## *Introduction*

This document provides comprehensive details about the Garage System Desktop Application built with the .NET Framework and C#. The application is designed to manage reservations, calculate payments, and provide statistics for a garage with a user-friendly interface. It includes forms for logging in, making reservations, visualizing statistics, and booking tickets with various payment options.

* ***System Requirements***
* Software: .NET Framework (version), Visual Studio (version), SQL Server
* Hardware: Any standard desktop or laptop with sufficient memory and processing power to run the .NET application
* ***Architecture***

The application follows a client-server architecture:

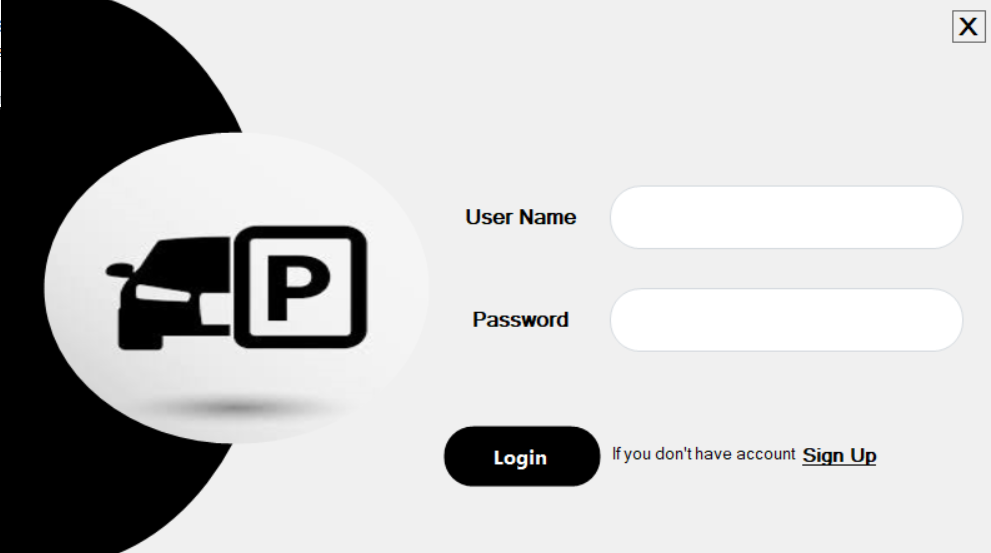
* Client-side: C# application with multiple forms for user interactions
* Server-side: SQL database for storing user, reservation, and trouble data

***Splash Form***

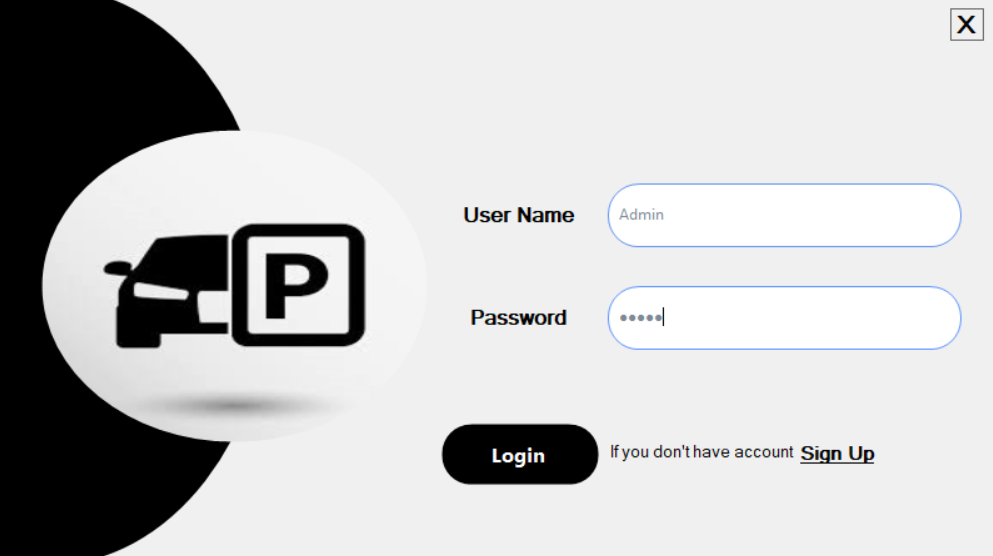
This is A loading until all initializations finished successfully.



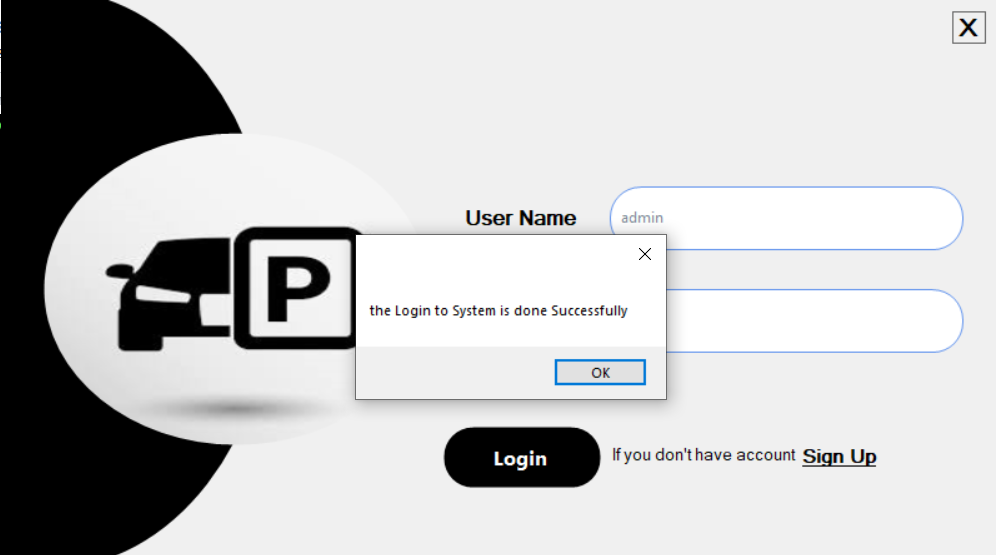
* ***Login Form***



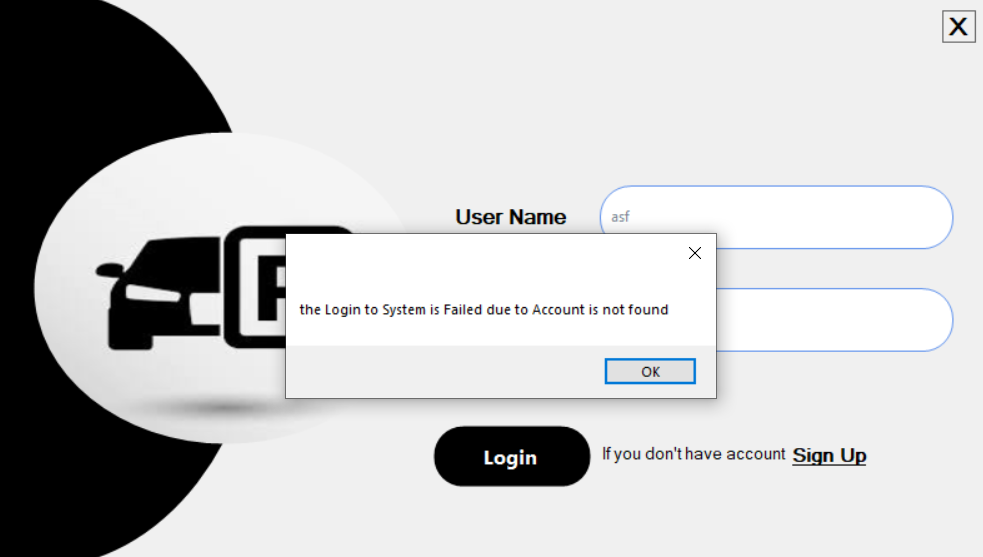
This fields to enter the previously signed up and inserted already in the data Base

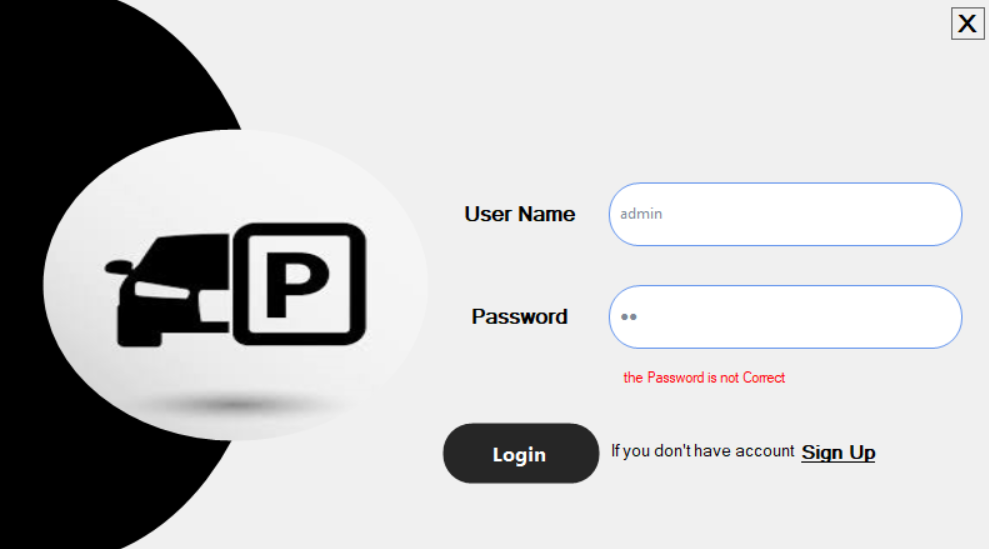


If I entered the Username/Email and its password true and press Login button ,this message will appear

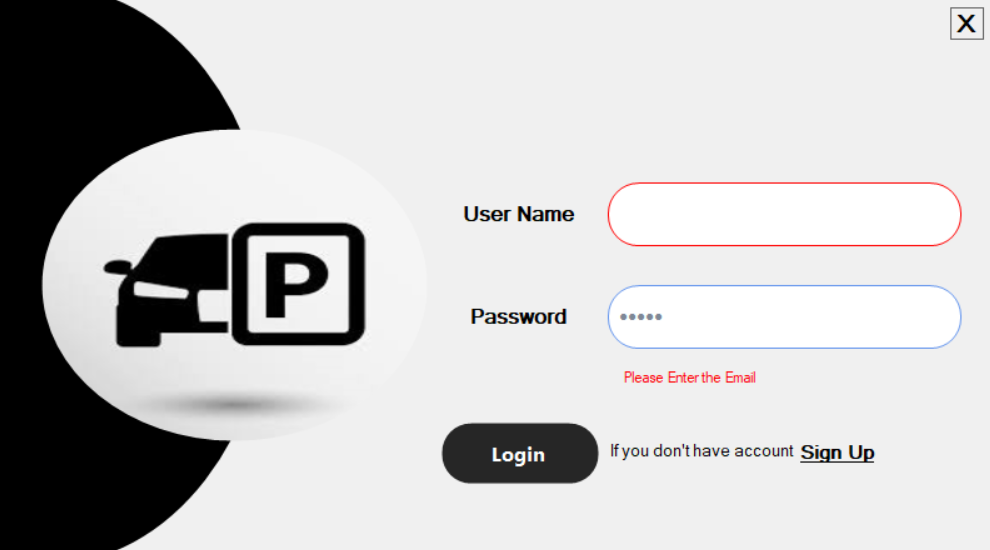


If I entered the Username/Email or password false and press Login button ,these messages will appear

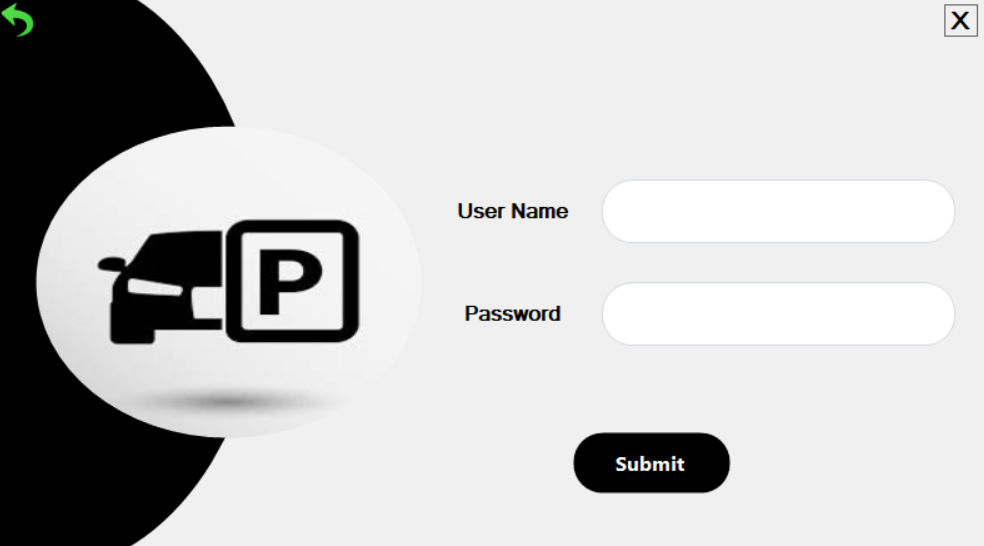




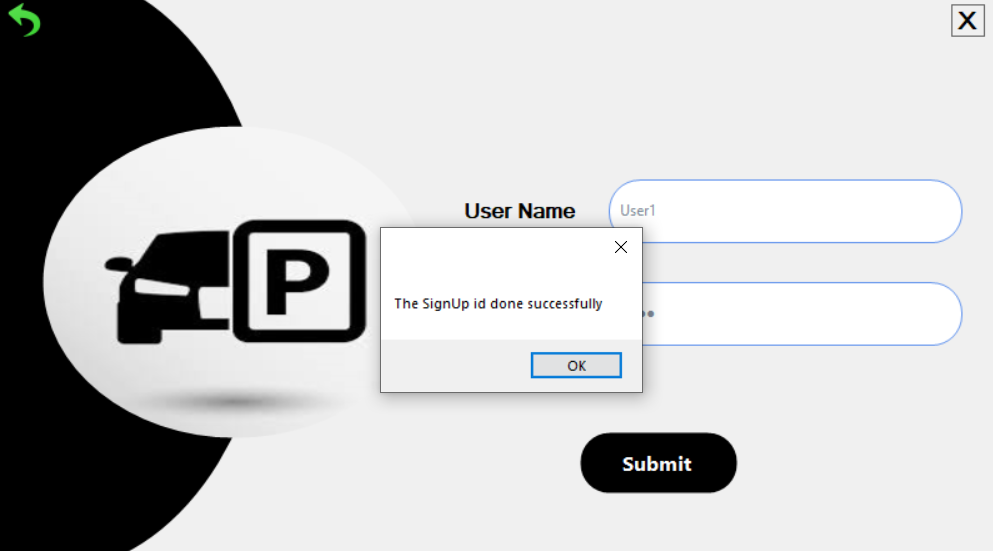
If I left any one of the fields, it notifies me to enter this field previously.



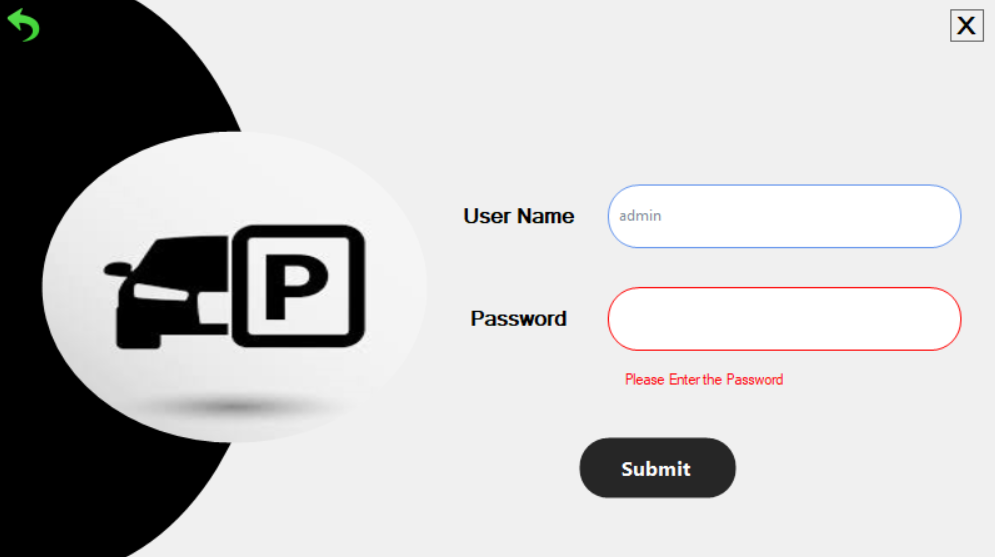
* ***Sign Up Form***



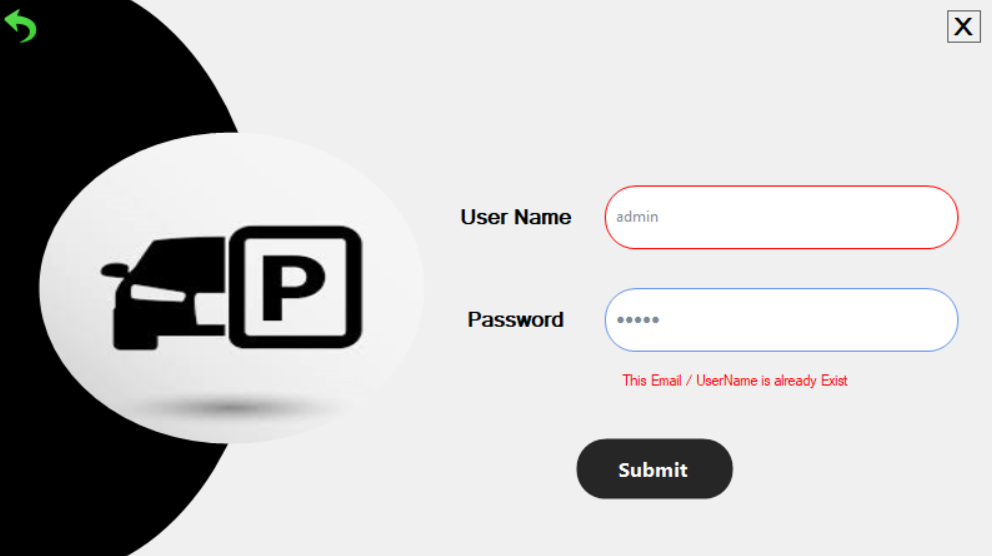
If i entered Username/Email doesn’t exist before and entered Password for it ,then pressing on Submit button, the sign up will be done Successfully.



If I left any one of the fields, it notifies me to enter this field previously.



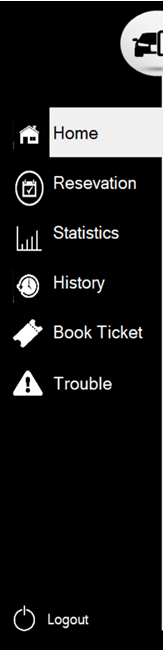
If i entered Username/Email already exists before, it notify me that Username/Email already exists, and I must enter another Username and retry again.

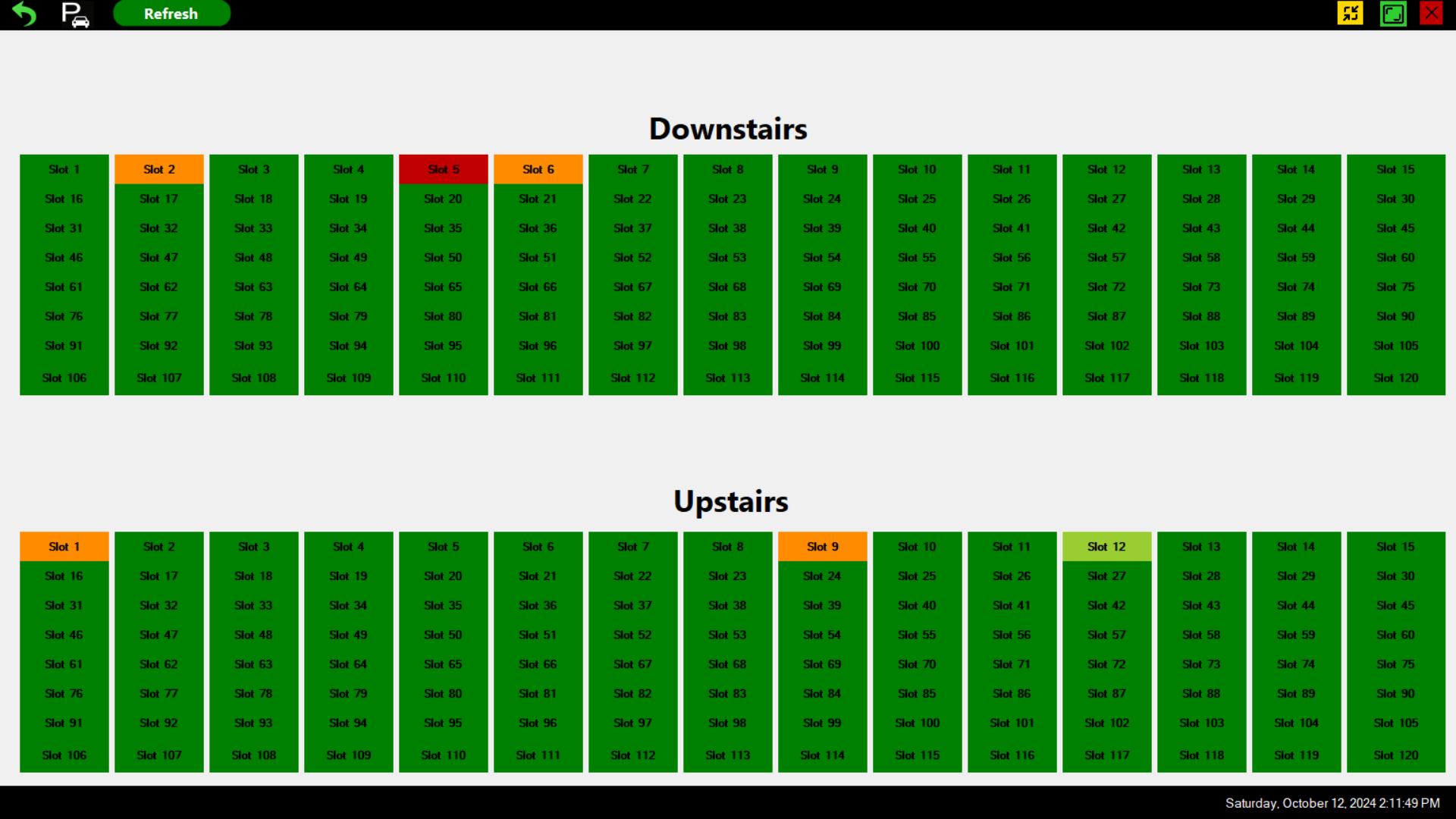


The Green arrow  , its functionality to return to Login Form I want to go to sign in option.

* ***Home Form***

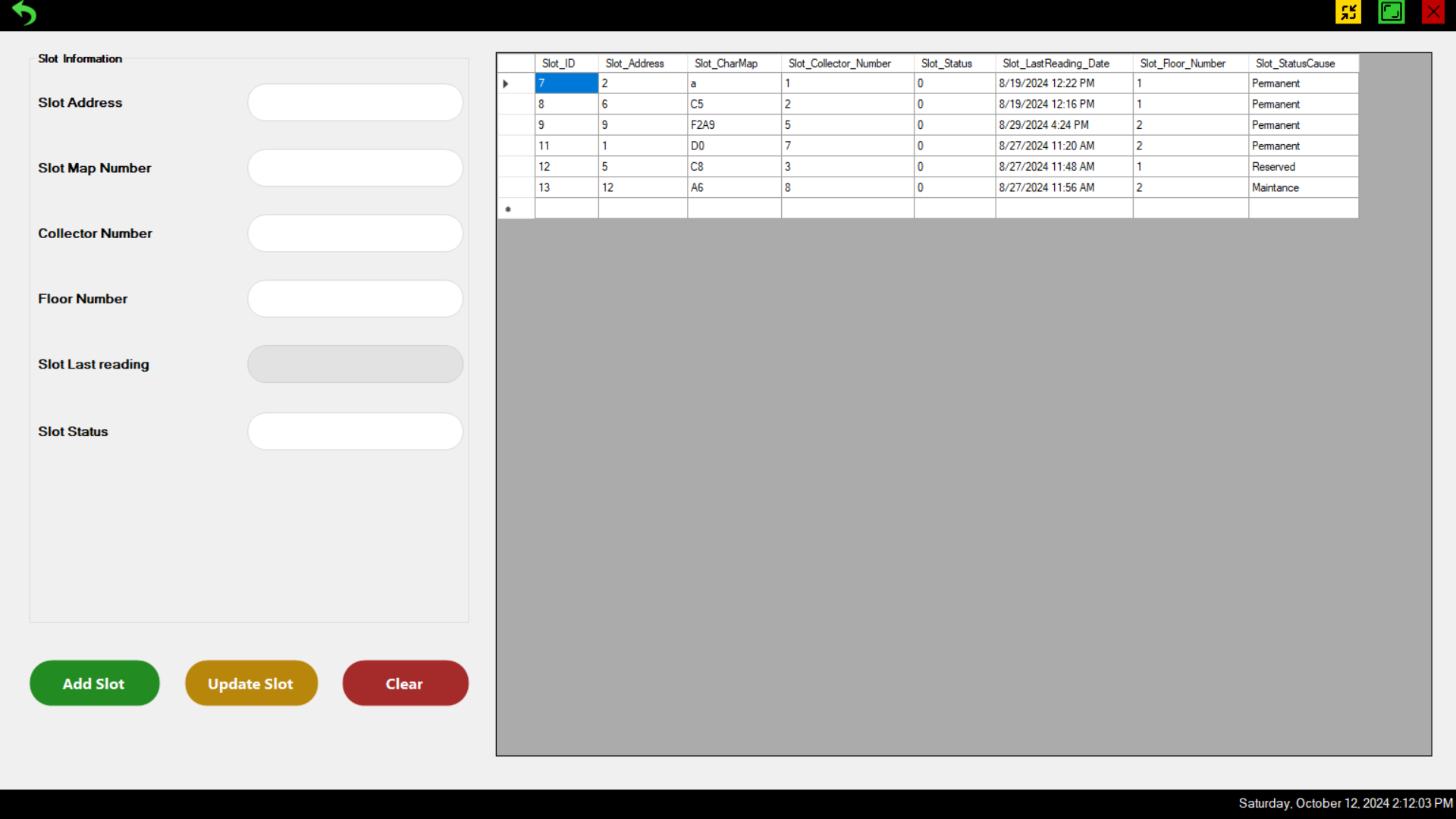


* For the global bar 
* this red button  to Exit and Close the window.
* this green button  to switch between Maximize the window and the normal form size.
* this yellow button  to Minimize the window.
* For the global Left Tab
* Each button from this tab to Switch from Home Form to other form ,as an example when I pressed ‘Statistics’ Button to Switch from’ Home’ Form to ‘Statistics’ Form then if I pressed ‘Reservation’ button , Switch from ‘Statistics’ Form to ‘Reservation’ Form.
* ‘Logout’ button  to logout from the User Account and return to ‘Login’ Form.
* For global Lower bar  it give me information about  
  runtime date and time.
* For “Map slots” Switch  it switches into ‘Slot Map’ Form
* ***Slots Map Form***



In this form, We suppose that we have 2 floors (Upstairs, Downstairs) , each one of then include 120 Slot.

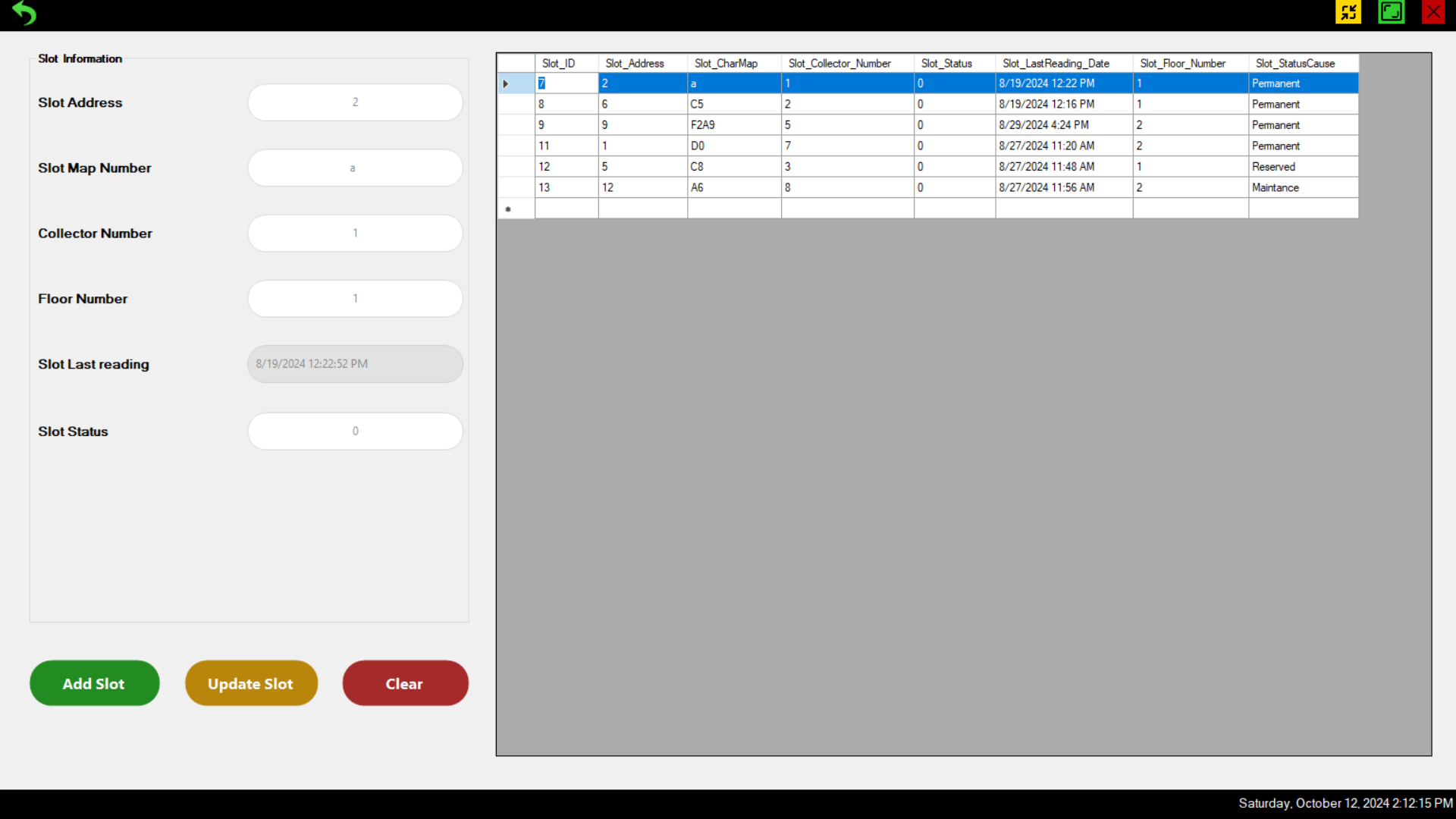
* For Color Mapping in the Slot visualization:
* the Color “Green” indicate that the slot is “Available” now to any Parking.
* the Color “Red” indicate that the slot is “Normal Reserved” and isn’t allowed to any reservations until this slot returns again to “Green” Color.
* the Color “Orange” indicate that the slot is “Permanent Reserved” and isn’t allowed to any reservations happen until the reservation period expired and returns again to “Green” Color.
* the Color “Yellow-Green” indicate that the slot is “Maintenance Reserved” and isn’t allowed to any reservations until this maintenance finished and return again to “Green” Color.
* “Refresh” Button  to load last update of slots status from the database.
* “Slot Dealing” button to enter to “Slot Dealing” Form.
* “Return” Arrow  to return into “Home” Form.
* ***Slots Dealing Form***



The system has 240 slots, each floor has half the number of slots 120 slot, if I recommended 30 slots to each collector, so we have 4 collectors in each floor (totally 8 collectors).

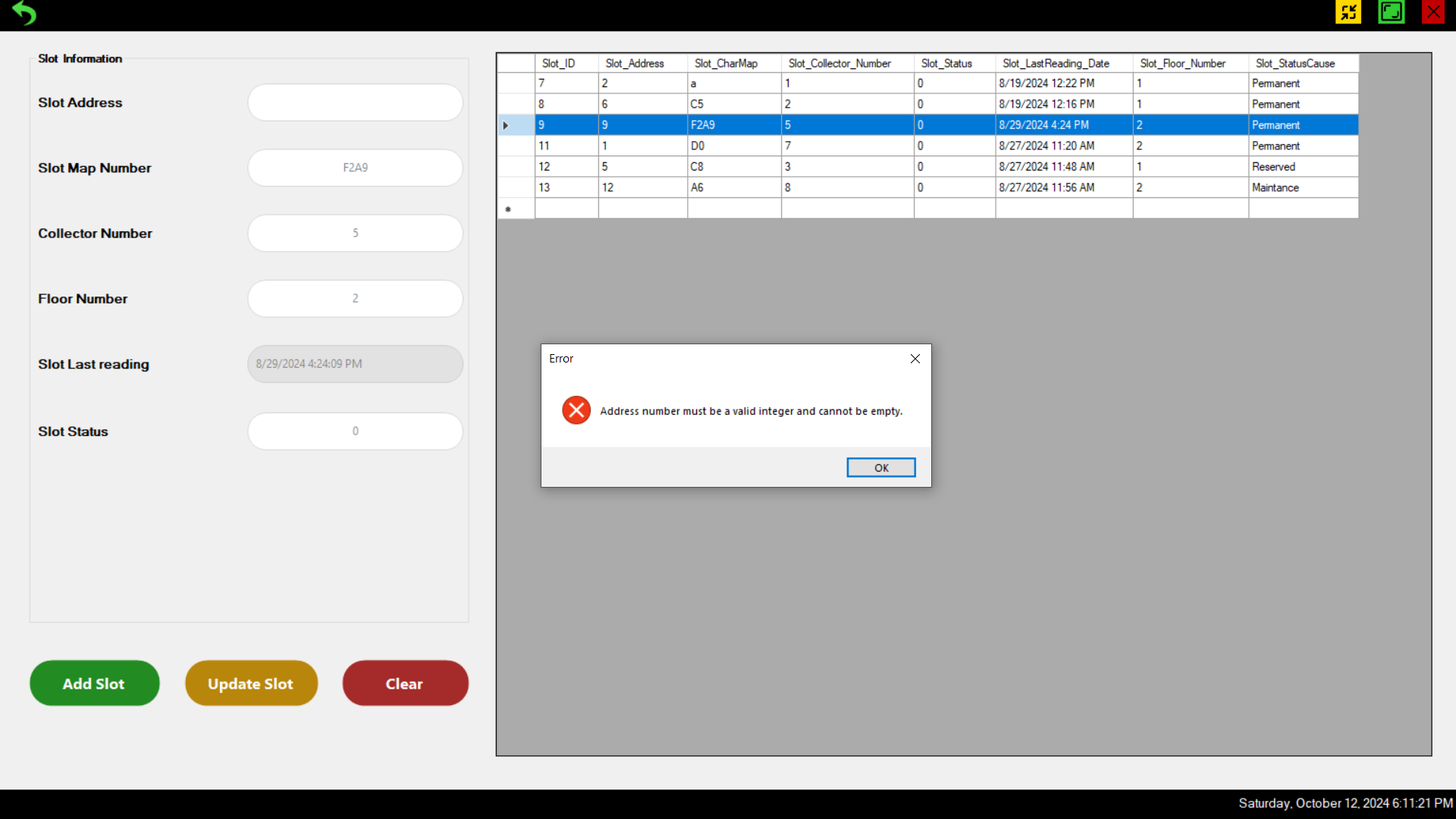
This form to visualize the database table “Slots” which has may attributes:

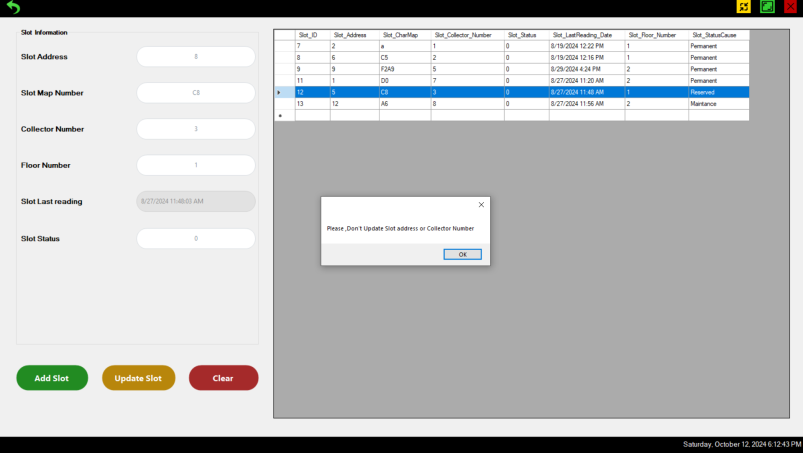
* “Slot ID” this is the index is automatic generated in creating new record.
* “Slot Address” is the unique address, which is get from the Slot HW, which is specified in hardware switches, its range (0, 255).
* “Slot Char Map” begin with “F” char then floor number, then Collector char “A, B, C, D”, then the index of the slot in the collector.  
  ex: F2B3: this indicate that this slot in floor 2, in second collector (B) with slot Address (30x(Collector\_number-1)+3)= 33.
* “Slot Status” this to indicate the slot status 0 for reservation (Normal, Permanent, Maintenance), 1 for Availability.
* “Slot Last Reading” this to indicate the last reading date and time, from this attribute we may know which slot worn out and in need of maintenance.
* “Slot Floor Number” to indicate the floor number.

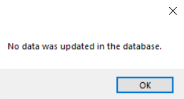


This form basically to deal with database in insertion and update slots data manually.

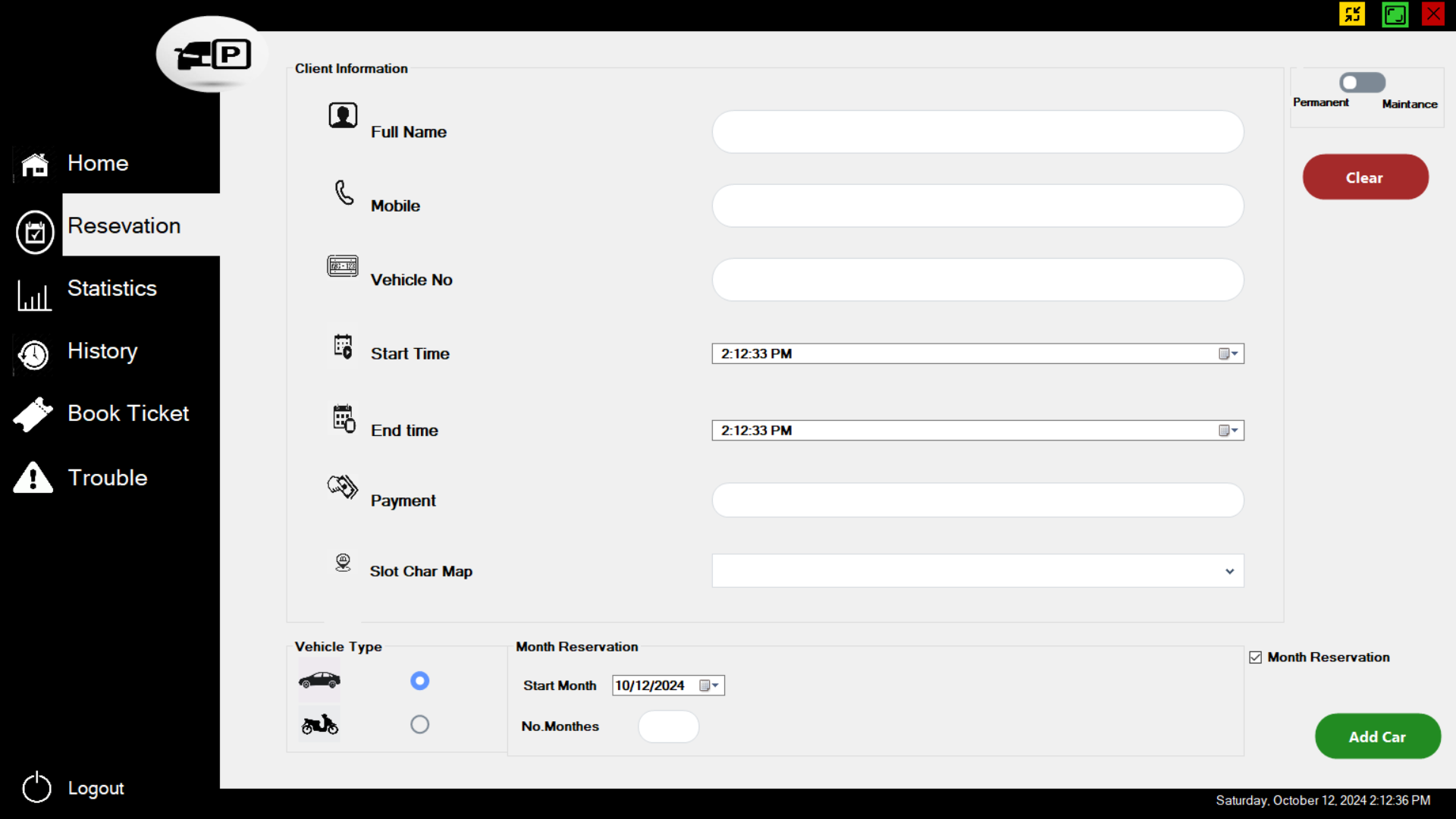
* If you pressed in any cell, the selected record would copy the text fields.
* “Add Slot” button  to take all entered data and insert it as a new record in the database. If any field don’t exist it notify me that this field must be field.



* “Update Slot” to update data of previously entered data slot. If you update Slot address or collector number, if refused to update anything.



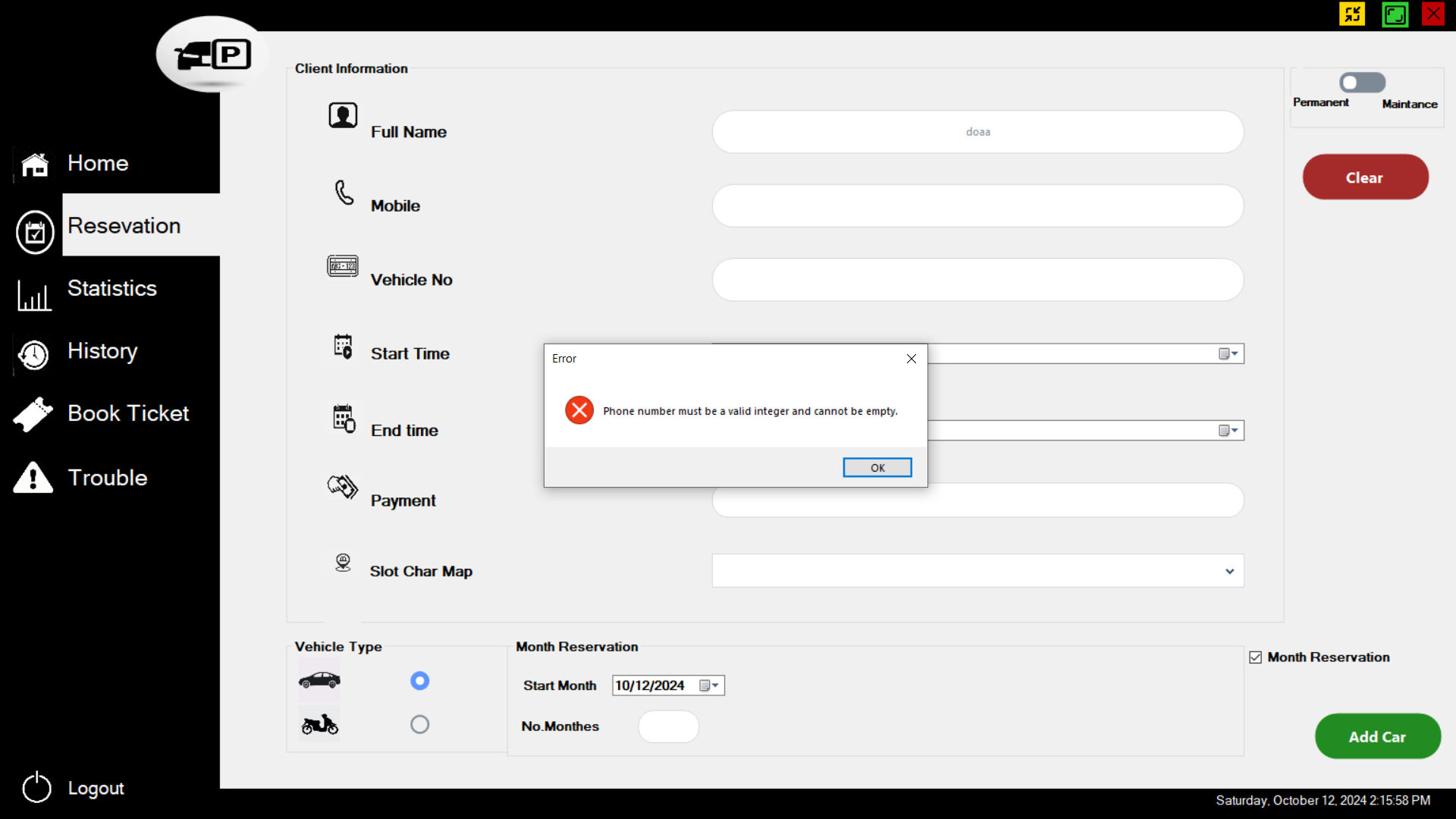
* ***Reservation Form***



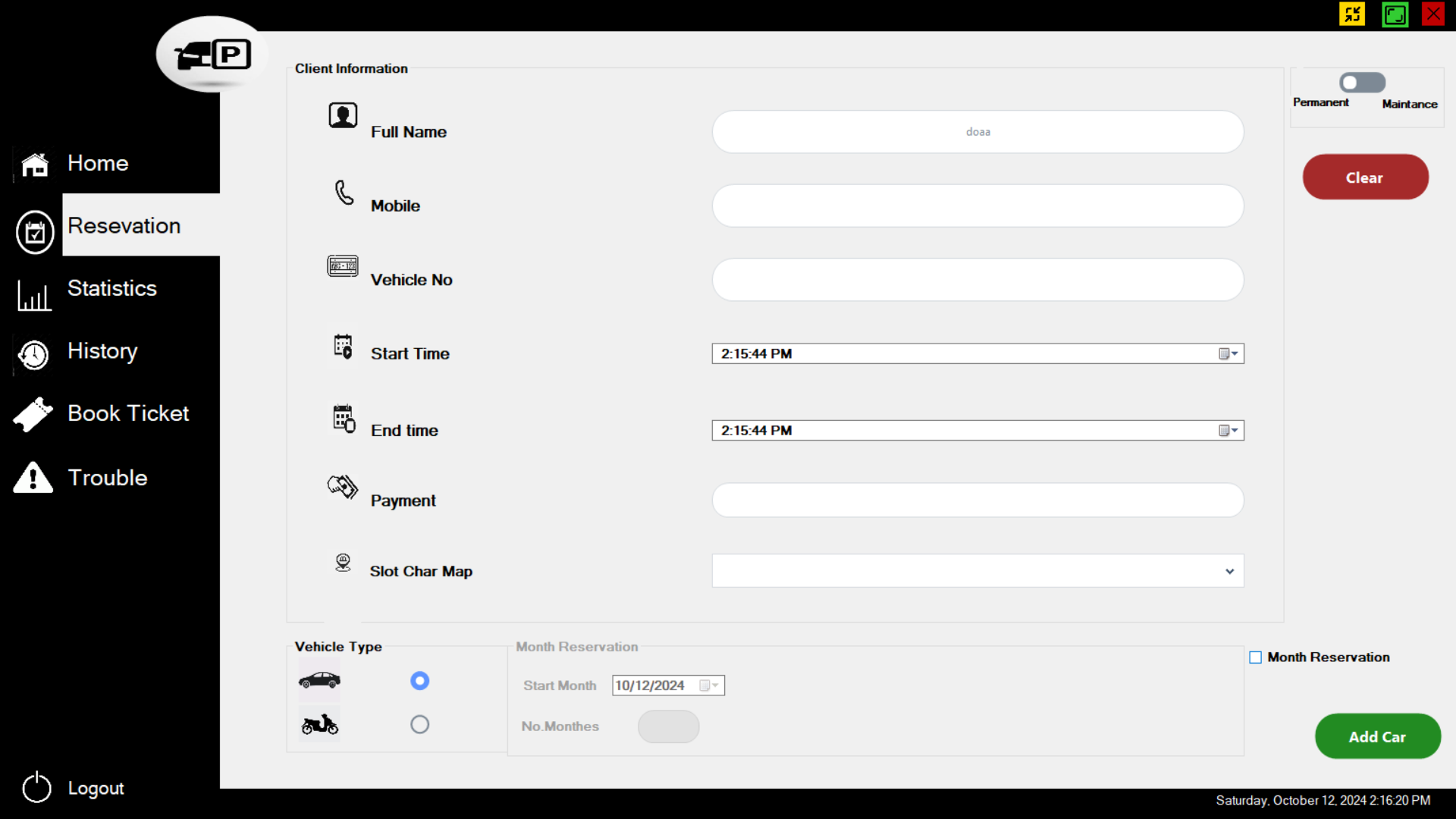
This Form is responsible for manual reservation by specific time or by selecting months using a checkbox.

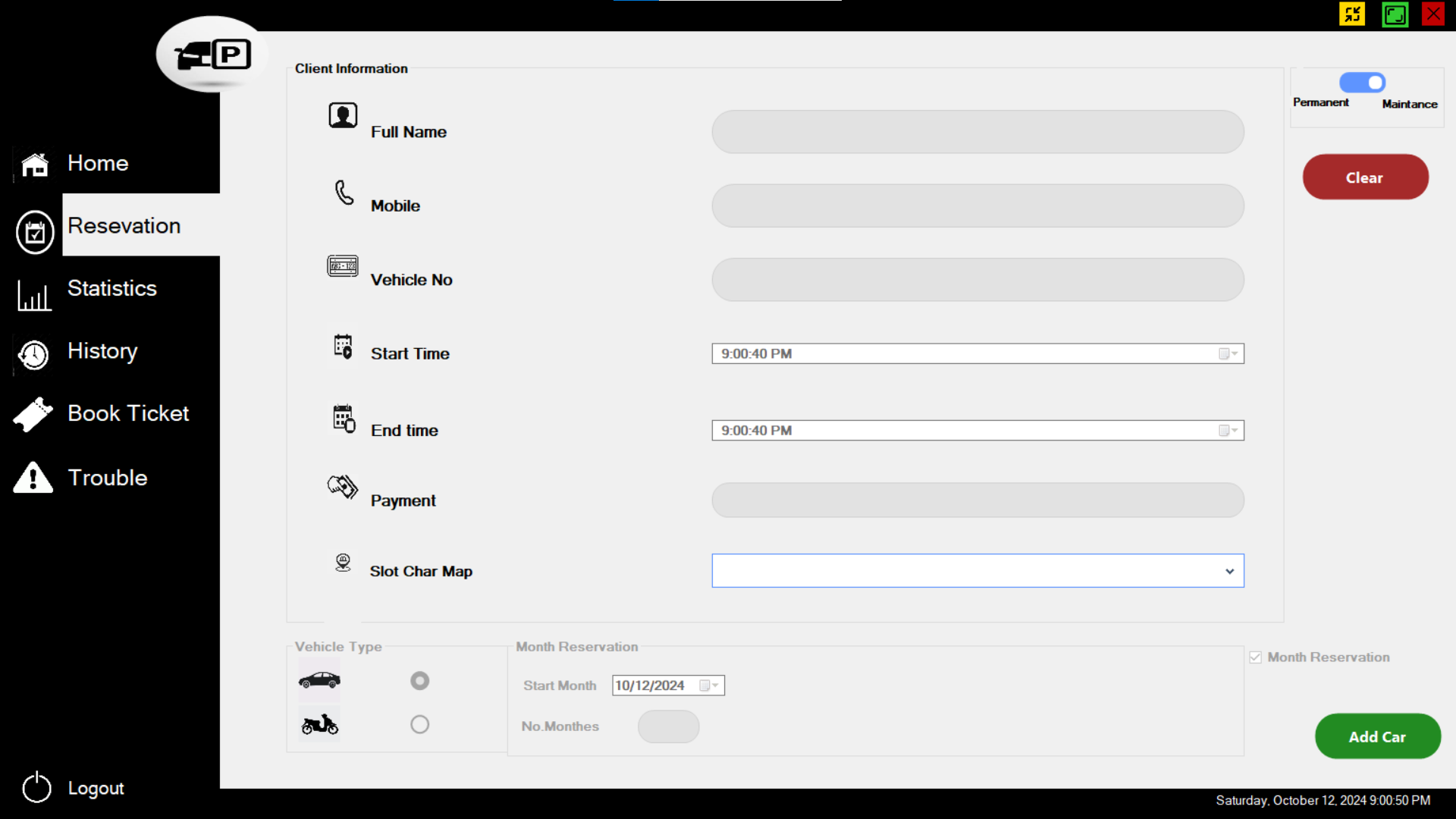
* Group Client Information Has the data of the client info such as Username, phone number, vehicle number, reservation start and end times. Payment system calculation based on the entered duration or selected months. And based on the chosen payment system in “Book Ticket” Form, the Money calculated and displayed in Payment field. And I must assign this reservation to specific available slot.
* If I want monthly reservation not specific time, you must press on “Month Reservation”  
  checkbox and enter the “Start Month” and the number of months.
* You must enter the type of the vehicle to help the HW sensor, if it doesn’t recognize it.
* If I make Maintenance to specific slot, I must choose the option of it, as the default is the permanent reservation.

If i don’t enter any field of this data, then press “Add car” button, it notify that this field must be filled before and operation.

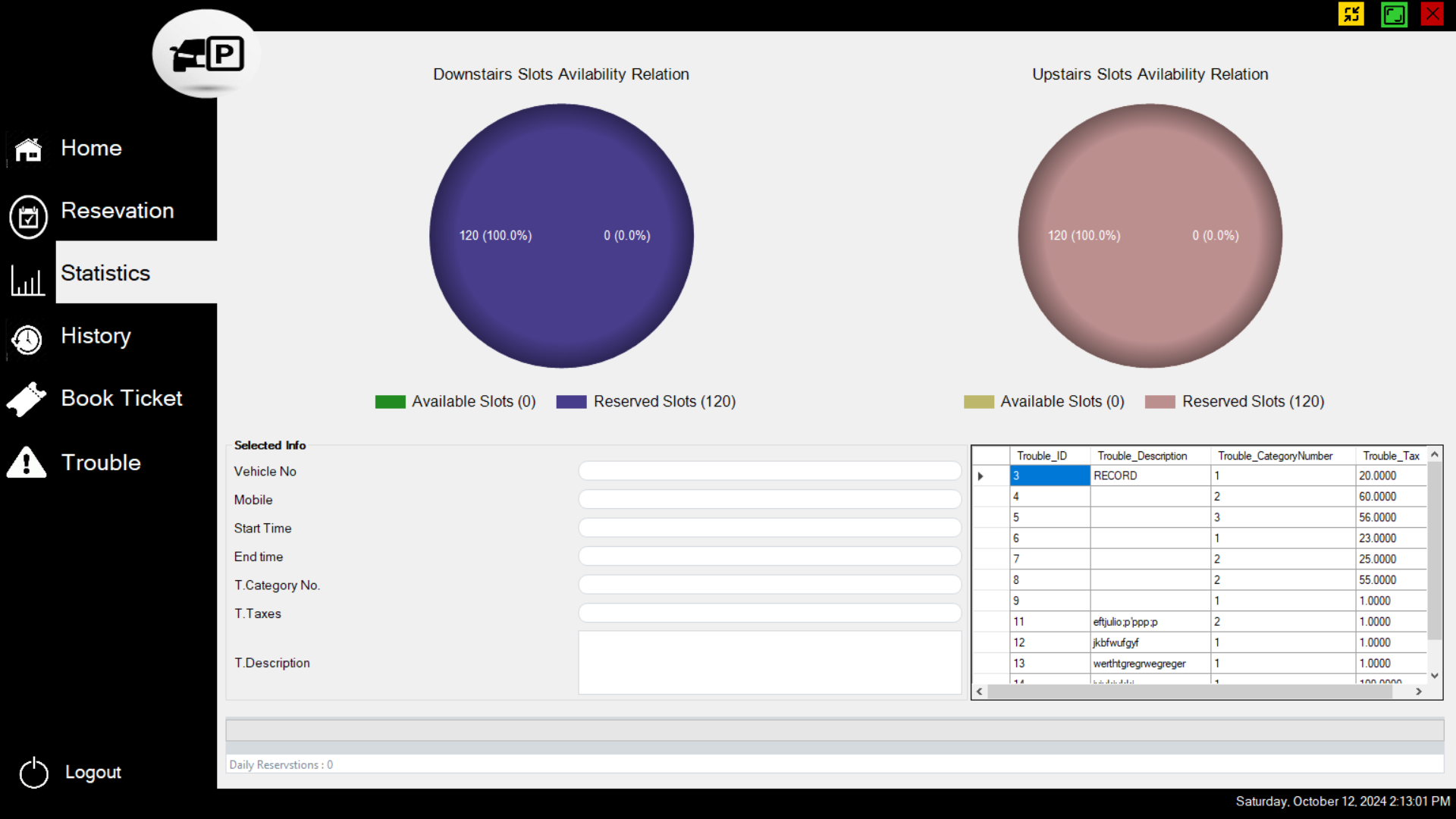


When I make specific time for reservations, not monthly reservations



When I assign maintenance reservation all blocks are disabled instead of "Slot char Map" field to choose the slot of need a maintenance.

* ***Statistics Form***

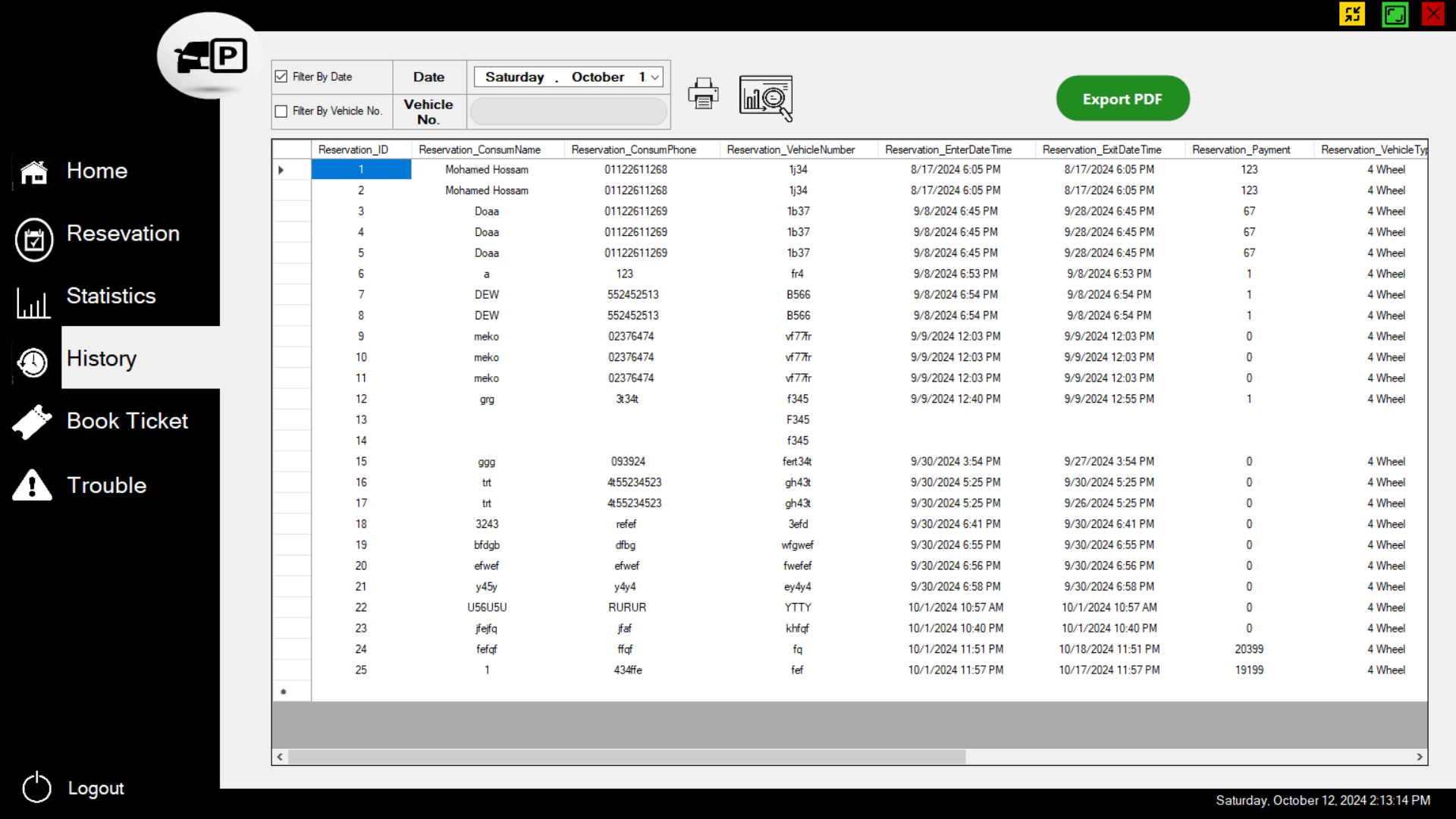


 **Slot Availability**: Pie chart displaying the percentages of available and reserved slots.

 **Trouble Table**: List all troubles with ID, description, category, and tax amount.

 **Daily Reservations**: Display daily counts of reservations for management tracking.

* ***History Form***



The History Form provides users with a comprehensive view of all past reservations in the garage system. Users can filter reservations by date, vehicle number, or both to narrow down the list. The form also includes options for printing all reservations, daily reservations, or filtered reservations based on the selected criteria.

**Display of Reservation History**

* **Reservation Table**:
  + **Columns**: The table displays key reservation details, such as Reservation ID, Date, Vehicle Number, Username, and Reservation Duration.
  + **Dynamic Filtering**: The table updates automatically based on the selected filter criteria, showing only the relevant reservations.

**Form Fields and Filters**

* **Search by Date**: A date picker field that lets users select a specific date to filter reservations. This filter will narrow down the results to show only reservations made on the selected date.
* **Search by Vehicle Number**: A text field where users can enter a vehicle number to filter reservations associated with that vehicle.
* **Apply Both Filters**: Users can utilize both filters simultaneously to locate reservations for a particular vehicle on a specific date.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

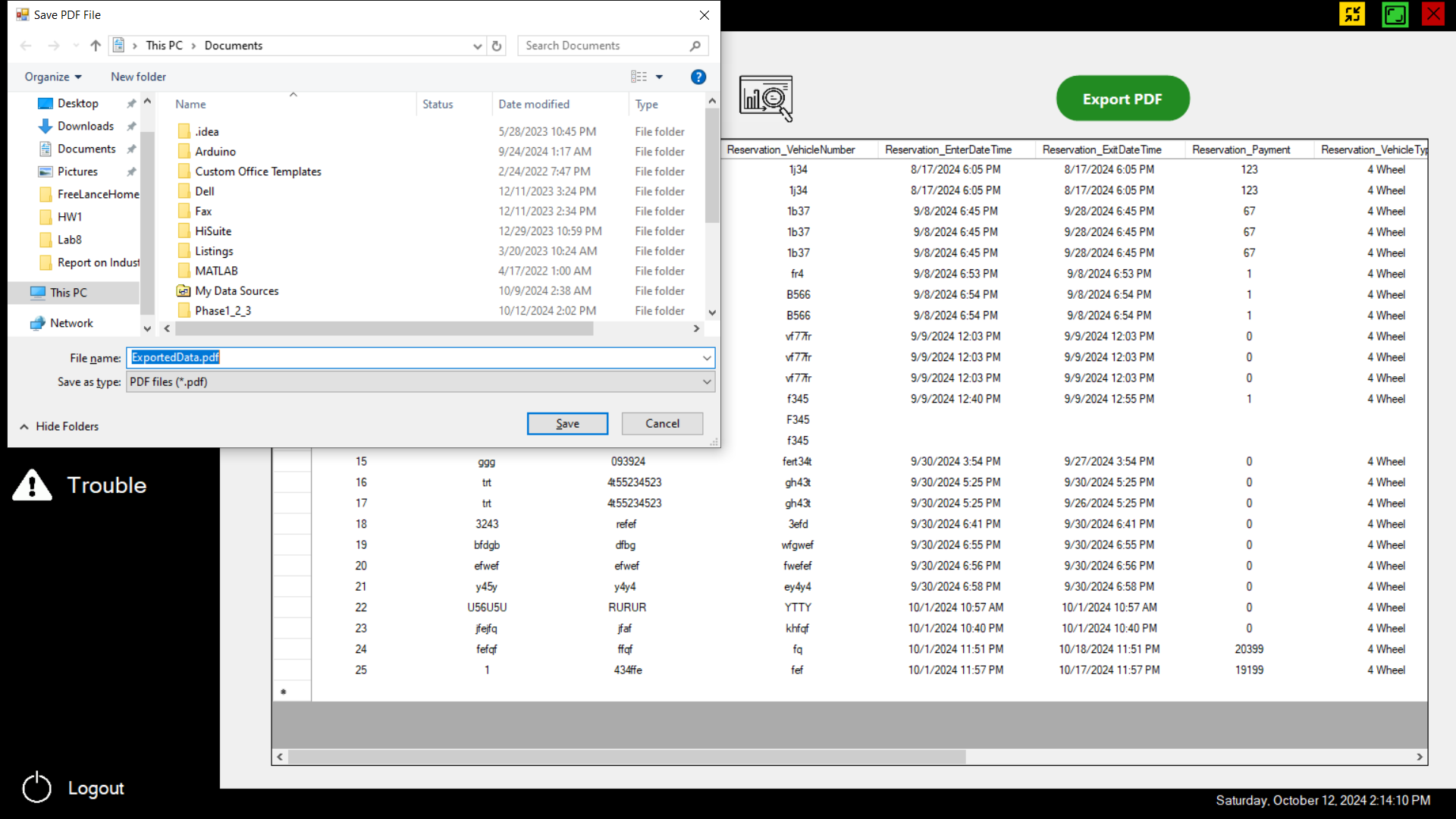
Description automatically generated

**A Printing Options**

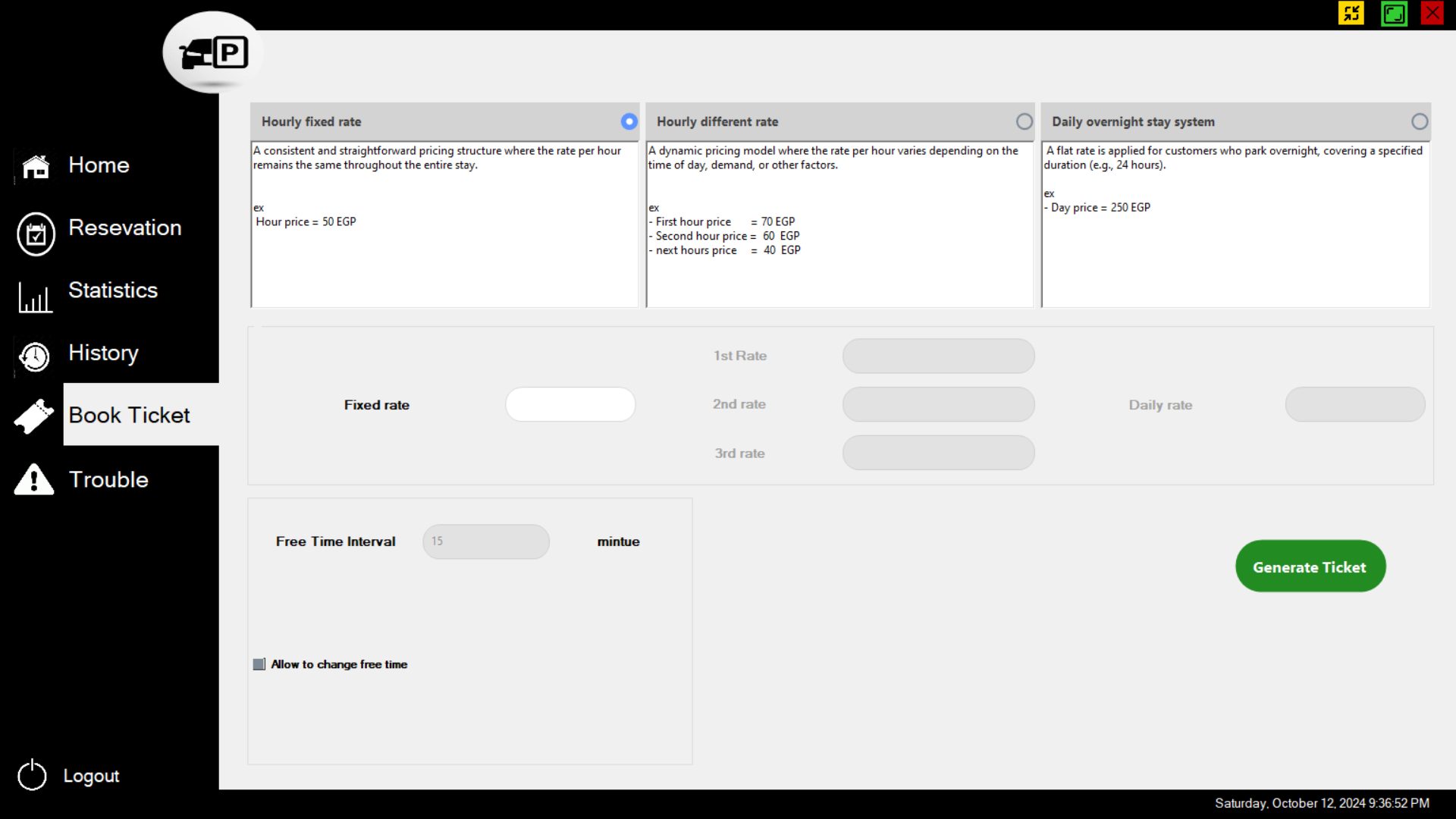
The History Form includes three distinct printing options to allow users to generate hard copies of reservation data:

1. **Printing All Reservations**: 
   * This option prints a complete list of all reservations in the database, regardless of any filters applied.
2. **Printing Daily Reservations**: A graphic with a magnifying glass and graph

   Description automatically generated
   * Allows users to print reservations filtered by the selected date. This option is ideal for generating daily summaries or reports.
3. **Printing Filtered Reservations**: 
   * Prints only the reservations that match the filter criteria (by date, vehicle number, or both), providing a customized report based on the user’s search.



* ***Book Ticket Form***

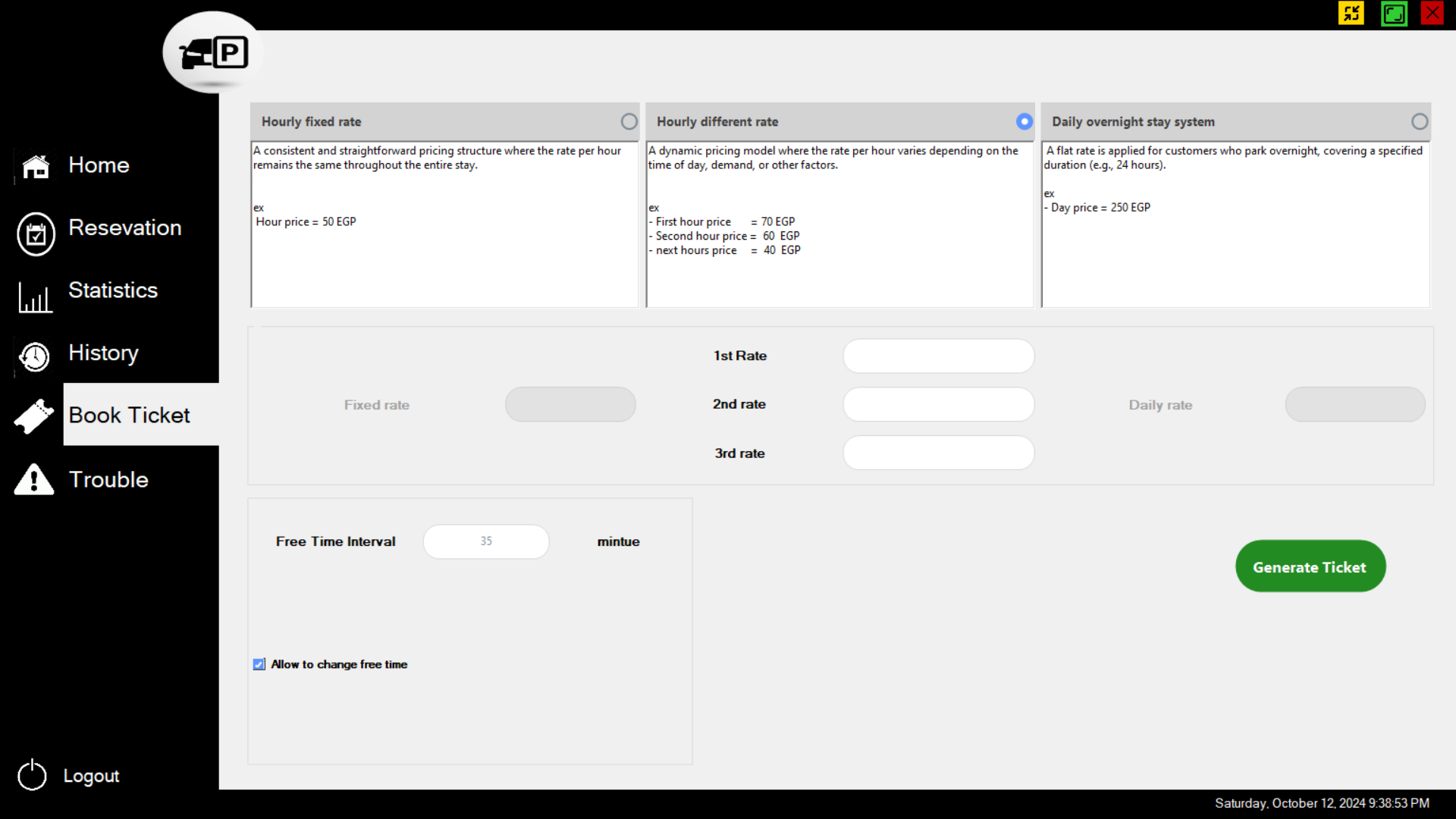


 Payment **Systems**:

* **Fixed Hourly Rate**: Hourly rate remains constant (e.g., 50 EGP/hour).
* **Dynamic Hourly Rate**: Varies by hour (e.g., 70 EGP for the first hour, 60 EGP for the second, 40 EGP for subsequent hours).
* **Daily Overnight Rate**: Flat rate for 24-hour stays (e.g., 250 EGP).

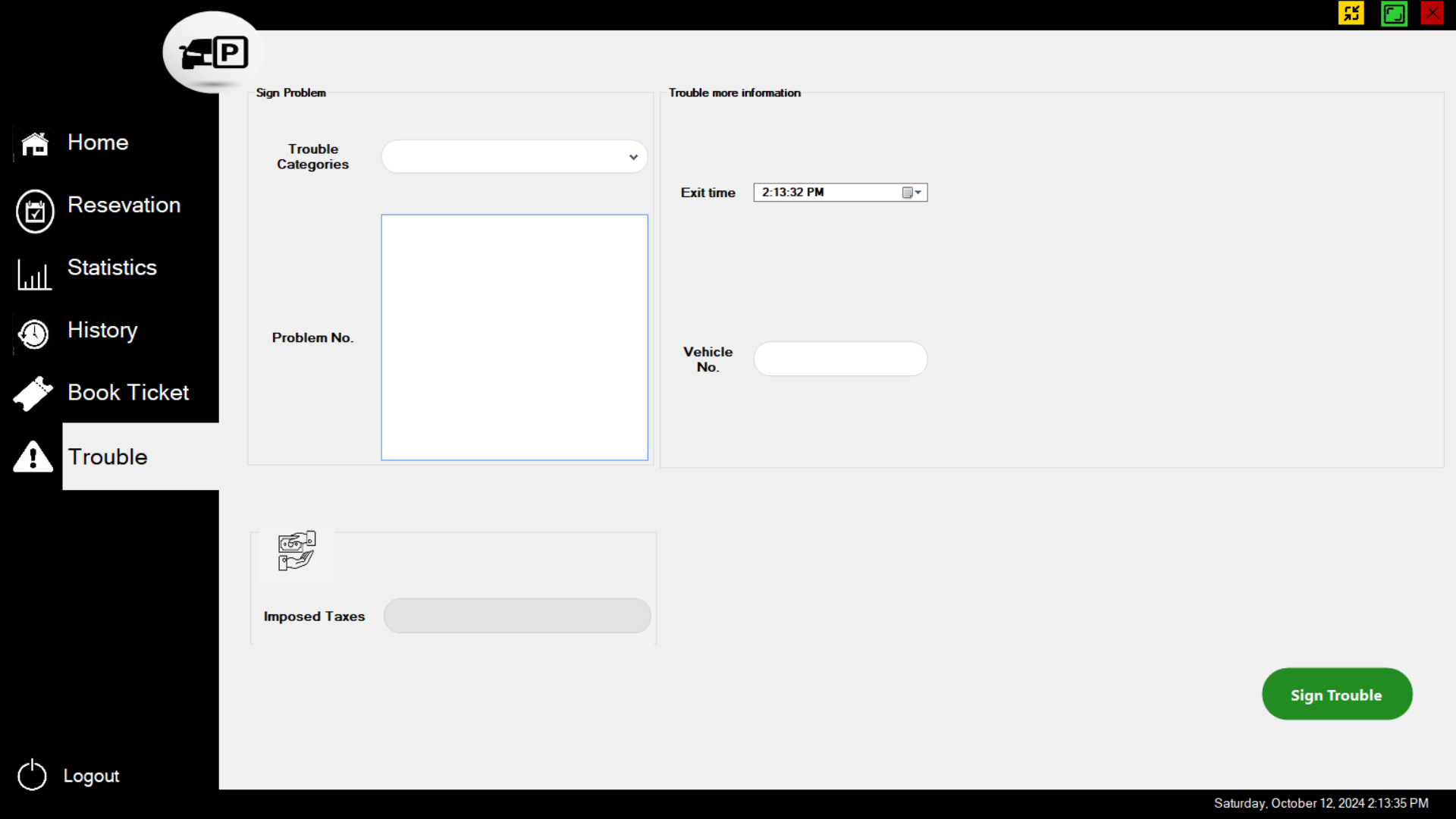
And this Fields to override new rate instead the Default values. And these values affected in “Reservation” form.

 Free **Time Interval**: Adjustable field defaulting to 15 minutes., if I want to change it, I must press the bellow checkbox .

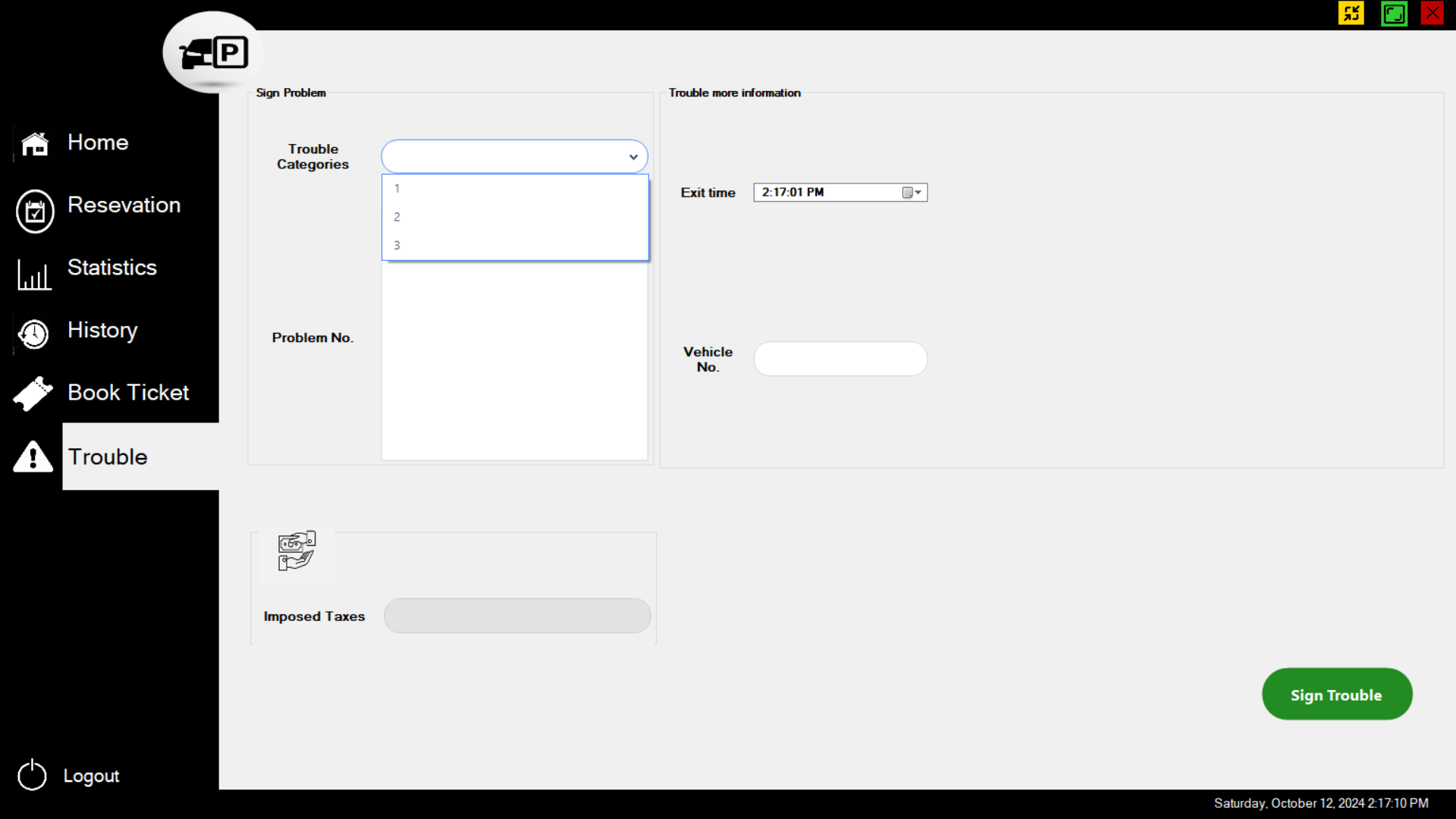


The button  is to set this information and generate the Ticket with enter time and id.

* ***Trouble Form***



This form to enter troubles happen in garage to store it in data base , so if any incoming problem will happen related to previous problem.



The Trouble Form allows users to record details of any issues or troubles encountered with a vehicle. These troubles are categorized into three levels (1, 2, and 3), which correspond to the severity of the issue. Each category has an associated tax rate, which is calculated and displayed based on the chosen category.

**Features**

* **Categorization by Severity**: The trouble is classified into three categories (1, 2, or 3), with each category representing the scale or severity of the problem.
* **Automatic Tax Calculation**: Based on the selected category, the form calculates the applicable tax and displays it in real-time.
* **Category 1 (Minor Trouble)**:

Imposed Tax: 50 EGP

* **Category 2 (Moderate Trouble)**:

Imposed Tax: 100 EGP

* **Category 3 (Severe Trouble)**:

Imposed Tax: 150 EGP

* **Data Entry Fields**: Fields for entering a description of the trouble, vehicle exit time, and vehicle number.
* **Trouble Description**: A text field where users can provide details about the trouble encountered with the vehicle.
* **Exit Time of Car**: A date-time picker to record the time the vehicle left the garage.
* **Vehicle Number**: A text field to enter the vehicle’s registration number.
* **Trouble Category**: A dropdown menu or radio buttons to select the trouble category (1, 2, or 3).
* **Imposed Tax**: A read-only field where the calculated tax based on the category is displayed.