
Software Requirements Specification

for

Car Sale System

Version 1.0 approved

Prepared by

Melisa Kamacı – Begüm Akdeniz – İrem Tamay – Tamer Alkım Tokuç

TED University – CMPE 313

12 / 12 / 2021

Table Of Contents

1. Introduction	1
1.1 Purpose	1
1.2 Document Conventions	1
1.3 Intended Audience and Reading Suggestions	1
1.4 Product Scope	1
1.5 References	1
2. Overall Description	2
2.1 Product Perspective	2
2.2 Product Functions	3
2.3 User Classes and Characteristics	3
2.4 Operating Environment	4
2.5 Design and Implementation Constraints	4
2.6 User Documentation	4
2.7 Assumptions and Dependencies	4
3. External Interface Requirements	5
3.1 User Interfaces	5
3.1.1 Login UI	5
3.1.2 Main Menu UI	6
3.1.3 Create Record UI	6
3.1.4 View Records UI	7
3.1.5 View Inventory UI	7
3.1.6 Delete Record UI	8
3.1.7 Maintenance UI	9
3.1.8 Update Profile UI	9
3.1.9 Error UI	10
3.2 Hardware Interfaces	10
3.3 Software Interfaces	10
3.4 Communications Interfaces	10
4. System Features	11
4.1 Login to System	11
4.2 Logout from System	12
4.3 Update Profile	12
4.4 Create Sale Record	14
4.5 Create Buy Record	14
4.6 Delete Sell Record	16
4.7 Delete Buy Record	16
4.8 View Sell Record	17
4.9 View Buy Record	17
4.10 View The Cars In The Inventory	18
4.11 Verify Auto Maintenance	19
4.12 Schedule Maintenance	21
4.13 View The Maintenance Schedule	21
5. Other Non-functional Requirements	22
5.1 Performance Requirements	22
5.2 Safety Requirements	22
5.3 Security Requirements	22
5.4 Software Quality Attributes	22
5.5 Business Rules	23
6. Other Requirements	23
Appendix A: Glossary	23
Appendix B: Analysis Models	24
Appendix C: To Be Determined List	29

Revision History

1. Introduction

1.1 Purpose

The car sale system targets car dealerships of any size that would like to digitalize some procedures in car sales which will result in minimizing the manual work done by the dealership. This product will make it easier to manage records for salespeople, oversee the general process for managers, keep track of maintenance for mechanics, and calculate economic data for accountants. In other words, the product provides the tools to assist in automating the record-keeping, maintenance, and accounting processes that would otherwise have to be performed manually.

1.2 Document Conventions

The document will adopt size 12, Times font. The main headings will be size 18 and bold. Subheadings will be size 14 and bold. Emphasizing will be indicated with bold font. Each requirement will be stated on its own since there is no hierarchy between functionalities.

1.3 Intended Audience and Reading Suggestions

This document is intended for developers, project managers, testers, and documentation writers. The document contains an overall description of the project, requirements, features, and dependencies. For technical roles like developers and testers, it is recommended to start from the 1st header, Overall Description, to the end of the document. For other roles, it is recommended to read from 1st header to 5th header, Other Non-functional Requirements.

1.4 Product Scope

The product aims to contribute to new and second-hand car dealers' business flow. It will help to minimize wastepaper since every record-keeping and customer-end operation will be done digitally. This would help dealers to create a greener image. The program will help to keep track of sales and reduce human error. As a result of reduced error rates, better presentation of inventory, and faster processes, sales are expected to increase.

1.5 References

Use case diagrams in Appendix B are from the "Assignment-2-Use Case Diagrams of the Car Sale System" document, available [here](#). Other than that, no other external sources have been addressed.

2. Overall Description

2.1 Product Perspective

The overall system will be built from scratch, in other words, it will not be a replacement or improved version of an existing system. The following are the primary components of the project. The system's functionalities are carried out via the links between them. The interconnection of the stated sub-components is depicted in the diagram (Figure 2.1). Authorization, inventory management, car maintenance subsystems, and database makes up the program. Authorization is vital to access functionalities. After successful authorization, inventory management systems and car maintenance systems are accessible. Every subsystem is connected to the database since information is stored there.

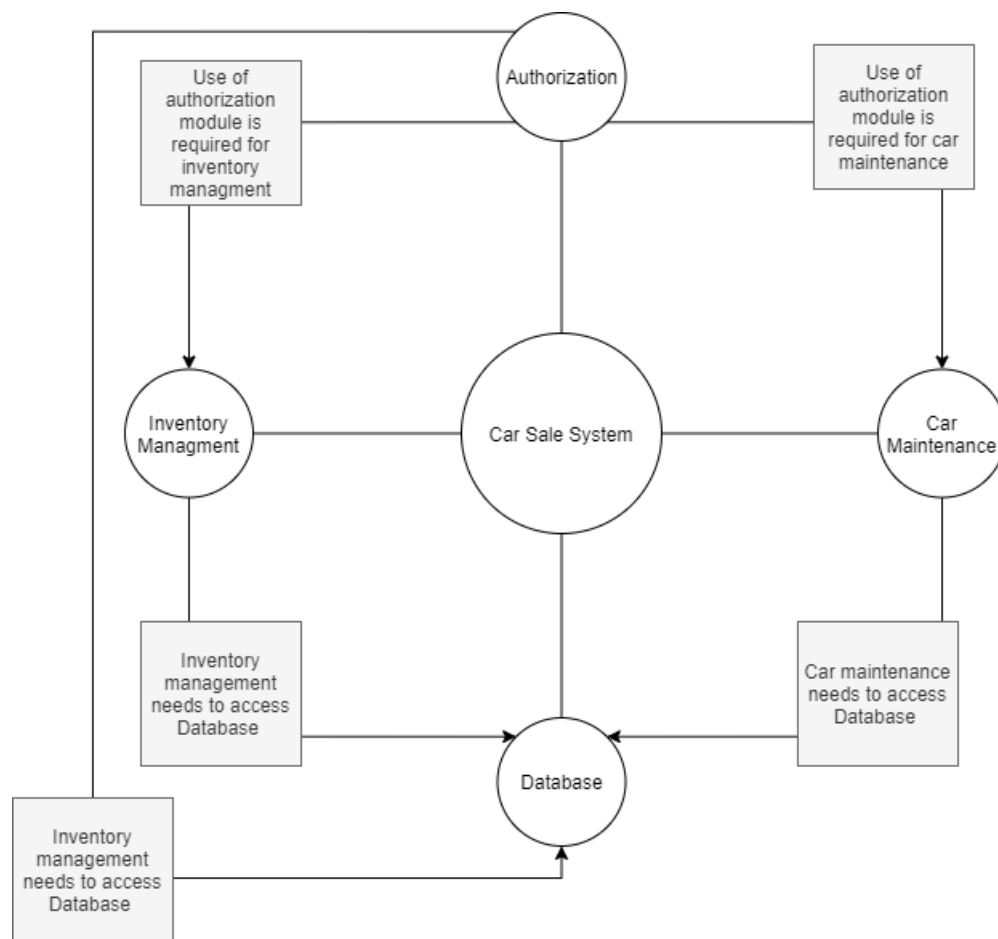


Figure 2.1: Top Level Data Flow Diagram

2.2 Product Functions

The system will have the following major functions:

- Salespeople can create sell records by indicating necessary details of the sale procedure.
- Salespeople can create buy records after the completion of buying process of the second-hand car from the seller.
- Salespeople and dealership managers can view the current stock by using certain filters.
- Mechanic staff who performed the maintenance of the car highlights this information on the system by modifying the schedule.

The primary actions and their associated flow of data is visualized in the top-level data flow diagram (Figure 2.2).

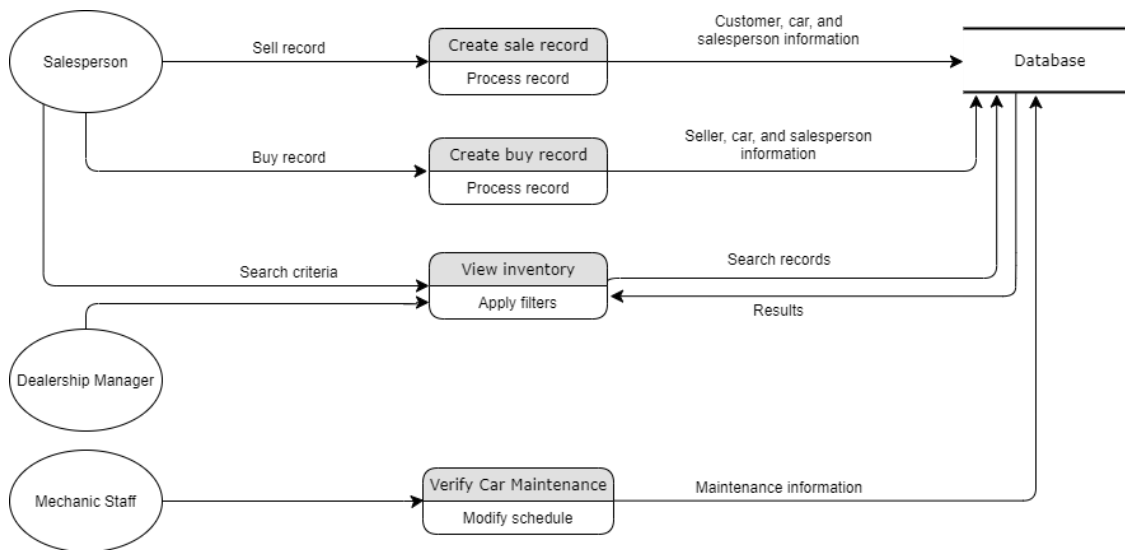


Figure 2.2: Top level data diagram

2.3 User Classes and Characteristics

Car dealership salespeople, managers, accountants, and mechanic staff with basic computer knowledge are the consumers of this product. The salespeople can be seen as the primary users of the system since their operations on the system has the biggest share and importance. The salespeople are expected to use the program daily hence they are required to be qualified and educated enough in terms of technology to use the system effectively. They are mainly responsible for performing the functions regarding inventory management. The dealership managers are the head of every operation performed in the dealership which makes them have the biggest privilege. Hence, they need access to the system weekly to control the taken actions. The dealership managers are assumed to be experienced in computer usage well enough to get informed about the performed functions. The accountants have a need to use the system monthly to carry on their monthly financial calculations easily. Their privilege levels are limited to just getting the required information by viewing sale and buy records. They are assumed to have a basic understanding of computer technologies. The mechanic staff who are responsible for performing the maintenance on cars indicate this information whenever needed. They're not expected to use the program daily however the frequency of use depends on the schedule of the maintenance. They're only allowed to use the maintenance subsystem other than authorization, hence they are not required to have advanced computer skills.

2.4 Operating Environment

The system will be operated on the computers which are being used in the dealership. The required operating system will be minimum Windows 7, OS X 10.10, or Ubuntu 16.04 for the client-side and Windows Server 2012 for the server-side. For the client-side, 256 MB free space, 126 MB available RAM, and Intel Pentium 4 or equivalent CPU are minimum requirements. For server-side, 32 GB RAM, Intel Xeon E7520, or equivalent processor, 126 GB free space is needed however, 500 GB is advised for large-scaled dealerships. The program will not be seen as a threat by antivirus programs that are installed on the computer being used.

2.5 Design and Implementation Constraints

Java will be used in the development process of the project. Dealership-related records and user information will be kept using the PostgreSQL database, hence necessary libraries will be implemented. Regulatory bodies require that the developers should comply with local personal data protection constraints. They also should apply database safety practices. The server must be isolated, meaning that any device which is not connected to the LAN of the server cannot access it. Every login detail of consumers is required to be encrypted using the SHA-256 algorithm. For timing, a report to be delivered to the user must be provided biweekly, a prototype must be available after 2 months, finally, the full-fledged program must be provided after 3 months. After delivery, maintenance will be done once every 3 months which will keep the server updated and functioning. Database backups are optional but highly recommended. If desired, backups will be created on a cloud or physical server. No further updates on software are planned other than bug fixes.

2.6 User Documentation

Alongside the system, user documentation will be delivered to the end-users of the program. This user documentation includes tutorials for each subsection to give assistance. To provide a healthy maintenance process, the intercommunication between the help centre and the system users will be preserved by guidance through online help. The user documentation will be handed over Product / User Manual format which covers the product's basic functionalities with illustrations.

2.7 Assumptions and Dependencies

The project demands JRE to be installed on the computer since it will run on Java. To ensure that the connection between computers and the server is uninterrupted, a high-speed internet connection must be available. It is assumed that every machine in the system fulfils the technical requirements. PostgreSQL will be used on the server and the system will be built accordingly. If the dealership decides to move to a different server system, it is crucial to notice developers at least 1 month in advance to modify the program accordingly and to migrate previous data to the new database. Failure to do so may cause complete loss of data. As a security precaution, the server may shut itself down in case of a spike in connected device count. In case of shut down, it is expected to give a notice to developers to deter the attack and reboot the server. Users are expected to write a bug report if they encounter a problem. Every report will be read thoroughly and will be worked on to fix the problem. Misuse of this procedure will damage both user and developer side operations hence it may be requested to pay an additional fee. While developing the system, no past components will be reused.

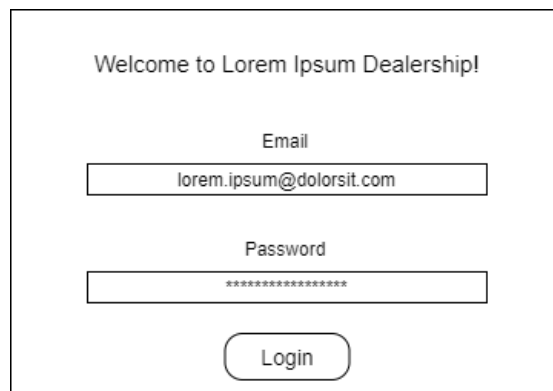
3. External Interface Requirements

3.1 User Interfaces

A minimalistic and clean GUI will be used throughout the system to ensure correct operation. Buttons will have a rounded rectangle shape. Textboxes will have a rectangle shape. Subsections will be enclosed in borders with rectangle shapes. Selection will be indicated with blue color. Every page, except the login page, will be full screen, adapting to the size and resolution of the screen. The size of the login box will be 500x700 px, without resizing option.

3.1.1 Login UI

Login interface will be presented in every run of the program. In the login interface, 2 textboxes with “Email” and “Password” written on top will be displayed in every login case (Figure 3.1). After providing the requested information, the login button on the bottom will be active. If login credentials match a record in the database, the main menu will appear. Otherwise, the “Wrong Email or Password” prompt will appear and require email and password again. The program will check whether if the email is in the correct form. If not, a warning will be given to the screen. If desired, provided email server will be prefilled in an email textbox, requiring only an email username.



The image shows a login interface for 'Lorem Ipsum Dealership'. It features a title 'Welcome to Lorem Ipsum Dealership!' at the top. Below the title are two text input fields. The first field is labeled 'Email' and contains the text 'lorem.ipsum@dolorsit.com'. The second field is labeled 'Password' and contains a series of asterisks '*****'. At the bottom of the form is a rounded rectangular button labeled 'Login'.

Figure 3.1: Login UI

3.1.2 Main Menu UI

In the main user interface, a greeting will be on the top left, changing between “Good Morning” and “Good Afternoon” according to the time of the day. A date and time text will be presented to users to provide easiness in operations. In the middle, there will be buttons to choose between operations (Figure 3.2). For example, salesperson UI is shown, but buttons change according to user privileges. Selecting these buttons will open operations available under that group. On the top left, a logout and update profile button will exist.

Figure 3.2: UI for salespeople

3.1.3 Create Record UI

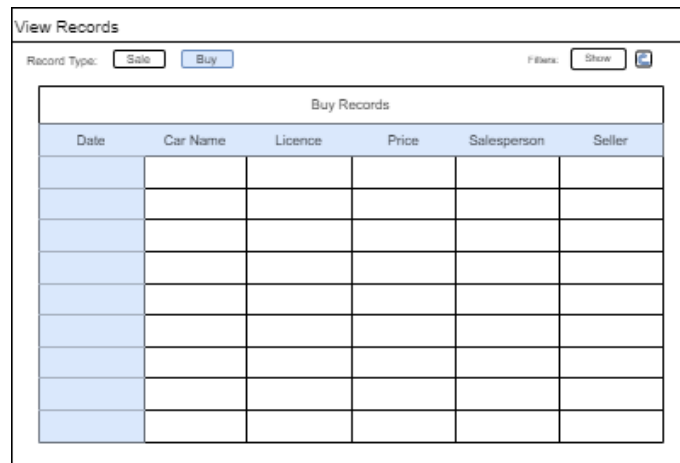
If the user with the appropriate access level selects “Create Record” on the main menu, create record menu will appear. In this menu, the user will be able to choose between creating a buy or sale record on the top left of the screen. This menu requires different information depending on the previous selection. The sell record menu (Figure 3.3.1) requires buyer information, license of car, price, and chosen payment method. The buy record menu (Figure 3.3.2) requires seller information, car information, and price.

Figure 3.3.1: Sale Record UI. *1 Part will be accessible only if the instalment option has been selected.

Figure 3.3.2: Buy Record UI

3.1.4 View Records UI

Salespeople, managers, and accountants can access this UI when selected. This section will provide records row by row, after selecting which kind of record would like to be seen, the related record page will open. Buy and sale record UI (Figure 3.4) will be similar in terms of look. On the top right, there will be two buttons for opening and clearing filters.

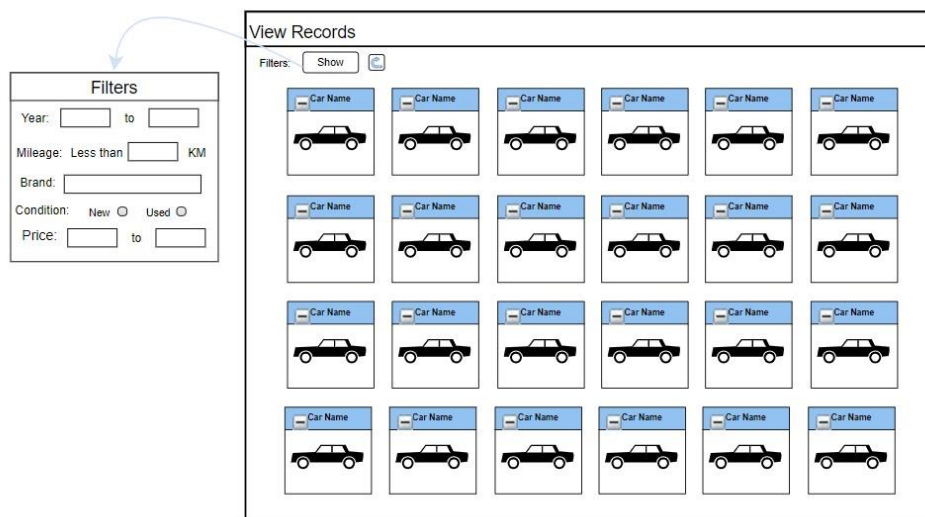


The 'View Records' window features a 'Record Type' section with 'Sale' and 'Buy' buttons. The 'Buy' button is selected. To the right, there is a 'Filters' button and a small icon. Below this is a table titled 'Buy Records' with the following columns: Date, Car Name, Licence, Price, Salesperson, and Seller. The table contains 8 empty rows.

Figure 3.4: UI for viewing buy records

3.1.5 View Inventory UI

The salesperson will show the inventory to customers and the manager will track inventory and give request new vehicles if inventory is not sufficient. There will be pictures of cars, with basic information underneath (Figure 3.5). Pictures will act as buttons and when clicked, it will provide more information. Filtering options will be on the top left and open the filters menu when clicked on it (Figure 3.5).



The 'View Records' window includes a 'Filters' button and a small icon. To the left, a separate 'Filters' panel is shown, which is linked to the main window by a blue arrow. The 'Filters' panel contains the following options:

- Year: [] to []
- Mileage: Less than [] KM
- Brand: []
- Condition: New ☐ Used ☐
- Price: [] to []

 The main 'View Records' window displays a grid of 24 car icons arranged in 4 rows and 6 columns. Each icon is labeled 'Car Name' at the top.

Figure 3.5: UI for viewing records and filtering options

3.1.6 Delete Record UI

Some situations may require record deletion. To delete records, salespeople must specify the type of record and license of the car. License and record type would be sufficient to delete the record since only one record will exist under license and every car license is unique. In UI, there will be a textbox to enter license in the specified format and choose to delete whether buy or sale record (Figure 3.6). If the specified record does not exist, a warning popup will appear.



The image shows a window titled "Delete Record". Inside the window, there is a label "Please enter licence" above a single-line text input field. Below the input field is a label "Please choose record type". Under this label are two buttons: "Buy" and "Sale". At the bottom of the window is a button labeled "Delete".

Figure 3.6: UI for deleting record

3.1.7 Maintenance UI

To help mechanic staff, a maintenance scheduling system will be implemented. In UI, there will be 3 buttons to choose from (Figure 3.7.1). Verify maintenance button will open the menu with the license textbox and verify button (Figure 3.7.2). This will update the last maintenance attribute of the specified car. The schedule maintenance menu will have 2 sections. One will list the cars which need maintenance, other will have textboxes that requests car license and scheduled maintenance time (Figure 3.7.3). Lastly, the view schedule will present a table that has planned operations weekly (Figure 3.7.4).

Figure 3.7.1: Maintenance main menu

Figure 3.7.2: Verify menu

Figure 3.7.3: Scheduling menu

Figure 3.7.4: View schedule

3.1.8 Update Profile UI

If a user desires to change their email or password, after they enter the main menu, they need to select the edit profile button on the left top (Figure 3.2). After they enter new details, the system will request the current password if the e-mail has changed and an old password if the password has changed. The UI will have a switch and a textbox to enter new details (Figure 3.9). After entering, another dialog box will open for the password.

Figure 3.9: UI for detail update

3.1.9 Error UI

In any case of error, an error dialog box will appear on the screen. A red diamond with an exclamation sign shape will appear on the same line with the “Error” text. Underneath them, an error description will be provided to notify the user about the error type. On the bottom right, a red button with “OK” written on it will be used to close the message. Error situations include network connection problems, database failure, input invalidity, and other software glitches. Figure 3.10 indicates an error caused by the loss of internet connection.

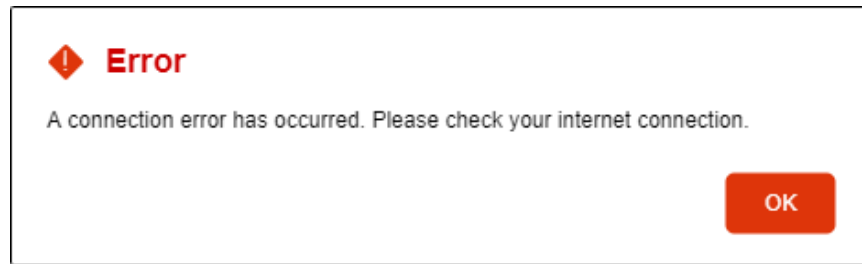


Figure 3.10: Example Error UI

3.2 Hardware Interfaces

The system requires an ethernet connection, so an ethernet cable will serve as a hardware interface. The program does not have any external hardware dependency, so there will not be any hardware interface other than that.

3.3 Software Interfaces

Interaction between the program and OS will be done by JRE since the program will be built using Java. Since the database technology of the program will be PostgreSQL, necessary Java packages will be used. The program will send SQL queries to execute operations.

3.4 Communications Interfaces

The software will communicate with the server via ethernet connection protocol. Login details will be delivered encrypted. Encryption will be done by the program using the latest encryption algorithms, specifically SHA-256 hashing. Other details will not be encrypted. The server will be isolated, so security breach risk is minimized.

4. System Features

4.1 Login to System

4.1.1 Description and Priority

This functionality is necessary to access the system to carry out the daily operations to be performed. User has to enter their login credentials such as company e-mail and password via the user interface. This is a high-priority task since it is the access point to the system and a security flaw in this point could cause information leaks. This service is expected to be utilized by the dealership staff every working day. The completion of this service is expected to be completed in at most 1 second. Figure 4.1 visualizes the subsequent steps to be performed as a sequence diagram.

4.1.2 Stimulus/Response Sequences

This feature will be initiated each time a dealership staff tries to authorize themselves via selecting the “Login” button that is placed on the opening screen. If entered password and e-mail match the data returned from the database, a login successful message will be directed to the screen, and the user will be able to use the program. If a mismatch occurs, a login unsuccessful message will be displayed to the user to ensure the second attempt.

4.1.3 Functional Requirements

REQ-1: The system shall request an e-mail and password from the user.

REQ-2: The system shall fetch login details from the database to check the validity of the login attempt.

REQ-3: According to the success status of the access request, a proper response will be generated.

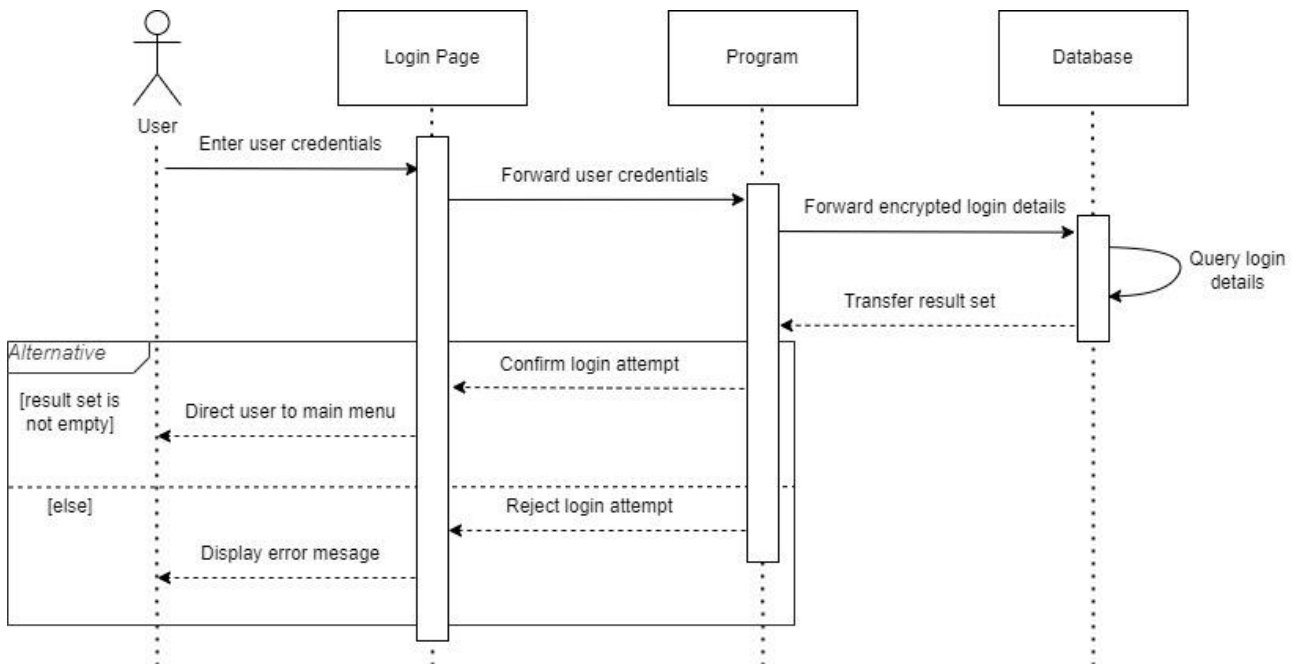


Figure 4.1: Sequence Diagram for Authorization

4.2 Logout from System

4.2.1 Description and Priority

To ensure safety, every user is expected to log out from the system if the services of the program are no longer needed. This function can be classified as a high-priority feature since it makes it possible to exit from the system and avoid unauthorized operations. The usage rate of the system will be parallel to the login counts. This service is expected to take no more than 1 second to complete.

4.2.2 Stimulus/Response Sequences

This functionality is issued when the user who had access to the system requests to log out from the system by clicking the “Logout” button that can be found under the profile icon. The system will always provide a positive response by making the user exit the system since it does not require any authorization or anything additional.

4.2.3 Functional Requirements

REQ-1: The system shall log the user out from the program when it is commanded so.

4.3 Update Profile

4.3.1 Description and Priority

This functionality can be used for changing any user credentials that are present on the profile when needed. This can be regarded as a low-priority task since it does not carry big importance on the general operations performed on the system. Modifiable attributes include passwords and e-mail. The rate of usage is not possible to predict as it totally depends on the preferences of the users over time. After the user enters the necessary fields, backend operations should take no longer than 1 second. Figure 4.2 visualizes the subsequent steps to be performed as a sequence diagram.

4.3.2 Stimulus/Response Sequences

This feature will be triggered by the user who desires to update their current data by choosing the “Update Profile” option in their profile. As the reaction, replacement of the previous data with new preferred data will be made.

4.3.3 Functional Requirements

REQ-1: The system shall ask for the part which the user wishes to reshape.

REQ-2: The system shall request new data for the specified section.

REQ-3: The system shall update the section with newly entered data on the database.

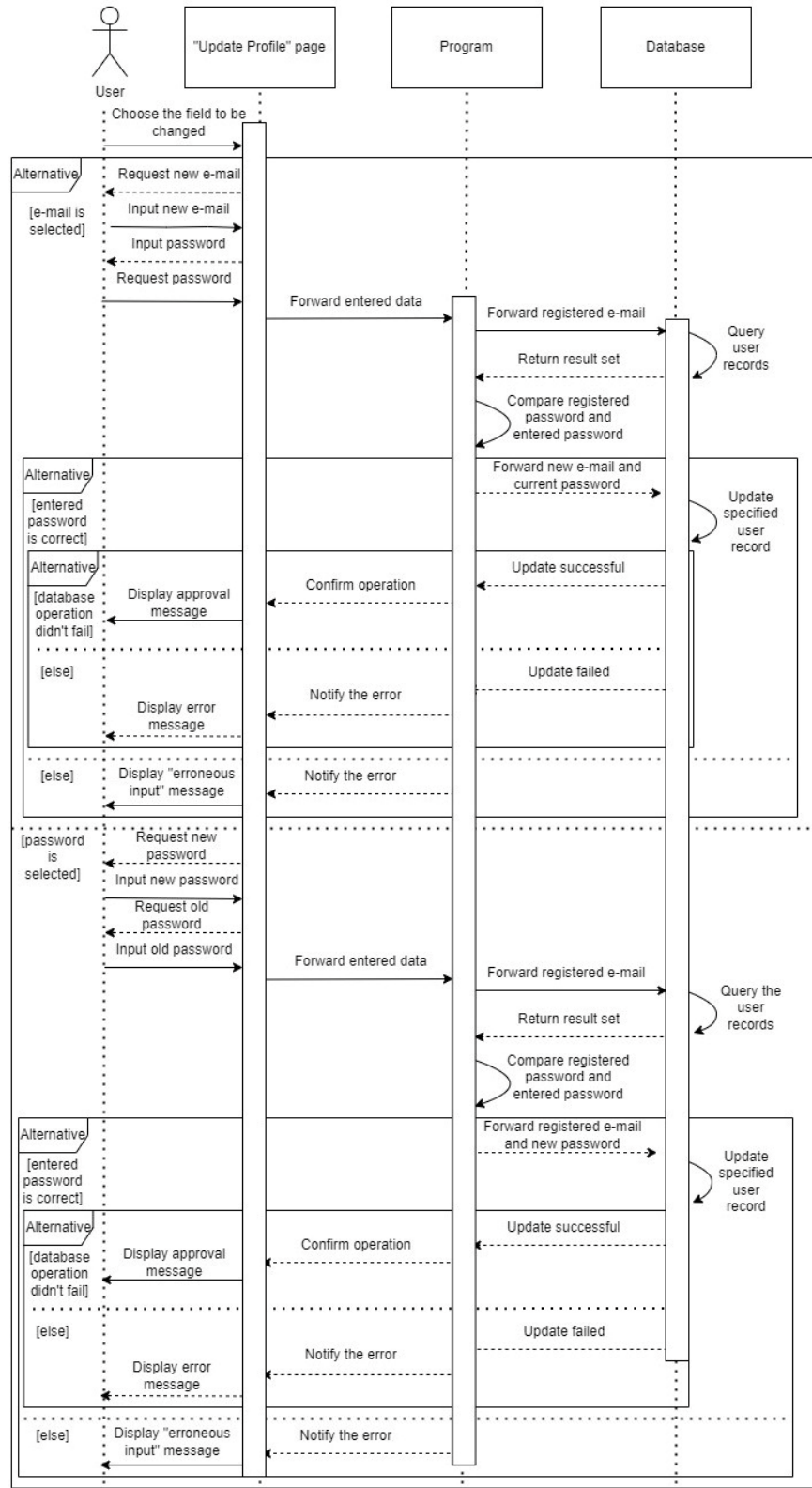


Figure 4.2: Sequence Diagram for Update Profile

4.4 Create Sale Record

4.4.1 Description and Priority

This feature represents the digitalized form of performed sale processes. Therefore, it is intended to include all related details about the agreed deal. As it is one of the main aspects of the system, it holds a high priority. The records of all occurred sales are kept in the system database to be accessed later. The anticipated frequency of use will match the sell rate of the company. It is assumed that this operation will not take more than 1 second to complete. Figure 4.3 depicts the next steps to be taken with time as a sequence diagram.

4.4.2 Stimulus/Response Sequences

This feature will be initiated by salespeople who have mediated the customer to purchase a car from the inventory via choosing the “Add Sale Record” option, after selecting “Add Record” from the main menu. The answer of the system will be to save the entered data to the database as a sell record.

4.4.3 Functional Requirements

REQ-1: The system shall ask for the information of the customer, car, and salesperson that had a role in the sell process.

REQ-2: The system shall add the newly created record to the database as a sell record.

4.5 Create Buy Record

4.5.1 Description and Priority

This feature represents the digitalized version of the deals made between a second-hand car seller and the dealership itself. Therefore, it is intended to include all related details about the purchase of the car. It has a high priority because it is one of the most essential parts of the business. The records of all bought cars are kept in the system database along with their details to be accessed later. The estimated density of use will correspond to the company's purchase rate. This process is expected to take no more than 1 second to execute. Figure 4.3 depicts the next steps to be taken with time for this functionality as a sequence diagram.

4.5.2 Stimulus/Response Sequences

Salespeople who arranged the purchase of a second-hand car from its owner will activate this option via choosing the “Add Buy Record” option from the menu, after selecting “Add Record” in main menu. The programs’ reply will be to save all acquired data as a buy record in the database.

4.5.3 Functional Requirements

REQ-1: The system shall request details about the seller, car, and salesperson that were involved in the purchase transaction.

REQ-2: The system shall update the database by adding the newly created buy record.

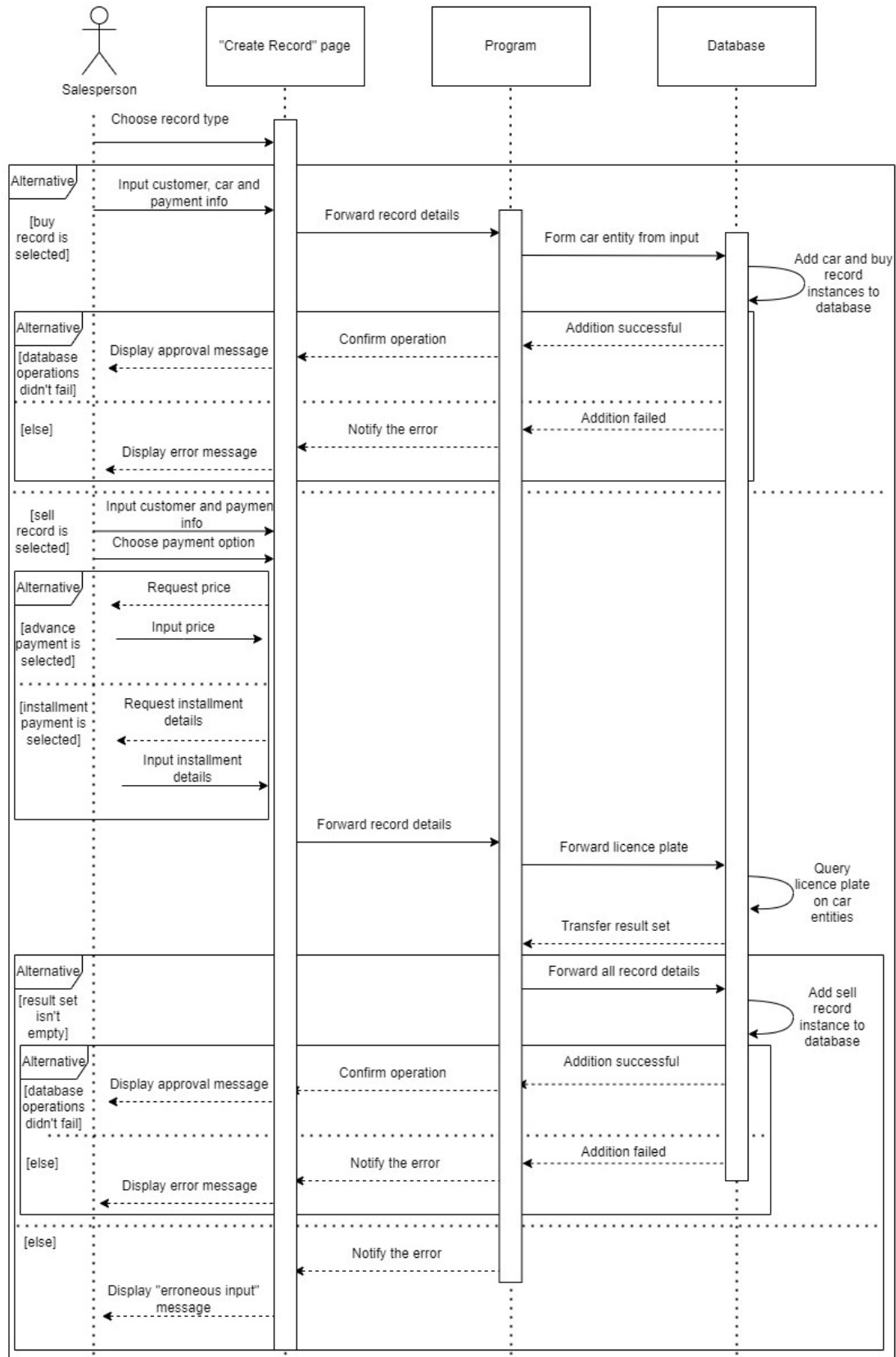


Figure 4.3: Sequence Diagram for Create Record

4.6 Delete Sell Record

4.6.1 Description and Priority

Accidental entries, disagreements between parties, and other unforeseen circumstances may necessitate the deletion of sale records from the system. If necessary, the salesman initiates this functionality to remove an invalid sell record. It has a medium priority because the occurrences of situations stated above are assumed to be low. Therefore, the anticipated usage rate is sparse. This operation shall take no more than one second to finalize. Figure 4.4 demonstrates the consecutive actions to be taken for this functionality throughout time as an activity diagram.

4.6.2 Stimulus/Response Sequences

Salespeople who are responsible for the record-keeping issues this feature whenever there's a problem with a specific buy record. It is initiated by choosing the "Delete Sell Record" option, after entering licence of the car. The programs' reply will be to remove the specified buy record from the database.

4.6.3 Functional Requirements

REQ-1: The system shall request the record id of the particular record that needs to be deleted.

REQ-2: The system shall update the database by removing the sell record with entered record id.

4.7 Delete Buy Record

4.7.1 Description and Priority

The salesman initiates this functionality if unexpected cases such as incorrect inputs, or parties' conflicts occur. It is given a medium priority since the likelihood of the above-mentioned events is considered to be low. As a result, the anticipated usage rate is sparse. This operation shall take no more than one second to finalize. Figure 4.4 demonstrates the consecutive actions to be taken for this functionality throughout time as an activity diagram.

4.7.2 Stimulus/Response Sequences

When a problem with a specific sell record arises, the salesperson in charge of record-keeping activates this feature by choosing the "Delete Sell Record" option after entering licence of the car. The program will respond by deleting the selected buy record from the database.

4.7.3 Functional Requirements

REQ-1: The system shall request the record id of the record that needs to be removed.

REQ-2: The system shall update the database by deleting the buy record with entered record id.

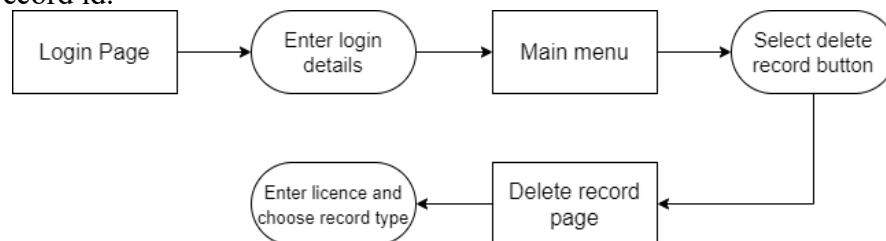


Figure 4.4: Activity Diagram for Delete Record

4.8 View Sell Record

4.8.1 Description and Priority

This functionality can be used by salespeople, accountants, and dealership managers to access the sell entries that are present on the system database. As it serves the purpose of accessing previous records for different intentions, the prioritize level can be labelled as high. The created sell entries in the database can be seen at any point after they have been generated. The expected usage frequency rate is high since the records can be viewed up to multiple times a day by various staff for different work-related reasons. View operations shall take less than 3 seconds if there are less than 1000 records. If more, it is acceptable to the 5-second limit.

4.8.2 Stimulus/Response Sequences

The initiation of this feature begins with the selection of the “View Sell Record” option on the provided menu by accountants, dealership managers, or salespeople. The system displays the available sell records according to the entered search filters as the response.

4.8.3 Functional Requirements

REQ-1: The system shall provide the filtering options which is to be used for selecting the suitable records.

REQ-2: The system shall display the extracted records with their details.

4.9 View Buy Record

4.9.1 Description and Priority

Salespeople, accountants, and dealership managers can utilize this feature to access the buy entries contained in the system database. The prioritize level can be categorized as high because it supports the goal of retrieving previous records. Buy entries in the database can be seen at any time after they've been generated. Because the records can be checked numerous times a day by diverse personnel for various work-related reasons, the estimated usage rate is high. View operations shall take less than 3 seconds. 5-second limit is acceptable if more than 1000 records are present in the database.

4.9.2 Stimulus/Response Sequences

This function is activated by clicking the "View Buy Record" option from the menu that appears. As a response, the system presents the eligible buy records based on the search filters specified.

4.9.3 Functional Requirements

REQ-1: The system shall provide the filtering parameters which is to be used for selecting the appropriate records.

REQ-2: The system shall show the suitable records with their details.

4.10 View The Cars In The Inventory

4.10.1 Description and Priority

Customers interested in purchasing a car from the dealership will be notified of the available vehicles by a salesman who will display inventory records. The search results can be filtered based on their preferences by the salesperson. Since it is used to display purchasable cars and is one of the most fundamental aspects of the system, it has a high priority. The density of use is high when the number of customers visiting the dealership and the needs of the dealership manager in terms of accessing this functionality are taken into consideration. If there are less than 1000 records view operations shall take less than 3 seconds. If more, it is acceptable to the 5-second limit. As a sequence diagram, Figure 4.5 depicts the sequential steps to be completed for this feature across time.

4.10.2 Stimulus/Response Sequences

This functionality is usually enabled by a salesperson interacting with a customer by selecting "View Inventory" from the main menu. As a result, based on the search filters set, the system displays the appropriate car data.

4.10.3 Functional Requirements

REQ-1: The system shall offer the filtering options that will be utilized to choose the fitting records.

REQ-2: The system shall display the appropriate records along with their data.

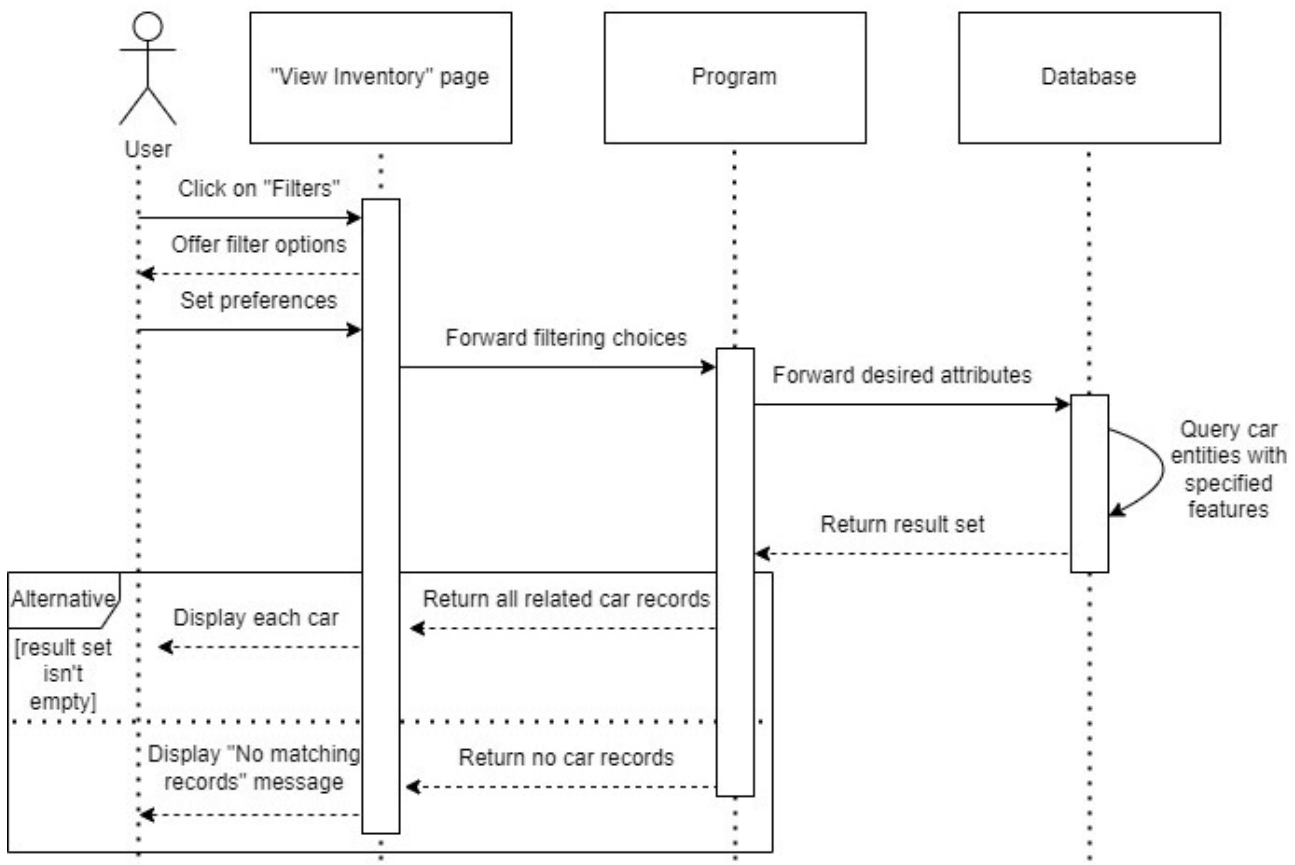


Figure 4.5: Sequence Diagram for View Inventory

4.11 Verify Auto Maintenance

4.11.1 Description and Priority

As every car requires maintenance on a regular basis, mechanic staff do this work and indicate the results using this feature. The indication is done by altering the maintenance schedule and updating the maintenance section of the cars on the database. The importance of this functionality is marked as high since performing maintenance is not discretionary. The anticipated use rate is dependent on the number of cars and their maintenance history. This procedure is expected to finish performing within 1 second. Figure 4.6 displays the consecutive steps that must be done for this feature over time as a sequence diagram.

4.11.2 Stimulus/Response Sequences

This functionality is enabled by the related mechanic staff who completed the maintenance on a car via selecting "Verify Auto Maintenance" from the main menu. As a result, the maintenance information of the car will be updated to the date the maintenance performed and the record associated with the car in the maintenance schedule will be removed from the database.

4.11.3 Functional Requirements

REQ-1: The system shall ask for the license plate and the date the maintenance was performed.

REQ-2: The system shall spot the car with the entered license plate in the database and update its maintenance column as the date entered.

REQ-3: The system shall find the associated schedule record with the car in the maintenance schedule section and remove it.

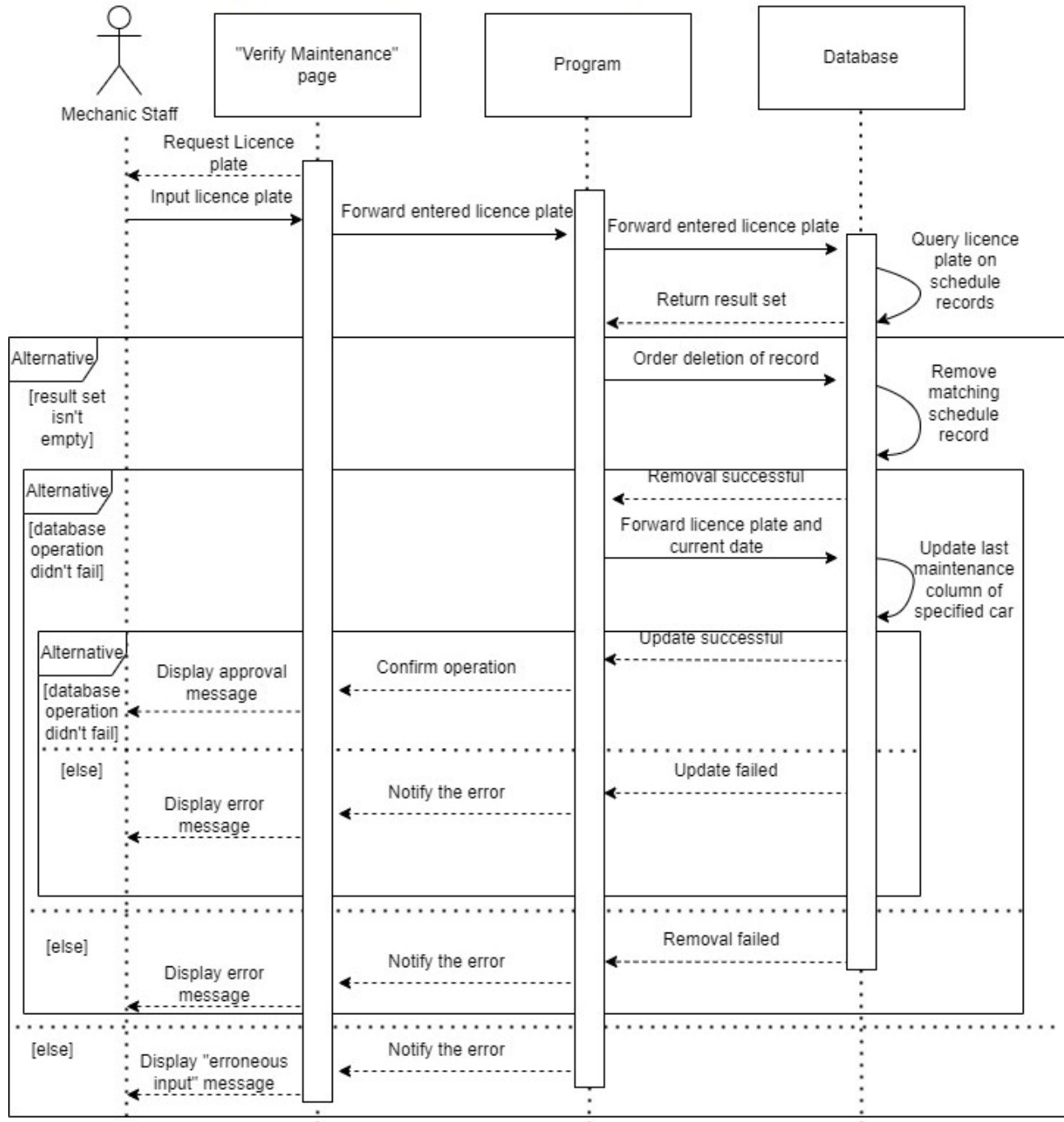


Figure 4.6: Sequence Diagram for Verify Maintenance.

4.12 Schedule Maintenance

4.12.1 Description and Priority

As periodic maintenance is a must-done operation for every car for safety purposes, the work to be done for each car must be scheduled. Therefore, the scheduling for the maintenance operations is organized by the related mechanic staff. Each member of the mechanic crew can modify the current schedule by making an addition, deleting, or changing dates. The priority of this functionality can be seen as high since organizing maintenance is required before the implementations. The expected rate of use is determined by the number of vehicles and their previous maintenance information. The system will list cars that passed their suggested maintenance time. The completion of this process should take no more than 2 seconds.

4.12.2 Stimulus/Response Sequences

This functionality is enabled by the mechanic staff who wishes to make changes to the current version of the maintenance schedule via selecting "Schedule Maintenance" from the main menu. As a result, the changes that the user stated are reflected in the schedule on the database.

4.12.3 Functional Requirements

REQ-1: The system shall ask for the type of change that will alter the current schedule and the license plate.

REQ-2: The system shall spot the record that matches with the inputs and apply the specified changes on the database.

REQ-3: The system shall list the cars which passed their suggested maintenance time.

4.13 View The Maintenance Schedule

4.13.1 Description and Priority

The mechanics need to look at the scheduled maintenance in order to organize the appropriate technical operations and keep track of the cars that need to be maintained. This capability has a high priority as viewing the most recent version of the maintenance schedule is required to get prepared for work and make long-term plans. The rate of usage is expected to increase during maintenance periods of cars, and it is assumed to be low in "maintenance-free" periods. The completion of this process should take no more than 1 second. As an activity diagram, Figure 4.7 illustrates the series of steps to be taken for this feature.

4.13.2 Stimulus/Response Sequences

This functionality is initiated by the mechanic staff who wishes to access the schedule via selecting the "View Maintenance Schedule " section from the main menu. As a result, the records that belong to the schedule are displayed as a list along with their details.

4.13.3 Functional Requirements

REQ-1: The system shall display the schedule records along with their details.

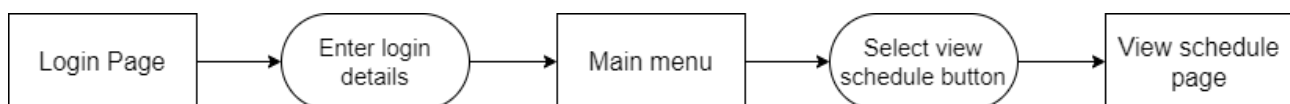


Figure 4.7: Activity Diagram for View Schedule

5. Other Non-functional Requirements

5.1 Performance Requirements

Logging in, logging out, updating profile, creating, and deleting records, viewing maintenance schedule, and verifying maintenance shall take less than 1 second. Scheduling maintenance shall be completed within 2 seconds. Listing operations shall take less than 3 seconds if there are less than 1000 records. If more records are present, a 5-second limit is acceptable. These timing restrictions are to ensure business flow is not slowed down by operations. This implementation will result in a high-performance rate in user commands and provide quick responses which will increase the overall system quality. Similarly, errors should be minimized as much as possible to avoid unwanted system breaks.

5.2 Safety Requirements

Records shall not be corrupted with no tolerance rate. Every input shall be checked to ensure no hostile input that will damage the records will be sent to the server. The system shall obey local privacy laws.

5.3 Security Requirements

Every user shall enter their credentials to access the system. Entered password shall be hidden from plain view. Every login detail shall be encrypted with the SHA-256 algorithm to prevent unauthorized access. Entered texts shall be checked whether if they are in the correct form to prevent SQLI attacks.

5.4 Software Quality Attributes

The system shall be available to use at least %98 percent of runtime. The program shall forward inputs to the server without any error. The program should not be blocked by the firewall. To fix bugs and problems that occurred during operation without creating further damage, the system shall be built easily maintainable. The program shall run on computers that fit the minimum requirements without problems. The program shall be able to adapt to change easily in response to various system and user requirements. The program shall not damage any input taken from the user and any data that is present on the database. The system shall be thoroughly testable for both small and large-sized applications. The system must be easy to operate and have a clean UI.

5.5 Business Rules

User with suitable authorization levels shall use their allowed functions. Every operation shall be available during uptime to authorized users. Every user shall be able to log in, log out, and change their details. Salespeople shall be able to create sell records when the sale procedure is done and create buy the record when the buy operation is completed. They shall be able to view records whenever there is a need. They shall be able to view the current inventory to show the customers the cars that fit their criteria. In case of problematic records, salespeople shall be able to delete records. Dealership managers shall be able to view the sell and buy records to monitor operations done in the dealership. They shall be able to view inventory to give focus on operation type. Managers shall be able to view the maintenance schedule to see the current situation of the mechanic operations. Mechanic staff shall be able to verify maintenance when the operation is done. They shall be able to schedule maintenance to plan future maintenance. They shall be able to view the schedule to see the cars to be maintained. Accountants shall be able to access sell and buy records to calculate tax-related duties.

6. Other Requirements

In the database, 3NF (Third Normal Form) and BCNF (Boyce-Codd Normal Form) normalization forms will be properly applied to avoid data redundancy and ensure that data is stored logically. Server and program fit Turkish privacy laws, but it may require additional modification to use internationally, especially in the EU. UI language is English, and no translation is planned. UTF-8 codec will be used systemwide to guarantee international users will not encounter any character problems. Due to the unique application of the project, only minor components of this project could be reused.

Appendix A: Glossary

GUI (Used in Section 3.1): Graphical user interface

SQL (Used in Section 3.3): Structured query language

SQLI (Used in Section 5.3): Structured query language injection.

UI (Used in Section 3.1): User interface

Appendix B: Analysis Models

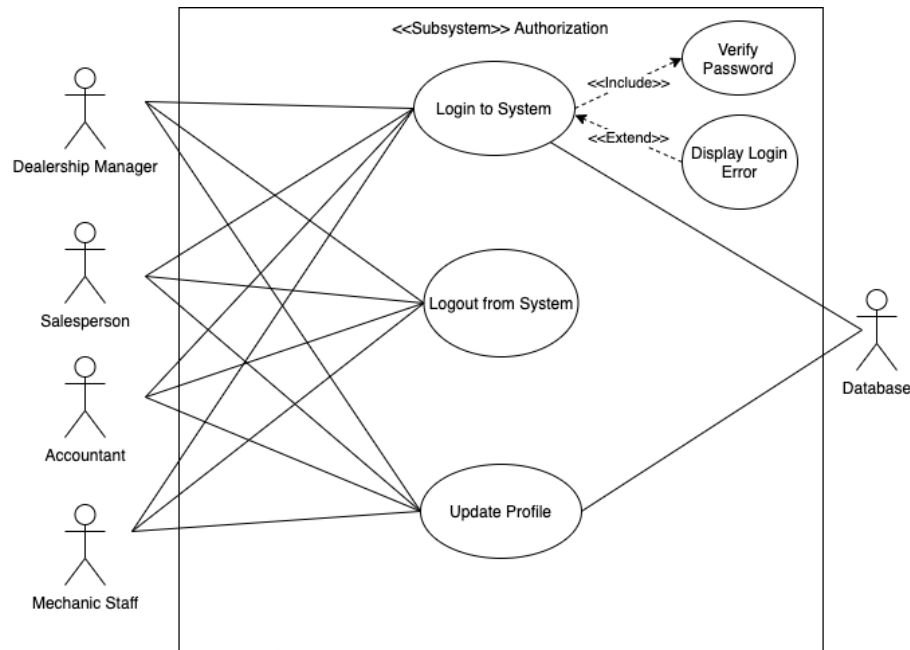


Figure A: Authorization Subsystem Use Case

Use Case	Login to System
Actor	Dealership Manager Salesperson Accountant Mechanic Staff Database
Description	Login to System allows dealership managers, salespeople, accountants, and mechanic staff to access their profiles to use the system according to their privileges. Since usernames and passwords are stored in the database, each time a user logs in to the system, verification of passwords is done by info taken from the database.

Use Case	Logout from System
Actor	Dealership Manager Salesperson Accountant Mechanic Staff
Description	Logout from System allows dealership managers, salespeople, accountants, and mechanic staff to exit the system to avoid unauthorized operations.

Use Case	Update Profile
Actor	Dealership Manager Salesperson Accountant Mechanic Staff Database
Description	Update Profile allows users to change their profile information which consists of personal data, position, and statistics about the user. Changing account passwords can also be done from this section. All the information regarding the users is held in the database. Every interaction with the profile requires a connection to the database.

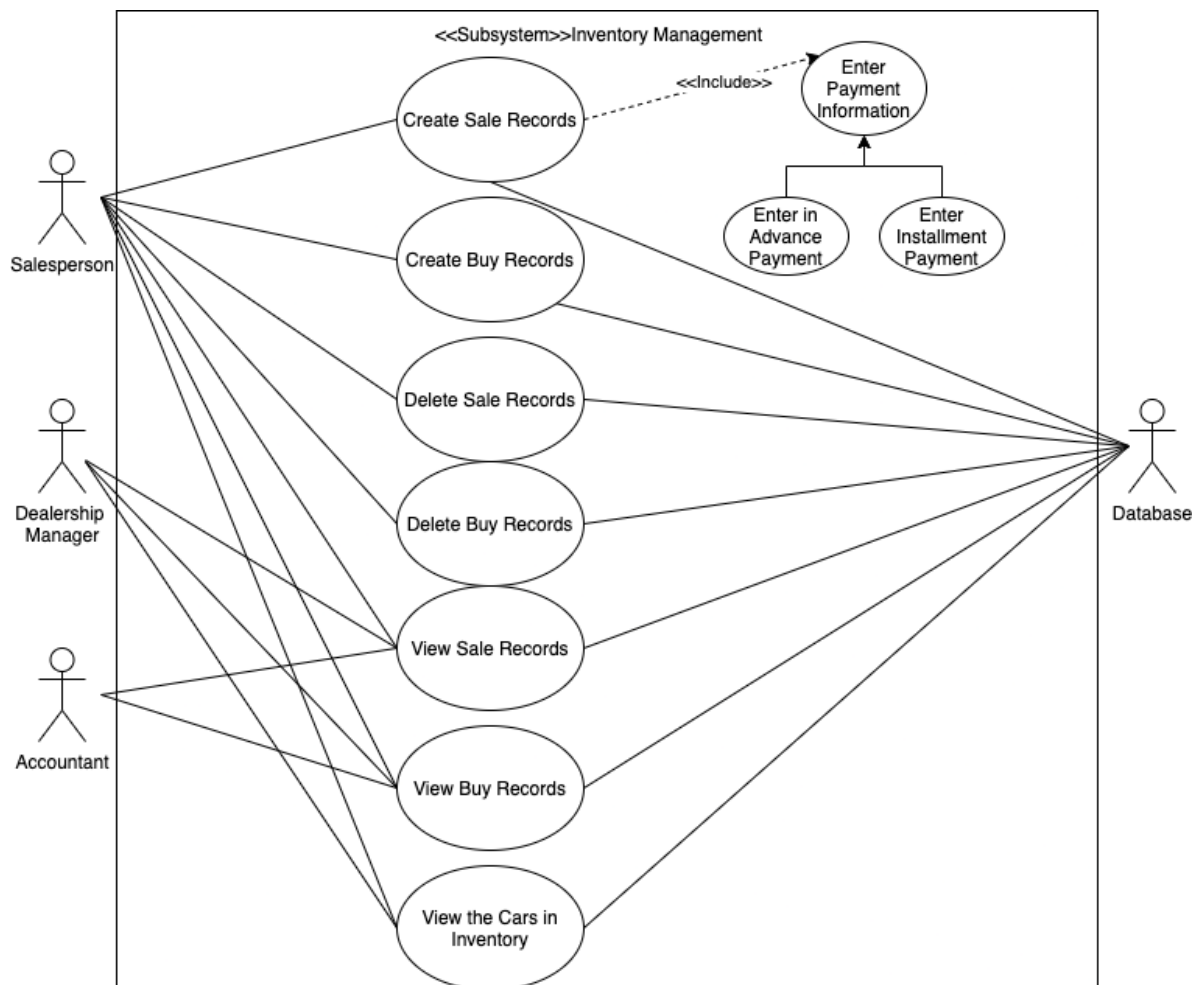


Figure B: Inventory Management Subsystem Use Case Diagram

Use Case	Create Sale Records
Actor	Salesperson Database
Description	After the sale process is done between customers and dealership, the salesperson initiates this use case. The sale record consists of necessary customer information, properties of the car, and process details including payment information. Payment can be done in advance or by installment. All sale records are kept in the database, therefore connection to the database is essential.

Use Case	Create Buy Records
Actor	Salesperson Database
Description	After buying the second-hand car from the seller, the salesperson initiates this use case. Buy record consists of necessary seller information, properties of the second-hand car, and process details including payment done to the seller. All buy records are kept in the database, therefore connection to the database is essential.

Use Case	Delete Sale Records
Actor	Salesperson Database
Description	In case of accidental entries, disagreement between sides, and other unexpected scenarios, sale records may need to be deleted from the system. The salesperson initiates this use case if necessary. Since all sale records are kept in the database, connection to the database is essential to delete them.

Use Case	Delete Buy Records
Actor	Salesperson Database
Description	In case of accidental entries, disagreement between sides, and other unexpected scenarios, buy records may need to be deleted from the system. The salesperson initiates this use case if necessary. Since all buy records are kept in the database, connection to the database is needed to be able to delete the records.

Use Case	View Sale Records
Actor	Salesperson Dealership Manager Accountant Database
Description	After the salesperson enters the log of sale, the manager, accountant, and salesperson may need to check records of sales. Manager may want to keep track of sales, personnel performances, and overall statistics. Similarly, the accountant may need to check records to manage financial operations such as calculation of PNL and taxes. All sale records are kept in the database; therefore, a database connection is needed.

Use Case	View Buy Records
Actor	Salesperson Dealership Manager Accountant Database
Description	After the salesperson enters the log of buy, the manager, accountant, and salesperson may need to check records of bought cars. Manager may want to view recently bought cars for management purposes. Similarly, the accountant may need to check records to manage financial operations such as calculation of PNL and taxes. All buy records are kept in the database, therefore connection to the database is necessary.

Use Case	View the Cars in Inventory
Actor	Salesperson Dealership Manager Database
Description	To show available cars to the customers, the salesperson must initiate this use case. Search results can be filtered according to the preferences of the customers. The dealership manager can arrange plans regarding the current situation of the inventory. All the information about the cars in the inventory is kept in the database, therefore connection to the database is required.

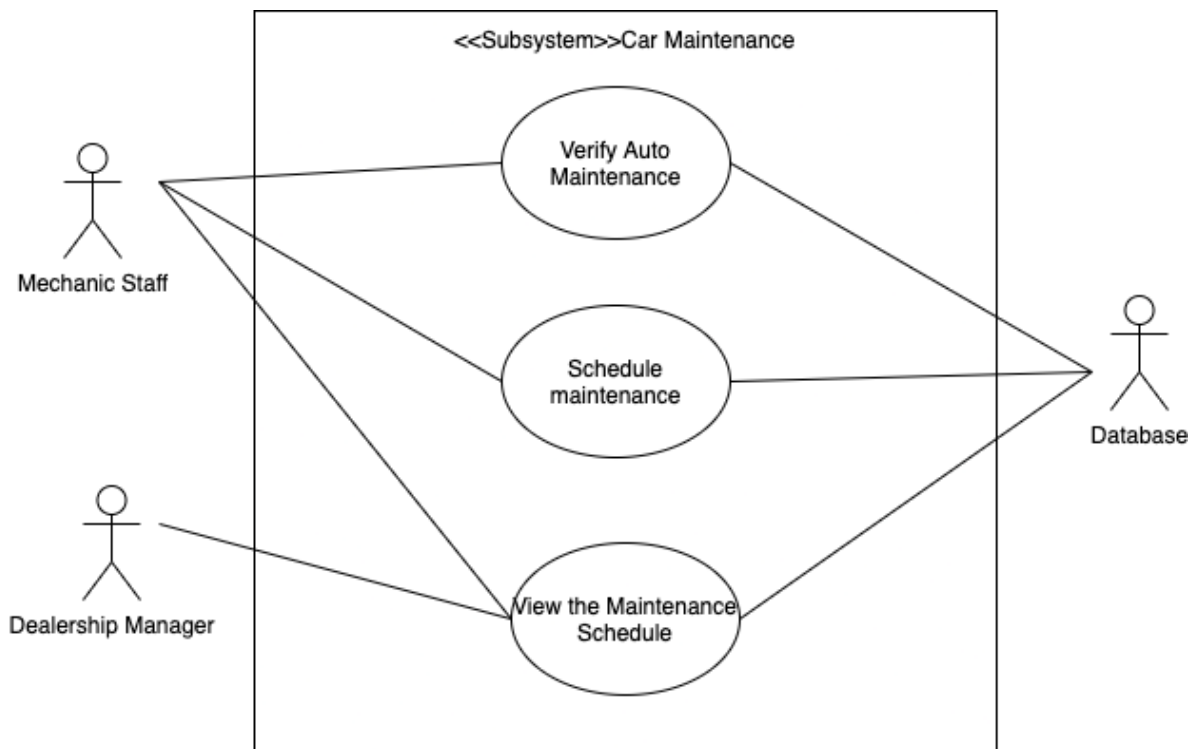


Figure C: Car Maintenance Subsystem Use Case Diagram

Use Case	Verify Auto Maintenance
Actor	Mechanic Staff Database
Description	Since every car needs periodic maintenance, mechanic staff performs maintenance on cars and indicates this information. Because the database stores the maintenance information for each automobile in the inventory, a database connection is requested.

Use Case	Schedule Maintenance
Actor	Mechanic Staff Database
Description	To keep track of maintained cars, mechanic staff can create a maintenance schedule. The records will consist of car details, date and time of the procedure, operations that are done on the car, and the name of the personnel who performed the maintenance. Schedules are held in the database.

Use Case	View the Maintenance Schedule
Actor	Mechanic Staff Dealership Manager Database
Description	To keep track of maintained cars, the performance of the staff and cars which needs to be maintained, the manager can access to the schedule. Mechanic staff can view the schedule to organize the necessary technical procedures. A database connection is required to perform this use case.

The subsystems of the program work independently from one another, meaning that they are not contributing to any kind of data flow to each other. The only notable relationship between them which is not possible to indicate in the use case diagram is how car maintenance and inventory management subsystems are requiring authorization subsystem to provide privilege to users to access functionalities within their scope.

Appendix C: To Be Determined List

Since every necessary detail has been addressed in the document, there are no "to be determined" specifics.