Table of contents

Chapter 1-introduction---------------------------------------------------------------------------------------------- 4

1.1-introduction-------------------------------------------------------------------- 4

1.2 –problem statement --------------------------------------------------------- 4

1.3-objective------------------------------------------------------------------------- 4

1.4- Project scope------------------------------------------------------------------- 4

1.5-output ----------------------------------------------------------------------------- 5

1.6-documentation organization ------------------------------------------------ 5

1.7-requirements------------------------------------------------------------------- 5

1.8 functional requirements ---------------------------------------- ------- ------5

1.9-non functional requirements ---------------------------------------- ------- 6

Chapter2 –chant chart--------------------------------------------------------------------- 8

Chapter 3- related work----------------------------------------------------------------------24

3.1-overview------------------------------------------------------------------ ------25

3.2-general background-------------------------------------------------------- 25

3.3-deep learning and traditional machine learning--------------------- 25

3.4 deep learning and machine learning table--------------------------- 29

3.5 –comparison with related work----------------------------------------- 32

3.6-references--------------------------------------------------------------------32

Chapter 4-prosed system and all diagrams----------------------------------------------33

4.1-context free diagram-------------------------------------------------------34

4.2-data flow diagram--------------------------------------- -------- 35

4.3-use case diagram--------------------------------------------------- 38

4.4-sequence diagram-------------------------------------------------- 45

4.5-work flow diagram------------------------------------------------- 49

Chapter 1

---------------------------------------------------

Introduction

Introduction:

Digital transformation has become one of the things that countries seek to implement in various fields, as it saves time and effort. Egypt is also laying out Egypt's 2030 vision for digital transformation, so we decided that the graduation project would be beneficial to the country and serve society. It is a system for identifying cars and collecting fees automatically through a program or a website without the need for toll gates on the road. But with a camera that captures the car and through an artificial intelligence model, we will recognize the car and add the fees to the account of the car owner. This system can be used for many services such as paying fines, taking care of stolen cars, and others. The owner of the car is not obligated to pay the fees at the same time as the traffic, but he has a time determined by the Traffic Department to pay, and he is not obligated to pay for himself, but he can activate the automatic payment feature, and the car can be photographed and show all the data of the car and the data of the owner of the car, and this feature is one of the powers of the traffic police and the traffic police will help So much so that you don't need to see it on your driver ID or driver's license

Problem statement:

Road gates force cars to stop to take cash, and some days the road is crowded, and these gates are the cause of road congestion. This causes problems for many citizens because they are late for their important appointments. And we do not need such procedures that we can replace with an automated system. The state also imposes construction and operating costs and salaries of employees and soldiers for protection. In cases of car theft, we cannot identify them

objective:

After the automatic toll collection system is established, cars do not need to stop and there will be no congestion on the road. We can identify stolen cars. We can also integrate this system into state agencies to track and trace stolen vehicles. This is an important step in implementing the country's digital transformation plan. All we need is a camera on the road instead of huge gates

**Project scope**

-There is a **model** that takes a picture of the license plate and starts doing some operations on it, and in the end, the model recognizes the car.

* There is an **application** from which fees are paid, and I can also enter the car plate number to determine the violations related to the car and detect cases of theft, the person can download this application on his mobile phone, and thus the wasted time for payment has been reduced

**Output**

Creating an automation system that contains an application

* fees are paid through it.
* Detecting car violations.
* Knowing the location and time of the car that passes through the system to help in cases of theft.

**Documentation organization**

* **Chapter 1 (**Introduction**):** this chapter contains the introduction to the project (Car Palate reorganization) and discusses the problem that was wanted to solve it also talks about the objectives and scope of the project and finally talks about the output of the system and Documentation organization.
* **Chapter 2 (**Survey**):**  this chapter contains the Overview of the project, also how this technology works and the general background of the project, and how the application differs from other similar applications, add comparison tables for the technology that we relied on to determine the appropriate tools.
* **Chapter 3 (**Analyses and Design**):** this chapter contains project methodology, input, and output system, system specification also has a system framework, and System Analysis (context diagram /DFD diagram/workflow diagram /use case diagram )

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:** thesystem 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:***

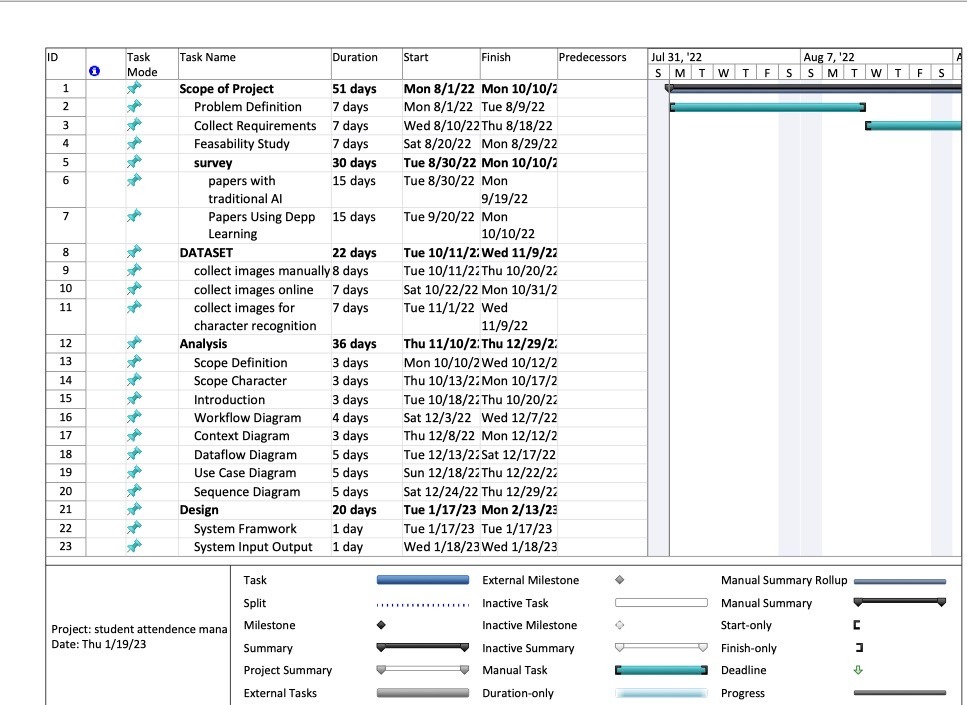
A non-functional requirement (NFR) is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviors. Non-functional requirements are often called the "quality attributes" of a system.  
Non-Functional requirements such as:

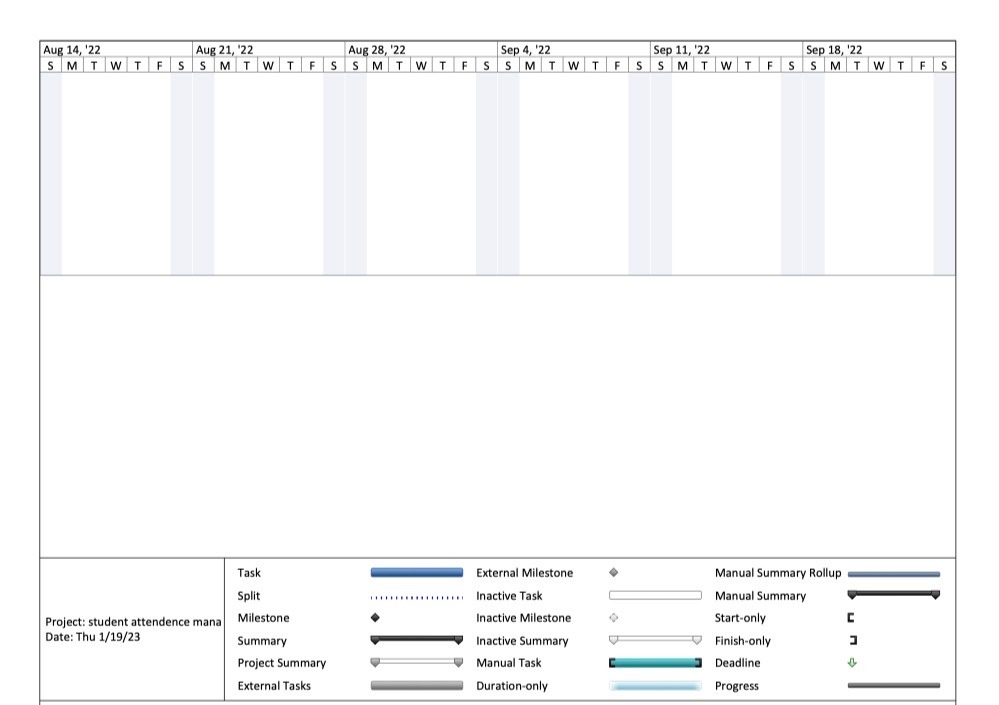
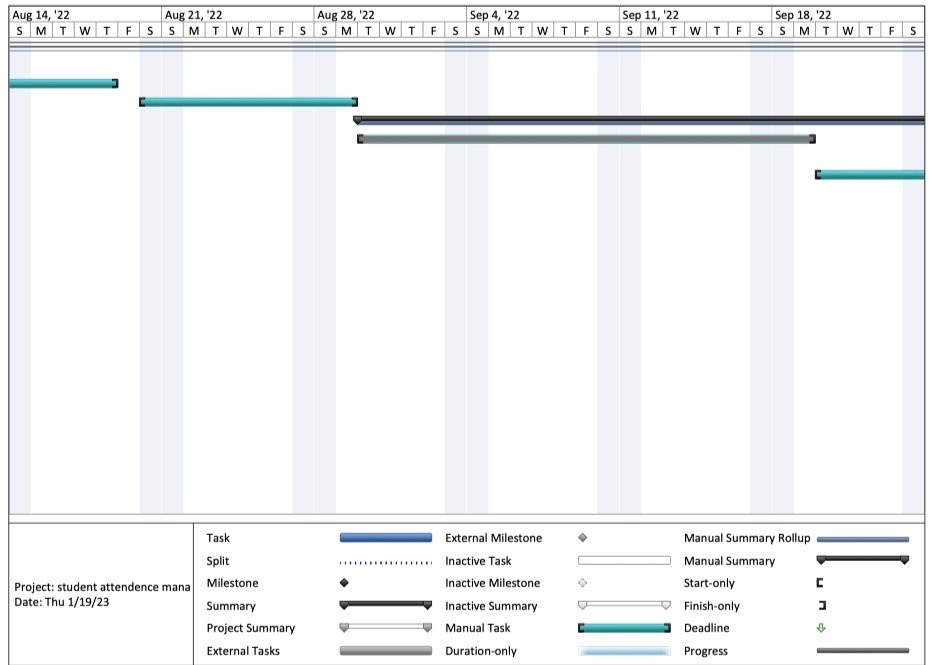
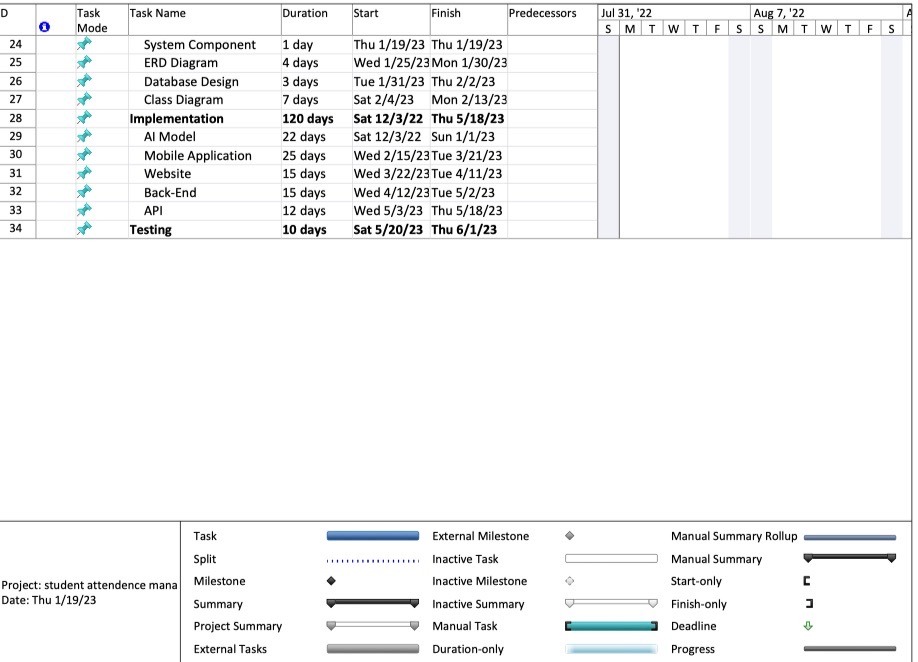
* **Appearance &usability:** simple look and easy-to-use application
* **Availability:** the system must be available at any time
* **Response:** the system is faster in to respond of requests of the user
* **Security:** users’ information should be secure
* **Operational & environmental:** the application 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

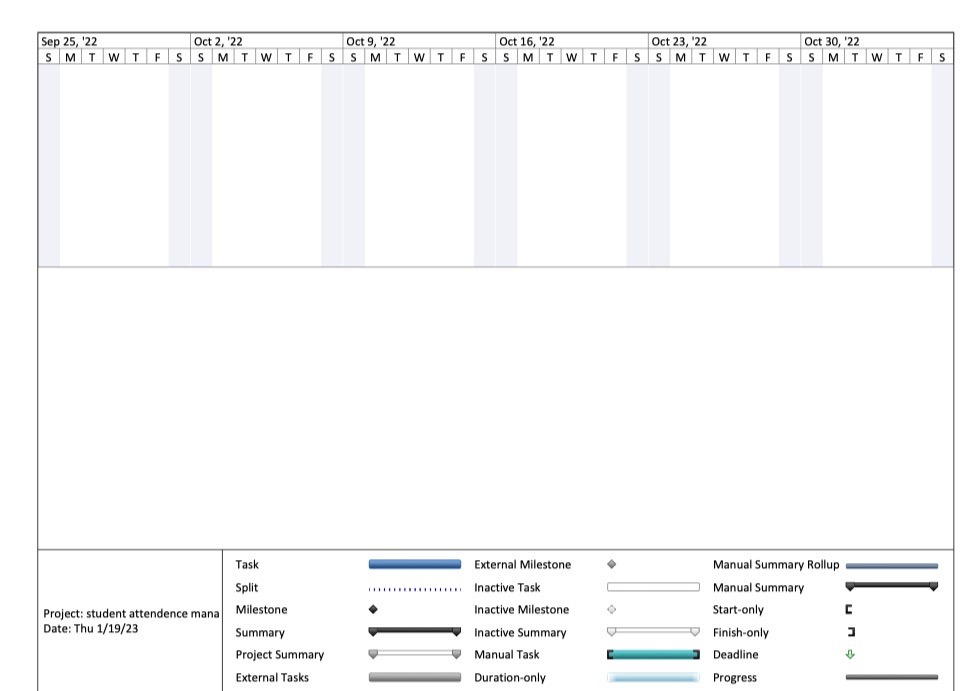
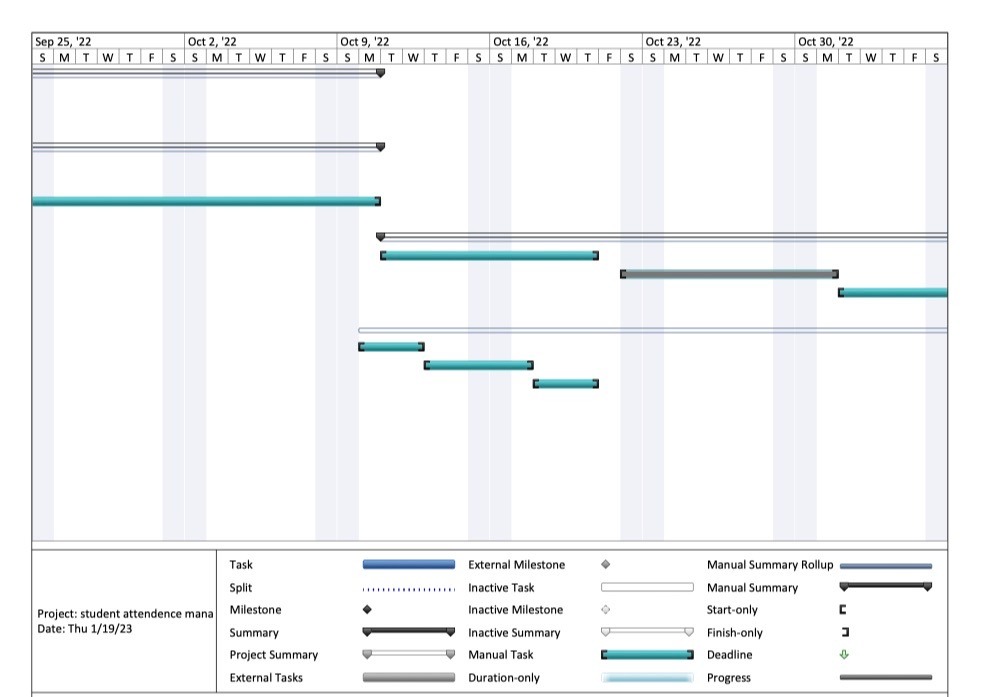
Chapter 2

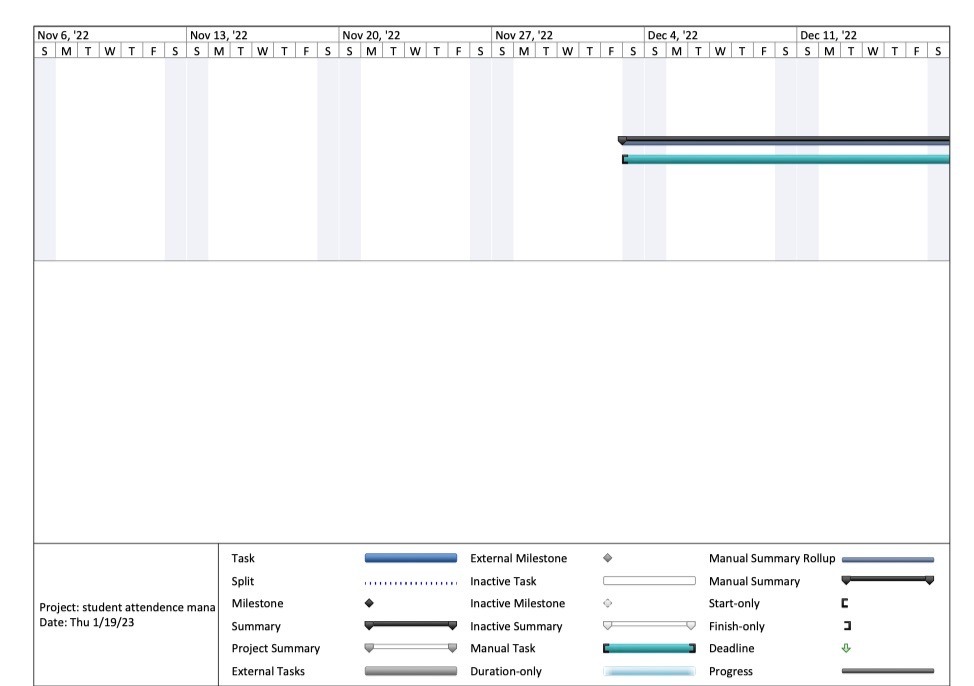
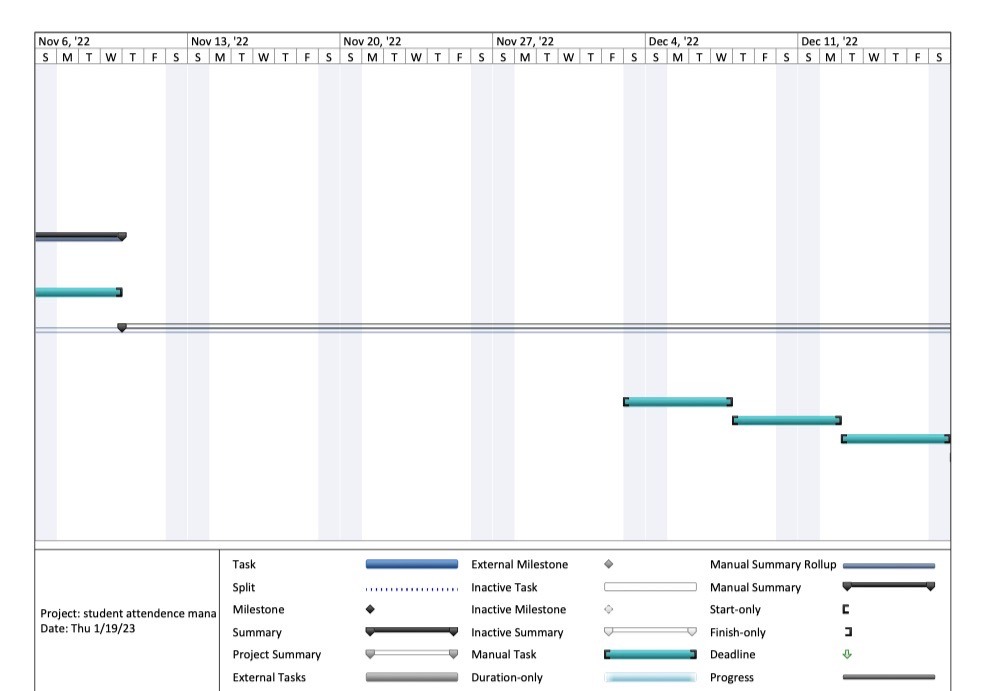
--------------------------------------------------

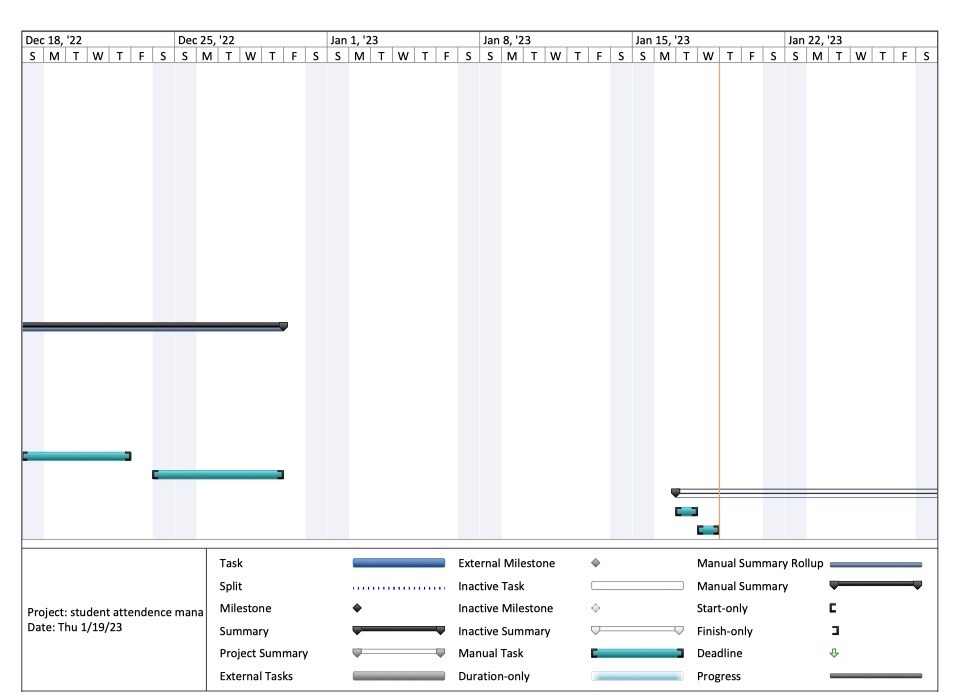
Gant chart

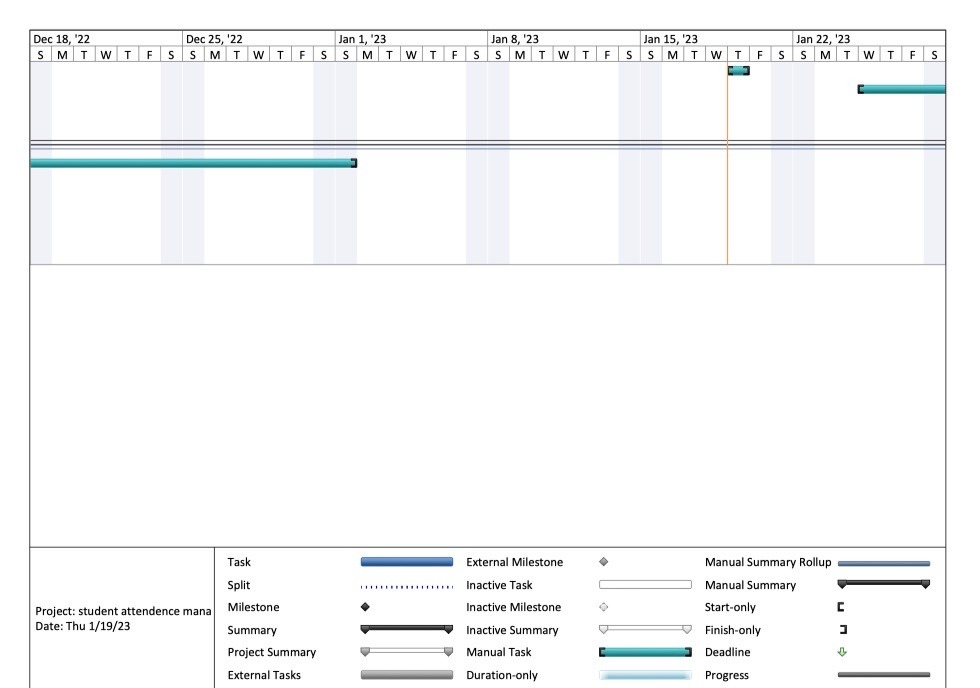


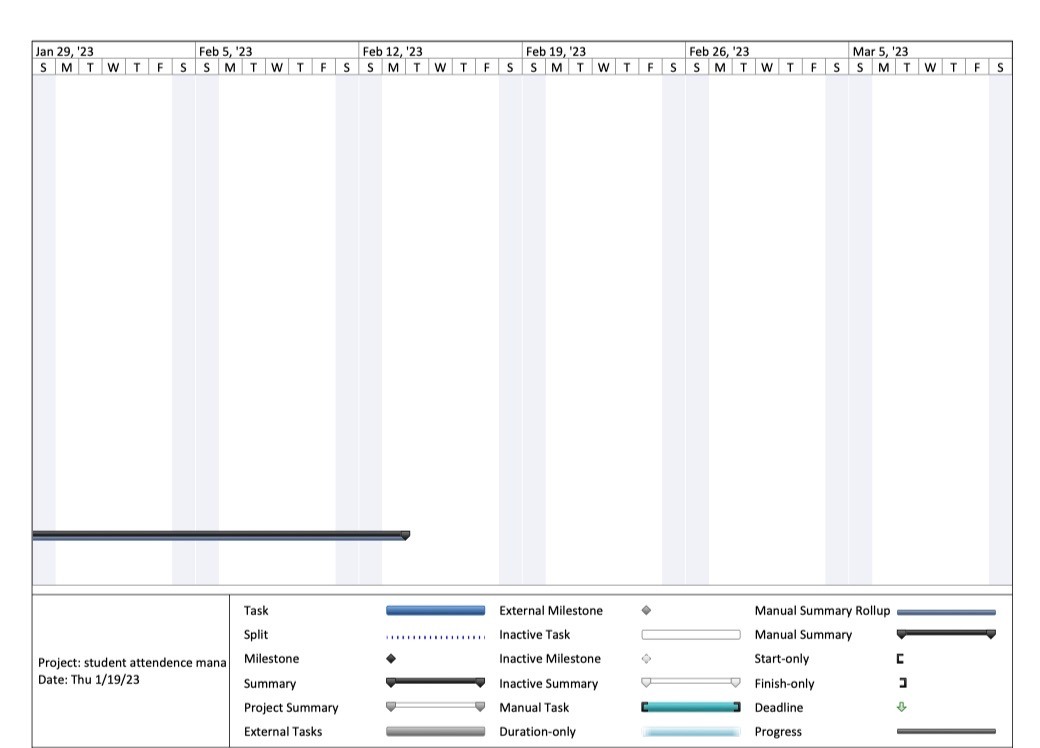


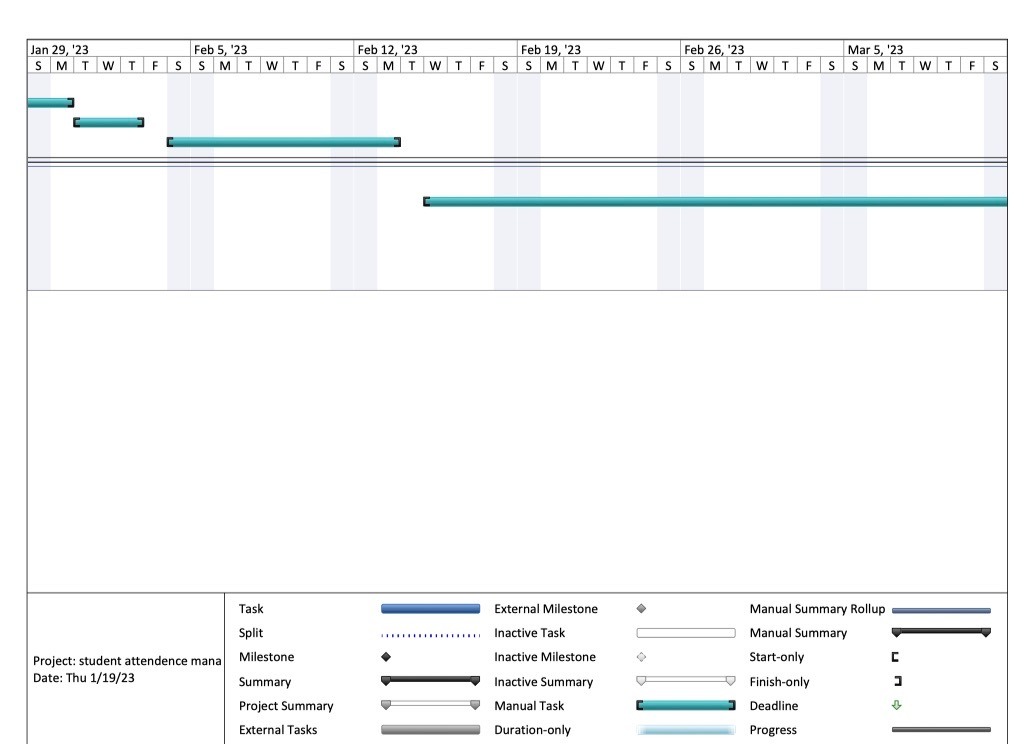


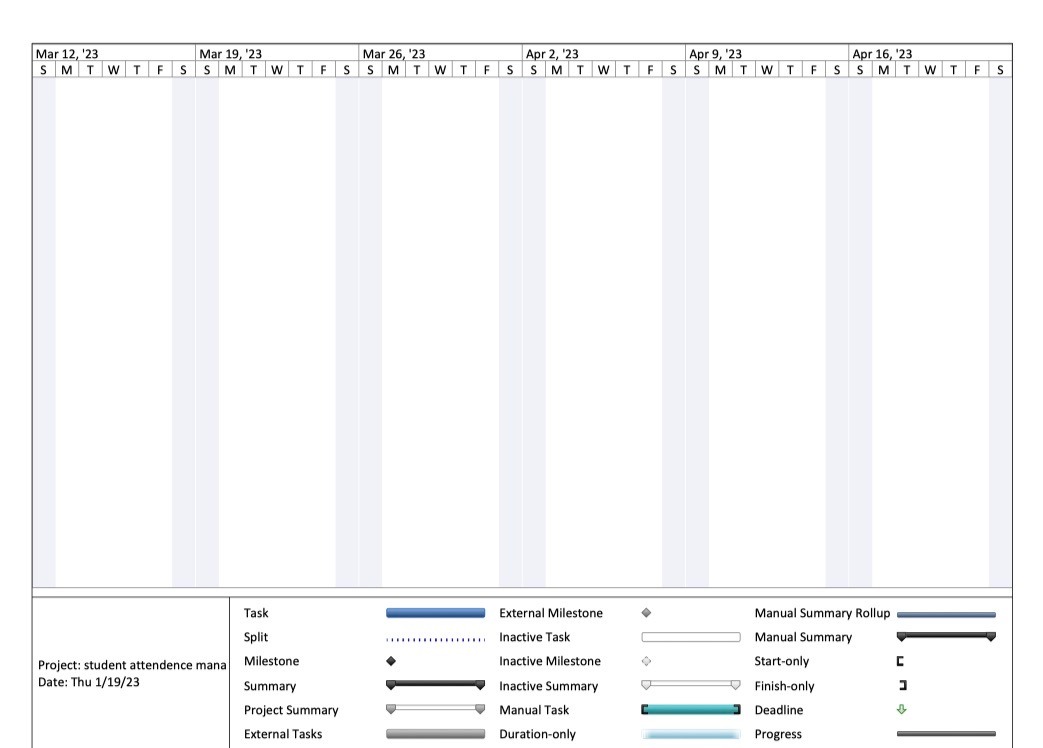


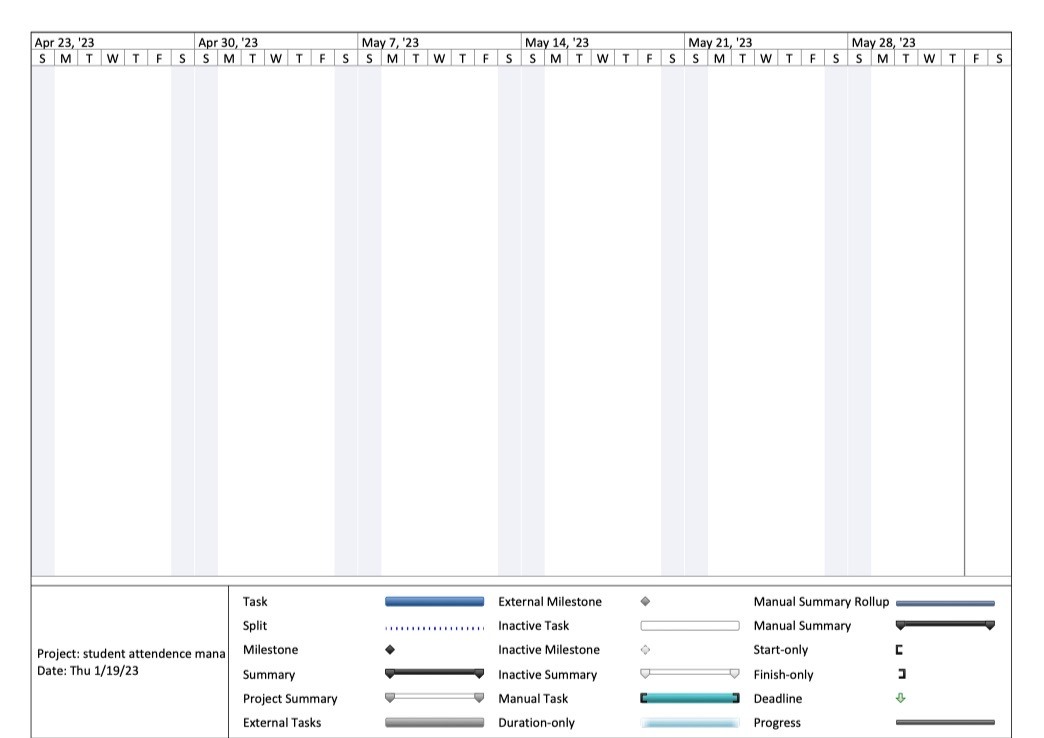


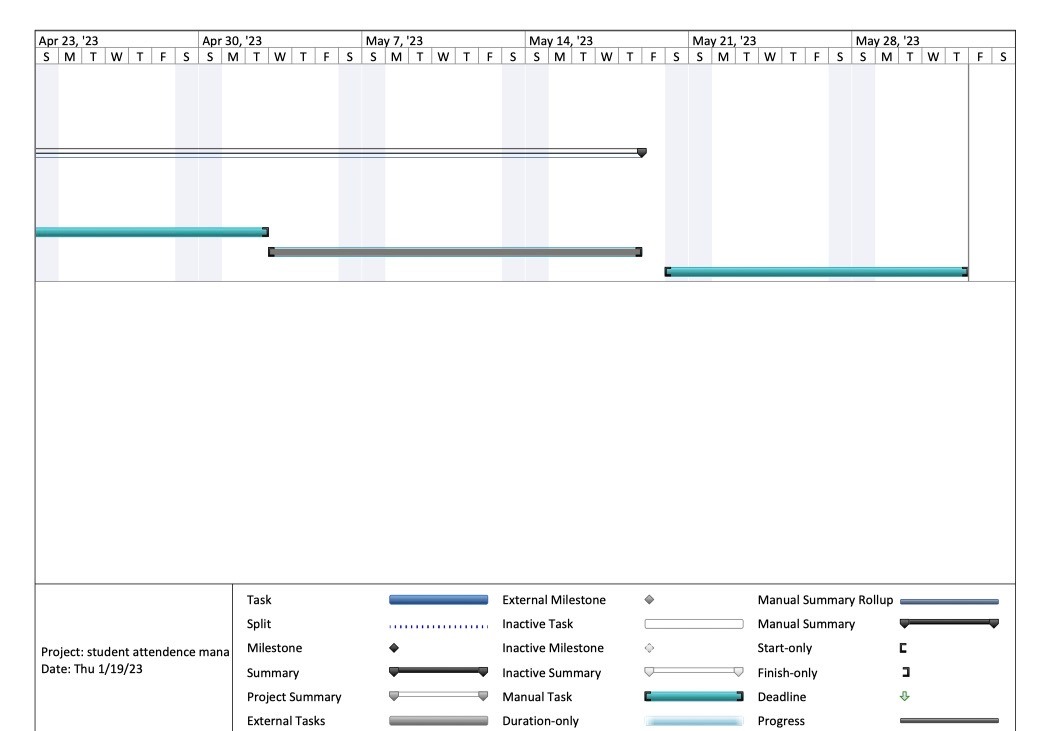












Chapter 3

------------------------------------------------------------------------------------------------------

Related work

**3.1 Overview:**

In this chapter we are talking about the general idea of the project and show the related work researches which are divided into ML and DL, and a table for all researches to describe the important points.

**3.2 General Background:**

The highway fees are taken from cars at the gate which is on the way, that gates waste too much time of the car because every car have to stop at the front of the gate and pay the fee then go, that process make a traffic jam at the gate because of the cars row.

The system we build uses cameras to capture images of cars in the highway then extract the car license plate number and add the fee of the way to the car account on the application that provide the owner to pay from his credit card or any payment method, that will save time of cars on the highway and let them drive at usual speed without stopping.

**3.3 Deep Learning and Traditional Machine Learning:**

**3.3.1 Traditional Machine Learning:**

Ahmed Abdelmoamen Ahmed et al. in [[p1]](file:///C:\Users\Mahmuod\Downloads\chapter3last%20version(survay).docx#p1) proposed automatic number plate recognition (ANPR) which automatically recognizes vehicle number plates using high-speed cameras. After capturing the image, it was converted it to gray scale, then a threshold function was applied for image binarization, followed by, Connected Component Analysis (CCA) for image post processing. Moreover, an image segmentation process was utilized to separates the relevant objects in the image. Finally, k nearest neighbor (KNN) was used as a classier to recognize the plate’s actual characters. The proposed model used more than 160 car images and the model classification accuracy was 95% on testing dataset, the model required around 10 ms only to detect the actual characters in the number plate. The overall processing time 29 ms. the model failed to recognize some characters correctly and confuses characters with similar structures such as D with O, 6 with 8, K with I, etc.

Ayman M. Hassan et al. in [[p2]](file:///C:\Users\Mahmuod\Downloads\chapter3last%20version(survay).docx#p2) proposed a car plate recognition system based on Support Vector Machine (SVM) for a vehicle building entrance. The proposed system consists of two gates (entrance - exit). Firstly, an ultrasonic sensor detects the vehicle existence, then a command to a camera to capture vehicle is sent to an Arduinoboard to control vehicle gates. After image preprocessing, both vertical and horizontal edge detection were used to detect car plate, followed by connected-component labeling to separate each number. Finally, histogram of oriented gradient (HOG) and SVM were used as features and classifier, respectively, to recognize the numbers. A total of 100 car images (70 training – 30 testing) were used to train and test the proposed system. Based on the depicted results, the proposed system achieved an overall accuracy of 83.3% on testing dataset with an overall processing time of 5 seconds. The proposed system failed to localize some car plates or failed in one of the three stages in recognizing plate numbers because the digits were ambiguous.

Manal Alghamdi et al. in [[p3]](file:///C:\Users\Mahmuod\Downloads\chapter3last%20version(survay).docx#p3) presented automatic number plate recognition model of Saudi License Car Plates to solve the problems of traffic regulation and traffic violations facing the society. After the image capturing, the image was preprocessed via a set of steps 1) image conversion to gray scale, 2) image resizing, scaling, and rotation, 3) removing high-frequency noise using Gaussian blurring. Then two threshold image segmentation was used to separates the relevant objects in the image. Finally, optical character recognition (OCR) technology was utilized to turn printed words of the preprocessed images into editable text. The used dataset contains 50 images of various parked cars. The proposed model achieved an accuracy of 92.4% for Arabic and 96% for English texts. The car plate size and location may effect on the performance of the model.

Nur Liyana Yaacob et al. in [[p4]](file:///C:\Users\Mahmuod\Downloads\chapter3last%20version(survay).docx#p4) License plate recognition for campus auto gate system to the university gate. License plate localization: image enhancement, crop and convert it to gray scale, Sobel edge detection, erosion, image closing, image banalization, and locating license plate region. Character segmentation: morphological operations, skew correction, row detection and character segmentation. Character recognition: template matching method is utilized in the character recognition stage. Data set is a total of 95 images were used as the input images, the success rate for the license plate localization, character segmentation and character recognition are 93.68%, 93.25% and 84.33% respectively. the license plate localization performance will be affected if the vehicle has text stickers on the vehicle body or vehicle model written on the front body and if the vehicle has a complex background, for the character segmentation stage, the performance will be affected if the characters are too close to each other, In the character recognition stage, there are some misclassification occurred. Because of the resemblances between the characters.

Fattah Alizadeh and SazanLuqman. in [[p5]](file:///C:\Users\Mahmuod\Downloads\chapter3last%20version(survay).docx#p5) proposed an automatic car plate recognition (ACPR) system that automatically recognizes vehicle number plates using high-speed cameras. After capturing the image, the image was resized to 200\*200 pixels, converted to a gray scale and binary one, and then the de-noised. The proposed system utilized Speeded up Robust Features (SURF) descriptor as features for separating the car plate region from the whole image. Then, an image segmentation process was applied to separate the relevant objects in the image, and then the underlying descriptors of the digits later will be compared to those which have already been saved in the dataset. The minimum distance between the target image and the saved ones indicated the digit. Dataset of 200 images which was categorized into 5 groups, in terms of distance from the camera, view angle, alignment, plate color, and weather condition was used to evaluate the proposed system. The images were taken in different locations of the region. The proposed model achieved an accuracy of 94% on testing dataset, when the car plate is 1 Meter far from the camera, the highest accuracy achieved is 94%, but for the 4-meter distance, the localizing and recognition is not satisfactory at all (27%), so the model accuracy depends on the distance between the car and camera in addition to the weather.

**3.3.2 Deep Learning:**

Kuken Raj Pugalenthy et al. in [[p6]](file:///C:\Users\Mahmuod\Downloads\chapter3last%20version(survay).docx#p6) proposed using computer vision to recognize the car plate registration number in order to monitor the vehicle movement. When capture the car image they use NumPy slicing to get right region of interest and make Image Warping, they use YOLOv3 to detect the position of car registration plate and resize images by OpenCV, NMS technique solve the problem on multiple detection and make image thresholding for color inversion, after that use PyTesseract to read the characters from the image and Regular Expression to eliminate the weak predictions from the PyTesseract results. They didn’t mention the dataset, the model achieved 100% accuracy in recognizing. The car registration number makes harder to look in moving motion. So here, fast operation is needed to capture the picture or also make a video out of the captures. the main scope of this paper is to identify and recognize the vehicle registration license plate under normal weather daylight condition.

Mau-Luen Tham et al. In [[p7]](file:///C:\Users\Mahmuod\Downloads\chapter3last%20version(survay).docx#p7) IoT Based License Plate Recognition System Using Deep Learning and Open VINO to recognize Malaysian car plate is attained via transfer learning. the YOLOv4 trained model’s weights file is inputted into the Open VINO toolkit, Node-RED is chosen as it is a lightweight and convenient flow based editor that uses a web browser to build workflows and MQTT as a cloud, before passing the cropped car plate frame to the Tesseract OCR, convert it to gray scale, blurring, thresholding and dilation to Last, character segmentation is enabled by contour function, we adopt the Tesseract OCR for image to-text conversion. The dataset consists of a total of 500 Malaysian car images, which are divided into 80% for training and 20% for validation of the detection model. YOLOv4, together with Tesseract OCR recognize accuracy with 100% detection accuracy and 89.22% OCR accuracy, but High similarity in numbers and shapes may effect on the performance.

TaoufikSaidani et al. in [[p8]](file:///C:\Users\Mahmuod\Downloads\chapter3last%20version(survay).docx#p8) proposed an automatic license plate recognition system, which is an intelligent system based on analyzing visual data to detect and recognize characters in license plates. The proposed system based on the Faster R-CNN improved by adding an adaptive attention network for the segmentation of the license plate to retrieve the numbers and the letters of identification. Moreover, a deconvolution layer at the top of the features extraction network was added to detect the small size of the target license plate. The anchor sizes, scales, and aspect ratios were modified to fit the license plate, speed up the processing speed to achieve real-time conditions by using the Inception v3 model. These plates are from Egypt, KSA, and UAE, because they are similar in their Arabic numbers. The model fails to recognize the plates in difficult weather times of rain and sandy winds, as well as in low light. The model has been developed to be able to process images at speeds up to 23 FPS. the model achieves a recall of 98.65 % and a precision of 97.46 %. We couldn't get dataset public on the internet and I collected it from the streets and python script then it was manually filtered and labeled and it has a size of 6000 images.

Sergio M. Silva and CláudioRosito Jung .in [[p9]](file:///C:\Users\Mahmuod\Downloads\chapter3last%20version(survay).docx#p9) proposed an Automatic License Plate Recognition (ALPR) that automatically recognizes vehicle number plates using high-speed cameras. After capture the image resizing it to 320 × 320 pixels, converting the color image into a grey scale and binary one, and then the de-noising process, use Improved Warped Planar Object Detection Network (IWPOD-NET) that is able to detect the four corners of an LP in a variety of condition from the whole image. They trained two distinct networks, one to detect cars and LPs jointly, and another to perform Optical Character Recognition OCR, for classification they use two distinct CNNs then using YOLOv3 for recognition the car plate. This network was trained with a small amount of annotated data (only 693 images), and showed remarkable generalization to handle both car and motorcycle LPs capture data variety of illumination conditions and viewpoints. tolerances in the string edit distance. For some datasets, the system was not trained to capture strong geometric distortions. The system achieves 100% accuracy, which is also an indirect clue that the LPD module can correctly locate the LP.

**3.4 Deep Learning and Machine Learning (Table [T1]) :**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Study** | **Objectives** | **Methodology** | **Classifier** | **Dataset** | **Performance Measures** |
| [**P****1**](file:///C:\Users\Mahmuod\Downloads\chapter3last%20version(survay).docx#r1) | enhance the accuracy of recognizing number plates in real time & time of processing | 1) Using high-speed cameras to capture the car.  2) Convert the image into grayscale one.  3)  Use Threshold function to be a binary image.  4) Use (CCA) for image post processing.  5) Image segmentation process to separate the relevant objects in the image.  6) Enters the feature vector to (KNN) classier to recognize the plate’s actual characters. | KNN | Kaggle & Google Web Scraper (160 Image) | The model classification accuracy is 95% |
| **[P2](file:///C:\\Users\\Mahmuod\\Downloads\\chapter3last%20version(survay).docx" \l "r2)** | Design a model for a vehicle building entrance using a license plate recognition algorithm | 1) An ultrasonic sensor detects the vehicle then send a command to a camera to capture vehicle.  2) An Arduinoboard to control vehicle gates.  3) Image Preprocessing to prepare the image (rgb2gray2binary).  4) Use edge detection-vertical and horizontal edge histogram projection to detect car plate.  5) Use connected-component labelling to separate each number.  6) Use (HOG) and SVM to recognize the numbers. | SVM | \_ | The model overall accuracy is 83.3% with 30 testing images and 70 training images. |
| **[P3](file:///C:\\Users\\Mahmuod\\Downloads\\chapter3last%20version(survay).docx" \l "r3)** | Recognition of Saudi car plates with the use of edge  detection, segmentation, and contouring techniques | **CNN** to detect the language and the country of the LP | Binary K-means | 50 images of various parked cars. | Results in an accuracy of 92.4% for Arabic and 96% for  English texts |
| **[P4](file:///C:\\Users\\Mahmuod\\Downloads\\chapter3last%20version(survay).docx" \l "r4)** | Designed to work in an IR sensor is used to detect the motion of the car and the camera will be triggered to capture the vehicle image | License plate detection (LPD)  method | Template matching | 95 images with an 8MP iPhone camera | License plate localization, character segmentation and character recognition are 93.68%, 93.25% and 84.33% respectively |
| **[P5](file:///C:\\Users\\Mahmuod\\Downloads\\chapter3last%20version(survay).docx" \l "r5)** | Create an automatic system to detect, extract, segment, and recognize the car plate numbers in the Kurdistan Region of Iraq (KRI) | Speeded Up Robust Features SURF descriptor | \_ | 200 images. | accuracy up to 94% |
| **[P6](file:///C:\\Users\\Mahmuod\\Downloads\\chapter3last%20version(survay).docx" \l "r6)** | Identify and recognize the vehicle registration license plate under normal weather daylight condition to monitor the vehicle movement | 1) Capture the car image.  2) Use ‘NumPy’ slicing to get right region of interest.  3) Image Warping.  4) Use (YOLOv3) to detect the position of car registration plate.  5) Resize images by OpenCV.  6) Use (NMS) technique for multiple detection.  7) Image thresholding for color inversion.  8) Use (PyTesseract) to read the characters.  9) Use (RGX function) to eliminate the weak predictions. | \_ | \_ | The model achieved 100% accuracy in recognizing. |
| **[P7](file:///C:\\Users\\Mahmuod\\Downloads\\chapter3last%20version(survay).docx" \l "r7)** | Identify and Recognition of  License Plate Recognition System Using Deep  Learning | **CNN-** learned features.  **HOG** feature for detection. | KNN | 102  images | accuracy with 100% detection  accuracy and 89.22% OCR accuracy |
| **[P8](file:///C:\\Users\\Mahmuod\\Downloads\\chapter3last%20version(survay).docx" \l "r8)** | Detect and recognize characters in license plate | - Developing an automatic license plate detection system (ALPD) based on the Faster  R-CNN model.  - Improving the performance of the Faster R-CNN model by adding the adaptive attention network and the deconvolution layer.  - Modifying the anchor sizes, scales, and aspect ratios to fit the license plate.  - Speed up the processing speed to achieve real-time conditions by using the Inception v3 model. | \_ | 6000 images  Collected by python script and manually filtered | Recall of  98.65 % and a precision of 97.46 % |
| **[P9](file:///C:\\Users\\Mahmuod\\Downloads\\chapter3last%20version(survay).docx" \l "r9)** | an automatic system to detect, extract, segment, and recognize the car plate numbers | -Improved Warped Planar Object Detection Network (IWPOD-NET) that is able to detect the four corners of an LP in a variety of condition from the whole image.  -YOLOv3 for recognition the car plate.  - CNN | CNN | 693 images | Accuracy 100% |

**3.5 comparison with related work:**

Salik in Dubai is characterized by allowing vehicles to pass without having to stop in front of gates or stations to pay fees, vehicles

cross the highway at its usual speed without stopping.

The Abu Dhabi Darb application, via smart phones and computers, provides an integrated set of interactive maps and several services.

**3.6 references:**

1. [-Ahmed, Ahmed Abdelmoamen, and Sheikh Ahmed. "A Real-Time Car Towing Management System Using ML-Powered Automatic Number Plate Recognition." *Algorithms* 14.11 (2021): 317.‏](https://www.mdpi.com/1999-4893/14/11/317)
2. [-Hassan, Aeyman M., Sami A. Ghoul, and Aya A. Alkabir. "Libyan Vehicle License Plate Recognition with Support Vector Machine." *Al-Mukhtar Journal of Sciences* 37.1 (2022): 1-13.‏](https://omu.edu.ly/journals/index.php/mjsc/article/view/525)
3. [-Antar, Rayana, et al. "Automatic Number Plate Recognition of Saudi License Car Plates." *Engineering, Technology & Applied Science Research* 12.2 (2022): 8266-8272.‏](https://www.etasr.com/index.php/ETASR/article/view/4727)
4. [-Yaacob, N. Liyana, et al. "License plate recognition for campus auto-gate system." *Indonesian Journal of Electrical Engineering and Computer Science* 21.1 (2021): 128-136.‏](https://d1wqtxts1xzle7.cloudfront.net/81735078/14518-libre.pdf?1646453585=&response-content-disposition=inline%3B+filename%3DLicense_plate_recognition_for_campus_aut.pdf&Expires=1674145129&Signature=bQ22TdXryrnvHuqX1HwY7BtuHe821mcUetE8YYNRF9LieFc4VO5vwlEY5loRJabl5JesZdr47HWFntTqT4UjbnRSbSexuXv-sCPcwYcxlQUkpCasy5Cn-8vtXRUj9-EvcfqGfZb2dYyqDEjeAeq4-KvRta87MoNcCUvokYWc-tSPKGjBkWrkHC0b38rfIphGymZWgzucJUu7AixtFt4d45KsHUgaedt-VeL1qsIryNHdJFS5i8cjKO4biC004gLP4~YFsylIRAJdDVKswzTQYboTQyyf-Tu6epxERQGxVKd2ArL6c68luzX9yeD~tC5AapJYDkaSxGSqQaEW0BC1Ag__&Key-Pair-Id=APKAJLOHF5GGSLRBV4ZA)
5. [-Alizadeh, Fattah, and Sazan Luqman. "A Novel Approach for Automatic Car Plate Detection and Recognition." *UKH Journal of Science and Engineering* 5.2 (2021): 1-9.‏](https://journals.ukh.edu.krd/index.php/ukhjse/article/view/196)
6. [-Pugalenthy, Kuken Raj, et al. "Malaysian Vehicle License Plate Recognition Using Deep Learning and Computer Vision." *Proceedings of the 6th International Conference on Electrical, Control and Computer Engineering*. Springer, Singapore, 2022.‏](https://link.springer.com/chapter/10.1007/978-981-16-8690-0_88)
7. [-Tham, Mau-Luen, and Wei Kun Tan. "IoT Based License Plate Recognition System Using Deep Learning and OpenVINO." *2021 4th International Conference on Sensors, Signal and Image Processing*. 2021.‏](https://dl.acm.org/doi/abs/10.1145/3502814.3502816)
8. [-Saidani, Taoufik, and Yamen El Touati. "A vehicle plate recognition system based on deep learning algorithms." *Multimedia Tools and Applications* 80.30 (2021): 36237-36248.‏](https://link.springer.com/article/10.1007/s11042-021-11233-z)
9. [-Silva, Sergio M., and Cláudio Rosito Jung. "A flexible approach for automatic license plate recognition in unconstrained scenarios." *IEEE Transactions on Intelligent Transportation Systems* (2021).‏](https://ieeexplore.ieee.org/abstract/document/9357409)

Chapter 4

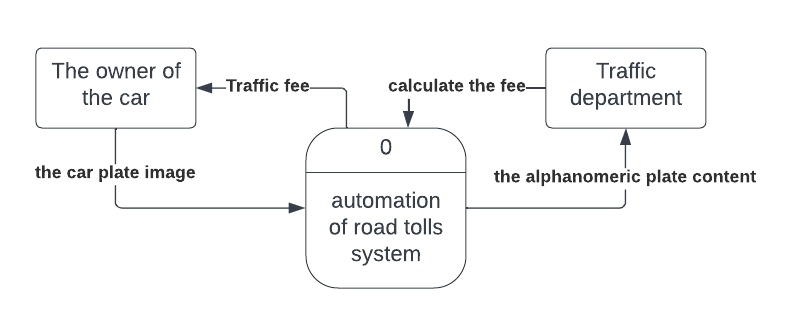
------------------------------------------------------------------

Proposed system and all diagrams

The input of our project will be the photo the comes from the camera that is in the traffic gate, our first model will do some image preprocessing in the image like binary converting or resizing the image, then aggregation in addition to segmentation to get the car plate form the hole car image more over get the car plate content to know the owner of the car and then adding fine to him.

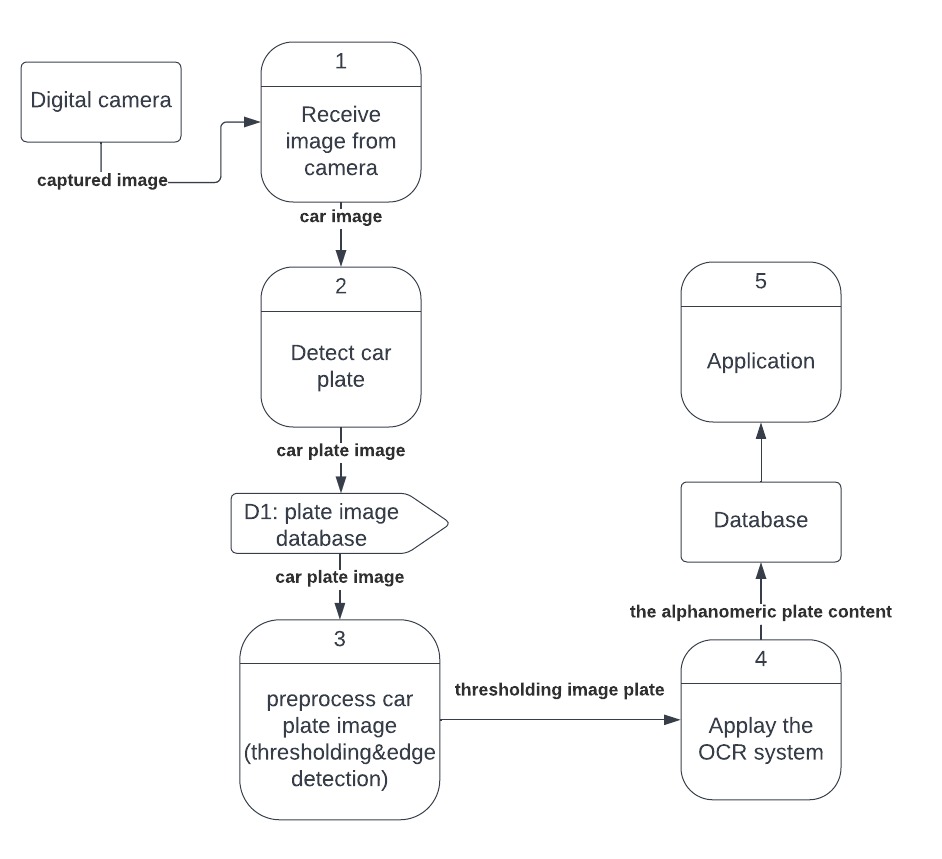
Output of our system will be report to the user contain the details of the user.

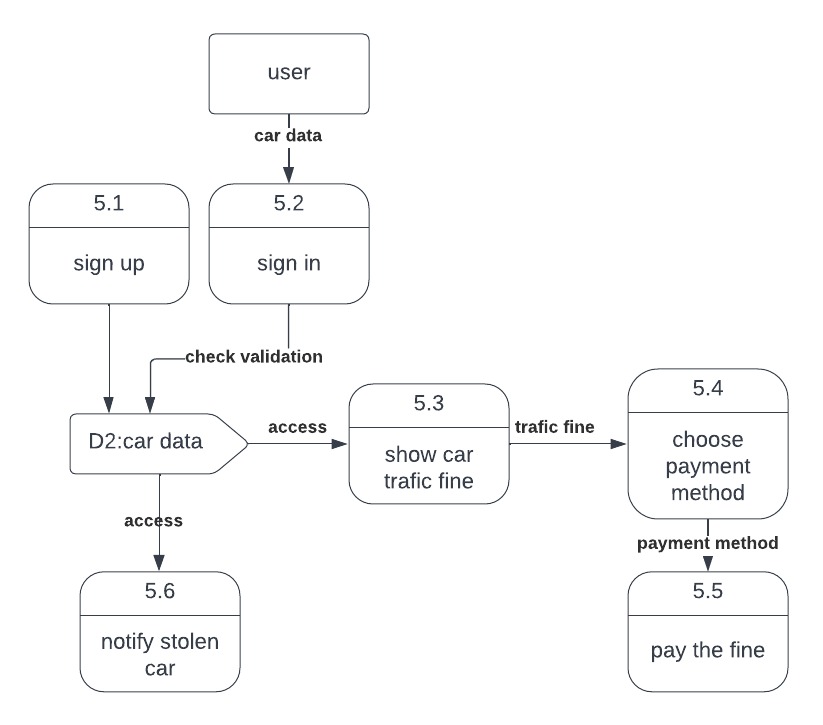
Context free Diagram



Data Flow Diagram

Figure, represents a scenario for data flow between model and application. In this scenario the model takes input image to do its operations for the license plate recognition through detecting the car plate, character recognition. In the application scenario the user signs up if it’s first time, sign in to access his data, pay the car traffic fine, notify if the car is stolen





Use Case Diagram

Use Case Diagram :-

* Use cases represent system functionality from the user`s perspective.
* Use case diagrams describe who will use the system and in what ways the user expects to interact with the system.
* Use case diagrams represent the interactions between use cases and actors.
* Use case diagram represents the interactions between system, external systems, and users.

Use case scenario:-

* a use case scenario represents the sequence of events along with other information that relates to this use case.
* A typical use case specification template includes the following information:-

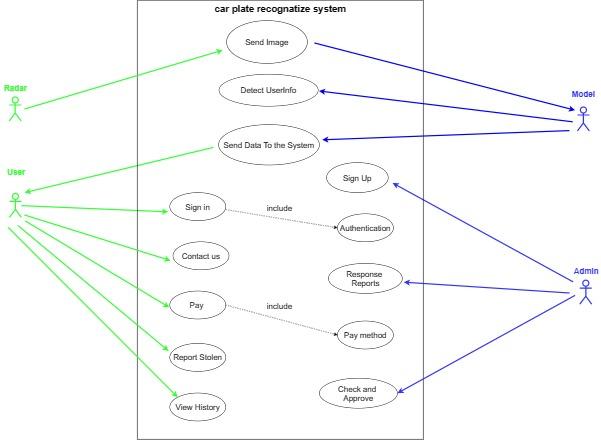
-Primary Actor

(Secondary Actor (if exist

(Include use case (if exist

-Purpose

-Pre and post condition



|  |  |
| --- | --- |
| 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 |

sequence Diagram

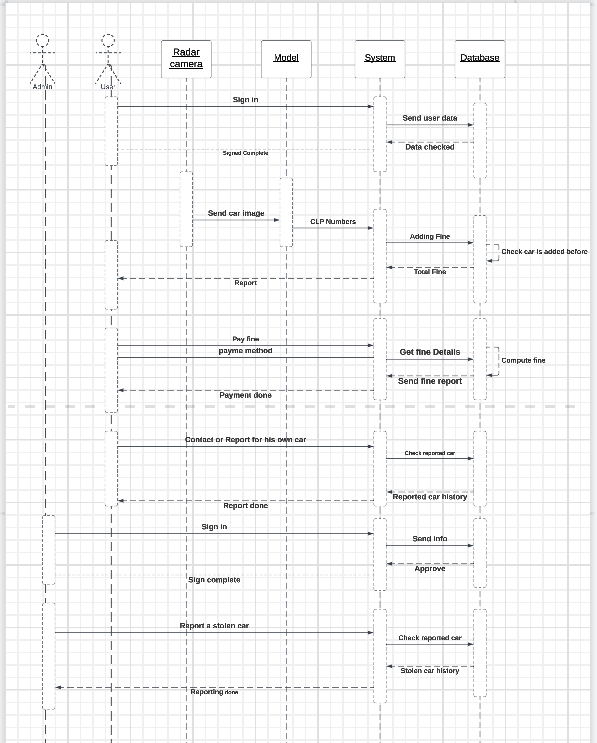
A sequence diagram is a type of interaction diagram because it describes how -and

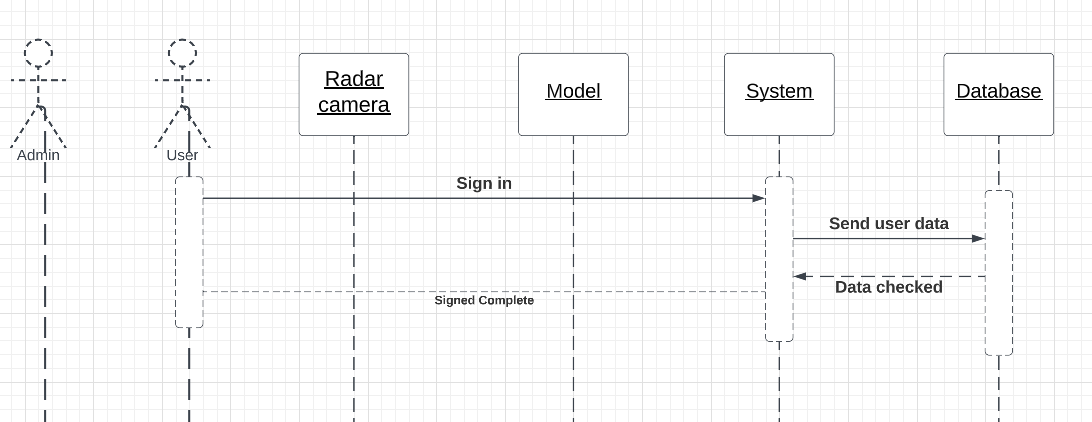
in what order- a group of objects works together. These diagrams are used by

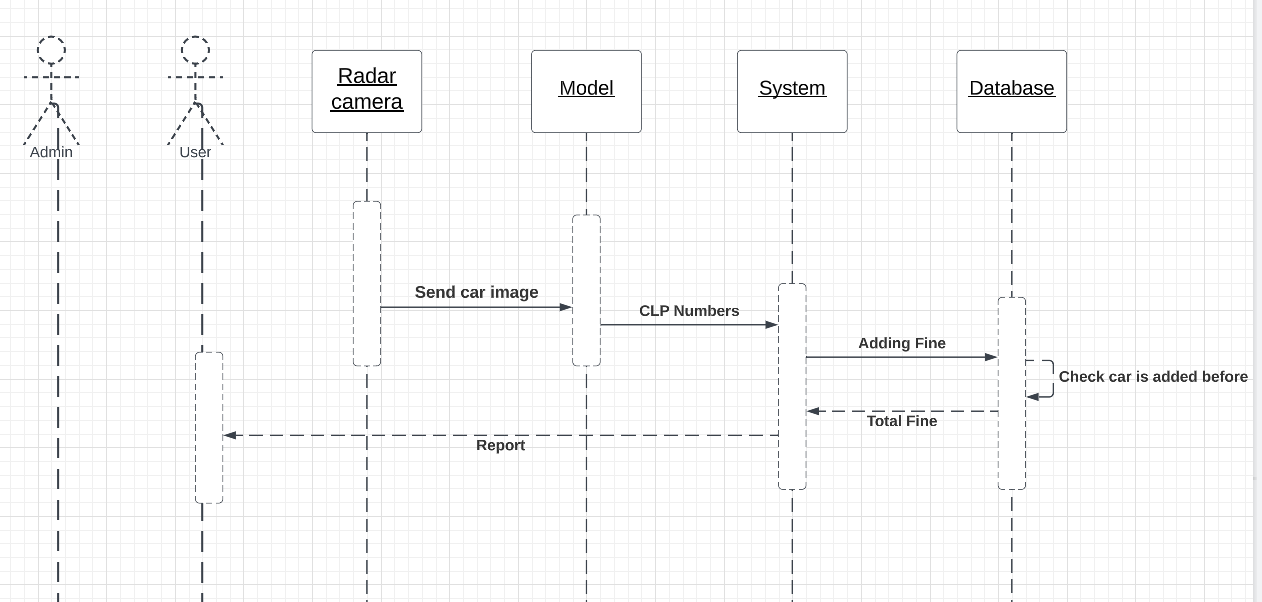
software developers and business professionals to understand requirements for a

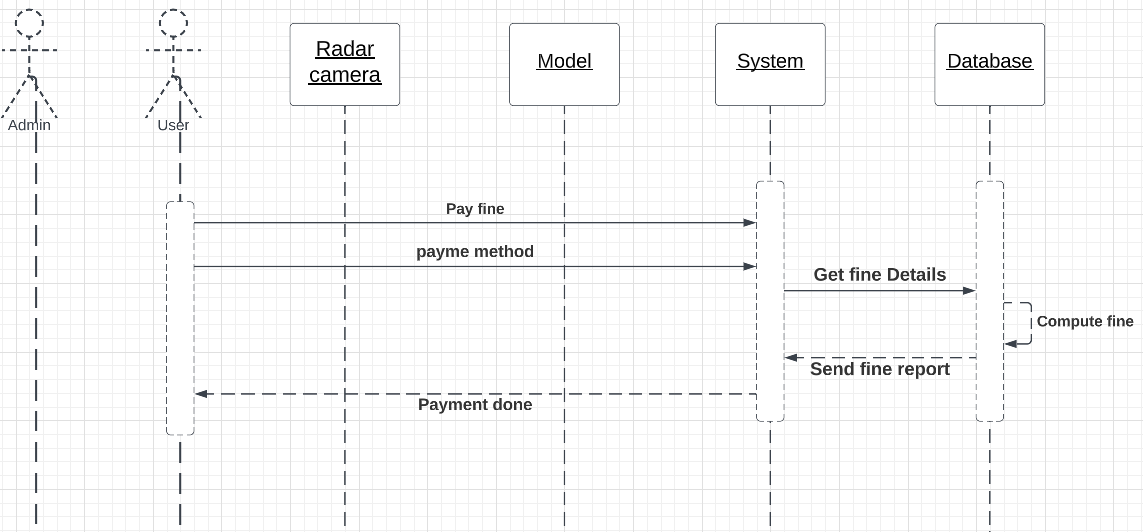
new system or to document an existing process

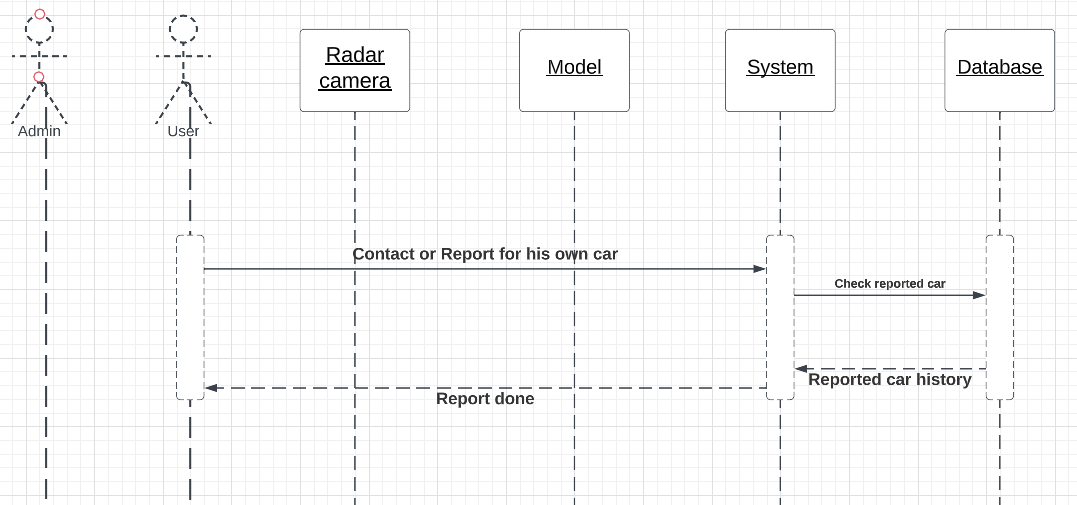
In the following section we display Sequence Diagram of system

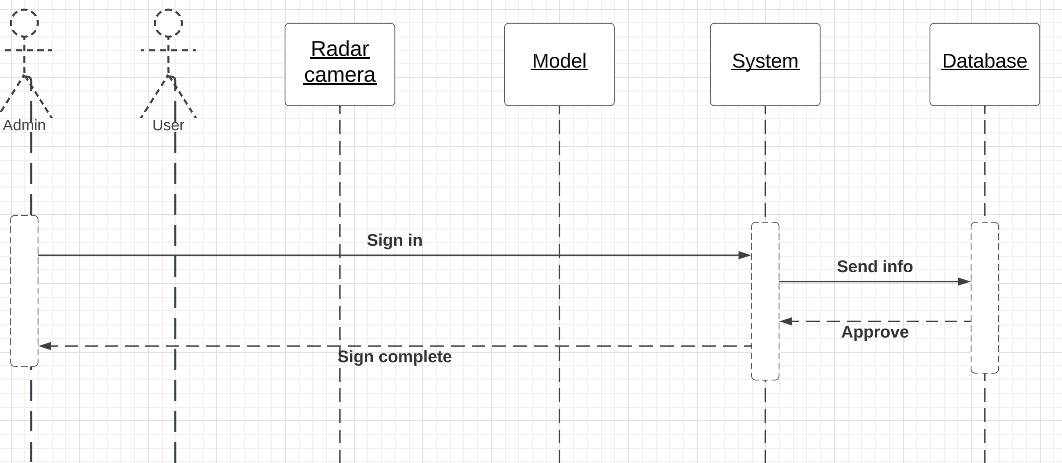


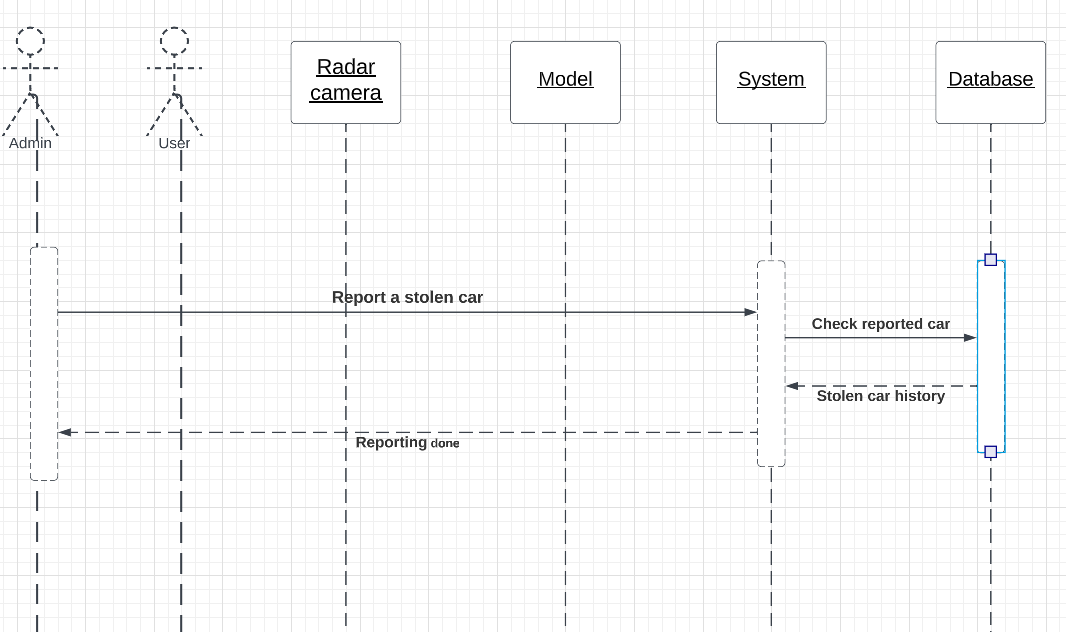












Work Flow Diagram

