EV Info Management System

Presented by Team – 3 (Python Ninjas)

SSM – 15089

SSM – 15431

SSM – 15432

SSM - 5117

Acknowledgements

We sincerely thank Teacher Autumn and Teacher Dr. Cinthia White for their support and guideline throughout the project, contributing and teaching the Python programming language, Object-Oriented Programming (OOP) concept. We also would like to express our gratitude to Spring University Myanmar (SUM) for their effort and arranging the class as python is commonly used for developing websites and software, task automation, data analysis, and data visualization. We acquired new skills and it can also support our career development and professional lives.

Finally, we would like to thank to all our team members who contribute their effort and giving time to complete the EV Info Management System.

Abstract

The Electric Vehicle (EV) Info Management System is a desktop application developed using Python programming language and runs on Windows. The system allows the user to add the information of electric cars, modify the existing data, inquiry the required information that has been existed in the system and display the list of the electric cars with their specifications.

Chapter (1)

Introduction

Introduction

An electric car known as electric vehicle (EV), is powered fully by an electric motor and battery instead of a combustion engine powered by gasoline. Electric vehicles use energy stored in main battery, which are recharged by household electricity or charging station. EVs reducing greenhouse gas emissions. The most used electric cars can operate sufficiently to cover the average commute, they are ideal for around city use. The number of public charging stations is increasing every year, but most often battery charging is done at home.

Electric vehicles have other advantages over those powered by combustion engines:

- No fuel required so you save money on gas
- Environmental friendly as they do not emit pollutants
- Lower maintenance due to an efficient electric motor

An information management system (IMS) refers to any framework of software that facilitates the collection, storage, organization, and distribution of information. There are many management information systems, including:

- Process control.
- Management reporting system.
- Inventory control.
- Sales and marketing systems.
- Human resource systems.
- Accounting and finance systems.
- Decision support systems.

We have developed the desktop-based application that can provide the EV information and manipulation of these information. Admin and user will be available to access the system. Admin have full access right that are inserting data to the system, modifying and viewing all the records in the system. User will have limitation of access to the system that are search and view processing.

Objectives

The purpose of EV Info Management System is

- To build a desktop application to reduce manual work
- To allow users to easily store, retrieve and manipulate vehicle data
- To support users to view vehicle information together with vehicle image
- To increase productivity on daily processing

Chapter (2)
System Design

Flow Chart

A flowchart is a type of diagram that represents a workflow or process. A flowchart can also be defined as a diagrammatic representation of an algorithm, a step-by-step approach to solving a task. The flowchart shows the steps as boxes of various kinds, and their order by connecting the boxes with arrows. This diagrammatic representation illustrates a solution model to a given problem. Flowcharts are used in analyzing, designing, documenting or managing a process or program in various fields

System Flow Chart of EV Information Management System

The admin is required to upload the EV car information using csv file. After that the user is available to view the list of the car's information with their specification, able to update and delete the records, these modify record will be saved to the file.

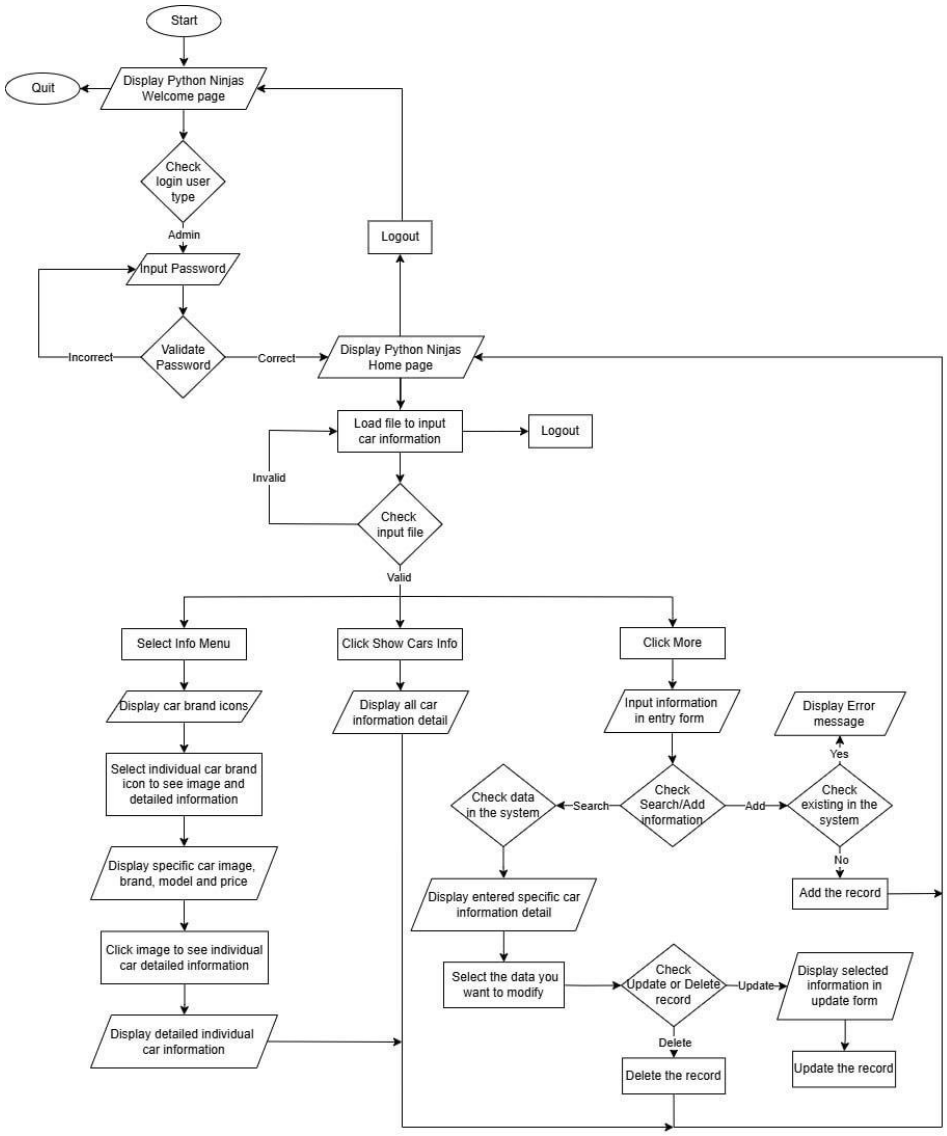


Figure: Admin Flowchart

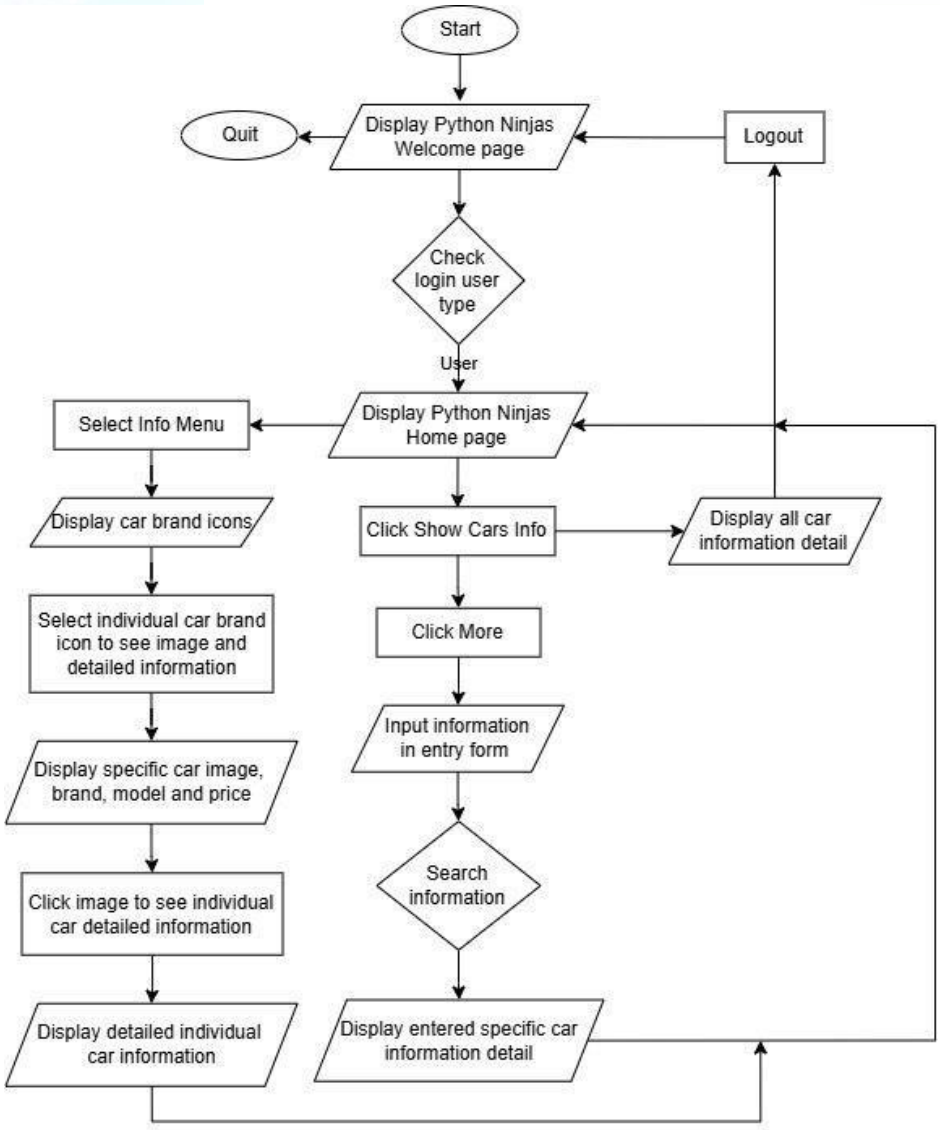


Figure: User Flowchart

Algorithm

1. Start
2. Welcome page requesting login info
    - 2.1 Home menu (data is already loaded for user)
        - 2.1.1 Main page showing 3 buttons
            - 2.1.1.1 Click Show cars info
                - 2.1.1.1.1 Display all information of car data
                - 2.1.1.1.2 Clicking back will return to home page
            - 2.1.1.2 Clicking More
                - 2.1.1.2.1 Display Entry frame with two buttons
                - 2.1.1.2.2 Fill desired entry and click search
                - 2.1.1.2.3 Show searched output in treeview
                - 2.1.1.2.4 Clicking back will return to main page
            - 2.1.1.3 Click Log out will log out and back to welcome page
    - 2.2 Info menu
        - 2.2.1 Display brands logo of data
            - 2.2.1.1 Click brand logo
                - 2.2.1.1.1 Show all models of that brand
                - 2.2.1.1.1.1 Click model image
                - 2.2.1.1.1.2 Show information of that model
        - 2.2.2 Clicking back will return to brands logo
        - 2.2.3 Clicking back will return to main page
    - 2.3 Home menu Admin (verify using log in password)
        - 2.3.1 Main page showing 4 buttons
            - 2.3.1.1 Load file has to be clicked first
                - 2.3.1.1.1 Search and open csv file
            - 2.3.1.2 Click Show cars info
                - 2.3.1.2.1 Display all information of loaded datas
                - 2.3.1.2.2 Clicking back will return to home pages

2..3.1.3 Clicking More

2.3.1.3.1 Display Entry frame with four buttons

. 2.3.1.3.1.1. Fill desired entry and click search

2.3.1.3.1.2 Show searched output in treeview

2.3.1.3.1.3 Fill entry and click add

2.3.1.3.1.4 If first 2 frames wasn't filled or second entry has same name with one data Display error info

2.3.1.3.1.5 If it satisfies desired conditions

2.3.1.3.1.6 Add data with default image(owned image link can be provided in entry)

2.3.1.3.1.7 select desired one row from searched treeview and click update

2.3.1.3.1.8 Show editable entry boxes filled with selected row data and image entry

2.3.1.3.1.9 click save

2.3.1.3.1.10 check changes If it has changes, display changes info with two buttons

2.3.1.3.1.11 Clicking confirm will show message box and save data

2.3.1.3.1.12 Clicking back will back to cancel display changes info

2.3.1.3.1.13 Show message box, save data and back to more page

2.3.1.3.1.14 Select one or more data from searched treeview and click delete

2.3.1.3.1.15 Display (ok, cancel)message box

2.3.1.3.1.16 Click cancel will cancel the process to delete and back to more page

2.3.1.3.1.17 Clicking ok will delete selected row(s), save data and back to more page .5. Clicking back will return to main page

2.3.1.4 Click Log out will log out and back to welcome page

2.3.1 Click Info submenu

2.3.1.1 Display brands logo of data

2.3.1.1.1 Click brand logo

2.3.1.1.1.1 Show all models of that brand

2.3.1.1.1.2 Click model image

2.3.1.1.1.3 Show information of that model

2.3.1.1.1.4 Clicking back will return to brands logo

2.3.1.1.2 Clicking back will return to main page

Technologies Used

Tkinter: Python's built-in library for creating graphical user interfaces (GUIs), to design the application window and handle user interactions.

CSV: You can use CSV files to store and manage data (e.g., vehicle details) in a structured format.

OS: It provides functions for creating and removing a directory (folder), fetching its contents, changing and identifying the current directory, etc.

Pandas: Pandas (styled as pandas) is a software library written for the Python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series.
It can do following:

- Import datasets comma-separated values (CSV) files

PIL (Pillow): PIL (Python Imaging Library) or its fork, Pillow, allows you to work with images, including resizing, cropping, and converting formats.

Ttk :The ttk module provides themed widgets for a more polished GUI appearance.

Code Explanation

Features

- **Tkinter GUI:** Create an intuitive user interface using Tkinter.
- **Data Management:**
  - Store vehicle information (brand, model, price, etc.).
  - Sorting data alphabetically.

- **User Interaction:**
  - Add new vehicle records.
  - Update existing records.
  - Search for specific vehicles.
  - Delete records.
- **File Handling:**
  - **Using csv module**
    - Load data from CSV files
    - Save data to CSV files.
  - **Using pandas module**
    - Search data using user provided info
- **Image Display:**
  - Show vehicle images using PIL (Pillow).
- **Event Binding:**
  - Bind events (e.g., button clicks) to specific actions.
- **Data Handling**:
  - In this project we target to use csv file contains less column numbers and it is easy to create, manage and work on. We refer our csv data from this evcars data set[1], and subset and store it in csv file. In code, we read and write csv files using csv module. We used Pandas module to handle search information that is nearly equal to user provided data.

Chapter (3)

System Implementation

Welcome Form

When starting EV info management system, the welcome page will be show and user need to login to use the system. There will be two user types, admin and user.

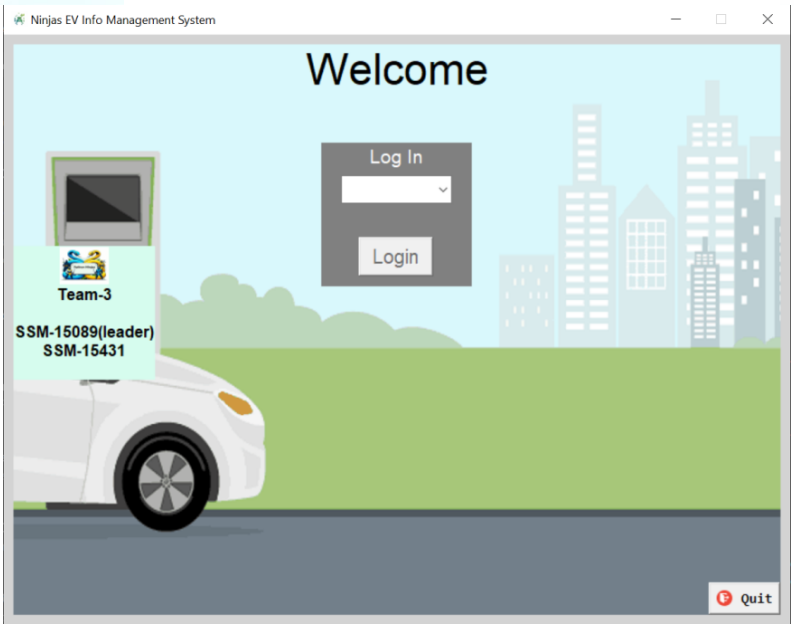


Figure: Welcome Form

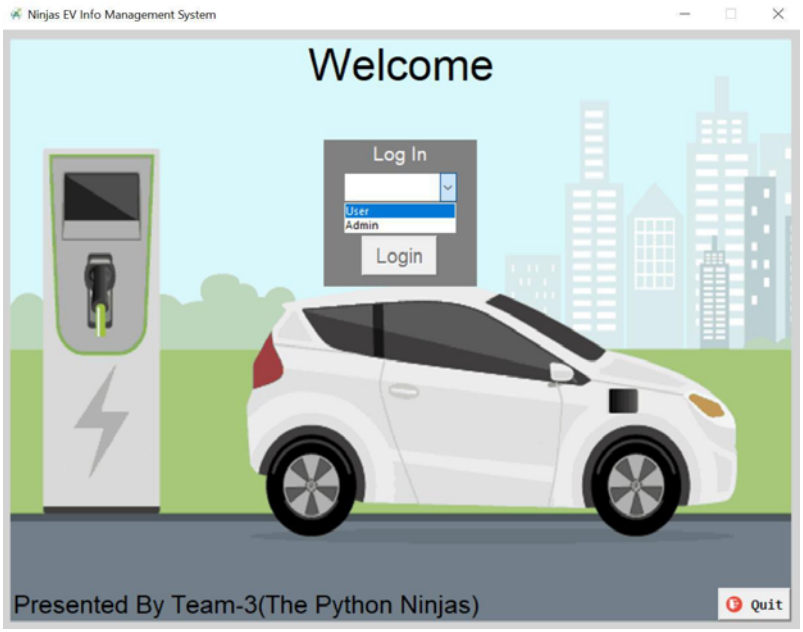


Figure : Login User Type

Password is required to be login as an admin user type.

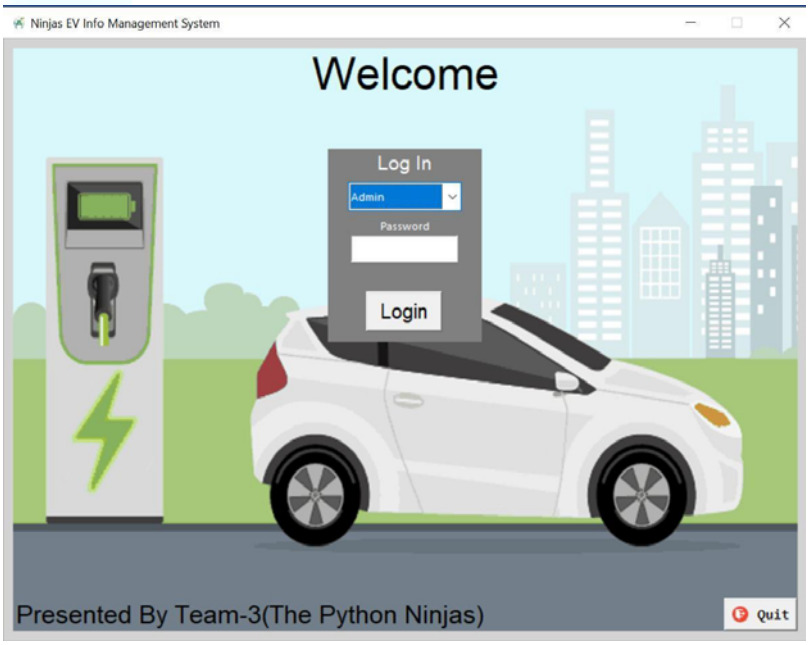


Figure: Admin Login Form

After admin login successfully, home form will be displayed.

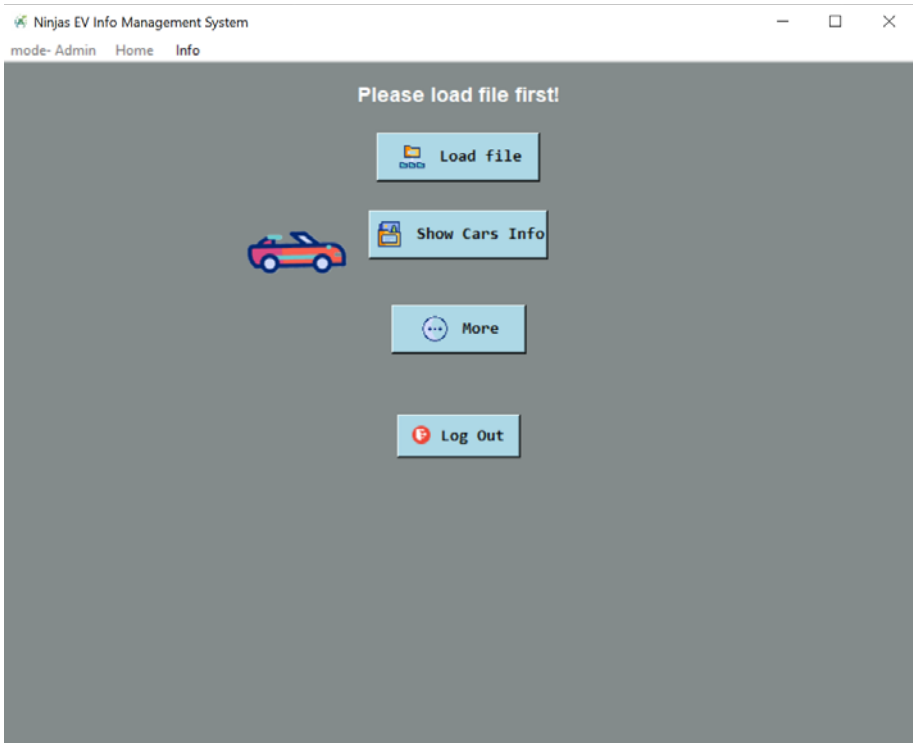


Figure : Home Form

Admin need to upload the car information csv file to input the data into the system. Click 'Load file' button from home form and select the evcars_subset.csv file.

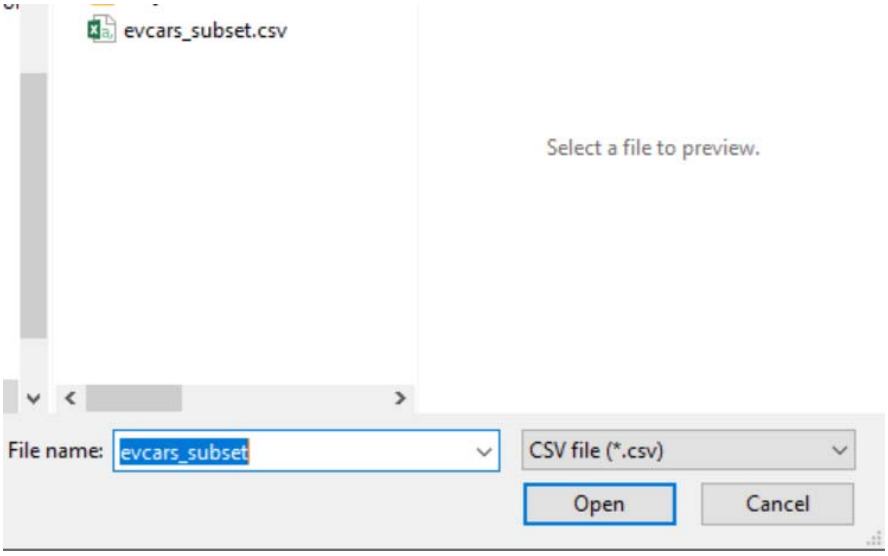


Figure: Upload csv file

After the file is uploaded, 'evcars_subset.csv is loaded' message will be shown.

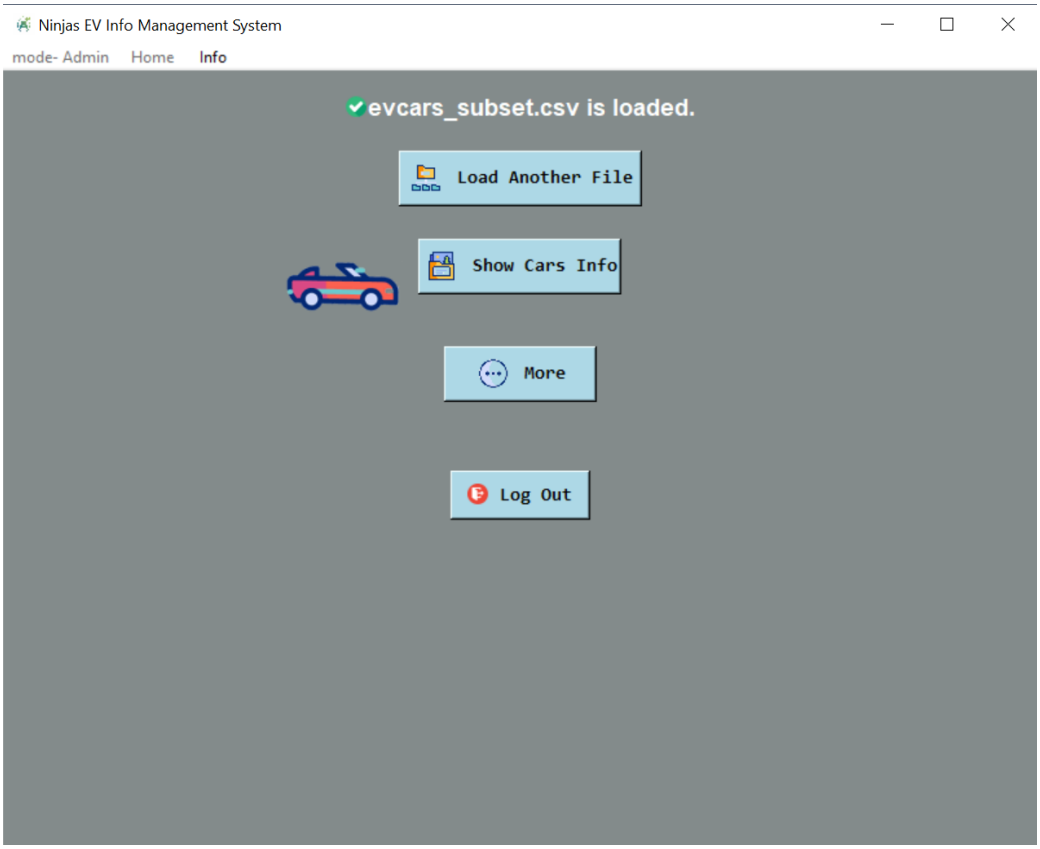


Figure: Uploaded CSV file

Admin click 'Show Cars Info' button from home page then all the information that are uploaded from the csv files will be displayed.

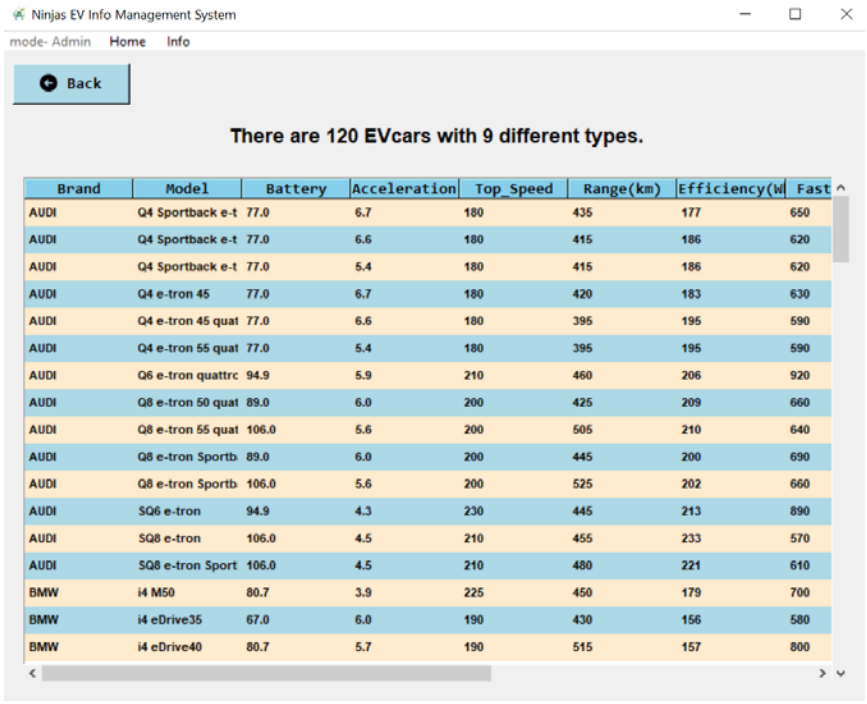


Figure: Show car list form

Admin user click 'More' button in home form, the following form will be displayed to search, add, update and delete functions are available to process.

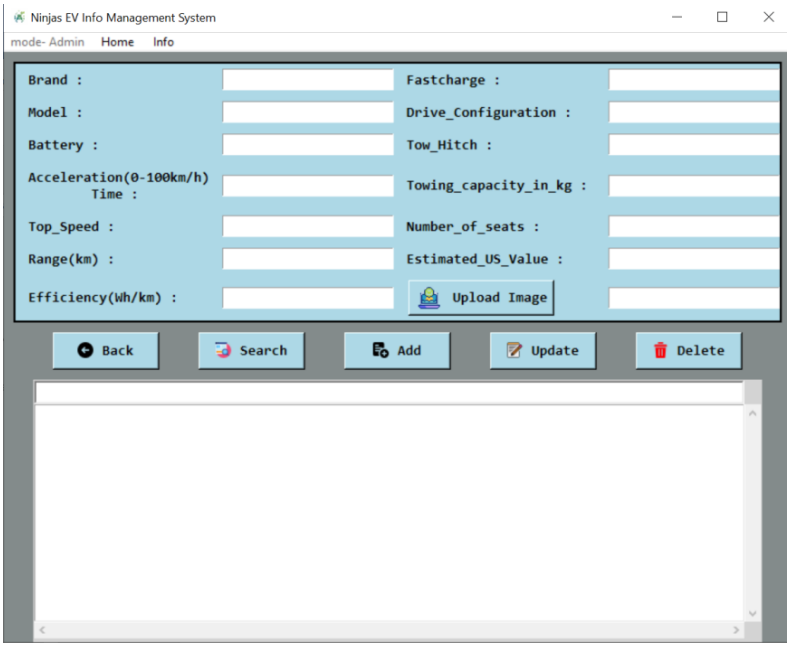


Figure: More function form

Admin provide data and click ‘Search’ button, the related information will be displayed.

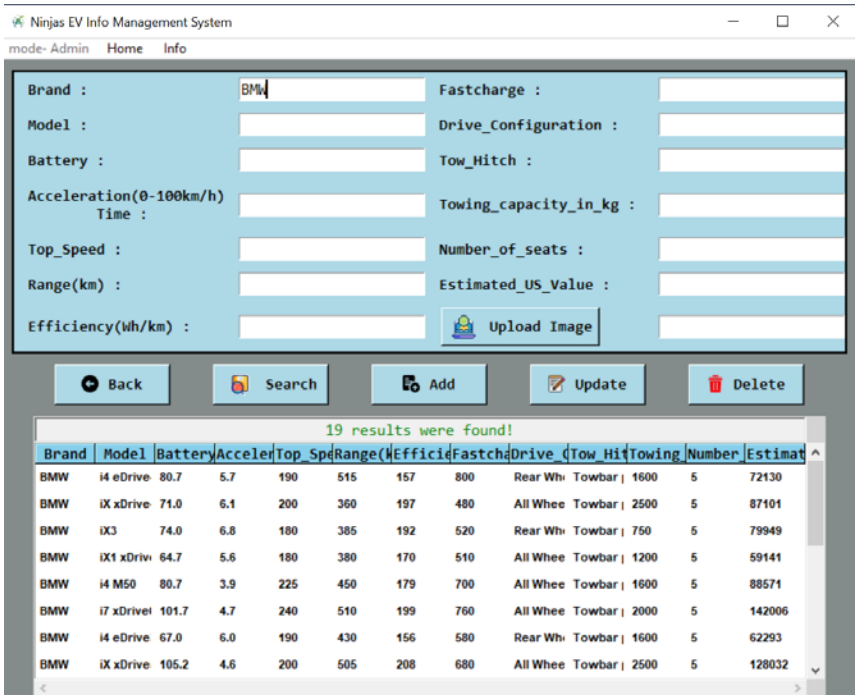


Figure: Search Form

Admin input the data to add to the system and click ‘Add’ button, then the record will be inserted to the system.

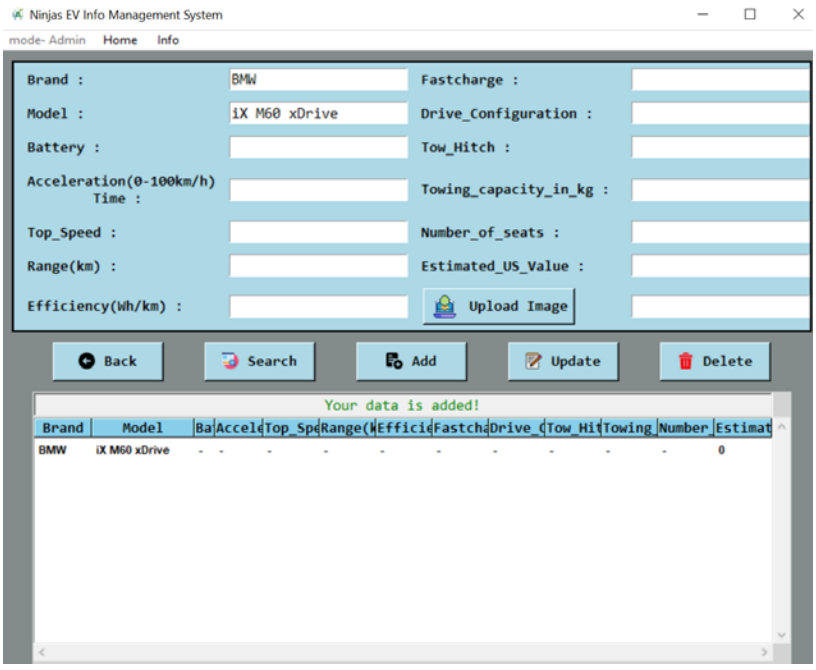


Figure: Add Form

Admin search data, select record and click ‘Update’, then update form will be displayed. The user input information will be saved after confirming changes when click ‘Save’ button.

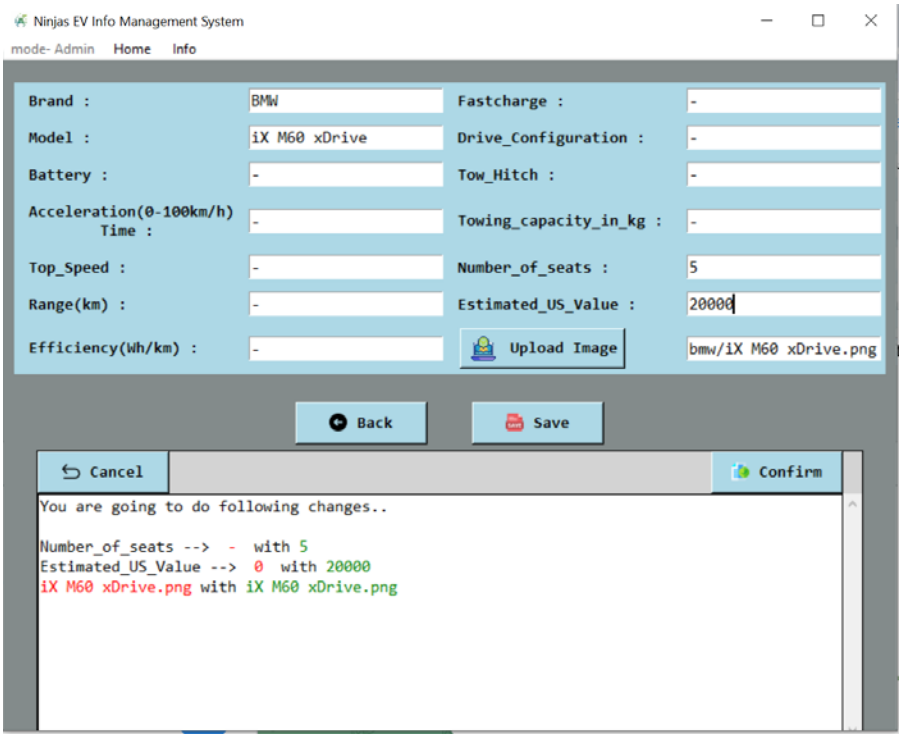


Figure: Update form

Figure: The updated record

Admin can delete the existing record from the system by selecting record from the list then clicking 'Delete' button from More function form.

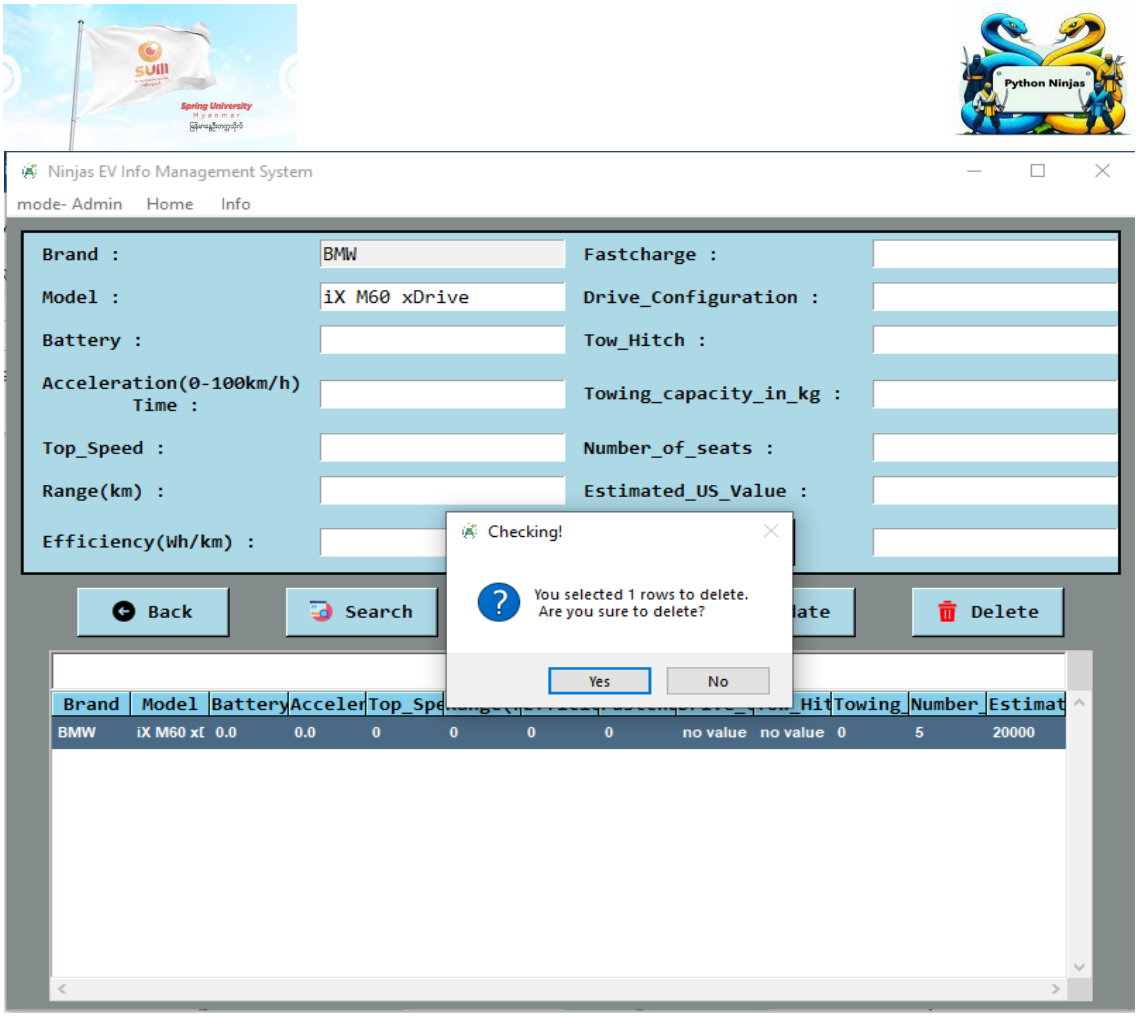


Figure: Delete form

Figure: Deleted Record

Click ‘Info’ menu and car brand icons will be displayed and click the icon to see the car lists of each brand.

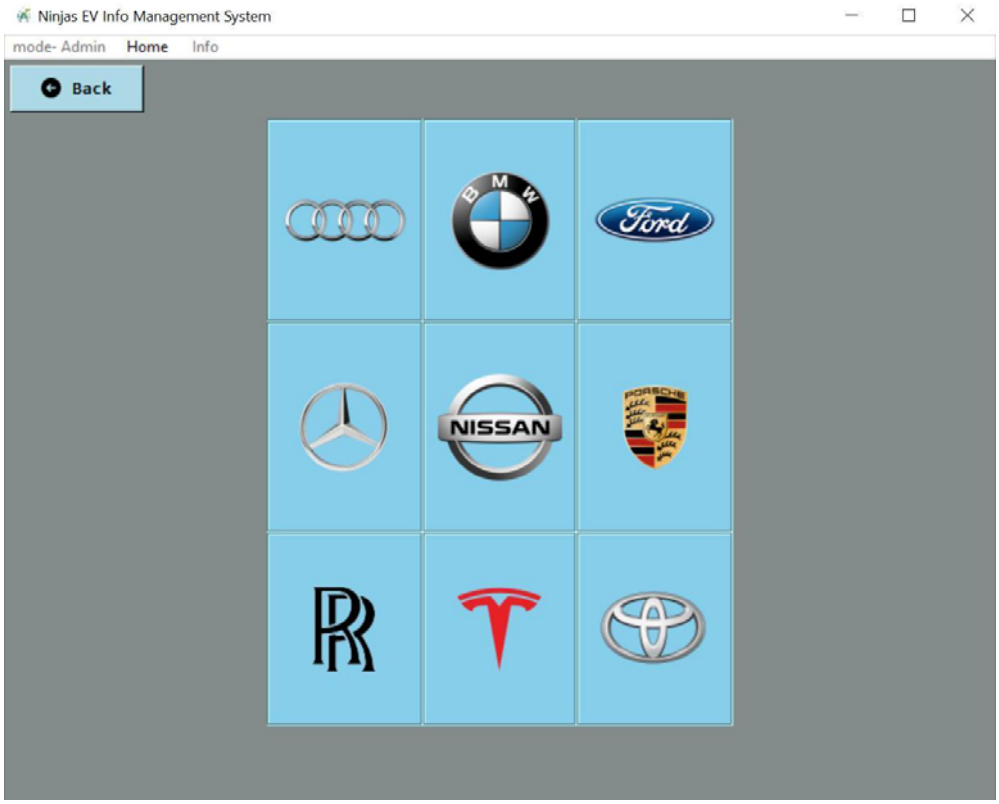


Figure: Show Car brand

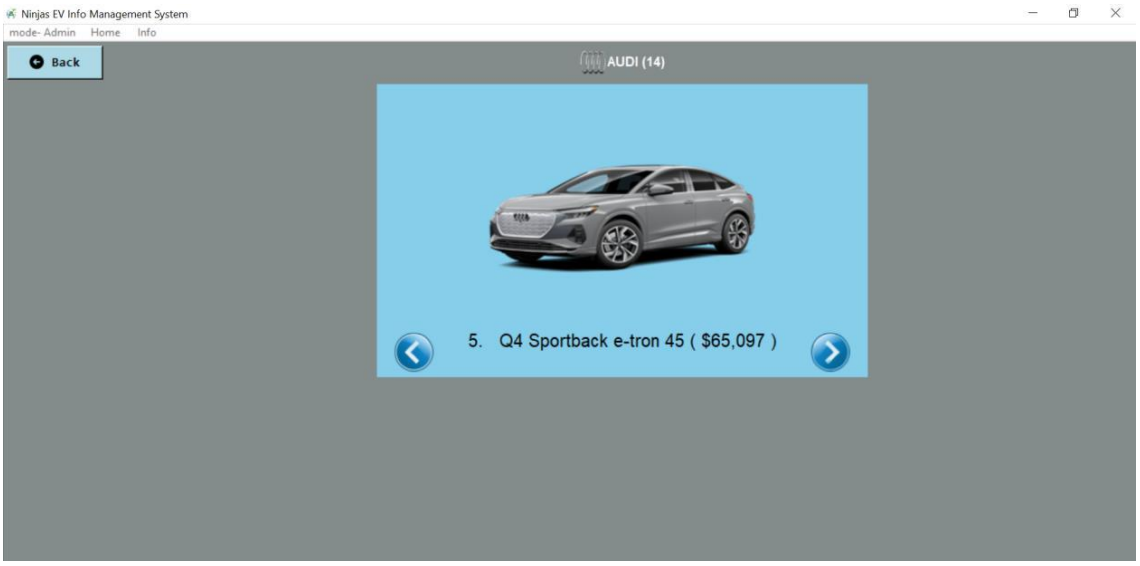


Figure: Show selected car brand

When click the individual car image, the detail specifications will be displayed.

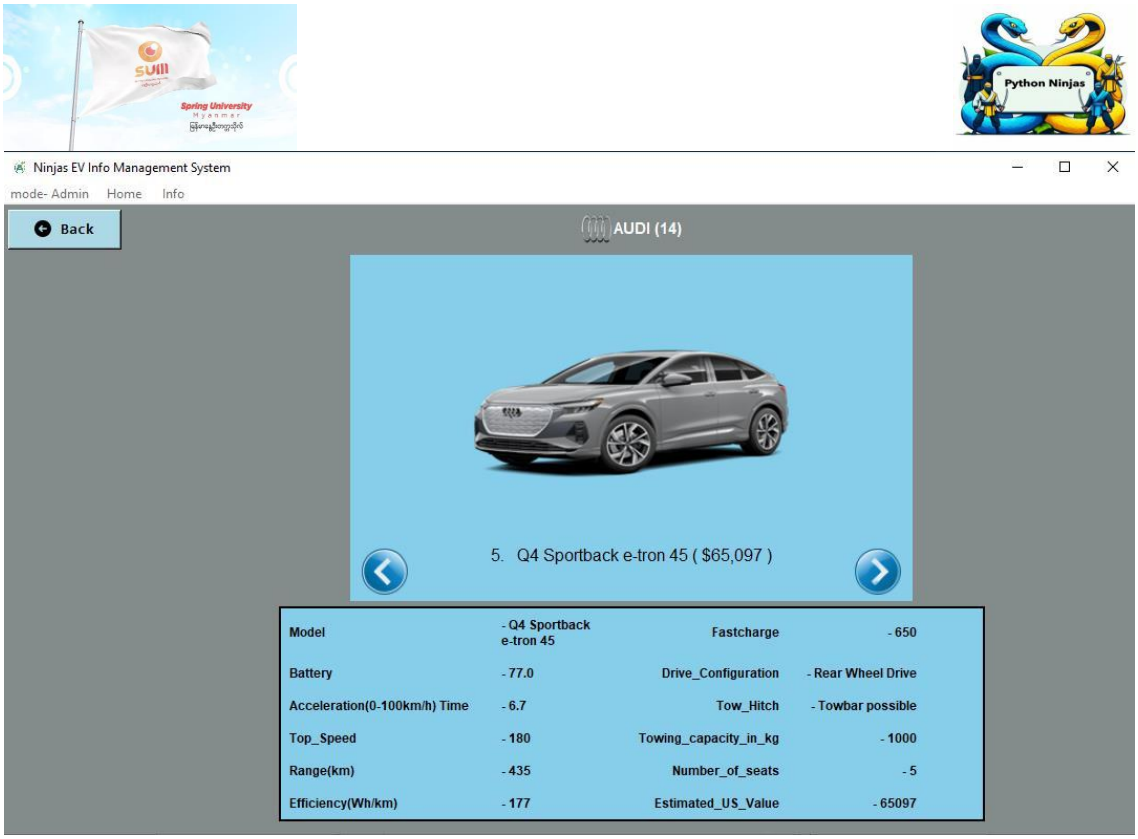
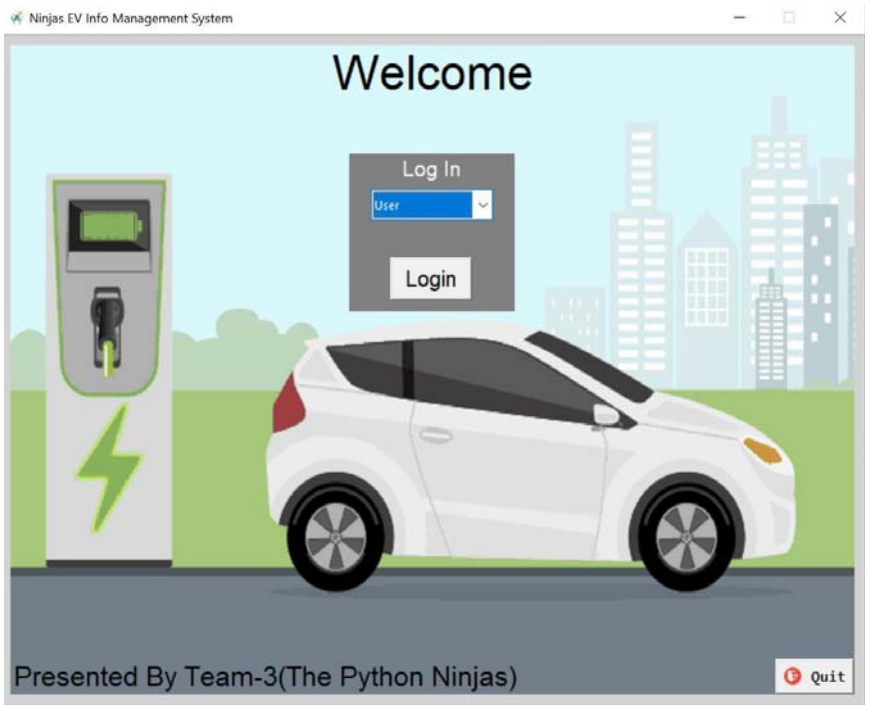


Figure: Show selected car brand detail

The normal user is not required to provide password to login and select ‘User’ and click ‘Login’ button from welcome form.



User Home form will have Show Cars Info, More and log out functions. Show cars Info and Log out will be same as Admin function.

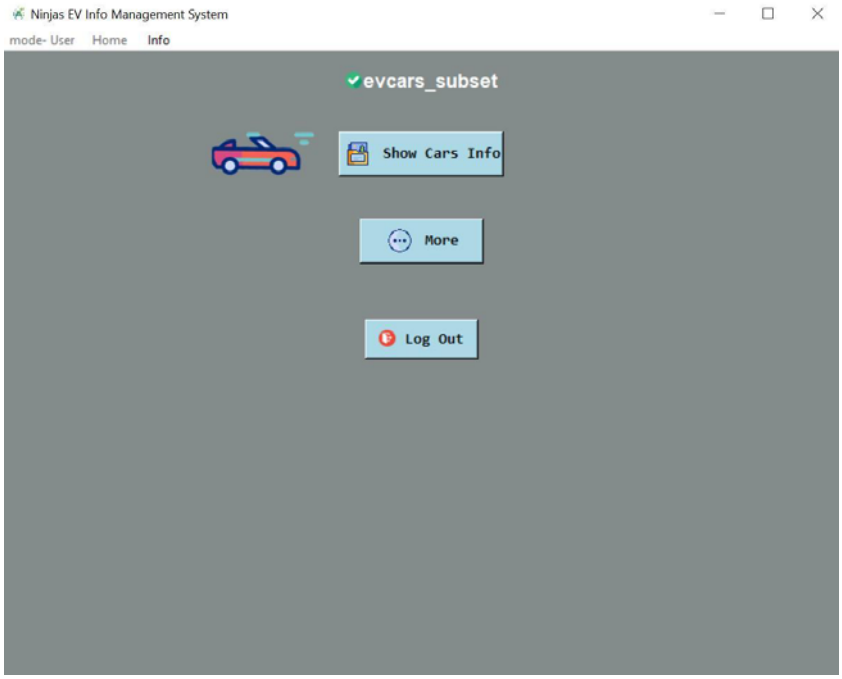


Figure : User Home Form

When click 'More' function, user able to see Search function. Input data to search and click 'Search' button, then will display the information if it is existed in the system, otherwise will how the error message.

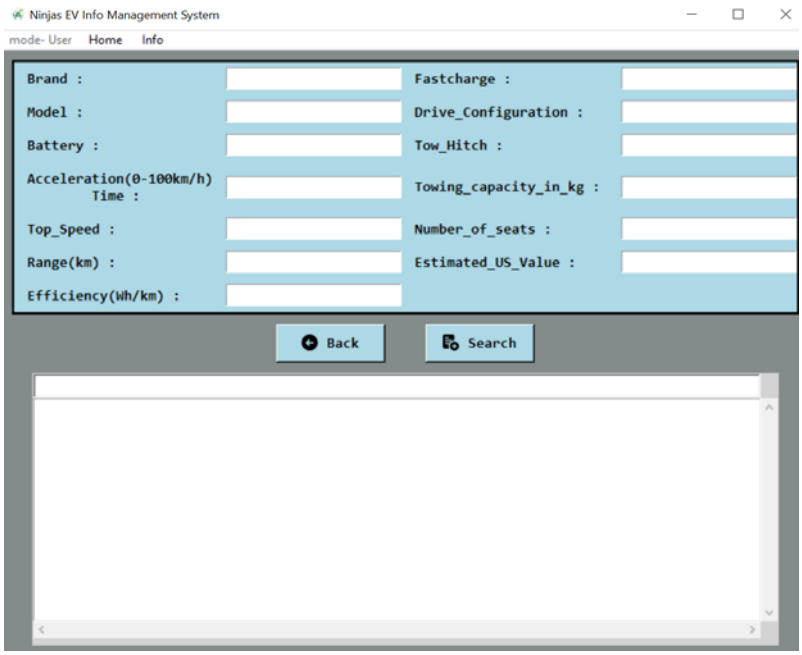


Figure: Search Form

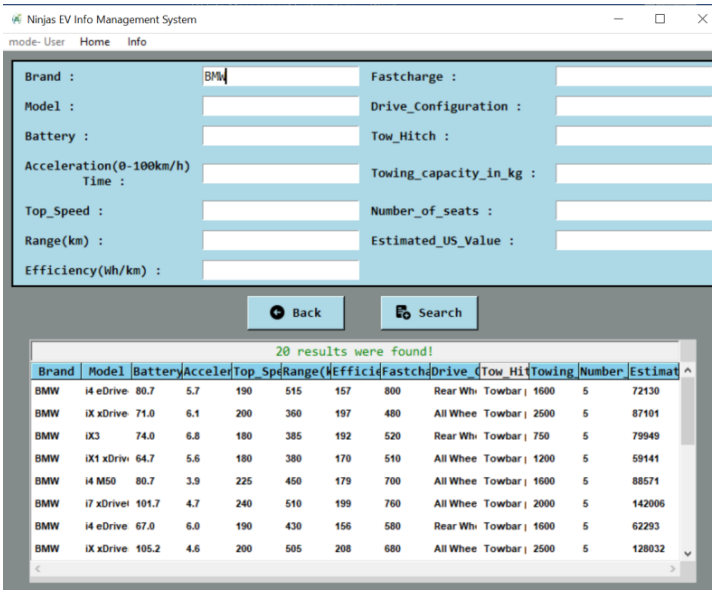


Figure : Display search result

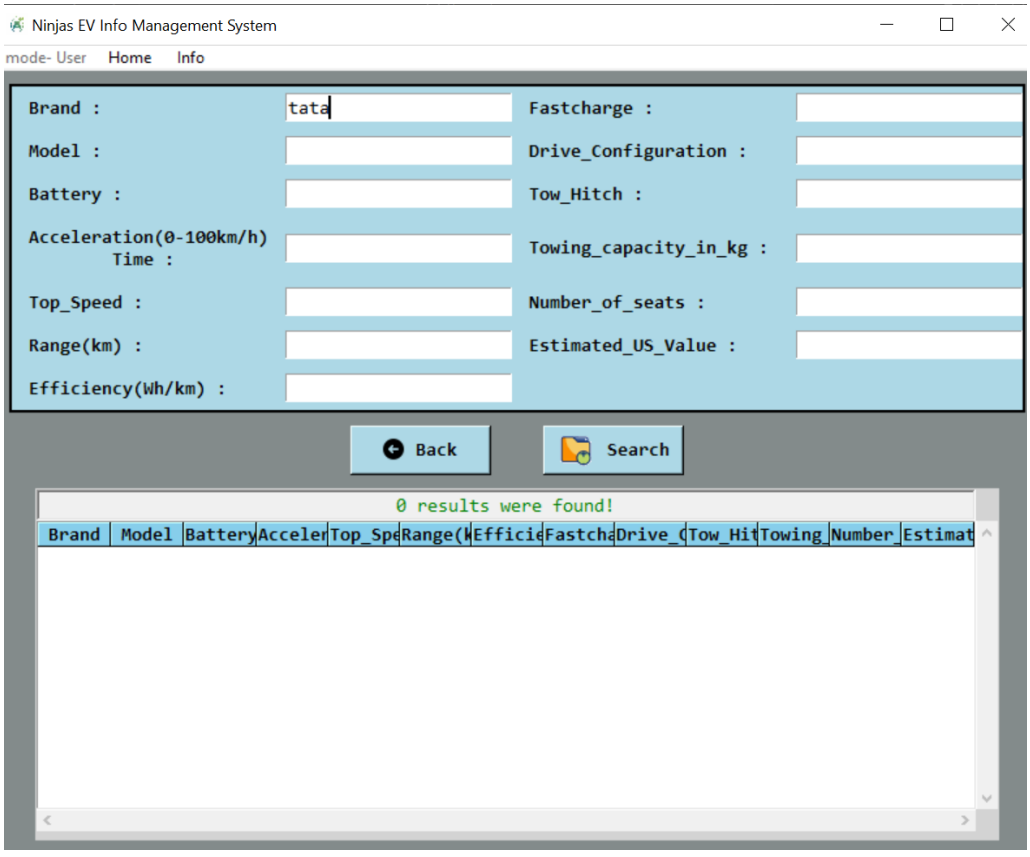


Figure : Display No data result

Info menu is same function as Admin, able to see car brand icon and click respective icon to see each brand car list and detail specification.

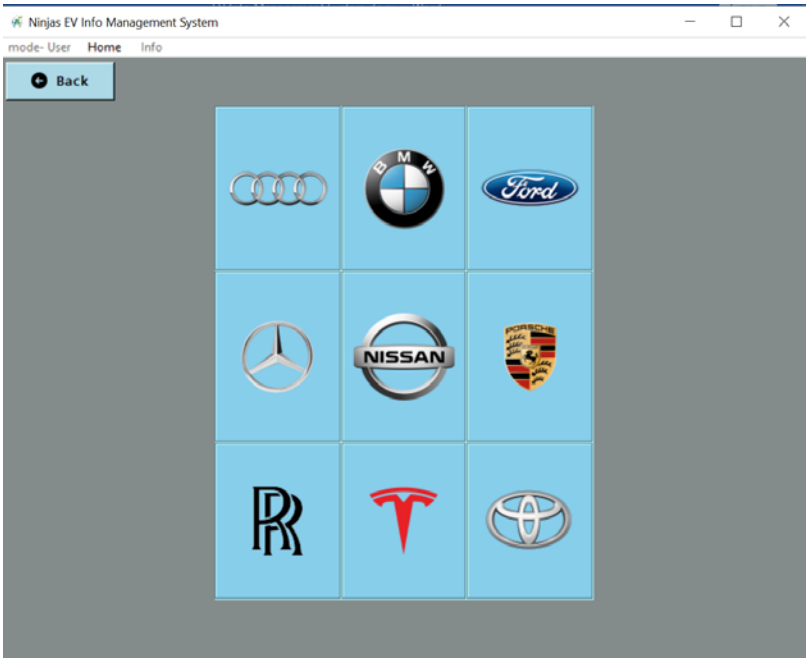


Figure : Info menu

To logout from the system, admin or user click 'Log Out' from home page. Then the welcome form will be displayed to leave from the system. Click 'Quit' button to stop using the system.

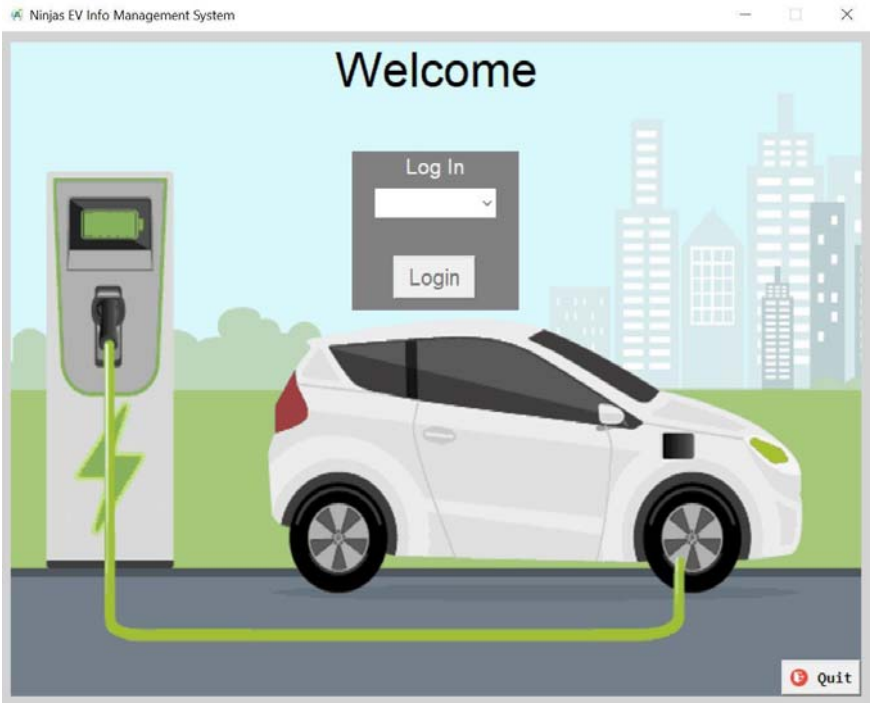


Figure : Quit form

Chapter (4)

Conclusion

EV Info Management System is easy to use for end user. The admin user who is required to login by providing valid password has fulll acces, and normal user has limited to use few features in the system. Admin user is able to input vehicle information by loading csv file, add the new record for the car information, able to manipulate the existing record, remove from the system, search and view all the information that are existing in the system. In the meanwhile, normal user is not allowed to modify the records, but have accessible to search and view the vehicle information in the system. By using the EV Info Management system, the user can reduce manual work and increase productivity on daily processing.

Limitation and Future work of the system

In the existing system, the vehicle data is required to load from CSV file so it has limitation of the records like adding new columns, handling many columns. In the future, we are planning to use database in place to store and manage large amounts of data, to support a wide range of activities, including data storage, data analysis, and data management. When the system is implemented by using database, it will be more convenient and accessible for the system users. The Ninjas EV Cars Info Management System is well on its way to becoming a versatile and user-friendly application.

We are also planning to do following updates:

1. Compatibility:

    - Ensure that our code can handle various CSV files, not just the one(eg.evcars_subset.csv) specific to this project.
    - Make it adaptable to different car datasets and other datasets for images, etc.
    - Manage charging station data (locations, availability, charging rates).

2. Compatibility:

    - Ensure that our code can handle various CSV files, not just the one(eg.evcars_subset.csv) specific to this project.

- Make it adaptable to different car datasets and other datasets for images, etc.
- Manage charging station data (locations, availability, charging rates).

3. User-Friendly Features:
    - Implement features like data reverting (undo changes) for admin
    - filtering based on conditions
    - tracking charging station locations, showroom service centers
    - managing maintenance schedules (service dates, reminders)
    - downloading images
    - add cars to their favorites
    - Enhance the GUI by adding tooltips, progress bars
4. Image Display:
    - Fine-tune the image display using PIL (Pillow). Optimize resizing, cropping, and handling transparent backgrounds just in application
5. Event Handling:
    - Bind events effectively (using keyboard shortcuts or context menus) for better usability
    - redirecting car brand main website by mouse actions

Reference

[1] https://www.kaggle.com/datasets/vanillatyy1/electric-vehicle-dataset

https://electriccarstrade.com/index/evinfo

https://en.wikipedia.org/wiki/Flowchart