A logo for a university

Description automatically generated

**AutoZen Car Management System**

Project Report

(Desktop Software Application)

GHHSM Samarasinghe

CE208.3-Software Development Project

32504

**Table of Contents**

**1. Introduction**

1.1 The Problem to Be Solved

1.2 Assumptions

1.3 Limitations of the Scope

**2. Similar Systems and Novelty of the System**

2.1 Similar Systems

2.2 Novelty of AutoZen

**3. System Requirement And Analysis**

3.1 Functional Requirements

3.2 Non-Functional Requirements

**4. Design of the System**

4.1 Class Diagram

4.2 ER Diagram

4.3 Use Case Diagram

4.4 Sequence Diagram

**5. Testing and Verification**

5.1 Functional Testing

5.2 Validation Testing

5.3 Integration Testing

**6. Conclusion**

**1. Introduction**

**1.1 The Problem to Be Solved**

In today’s competitive automobile dealership industry, managing vehicle records, tracking expenses, and calculating profit/loss manually or with basic spreadsheets often leads to inefficiencies, data loss, and lack of analytical insight. Small and medium car dealership businesses like ours face difficulties in organizing vital information such as purchase details, selling records, maintenance costs, and operating expenses in a centralized and accessible format.

Currently, dealership owners or managers must track these elements using separate tools or paper-based methods, which are time-consuming, error-prone, and do not provide a clear view of overall business performance. There is a growing need for a comprehensive, user-friendly desktop application that can streamline these operations and provide meaningful insights into profitability and cost control.

The **AutoZen** software aims to solve this problem by offering an all-in-one car dealership management system. It allows users to:

* Record vehicle details including purchase and selling information.
* Record utility details including water bill,light bill,rent fee and etc.
* Track all related expenses (e.g., tax, transport, repair).
* Calculate profits per vehicle.

**1.2 Assumptions**

To ensure smooth development and functionality of the AutoZen system, the following assumptions are made:

* The application is used by small or mid-sized dealerships that handle a manageable number of cars monthly.
* User have basic computer knowledge.
* Users are familiar with basic desktop software operations (clicking buttons, entering data in forms).
* All records are managed on a single-user system (multi-user or network-based access is not required for this version).
* The database system (MySQL) will be set up locally on the machine running the software.
* Input data provided by the user is assumed to be correct and valid (e.g., numeric values for prices).

**1.3 Limitations of the Scope**

While AutoZen offers many essential features, the current version has a few limitations:

* **Limited user access:** The system does not support multi-user roles(only admin and employee).
* **Limited reporting:** The reports are basic and do not include advanced analytics or graphs.
* **Manual data backup:** Users must manually back up the database; auto-backup features are not implemented.
* **No integration with external systems:** It does not connect with online payment systems, accounting tools.

**2. Similar Systems and Novelty of the System**

**2.1 Similar Systems**

There are several commercial vehicle management and dealership software solutions available in the market that provide features like inventory management, billing, expense tracking, and reporting. Some notable examples include:

* **Dealertrack DMS**  
  A cloud-based dealership management system that supports inventory, customer management, and accounting.
* **AutoManager DeskManager**  
  A powerful desktop application for managing vehicle sales, expenses, inventory, and customer information. It supports reporting and document generation.
* **Autosoft DMS**  
  An end-to-end system used by automotive retailers for accounting, sales, inventory, and fixed operations. It is mainly suited for medium to large businesses.
* **CarDealerTracker**   
  A lightweight solution for small dealerships to manage stock, customers, and basic reporting features. However, it lacks deep customization and is limited in usability for localized needs.

**2.2 Novelty of AutoZen**

The AutoZen software introduces a customized solution specifically tailored for small and medium-scale car dealerships with the following novel features:

* Simplified Profit Calculation:  
  Unlike generic accounting tools, AutoZen provides a built-in formula to calculate profit per car and overall business profit/loss, considering both purchase and operational expenses.
* User-Focused Design:  
  Designed with simplicity in mind for users who may not have technical expertise. It focuses only on essential features rather than overwhelming users with complex modules.
* Offline and Free to Use:  
  No internet or subscription is required. The system is built as a desktop application with local database access, making it ideal for budget-constrained environments.

**3. System Requirement And Analysis**

**3.1 Functional Requirements**

**Vehicle Management**

* Add new vehicle details (model, year, purchase date, condition, etc.)
* Record transport fees, tax fees, repair cost, buying cost, and selling price
* Update or delete vehicle records
* Automatically calculate individual car profit:

Car Profit = Selling Price - (Buying Price + Tax Fee + Repair Cost)

**Expense Management**

* Enter and store monthly business expenses:
  + Rent
  + Water bill
  + Light bill
  + Other expenses
* Calculate total monthly expenses:

Total Cost = Rent + Water Bill + Light Bill + Other Expenses

**Profit and Loss Summary**

* View total profit/loss for a selected time range:

Total Profit/Loss = Car Profit - Total Cost

**Basic User Interaction**

* Simple form-based UI for entering and viewing data
* Use buttons for actions like Add, Save, Update, Delete, and Generate Invoice

**3.2 Non-Functional Requirements**

**Usability**

* + Easy-to-use graphical interface (built with Java Swing)
  + Designed for non-technical users

**Performance**

* + Quick data processing for calculations and report generation
  + Invoice creation within seconds

**Portability**

* + Runs on Windows-based desktop/laptop systems
  + Does not require high-end hardware

**Security**

* + Data stored locally in MySQL with limited user access
  + Only the system user can modify records (no login for now)

**Reliability**

* + Application should function smoothly without frequent crashes
  + Handles basic exception scenarios (e.g., missing input, incorrect data types)

**Offline Capability**

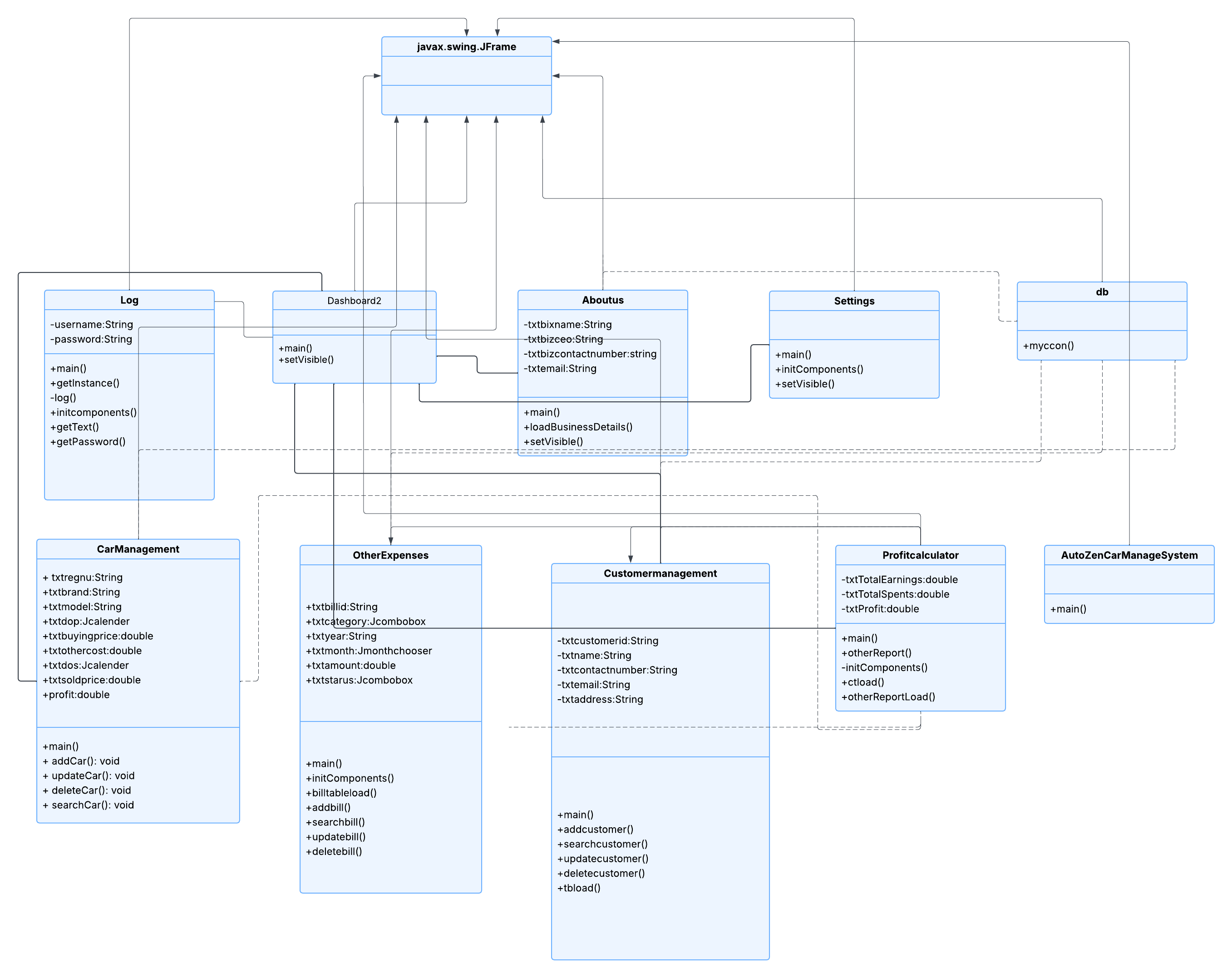
* + Entire system functions without internet connectivity

**Maintainability**

* + Well-structured codebase allows future updates and feature enhancements
  + Separate classes for car data, expenses, invoice, and profit logic

**3. Design of the System**

**3.1 Class Diagram**



**3.2 ER Diagram**

**A screenshot of a computer

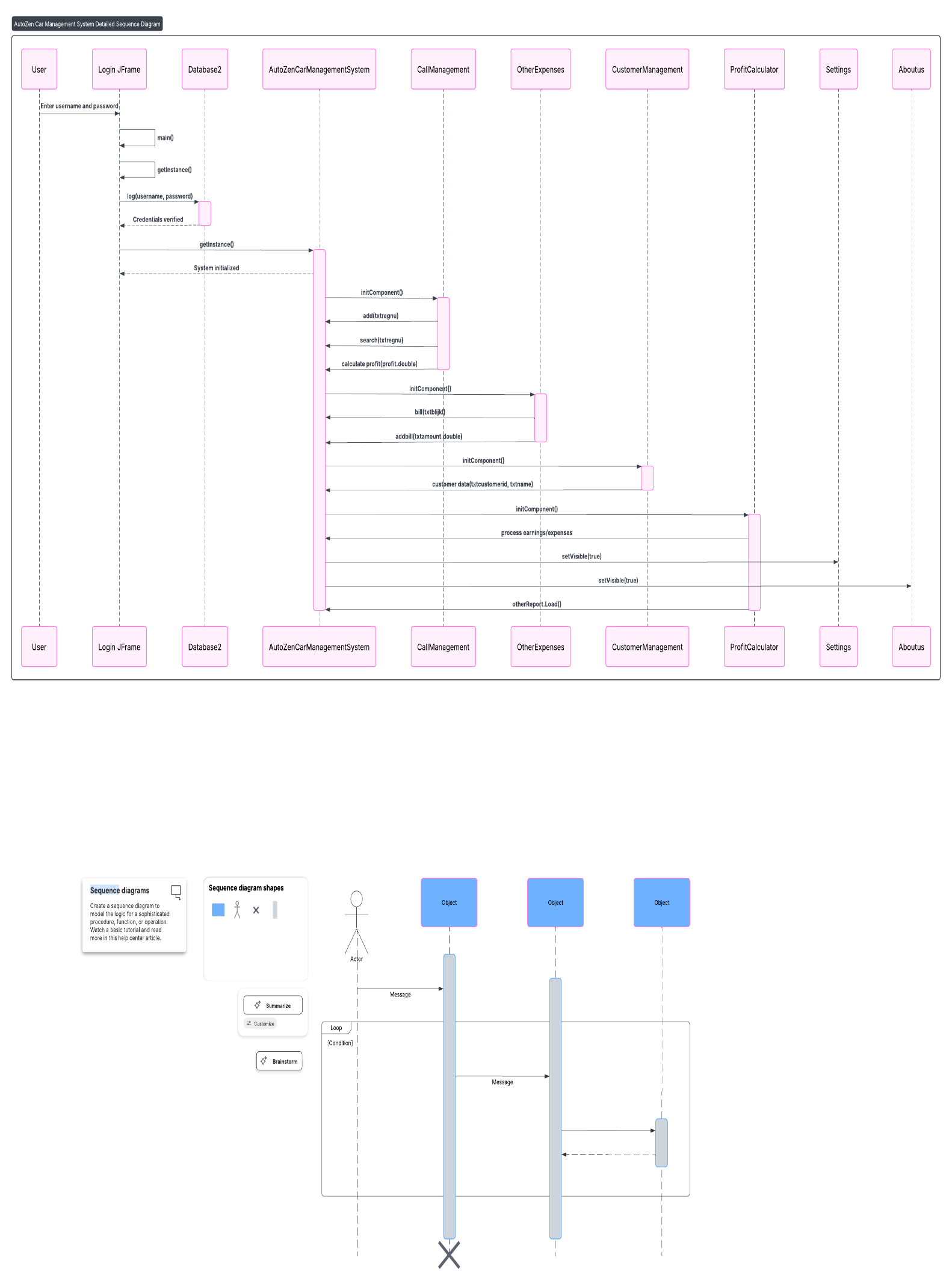
Description automatically generated**

**3.3 Use Case Diagram**

A diagram of a person with lines and dots

Description automatically generated

**3.4 Sequence Diagram**

****

**5. Testing and Verification**

To verify system correctness, a combination of manual functional testing, interface testing, and result validation was used. Each feature was tested with valid and invalid inputs to ensure that the system responded correctly and handled exceptions gracefully.

The following testing types were conducted,

* Functional Testing
* Validation Testing
* Integration Testing

**5.1 Functional Testing**

This section includes screenshots that demonstrate the main functionalities of the AutoZen application operating as expected. Functional testing was carried out to verify that each key module performs its intended operations correctly when provided with valid input data.

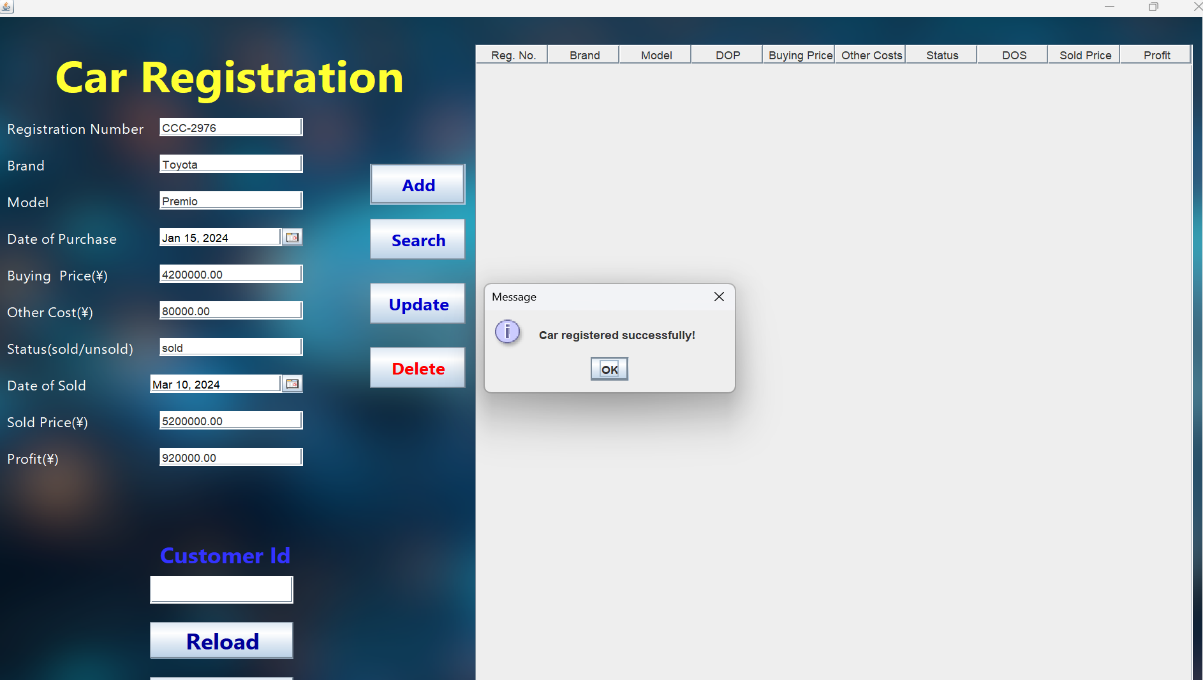
The features tested include adding car details, entering various types of business expenses, and calculating profit or loss for a selected time period. For car management, the system successfully saved vehicle data such as model, purchase price, tax fees, and selling price. These entries were properly displayed in the car table and stored in the MySQL database.

The expense module was tested by entering rent, utility bills, and other expenses. These were accurately recorded and later factored into the profit calculation. In the profit calculator, users selected a specific month, and the system correctly retrieved car sales and expenses, applied the formula, and displayed the net profit.

All modules responded as expected, and no errors were encountered during testing. The consistency of output between the manual and system-based calculations confirmed the accuracy and reliability of the AutoZen application’s core functionality.

* **Car Management Verification:-**

To demonstrate that the “Add Car” functionality in the AutoZen system works correctly, a real-world scenario was used involving a Toyota Premio vehicle. The car Reg number CCC-2976 was purchased on 15th January 2024 for Rs. 4,200,000, with additional costs including a tax fee of Rs. 250,000. The vehicle was later sold on 10th March 2024 for Rs. 5,200,000.You can see the the functions according to this scenarios as follows,



A screenshot of a computer

Description automatically generated

**Record added successfully to the system**

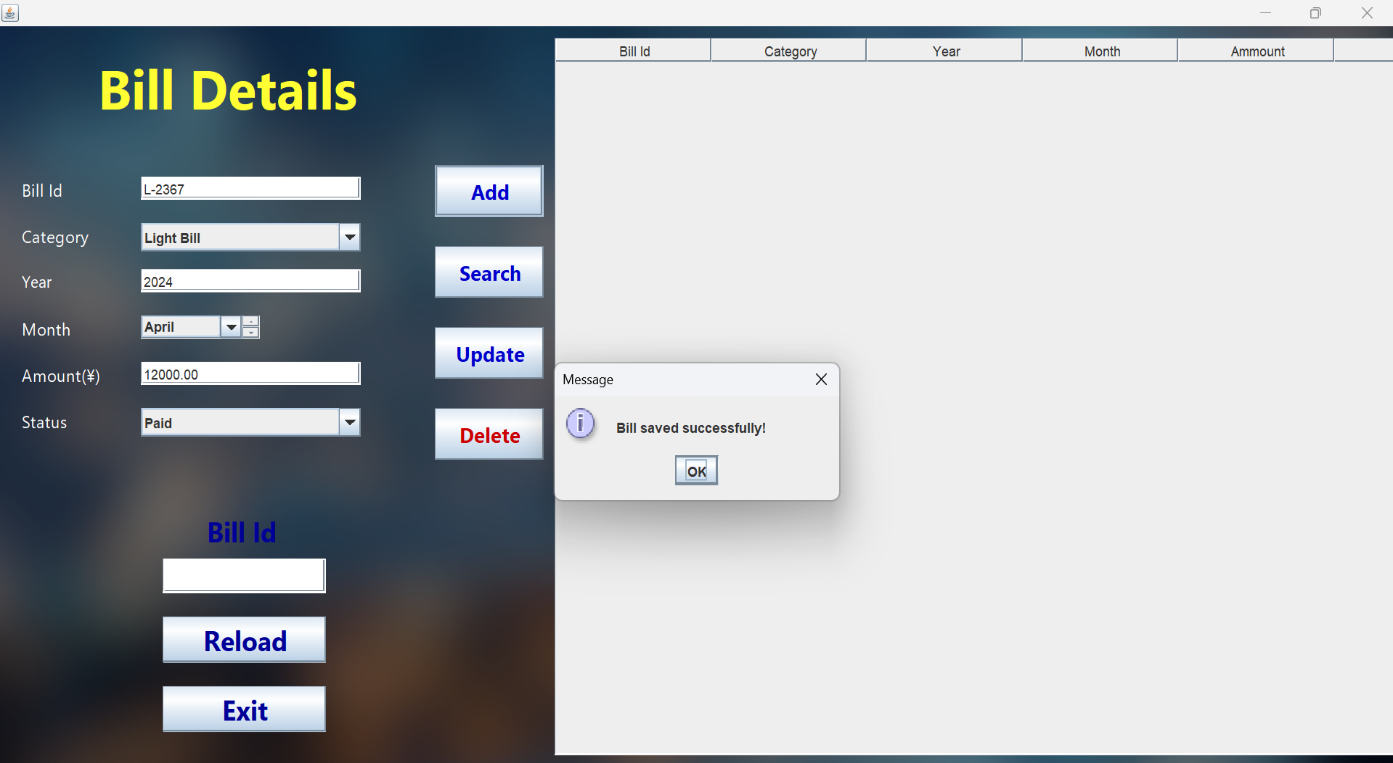
A screenshot of a computer

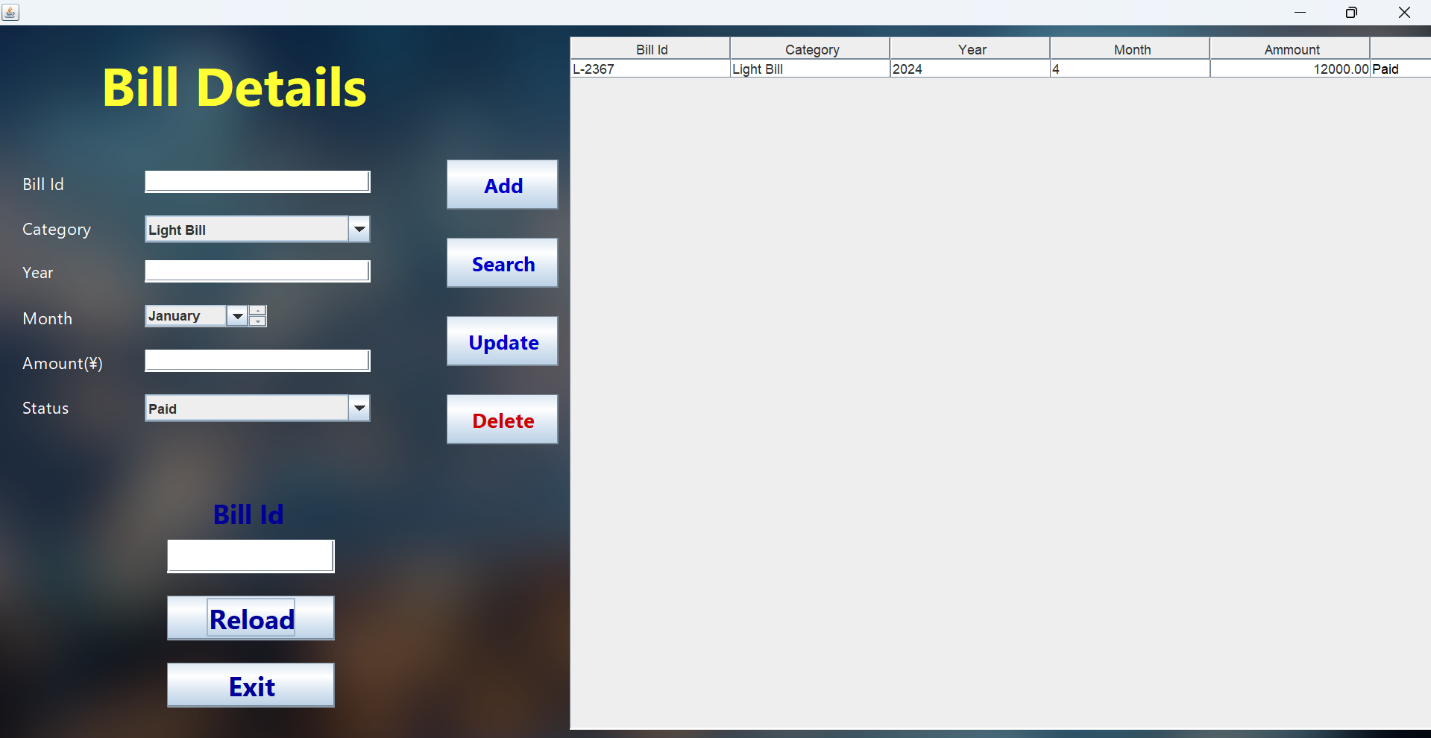
Description automatically generated

**Record also added successfully to the database**

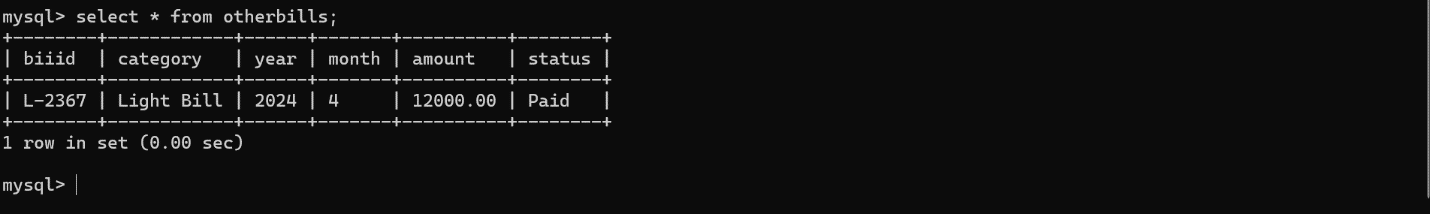
* **Other Expenses Management Verification:-**

To demonstrate that the “Other Expenses” functionality in the AutoZen system works correctly, a real-world scenario was used involving a utility payment. On April 2024, the dealership paid the monthly light bill(bill id L-2367) of Rs. 12,000.





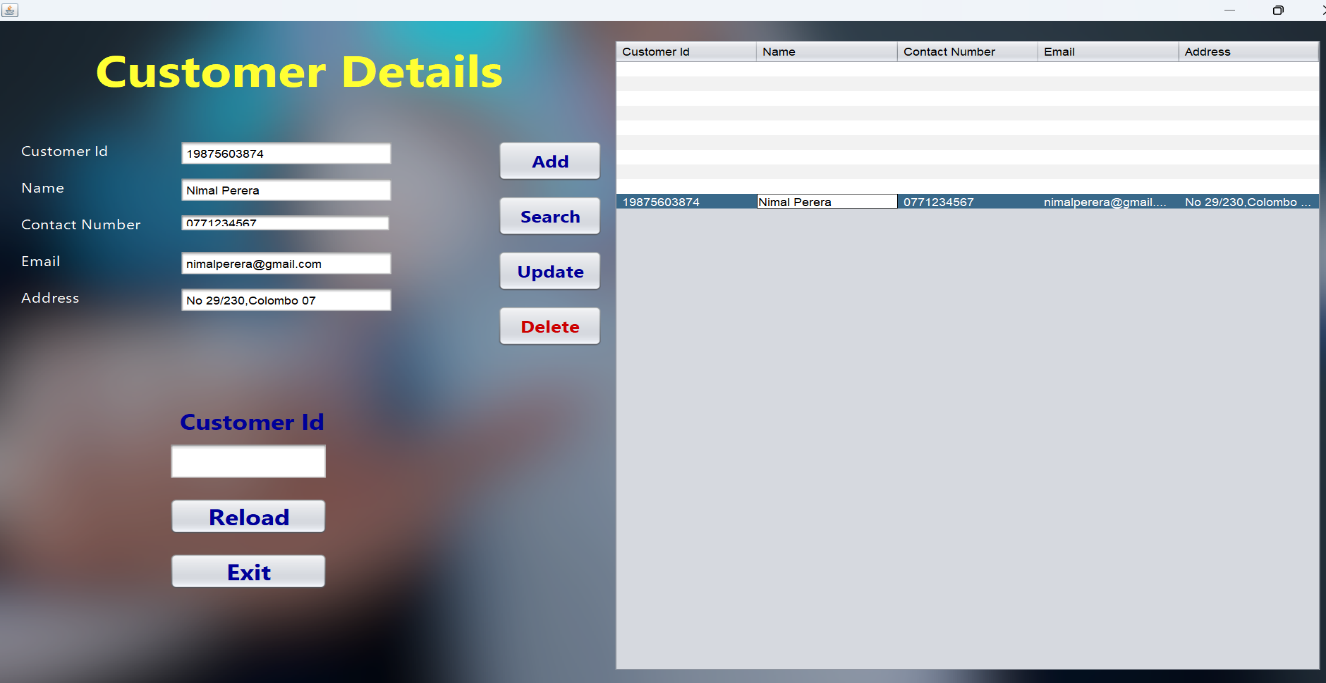
**Record added successfully to the system**



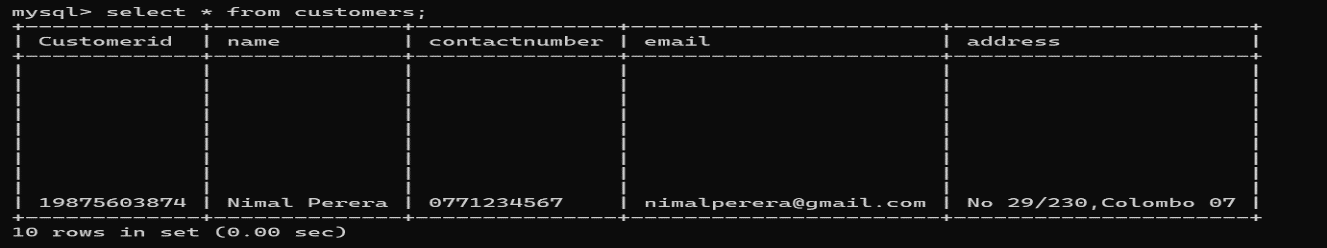
**Record also added successfully to the database**

* **Customer Management Verification:-**

To demonstrate that the “Customer Management” functionality in the AutoZen system works correctly, a real-world scenario was used involving a customer named Nimal Perera. On 10th March 2024, Nimal visited the dealership and provided his personal details, including his name, age (35), email address (nimal.perera@gmail.com), and telephone number (0771234567). These details were added to the system using the Customer Management form. The system successfully saved the information, making it easily accessible for future communication, sales tracking, and customer service. This verifies that the feature is working correctly and reliably stores essential customer data.



**Record added successfully to the system**



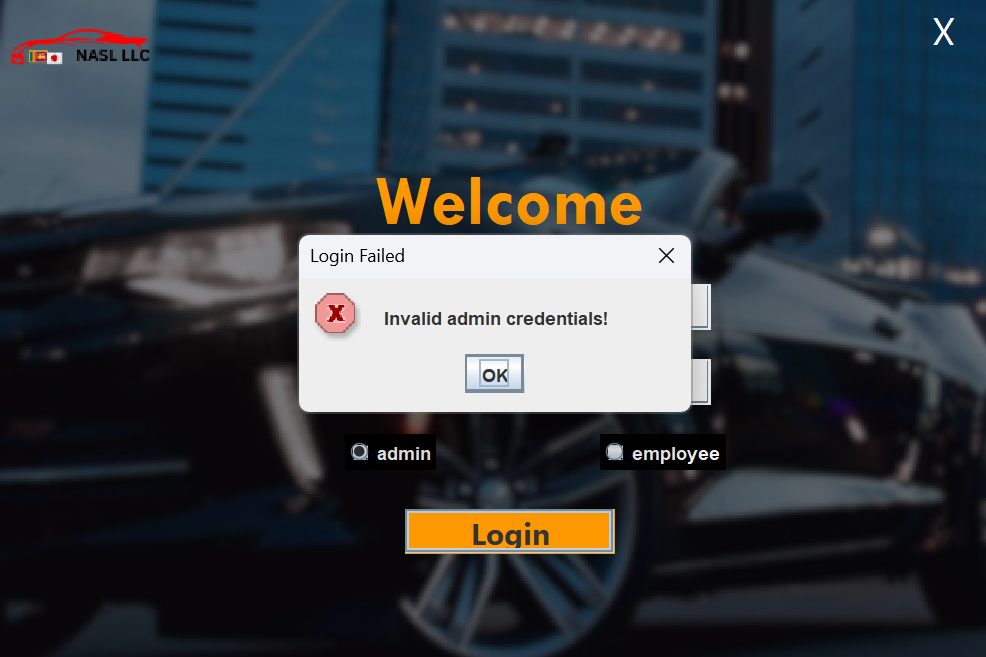
**Record also added successfully to the database**

**5.2 Validation Testing**

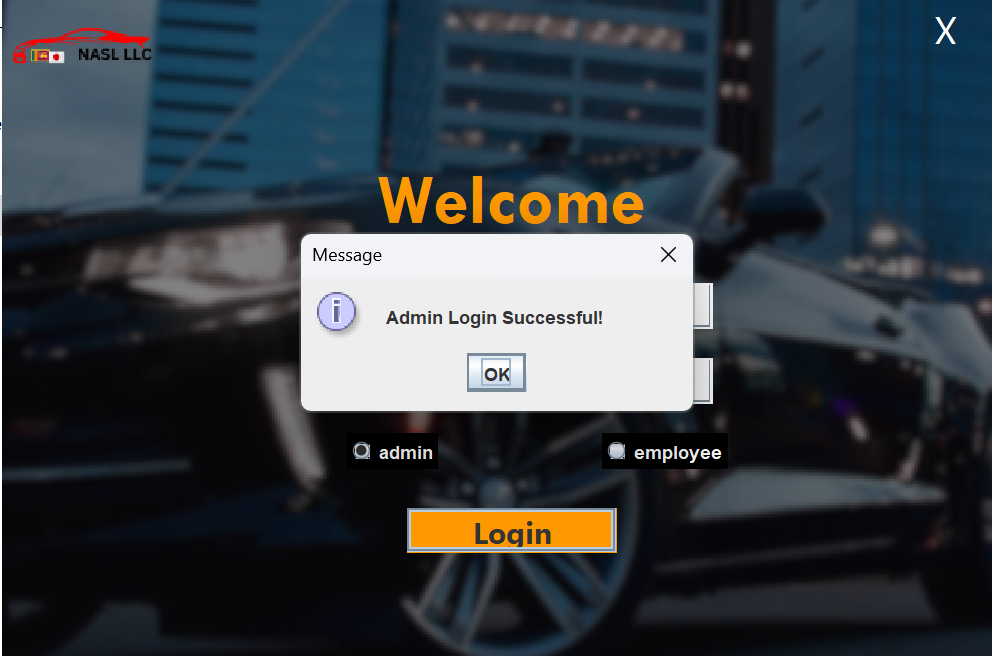
Validation testing ensures that the AutoZen application correctly handles invalid, missing, or incorrect user input. It helps verify that the system does not accept incomplete or faulty data and provides appropriate feedback to the user through error messages or dialog boxes.

* **Loging Validation**

Login validation testing ensures that only valid users can access the system and that meaningful errors are shown when incorrect or missing credentials are entered.

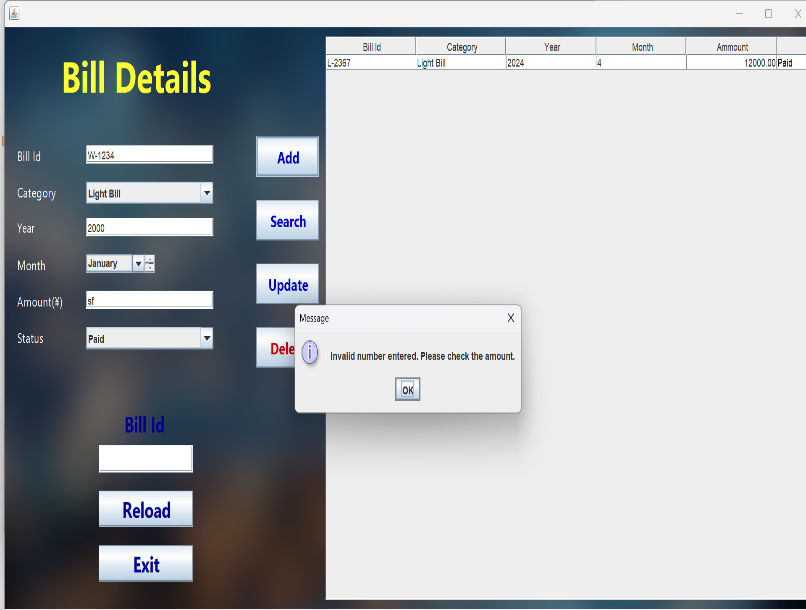
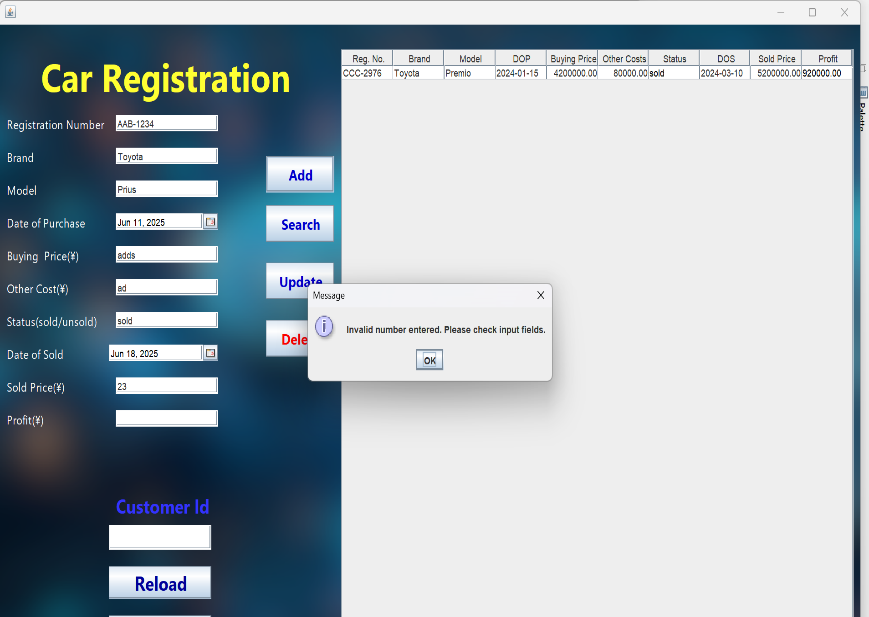
****

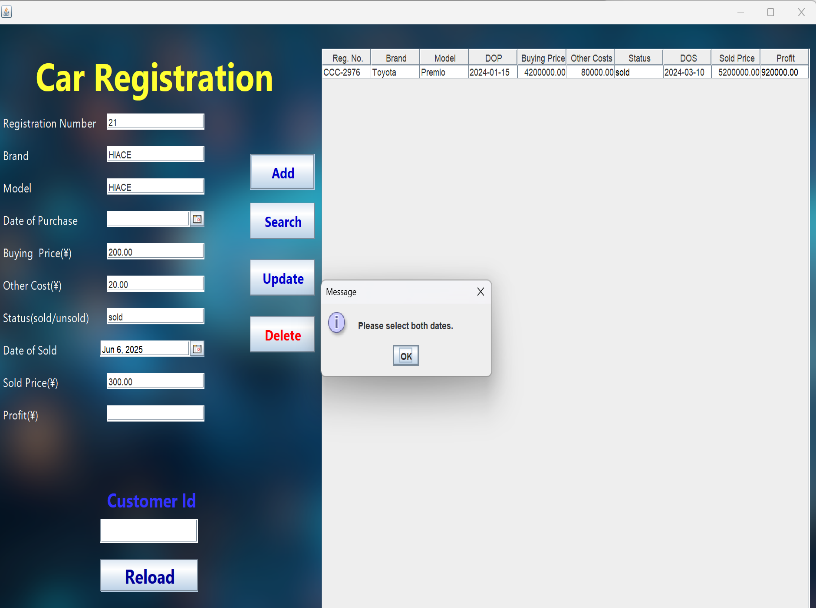
**Unsuccessful Login(Incorrect username/password or do not fill both fields)**

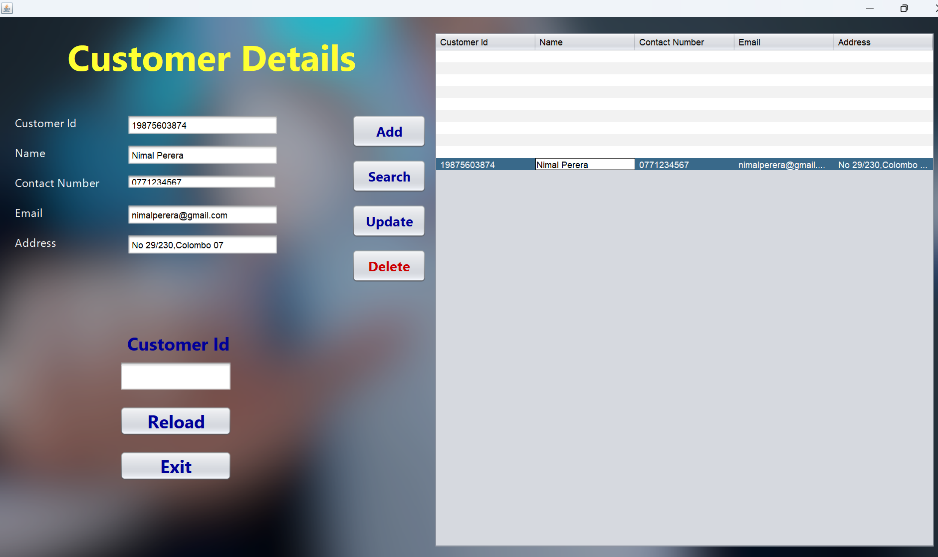
****

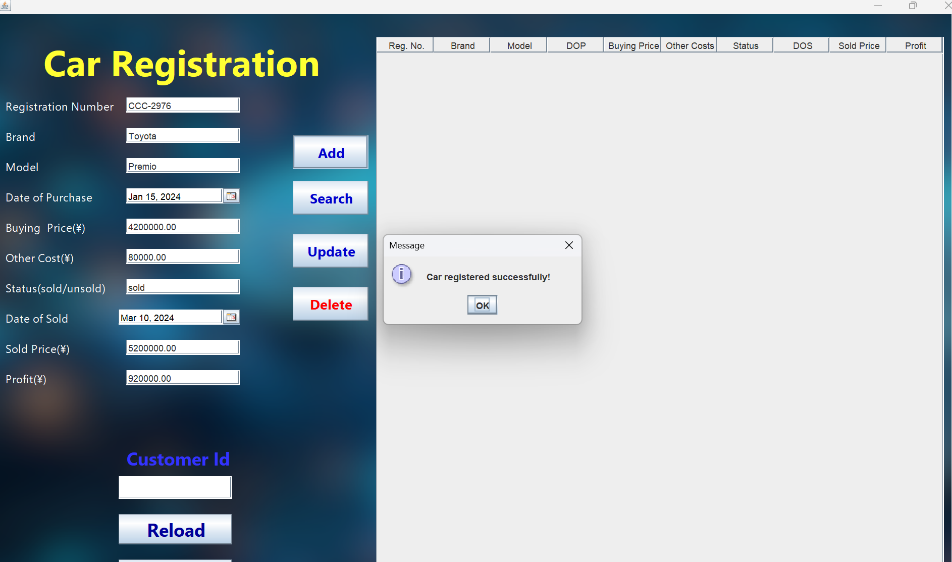
**Successful Login**

* **Management Validation**

****A screenshot of a computer

Description automatically generatedManagement validation testing ensures that all data entered in car, customer, and expense management forms is complete and valid. It checks for missing fields, incorrect data types, and prevents saving invalid information to maintain data accuracy and system reliability.





Valid Records

**5.3 Integration Testing**

Integration testing focuses on verifying that the different modules of the AutoZen system work together as intended. While individual features such as car management, expense tracking, and profit calculation were tested separately, integration testing ensures that they function correctly when combined into complete workflows. This type of testing helps identify issues that may occur when data flows between modules, such as calculating profit based on both car sales and expenses. It ensures that the overall system operates smoothly and delivers consistent results across all connected components.

As a example, the car dealer conducted 3 vehicle sales and paid 4 types of business expenses in December 2024. The system must calculate the total profit or loss for the month by subtracting all expenses from the total profit earned from car deals.

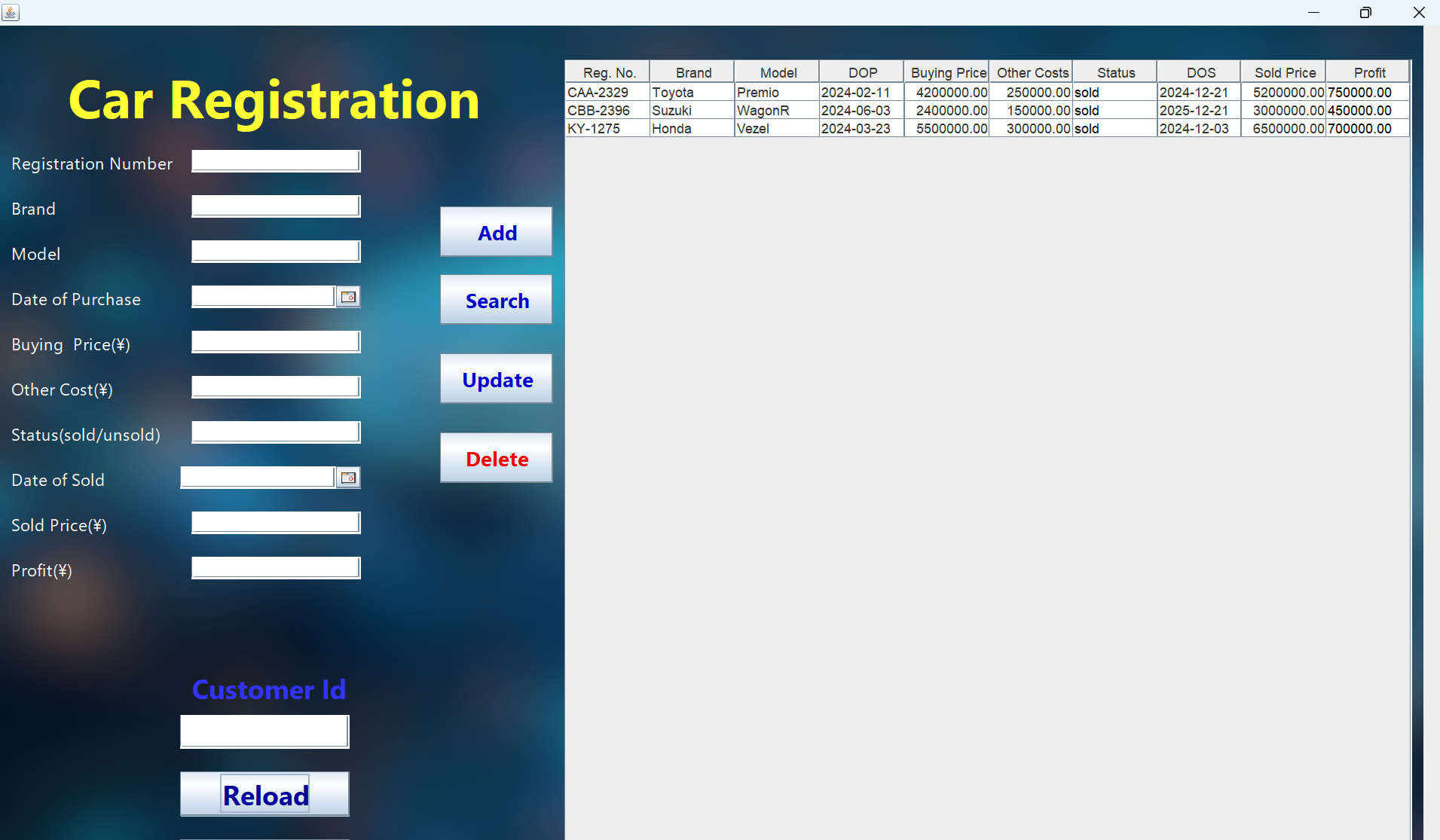
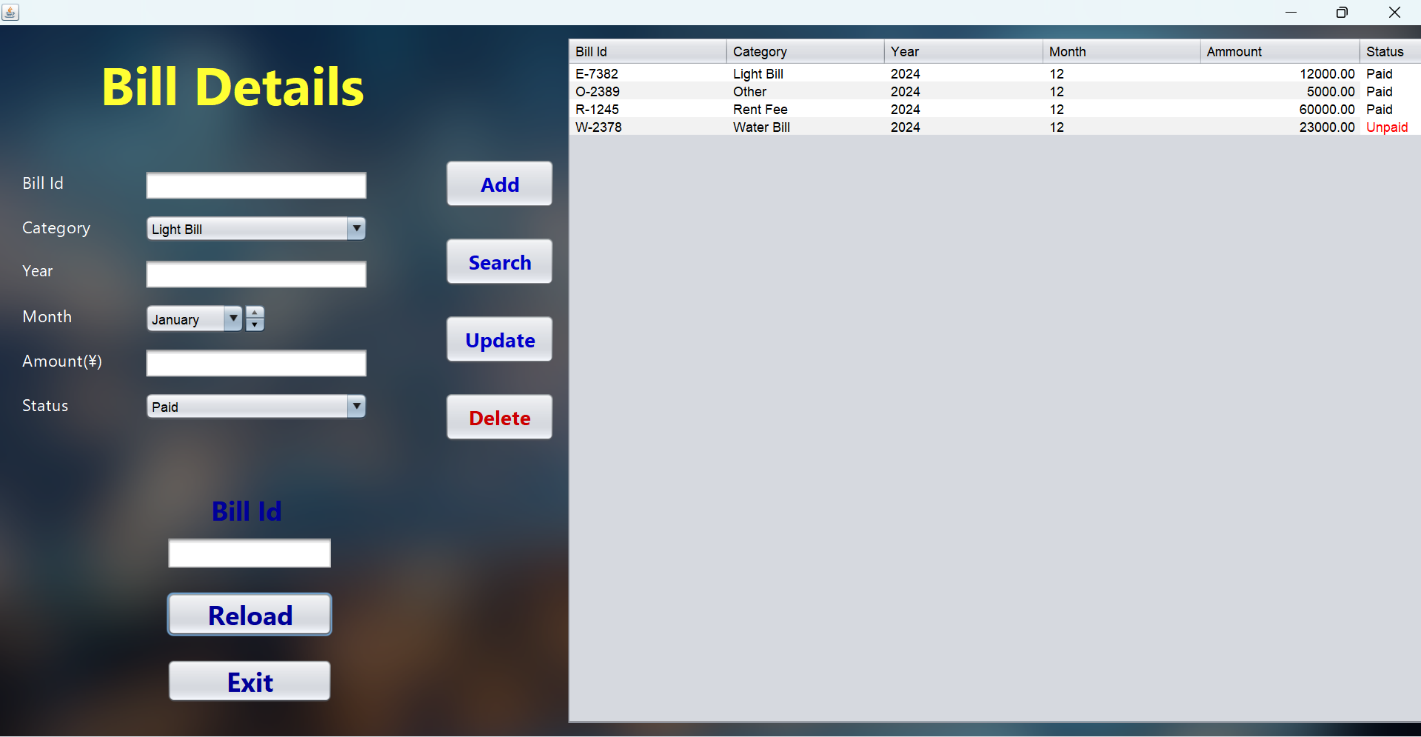
Car Deal Details

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Reg Nu. | Car | Date of purchase | Price | Extra Cost | Date of sold | Sold Price |
| CAA-2329 | Toyota Premio | 11/02//2024 | 4,200,000 | 250,000 | 21/12/2024 | 5,200,000 |
| CBB-2396 | Suzuki WagonR | 3/06/2024 | 2,400,000 | 150,000 | 25/12/2024 | 3,000,000 |
| KY-1275 | Honda Vezel | 23/03/2024 | 5,500,000 | 300,000 | 3/12/2024 | 6,500,000 |

Other Expenses Details

|  |  |  |  |
| --- | --- | --- | --- |
| Bill Id | Bill Type | Month | Amount and Status |
| R-1245 | Rent | December | 60,000.00/Paid |
| E-7382 | Electricity | December | 12,000.00/Paid |
| W-2378 | Water Bill | December | 23,000.00/Unpaid |
| 2389 | Other Bills | December | 5000.00/Paid |

* Total Profit from All Cars : 750,000 + 450,000 + 700,000 = Rs. 1,900,000
* Total Spends (Paid Only)Total : 60,000 + 12,000 + 5,000 = Rs. 77,000.00
* Total Income: 1,900,00-77,000= 1,823,000.00

The above profit and expense calculations were manually computed to demonstrate a real-world scenario. However, the same process can be seamlessly performed using the AutoZen application.

Other Expenses records inserted to the System system

Car records inserted to the system



Total profit is calculated

Both the manual calculations and the AutoZen system produced the same net profit value of **Rs. 1,823,000.00**, confirming that the application works correctly. This verifies that AutoZen accurately integrates data from car sales and expenses, and has successfully passed integration testing.

**6. Conclusion**

The AutoZen application was developed to provide a comprehensive, efficient, and user-friendly solution for managing operations in small to medium-scale car dealership businesses. This project specifically aimed to solve key problems faced by dealers, such as manual record-keeping, fragmented profit tracking, and inconsistent customer data management. By integrating multiple business functionalities into a single cohesive system namely vehicle management, customer details handling, expense tracking, and profit/loss calculation AutoZen streamlines operational workflows and supports data-driven decision-making.

From a technical perspective, the application was built using Java Swing within the NetBeans IDE to design the graphical user interface, while MySQL was chosen as the backend database to ensure secure, scalable, and efficient data storage. The program logic follows core object-oriented programming principles such as inheritance, encapsulation, abstraction, and polymorphism to promote maintainability and future extensibility. Key features such as real-time profit calculations, and automatic data validation enhance usability and functionality.

Throughout the development lifecycle, rigorous testing was conducted including functional testing, user interface testing, validation testing, database testing, and integration testing. Each module was carefully verified to ensure correct behavior, seamless interaction across components, and accurate data processing. Real world scenarios involving multiple car sales and expense entries were tested manually and programmatically, with both yielding consistent results. This confirmed the system’s internal logic was correct and its outputs reliable, validating both functional and non-functional requirements.

One of the key strengths of AutoZen lies in its simplicity and accessibility allowing non-technical users such as sales or office staff to operate the system effectively without needing advanced technical knowledge. At the same time, it guarantees accuracy in financial reporting, significantly reducing the risk of human error and improving daily operational efficiency.

In conclusion, AutoZen has achieved its design goals by delivering a reliable, efficient, and scalable desktop software solution tailored to the needs of modern car dealerships. It not only increases productivity but also enhances transparency, data accuracy, and overall business decision-making. With strong technical foundations and potential for future enhancements such as online data access, role-based user authentication, and advanced analytics, AutoZen stands as a robust platform that supports the digital transformation of the vehicle trading industry.

**7. Timeline**

|  |  |
| --- | --- |
| **No. of the week** | **Task** |
| 1st week | Project initiation and planning |
| 2nd week | Preparing the project proposal |
| 3rd week | Design and Architecture |
| 4th week | Frontend and Backend Design |
| 5th week |
| 6th week |
| 7th week |
| 8th week |
| 9th week |
| 10th week |
| 11th week |
| 12th week | Testing |
| 13th week | Error Correction |
| 14th week | Demonstration |
| 15th week | Submit Final Documentation Report |

***End.***