# 

**Dr. D.Y. Patil School of MCA**

***Charoli (BK), PUNE-******412105***

**SAVITRIBAI PHULE PUNE UNIVERSITY**

**MASTER OF COMPUTER APPLICATION**

Project Report

on

VEHICLE MONITORING

SYSTEM

Under The Guidance Of

“ Prof. Asmita Hendre ”

**BY**

**“Sanskar Wagavkar”**

**Class : MCA-I( Sem-II)**

**Year : 2022-2023**

Date: -

**CERTIFICATE**

This is to certify that Mr. / Ms. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_, has successfully / partially

Completed his/her project work entitled “VEHICLE MONITORING SYSTEM” in

fulfillment of MCA - I SEM – II Mini Project for the year 2022-2023. He / She have

worked under us guidance and direction.

**Place:**

**Date:**

**Prof. Asmita Hendre Prof. Ashok Deokar Dr. E. B. Khedkar**

**(Project Guide) HOD DYPSOMCA (Director,DYPSOMCA)**

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **Sr. No.** | **CHAPTER DETAILS** | **Page Number** |
| **1** | **INTRODUCTION** | **5** |
| 1.1 | INTRODUCTION | 5 |
| 1.2 | EXISTING SYSTEM AND NEED FOR A SYSTEM | 6 |
| 1.3 | LIMITATION OF EXISTING SYSTEM | 7 |
| **2** | **PROPOSED SYSTEM** | **8** |
| 2.1 | PROBLEM STATEMENT | 8 |
| 2.2 | PRODUCT OVERVIEW | 9 |
| 2.3 | OBJECTIVES OF PROPOSED SYSTEM | 10 |
| 2.4 | FUNCTIONAL REQUIREMENTS | 11 |
| 2.5 | NON-FUNCTIONAL REQUIREMENTS | 12 |
| 2.6 | MODULE SPECIFICATION | 13 – 18 |
| 2.7 | OPERATING ENVIRONMENT | 19 |
| **3** | **REQUIREMENT DETERMINATION AND ANALYSIS** | **20** |
| 3.1 | FACT FINDING METHODS | 20 |
| 3.2 | FEASIBILITY STUDY | 21 |
| **4** | **SYSTEM ANALYSIS AND DESIGN** | **22** |
| 4.1 | USE CASE DIAGRAM | 22 - 27 |
| 4.2 | ACTIVITY DIAGRAM | 28 – 33 |
| 4.3 | SEQUENCE DIAGRAM | 34 – 39 |
| 4.4 | CLASS DIAGRAM | 40 |
| 4.5 | OBJECT DIAGRAM | 41 |
| 4.6 | ENTITY RELATIONSHIP DIAGRAM | 42 |
| 4.7 | USER INTERFACE DESIGN SCREEN | 43 – 45 |
| **5** | **CONCLUSTION** | **46** |
| **6** | **BIBILOGRAPHY** | **47** |

**ABSTRACT**

The purpose of this project is to develop a vehicle detection system using Python and OpenCV to monitor the vehicles entering and exiting a college campus. The system also has a front-end interface developed using HTML, CSS, and Flask framework, and a back-end database using MySQL to store the relevant information about the vehicles. The vehicle detection system works by processing live video footage of the entrance and exit points of the college campus using OpenCV. The system can detect the presence of vehicles in the video feed and classify them as either cars, trucks, or bikes. The information about the detected vehicles is then stored in the MySQL database along with the vehicle's registration number, date and time of entry, and exit. The front-end interface developed using HTML, CSS, and Flask framework provides an easy-to-use web interface for the college administrators to view the real-time status of the vehicles entering and exiting the campus. The interface also allows the administrators to search for specific vehicles and view their detailed entry and exit logs.

**CHAPTER 1: INTRODUCTION**

* 1. **INTRODUCTION**

Vehicle detection systems have become an essential part of modern-day traffic management, especially in crowded areas such as college campuses.

A vehicle detection system can help monitor the movement of vehicles entering and exiting a campus, which can help improve security, traffic flow, and overall management of the campus. In this project, we have developed a vehicle detection system using Python and OpenCV to monitor the vehicles entering and exiting a college campus. The system is designed to process live video footage of the entrance and exit points of the college campus using OpenCV.

The system can detect the presence of vehicles in the video feed and classify them as cars, trucks, or bikes. The information about the detected vehicles is then stored in a MySQL database, which can be accessed through a web interface developed using Tkinter framework. The primary goal of this project is to provide a user-friendly interface for the college administrators to monitor the real-time status of the vehicles entering and exiting the campus.

The interface also allows administrators to search for specific vehicles and view their detailed entry and exit logs. The back-end MySQL database provides a scalable and efficient way to store and manage the large amounts of data generated by the vehicle detection system.

**1.2 EXISTING SYSTEM & NEED FOR THE SYSTEM**

* Currently, most college campuses rely on manual methods such as security personnel to monitor the entry and exit of vehicles. These methods are often ineffective, time-consuming, and prone to errors.
* Manual monitoring can also lead to traffic congestion and security breaches, especially during peak hours. Some campuses may have installed automatic gates or barriers to control the entry and exit of vehicles.
* However, these systems are expensive to install and maintain, and they may not be able to provide real-time monitoring and analysis of the vehicles entering and exiting the campus.

**Need for the System:**

* The increasing number of vehicles and the growing concern for campus security have highlighted the need for an efficient and automated vehicle monitoring system in college campuses. The manual monitoring methods are often ineffective and time-consuming, which can lead to traffic congestion and security breaches.
* Therefore, there is a need for a cost-effective and reliable system that can monitor the entry and exit of vehicles in real-time, store the relevant information about the vehicles, and provide a user-friendly interface for campus administrators to access the information.
* The vehicle detection system developed in this project addresses these needs by providing a cost-effective and reliable solution for monitoring the vehicles entering and exiting a college campus.
* The system can help improve traffic flow, security, and overall management of the campus, while also providing valuable insights for future planning and decision-making.

**1.3 SCOPE OF SYSTEM**

* The scope of the vehicle detection system developed in this project is to provide an efficient and automated solution for monitoring the vehicles entering and exiting a college campus. The system is designed to process live video footage of the entrance and exit points of the college campus using OpenCV and classify the detected vehicles as cars, trucks, or bikes.
* The information about the detected vehicles is then stored in a MySQL database, which can be accessed through a web interface developed using Tkinter framework. The system's scope also includes providing a user-friendly interface for college administrators to monitor the real-time status of the vehicles entering and exiting the campus.
* The interface also allows administrators to search for specific vehicles and view their detailed entry and exit logs. The back-end MySQL database provides a scalable and efficient way to store and manage the large amounts of data generated by the vehicle detection system. The system's scope also includes generating reports and analytics based on the data stored in the MySQL database, which can provide valuable insights for future planning and decision-making.
* In summary, the scope of the vehicle detection system developed in this project is to provide an efficient, automated, and user-friendly solution for monitoring the vehicles entering and exiting a college campus, which can help improve traffic flow, security, and overall management of the campus.

**1.4 OPERATING ENVIRONMENT**

The vehicle detection system developed in this project has specific hardware and software requirements to operate efficiently.

**Hardware Requirements:**

* + - A computer system with a multi-core processor (Intel Core i3 or higher)
    - Minimum 8 GB RAM
    - A camera with a minimum of 360p resolution
    - Sufficient storage space (minimum 10GB) for storing the captured video footage
    - Database files Stable internet connectivity

**Software Requirements:**

* Operating System: Windows 10 or Higher
* Python 3 with required libraries, including OpenCV, Tkinter
* MySQL Connector MySQL Server 8.0 or higher
* Tkinter for developing the interface

**1.5 BRIEF DESCRIPTION OF TECHNOLOGY USED**

**Operating System Used:**

* The operating system used in this project is Windows. The operating system provides a stable and secure platform for running the Python scripts, MySQL server, and web interface.
* The system requirements and dependencies are specific to the operating system, and the project is developed to run on either of the two operating systems.

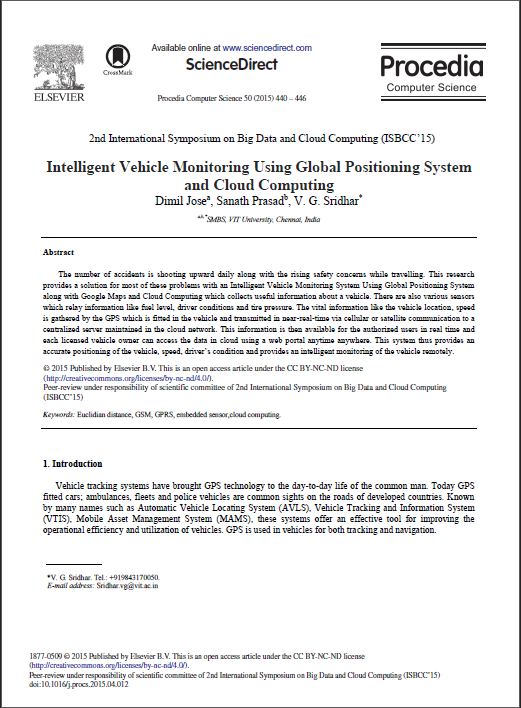
**RDBMS/MySQL:**

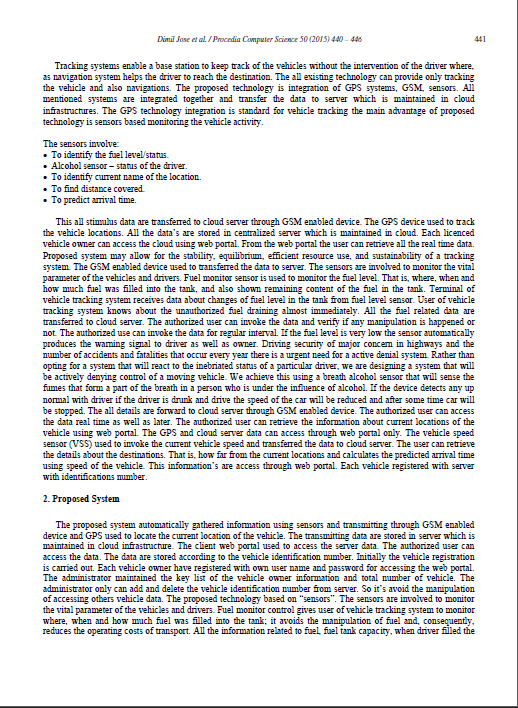
* MySQL is used in this project to build and manage the database. MySQL is an open-source relational database management system (RDBMS) that provides scalability, security, and performance for storing and managing large amounts of data.
* The system uses MySQL to store the data about the vehicles entering and exiting the college campus, including vehicle type, license plate number, entry and exit timestamps, and other relevant information. MySQL provides efficient data storage and retrieval capabilities, which are essential for a real-time vehicle monitoring system like the one developed in this project.
* The system uses the MySQL Connector library in Python to connect to the MySQL server and execute SQL queries for data insertion, retrieval, and modification. Overall, the use of MySQL in this project provides a robust and scalable solution for managing the large amounts of data generated by the vehicle detection system.

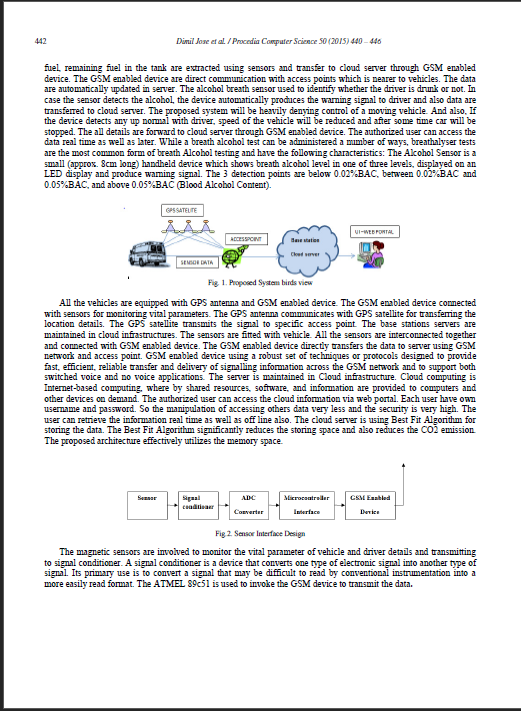
**CHAPTER 2: PROPOSED SYSTEM**

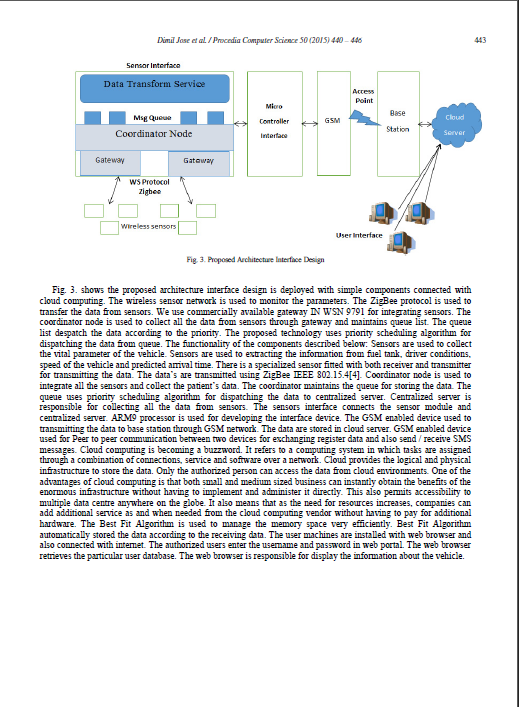
* 1. **STUDY OF SIMILAR SYSTEM**

Research Paper: - "Intelligent Vehicle Monitoring and Tracking System for College Campus Security" by R. Uma Rani and P. K. Srimani, published in the International Journal of Innovative Research in Science, Engineering and Technology in 2015.









* The paper highlights the importance of vehicle monitoring systems in college campuses for preventing unauthorized access, detecting suspicious activities, and improving overall security. The authors propose a system that uses automatic license plate recognition (ALPR) technology and a wireless sensor network (WSN) to track and monitor vehicles entering and exiting the campus.
* The proposed system uses an ALPR camera to capture the license plate number of vehicles, which is then transmitted to a central server using a WSN. The server processes the data and generates alerts if any suspicious activities are detected. The system also includes a web-based user interface that provides real-time monitoring and access to historical data.
* The paper describes the architecture, hardware, and software components of the proposed system and evaluates its performance through simulation experiments. The results show that the system can accurately detect and track vehicles, and generate alerts in real-time.
* Overall, the paper presents a comprehensive and innovative approach to vehicle monitoring and tracking in college campuses. The proposed system has the potential to improve the security of college campuses and provide valuable insights into the traffic flow and patterns of vehicles entering and exiting the campus.

**2.2 FEASIBILITY STUDY**

1. **Technical Feasibility:** 
   * 1. Hardware Requirements: Assess the technical capability of the hardware required to run the app effectively. Consider factors such as processing power, memory, storage, and camera capabilities.
     2. Software Requirements: Evaluate the compatibility of the app with the operating systems and versions targeted for deployment. Ensure that all necessary software dependencies are available and can be integrated seamlessly. Image Processing and Number Plate
     3. Recognition: Investigate the technical feasibility of implementing image processing algorithms and number plate recognition techniques using the chosen framework (Tkinter) and libraries.
     4. Database Integration: Assess the compatibility and feasibility of integrating a MySQL database with the app for storing and retrieving number plate data.
2. **Economic Feasibility:** 
   * 1. Cost Analysis: Evaluate the cost of developing the app, including hardware, software licenses, development resources, and any additional expenses such as server hosting or cloud services.
     2. Return on Investment (ROI): Determine the potential financial benefits and returns that can be derived from the app. Consider factors like revenue generation, cost savings, and potential market demand.

1. **Operational Feasibility:** 
   * 1. User Acceptance: Analyze the target users' needs, preferences, and expectations to ensure the app meets their requirements. Conduct user surveys or focus groups to gather feedback and assess user acceptance.
     2. User Interface (UI) and User Experience (UX): Evaluate the usability and intuitiveness of the app's interface. Consider factors such as ease of navigation, clarity of instructions, and overall user experience.
     3. Scalability and Performance: Determine if the app can handle the expected workload and scale effectively as the number of users and data volume increases. Evaluate the app's performance under various scenarios and identify any potential bottlenecks.

1. **Legal Feasibility:** 
   1. Privacy and Data Protection: Ensure compliance with relevant privacy laws and regulations when collecting, storing, and processing users' personal information. Implement appropriate security measures to protect the data stored in the MySQL database.
   2. Intellectual Property: Assess the legal implications of using image processing algorithms, number plate recognition techniques, or any third-party libraries. Verify that the app's development and deployment do not infringe on any patents or copyrights.

**2.3 OBJECTIVES OF PROPOSED SYSTEM**

* **Improve campus security: -**

The system aims to improve campus security by providing a more efficient and effective way of monitoring the vehicles entering and exiting the campus. The system can help identify unauthorized vehicles and potential security threats, which can help prevent security breaches.

* **Improve traffic flow: -**

The system aims to improve traffic flow by providing real-time information on the number of vehicles entering and exiting the campus. Campus administrators can use this information to optimize traffic flow and reduce congestion.

* **Provide detailed vehicle logs: -**

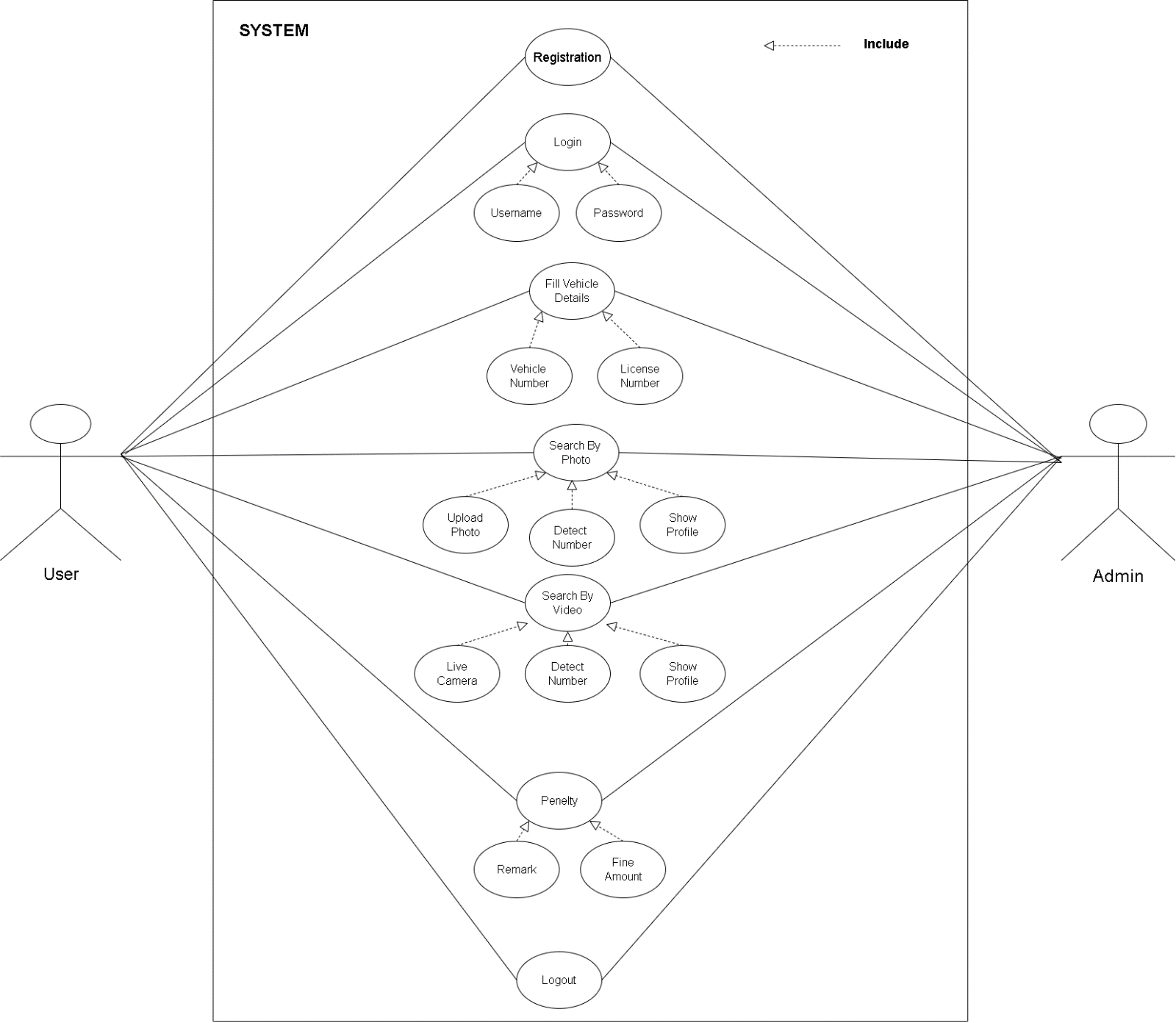
The system aims to provide detailed logs of the vehicles entering and exiting the campus, including the date, time, and vehicle type. This information can help campus administrators track the movement of vehicles on the campus and identify any suspicious activities.

* **Provide a user-friendly interface: -**

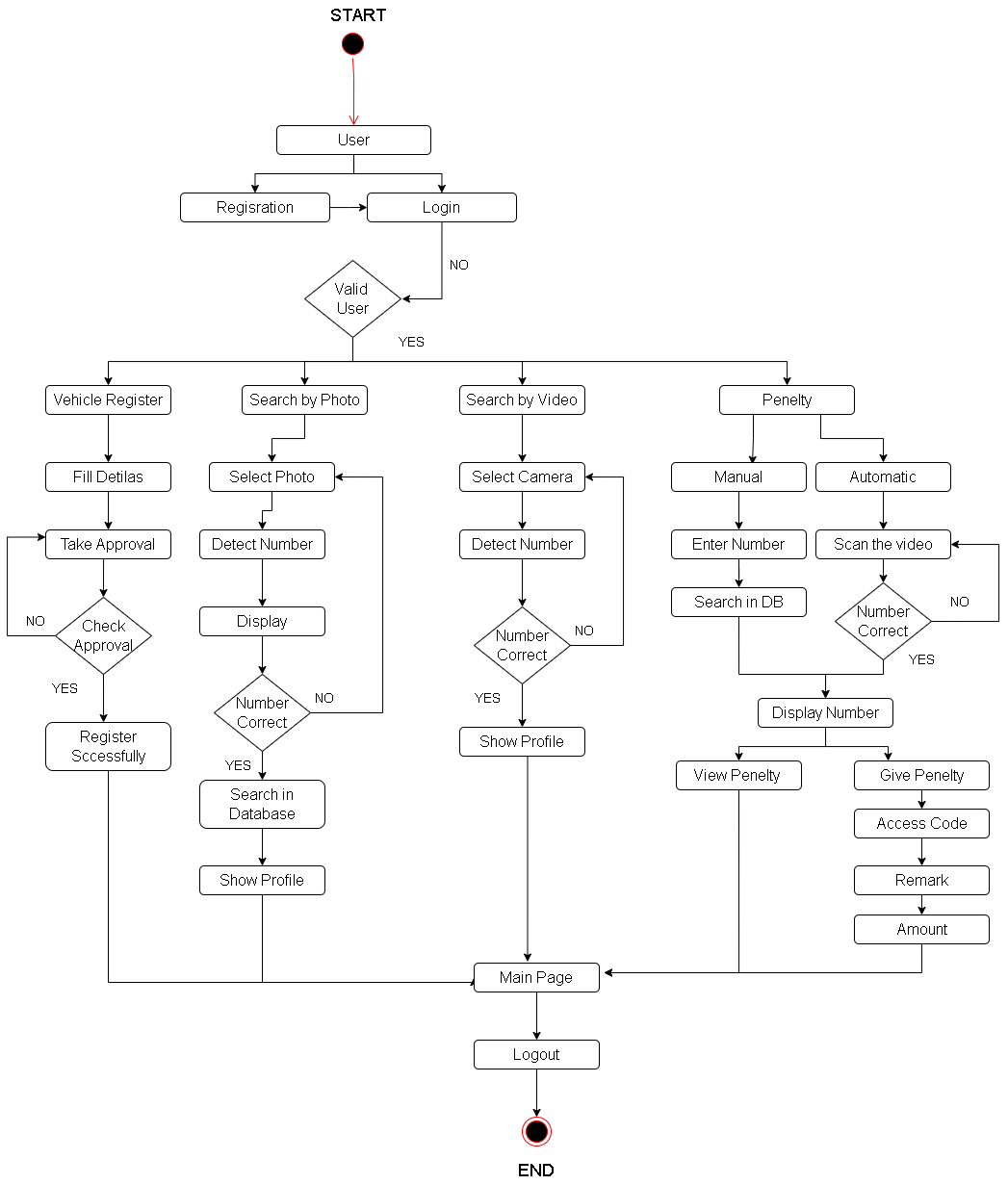
The system aims to provide a user-friendly web interface for campus administrators to access the vehicle monitoring system. The interface should be intuitive and easy to use, allowing administrators to view real-time data, search for specific vehicles, and generate reports.

**2.4 USER OF SYSTEM**

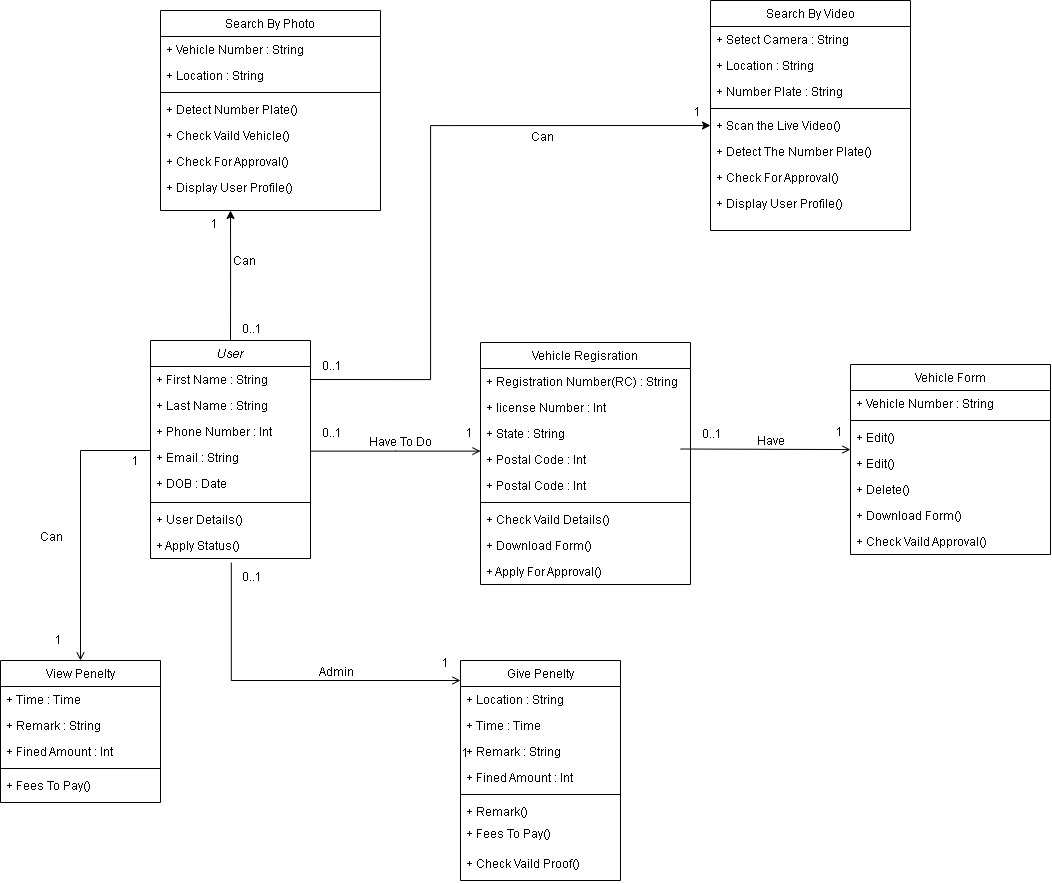
* The vehicle detection system developed in this project is designed to be used by **college administrators and security personnel** responsible for managing the entry and exit of vehicles on the college campus.
* The system provides a user-friendly web interface that allows users to monitor the real-time status of the vehicles entering and exiting the campus, search for specific vehicles, and view their detailed entry and exit logs.
* The system also provides reports and analytics based on the data stored in the MySQL database, which can provide valuable insights for future planning and decision-making. The system can also be used by the college's transportation department to monitor the usage of the college's transport vehicles and analyze their performance.
* The system's reports and analytics can provide valuable insights into the usage patterns, fuel efficiency, and maintenance requirements of the college's transport vehicles. In summary, the vehicle detection system developed in this project is designed to be used by college administrators and security personnel responsible for managing the entry and exit of vehicles on the college campus.
* The system's reports and analytics can also provide valuable insights for the college's transportation department.

**USE CASE DIAGRAM**

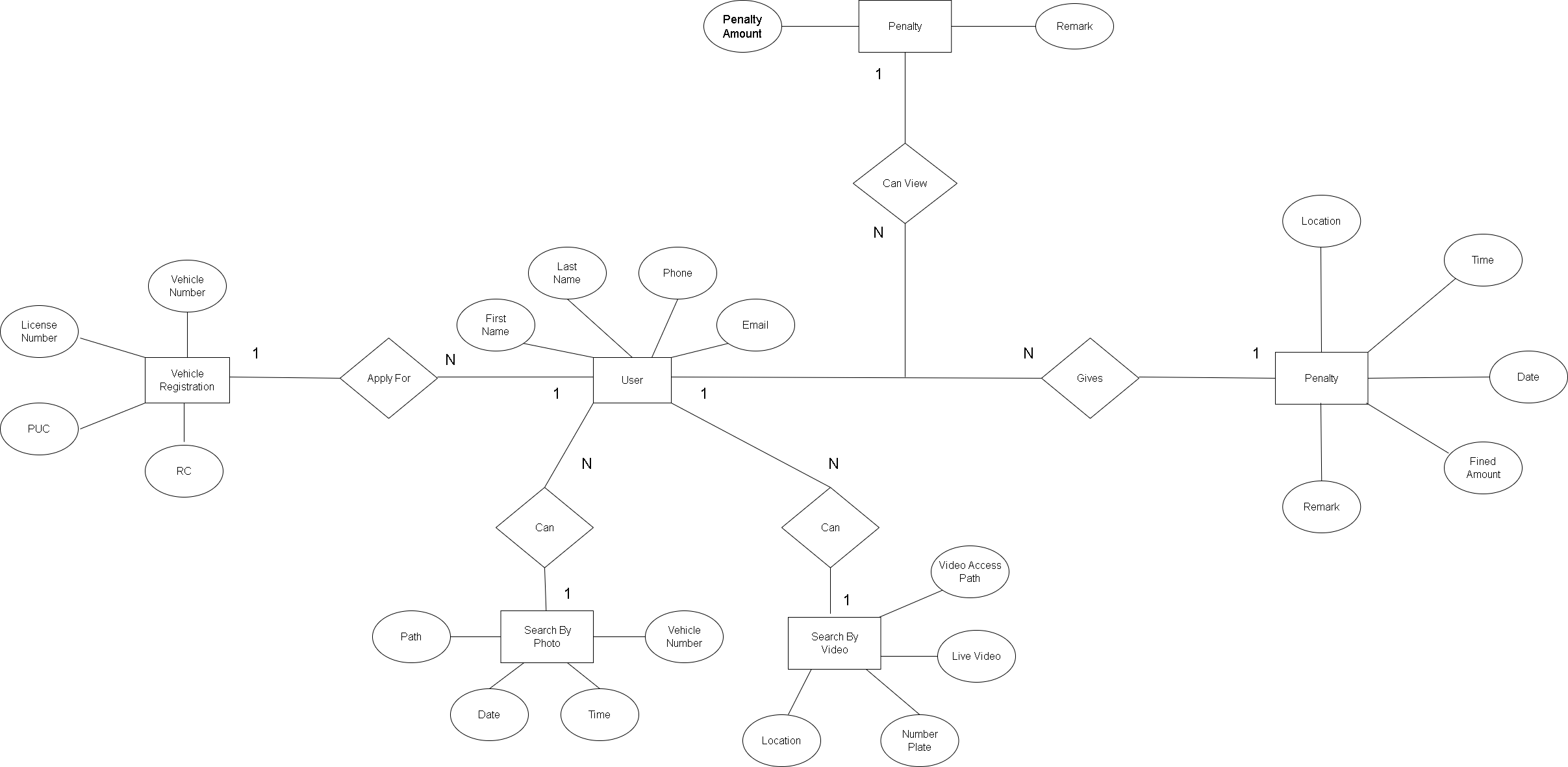
**ACTIVITY DIAGRAM**



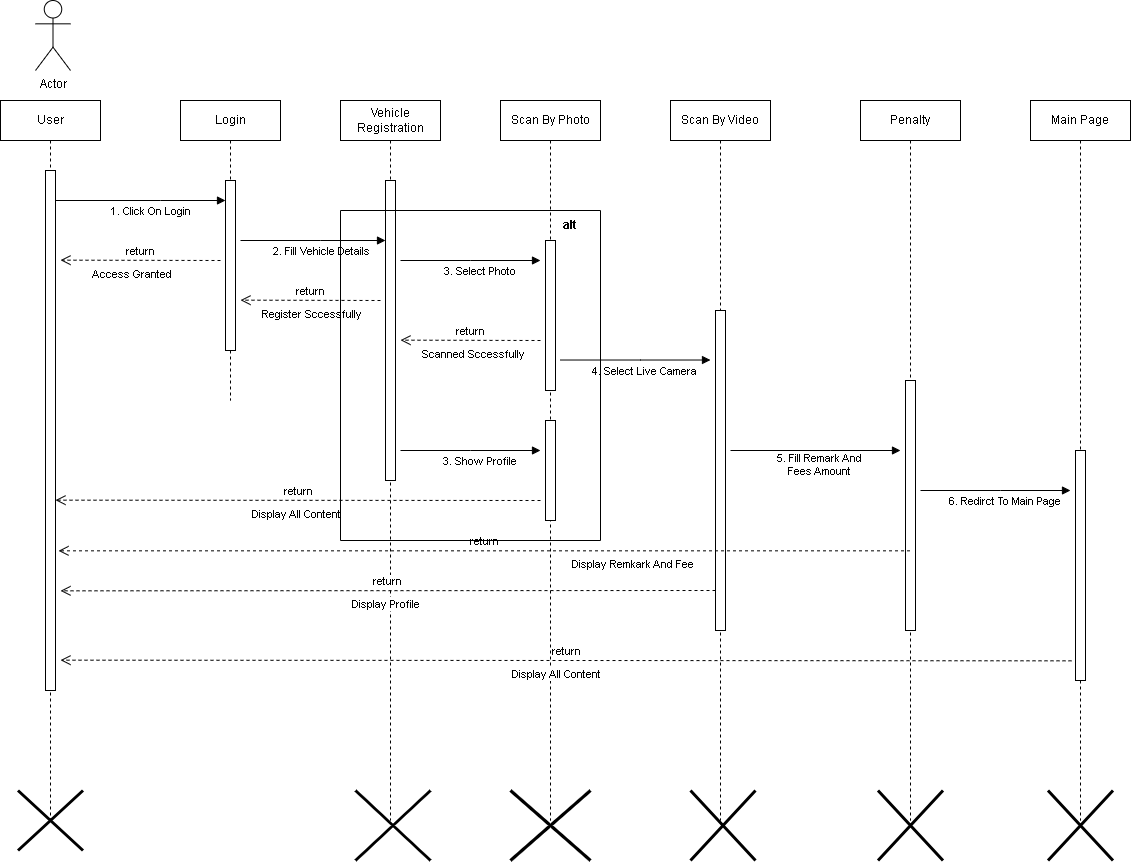
**CLASS DIAGRAM**



**ER DIAGRAM**



**SEQUENCE DIAGRAM**



**TABLE SPECIFICATION**

Registration Data: -

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Fields** | **Description** | **Type** | **Size** | **Constraints** |
| Name | Name of the User | VARCHAR | 20 | Not Null |
| Email | Email id of the User | VARCHAR | 20 | Not Null |
| Phone no | Phone no. of the User | INT | 10 | Primary Key |
| Username | Username of the User | VARCHAR | 20 | Not Null |
| Password | Password for the User | VARCHAR | 20 | Not Null |
| Confirm Password | Confirm Password for the User | VARCHAR | 20 | Not Null |

Vehicle Registration Data: -

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Fields** | **Description** | **Type** | **Size** | **Constraints** |
| First Name | First Name | VARCHAR | 20 | Not Null |
| Last Name | Last Name | VARCHAR | 20 | Not Null |
| Phone | Phone | INT | 10 | Primary Key |
| Email | Email | VARCHAR | 20 | Not Null |
| University | University Name | VARCHAR | 20 | Not Null |
| Course | Course Name | VARCHAR | 20 | Not Null |
| Image Path | Path to where Image stored | VARCHAR | 20 | Not Null |

Scan Data: -

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Fields** | **Description** | **Type** | **Size** | **Constraints** |
| Scan Type | Photo/Video | VARCHAR | 20 | Not Null |
| Date | Current Data | DATE | 20 | Not Null |
| Time | Current Time | TIME | 10 | Not Null |
| Number Detected | Scan Number Detected by Algorithm | VARCHAR | 20 | Not Null |
| Vehicle No | Vehicle Number with Name | VARCHAR | 20 | Primary Name |

Penalty Data: -

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Fields** | **Description** | **Type** | **Size** | **Constraints** |
| Vehicle Number | Vehicle Number | VARCHAR | 20 | Primary |
| Date | Current Data | DATE | 20 | Not Null |
| Time | Current Time | TIME | 10 | Not Null |
| Remark | Reason For Fined | VARCHAR | 100 | Not Null |
| Fined Amount | Fined Amount to be Pay | VARCHAR | 20 | Not Null |