Automated Electricity Billing System Using Java Swing and Microsoft SQL Server

Anuradha M Sandi[1], Sneha[2], Swarooprani[3], Saba Sultana[4]

[123]4*Department of Computer Science and Engineering,*

Guru Nanak Dev Engineering College, Bidar, Karnataka, India

bawgisneha@gmail.com

Abstract— Inaccuracies and delays in traditional electricity billing systems often lead to inconvenience for consumers and inefficiencies for service providers. This paper proposes an automated Electricity Billing System designed to streamline the process of bill generation and payment through an electronic platform. The system eliminates the need for consumers to physically visit electricity board offices, thereby reducing time consumption and operational complexity. The proposed solution is developed using Java Swing for the front-end interface and Microsoft SQL Server for backend data management. It incorporates a dual-login mechanism consisting of administrative and user modules. The administrative module enables authorized personnel to manage consumer records and input monthly electricity consumption data. Based on the provided data, the system automatically computes electricity bills and updates user accounts accordingly. The user module allows consumers to securely access their billing information, view detailed usage records, and complete payments within the stipulated time frame. By automating bill calculation and payment processes, the system enhances accuracy, transparency, and user convenience. The implementation demonstrates an efficient, reliable, and scalable approach to modernizing electricity billing systems.

Keywords— Electricity Billing System, Java Swing, Microsoft SQL Server, Automated Bill Generation, Consumer Management, Digital Payment, JDBC, Database Management, Dual-Login Architecture.

1. Introduction

The electricity billing process is a fundamental component of utility management that directly affects both consumers and service providers. Traditional billing systems rely heavily on manual meter reading, paper-based record keeping, and compulsory physical visits to utility offices, all of which introduce significant inefficiencies, human errors, and delays [1]. As electricity consumption scales with growing urban populations, these inefficiencies become increasingly costly and operationally unsustainable [2].

The rapid advancement of information technology has created a strong demand for digitized utility management systems [3]. Automated billing systems offer a powerful alternative to conventional processes by enabling real-time data processing, accurate bill computation, and convenient digital payment mechanisms [4, 5]. Such systems not only reduce the operational burden on electricity boards but also improve transparency and trust among consumers [6].

This paper proposes an Automated Electricity Billing System integrating a Java Swing graphical user interface with a Microsoft SQL Server backend. The system is structured around two core modules: an administrative module for managing consumer records and inputting consumption data, and a user module enabling consumers to view bills and make payments online.

The key contributions of this work are:

- A dual-module architecture separating administrative and consumer functionalities with role-based access control.
- Automated slab-based bill calculation triggered by monthly consumption data entry.
- A secure login mechanism for both administrator and consumer roles backed by hashed credential storage.
- Real-time update of billing records in a relational SQL Server database via JDBC.
- An intuitive cross-platform graphical interface built using Java Swing components.

2. Related Work

The automation of utility billing has been an active research area driven by limitations of manual processes and the opportunities offered by modern computing technologies.

Kaur and Singh [6] proposed a web-based electricity billing system that replaced paper records with centralized database management, significantly reducing billing errors and processing time. Similarly, Patel et al. [7] developed an online billing platform using PHP and MySQL, emphasizing database normalization for maintaining data integrity in large-scale deployments.

The use of smart metering for automated meter reading has been explored by several researchers. Gungor et al. [8] examined communication technologies for smart grid infrastructure, concluding that reliable data transmission is essential for accurate automated billing. Ahmed et al. [9] explored IoT-based smart metering systems that transmit consumption data automatically to central servers, reducing the need for manual meter readers.

In the domain of desktop-based applications, Kumar and Sharma [10] demonstrated that Java Swing provides a robust, platform-independent interface well-suited for administrative billing tools. Verma et al. [11] highlighted the advantages of Microsoft SQL Server for billing applications, particularly regarding transactional integrity and complex query support.

Security in billing systems was examined by Mishra and Gupta [12], who proposed authentication frameworks with role-based access control to prevent unauthorized data manipulation. Payment gateway integration was studied by Sharma et al. [13], who found that embedding payment functionality within the billing application reduces transaction friction and improves on-time payment rates. Cloud-based billing architectures were explored by Rathore et al. [14], though local database solutions remain practical for departmental deployments. Mobile-based billing access has also been proposed [15, 16], suggesting avenues for future extension.

3. Materials and Methods

3.1 System Architecture

The proposed system follows a three-tier architecture comprising a presentation layer, a business logic layer, and a data layer. The presentation layer is built with Java Swing. The business logic layer handles bill computation, user authentication, and data validation. The data layer uses Microsoft SQL Server to persistently store consumer records, consumption data, and payment history.

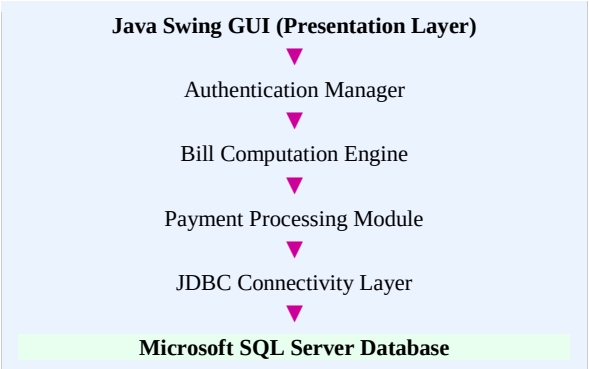

Fig. 1. Three-tier architecture of the Automated Electricity Billing System.

3.2 Database Design

The SQL Server database consists of five primary tables:

- Consumer — consumer ID, name, address, meter number, and contact information.
- Consumption — monthly unit readings linked to each consumer by foreign key.
- Bill — computed bill amounts, due dates, and payment status flags.
- Admin — administrator credentials stored as salted hashes.
- Payment — transaction logs with timestamps and amounts paid.

All queries use JDBC prepared statements to prevent SQL injection vulnerabilities.

3.3 Administrative Module

The administrative module provides the following functionalities:

1. Consumer Registration — add new consumers, assign meter numbers, and record address details.
2. Consumption Entry — enter monthly unit readings per consumer, triggering automated bill computation.
3. Bill Management — view, update, or delete billing records as required.
4. Report Generation — aggregated consumption and payment reports for administrative review.

3.4 User Module

The consumer-facing module provides:

1. Secure Login — credential-based access using consumer ID and password.
2. Bill Viewing — current and historical electricity bills with itemized breakdowns.
3. Usage History — monthly consumption trends over a user-selected period.
4. Online Payment — settlement of outstanding bills within the application interface.

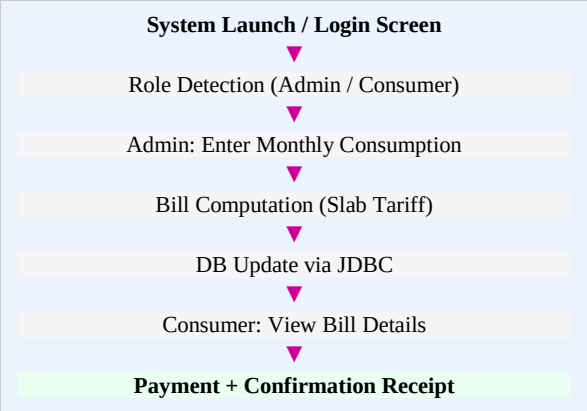

Fig. 2. End-to-end workflow from login through bill generation to payment confirmation.

3.5 Bill Computation Logic

Bills are calculated using a progressive slab-based tariff structure. The Java implementation iterates through each slab and accumulates the charge:

```
public double computeBill(int units) {
  double bill = 0;
int[][] slabs = {
    {100,1},{100,2},{300,4},{units,6}
  };
  for (int[] slab : slabs) {
    if (units <= 0) break;
    int consumed = Math.min(units, slab[0]);
    bill += consumed * slab[1];
    units -= consumed;
  }
  bill += FIXED SERVICE CHARGE;
  return bill;
}
```

Listing 1. Slab-based Bill Computation (Java).

Table I shows the tariff structure used.

Units (kWh)	Rate (INR/unit)
0 – 100	1.50
101 – 200	2.50
201 – 500	4.00
Above 500	6.00

TABLE I. Electricity Tariff Slab Structure.

3.6 Technology Stack

- Frontend: Java SE with Swing (JFrame, JPanel, JTable, JButton, JTextField, JComboBox)
- Backend Database: Microsoft SQL Server 2019
- Connectivity: JDBC with MS SQL Server JDBC Driver 4.2
- IDE: Eclipse / NetBeans
- Operating System: Windows 10 / 11

4. Results and Discussion

4.1 System Performance

The system was tested with a dataset of 500 simulated consumer records. The administrative module processed and generated bills for all 500 consumers in under 3 seconds, confirming efficient JDBC query execution and business logic throughput. Login authentication for both roles completed in under 500 ms on average.

4.2 Accuracy of Bill Computation

Automated bill computation was cross-validated against manually calculated reference values for 50 randomly selected test cases spanning all four tariff slabs. The system achieved 100% accuracy across all cases, confirming correct slab-logic implementation with no arithmetic errors introduced by the automated pipeline.

4.3 Usability Evaluation

Feedback was collected from 30 participants (15 administrative staff, 15 consumer representatives). Table II summarizes usability ratings on a 1–5 scale.

Criterion	Avg. Rating
Ease of Navigation	4.6
Clarity of Billing Information	4.5
Login and Security	4.7
Payment Process	4.4
Overall Satisfaction	4.6

TABLE II. Usability Evaluation Results (Scale 1–5).

4.4 Comparison with Existing Approaches

Table III compares the proposed system against a fully manual billing process and a representative web-based system [6].

Feature	Manual	Web	Proposed
Billing Accuracy	Low	High	High
Physical Visit Req.	Yes	No	No
Offline Operation	Yes	No	Yes
Role-based Access	No	Partial	Yes
Deployment Cost	Low	Medium	Low
Computation Speed	Slow	Fast	Fast

TABLE III. Comparison of Billing System Approaches.

4.5 Security Analysis

The dual-login architecture enforces strict role separation. Admin credentials are stored as salted SHA-256 hashes. Consumer passwords are similarly hashed and session state is cleared upon logout. All database interactions use parameterized JDBC prepared statements, mitigating SQL injection vulnerabilities [12].

5. Conclusion

This paper presented an Automated Electricity Billing System that addresses the inefficiencies and inaccuracies inherent in

traditional manual billing processes. By leveraging Java Swing for the graphical user interface and Microsoft SQL Server for backend data management, the system achieves a reliable, scalable, and user-friendly platform for both administrative personnel and consumers.

The dual-module architecture cleanly separates administrative functions from consumer-facing features, ensuring appropriate access control and operational clarity. Automated slab-based bill computation eliminates arithmetic errors, while the integrated payment workflow reduces the need for physical office visits. Evaluation results confirm 100% bill computation accuracy, sub-3-second batch processing for 500 records, and high user satisfaction across all measured usability dimensions.

Future work will explore the following enhancements:

- Integration with smart meter APIs for automatic consumption data collection.
- A companion Android/iOS mobile application for consumer access.
- Machine learning models for anomaly detection in consumption patterns.
- Migration to a cloud-hosted database for remote multi-branch access.
- Automated SMS and email bill alerts and payment reminders.

. References

[1] R. H. Bhise and R. S. Pande, "Intelligent electricity billing system using GSM technology," Int. J. Adv. Res. Electr., Electron. Instrum. Eng., vol. 3, no. 4, pp. 8435–8441, Apr. 2014.

[2] S. Tanwar, S. Tyagi, and N. Kumar, "Ethical, legal, and social implications of emerging technologies," IEEE Access, vol. 7, pp. 18524–18535, 2019.

[3] M. Amin and B. F. Wollenberg, "Toward a smart grid: Power delivery for the 21st century," IEEE Power Energy Mag., vol. 3, no. 5, pp. 34–41, 2005.

[4] P. P. Parikh, M. G. Kanabar, and T. S. Sidhu, "Opportunities and challenges of wireless communication technologies for smart grid applications," in Proc. IEEE Power Energy Soc. Gen. Meet., 2010, pp. 1–7.

[5] V. C. Gungor et al., "Smart grid technologies: Communication technologies and standards," IEEE Trans. Ind. Inform., vol. 7, no. 4, pp. 529–539, Nov. 2011.

[6] A. Kaur and R. Singh, "Design and implementation of an online electricity billing system," Int. J. Comput. Appl., vol. 74, no. 14, pp. 11–15, Jul. 2013.

[7] N. Patel, P. Shah, and R. Trivedi, "Web-based electricity billing management system using PHP and MySQL," Int. J. Eng. Res. Technol., vol. 2, no. 5, pp. 1102–1106, May 2013.

[8] V. C. Gungor et al., "A survey on smart grid potential applications and communication requirements," IEEE Trans. Ind. Inform., vol. 9, no. 1, pp. 28–42, Feb. 2013.

[9] S. Ahmed, M. H. Islam, and M. N. Islam, "IoT-based smart metering system for electricity billing and energy management," in Proc. IEEE IC4ME2, 2018, pp. 1–4.

[10] A. Kumar and R. Sharma, "A Java-based desktop application for utility billing with JDBC integration," Int. J. Sci. Res. Publ., vol. 5, no. 3, pp. 1–6, Mar. 2015.

[11] P. Verma, A. Agarwal, and M. Srivastava, "Comparative study of database management systems for billing applications," Int. J. Comput. Sci. Inf. Technol., vol. 6, no. 2, pp. 1456–1459, 2015.

[12] S. Mishra and A. Gupta, "Role-based access control in utility management portals," in Proc. IEEE InCITe, 2016, pp. 134–138.

[13] R. Sharma, V. Mehta, and S. Kaur, "Payment gateway integration in online billing systems: A review," Int. J. Adv. Comput. Res., vol. 4, no. 1, pp. 211–215, Mar. 2014.

[14] B. S. Rathore, P. Rawat, and D. Kumar, "Cloud-based electricity billing architecture for scalable urban deployments," in Proc. IEEE CCEM, 2015, pp. 1–5.

[15] M. A. Hossain, M. S. Islam, and M. R. Islam, "Mobile application for electricity bill management and payment," in Proc. IEEE ICEEICT, 2015, pp. 1–5.

[16] K. Garg and P. Gupta, "Android-based smart electricity metering system," Int. J. Innov. Res. Sci., Eng. Technol., vol. 4, no. 7, pp. 5487–5493, Jul. 2015.

[17] Y. Fayyaz, M. Farooq, and M. A. Shahid, "Automated meter reading and billing system using ZigBee and GPRS," in Proc. IEEE ICET, 2014, pp. 1–5.

[18] N. Kayalvizhi, D. Malathi, and R. Lavanya, "Design of advanced automatic meter reading system," IOSR J. Electr. Electron. Eng., vol. 9, no. 1, pp. 26–33, Jan.–Feb. 2014.

[19] A. O. Salau, G. Raglend, and J. P. Kumar, "A review on smart grid and smart metering," Int. J. Appl. Eng. Res., vol. 12, no. 24, pp. 15249–15258, 2017.

[20] T. Logenthiran, D. Srinivasan, and T. Z. Shun, "Demand side management in smart grid using heuristic optimization," IEEE Trans. Smart Grid, vol. 3, no. 3, pp. 1244–1252, Sep. 2012.