MAKERERE UNIVERSITY

COLLEGE OF COMPUTING AND INFORMATION SCIENCES

SCHOOL OF COMPUTING AND INFORMATICS TECHNOLOGY

BSE 23 -14

TRAFFIC LIGHTS COMPLIANCE AND PENALTY EMBEDDED SYSTEM WITH WEB INTERFACE

DEPARTMENT OF NETWORKS

A Project Report Submitted to the School of Computing and Informatics Technology for the Study Leading to a Project in Partial Fulfillment of the Requirements for the Award of the Degree of Bachelor of Science in Software Engineering of Makerere University.

NAME	REGISTRATION NUMBER	STUDENT NUMBER
NAKIBINGE BONNY	**19/U/12552/PS**	**1900712552**
KATEMBEKO CHRISTOPHER	**19/U/24692/PS**	**1900724692**
OTIM DERRICK MURUNGI	**19/U/12613/PS**	**1900712613**
NSUBUGA ALPHA FRANCIS	**19/U/8652/EVE**	**1900708652**

Table of Contents

List of figures

List of tables

1. INTRODUCTION

1.1 Purpose

This software design document describes the architecture and system design of the Traffic Lights Compliance and Penalty System (TLCPS). The purpose of this document is to provide a comprehensive overview and understanding of the system's functional and non-functional requirements, as well as its high-level architecture, component design, and implementation plan.

This SDD is intended for the software development team responsible for building and implementing the system, as well as other stakeholders such as project managers, quality

assurance testers, and system administrators. This document serves as a reference point for the software development team and provides a framework for ensuring that the software is designed to meet the requirements outlined in the software requirements specification (SRS) document

1.2 Scope Of The System

The traffic lights system software is designed to track drivers and riders who violate the traffic lights red signal. The system is composed of hardware components such as cameras and sensors that detect violations and software components that process the data in order to extract the license plates and identify the violating driver. The software will be capable of automatically issuing fines to violating drivers after capturing and analyzing the number plates and notifying the violating drivers though SMS notification.

1.3 Objective (s)

The primary goal of the software is to increase safety on the roads by reducing the number of traffic violations which can lead to accidents and injuries along the roads. The system will also help to enforce traffic laws by the law enforcement agencies and improve traffic flow in the junctions.

1.3.1 Specific Objectives

1. To identify and capture the license plate images of the vehicles that violate the traffic red light signal.
2. To extract the license plate number from the images captured using an OCR trained model.
3. To analyze the number in order to identify the violating driver and issue a penalty automatically to through a text.
4. To create evidence of the traffic light violators to ensure law enforcement with evidence.

1.4 The Benefits Of The System

1. Increased safety on the roads by reducing the number of traffic violations and accidents.

2. Increased traffic flow especially at the junctions by better enforcing the traffic red lights laws.
3. Reduce the work load of the traffic police officers who will no longer need to physically monitor the traffic flow on roads.
4. To a larger extent, revenue will be generated to the government through the fines issued to the traffic violators.

1.5 Overview Of The Document.

The document is comprised of eight sections. The section is an introduction that highlights the purpose of the document and its intended audience. The second section the system overview that provides a general description of the functionality, context and design of the project. The third section system architecture which comprises of the architectural design and the decomposition of the system. The fourth section data design which is composed of the data dictionary and the data description and dictionary of the system. The fifth section is the component design that looks at all the components proposed in the system. The sixth section is the human interface design which contains the overview of the human interfaces, screen images and actions of the system intended to be developed. The requirements matrix that traces components and data structures to the requirements in SRS document. The last section is the appendices which is optional but includes and other kind of information that may be useful to the system and can provide a better understanding to the system.

1.6 Definitions and Acronyms

Table 1.1: table showing Acronyms and definitions.

Acronyms	**Definitions**
API	Application Programming Interface
OCR	Optical Character Recognition.
RDBMS	Rational Database Management System
SDD	Software Design Document.
SRS	Software Requirement Specification.

TLCPS	Traffic lights compliance and penalty system.

2. SYSTEM OVERVIEW

The traffic lights compliance and penalty system is a software application that is designed monitor and manage traffic lights enforcement rules along roads. The system collects real time data from the sensors and the cameras and analyzes the image through optical character recognition (OCR) well trained model. The identified license plate is then stored and analyzed to obtain the driver details and finally, issues fines through notifying the traffic lights violators to control the violation of traffic lights.

The system is built using event-based architecture, where system modules intercommunicate with each other by sharing events. Events are triggered by motion sensed by the sensor and passed on to other components. Pipe and filter model architecture is also employed in development of the TLCPS in that the output from one module is used as the input in the other module.

Client-Server architecture is used only when the traffic officer interacts with the system interface to obtain reports and maintain the system functionality. In order to ensure system reliability and availability, the system is designed with redundant components and also built with robust security measures such as login details to protect against unauthorized access to unauthorized users. The project functionality is broken down into different steps that all together make a joint system design and they are as follows:

1. The system uses a machine learning model algorithm to analyze license plate and make intelligent decisions in identifying the number plates and OCR to analyze the license plate in order to obtain the license plate number.
2. The system also includes a violation detection module comprised of sensors to detect traffic violations such as running the red light.
3. The system is to ensure real-time automatic issuing of penalties through generating notifications to the traffic lights violating drivers through a system's notification gateway.

4. The system design includes a dashboard that allows authorized traffic officers to monitor and manage how the system performs in case the license plate number is not found in the database.

The project designs.

In order to ensure complete functionality and better architecture, the system is built in different levels or layers: license plate capture, number recognition and database identification search of the number, penalty implementation.

1. The license plate capture is the first layer that will be responsible for identifying motion of vehicles at red light and capturing the license plates of the violating drivers. The design at this level is composed of hardware components such as the PIR sensor, Raspberry camera connected to the raspberry pi board that manages the functionality of the components.
2. Number recognition is the second layer of the system that provide the general purpose as to why the project is necessary. Number recognition is where images of the system are analyzed to obtain the number plate text. To achieve the recognition, machine learning such as the use of the tensor flow is used to ensure license plate automation.

   In this level, license automation, this OCR is used to convert the image into text. The license plate image is converted into a text that can be stored in the data base and analyzed.
3. Database Identification search is the next deployment of the project. The database is involved to ensure store and identification of the driver details for the identified license plate. We assume that the database contains details of the drivers and if the identified license plate is check in the database, it is either found or not and if found, the corresponding details of the driver are obtained if not found that the traffic officers have to look for the vehicle in order to obtain the driver details such that the driver is included in the database. SQL statements are used to ensure full functionality in the database.
4. Penalty implementation the last step of the system design and help to fully implement the major goals and objectives of the system. Issuing penalties is the main reason as

to why the project is considered to be relevant. The design of this component is to use the SMS API to send a SMS message to the violating drivers whose number plates have been identified in the database.

3. SYSTEM ARCHITECTURE

3.1 Architectural Design

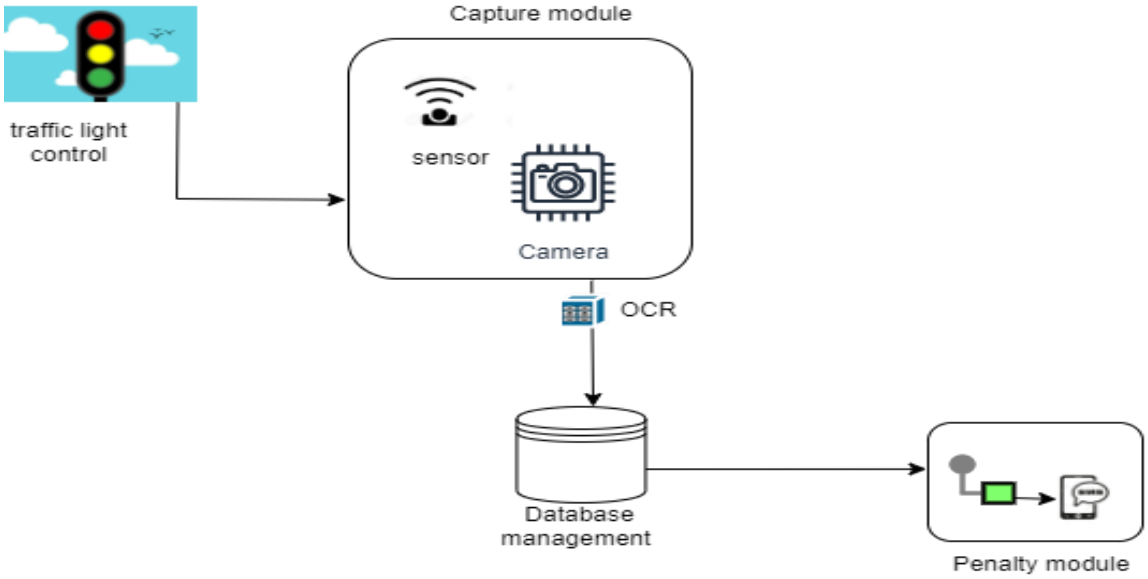


Figure 3.1:module diagram showing the system components.

The Traffic Lights Compliance and Penalty System is designed using a modular architecture to ensure flexibility, scalability and maintainability. These modules communicate with eachother through share events. The events are triggered by some action and passed to the different modules of the system. The modules of the system as shown in the diagram above as all together make up a fully functional system.

1. Motion Detection and Capture module: the capture module is triggered by an event from the traffic light when the traffic light turns red. This module belongs to eventbased architecture and will help in capturing of license plate of vehicles which is triggered by an event from the sensor after detecting motion of the vehicle at red light.
2. OCR: is a submodule of the system that will help to convert images the captured images into text. the module takes in the output of the capture module. Therefore, if

no images are captured then the sub – module is inactive. This submodule belongs to pipe and filter architecture where by it takes in the output from the capture module.

3. Database management: is the module of the system used to store, retrieve and run queries on data. The database of the system shall store driver information such as name, date of birth, license plate, permit number etc. and run queries to the license plate number that will be passed to it to identify the details of the violating drivers. the information shall be retrieved and can be viewed by the traffic officer to ensure full system functionality.
4. Penalty module is a system module that will issue penalties to the violating drivers therefore highlighting major functionality of the system build. The module will use the SMS API to issue SMS texts to the driver phones. This module demonstrates the last architecture which is the client-server architecture.

3.2 Decomposition Description

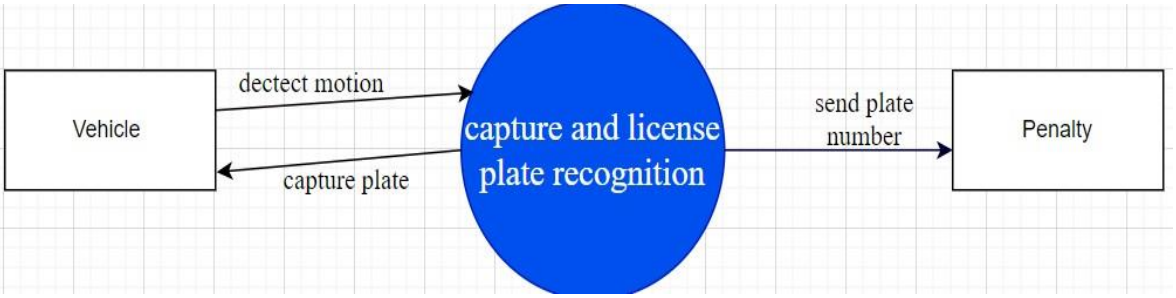


Figure 3.2: level 0 diagram showing major system components.

The diagram explains the level 0 diagram of the TLCPS. It shows the overall system components at level 0 meaning show how the system will operate at a basic level of understanding of the stakeholders.

The vehicle represents the environment that the system will operate In or collect information from. The basic understanding of the level 0 explains that the TLCPS will detect motion of the vehicle, capture the license plate of the vehicle and finally issue the penalty. Level 0 diagram doesn't look at the constraints of the system component such as the red light has to be on in order to detect motion.

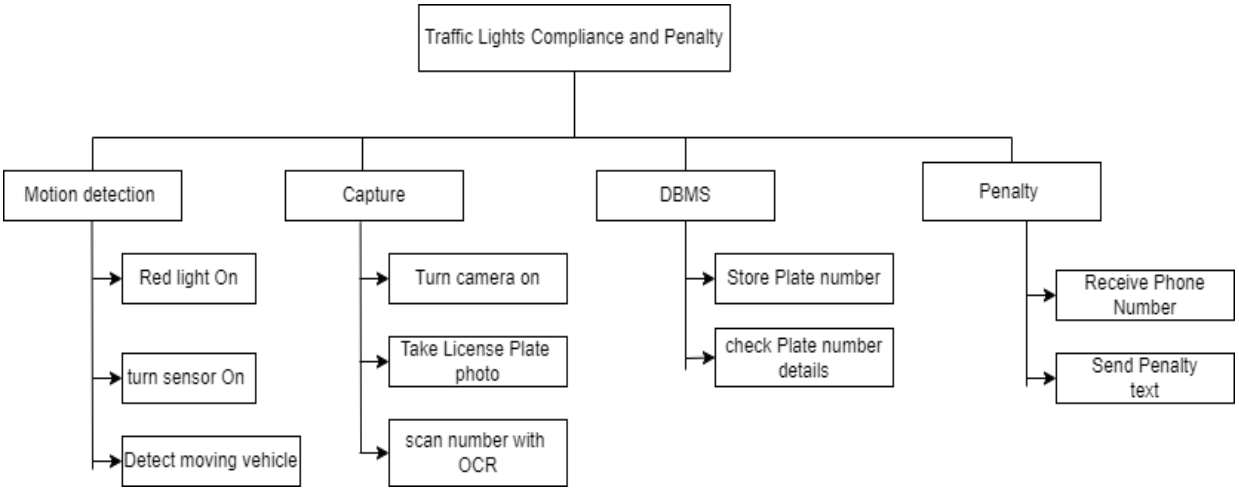


Figure 3.3: Structural decomposition diagram showing the decomposition of system components

The Structural decomposition diagram above represents the hierarchical relationships between the system components. The TLCPS is represented by one box at level 1 and broken down into subsystem at the next level. The subsystems (motion detection, capture, DBMS, and Penalty) are then each broken into subsystems on the next level as function that each component performs. This will allow designers know how the system components perform their tasks and know how the interact with each other. The structural decomposition diagrams above clearly identify what each component of the TLCP system is supposed to perform.

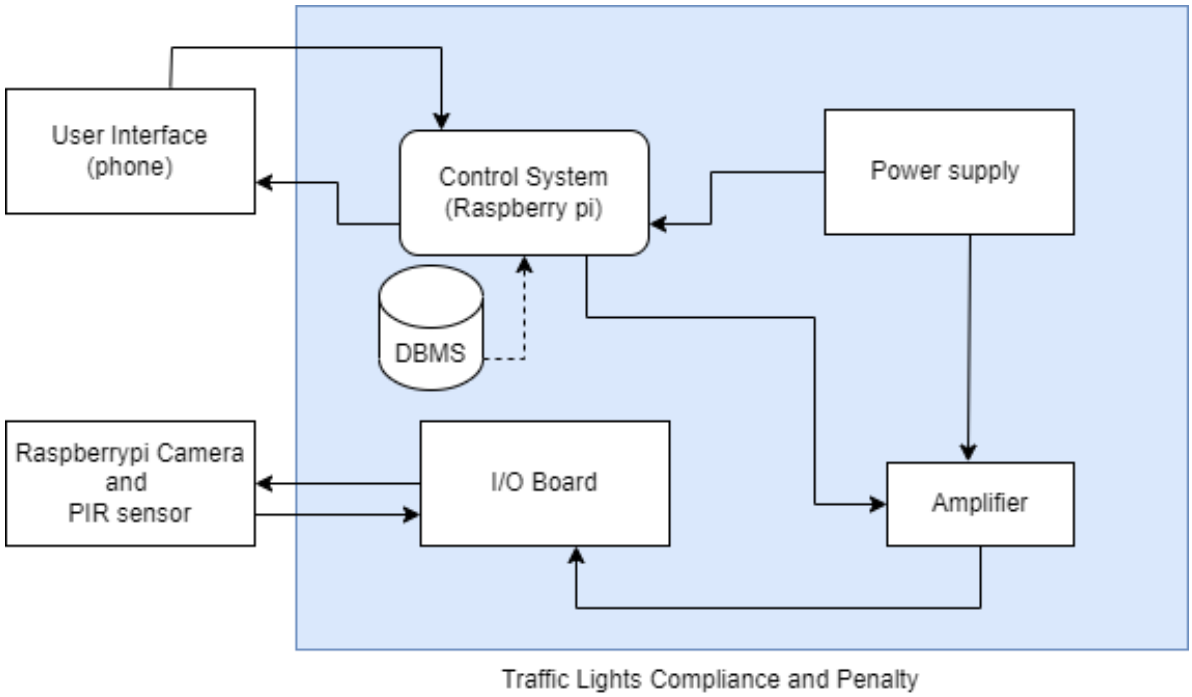


Figure 3.4: sub-system model diagram showing the other sub-system involved with the system.

The sub system module architectural diagram above identifies the other subsystem modules that the TLCP system components interact with and show the relationships between the subsystem and the modules of the system.

Some of the other subsystems associated with the TLCP system shall be:

1. Power Supply sub system: this Subsystem is responsible for providing power supply to the system components to ensure that they start up and perform their respective functions.
2. Amplifier: this is a subsystem that allow even distribution of the power across the other components of the system. Such as amplifies power to the control system (Raspberry Pi) and the I/0 Board connected to the sensors.

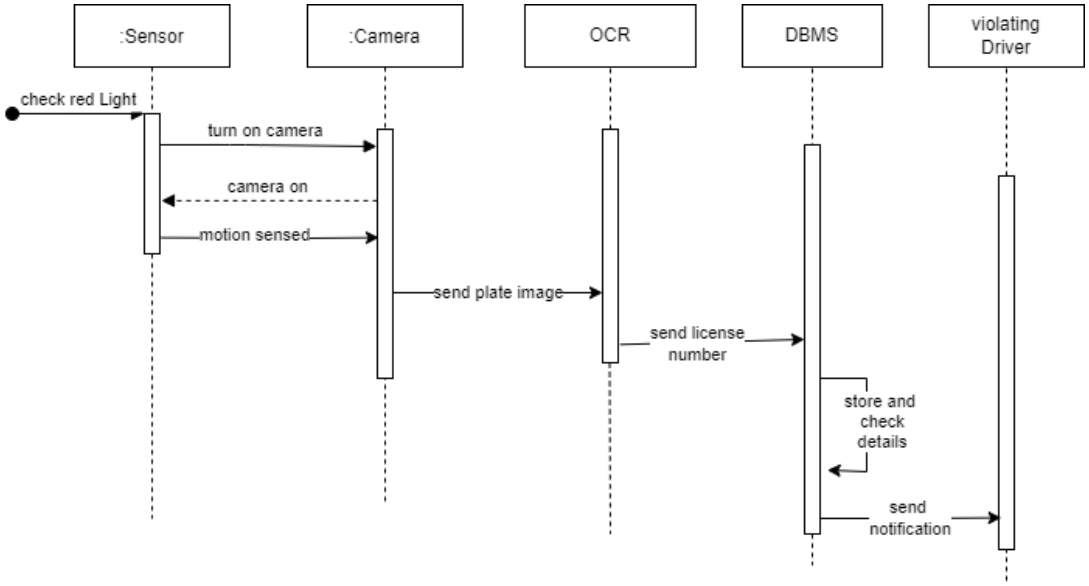


Figure 3.5: sequence diagram showing the follow of activities in the system

The TLCP system sequence architectural diagram represents the flow of activities within the system. It defines the individual components of the system and how events are activated and handled in the system through a sequential pattern. Actions are performed form the first to the last without going back and shows which component receives what information.

The diagram helps designers to gain a better understanding of how the system performs action to avoid loops and component interchange.

3.3 Design Rationale

An event-based architecture is selected because it best aligns with the system's requirements and needs for real-time processing and integration with external systems. Events are the basis for communication among the system components and the functionality is triggered by the events.

There are many reasons as to why the selected architecture best suits the design of the system but the major reason are as follows:

1. The system involves multiple modules that need to communicate with each other in an asynchronous and loosely coupled manner. The architecture allows each module to operate independently and respond to events as they occur, without needing to wait

for other modules to complete their tasks. For example, the capture module won't necessarily need

to wait for the penalty module to issue the penalty to the 1st violating driver in order for it to capture another image of a violating driver. This can make the system more scalable and easier to maintain over time.

2. The system involves real-time processing of data, such as capturing license plate images from violating drivers at red light and extracting license plate numbers. An event-based architecture allows the system to respond quickly to new data as it becomes available, without the need for manual intervention or scheduling.
3. The system shall be involved in integration with external systems, such as a database of registered vehicles and vehicle owners. An event-based architecture can facilitate this integration by allowing the system to send and receive events in a standardized format that can be easily consumed by external systems.

In terms of critical issues and trade-offs, there are a few considerations to keep in mind when using an event-based architecture. One of the potential issues we identified is the potential for event overload. This occurs the system receives too many events at once and becomes overwhelmed. This can be mitigated by implementing event throttling or other strategies to limit the number of events that the system can process at any given time.

Another trade-off considered is the complexity of designing and implementing an eventbased architecture. This architecture requires careful attention to event formats, event handling, and event-driven programming techniques. It also requires a clear understanding of the system's requirements and use cases to ensure that the events are designed appropriately.

In the development, Client - Server architecture will also be used in the development of the system to enable the traffic officers interact with the system. They act as clients who request requests from the system, in response the corresponding response is received. There could be many reasons as to why to a certain extent this architecture was used and they are not highlighted in this document but there is important reason that were identified in the deployment of the system and are as follows:

1. Client server architecture provides better security than other architectures. In the architecture the server is responsible for managing access to resources, which can be used to protect sensitive data from unauthorized access. Therefore, the architecture will provide a platform for only the authorized Traffic officer in the server to access the database to retrieve the information.
2. The client-server architecture is capable of managing the data and application, which mean that any updates or maintenance can be performed centrally on the server updates are accessed anywhere. Therefore, the architecture will ensure that all authorized traffic officers will be able to access the same updates and will be able to work on the same data and application.

However much the architecture was used and beneficial in the development, it is not efficient enough to communicate all the system component and is can't not handle multiple request or overload at the same time since the server is likely to crush due to over load. The TLCP system components do not involve request however they share information among them and to enable efficiency, each component works independently to the real time data which the client server architecture can not handle because all responses come from the server and therefore some components will have to wait for response.

Also due to the single point of operation, the client-server architecture was not chosen because if the server fails, due to multiple requests that come in every time, then the system will have to fail.

4. DATA DESIGN

4.1 Data Description

The system will use a database to store and retrieve information. The major data and system entities that will be stored, processed and organized are:

User profile: the user profile will contain information about each user of the system including the names, title, identification number, password and some other relevant

information such as location of work and residence. In the case of our traffic light compliance and penalty system, the user are the traffic officers IT personnels.

Driver profile: the driver profile will define the information about the drivers with there information including names, phone contact, age, sex, date of birth, permit number, vehicle license plate number, residence, place of work.

All our data will be stored in a data base using a relational database management system (RDBMS). The RDBMS will create tables to store the data from the user profile and Driver profile in tables and create the relationship in the data. With the help of MySQL database, we shall be able to manipulate the data. The data items involved in the system shall include: i. OCR license plate text ii. Penalty text data iii. Motion detection data from the sensor iv. License plate images from the camera.

4.2 Data Dictionary

Table 4.1: table showing the driver profile data entities, their relationships and description

Entity Name	Data Type	Field Length	Description
Age	Int	2	Age of all registered drivers
Contact	Int	10	Phone number for the driver on which they can be accessed.
Date Of Penalty	Date	9	Date when penalty is issued for violation
DOB	Date	9	Date of birth of the driver
id	Int	2	Unique identifier for a driver

license Plate	String	12	Number plate uniquely identifying each car.
Name	String	30	Name of the driver.
Permit No	String	15	Identifier that the driver is permitted to drive a vehicle
Place of work	String	10	Company or organization where the driver works from.
Residence	String	10	Place where the driver lives.
sex	String	1	Gender of the driver.

Table 4.2: table showing user profile and their entities and data structure

Entity Name	Data Type	Field Length	Description
Contact	Int	10	Phone number of the traffic officer
Id	Int	2	Unique identifier of the office
Name	String	24	Name of the traffic officer
Password	String	8	Secret character for logging onto the system
Place Of Work	String	10	Area or work station

Residence	String	10	Place where office lives
Sex	String	1	Gender of the traffic officer.

5. COMPONENT DESIGN

5.1 Motion detection module

The motion detection module is the module the triggers the events and is responsible for identifying violations of traffic lights compliance.

Pseudocode:

While traffic light is **red**.

Sensor is switched on and camera is turned on.

If vehicles move past the sensor,

Motion Is detected.

5.2 License plate capture module

The license plate capture is the module the triggers the event of capturing and takes on the event that was trigger by the sensor to detect motion to determine that the vehicle should be captured therefore, the license plate is captured.

This module also takes on the responsibility of capturing the license plate number as a text using the OCR sub system model. The OCR is a model trained to analyze the license plate in order to capture the text on the plate

Pseudocode:

While camera is on.

Capture license plate of moving vehicle.

If image captured.

Convert to text.
Clear the text to obtain the actual plate.

Confirm the degree of accuracy.

5.3 Data processing module

Data processing module is the module responsible for storing, analyzing and verifying the existence of the license plate in the database. The database uses SQL statements to analyze the license plate number inform of a text. The module is triggered by the actions from the output from the License plate capture module.

Pseudocode:

While text accuracy is greater than 80%.

Store text in the database.

Match/compare the text to check its existence.

If Text exits.

Using SQL statement obtain the contact number and permit details.

5.4 Penalty module

The penalty module is triggered after obtaining the analysis of the data processing module. It takes into account of the contact and permit number related to the license plate number that was captured.

A penalty text is forwarded to the contact obtained for the violating driver.

In addition, the penalty module will be responsible for managing reports. It will be able to pull the records of the license plate that has been captured and issued a violation penalty. It will follow the penalty by checking the status of the penalty.

The traffic officers will be involved in the module for analyzing and check the status of the penalties and verifying the degree of accuracy of the license plate number.

6. HUMAN INTERFACE DESIGN

6.1 Overview of User Interface

The user interface of system will aim at providing an intuitive and user-friendly experience for users. The system will mostly be an embedded system that takes into account of the sensor interface, camera interface to input the data into the system.

The PIR sensor interface will sense the motion of the vehicles the pass infront of it when the red light is on implying that the there is a violation of the traffic lights. And on the other hand, the camera interface is responsible for capturing the images of the vehicles and keeping track of the traffic light violations.

The system outputs the results of the analysis through sending messages to the violating drivers' mobile phones as texts and also provide reports to the user through a web interface that will provide the status of the penalty and feedback of invalid license plate numbers.

6.2 Screen Images

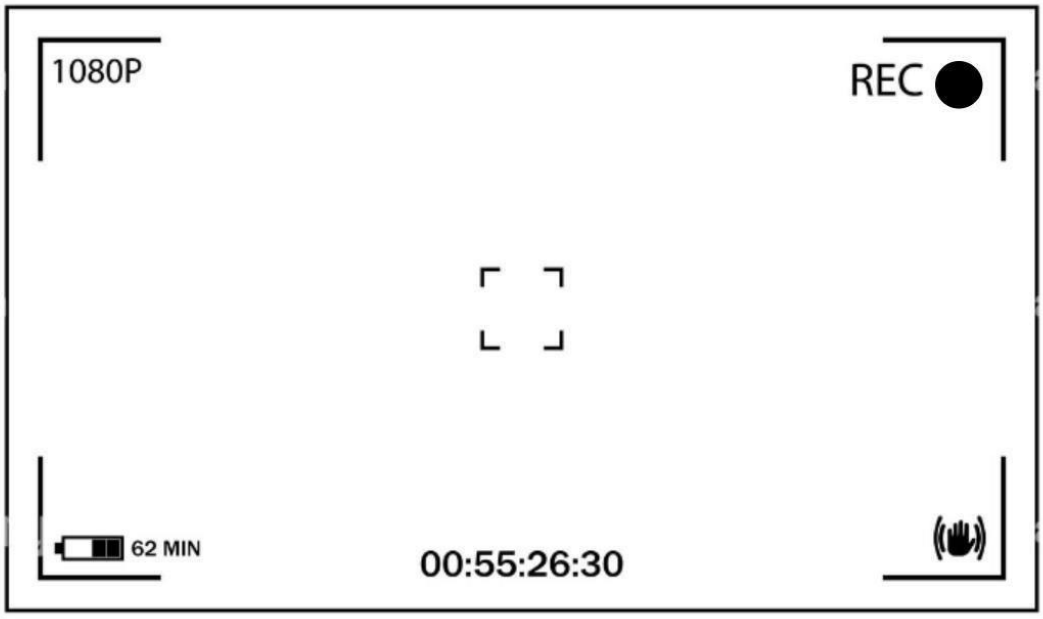


Figure 6.1: showing the camera interface.

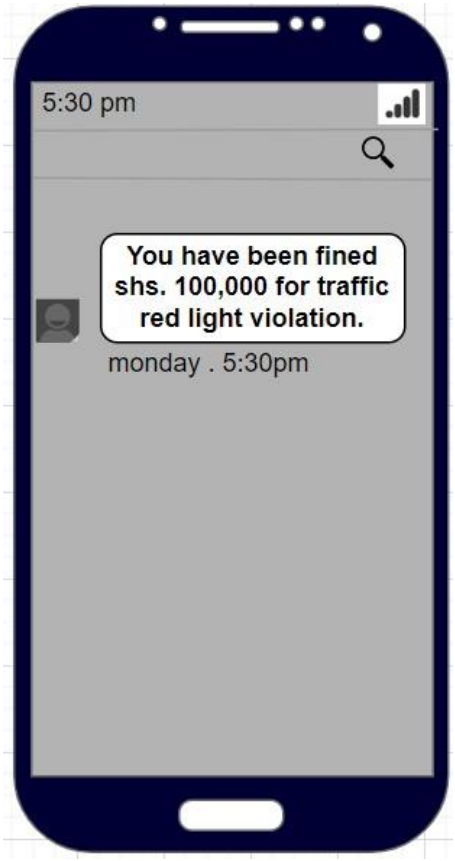


Figure 6.2: showing the driver interface

Figure 6.3: showing the user interface login form.

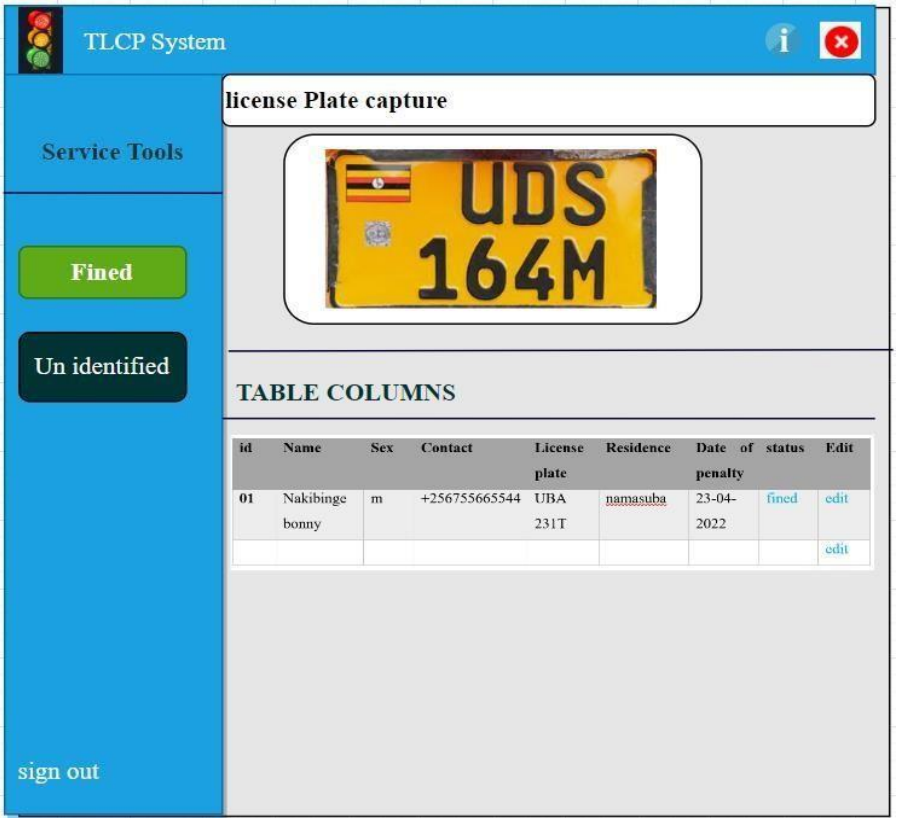
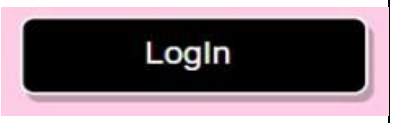
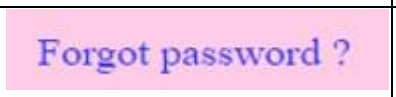


Figure 6.4: showing the user interface for the data.

6.3 Screen Objects and Actions

Table 6.1: Table showing screen objects and their actions

Screen Object	Screen object image	Action
Officer Id input		Allow Traffic Officer identification
Password input		Allow Traffic Officer authentication
Login button		Enter system display on click.
Forgot Password?		Go to forgot password menu for the traffic officer to reset the password.
Fined button		Open table filtered with fined traffic violators.
unidentified button		Filters license plates that are not identified.
Visualization Screen		Show the officer that system is operating.
Edit		Allows the traffic officer assign penalty manually.

Close		Close the system interface
About button		Provide information about the traffic lights compliance and penalty system
Sign out		Logs out the traffic officer personnel.

7. REQUIREMENTS MATRIX

Table 7.1: Table tracing the requirements in the SRS to the system components.

System Component	Functional Requirement	Priority	Description
Motion detection module	NPCS 3	High	Detect motion of the vehicle when it is red light.
License plate capture module	NPCS 1, NPCS 2	High	1. Capture license plate images 2. Scan the plate using OCR to obtain the license plate text.

Data processing module	DLM 1	High	1. Store the license plate text. 2. Match the plate to obtain Attached information eg name, contact, permit number of the driver
Penalty module	DF 1, DF 2	high	Issues penalty through the SMS notification.

8. APPENDICES

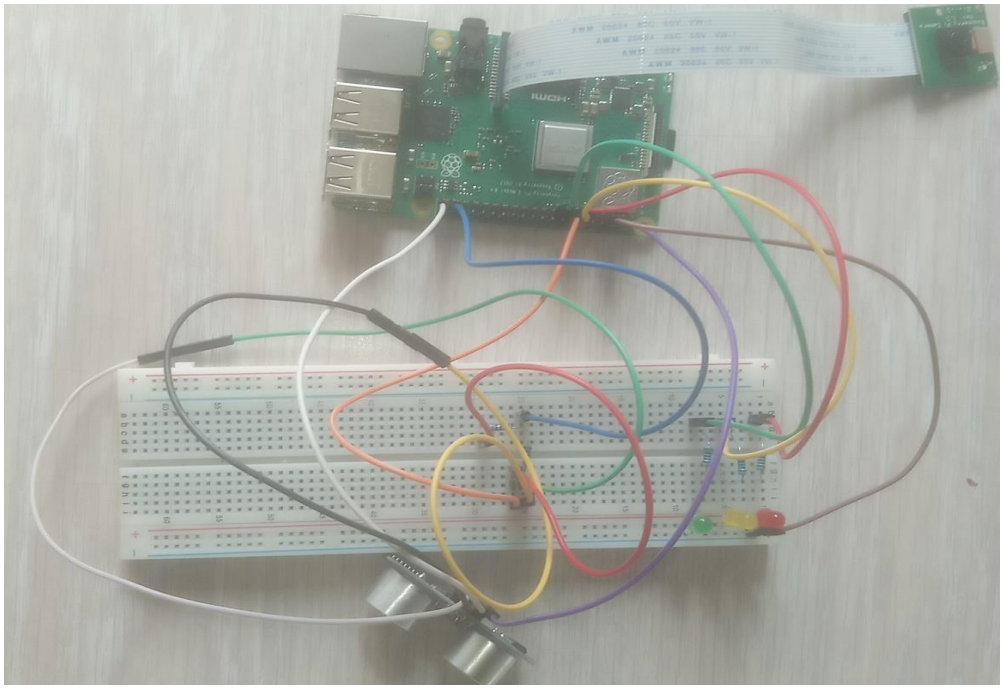


Figure 8.1: image of the PIR sensor

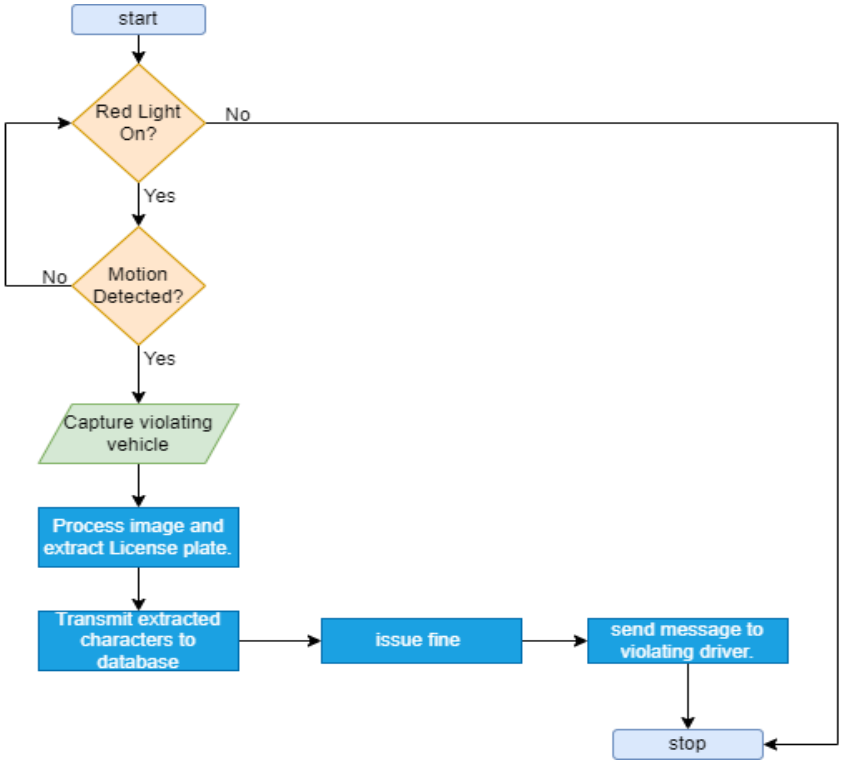


Figure 8.2: Data flow diagram showing how the data elements flow in the system.