VISVESVARAYA TECHNOLOGICAL UNIVERSITY

JnanaSangama, Belagavi - 590018

Mini project
On

"ELECTRICITY BILL GENERATOR"

BY

KSHITISH	4MT21IC025
BHARATH P	4MT21IC009
AVISH KUMAR	4MT21IC008
SHRIMAN JAIN	4MT21IC053
DHANUSH S U	4MT21IC012

Department of CSE (IoT and Cyber Security with Blockchain Technology)

DEPARTMENT OF CSE (IoT and Cyber Security with Blockchain Technology)

(*Accredited by NBA*)

MANGALORE INSTITUTE OF TECHNOLOGY & ENGINEERING

Accredited by NAAC with A+ Grade, an ISO 9001: 2015 Certified Institution
(*A Unit of Rajalaxmi Education Trust*®, *Mangalore - 575001*)
Affiliated to VTU, Belagavi, Approved by AICTE, New Delhi
BadagaMijar, Moodabidri-574225, Karnataka

2022-23

ABSTRACT

This project aims at simplifying the process of generating electricity bills for residential and commercial users. This project focuses on core functionalities like customer data management, electricity consumption calculation, and bill generation. By automating these tasks, it demonstrates the practical application of programming skills in a real-world context, offering an efficient solution for managing billing operations. The Electricity Bill Generator project is designed to streamline and automate the billing process for utility companies. Its primary purpose is to enhance operational efficiency by accurately calculating and generating electricity bills. The project encompasses customer information management, billing calculations, and reporting, ensuring a comprehensive solution for utility companies. Key features include a customer database, tariff management, automated calculation, bill generation in various formats, robust security measures. This project aims to improve customer satisfaction, reduce manual effort, and enable scalable billing operations for utility providers.

INTRODUCTION

The "Electricity Bill Generator" project represents a practical application of software engineering principles and programming expertise. This project is designed to address the complexities and inefficiencies associated with manual electricity billing processes commonly encountered by utility companies. By leveraging the power of automation and accurate calculations, the project aims to streamline the billing operations, improve customer service, and enhance the overall efficiency of the utility company's billing department.

Background:

Traditional electricity billing methods often involve labor-intensive tasks, including manual meter readings, paper-based billing calculations, and data entry, which can lead to errors, delayed billing, and customer dissatisfaction. This project seeks to modernize and optimize these processes by digitizing data management, automating billing calculations.

Objective:

- Automated Billing Calculation:

- To automate the calculation of electricity charges based on meter readings and tariff rates.

- To eliminate manual errors in the calculation process, ensuring accuracy.

- Efficient Data Management:

- To create a secure and organized database for storing customer information.

- To centralize and manage customer data, including names, addresses, meter readings, and billing history.

- Bill Generation:

- To generate detailed electricity bills for each customer, clearly specifying the billing period, consumption charges, and due dates.

- To ensure that bills are professional, easy to understand, and compliant with industry standards.

- Tariff Rate Management:

- To provide a user-friendly interface for managing and updating tariff rates for different customer categories and billing periods.

- To ensure that tariff changes are reflected accurately in billing calculations.

- Data Security and Privacy:

- To implement robust security measures to safeguard sensitive customer information.

- To comply with data privacy regulations and protect customer confidentiality.

- User Access Control:

- To implement role-based access control to restrict access to sensitive functions and data.

- To ensure that only authorized personnel can perform specific actions within the system.

These detailed objectives collectively define the project's scope and goals, aiming to transform the electricity billing process into a more efficient, accurate, and customer-friendly operation. The project seeks to enhance utility company operations, customer satisfaction, and financial management through automation and modernized data management techniques.

TECHNOLOGIES USED

In this project several technologies, frameworks, libraries, and tools are used to develop a functional and user-friendly utility for managing electricity billing operations. Here's a brief description of some of the key components:

- **Standard C Libraries:**

  - The project utilizes standard C libraries such as `stdio.h`, `stdlib.h`, and `time.h` for essential functions like file handling, memory allocation, and time-related operations.

- **Windows API (windows.h):**

  - The `windows.h` header is used to interact with the Windows API. It is crucial for console-related operations, including setting the console title and size.

- **Custom Header Files:**

  -The project uses custom header files such as `struct.h`, `interface.h`, and `fileinput.h` to organize and modularize the code. These files contain custom functions and structures for managing linked lists, user interfaces, and file input operations.

- **Console Interface:**

  - The project relies on console-based interfaces for user interaction. Users navigate through menus and input data using standard input and output functionalities provided by the C language.

- **.File Handling:**

  - File handling functions from the C standard library are used to read data from files and store it in linked lists. This enables the project to manage customer information, tariff rates, and billing history efficiently.

- **User Input and Validation:**

  - Custom functions are developed for collecting user input, including usernames and passwords, while providing a secure and user-friendly interface.

- **Time Management (time.h):**

  - The `time.h` library is used to handle time-related operations, such as time stamping and displaying current date and time.

- **Dynamic Memory Allocation:**

- Dynamic memory allocation functions like `malloc` and `free` are used to manage memory efficiently, especially for dynamic data structures like linked lists.

- **Console Window Management:**

  - The Windows API is employed to set the console window's title and dimensions, ensuring a consistent and user-friendly interface.

- **Exit Confirmation:**

  - The project includes a feature to confirm user exits, allowing users to decide whether to return to the main menu or exit the application.

- **Custom Functions:**

  - Various custom functions are developed to implement project-specific features such as changing base tariff rates, managing billing deadlines, and navigating menus.

The primary development tool and environment:

When developing the "Electricity Bill Generator" project in VS Code, the primary tool for writing, editing, debugging, and managing our C code was Visual Studio Code itself. It provides a versatile and user-friendly environment for software development

- **Integrated Development Environment** (IDE):

Visual Studio Code (VS Code) itself serves as the primary development tool when working on C or C++ projects. It's a highly versatile and extensible code editor that supports multiple programming languages, including C and C++. VS Code provides features like code highlighting, debugging, version control integration, and extensions for enhanced development capabilities.

- <u>Here are some key aspects related to using VS Code for the project:</u>

- Code Editing: You'll write, edit, and manage your C code files within the VS Code editor. VS Code provides an excellent code editing experience with features like auto-completion, syntax highlighting, and code formatting.
- Integrated Terminal: VS Code includes an integrated terminal that allows you to compile and run your C programs directly within the editor.
- Extensions: VS Code supports extensions, and you can find extensions related to C/C++ development that can enhance your workflow. For example, you can use extensions like "C/C++" or "Code Runner" to facilitate C program execution.
- Version Control: VS Code has built-in support for Git, which is useful for version control and collaboration if you're working on the project with others.
- Task Runner: You can configure tasks in VS Code to automate common development tasks, such as building and running your C programs.

- Debugging: VS Code provides debugging capabilities for C and C++ programs, allowing you to set breakpoints, inspect variables, and step through code.
- IntelliSense: IntelliSense, a code completion and code suggestion feature, is available for C and C++ in VS Code, helping you write code more efficiently.
- Extension Marketplace: You can explore the VS Code Extension Marketplace to find additional extensions and tools that might be helpful for your specific project needs.

GitHub plays a crucial role in collaborative development and version control for your "Electricity Bill Generator" project. When used in conjunction with Visual Studio Code, it offers a powerful and streamlined development environment for your team

- **Gather:**

Gather is a web-based platform for version control and collaborative software development. It allows you to host your project's source code, track changes, collaborate with team members, and manage your project's development process. Here's how you can integrate Gather into your development workflow:

- Code Hosting: Gather serves as a remote repository for your project. You can push your code to Gather, making it accessible and shareable with collaborators.
- Version Control: Gather provides Git-based version control, allowing you to track changes to your project's codebase. You can commit changes, create branches, and merge code seamlessly.
- Collaboration: If you are working on the project with a team, Gather facilitates collaborative development. Team members can clone the repository, make changes, and create pull requests for code review and integration.
- Issue Tracking: Gather includes an issue tracking system that helps you manage tasks, bug reports, and feature requests related to your project. You can create and assign issues to team members, track progress, and maintain a structured development process.
- Pull Requests: When team members want to contribute changes to the project, they can do so by creating pull requests. Pull requests provide a way to review, discuss, and merge code changes into the main branch.
- Branch Management: Gather allows you to work with feature branches, bug-fix branches, and other development branches. You can manage branches, switch between them, and merge them as needed.
- Continuous Integration (CI): You can set up CI/CD (Continuous Integration/Continuous Deployment) pipelines using services like Gather Actions. CI pipelines can automate code testing, building, and deployment processes.

- **Visual Studio Code Integration with Gather:**

Visual Studio Code offers excellent integration with Gather, making it seamless to work with Git and Gather from within the code editor:

- Git Extension: VS Code includes a Git extension that provides a graphical interface for Git commands. You can commit, push, pull, and manage branches without leaving the editor.
- Gather Integration: You can connect your Gather account to VS Code, enabling you to clone repositories, create new repositories, and manage Gather-related tasks directly from the editor.
- Pull Requests: VS Code supports reviewing and managing pull requests through extensions, making it easy to collaborate on Gather-hosted projects.

Overall, the "Electricity Bill Generator" project combines the power of the C programming language, standard libraries, and custom functions to create a functional utility that automates and streamlines the electricity billing process. It offers an efficient and user-friendly solution for utility companies.

DESIGN AND IMPLEMENTATION

Design Overview :

The project's design aims for modularity, data structure optimization, and user-friendly interaction. Key design components include

- **Customer Data Structure :**

A custom structure (`struct`) is designed to encapsulate customer information. It includes fields such as:

- Customer Name

- Address

- Account Number

- Previous Meter Reading

- Current Meter Reading

- Tariff Rates (Per Unit Cost, Fixed Charges)

- Units Consumed

- Bill Amount

This structured approach ensures organized data storage and retrieval.

- **Linked List :**

Customer records are managed using a linked list data structure. Each node in the linked list represents a customer and contains the customer data structure. This dynamic organization allows efficient addition, deletion, and traversal of customer records.

- **User Interface (CLI) :**

A user-friendly command-line interface (CLI) guides users through the process of inputting customer data, performing calculations, and viewing bills. It ensures accessibility for both utility company staff and consumers.

Implementation Overview:

The project is implemented in the C programming language, chosen for its efficiency and portability. Key implementation details include:

- **Data Validation**

  Robust data validation mechanisms are implemented to ensure data accuracy and integrity. Validation checks include verifying valid meter readings (within a valid range), non-negative values, and appropriate data types. Informative error messages are provided to assist users in case of incorrect inputs.

- **Error Handling**

  Effective error-handling mechanisms are incorporated to provide informative error messages for incorrect inputs or exceptional situations, thereby enhancing the user experience. This includes handling issues such as invalid data types, negative values, and inconsistencies.

- **Calculation Module**

  The core module of the system performs precise calculations, including the determination of units consumed and the calculation of the bill amount. This is accomplished using the formula: `Bill Amount = (Units Consumed * Per Unit Cost) + Fixed Charges`, based on user-provided tariff rates.

- **Output and Reporting**

  The system generates detailed electricity bills for each customer, displaying customer information, meter readings, tariff details, units consumed, and the final bill amount. Optionally, the project offers the functionality to save bill details to files, which can be beneficial for record-keeping, auditing, and customer service purposes.

- **Scalability and Maintainability**

  The code base is modular, well-documented, and designed for scalability. This design consideration ensures that the project can accommodate a growing number of customers and supports future updates and enhancements, maintaining its relevance and efficiency.

FEATURES AND FUNCTIONALITY

- **User-Friendly Interface**

  The project offers a user-friendly command-line interface (CLI) that guides users through the billing process, making it accessible to both utility company employees and consumers.

- **Customer Information Management**

  The system allows for efficient management of customer data, including inputting customer details, meter readings, and tariff rates.

- **Accurate Calculation**

  The project calculates units consumed and bill amounts accurately, ensuring transparent and precise billing.

- **Data Validation and Error Handling**

  Robust data validation and error-handling mechanisms improve data accuracy and user experience by providing clear and informative error messages.

- **Display and Reporting**

  The system generates detailed electricity bills, including customer information and bill amounts, making billing transparent and accessible.

- **Optional Data Logging**

  Users can optionally save bill details to files for record-keeping, facilitating auditing and customer service.

Functionality

- **Data Input and Validation**

  Users input customer information, meter readings, and tariff rates. Robust data validation checks ensure data accuracy and provide error feedback.

- **Calculation**

  The system accurately calculates units consumed and bill amounts based on user-provided data.

- **Error Handling**

Effective error-handling mechanisms provide clear and informative error messages for incorrect inputs, guiding users in resolving issues.

- **Display and Reporting**

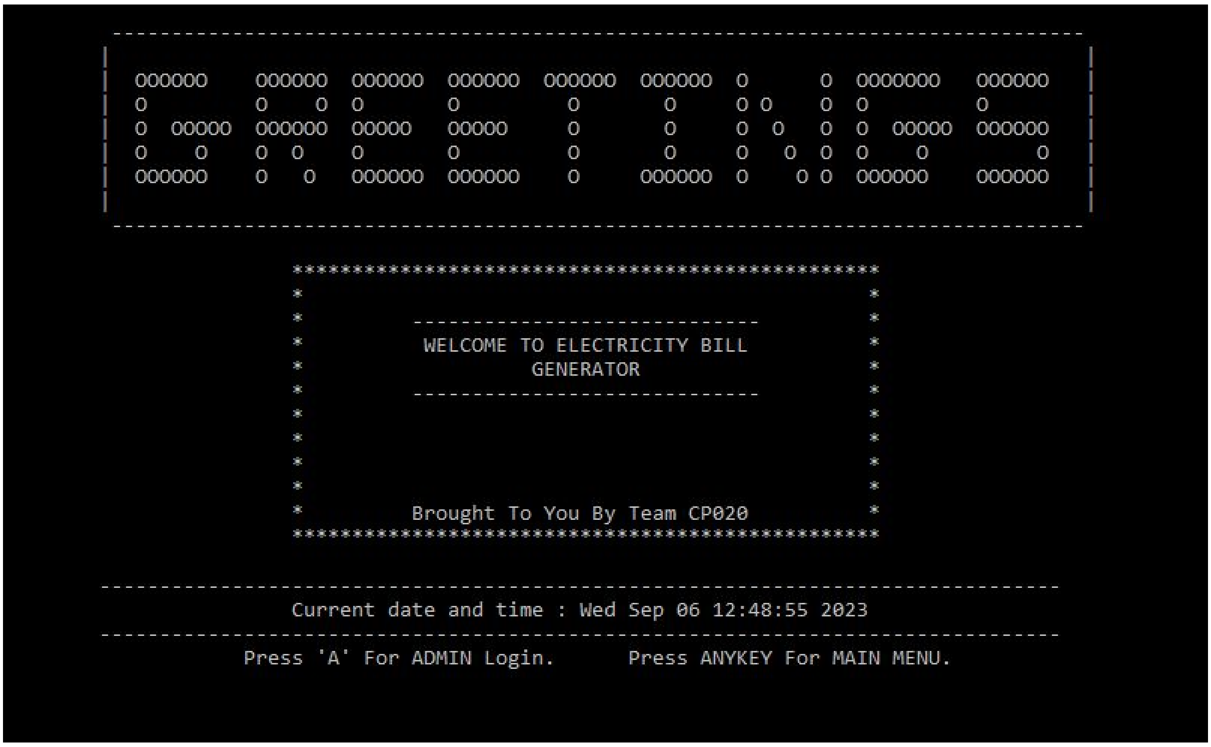
  The project displays detailed electricity bills to users, enhancing transparency and user experience.

- **Optional Data Logging**

  Users can choose to save bill details to files for record-keeping, facilitating auditing and customer service.

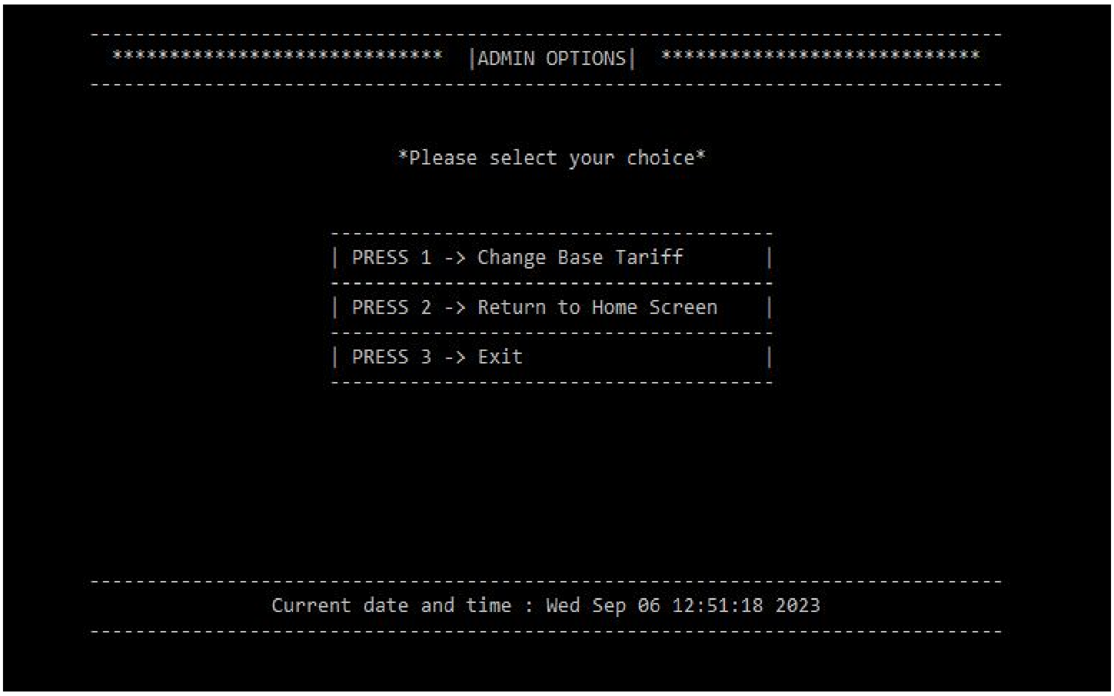
RESULT

- HOME PAGE



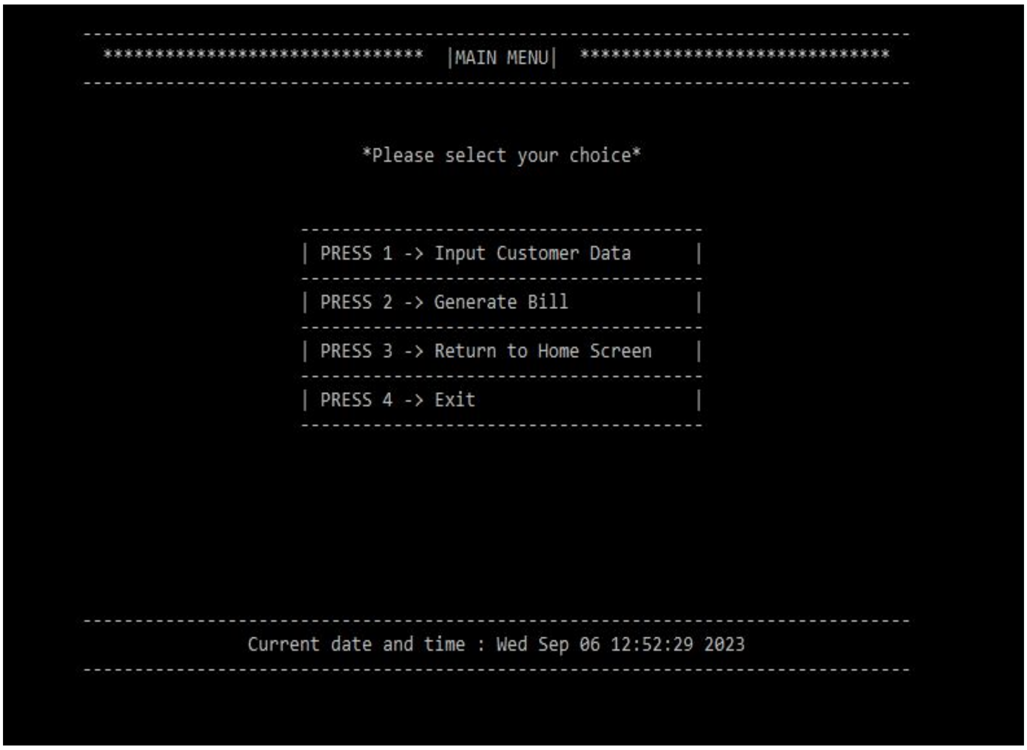
```
        --------------------------------------------------------------------------------
       |                                                                                |
       |  000000    000000  000000  000000  000000  000000  0      0  0000000   000000   |
       |  0         0    0  0       0         0       0     0 0    0  0         0        |
       |  0  00000  000000  00000   00000     0       0     0  0   0  0  00000  000000   |
       |  0    0    0  0    0       0         0       0     0   0  0  0    0         0   |
       |  000000    0   0   000000  000000    0     000000  0    0 0  000000    000000   |
       |                                                                                |
        --------------------------------------------------------------------------------

                        **************************************************
                        *                                                *
                        *          ----------------------------          *
                        *           WELCOME TO ELECTRICITY BILL          *
                        *                    GENERATOR                   *
                        *          ----------------------------          *
                        *                                                *
                        *                                                *
                        *                                                *
                        *                                                *
                        *          Brought To You By Team CP020          *
                        **************************************************

       --------------------------------------------------------------------------------
                       Current date and time : Wed Sep 06 12:48:55 2023
       --------------------------------------------------------------------------------
                   Press 'A' For ADMIN Login.      Press ANYKEY For MAIN MENU.
```

- ADMIN PAGE



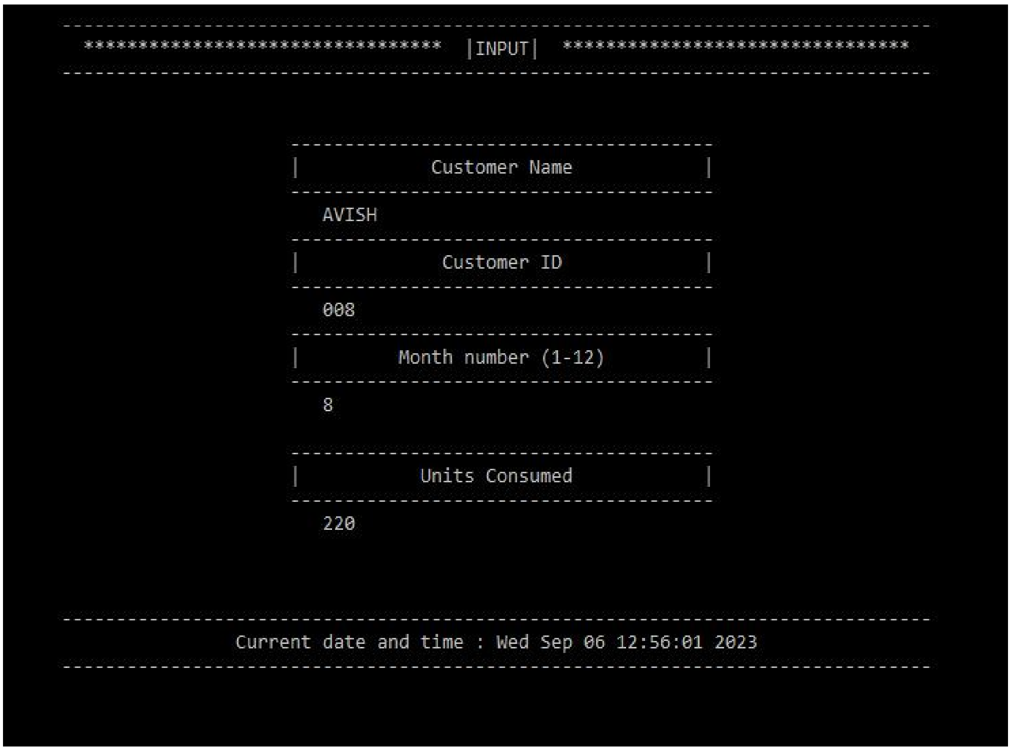
```
       ----------------------------------------------------------------------------
         ****************************  |ADMIN OPTIONS|  ***************************
       ----------------------------------------------------------------------------


                                   *Please select your choice*


                            -------------------------------------
                            | PRESS 1 -> Change Base Tariff       |
                            -------------------------------------
                            | PRESS 2 -> Return to Home Screen    |
                            -------------------------------------
                            | PRESS 3 -> Exit                     |
                            -------------------------------------





       ----------------------------------------------------------------------------
                         Current date and time : Wed Sep 06 12:51:18 2023
       ----------------------------------------------------------------------------
```

- USER PAGE



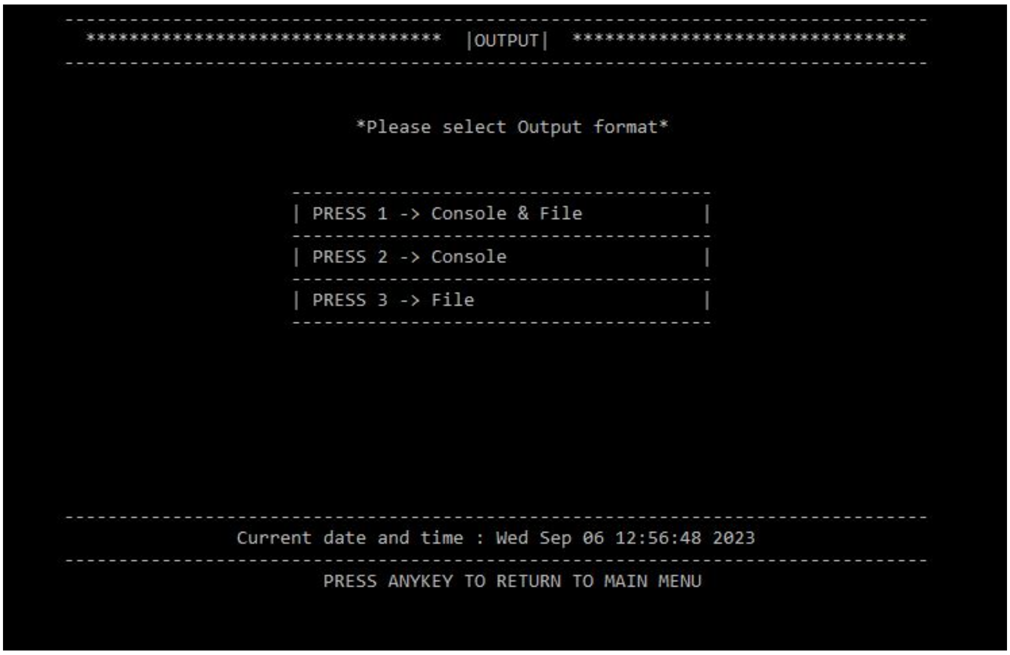
```
--------------------------------------------------------------------------------
  ******************************  |MAIN MENU|  ******************************
--------------------------------------------------------------------------------


                        *Please select your choice*


                   --------------------------------------
                   | PRESS 1 -> Input Customer Data     |
                   --------------------------------------
                   | PRESS 2 -> Generate Bill           |
                   --------------------------------------
                   | PRESS 3 -> Return to Home Screen   |
                   --------------------------------------
                   | PRESS 4 -> Exit                    |
                   --------------------------------------






--------------------------------------------------------------------------------
               Current date and time : Wed Sep 06 12:52:29 2023
--------------------------------------------------------------------------------
```

- INPUT PAGE

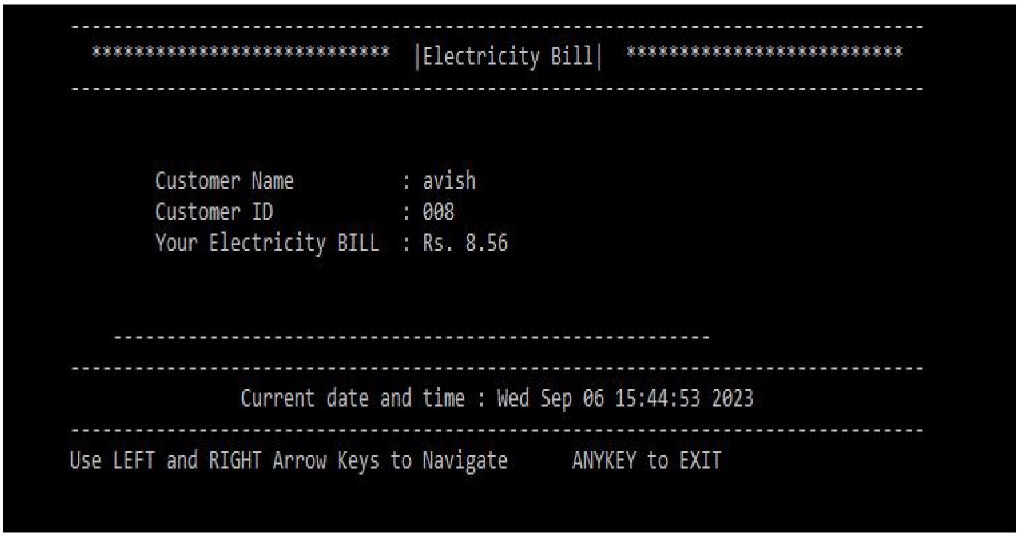


```
--------------------------------------------------------------------------------
  *********************************  |INPUT|  *********************************
--------------------------------------------------------------------------------


                     ---------------------------------------
                     |            Customer Name            |
                     ---------------------------------------
                        AVISH
                     ---------------------------------------
                     |             Customer ID             |
                     ---------------------------------------
                        008
                     ---------------------------------------
                     |         Month number (1-12)         |
                     ---------------------------------------
                        8

                     ---------------------------------------
                     |           Units Consumed            |
                     ---------------------------------------
                        220



--------------------------------------------------------------------------------
                Current date and time : Wed Sep 06 12:56:01 2023
--------------------------------------------------------------------------------
```

- GENERATE BILL PAGE



```
--------------------------------------------------------------------------------
  *********************************   |OUTPUT|   ******************************
--------------------------------------------------------------------------------


                          *Please select Output format*


                      ---------------------------------------
                      | PRESS 1 -> Console & File           |
                      ---------------------------------------
                      | PRESS 2 -> Console                  |
                      ---------------------------------------
                      | PRESS 3 -> File                     |
                      ---------------------------------------







--------------------------------------------------------------------------------
                Current date and time : Wed Sep 06 12:56:48 2023
--------------------------------------------------------------------------------
                        PRESS ANYKEY TO RETURN TO MAIN MENU
```



```
--------------------------------------------------------------------------------
  ****************************   |Electricity Bill|   **************************
--------------------------------------------------------------------------------


      Customer Name          : avish
      Customer ID            : 008
      Your Electricity BILL  : Rs. 8.56



    --------------------------------------------------------
--------------------------------------------------------------------------------
                Current date and time : Wed Sep 06 15:44:53 2023
--------------------------------------------------------------------------------
Use LEFT and RIGHT Arrow Keys to Navigate      ANYKEY to EXIT
```

CONCLUSION

To sum it up, the Electricity Bill Generator is a valuable tool that makes electricity billing easier and more accurate for a retailer or a company. It helps utility companies and customers by automating complex implementation and providing an easy-to-use interface. In the future, we can make it even better with features like user-friendly screens, more detailed information, and mobile apps. We're committed to improving this project and making it even more helpful for everyone involved.

This project shows our commitment to making things better in the world of electricity billing, and we're excited to keep working on it.