**Chapter 4**

**4.1 Overview:**

In this chapter we are talking about the system diagrams of the project like Data Flow Diagram(DFD), Context Diagram, System Framework, Workflow Diagram, Use Case Diagram and Sequence Diagram, the Actors of the system and their interaction with the components.

And also include the requirements that the system must satisfy are of two types, which are the functional and the non-functional requirements.

**Functional requirements**: The functional requirement will describe a particular behavior or function of the system when certain conditions are met and developers must implement them to enable users to accomplish their tasks.

User: For users these functions include

-Login: the system must allow the user to log in

-Payment: the system must allow the user to pay

-View history: system shows the history of the car

-Contact us: contact with the admin of the system

-Report stolen: add report stolen to the system

Admin: For admin these functions include

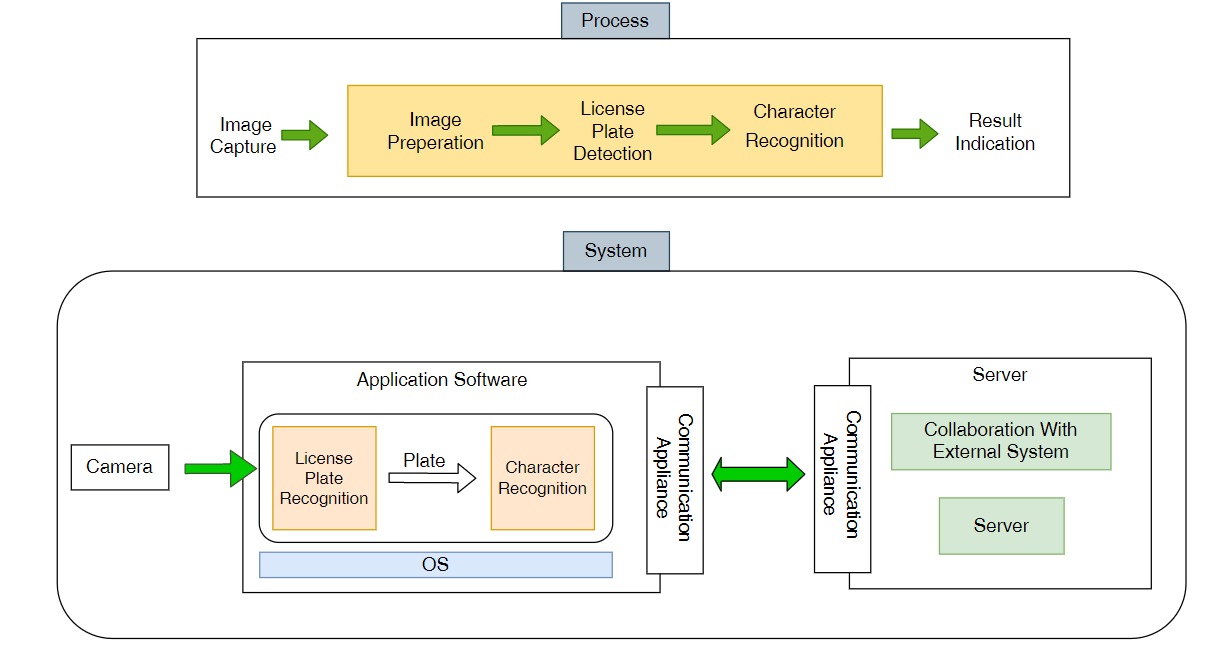
-Control reports: system reports must contain all reports of users and view the history of the car

-Database: system database must contain all related information about the user and his car (check and approve)

-Sign up: the system must allow to add a new user and sign up

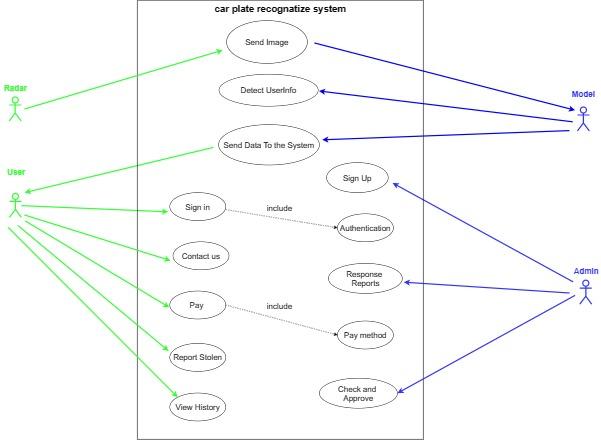
**Non Functional requirements**: A non-functional requirement (NFR) is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviors. Non-functional requirements are often called the "quality attributes'' of a system. Non-Functional requirements such as: 6 Appearance & usability: simple look and easy-to-use application Availability: the system must be available at any time Response: the system is faster in to respond of requests of the user Security: users’ information should be secure Operational & environmental: the application shall be used in variable lighting conditions Reliability: consistency of the system appears in the system being available all the time. Scalability: the ability of the system to handle the growth of data.

**4.2 System Framework:**



**4.3 Analysis:**

4.3.1 System Specification Use Case:



| Send Image | Use case Name |
| --- | --- |
| Radar Camera | Primary Actor |
| Model | Secondary Actor |
| Radar send token image to the model. | Purpose |
| Radar took an image | Pre-conditions |
| The model Receives the image and processes it. | Post-conditions |

| Detect User Information | Use case Name |
| --- | --- |
| Model | Primary Actor |
| The model process the image and illustrate the required details of the image. | Purpose |
| The model has an image to process it. | Pre-conditions |
| the image details save in the server of the model. | Post-conditions |

| Send data to the system | Use case Name |
| --- | --- |
| Model | Primary Actor |
| user | Secondary Actor |
| The model sends the car details to the user account in the system.. | Purpose |
| The model must have the image details that have been processed. | Pre-conditions |
| The image details saved in user account in the system. | Post-conditions |

| Sign Up | Use case Name |
| --- | --- |
| Admin | Primary Actor |
| Admin add user access information to the system. | Purpose |
| Admin have the user information. | Pre-conditions |
| User's information will be added to the system. | Post-conditions |

| Sign in | Use case Name |
| --- | --- |
| User | Primary Actor |
| User enter to the system using his sign in information. | Purpose |
| Authentication: To authenticate if a user's information is in the system or not. | Include use case |
| User has the right information to get into the system. | Pre-conditions |
| Get Access to the system successfully. | Post-conditions |

| Contact Us | Use case Name |
| --- | --- |
| User | Primary Actor |
| User can send report about.his problem. | Purpose |
| User must be logged to the system. | Pre-conditions |
| The report will be sent to the system.. | Post-conditions |

| Response Reports | Use case Name |
| --- | --- |
| Admin | Primary Actor |
| Admin can response to the reports of the users.. | Purpose |
| There are some reports to deal with it. | Pre-conditions |
| Send feedback to the user's reports. | Post-conditions |

| Pay | Use case Name |
| --- | --- |
| User | Primary Actor |
| User can pay his fines using pay function. | Purpose |
| Pay method: user choose how he will pay for the fine | Include Use case |
| There is a fine in the user's account. | Pre-conditions |
| The fine is canceled after the payment. | Post-conditions |

| Report Stolen | Use case Name |
| --- | --- |
| User | Primary Actor |
| User can report the system about his stolen car. | Purpose |
| His car was stolen. | Pre-conditions |
| If the report approved by the system it will send a warning to the road patrols | Post-conditions |

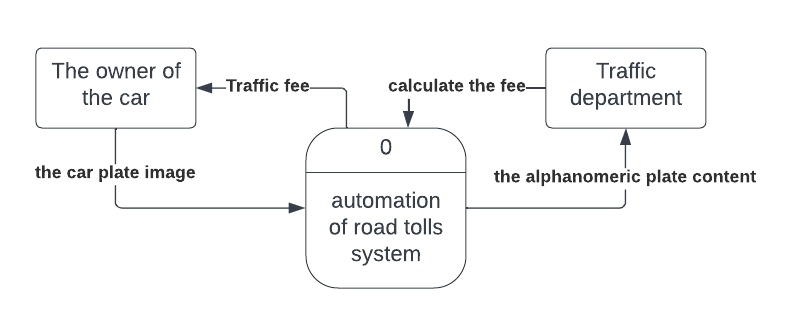
| Check and Approve | Use case Name |
| --- | --- |
| Admin | Primary Actor |
| Admin check if the user's report for stolen car true or by mistake, if true then approve the report.. | Purpose |
| There is a report about stolen car and check if the report right or wrong. | Pre-conditions |
| If the report was right the admin will approve the stolen report. | Post-conditions |

| View History | Use case Name |
| --- | --- |
| User | Primary Actor |
| User can view his historical fines. | Purpose |
| User must be logged into the system. | Pre-conditions |
| Details of the fines will e viewed to the user. | Post-conditions |

**4.3.2 Non Functional Requirements:**

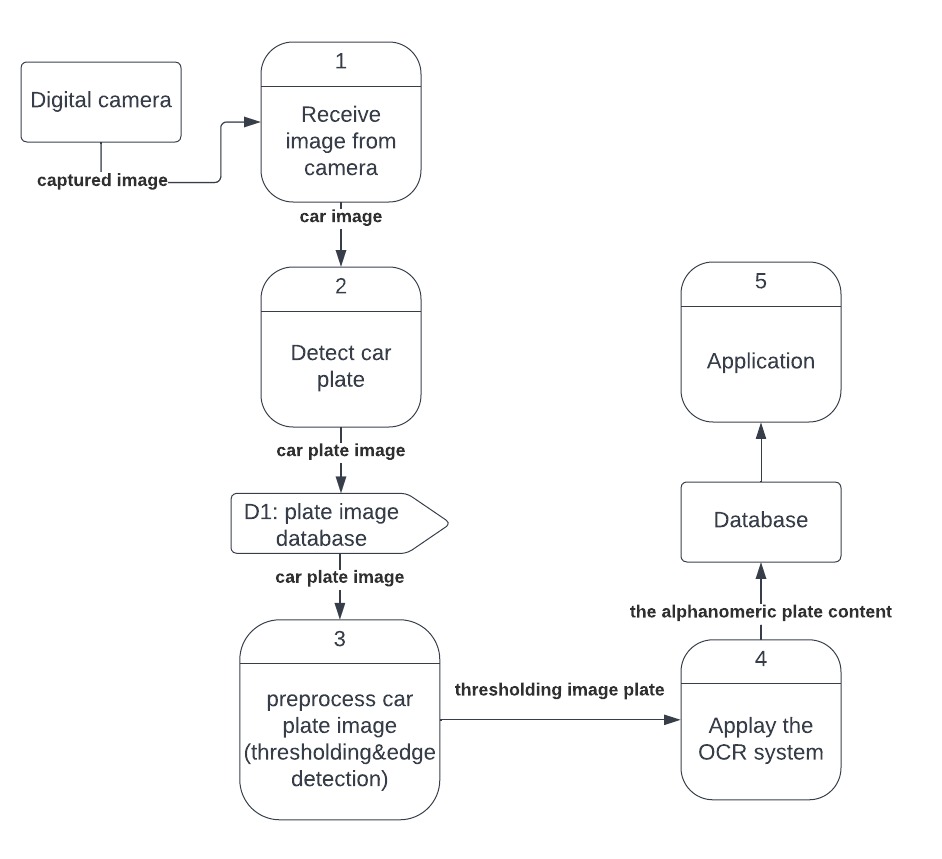
A non-functional requirement (NFR) is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviors. Non-functional requirements are often called the "quality attributes'' of a system. Non-Functional requirements such as: 6 Appearance & usability: simple look and easy-to-use application Availability: the system must be available at any time Response: the system is faster in to respond of requests of the user Security: users’ information should be secure Operational & environmental: the application shall be used in variable lighting conditions Reliability: consistency of the system appears in the system being available all the time. Scalability: the ability of the system to handle the growth of data.

**4.3.3 Context Diagram:**

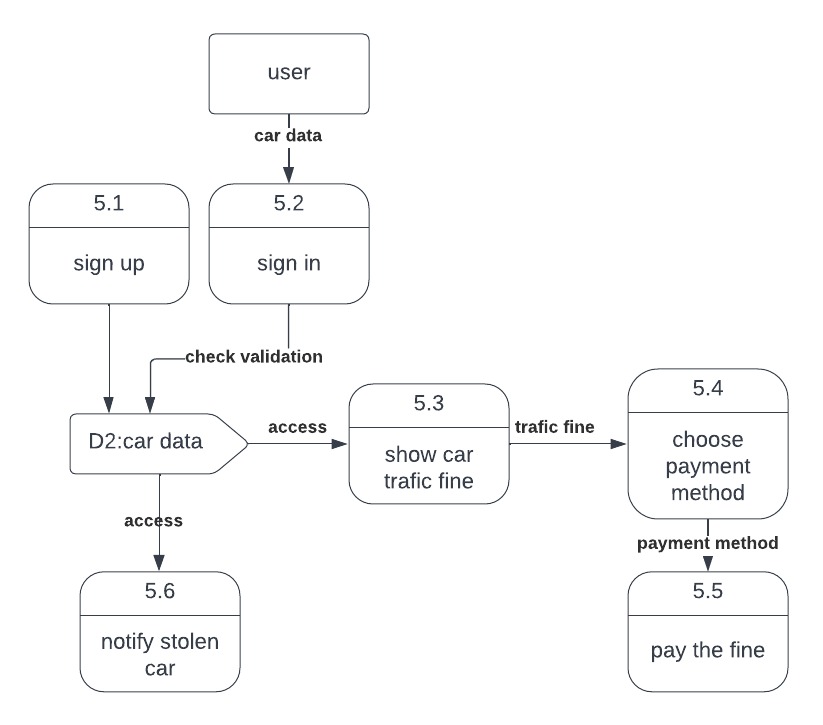


4.3.4 Data Flow Diagram:

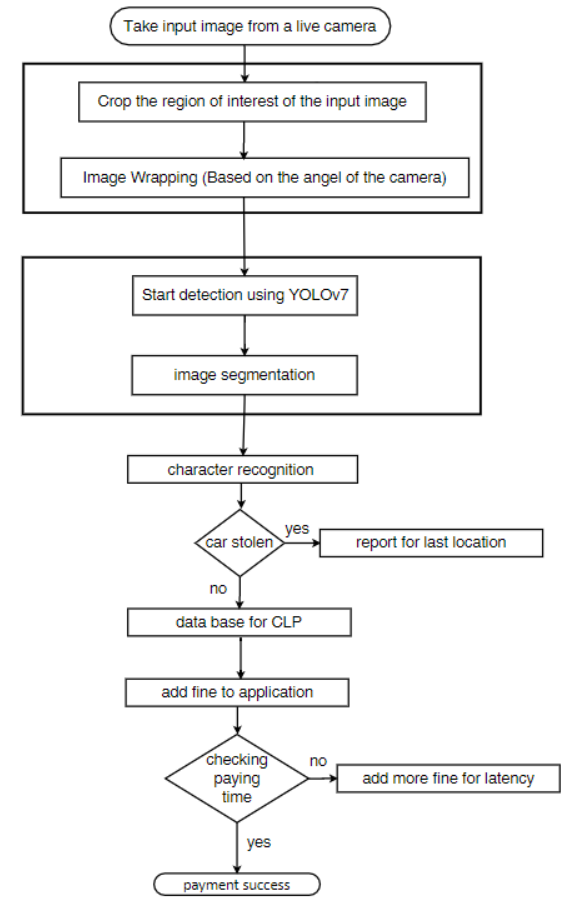
System Flow:



Application flow:



4.3.5 Workflow Diagram:

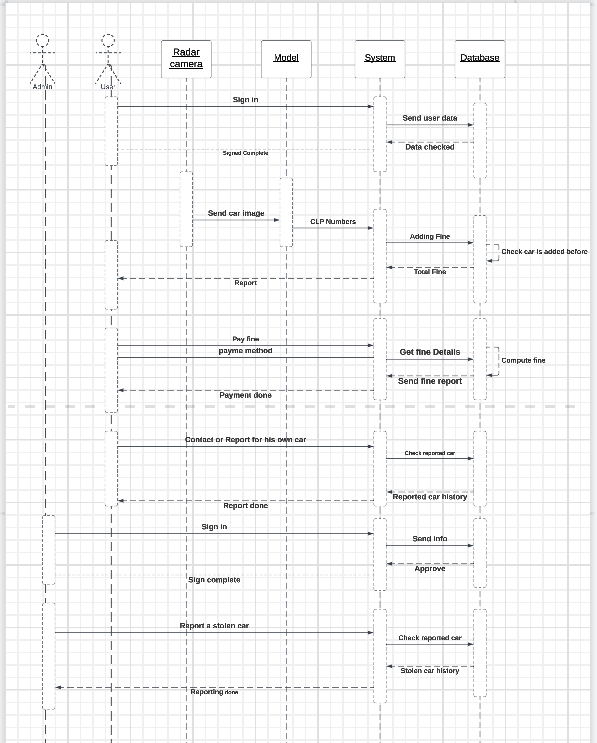


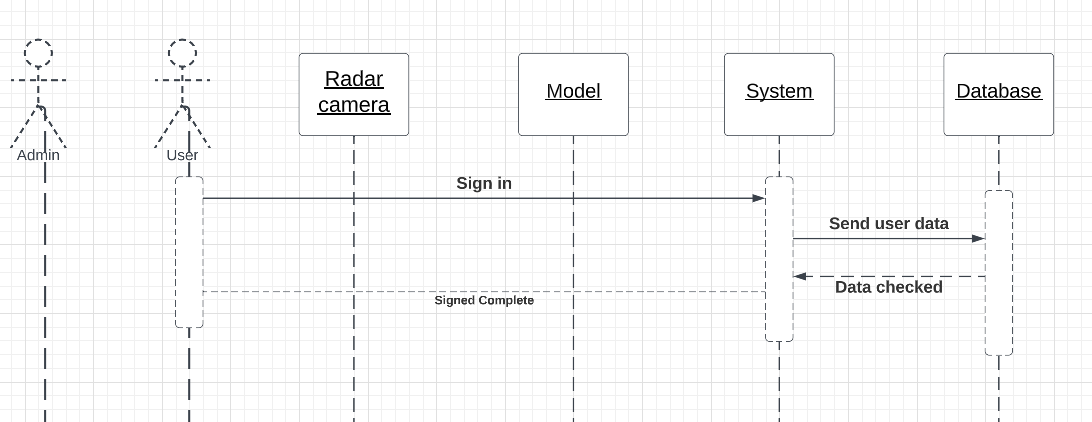
**4.4 Design:**

**4.4.1 System Sequence Diagram:**

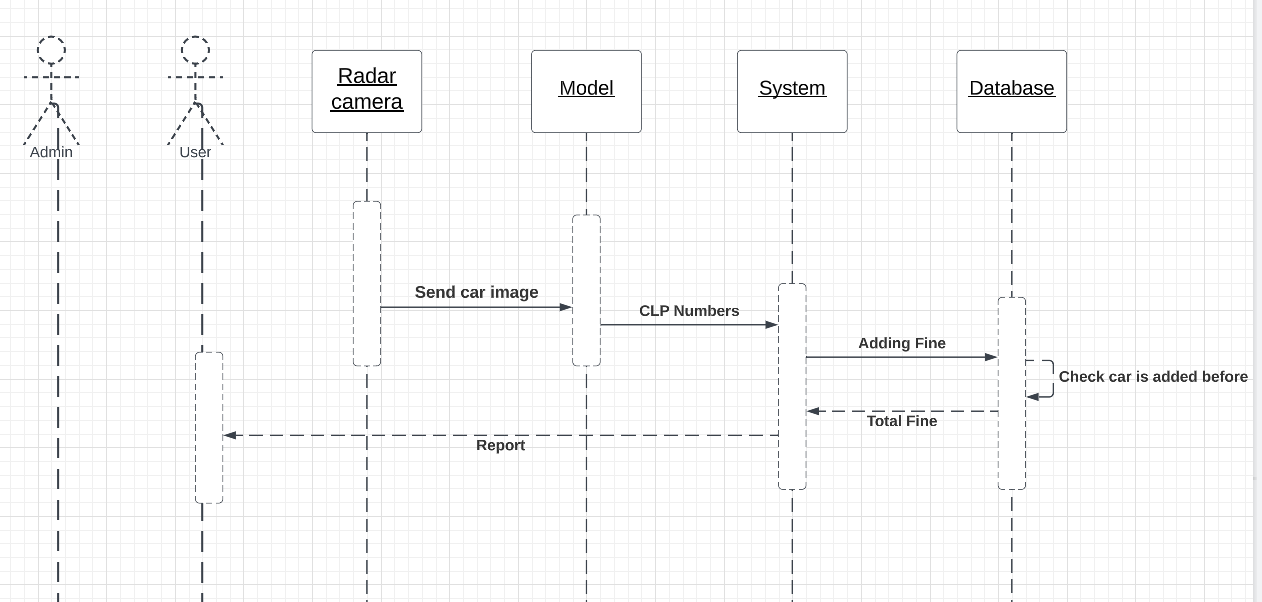
A sequence diagram is a type of interaction diagram because it describes how -and in what order- a group of objects works together. These diagrams are used by software developers and business professionals to understand requirements for a new system or to document an existing process.

In the following section we display Sequence Diagram of system:

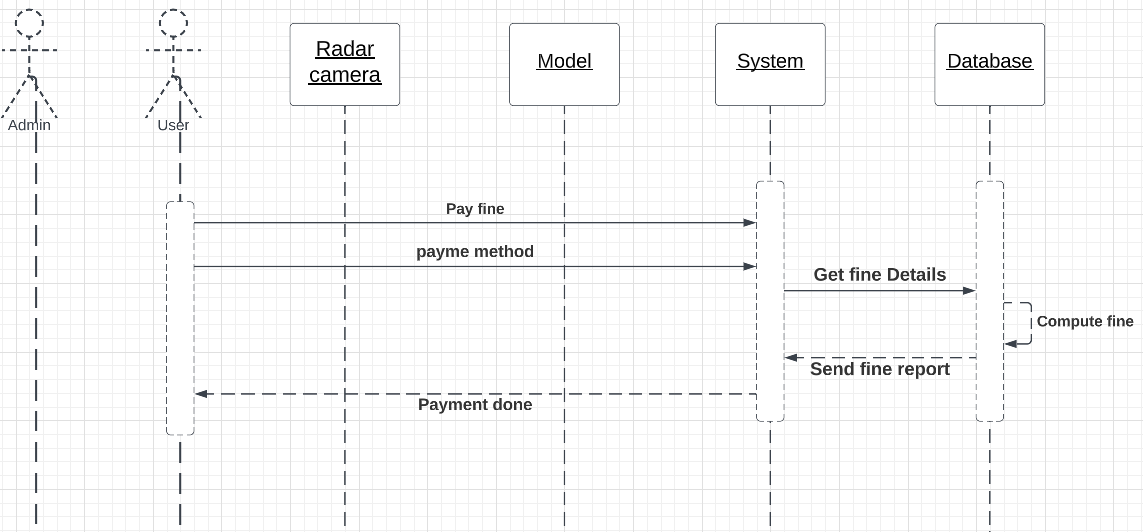




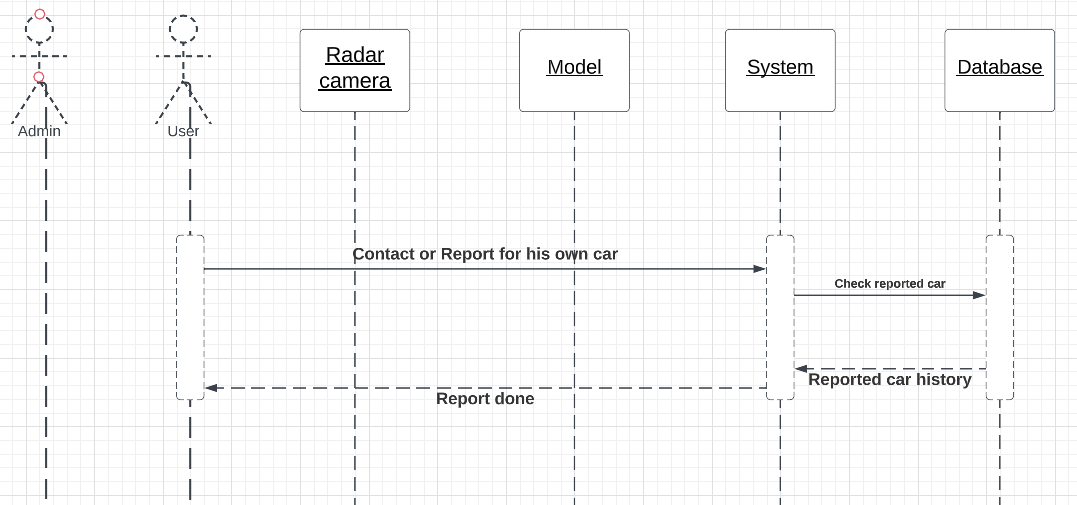
User can enter the system by signing in with his information as he will send the info to the system then the system sends it to the database to check if this data is true or not and whether this user is in the system or not.



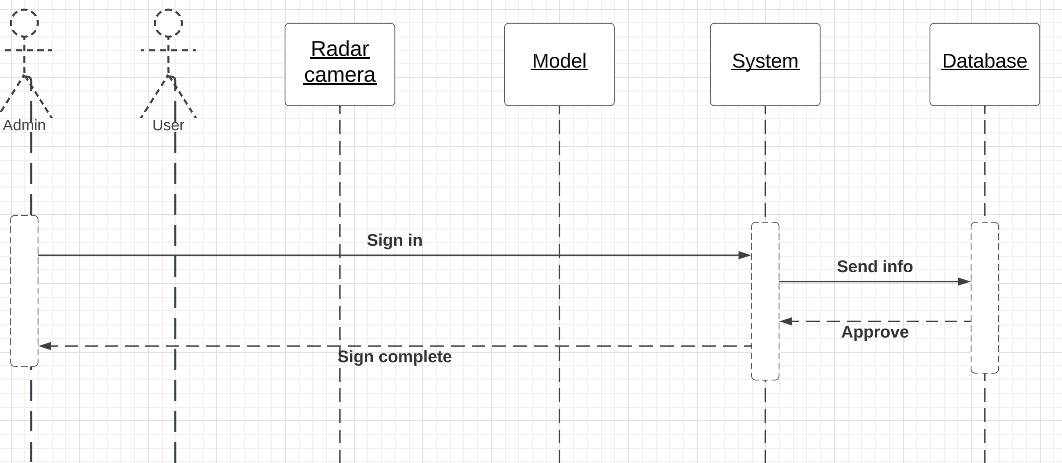
Radar Camera sends the photo that taken from the gate to the model, the model will be able to get the photo in addition to doing some processes to it and the the model detect the car plate license numbers moreover send it to the system to detect the user information and adding the suitable fine to him then store it into the database, finally the system will send report contains the fine and details to the user.



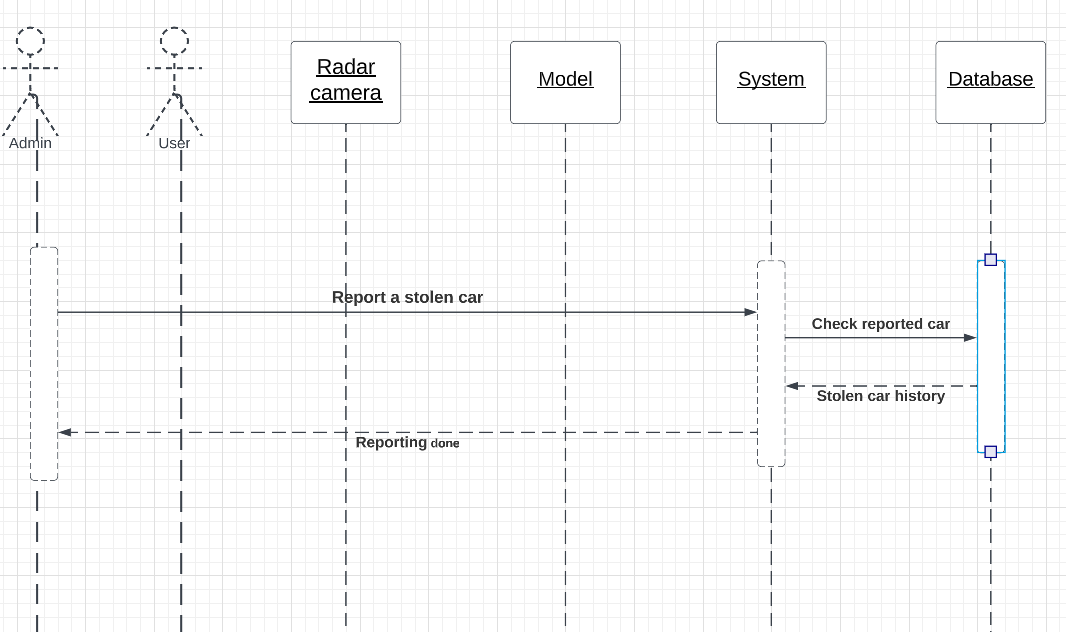
User can pay his fine py enter the system and send a request of paying and choose the payment method that he want, system send a request to the database to get the fine details, then the database compute the total fine and send it to the system, after that the user can complete the payment successfully.



If any user has any problem with his account or his car stolen, the user able to send a report to the system and his problem will be solved.



Admin of the system can sign up into the system and create any account by sending the information and then the info will be stored in the database.



Admin send a report to the system if there is a stolen car into the system.