount to be paid for the selected bill.
- Total Electricity Consumption: The total amount of electricity consumed during the billing period.
- Payment Due Date: The date by which the payment for the selected bill is due.

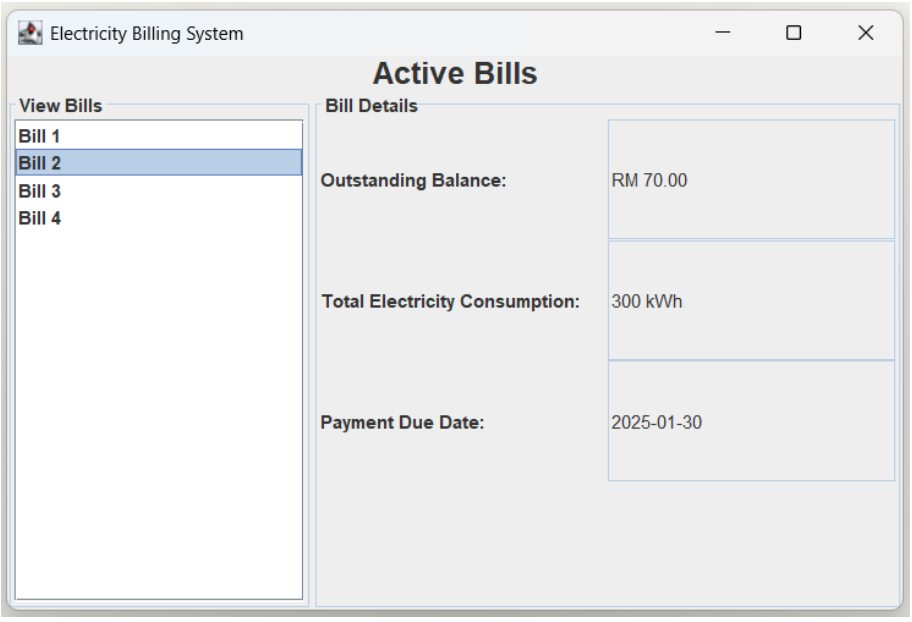


Figure 32: View bill details screen

Make a payment screen

This screen enables users to select bills for payment and proceed with the payment process.

- Select Bills: A list displaying all active bills associated with the customer's account. Each bill is presented with a checkbox to select and its corresponding amount. Customers can select multiple bills for payment.
- Outstanding Balance: Displays the total amount due for the selected bills.
- Mode of Payment: A dropdown menu allowing users to select their preferred payment method (e.g., Credit/Debit Card, Bank Transfer).
- Total Amount: Displays the total amount to be paid, which matches the "Outstanding Balance" of selected bills.

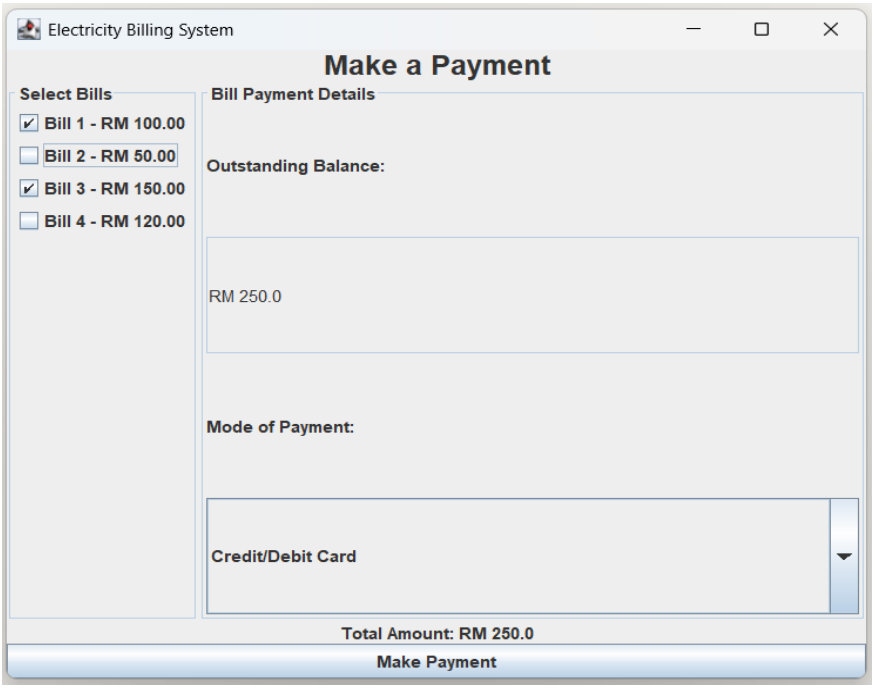


Figure 33: Make a payment screen

5.3 Subsystem 2: Utility Provider Screens

Main Screen

The initial screen of the Utility Provider interface provides an organized and user-friendly environment for the utility provider to efficiently manage finance-related tasks within the Electricity Billing System. Below are the key elements visible on this screen:

- **Welcome Message**:

The Utility Provider is greeted with a welcoming message upon successful login.

- **Meter Readings and Tariffs:**

This option allows Utility Provider to check the current rates for the meter readings and tariffs, and to update the data values. This helps the staff be able to calculate the bill to accurately charge the customers based on the rates.

- **Track Overdue Bills:**

This option helps the Utility Provider check which customer has outstanding fees. The Utility Provider can choose to give them one more week as the final deadline or

disconnect the electricity. This helps the Utility Provider to deal with customers with overdue fees efficiently.

- **Schedule Monthly Bill:**

This option lets the Utility Provider schedule a date on every month to generate the bill for each customer. This helps the Utility Provider to automate the billing process, with great accuracy and efficiency.

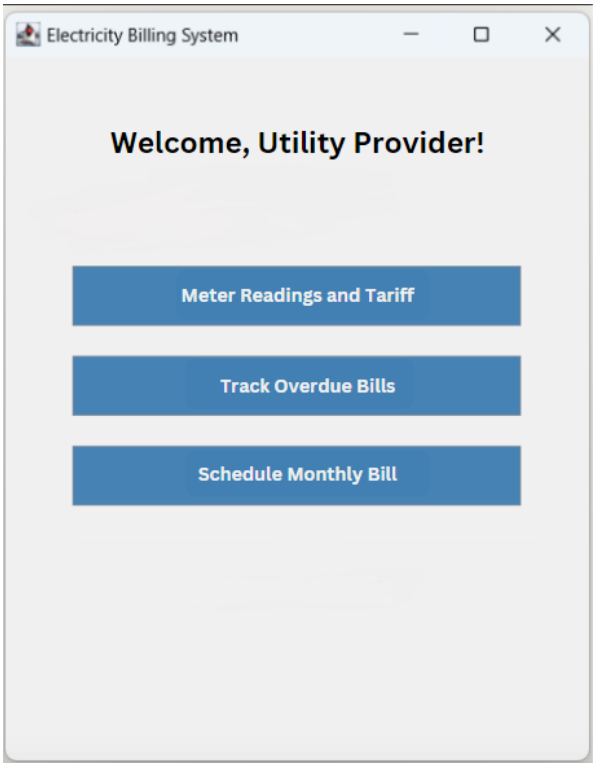


Figure 34: Utility provider main screen

Meter Readings and Tariff screen

The Meter Readings and Tariff page is accessible by clicking the “Meter Readings and Tariff” button, it lets the utility provider check the current tariff values to make sure the data is accurate.
The column displays data such as:
Tariff category: The category of tariff.
Unit: The unit of tariff.
Current Rate: The current rate of tariff which the customer should be charged.

Navigation Bar:
The navigation bar provides quick access to various sections within the system:

- **Home**: Redirects the Utility provider to the main screen.
- **Update Rates**: Brings the Utility provider to the Update Rates screen.

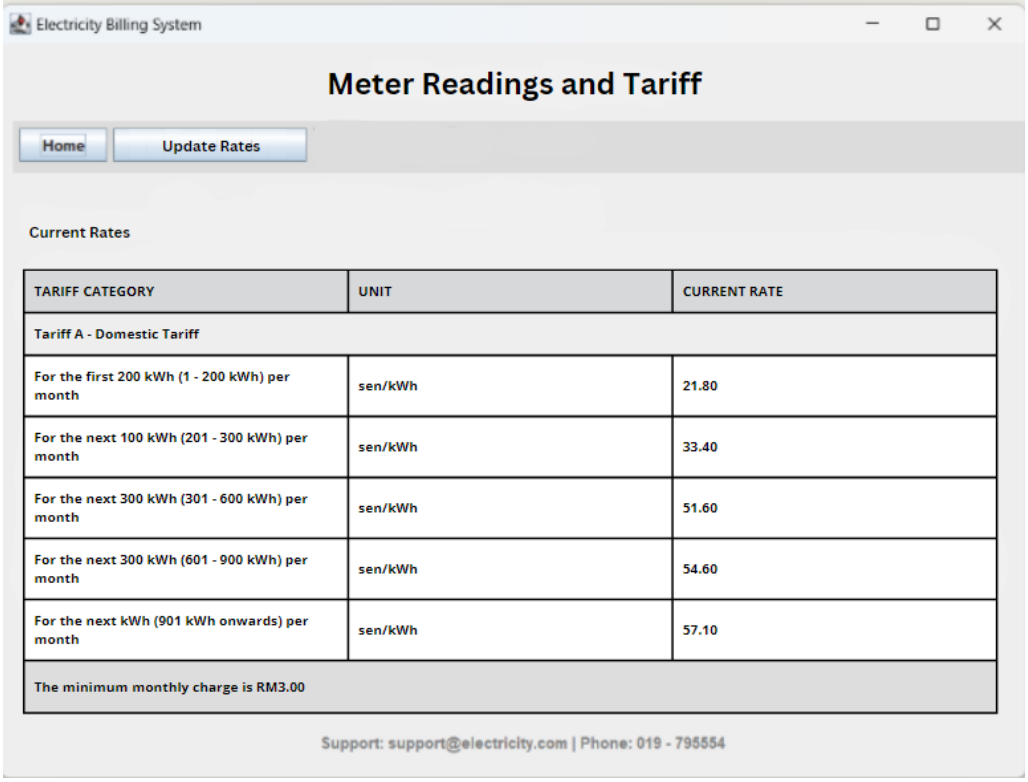


Figure 35: Meter readings and tariff screen

Update Rates screen

The "Update Rates" screen prompts the utility provider to key-in the new tariff rates, there is a drop-down menu to let them choose which tariff category to update.

Figure 36: Update rates screen

Track Overdue Bills Screen

The "Track Overdue Bills" screen is designed to help utility providers monitor, verify, and take necessary actions to manage overdue bills in the system. It ensures that overdue payments are identified and handled effectively.

Home Button:
The "Home" button allows the utility provider to go back to the main screen.
Overdue Bills List:
The table displays a list of overdue bills with columns such as:

- **Bill ID:** Unique identifier for each bill.

- **Customer Name:** The name of the account holder.
- **Amount:** The total amount due for the bill.
- **Date:** The date when the bill is generated.
- **Overdue by:** Indicates how long the bill has been overdue by.
- **Actions:** A "View Details" button to view a more detailed information of the bill details, and a "Set Penalty Amount" button for the utility provider to set a penalty for the customer and disconnect their electricity.

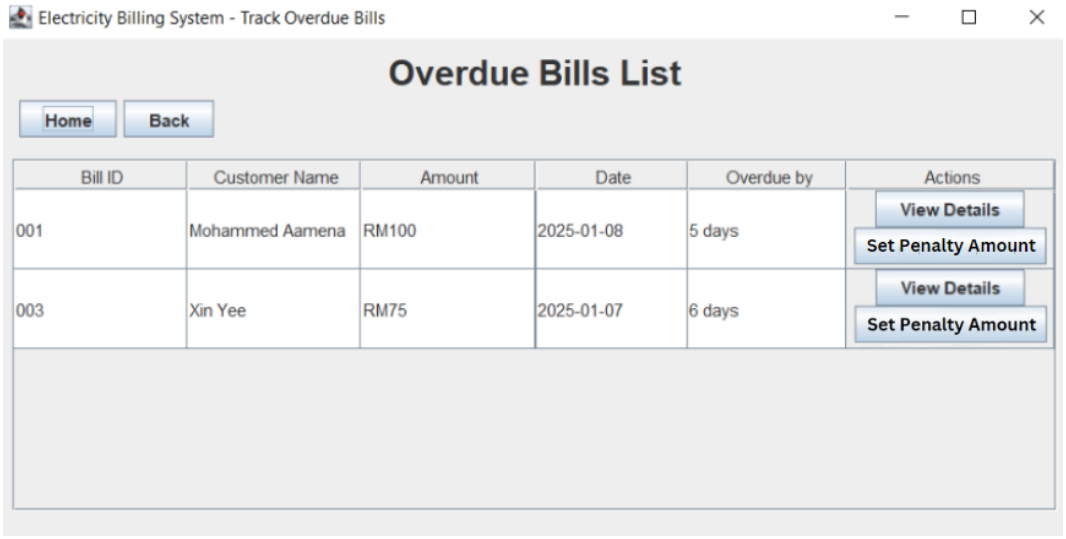


Figure 37: Track overdue bills (Utility Provider) screen

View Details Screen

The "View Details" screen provides a detailed description of the customer information, their billing information and payment histories. Utility providers can review and verify the necessary information.

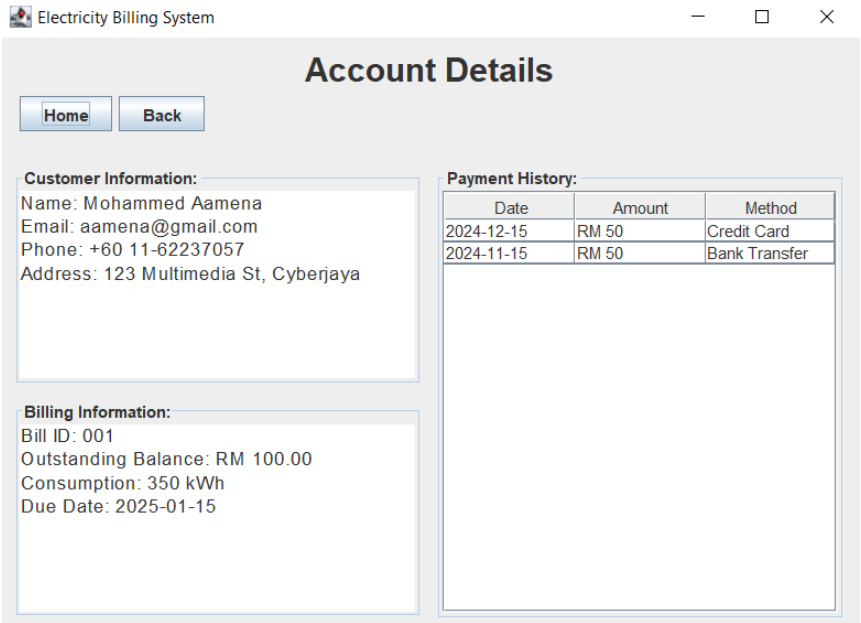


Figure 38: View details screen

Set Penalty Screen

The “Set Penalty" screen prompts the utility provider to key-in the penalty amount of the customer. This will disconnect the electricity for the customer until they pay the penalty.

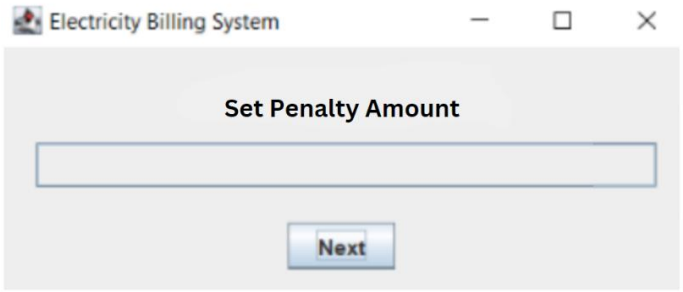


Figure 39: Set penalty screen

Schedule Monthly Bill screen

The “Schedule Monthly Bill” screen lets the utility provider to set a scheduled date to generate the monthly bill automatically, there is a drop-down menu with a calendar to assist them with scheduling.

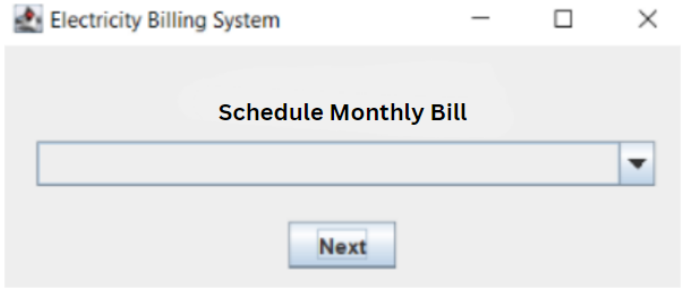


Figure 40: Schedule monthly bill screen

5.4 Subsystem 3: Support Admin Screens

Main Screen

The initial screen of the Support Admin interface provides an organized and user-friendly environment for Support Administrators to efficiently manage customer-related tasks within the Electricity Billing System. Below are the key elements visible on this screen:

Welcome Message ("Welcome to Support Center!"): This greeting establishes a friendly and approachable tone for the interface, signalling the start of the Support Admin's session.

View Customer Bills: By selecting this option, Support Admins gain immediate access to customer billing details. This functionality is essential for troubleshooting billing discrepancies, answering customer inquiries, or providing clarifications on bill charges.

View Customer Feedback: This option allows Support Admins to review customer feedback, including complaints, suggestions, and compliments. Access to this information is critical for understanding customer satisfaction, identifying recurring issues, and prioritizing areas for service enhancement.

Resolve Customer Issues: Clicking this button directs Support Admins to a section dedicated to issue resolution. This includes features for tracking open issues,

communicating with customers, managing resolution workflows, and accessing a knowledge base of common solutions.

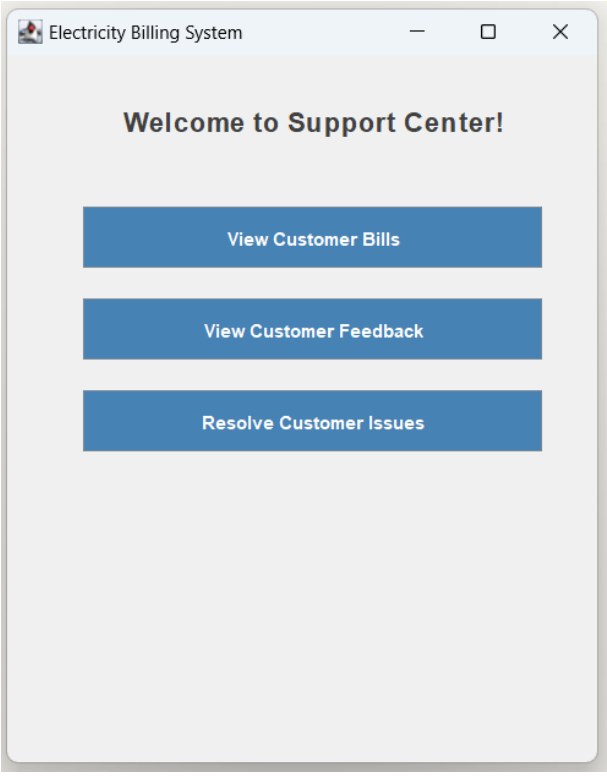


Figure 41: Support admin main screen

View Customer Bill Screen

The primary function of the Bill Management Screen is to allow Support Admins to search, view, and manage customer bills in a streamlined interface. This screen displays a list of customer bills and offers the ability to filter and search for specific bills based on various criteria.

Navigation Bar: The navigation bar provides access to various sections within the system:

- **Home**: Takes the Support Admin to the main dashboard or central hub of the system.
- **Customer Feedback**: Directs to a section where customer feedback is managed.
- **Customer Issues**: The current screen where the list of customer issues is displayed.

Search Functionality:

- **"Search Bills" Textbox:** Allows Support Admins to enter search criteria, such as customer name or bill ID, to quickly locate specific bills.
- **"Search" Button:** Initiates the search process, filtering the displayed bills based on the entered criteria.

Bill Information List:

- **Bill ID**: A unique identifier for each bill.
- **Customer Name**: The name of the customer associated with the bill.
- **Amount**: The total amount due for the bill.
- **Status**: Indicates the status of the bill (e.g., "Pending" if payment is not received, "Paid" if payment has been made).
- **Date**: The date when the bill was generated.

Clicking on an element on the list for a specific bill would navigate the Support Admin to the **"Bill Details"** screen, where they can view detailed information about the selected bill, including customer details, payment history, and billing details.

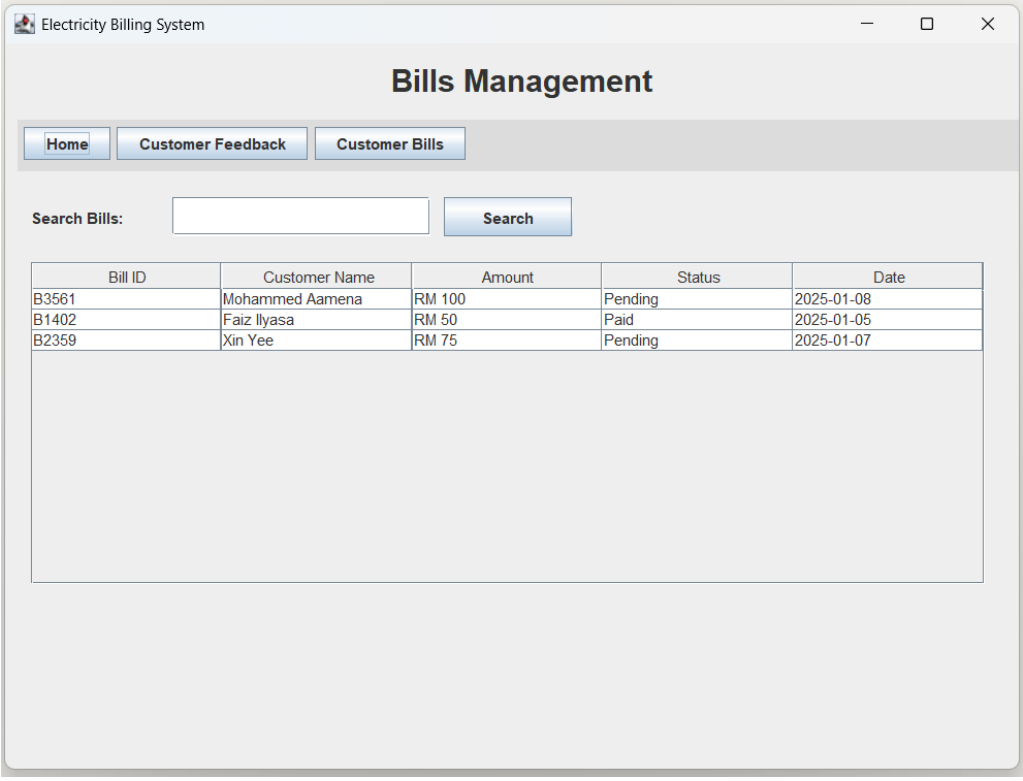


Figure 42: View customer bill screen

View Bill Details Screen

The "Bill Details" screen is accessible from the "View Customer Bills" section by the Support Admin. It provides a comprehensive view of the customer's billing and payment history, ensuring that the Support Admin has all the relevant information required for resolving issues or assisting with customer inquiries.

The "Bill Details" screen includes the following sections:

Customer Information

This section presents essential customer details, including:

- Customer Name: Full name of the customer.
- Email Address: Contact email for communication.
- Phone Number: Customer's phone number for support contact.
- Customer Address: Address of the customer for record-keeping and contact purposes.

These details help the Support Admin identify the customer and contact them if necessary.

Billing Information

This section provides critical billing data, such as:

- Bill ID: Unique identifier for the bill.
- Outstanding Balance: The total amount owed by the customer.
- Total Electricity Consumption: The electricity usage during the billing period, measured in kilowatt-hours (kWh).
- Due Date: The date in which the payment is due.

This section is essential for understanding the customer's financial obligations and tracking their billing cycle.

Payment History

Displays a chronological list of payments made toward this bill, which includes:

- Payment Date: Date when each payment was made.
- Amount Paid: Amount paid by the customer for each payment.
- Payment Method: Mode of payment used (e.g., credit card, bank transfer, e-wallet).

Payment history will provide the Support Admin with details on the status of previous payments to identify if any discrepancy or payment is missed.

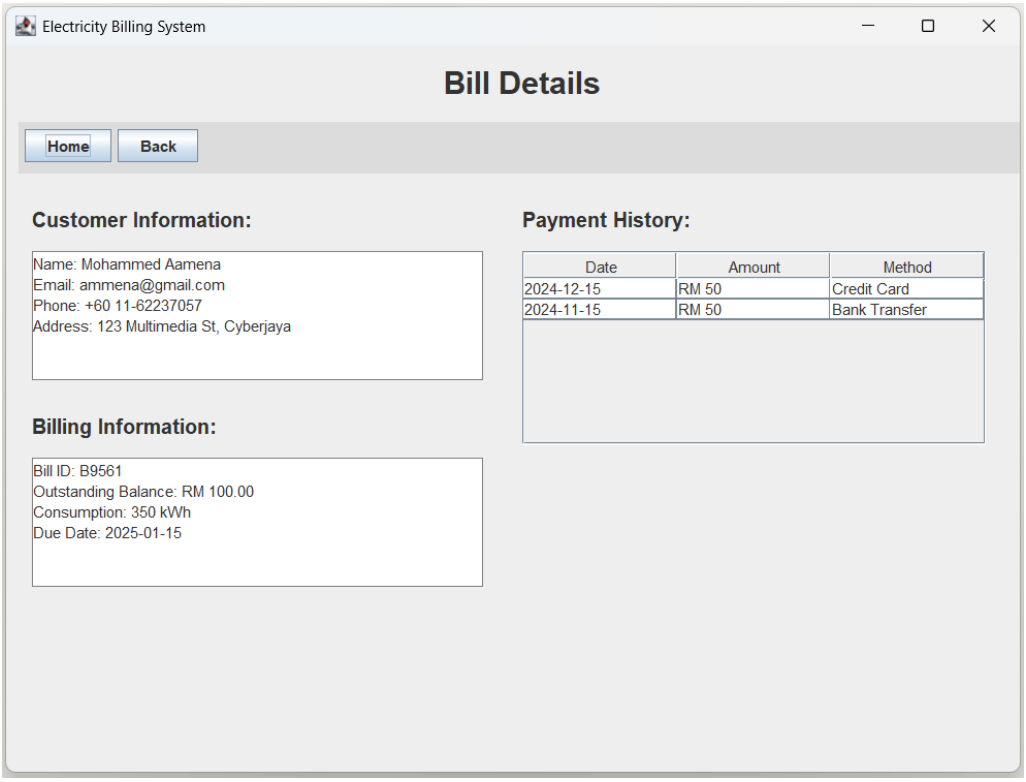


Figure 43: View bill details screen

View Customer Issues Screen

The major role of the Customer Issues Management Screen would be the centralization of all views for the Support Admin about issues reported by customers. Therefore, enable the Support Admin to trace the running issues together with Support Tickets on this screen in a move aimed at taking swift action towards their resolution.

Navigation Bar

The navigation bar provides quick access to various sections within the system:

- **Home**: Redirects the Support Admin to the main dashboard.
- **Customer Feedback**: Leads to a section where customer feedback, including complaints and suggestions, is managed and reviewed.
- **Customer Bills**: Takes the Support Admin to the screen for managing and reviewing customer billing information.

Issue List

Presents a list of customer issues, displaying the following key columns:

- **Ticket ID**: A unique identifier assigned to each support issue.
- **Issue Title**: A brief about what the issue reported by the customer.
- **Status**: Indicates the status of the issue, such as "In Progress," or "Resolved."

Provides options for Support Admins to interact with each issue:

- **View Details**: Opens a detailed view of the specific issue, providing in-depth information, the full issue description, and any relevant attachments.
- **Update Status**: Allows the Support Admin to modify the status of the issue (e.g., updating it from "In Progress" to "Resolved") and add comments.

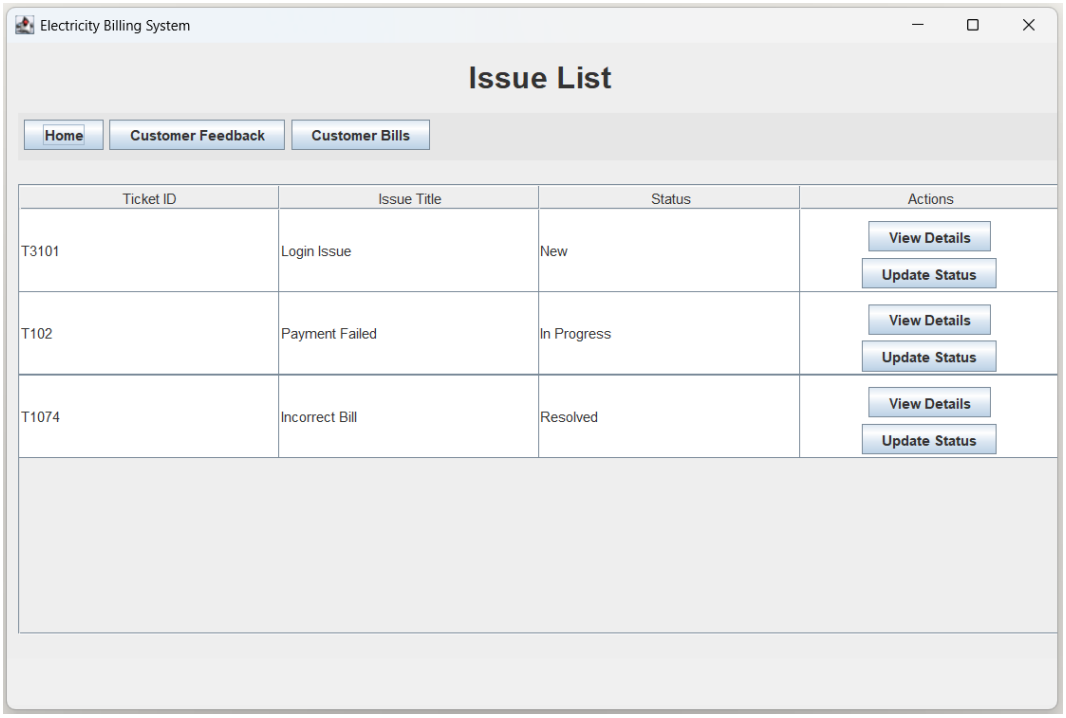


Figure 44: View customer issue screen

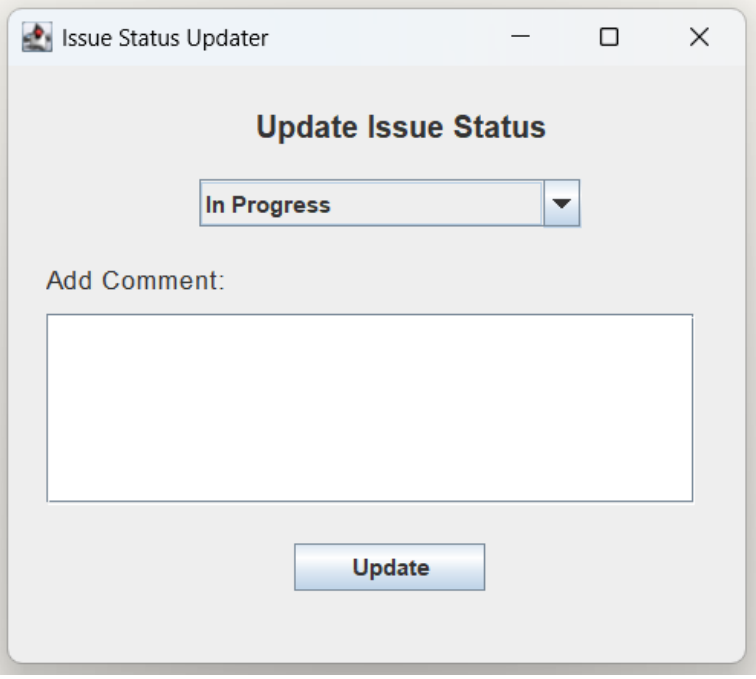


Figure 45: Update issue status screen

View Customer Feedback

This screen provides a centralized view of all customer feedback submitted by customer, enabling efficient analysis and management of customer sentiment and service quality.

- **Navigation:** The "Back" button allows the user to navigate to the previous screen in the application.
- **Apply Filters:** The "Apply Filters" button enables users to filter the displayed feedback data based on criteria such as:
  Date range (e.g., "Last month," "This year")
  Customer name (partial or exact match)
  Rating (e.g., "Positive," "Negative," "Neutral")
- **Clear Filters:** The "Clear Filters" button removes all applied filters, displaying the complete set of customer feedback.
- **Feedback List:** The screen presents a view of customer feedback entries, including:
  Feedback ID: A unique identifier for each feedback entry.
  Customer Name: The name or identifier of the customer who submitted the feedback.
  Rating: A visual representation of the customer's rating (e.g., star rating).
  Comments: The customer's written feedback or comments.
  Date Submitted: The date and time when the feedback was submitted.

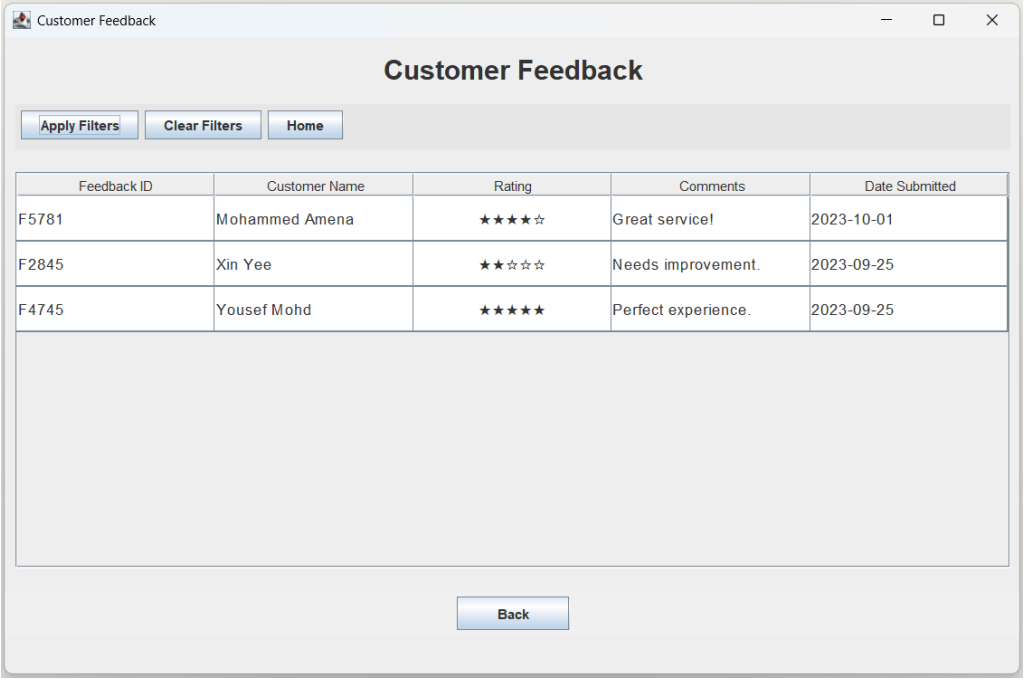


Figure 46: View customer feedback screen

5.5 Subsystem 4: Staff Screens

Main Screen

The main screen of the Staff interface allows the Staff to navigate between three tasks in the Electrical Billing System. It provides an intuitive and welcoming interface designed to simplify navigation. Below are the features of the main staff screen.

Welcome Message:
The staff is greeted with a welcoming message upon successful login as a staff.
Manage Customer Accounts:
The “Manage Customer Accounts” button takes staff to the customer account management page where they can view and update customer information.
Track Overdue Bills:
The “Track Overdue Bills” button navigates staff to overdue bills page to review and manage overdue bills.
Usage Monitoring:
The “Usage Monitoring” button provides access to monitor customer electricity usage trends.

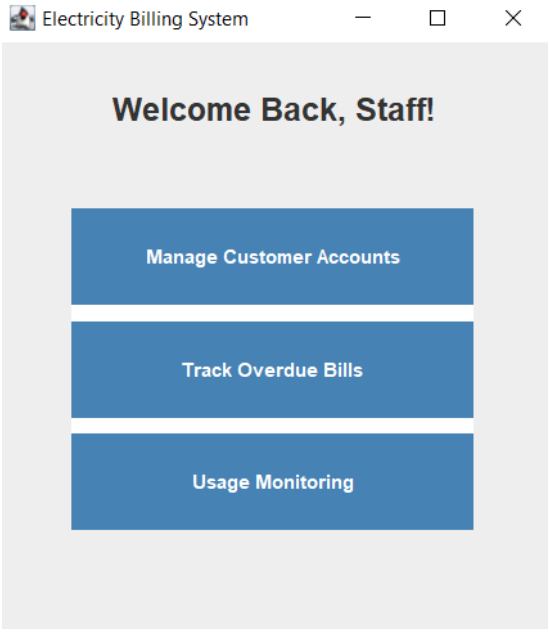


Figure 47: Staff main screen

Manage Customer Accounts Screen

The "Manage Customer Accounts" screen allows staff to manage and maintain a database of customer accounts with detailed information. This interface ensures that all customer records are easily accessible and modifiable.

Home Button:
The "Home" button allows the staff to go back to the main screen.

Search Bar:
The staff can search for a specific customer account using their account ID or name. The screen will be filtered based on the search.

Customer Accounts List:
The table displays a list of customer accounts with columns such as:

- **Account ID:** Unique identifier for each customer account.
- **Customer Name:** The name of the account holder.
- **Email:** The customer's email address.
- **Phone:** The customer's phone number.
- **Status:** Indicates the status of the customer account.
- **Actions:** A "View Details" button to view a more detailed information of the customer account, and an "Update Details" button for the staff to update the customer account details.

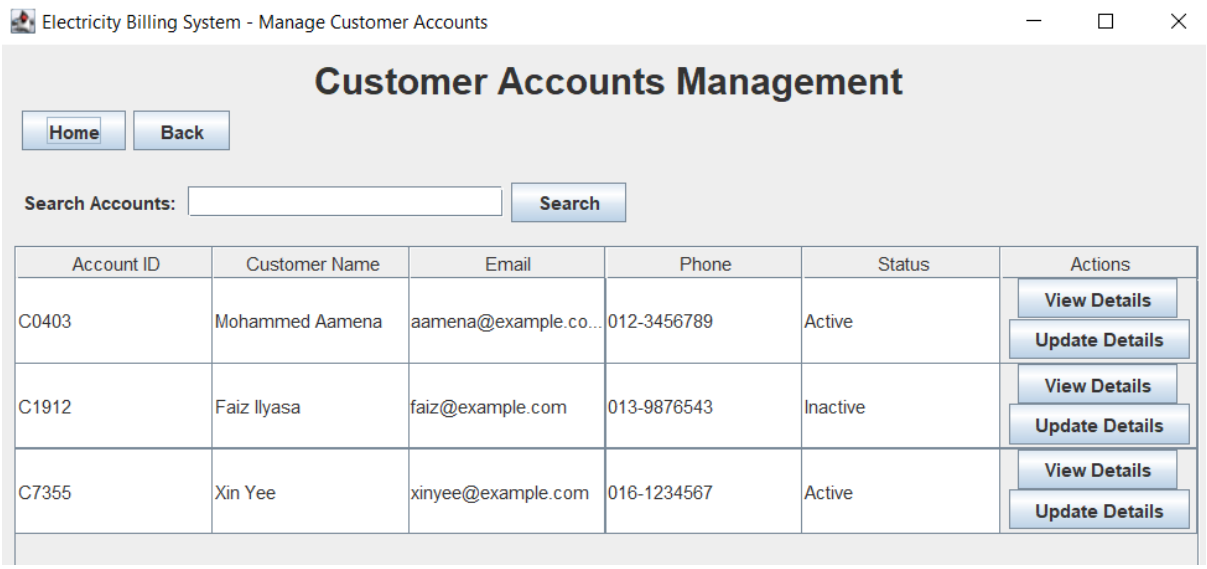


Figure 48: Manage customer accounts screen

Update Details Screen

The staff can update the information of the customer account by clicking the "Update Details" button. The staff can then input the new details such as meter ID, name, email, phone number, and address of the customer.

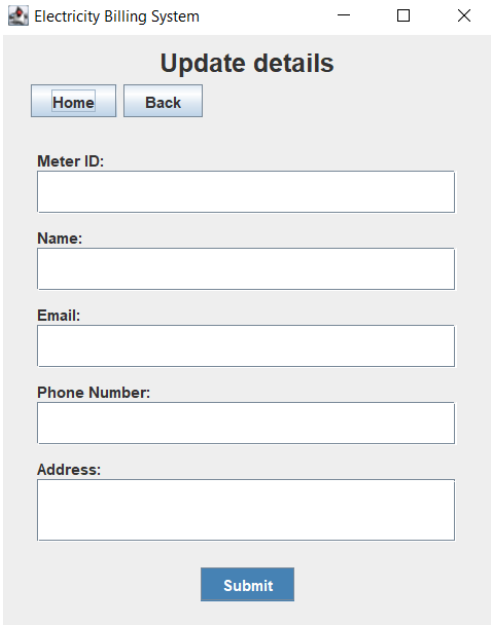


Figure 49: Update details screen

Track Overdue Bills Screen

The "Track Overdue Bills" screen is designed to help staff monitor, verify, and take necessary actions to manage overdue bills in the system. It ensures that overdue payments are identified and handled effectively.

Home Button:

The "Home" button allows the staff to go back to the main screen.

Overdue Bills List:

The table displays a list of overdue bills with columns such as:

- **Bill ID:** Unique identifier for each bill.
- **Customer Name:** The name of the account holder.
- **Amount:** The total amount due for the bill.
- **Date:** The date when the bill is generated.
- **Overdue by:** Indicates how long the bill has been overdue by.
- **Actions:** A "View Details" button to view a more detailed information of the bill details, and a "Set Final Due Date" button for the staff to set a new due date for the customer to pay their overdue bill.

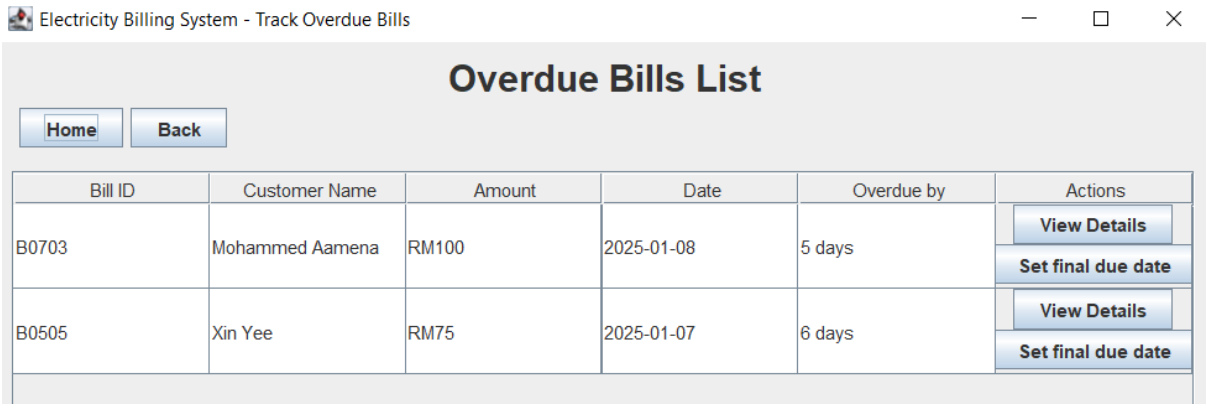


Figure 50: Track overdue bills (Staff) screen

View Details Screen

The "View Details" screen is accessible from the "Manage Customer Accounts" section by the staff. It provides a detailed description of the customer information, their billing information and payment histories. Staff can review and verify the necessary information.

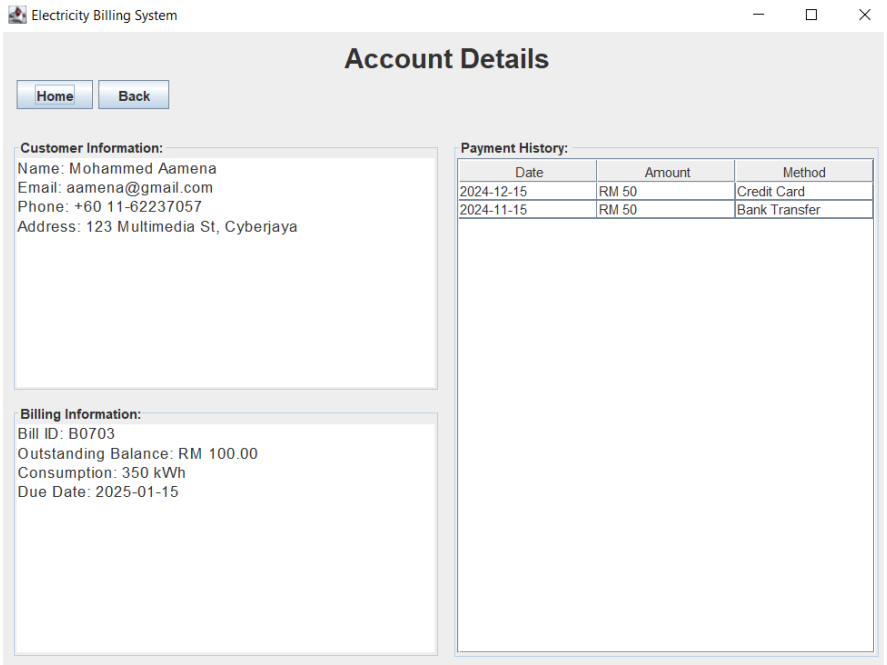


Figure 51: View details screen

- **Set Final Due Date Screen**
  The "Set Final Due Date" screen prompts the staff to key-in a new due date from a drop-down menu. This will set the final due date for the customer to pay their overdue bill.

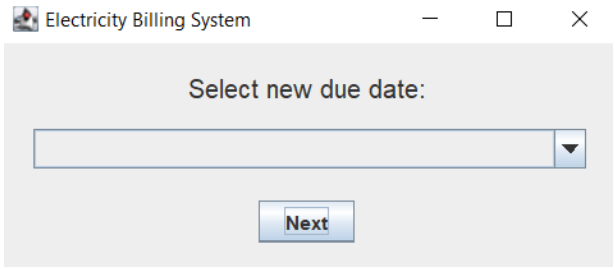


Figure 52: Set final due date screen

- **Usage Monitoring Screen**
  The "Usage Monitoring" screen offers the staff to track and analyze electricity usage trends for the overall system. This helps staff identify irregularities or patterns in usage.

  Home Button:
  The "Home" button allows the staff to go back to the main screen.

  Month and Year Filter:
  This allows the staff to filter the analytics based on a specific month and year.

  Usage Metrics:
  The usage metrics section shows a breakdown of important analytics for the staff to view such as total customers, total bills generated, total bills paid, total electricity usage, average electricity usage, and trend.

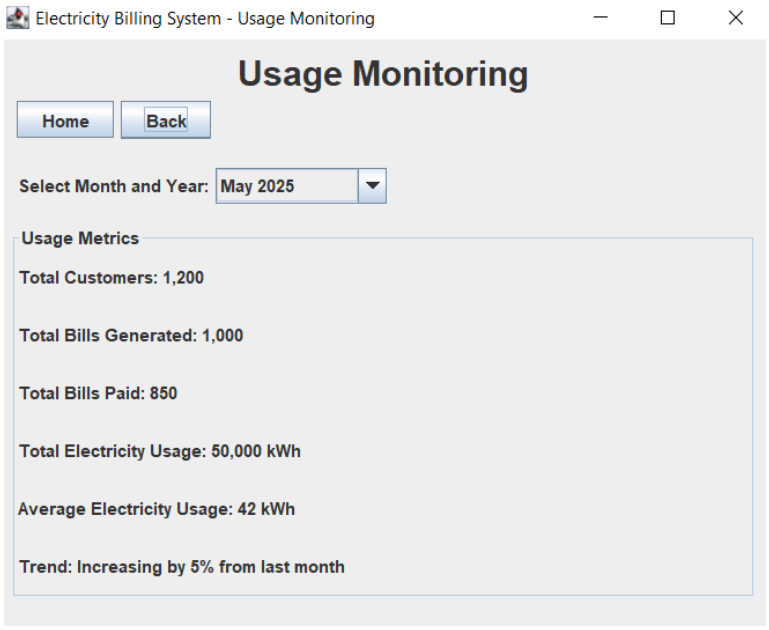


Figure 53: Usage monitoring screen

6 Component Design

- **Main Components**
  This section provides an overview of the core building blocks that comprise the Electricity Billing System. It outlines the major components and their interdependencies, forming the foundation of the system's architecture. This understanding is crucial for comprehending the system's overall design and functionality.

6.1.1 Component 1: System Registration

The activity diagram illustrates in *Figure 54* the workflow for the System Registration process. The process begins when the customer clicks on the "Register" button, initiating the opening of the registration page. The customer is then prompted to enter their Meter ID. The system validates the entered Meter ID. If valid, the full registration form is displayed, requiring the customer to enter their personal data. Upon clicking "Submit," the system validates the entered data for completeness and correctness. If all validations are successful, a new customer account is created, and a success message is displayed to the customer.

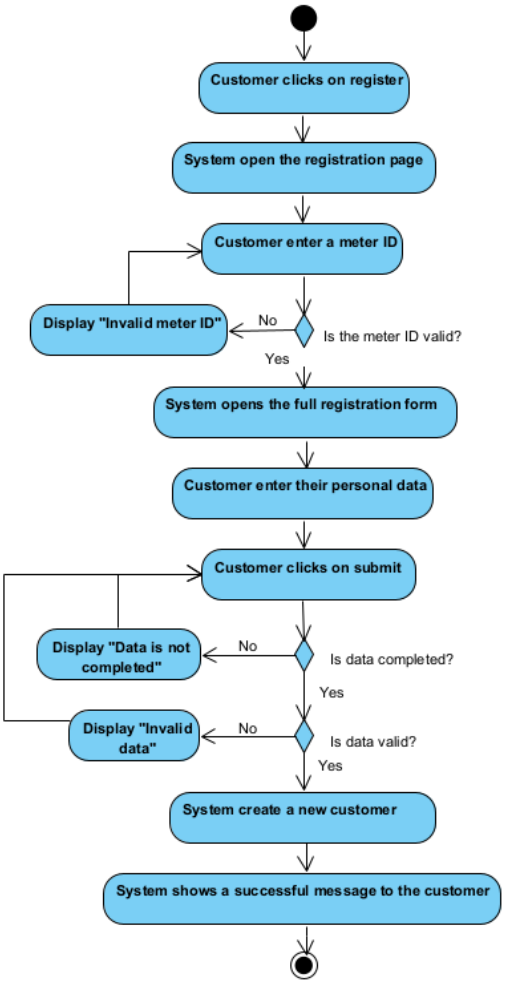


Figure 54: System registration activity diagram

6.1.2 Component 2: Log in to the System

The activity diagram in *Figure 55* illustrates the user login process. The process begins with the user entering their login data, which typically includes their username and password. Once the user clicks the "Login" button, the system validates the provided data. If the data is invalid (e.g., incorrect username or password), the system displays an error message. However, if the data is valid, the system successfully logs in the user and redirects them to their respective dashboard based on their assigned role within the system.

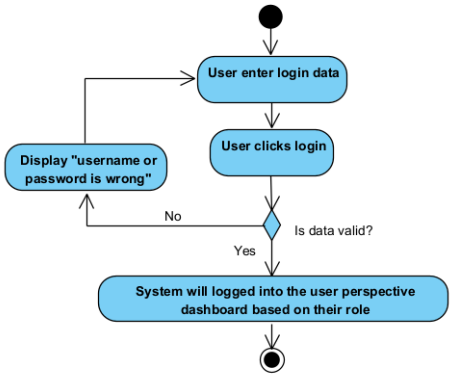


Figure 55: Login activity diagram

6.1.3 Component 3: Reset Password

The activity diagram in *Figure 56* depicts the process for resetting a forgotten password. The sequence begins with the customer initiating the process by accessing the "Reset Password" page. The system then presents the customer with a form to enter their personal information for verification. Following the entry of personal information, the system validates the provided details. If the information is deemed valid, the system proceeds to the stage where the customer sets a new password. Subsequently, the system enforces password policies, ensuring the new password meets the established security criteria. Upon successful validation of the new password, the system updates the customer's password in the database. The process concludes with the system displaying a success message to the customer, confirming the successful completion of the password reset.

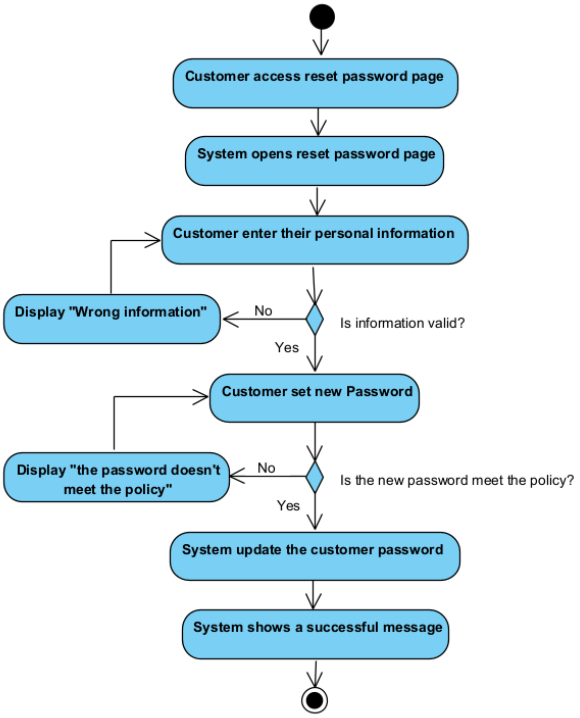


Figure 56: Reset password activity diagram

6.1.4 Component 4: Make a Payment and Receive Payment Receipt

The activity diagram in *Figure 57* illustrates the workflow for making a payment within the Electricity Billing System. Initiated by the customer clicking the "Make a Payment" button, the system first checks for the presence of any active bills. If no active bills are found, the system displays a message informing the customer. Conversely, if active bills exist, the system presents the customer with a list of active bills for selection. Subsequently, the customer chooses the desired bill and payment method and confirms the payment. The system then redirects the customer to the selected payment gateway to complete the transaction. Upon completion, the system verifies the payment status. If the payment is successful, a success message is displayed; otherwise, an error message is presented.

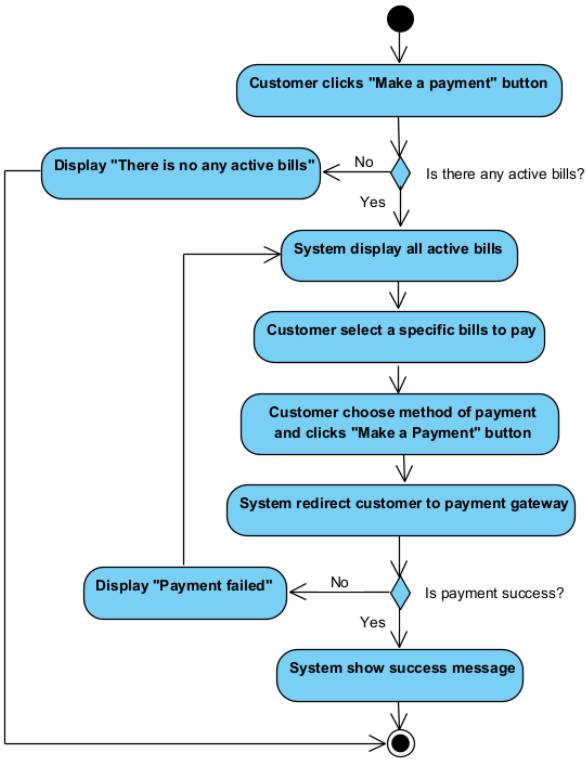


Figure 57: Make a payment activity diagram

6.1.5 Component 5: View Bill Details

The activity diagram shown in *Figure 58* illustrates the process of viewing bill details for a customer. The process begins when the customer navigates to the "View Bill Details" section. The system then checks if there are any active bills associated with the customer. If no bills are found, the system displays a message indicating "No bills found." If active bills are found, the system displays a list of all active bills to the customer. The customer then selects a specific bill from the list. Finally, the system retrieves and displays the detailed information of the selected bill to the customer.

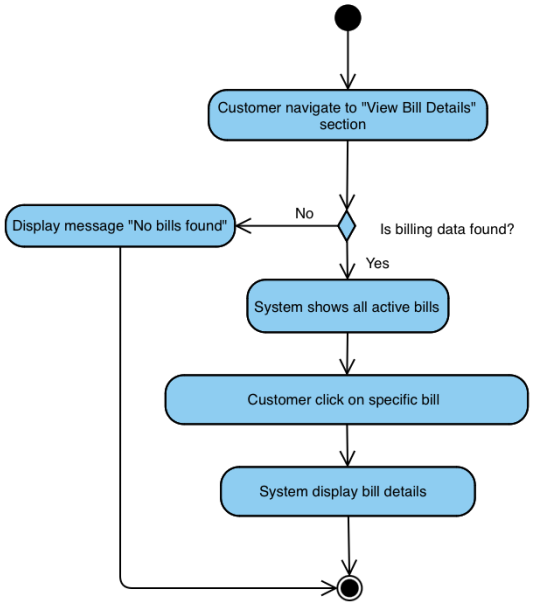


Figure 58: View bill details activity diagram

6.1.6 Component 6: Update Meter Readings and Tariff

The activity diagram in *Figure 59* shows the process of updating the meter readings and tariff rates for the utility provider. The process begins when the utility provider accesses the “Meter Readings and Tariff" page. Then, the system will present them with a form to update the rates data. Once the new data is entered by the utility provider, the system will check with the database to validate the data entered. If the data is invalid, an error message will be shown, otherwise the system will display a success message and update the database with the new rates.

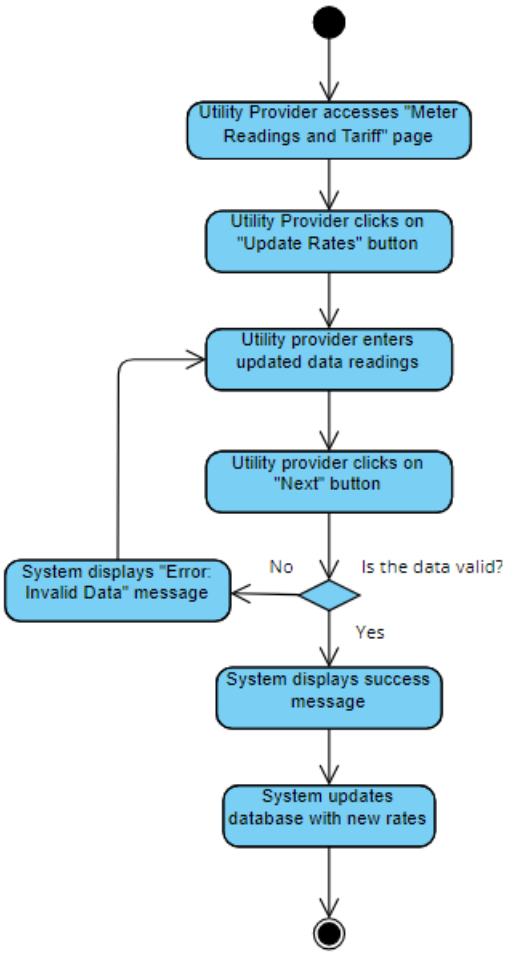


Figure 59: Update meter reading and tariffs activity diagram

6.1.7 Component 7: Track Overdue Bills

The activity diagram in *Figure 60* illustrates the tracking of overdue bills for the utility provider. The process begins when the utility provider accesses the “Track Overdue Bills" page. Then, the system will first check if there are any overdue bills. If there are no overdue bills, the system will display “No overdue bills found” and end the process. If there are overdue bills, the system displays them and the utility provider can verify them. The utility provider can then proceed to set a penalty amount to the customers who have overdue bills passed the final due date set by staff. If they have not passed

the final due date, the system will display “Final due date not yet passed”. The utility provider will then enter the penalty amount for the system to save into the database and the customer will be notified.

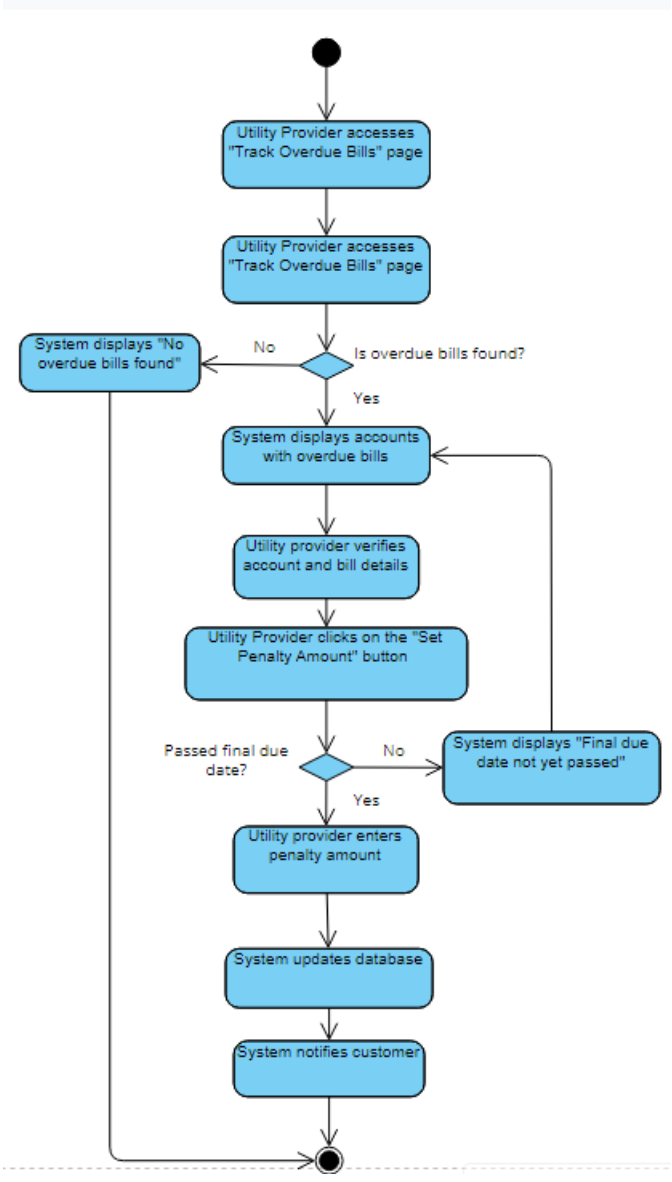


Figure 60: Track overdue bills activity diagram

6.1.8 Component 8: Generate Monthly Bills

The activity diagram shown in the *Figure 61* illustrate the process of generating and delivering the electricity bills. The process begins when the utility provider schedules the date for bill generation within the system. Subsequently, the system receives the customer data from the database, including consumption details and relevant account information. The system calculates the bill amount based on the customer’s

consumption and applicable tariffs. The system generates a detailed bill and saves it into the database. The customer then receives a notification about the bill.

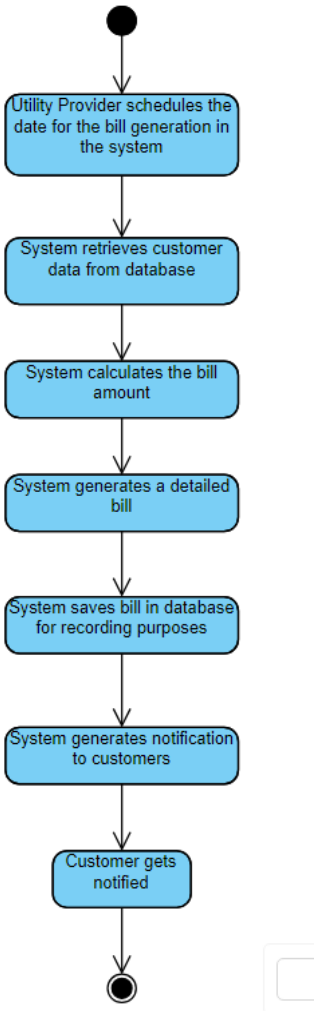


Figure 61: Generate monthly bills activity diagram

6.1.9 Component 9: Submit Feedback

The activity diagram in *Figure 62* outlines the process of submitting customer feedback within the Electricity Billing System. The process begins when the customer navigates to the designated feedback submission section. Subsequently, the customer is required to complete the feedback form by providing the necessary information. Upon clicking the "Submit" button, the system validates the form for completeness. If any required fields are missing, the system displays an error message, prompting the customer to complete the form. If the form is complete, the system generates a unique identifier for the submitted feedback. Finally, the system stores the feedback, along with the associated identifier, within the system's database.

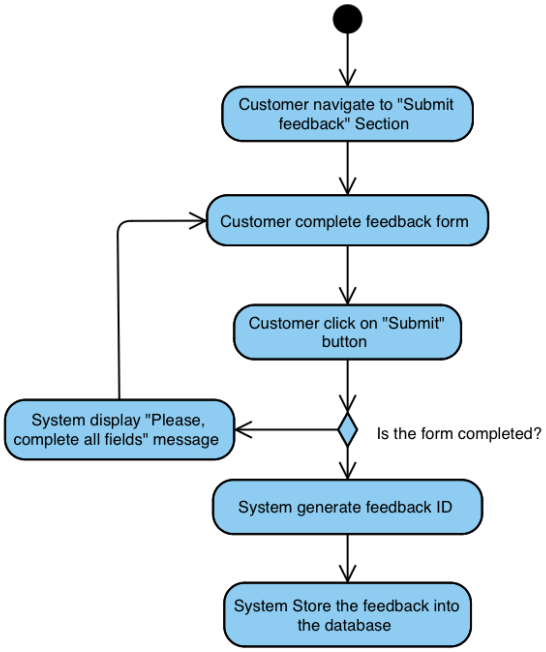


Figure 62: Submit feedback activity diagram

6.1.10 Component 10: View Customer Bills

The activity diagram in *Figure 63* depicts the workflow for the Support Admin when viewing and managing customer bills within the Electricity Billing System. The process commences with the Support Admin navigating to the "View Customer Bill" section. Subsequently, the system retrieves a comprehensive list of customer bills. The admin then proceeds to enter search keywords to locate a specific customer bill and initiates the search by clicking the designated button. If no bills match the search criteria, the system displays a notification indicating "No bills found," prompting the admin to refine the search keywords. If a match is identified, the admin selects the specific bill. The system then retrieves the detailed billing information pertaining to the selected bill. If no billing information is available for the selected bill, the system displays a notification indicating "No billing information available" Conversely, if billing data is retrieved, the system presents the billing information to the Support Admin. Finally, the Support Admin reviews the billing information and proceeds to assist the customer with any inquiries or issues related to their bill.

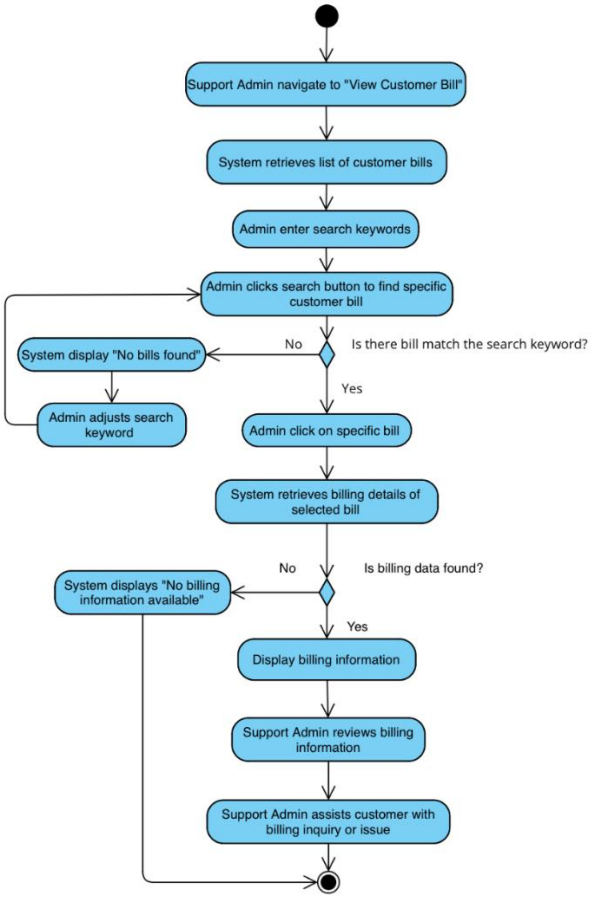


Figure 63: View customer bills activity diagram

6.1.11 Component 11: View Customer Feedback and Analysis

The activity diagram shown in *Figure 64* illustrates the workflow for a Support Admin when viewing and analyzing customer feedback within the Electricity Billing System. The process begins with the Support Admin selecting the "View Customer Feedback" option. The system then retrieves and displays a comprehensive list of all submitted customer feedback.

The Support Admin applies filters to narrow down the feedback based on specific criteria, such as star rating, keywords, or comments. If no feedback matches the selected filters, the system displays a message stating, "No feedback found for the selected filters," prompting the Admin to modify the filter criteria.

If matching feedback entries are found, the system presents the filtered list to the Admin. The admin analyzes this feedback to identify trends, areas for improvement, and potential service issues. Based on the analysis, the admin takes appropriate actions to enhance the electricity system services.

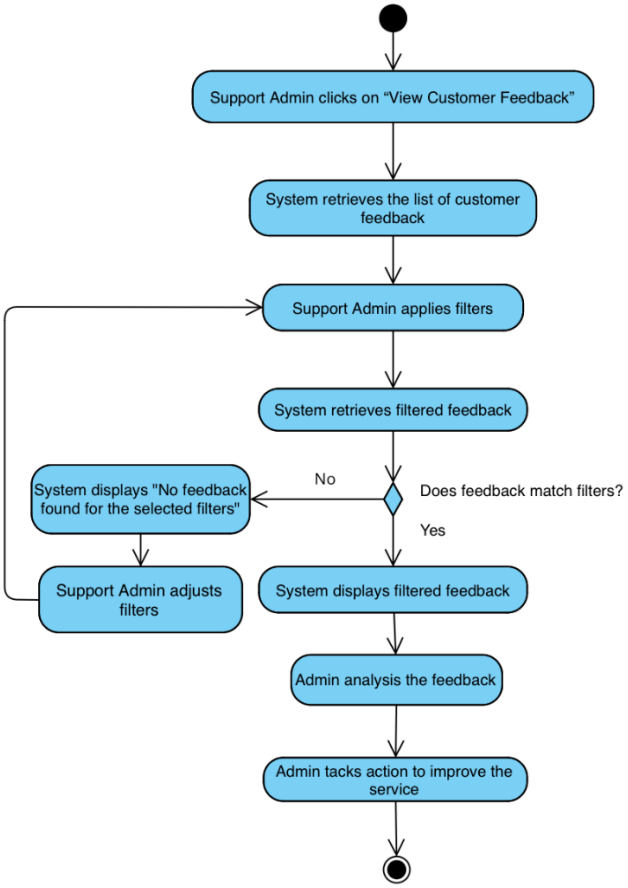


Figure 64: View customer feedback activity diagram

6.1.12 Component 12: Submit and Resolve Customer Issue

The activity diagram below in *Figure 65* describes the workflow followed in the Electricity Billing System for addressing customer issues. The process starts with a customer going to the Support part of the site and finding the reporting form. The customer fills out the form, providing most everything that needs to be provided, as in what the issue is, time, date, and time of occurrence, and supporting information including any attachment that might help. When completed, it could be submitted via clicking at the bottom of the form the 'submit' button.

After submission, the system validates the form's completeness. Depending on the result of validation, if some required fields were indicated empty or they showed invalid data, an error message will appear alerting the customer to rectify this before submitting it again. If validation turns out to be successful, the system will identify the ticket uniquely to allow tracking of the issue and classify it as 'In Progress'.

At the same time, the system sends a confirmation message to the customer, acknowledging him for submitting the issue, along with the ticket ID which serves as reference to it. The system then assigns this particular issue to the support administrator who handles it. The administrator will look through the issue details, ascertain whatever needs to be done to resolve it, and then take necessary action. Such actions are troubleshooting, if need be, liaising with other departments, or even contacting the customer directly.

Once the issue is resolved, the respective administrator will change the system status of the issue to reflect that it is completed. The system then alerts the user that their request has been resolved, providing a summary or follow-up, if applicable. The process is then closed for resolution of issues pertaining to the Electricity Billing System.

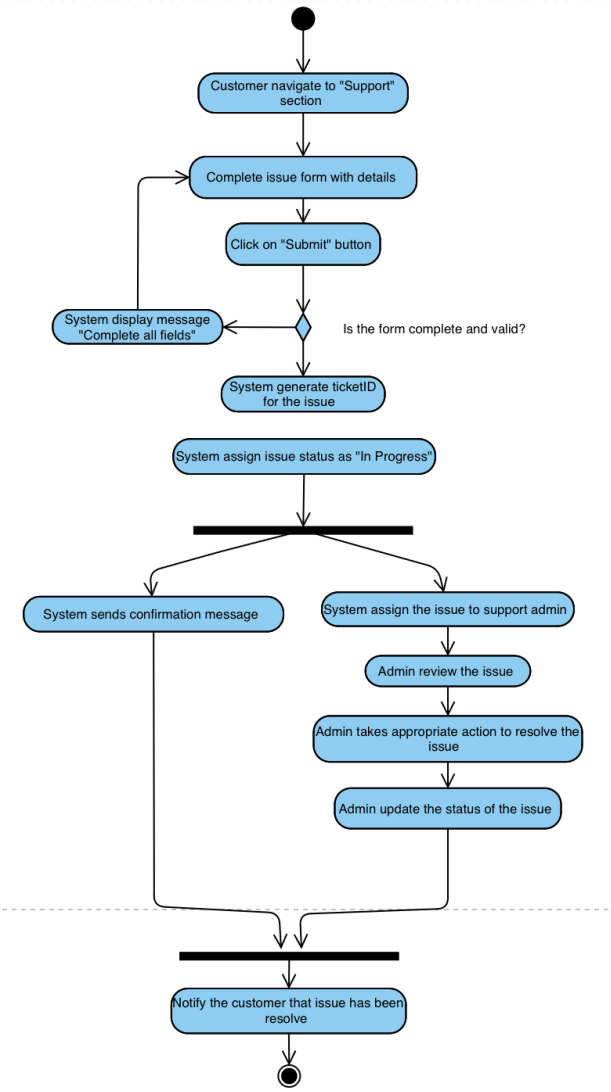


Figure 65: Submit and resolve customer issue activity diagram

6.1.13 Component 13: Manage Customer Accounts

The staff can navigate to the "Manage Customer Accounts" page from the staff's main menu. Once entered, the system will retrieve the list of customers accounts which will be displayed on the screen. For ease of managing accounts, staff can search for a specific account via the search field. The system checks if the search is valid and will display an error message if no accounts is found such as "No account found". The staff can then view and update the customer account to make necessary changes. The system will validate the input and displays "Update failed. Please try again." if update is unsuccessful. After the system has been updated, a notification will be sent to the customer.

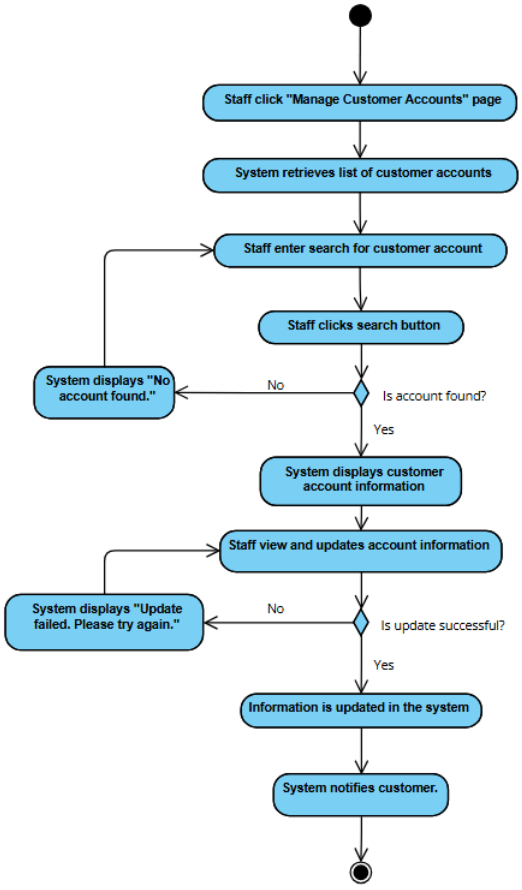


Figure 66: Manage customer accounts activity diagram

6.1.14 Component 14: Track Overdue Bills

Staff can access the "Track Overdue Bills" page from the staff's main menu. The system will automatically retrieve a list of customer accounts with overdue bills. If no accounts with overdue bills are found, the system will display "No overdue bills found". If there are overdue bills, the staff can view the bill details to monitor and verify overdue bills. The staff is also able to set a new due date for the overdue bill. The system will validate the date and if the date is invalid, it will display "Please select a valid date". Upon a valid date, the system will be updated, and it will send the customer a notification.

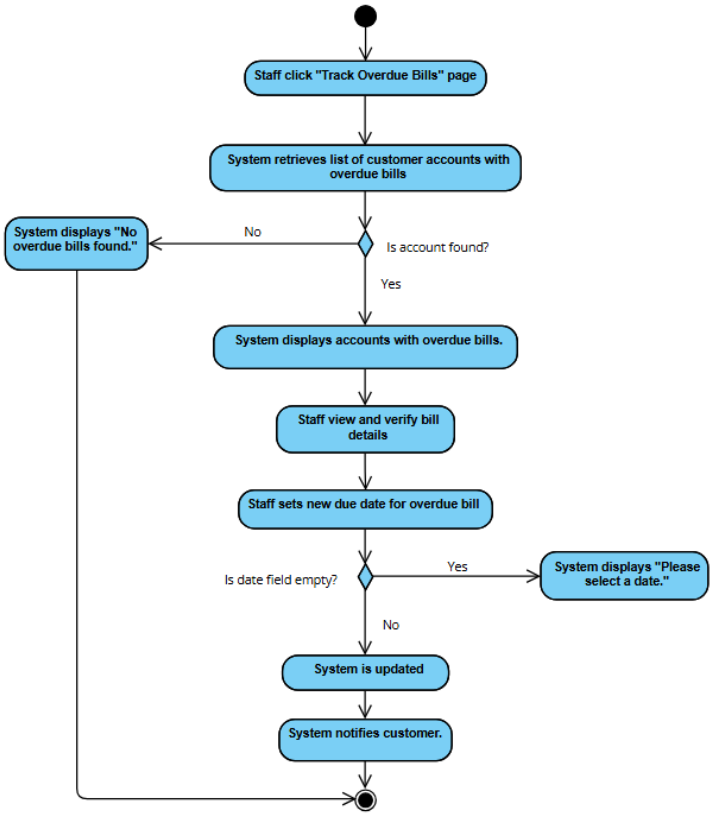


Figure 67: Track overdue bills activity diagram

6.1.15 Component 15: Usage Monitoring

The staff can select the “Usage Monitoring” page from the staff’s main menu. The staff can then filter the usage metrics by month and year. If no data is available, the system will display “No data available”. If data is available, the system will display the metrics which the staff can review and monitor.

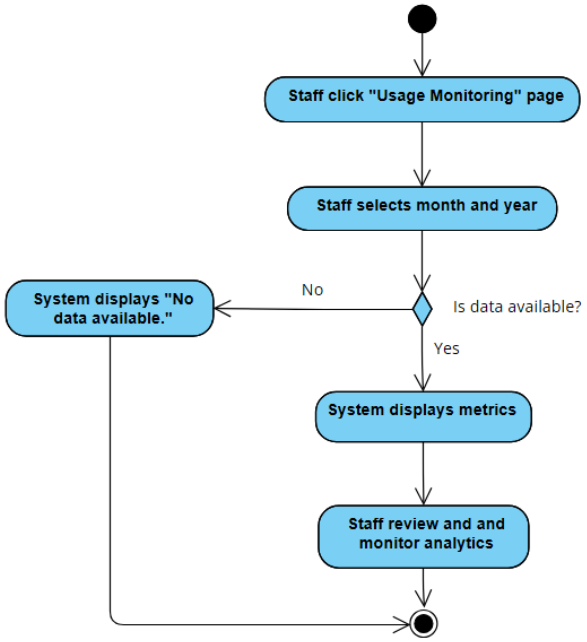


Figure 68: Usage Monitoring activity diagram

7 Deployment Design

7.1 Deployment Diagram

7.1.1 Customer Deployment Diagram

This deployment diagram illustrates the architecture of the Electricity Billing System. The system is comprised of a Web Server hosting the core application, which encompasses components such as System Registration, Log in to the System, Reset Password, Make Payment and Receive Payment Receipt, Submit Feedback, Submit Issues and View Bill Details. These components interact with a suite of databases, including Bill Database, Meter Database, Payment Database, Customer Database, Feedback Database, and Issue Database, for data storage and retrieval. The system is accessed by Customer PCs equipped with web browsers, enabling customers to interact with the application and perform various tasks such as registration, login, payment, feedback submission, and issue reporting.

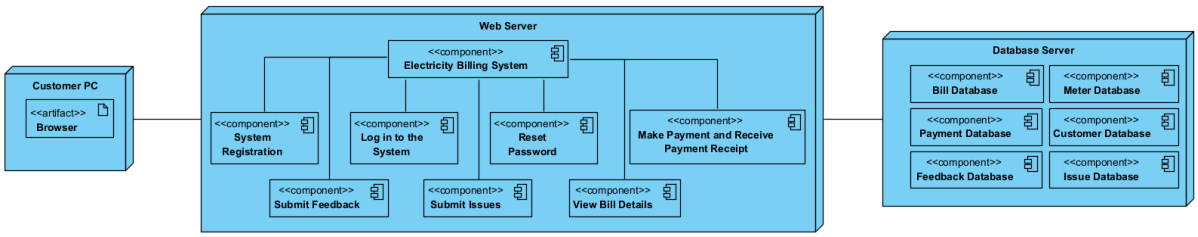


Figure 69: Customer deployment diagram

7.1.2 *Utility Provider*

The Utility Provider Deployment Diagram shows components that can be accessed by the Utility Provider. When the utility provider logs into their PC, they can access three components which are Track Overdue Bills, Generate Monthly Bills, and Update Meter Readings and Tariffs. These components are supported by databases such as Utility Provider Database, Customer Database, Bill Database, Payment Database, Meter Database, Issue Database, and Feedback Database.

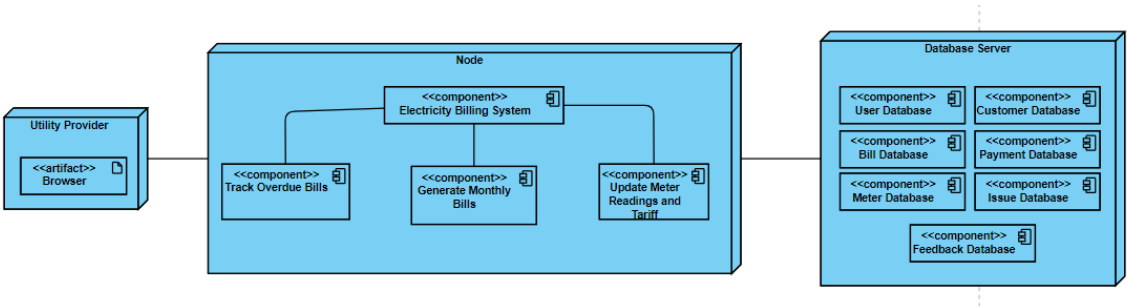


Figure 70: Utility provider deployment diagram

7.1.3 Support Admin

The Support Admin Deployment Diagram illustrates the architecture of the Electricity Billing System with a focus on the components accessible by support admins. The system is built around a central Web Server that hosts the application, which includes components such as View Customer Feedback, View Customer Bills, and Resolve Customer Issues. These components interface with a set of databases that store essential data for the system's operation. The databases include the Customer Database, Meter Database, Feedback Database, Payment Database, Bill Database, Issue Database, and SupportAdmin Database to retrieve the necessary data. The entire system can be accessed via Admin PCs with web browsers so that the members of administration can interface with the application to do their administrative work.

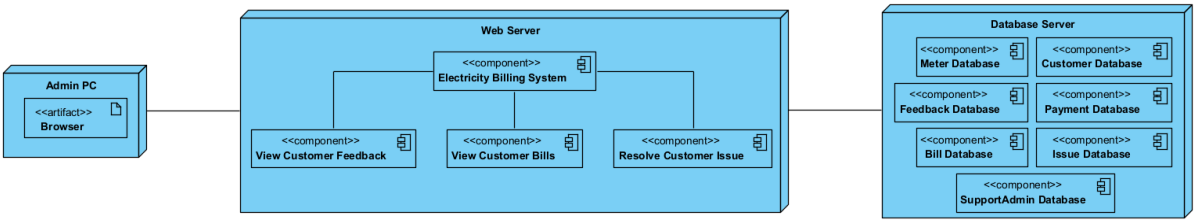


Figure 71: Support admin deployment diagram

7.1.4 Staff

The Staff Deployment Diagram shows that when a staff member uses their PC to log into the Electricity Billing System, they can access three components which are Manage Customer Accounts, Track Overdue Bills, and Usage Monitoring. Each of these components are linked to their corresponding databases such as Staff Database, Customer Database, Bill Database, and UsageMetric Database to retrieve necessary data.

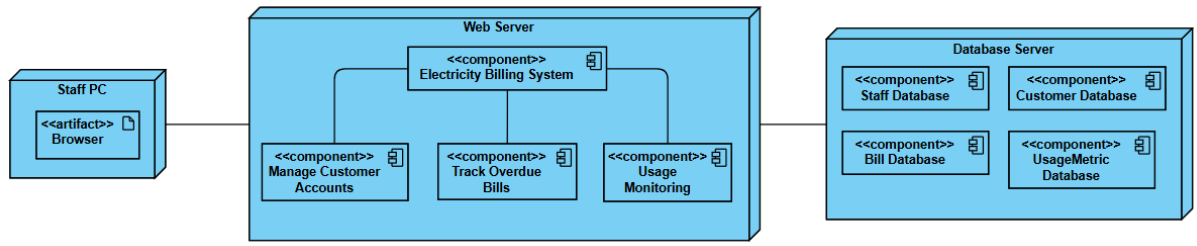


Figure 72: Staff deployment diagram