**Electricity Billing System**

**Table of Contents**

* [Overview](https://chatgpt.com/c/67ed8ae0-b110-8004-a781-427bea3f7ed0#overview)
* [Features](https://chatgpt.com/c/67ed8ae0-b110-8004-a781-427bea3f7ed0#features)
* [Technologies Used](https://chatgpt.com/c/67ed8ae0-b110-8004-a781-427bea3f7ed0#technologies-used)
* [Installation](https://chatgpt.com/c/67ed8ae0-b110-8004-a781-427bea3f7ed0#installation)
* [Usage](https://chatgpt.com/c/67ed8ae0-b110-8004-a781-427bea3f7ed0#usage)
* [Contributing](https://chatgpt.com/c/67ed8ae0-b110-8004-a781-427bea3f7ed0#contributing)
* [License](https://chatgpt.com/c/67ed8ae0-b110-8004-a781-427bea3f7ed0#license)
* [Contact](https://chatgpt.com/c/67ed8ae0-b110-8004-a781-427bea3f7ed0#contact)

**Overview**

The **Electricity Billing System** is a software application designed to automate the process of electricity bill generation and management. This system helps electricity providers efficiently manage customer billing, payments, and usage tracking.

**Features**

* User registration and authentication
* Customer account management
* Automated bill calculation based on electricity consumption
* Online bill payment integration
* Usage history and bill tracking
* Admin dashboard for managing users and bills
* Report generation for usage statistics

**Technologies Used**

* **Frontend:**Java
* **Backend:** Java
* **Database:** MySQL
* **Authentication:** JWT or OAuth
* **Payment Integration:** PayPal/Stripe/Razorpay (optional)

**Installation**

**Prerequisites**

* Node.js and npm installed
* MongoDB set up (or any other database you prefer)

**Steps to Install**

1. Clone the repository:
2. git clone https://github.com/your-username/electricity-billing-system.git
3. cd electricity-billing-system
4. Install dependencies:
5. npm install
6. Set up environment variables in a .env file:
7. PORT=5000
8. DATABASE\_URL=mongodb://localhost:27017/electricity\_billing
9. JWT\_SECRET=your\_secret\_key
10. Run the server:
11. npm start

**Usage**

* Register/Login as a customer or admin
* View electricity usage and billing history
* Generate and pay electricity bills
* Admin can manage users and generate reports

SQL

drop database bills;

CREATE DATABASE IF NOT EXISTS bills;

USE bills;

DROP DATABASE bills;

-- Create the login table with correct column names

CREATE TABLE IF NOT EXISTS login (

meter\_no INT PRIMARY KEY,

name VARCHAR(50),

username VARCHAR(50),

pass VARCHAR(50)

);

create table if not exists meter\_info(

meter\_no INT PRIMARY KEY,

meter\_location varchar(255),

meter\_type varchar(255),

phase\_code int,

bill\_type varchar(255),

days int

);

drop table meter\_info;

CREATE TABLE IF NOT EXISTS unit\_consumption (

meter\_no INT,

units\_consumed INT,

month VARCHAR(20)

);

-- Create the customer table with a foreign key relationship

CREATE TABLE IF NOT EXISTS customer (

meter\_no INT,

address VARCHAR(100),

city VARCHAR(50),

state VARCHAR(50),

email VARCHAR(50),

phone BIGINT

-- FOREIGN KEY (meter\_no) REFERENCES login(meter\_no)

);

drop table customer;

-- Create the tax table to store tax information

CREATE TABLE IF NOT EXISTS tax (

cost\_per\_unit INT,

meter\_rent INT,

service\_charge INT,

service\_tax INT,

swacch\_bharat\_cess INT,

fixed\_tax INT

);

-- Create the bill table to store calculated bills

CREATE TABLE IF NOT EXISTS bill (

meter\_no INT,

month VARCHAR(20),

units INT,

total\_bill INT,

status VARCHAR(20)

);

create table if not exists adminLogin(

name VARCHAR(50),

username VARCHAR(50),

pass VARCHAR(50)

);

-- Insert sample data into login

-- INSERT INTO login (meter\_no, name, username, pass) VALUES

-- (8396485, 'John Doe', 'john123', 'pass123'),

-- (8396486, 'Jane Smith', 'jane456', 'pass456');

-- Insert sample data into customer

INSERT INTO customer ( meter\_no, address, city, state, email, phone) VALUES

(8396485, '123 Elm St', 'New York', 'NY', 'john.doe@email.com', 1234567890),

(8396486, '456 Oak St', 'Los Angeles', 'CA', 'jane.smith@email.com', 9876543210);

-- Insert sample data into tax

INSERT INTO tax (cost\_per\_unit, meter\_rent, service\_charge, service\_tax, swacch\_bharat\_cess, fixed\_tax) VALUES

(7, 50, 20, 18, 5, 30);

-- Verify data

SELECT \* FROM login;

SELECT \* FROM customer;

SELECT \* FROM tax;

select \*from unit\_consumption;

select \*from meter\_info;

SELECT login.name, customer.address

FROM login

JOIN customer ON login.meter\_no = customer.meter\_no

WHERE login.meter\_no = 8396485;

select \* from bill;

SELECT \* FROM customer;

select \* from adminLogin;

**Contributing**

Contributions are welcome! Please follow these steps:

1. Fork the repository
2. Create a feature branch (git checkout -b feature-name)
3. Commit changes (git commit -m "Add feature")
4. Push to your branch (git push origin feature-name)
5. Open a pull request

**License**

This project is licensed under the [MIT License](https://chatgpt.com/c/LICENSE).

**Contact**

For any queries, reach out via:

* Email:kirtikushwaha000@gmail.com