

fsoft2025_1DA_2: Iteration 1 - Technical Report

Domain Model, Glossary, Functional and Non-Functional Requirements, System Sequence Diagram

Dinis Gonçalves, Lucas Santos, Rúben Silva, and Vítor Teixeira

ISEP, Instituto Superior de Engenharia do Porto,
Rua do Dr António Bernardino de Almeida 431, 4249-015 Porto, Portugal
{1241099,1241008,1240708,1232067}@isep.ipp.pt
<https://www.isep.ipp.pt>

Abstract. In the following pages, the reader will be able to be informed about the application's domain model, glossary, functional and non-functional requirements, containing use cases, and their specifications, and system sequence diagrams (SSD).

Key words: Business, C++, Fleet Management, IPP, ISEP, Management, Programming, Software Development, Command-line Interface, Binary Storage, Expense Tracking.

1 Introduction

In the contemporary business landscape, the efficient management of automotive fleets has become a critical factor in organizational success. This programming project aims to develop a comprehensive Fleet Management System designed to streamline operations, optimize resource allocation, and enhance decision-making processes related to vehicle fleets.

We named the application, being developed in this project, "FleetManager".

The system's architecture incorporates four key stakeholder interfaces, each serving distinct but interconnected roles within the FleetManager's ecosystem:

- Fleet Manager: Responsible for overseeing the entire vehicle inventory, including adding vehicles, removing vehicles, see the whole list of vehicles, add storage locations, remove storage locations, consult storage locations and list all storage locations.
- Trip Manager: Focuses on journey planning and route optimization. There will be some functionalities associated such as adding trips, removing trips and listing all the trips in the log.
- Financial Manager: Concentrates on cost control and financial analysis related to the fleet. The functionalities associated to this profile will be to add expenses, remove expenses, and list all the expenses.

- Driver Manager: Handles all aspects related to driver administrations. It will be able to add drivers, remove drivers and list all the drivers inserted in the log.

The objective of FleetManager is to provide a solution that optimizes all operations related to corporate vehicle administration. So we interconnect the fields that are crucial to the success of this type of business.

The system will be developed in C++, which is the programming language taught in this class.

2 Domain Model

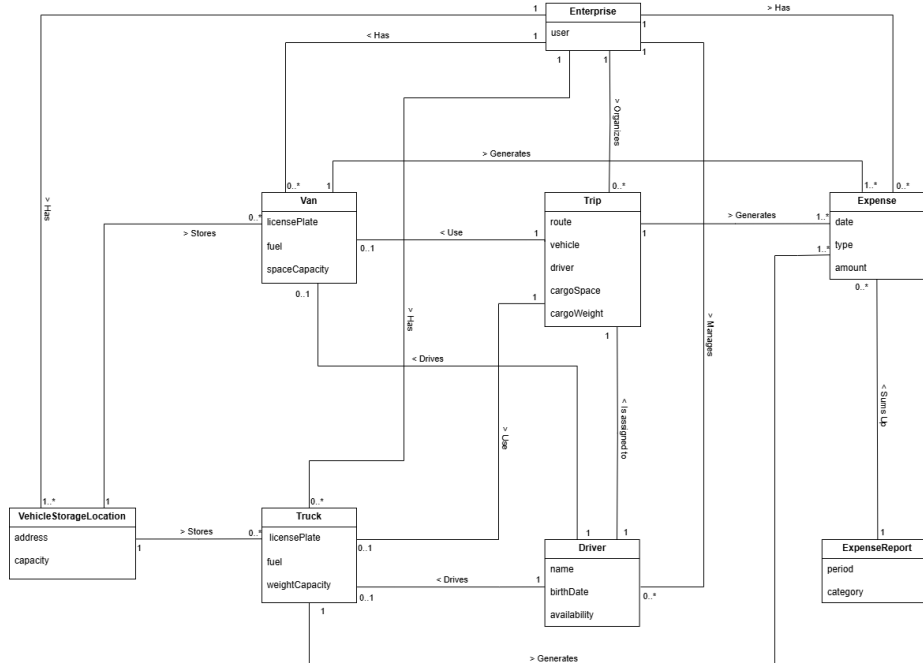


Fig. 1: FleetManager's Domain Model

3 Glossary

| Term/Expression (EN) | Termo/Expressão (PT) | Definition/Description |
|--------------------------|------------------------------------|--|
| Driver | Condutor | Employee who drives the vehicle and to whom is assigned a trip |
| Expense | Despesa | Amount of money spent. This includes fuel, tolls, fines, inspection and insurance expenses. |
| Expense Report | Relatório de Despesa | Expenses recorded in the system. These expenses can be separated by driver, trip, vehicle and time period. |
| Trip | Viagem | Route from the start to the end point. Specified by the fleet manager and carried out by a driver. |
| Truck | Camião | A type of vehicle used by the company. Preferentially used when big cargo space (or a big weight capacity) is needed. |
| Van | Carrinha | A type of vehicle used. Preferably employed when the cargo weight, and space, is not excessive and speed is a surplus value. |
| Vehicle | Veículo | Vehicle of the fleet, it can be either a truck or a van, that will be used by a driver to carry out a trip. |
| Vehicle Storage Location | Local de Armazenamento de Veículos | Locations where, after a trip, the drivers can park/store the vehicles they've used. |

Table 1: FleetManager Glossary

4 Non-Functional Requirements

4.1 Usability

- Command-line based user interface
- Informative, yet simple, error messages
- Easy-to-use option menus

4.2 Reliability

- Data storage must be reliable
- System must be available at any time
- Validation of all user inputs

4.3 Performance

- Response time under 1 second

4.4 Supportability

- Full application support through GitHub page
- Recurrent application testing

4.5 Design Constraints

- Data persistent and kept in binary files

4.6 Implementation Constraints

- C++ programming language

4.7 Interface Constraints

- No

5 Functional Requirements

5.1 Actor: Fleet Manager

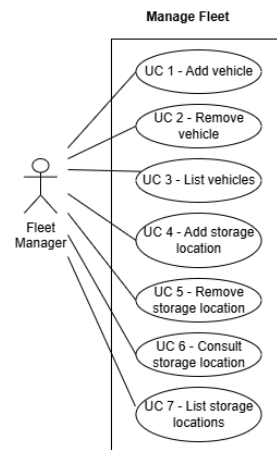


Fig. 2: Fleet Manager's Use Case Diagram

Use Case: Add vehicle (Specification)

| | |
|-------------------------|---|
| Actor | Fleet Manager |
| Use case name | Add vehicle |
| Description | The user adds a vehicle to the system |
| Precondition | The "Fleet Manager" user must be selected |
| Post-condition | A vehicle is added to a log |
| Main flow | <ol style="list-style-type: none"> 1. System displays a option menu 2. User selects "Add vehicle" 3. System displays an input space 4. User fills the input space with information 5. System will check if the information type is correct 6. Use case returns to step 3 until all of the vehicle's properties are filled 7. A vehicle is added to a log |
| Alternative path | <ol style="list-style-type: none"> 5. - Information type is incorrect - User receives an error message - Use case resumes at step 3 |
| Exceptions | If the user, at any point, wishes to cancel, the user can type "cancel operation" and the use case resumes at step 1 |

Table 2: Use Case: Add Vehicle

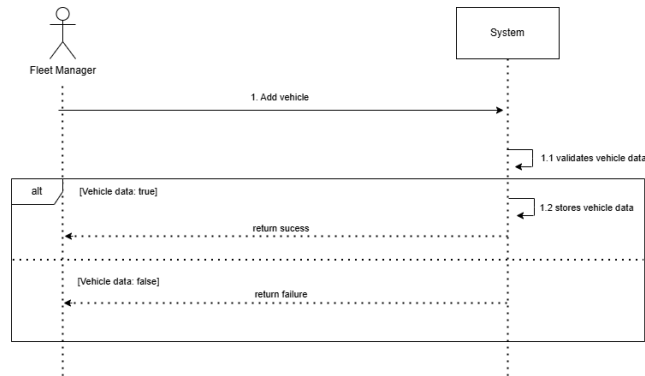
SSD: UC 1 - Add vehicle

Fig. 3: SSD: Fleet Manager's UC 1

Use Case: Remove vehicle (Specification)

| | |
|-------------------------|--|
| Actor | Fleet Manager |
| Use case name | Remove vehicle |
| Description | The user removes a vehicle from the system |
| Precondition | The "Fleet Manager" user must be selected and, at least, one vehicle must be logged in the system |
| Post-condition | A vehicle is removed from the log |
| Main flow | <ol style="list-style-type: none"> 1. System displays a option menu 2. User selects "Remove vehicle" 3. System displays an input space 4. User inserts the vehicles license plate 5. System check if the license plate is associated to an existing vehicle 6. The vehicle is removed from the log |
| Alternative path | <ol style="list-style-type: none"> 5. - License plate is not associated - User receives an error message - Use case resumes at step 3 |
| Exceptions | If the user, at any point, wishes to cancel, the user can type "cancel operation" and the use case resumes at step 1 |

Table 3: Use Case: Remove Vehicle

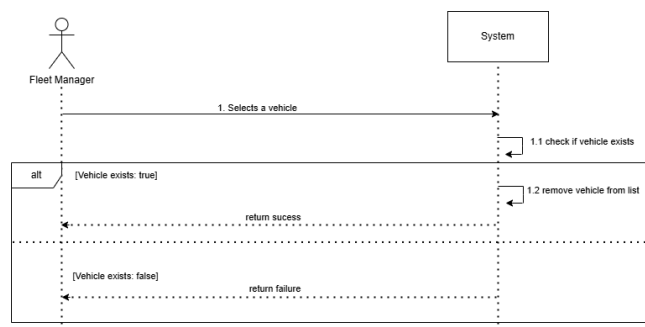
SSD: UC 2 - Remove vehicle

Fig. 4: SSD: Fleet Manager's UC 2

Use Case: List vehicles (Specification)

| | |
|-------------------------|---|
| Actor | Fleet Manager |
| Use case name | List vehicles |
| Description | It's shown a list of all vehicles to the user |
| Precondition | The "Fleet Manager" user must be selected and, at least, one vehicle must be logged in the system |
| Post-condition | None |
| Main flow | <ol style="list-style-type: none"> 1. System displays a option menu 2. User selects "List vehicles" 3. System checks if any vehicles are on the system 4. System displays a list of all vehicles logged in the system |
| Alternative path | <ol style="list-style-type: none"> 3. - No vehicles are logged in the system - System sends a "No Vehicles" message - Use case resumes at step 1 |
| Exceptions | None |

Table 4: Use Case: List Vehicles

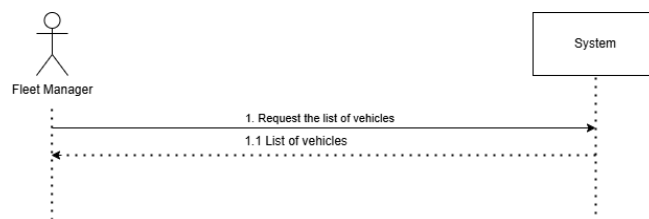
SSD: UC 3 - List vehicles

Fig. 5: SSD: Fleet Manager's UC 3

Use Case: Add storage location (Specification)

| | |
|-------------------------|--|
| Actor | Fleet Manager |
| Use case name | Add storage location |
| Description | The user adds a storage location to the system |
| Precondition | The "Fleet Manager" user must be selected |
| Post-condition | A storage location is added to a log |
| Main flow | <ol style="list-style-type: none"> 1. System displays a option menu 2. User selects "Add storage location" 3. System displays an input space 4. User fills the input space with information 5. System will check if the information type is correct 6. Use case returns to step 3 until all of the storage location's properties are filled 7. A storage location is added to a log |
| Alternative path | 5. - Information type is incorrect - User receives an error message - Use case resumes at step 3 |
| Exceptions | If the user, at any point, wishes to cancel, the user can type "cancel operation" and the use case resumes at step 1 |

Table 5: Use Case: Add Storage Location

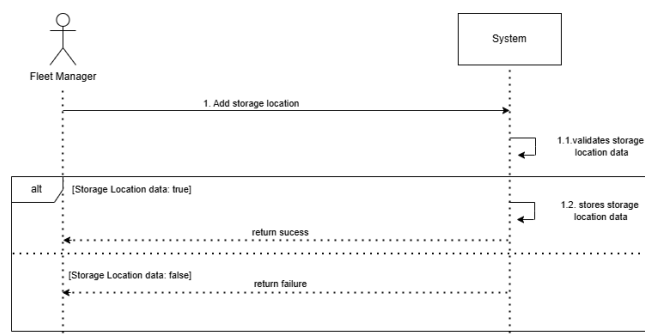
SSD: UC 4 - Add storage location

Fig. 6: SSD: Fleet Manager's UC 4

Use Case: Remove storage location (Specification)

| | |
|-------------------------|--|
| Actor | Fleet Manager |
| Use case name | Remove storage location |
| Description | The user removes a storage location from the system |
| Precondition | The "Fleet Manager" user must be selected and, at least, one storage location must be logged in the system |
| Post-condition | A storage location is removed from the log |
| Main flow | <ol style="list-style-type: none"> 1. System displays a option menu 2. User selects "Remove storage location" 3. System displays an input space 4. User inserts the storage location's ID number 5. System check if the ID is associated to an existing storage location 6. The storage location is removed from the log |
| Alternative path | <ol style="list-style-type: none"> 5. - ID is not associated - User receives an error message - Use case resumes at step 3 |
| Exceptions | If the user, at any point, wishes to cancel, the user can type "cancel operation" and the use case resumes at step 1 |

Table 6: Use Case: Remove Storage Location

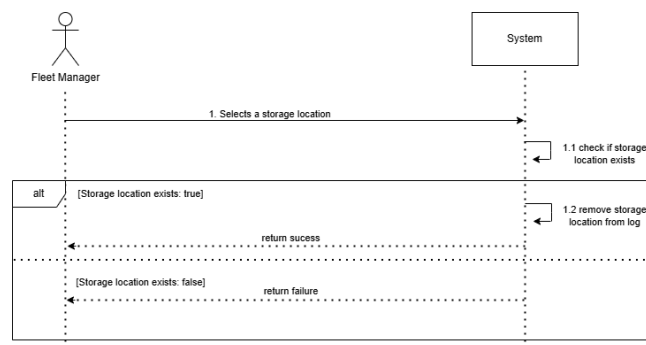
SSD: UC 5 - Remove storage location

Fig. 7: SSD: Fleet Manager's UC 5

Use Case: Consult storage location (Specification)

| | |
|-------------------------|---|
| Actor | Fleet Manager |
| Use case name | Consult storage location |
| Description | It's shown a list of all vehicles stored in a storage location to the user |
| Precondition | The "Fleet Manager" user must be selected and, at least, one storage location must be logged in the system |
| Post-condition | None |
| Main flow | <ul style="list-style-type: none"> - System displays a option menu - User selects "Consult storage location" - System checks if any vehicles are on the system - System displays an input space - User inserts the storage location's ID number - System check if the ID is associated to an existing storage location - System displays a list of all vehicles stored in the storage location |
| Alternative path | <ul style="list-style-type: none"> 3. - No vehicles are logged in the system - System sends a "No Vehicles" message - Use case resumes at step 1 6. - ID is not associated to any storage location - User receives an error message - Use case resumes at step 4 |
| Exceptions | If the user, at any point, wishes to cancel, the user can type "cancel operation" and the use case resumes at step 1 |

Table 7: Use Case: Consult Storage Location

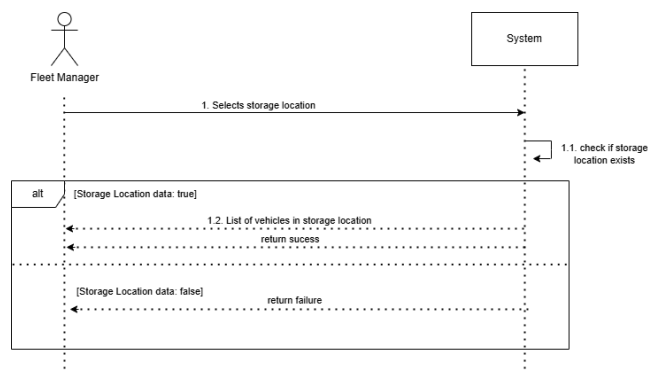
SSD: UC 6 - Consult storage location

Fig. 8: SSD: Fleet Manager's UC 6

Use Case: List storage locations (Specification)

| | |
|-------------------------|--|
| Actor | Fleet Manager |
| Use case name | List storage locations |
| Description | It's shown a list of all storage locations logged in the system to the user |
| Precondition | The "Fleet Manager" user must be selected and, at least, one storage location must be logged in the system |
| Post-condition | None |
| Main flow | <ol style="list-style-type: none"> 1. System displays a option menu 2. User selects "List storage locations" 3. System check if there are any storage locations logged in the system 4. System displays a list of all logged storage locations |
| Alternative path | <ol style="list-style-type: none"> 3. - No storage locations are logged in the system - System sends a "No Storage Locations" message - Use case resumes at step 1 |
| Exceptions | None |

Table 8: Use Case: List Storage Locations

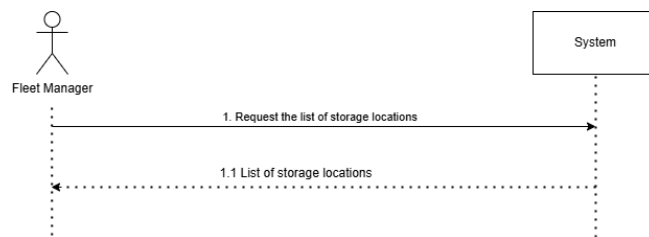
SSD: UC 7 - List storage locations

Fig. 9: SSD: Fleet Manager's UC 7

5.2 Actor: Trip Manager

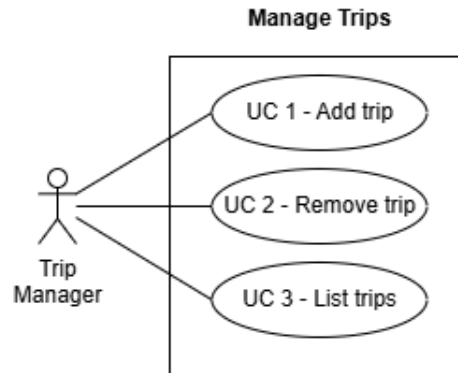


Fig. 10: Trip Manager's Use Case Diagram

Use Case: Add trip (Specification)

| | |
|-------------------------|--|
| Actor | Trip Manager |
| Use case name | Add trip |
| Description | The user adds a trip to the system |
| Precondition | The "Trip Manager" user must be selected |
| Post-condition | A trip is added to the log |
| Main flow | <ol style="list-style-type: none"> 1. System displays an option menu 2. User selects "Add trip" 3. System displays an input space 4. User fills the input space with information 5. System checks if the information type is correct 6. A trip is added to the log |
| Alternative path | <ol style="list-style-type: none"> 5. Information type is incorrect <ul style="list-style-type: none"> - User receives an error message - Use case resumes at step 3 |
| Exceptions | If the user, at any point, wishes to cancel, the user can type "cancel operation" and the use case resumes at step 1 |

Table 9: Use Case: Add Trip

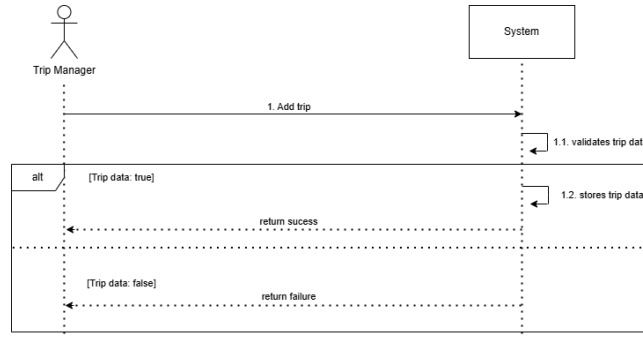
SSD: UC 1 - Add trip

Fig. 11: SSD: Trip Manager's UC 1

Use Case: Remove trip (Specification)

| | |
|-------------------------|---|
| Actor | Trip Manager |
| Use case name | Remove trip |
| Description | The user removes a trip from the log |
| Precondition | The “Trip Manager” user must be selected |
| Post-condition | A trip is removed from the log |
| Main flow | <ol style="list-style-type: none"> 1. System displays an option menu 2. User selects “Remove trip” 3. System displays an input space 4. User fills the input space with information 5. System checks if the information type is correct 6. A trip is removed from the log |
| Alternative path | <ol style="list-style-type: none"> 5. Information type is incorrect <ul style="list-style-type: none"> - User receives an error message - Use case resumes at step 3 |
| Exceptions | If the user, at any point, wishes to cancel, the user can type “cancel operation” and the use case resumes at step 1 |

Table 10: Use Case: Remove Trip

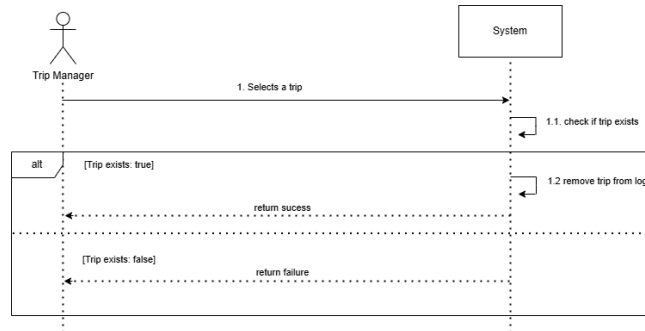
SSD: UC 2 - Remove trip

Fig. 12: SSD: Trip Manager's UC 2

Use Case: List trips (Specification)

| | |
|-------------------------|---|
| Actor | Trip Manager |
| Use case name | List trips |
| Description | The user consults the trip's log |
| Precondition | The "Trip Manager" user must be selected |
| Post-condition | Lists all the trips in the system |
| Main flow | 1. System displays an option menu 2. User selects "List trips" 3. System checks if any trips are on the system 4. System displays a list of all trips logged in the system |
| Alternative path | 3. Information type is incorrect - User receives a "No Trips" message - Use case resumes at step 2 |
| Exceptions | If the user, at any point, wishes to cancel, the user can type "cancel operation" and the use case resumes at step 1 |

Table 11: Use Case: List Trips

SSD: UC 3 - List trips

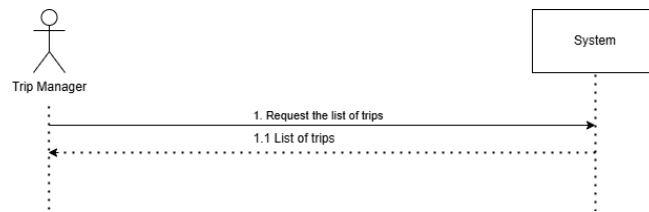


Fig. 13: SSD: Trip Manager's UC 3

5.3 Actor: Financial Manager

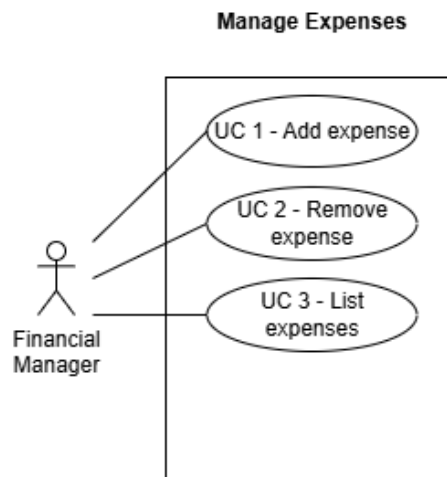


Fig. 14: Financial Manager's Use Case Diagram

Use Case: Add expense (Specification)

| | |
|-------------------------|--|
| Actor | Financial Manager |
| Use case name | Add expense |
| Description | The user adds an expense to the system |
| Precondition | The "Financial Manager" user must be selected |
| Post-condition | An expense is added to a log |
| Main flow | <ol style="list-style-type: none"> 1. System displays a option menu 2. User selects "Add expense" 3. System displays an input space 4. User fills the input space with information 5. System will check if the information type is correct 6. Use case returns to step 3 until all of the expense's properties are filled 7. An expense is added to a log |
| Alternative path | <ol style="list-style-type: none"> 5. - Information type is incorrect - User receives an error message - Use case resumes at step 3 |
| Exceptions | If the user, at any point, wishes to cancel, the user can type "cancel operation" and the use case resumes at step 1 |

Table 12: Use Case: Add Expense

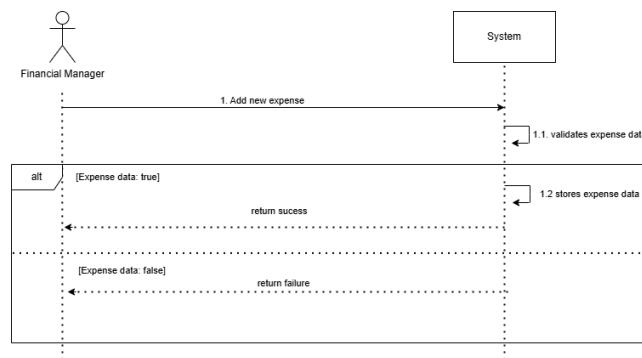
SSD: UC 1 - Add expense

Fig. 15: SSD: Financial Manager's UC 1

Use Case: Remove expense (Specification)

| | |
|-------------------------|---|
| Actor | Financial Manager |
| Use case name | Remove expense |
| Description | The user removes an expense from the system |
| Precondition | The "Financial Manager" user must be selected and, at least, one expense must be logged in the system |
| Post-condition | An expense is removed from the log |
| Main flow | <ol style="list-style-type: none"> 1. System displays a option menu 2. User selects "Remove expense" 3. System displays an input space 4. User inserts the expense's ID 5. System check if the ID is associated to an existing expense 6. The expense is removed from the log |
| Alternative path | <ol style="list-style-type: none"> 5. - ID is not associated - User receives an error message - Use case resumes at step 3 |
| Exceptions | If the user, at any point, wishes to cancel, the user can type "cancel operation" and the use case resumes at step 1 |

Table 13: Use Case: Remove Expense

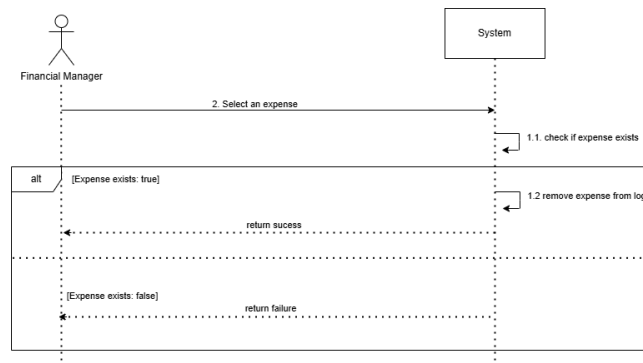
SSD: UC 2 - Remove expense

Fig. 16: SSD: Financial Manager's UC 2

Use Case: List expenses (Specification)

| | |
|-------------------------|---|
| Actor | Financial Manager |
| Use case name | List expenses |
| Description | It's shown a list of expenses to the user |
| Precondition | The "Financial Manager" user must be selected |
| Post-condition | None |
| Main flow | <ol style="list-style-type: none"> 1. System displays a option menu 2. User selects "List expenses" 3. System displays an input space 4. User inserts the expense's date 5. System displays an input space 6. User inserts the expense's category 7. System checks if there are any expenses with those parameters 8. System displays a list of expenses logged in the system |
| Alternative path | <ol style="list-style-type: none"> 7. - No expenses with such parameters are logged in the system - System sends a "No Expenses" message - Use case resumes at step 4 |
| Exceptions | <p>If the user, at any point, wishes to cancel, the user can type "cancel operation" and the use case resumes at step 1.</p> <p>If the user types "all" on both input spaces, the program shows all expenses records on the system.</p> |

Table 14: Use Case: List Expenses

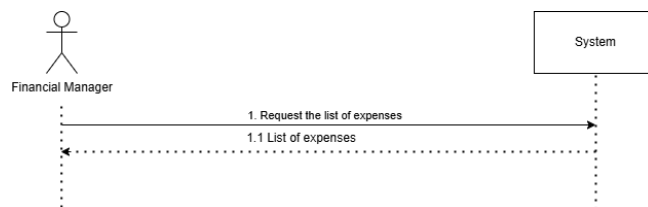
SSD: UC 3 - List expenses

Fig. 17: SSD: Financial Manager's UC 3

5.4 Actor: Driver Manager

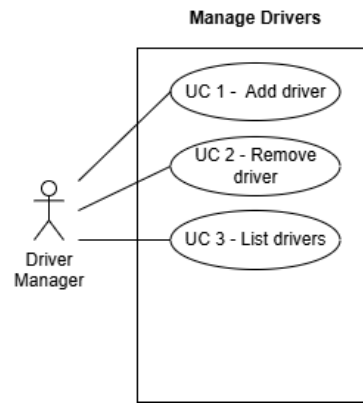


Fig. 18: Driver Manager's Use Case Diagram

Use Case: Add driver (Specification)

| | |
|-------------------------|---|
| Actor | Driver Manager |
| Use case name | Add driver |
| Description | The user adds a driver to the system |
| Precondition | The "Driver Manager" user must be selected |
| Post-condition | A driver is added to a log |
| Main flow | <ol style="list-style-type: none"> 1. System displays an option menu 2. User selects "Add driver" 3. System displays an input space 4. User fills the input space with information 5. System checks if the information type is correct 6. Use case returns to step 3 until all of the vehicle's properties are filled 7. A vehicle is added to a log |
| Alternative path | <ol style="list-style-type: none"> 5. Information type is incorrect <ul style="list-style-type: none"> - User receives an error message - Use case resumes at step 3 |
| Exceptions | If the user, at any point, wishes to cancel, the user can type "cancel operation" and the use case resumes at step 1 |

Table 15: Use Case: Add Driver

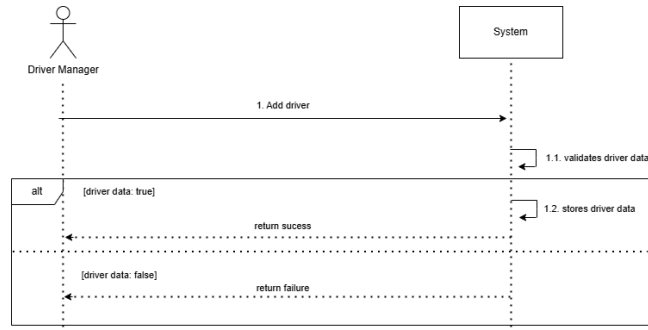
SSD: UC 1 - Add driver

Fig. 19: SSD: Driver Manager's UC 1

Use Case: Remove driver (Specification)

| | |
|-------------------------|---|
| Actor | Driver Manager |
| Use case name | Remove driver |
| Description | The user removes a driver from the system |
| Precondition | The "Driver Manager" user must be selected |
| Post-condition | A driver is removed from the log |
| Main flow | <ol style="list-style-type: none"> 1. System displays an option menu 2. User selects "Remove driver" 3. System displays an input space 4. User fills the input space with information 5. System checks if the information type is correct 6. Use case returns to step 3 until all of the vehicle's properties are filled 7. A driver is removed from the log |
| Alternative path | <ol style="list-style-type: none"> 5. Information type is incorrect <ul style="list-style-type: none"> - User receives an error message - Use case resumes at step 3 |
| Exceptions | If the user, at any point, wishes to cancel, the user can type "cancel operation" and the use case resumes at step 1 |

Table 16: Use Case: Remove Driver

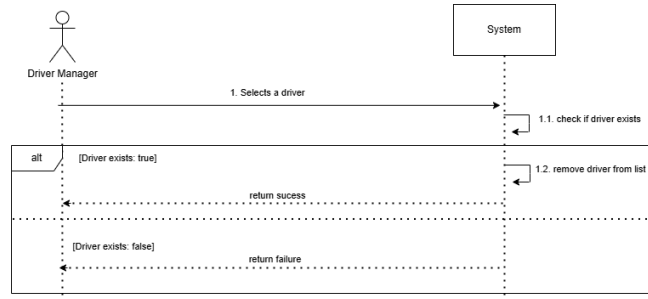
SSD: UC 2 - Remove driver

Fig. 20: SSD: Driver Manager's UC 2

Use Case: List drivers

| | |
|-------------------------|---|
| Actor | Driver Manager |
| Use case name | List drivers |
| Description | The user consults all the drivers available |
| Precondition | The "Driver Manager" user must be selected |
| Post-condition | Shows the drivers inserted in the log |
| Main flow | 1. System displays an option menu 2. User selects "List drivers" 3. System displays an input space 4. User fills the input space with information 5. System checks if the information type is correct 6. Shows all the drivers available |
| Alternative path | 5. Information type is incorrect - User receives a "No Drivers" message - Use case resumes at step 3 |
| Exceptions | If the user, at any point, wishes to cancel, the user can type "cancel operation" and the use case resumes at step 1 |

Table 17: Use Case: List Drivers

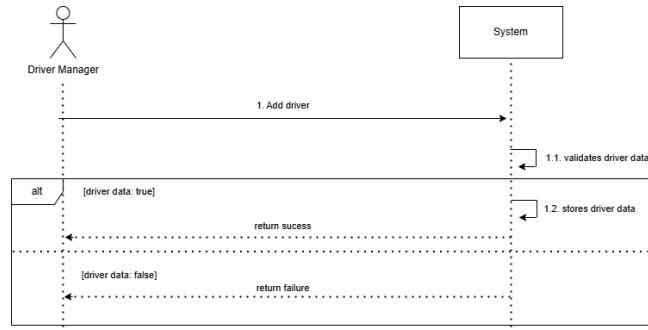
SSD: UC 3 - List drivers

Fig. 21: SSD: Driver Manager's UC 3