Chapter 1

INTRODUCTION

1.1 Introduction

In today's fast-paced urban landscapes, managing traffic flow is crucial for maintaining safety and efficiency on our roads. Traffic Management Systems (TMS) are the backbone of this effort, employing advanced technology and strategic planning to keep vehicles moving smoothly through city streets.

The **Online Traffic Offense Management System** is easy to use and has a pleasant user interface. It requires system users' credentials in order for the management/staff to access the data and functionalities of the project. The system has 2 types of users which are the **Admin** and the **Staff**. The **Admin** can access and manage all the data and features of the project while the **Staff** has only limited access. This system stores the list of traffic offenses and along with this data is the fine or penalty rate in each of these. In every traffic offense ticket, the violator can be fined for multiple offenses. Traffic offence management is a major concern in cities around the world,Mobilized Online Traffic System is a powerful web/mobile based application that records all the traffic offences committed citywide.

This project also generates a printable **Driver's Info and Records, Traffic Offense Ticket, and Reports.** Traffic offence management is a major concern in cities around the world. Mobilized Online Traffic System is a powerful web/mobile based application that records all the traffic offences committed citywide.

I have created a **PHP Project** entitled **Online Traffic Offense Management System**. The project is a web-based application that manages the driver's traffic violations or offenses records. The system can help the land transport management/department to have an online platform for managing the said record. This can help also for fast and easy retrieval of information and records of each driver.

This web application was developed using **PHP, MySQL Database, HTML, CSS, JavaScript (Ajax & jQuery), Bootstrap, AdminLTE Template, and some other libraries/plugins**. I created this project using XAMPP version 3.30 and does have a **PHP version of 8.0.7**.

Chapter 2

SYSTEM ANALYSIS

2.1 Proposed System

The proposed Traffic Management System (TMS) aims to streamline the monitoring and control of vehicular traffic within urban areas. Leveraging the capabilities of modern technology and database management systems, the system will provide real-time insights into traffic flow, incidents, and infrastructure conditions to facilitate proactive decision-making by traffic authorities. It requires system users credentials in order for the management/staff to access the data and functionalities of the project. This system stores the list of traffic offenses and along with this data is the fine or penalty rate in each of these.

2.2 Objective of Proposed System

The primary objectives of the proposed Traffic Management System are as follows:

- **Enhance Traffic Flow**: By providing real-time insights and dynamic control mechanisms, the system aims to optimize traffic flow, reduce congestion, and minimize travel times for commuters.
- **Improve Road Safety**: Through automated incident detection and rapid response capabilities, the system seeks to enhance road safety by mitigating the impact of accidents and hazards on traffic flow.
- **Increase Efficiency**: By automating routine tasks such as incident detection and signal optimization, the system will enable traffic authorities to allocate resources more efficiently and focus on strategic planning initiatives.
- **Enhance Data-driven Decision Making**: By consolidating and analyzing vast amounts of traffic data, the system will empower traffic engineers and decision-makers to make informed choices regarding traffic management strategies and infrastructure investments.
- **Provide a Seamless User Experience**: Whether it's drivers seeking optimized routes or traffic engineers monitoring traffic conditions, the system will prioritize usability and accessibility to ensure a seamless user experience for all stakeholders.

Chapter 3

SYSTEM REQUIREMENT SPECIFICATION

The Traffic Management System requires robust hardware infrastructure, including servers, network equipment, and sensors for real-time data processing and traffic monitoring. Software components encompass operating systems, databases, and specialized traffic management applications. Infrastructure requirements include a resilient network architecture with sufficient bandwidth and cloud services for scalability. Performance objectives emphasize minimal response times for incident detection, high throughput for data processing, and scalability to accommodate increasing traffic volumes.

3.1 Hardware Requirement

The hardware requirements for the Traffic Management System Management System encompass various components and infrastructure elements essential for its deployment and operation. These include:

- Computer with a 1.1 GHz or faster processor
- Minimum 2GB of RAM or more
- 2.5 GB of available hard-disk space
- 5400 RPM hard drive
- 1366 × 768 or higher-resolution display

3.2 Software Requirement

The software requirements for the Traffic Management System encompass the various software components, frameworks, dependencies necessary for its development, deployment, and operation. These include:

Frontend : HTML, CSS, Java Script, Bootstrap

- Google Chrome/Internet Explorer
- XAMPP (Version-3.7)
- Workspace editor: Visual Studio
- Operating System: Windows 10

Chapter 4

SYSTEM DESIGN

System design encompasses the process of conceptualizing, planning, and architecting the various components and modules of the Traffic Management System (TMS). Through a systematic approach to design, stakeholders can translate requirements into tangible solutions, ensuring scalability, reliability, and maintainability.

4.1 Database Design

The Traffic Management System (TMS) is designed as a comprehensive software solution with modular architecture, featuring a user-friendly interface and scalable infrastructure to efficiently manage traffic monitoring, signal control, incident management, and data analytics, ensuring seamless integration with existing transportation systems while prioritizing data security, system reliability, and user satisfaction to optimize traffic flow and enhance road safety outcomes.

For traffic management system, a streamlined relational database design will organize vehicle details, traffic violations, and flow data efficiently. Normalization techniques will prevent redundancy, and establishing relationships between tables using primary and foreign keys ensures data integrity. Efficient indexing of frequently queried fields will optimize performance for real-time traffic management.

4.1.1 Schema Diagram

The schema diagram provides a visual representation of the database schema, illustrating the tables, fields, relationships, and constraints that comprise the system's data model. It helps stakeholders understand the logical structure of the database and how data entities are organized and related to each other.

In a Traffic Management System, the schema typically includes tables for traffic monitoring (such as sensor data, traffic flow, and congestion levels), signal control (including signal timings, intersections, and priority lanes), incident records (with details on accidents, road closures, and detours), user accounts (for system access and permissions), administrative data (such as system logs and configuration settings.

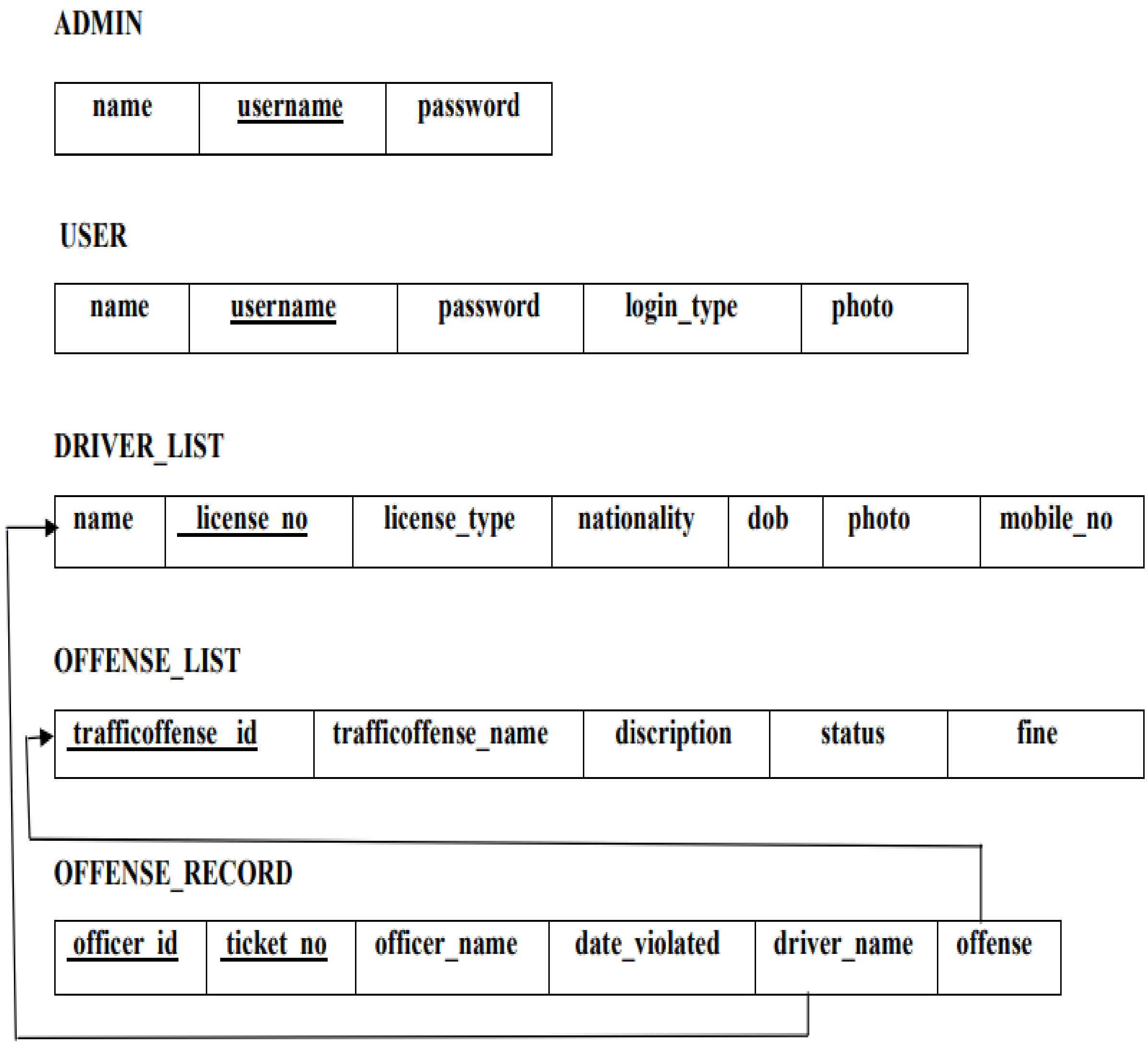


Figure 4.1: Schema Diagram for Traffic management system

Figure 4.1 shows the representation of a schema diagram which contains entities and the attributes that will define that schema. A schema diagram only shows us the database design. It does not show the actual data of the database. Schema can be a single table or it can have more than one table. The schema diagram of the Traffic Management System illustrates the database schema, depicting the tables, attributes, relationships, and constraints. It provides a visual representation of how different entities are interconnected within the database, guiding the development of the underlying data model.

4.1.2 ER Diagram

The entity-relationship (ER) diagram depicts the entities, attributes, and relationships within the database schema, providing insights into the data model's conceptual design. It helps stakeholders visualize the entities and their relationships, identify dependencies, and refine the data model as needed.

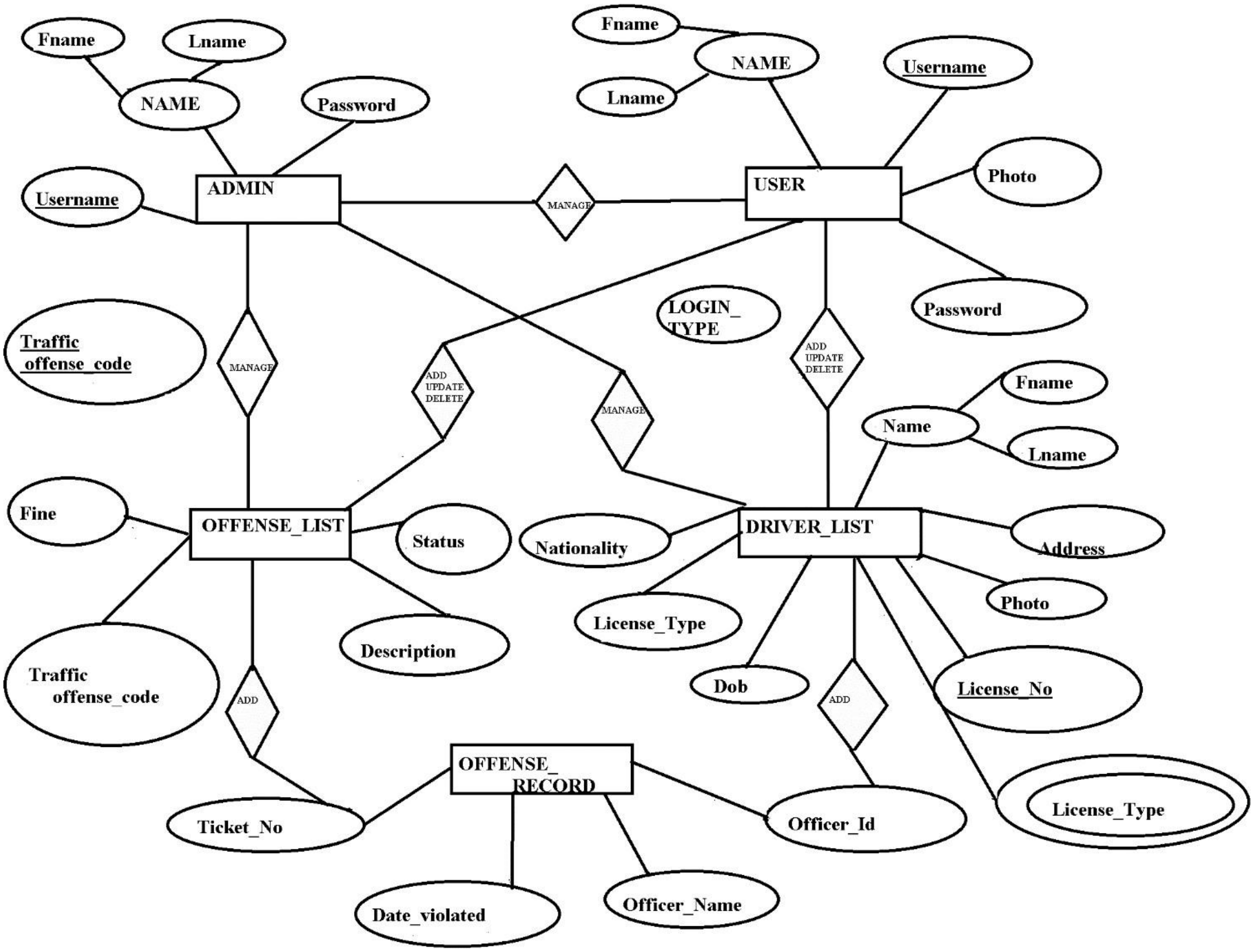


Figure 4.2: ER Diagram for online traffic offense management system

Figure 4.2 shows the representation of ER diagram of Online Traffic Offense Management System. It contains the connection i.e., relation between the entities and the participation ratio and the primarykey is underlined as we see in figure and foreign keys are the keys that relate to primary key of anothertable represented by connecting to that table.

4.1.3 Data-flow Diagram:

The data-flow diagram (DFD) illustrates the flow of data within the system, depicting how information moves between processes, data stores, and external entities. It provides a high-level view of the system's data processing and interaction mechanisms, helping stakeholders understand the data flow and identify potential bottlenecks or inefficiencies.

The data flow diagram illustrates the flow of information within a traffic management system. External entities, namely Admin and User, input data into processes responsible for managing the driver list, offense list, and recording offenses. These processes interact with data stores storing information about drivers and offenses. Admin and User inputs trigger processes to update these data stores, reflecting changes made to the driver and offense lists, as well as recording new offenses. This Figure 4.3 provides a concise overview of how data moves through the system, facilitating effective management of drivers and offenses within the traffic management system.

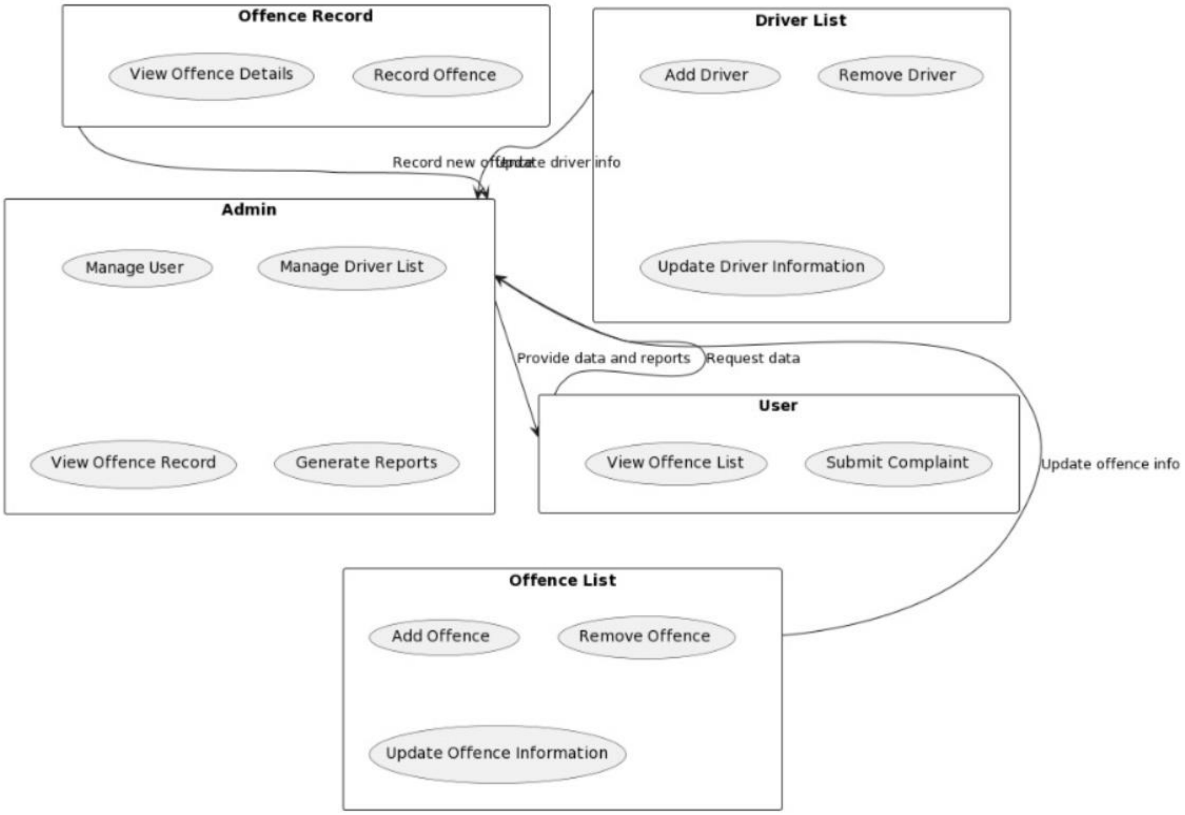


Figure 4.3: Data-Flow Diagram for Traffic Management System

4.2 Implementations

In implementing the Online Traffic Offense Management System, we first established our development environment by configuring XAMPP version 3.30 and PHP version 8.0.7 to ensure compatibility. Leveraging PHP, MySQL Database, HTML, CSS, JavaScript (Ajax & jQuery), Bootstrap, AdminLTE Template, and other libraries, we constructed a robust technology stack that facilitated the system's development. Our database design centered around a well-defined schema, meticulously organizing tables, fields, and relationships to efficiently store and manage system data. Adopting a client-server architecture with an emphasis on the MVC (Model-View-Controller) pattern, we orchestrated the interaction between different components to ensure seamless operation. During the implementation phase of the Traffic Management System, the focus is on translating the design specifications into functional modules and components.

4.2.1 Libraries and Framework PHP

PHP is Hypertext Pre-processor is a general-purpose programming language originally designed for web development.

HTML

Hypertext Markup Language is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets and scripting languages such as JavaScript.

CSS

Cascading Style Sheets is a style sheet language used for describing the presentation of a document written in a markup language like HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript. Functional Modules

JAVASCRIPT

JavaScript, often abbreviated as JS, is a high-level, interpreted scripting language that conforms to the ECMAScript specification. JavaScript has curly-bracket syntax, dynamic typing, prototype-based object-orientation, and first-class functions.

4.2.2 Backend(MySQL)

Database

A Database Management System (DBMS) is computer software designed for the purpose of managingdatabases, a large set of structured data, and run operations on the data requested by numerous users. Typical examples of DBMSs include Oracle, DB2, Microsoft Access, Microsoft SQL Server, Firebird, PostgreSQL, MySQL, SQLite, FileMaker and Sybase Adaptive Server Enterprise. Typical examples of DBMS use include accounting, human resources and customer support systems. Originally found onlyin large companies with the computer hardware needed to support large data sets, DBMSs have morerecently emerged as a fairly standard part of any company back office.

A DBMS is a complex set of software programs that controls the organization, storage, management, and retrieval of data in a database. A DBMS includes:

- A modeling language to define the schema of each database hosted in the DBMS, according to theDBMS data model.
- The dominant model in use today is the ad hoc one embedded in SQL, despite the objections of purists who believe this model is a corruption of the relational model, since it violates several of its fundamental principles for the sake of practicality and performance. Many DBMSs also support the Open Database Connectivity API that supports a standard way for programmers to access the DBMS.
- Data security prevents unauthorized users from viewing or updating the database. Using passwords,users are allowed access to the entire database or subsets of it called sub schemas. For example, an employee database can contain all the data about an individual employee, but one group of users may be authorized to view only payroll data, while others are allowed access to only work history and student data.
- If the DBMS provides a way to interactively enter and update the database, as well as interrogate it,this capability allows for managing personal databases. However, it may not leave an audit trail of actions or provide the kinds of controls necessary in a multi-user organization.

4.2.3 Structured Query Language (SQL)

SQL: Structured Query Language (SQL) is the language used to manipulate relational databases. SQLis tied very closely with the relational model.

- In the relational model, data is stored in structures called relations or tables. SQL statements areissued for the purpose of: online traffic offense management system
- Data definition: Defining tables and structures in the database (DDL used to create, alter and dropschema objects such as tables and indexes) .

Stored procedure

A stored procedure is a precompiled set of SQL statements stored in a database. It enables modular programming, accepts input parameters, performs operations, and can return results. Stored procedures enhance security, promote code reusability, and improve performance by reducing data transfer and optimizing execution plans.

Routine name: proc

Type: procedure

Definition: Select * drivers_list;

Triggers

Triggers are like automatic actions in a database. They are set to occur when certain events happen, like adding, changing, or deleting data. They're handy for enforcing rules or keeping track of changes without needing manual intervention.

Triggers used :

- Trigger name: on insert Table: register Time: after Event: insert INSERT INTO trig VALUES(null,NEW.id,'Driver Inserted',NOW())
- Trigger name: on delete Table: register Time: after Event: delete Definition: INSERT INTO trig VALUES(null,OLD.id,'Driver Deleted',NOW())
- Trigger name: on update Table: register Time: after Event: update Definition: INSERT INTO trig VALUES(null,NEW.id,driver updated,NOW())

4.2.4 Modules Description

The Online Traffic Offense Management System comprises several interconnected modules designed to streamline the management of traffic violations records. The Authentication Module ensures secure access to the system, verifying user credentials and granting appropriate access privileges. The Dashboard Module provides a centralized interface for monitoring system activities and accessing key functionalities. The Offense Management Module facilitates the creation, modification, and deletion of offense records, including details such as offense type and associated penalties. User Management allows administrators to manage user accounts and access permissions, while the Driver Management Module organizes driver profiles and offense history. The Offense Ticket Management Module handles the generation and management of offense tickets, and the Reporting Module generates comprehensive reports for data analysis. Additionally, the System Configuration Module enables administrators to customize system settings, and the Account Management Module ensures the security of user accounts. Together, these modules enhance the efficiency, usability, and security of the system in managing traffic violations records.

Authentication Module:

module handles user authentication, ensuring secure access to the system. It verifies user credentials and grants appropriate access privileges based on the user role (Admin or Staff).

- **Dashboard This Module**

  The dashboard module provides a centralized interface for users to monitor system activities, view statistics, and access essential functionalities at a glance.

- **Offense Management Module**

  Facilitates the management of traffic offense records, including the creation, modification, and deletion of offense entries. It also manages associated fine or penalty rates for each offense.

- **User Management Module**

  Allows administrators to manage user accounts and access privileges within the system. It includes functionalities for user registration, modification, and deletion.

- **Driver Management Module**

  Handles the management of driver profiles and records within the system. It allows for the creation, modification, and deletion of driver entries, as well as tracking of offense history for each driver.

- **Offense Ticket Management Module**

  Enables the generation and management of offense tickets for violators. It includes functionalities for creating, printing, and archiving offense tickets.

- **Reporting Module**

  Facilitates the generation of comprehensive reports based on system data. It includes functionalities for generating date-wise reports, analyzing offense trends, and extracting actionable insights.

- **System Configuration Module**

  Allows administrators to configure system settings and parameters to suit specific requirements. It includes functionalities for updating system information and settings.

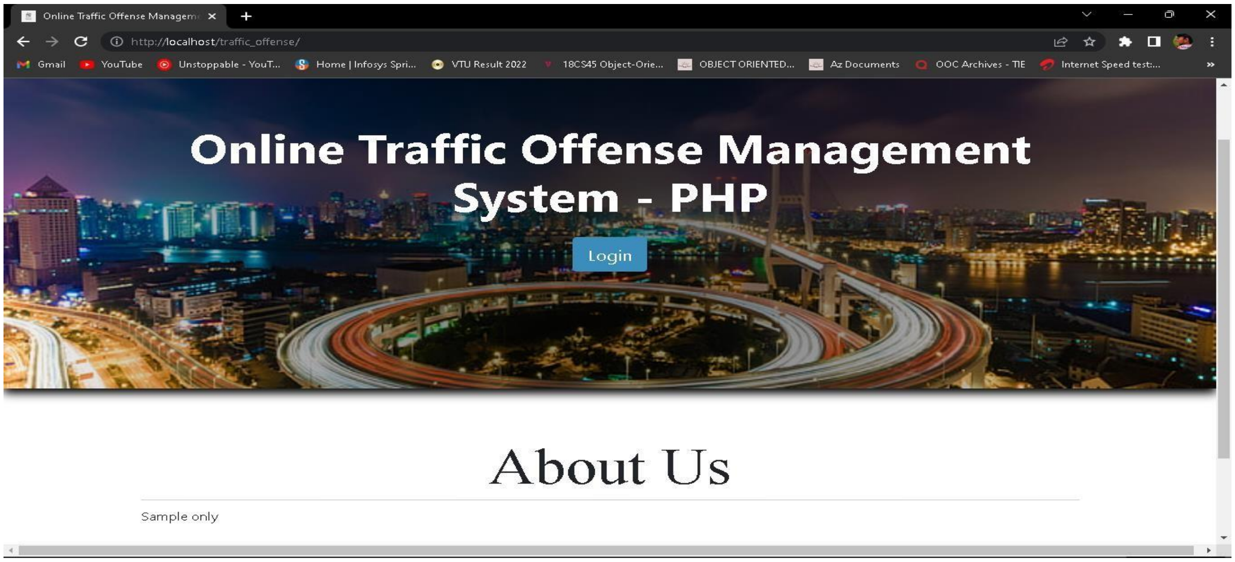
- **Account Management Module**

  Facilitates the management of user accounts and credentials within the system. It includes functionalities for updating account credentials, resetting passwords, and enforcing security measures.

Chapter 5

RESULT

Login Portal



http://localhost/traffic_offense/

Online Traffic Offense Management System - PHP

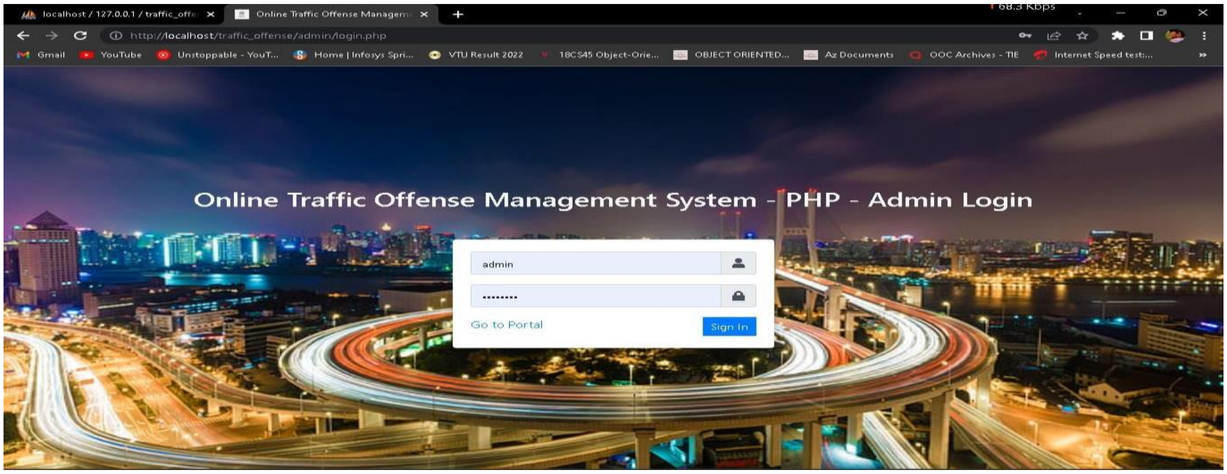
About Us

Sample only

Figure 5.1 Login Portal

Figure 5.1and 5.2 represents Admin or User needs to enter User name and Password and press Login.If username and password correct then admin will be switched on to next page. If incorrect passwordthen he is not able to log in.

Admin Login page



http://localhost/traffic_offense/admin/login.php

Online Traffic Offense Management System - PHP - Admin Login

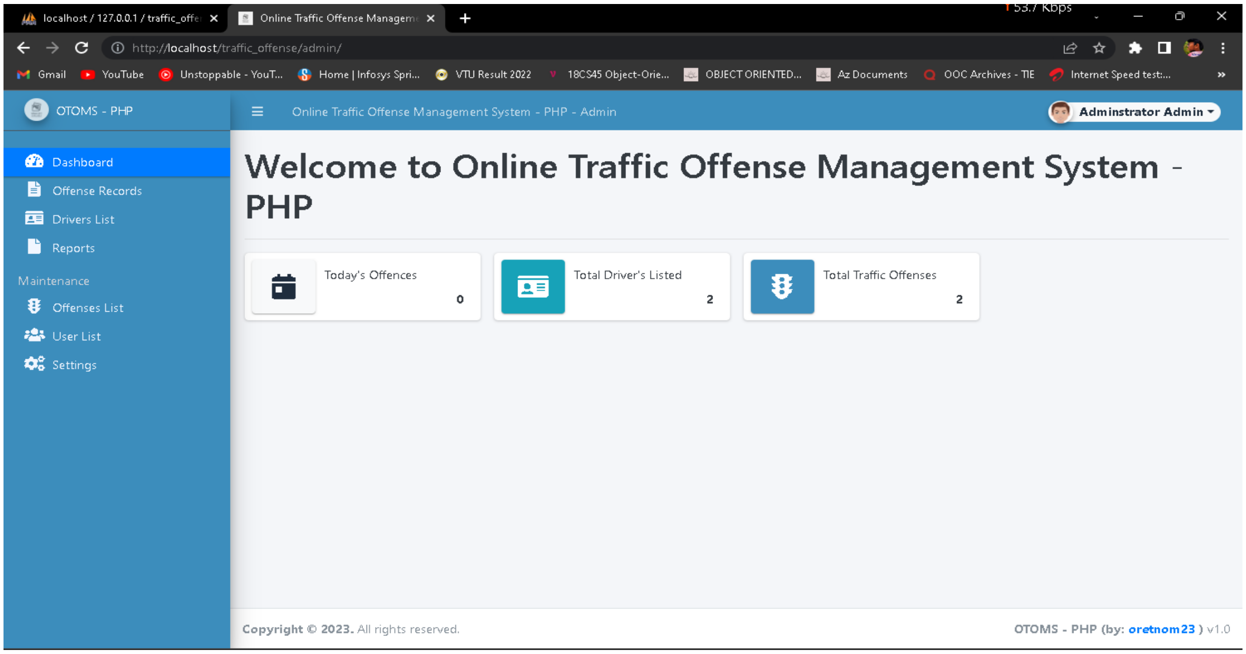
admin

••••••••

Figure 5.2 Admin Login

Dashboard

Figure 5.3 shows a Dashboard of Our Project In this page Dashboard we will get all the information of Total offense's and Total driver's List . It is a Welcome page for Online Traffic Offense Management System



Welcome to Online Traffic Offense Management System - PHP

Today's Offences 0

Total Driver's Listed 2

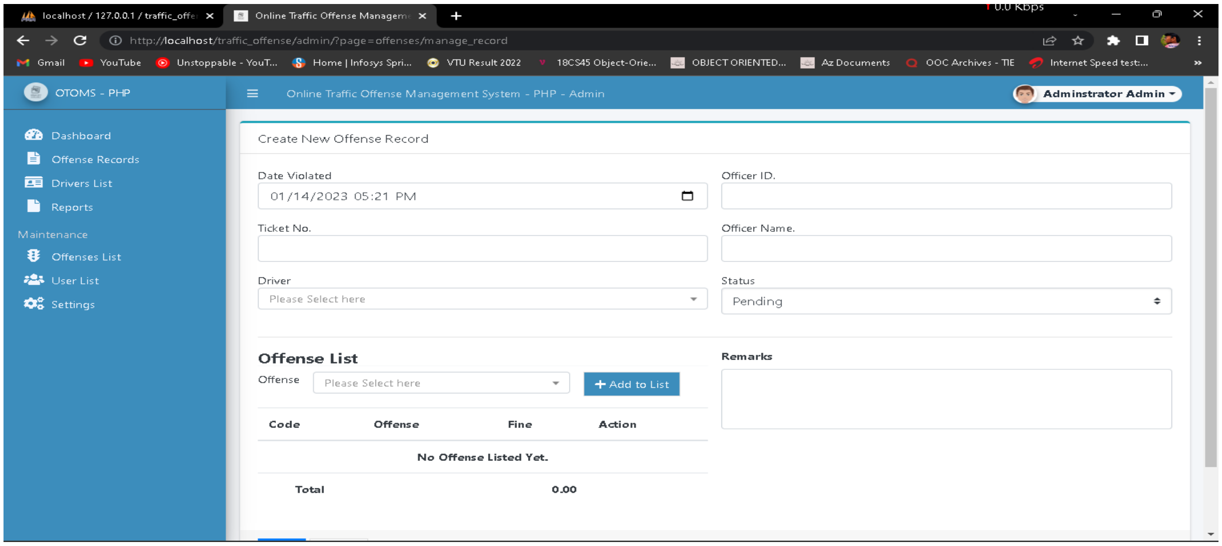
Total Traffic Offenses 2

Copyright © 2023. All rights reserved.

OTOMS - PHP (by: oretnom23) v1.0

Figure 5.3 Dashboard

Creation of Offense Record



Create New Offense Record

Date Violated: 01/14/2023 05:21 PM

Officer ID.:

Ticket No.:

Officer Name.:

Driver: Please Select here

Status: Pending

Offense List

Offense: Please Select here

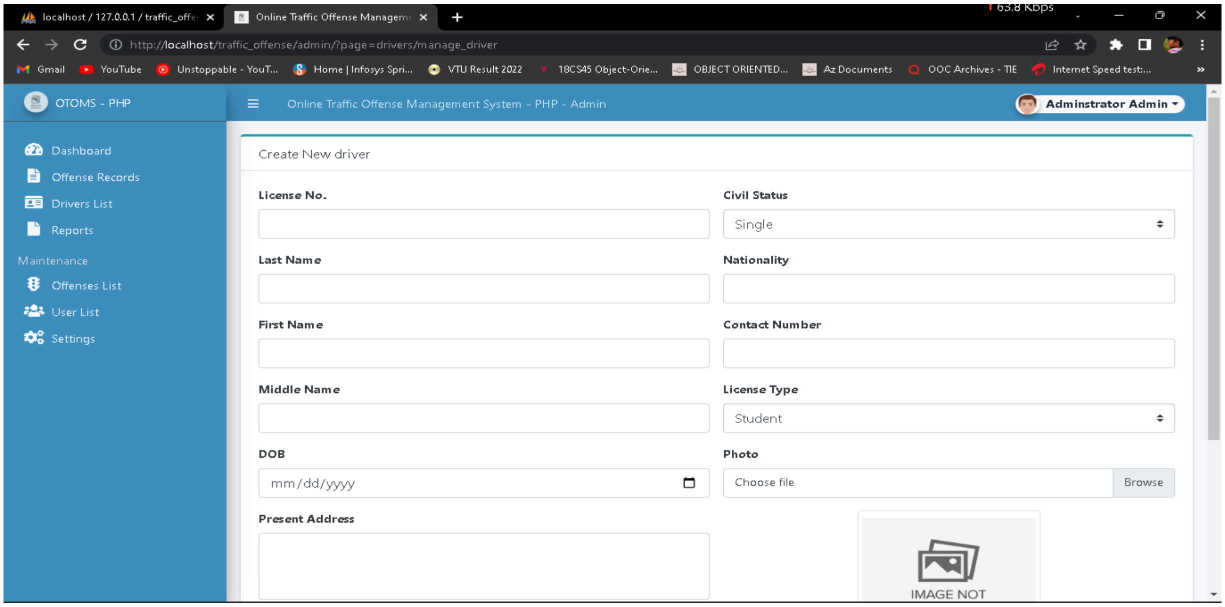
Remarks

Code	Offense	Fine	Action
	No Offense Listed Yet.		
Total		0.00	

Figure 5.4 Creation of Offense Record

Figure 5.4 represents the creation of offense list into the offense entity. The admin can even logout ofthe session anytime by pressing the log out icon.

Creation of Driver List



Create New driver

License No.

Civil Status: Single

Last Name

Nationality

First Name

Contact Number

Middle Name

License Type: Student

DOB: mm/dd/yyyy

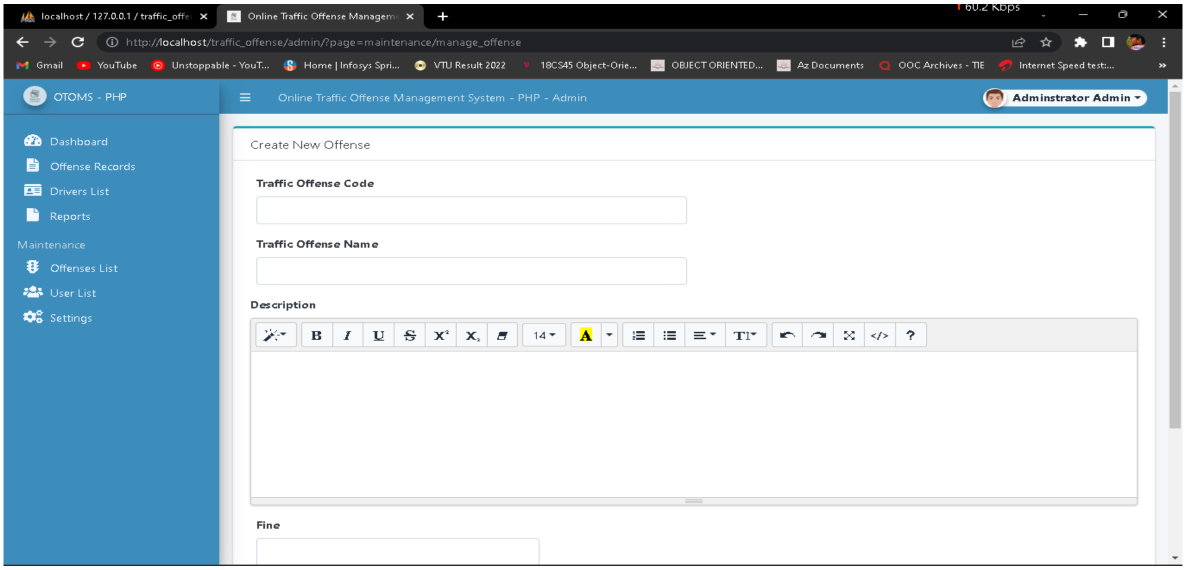
Photo: Choose file

Present Address

Figure 5.5: Creation of Driver List

Figure 5.5 represents addition of new offense into the offense entity. Once added you get the message that “Driver added successfully”.

Addition of New Offense



Create New Offense

Traffic Offense Code

Traffic Offense Name

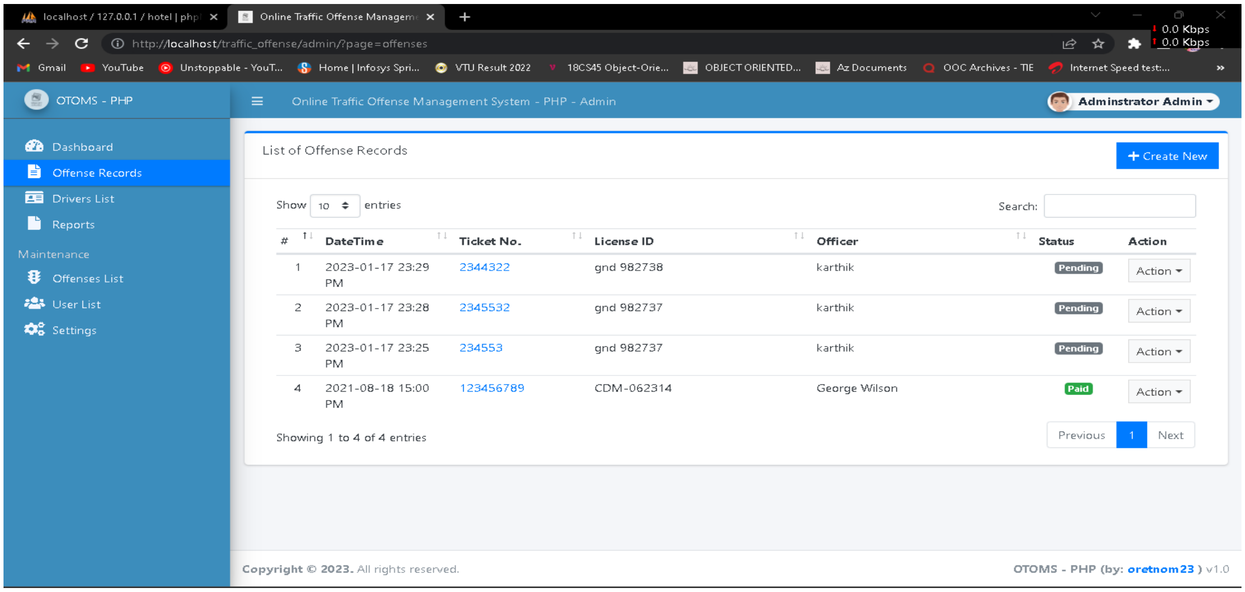
Description

Fine

Figure 5.6 Addition of New Offense

Figure 5.6 represents addition of new offense into the offense entity. Once added you get the message that “offense added successfully”

List of Offense Record



List of Offense Records

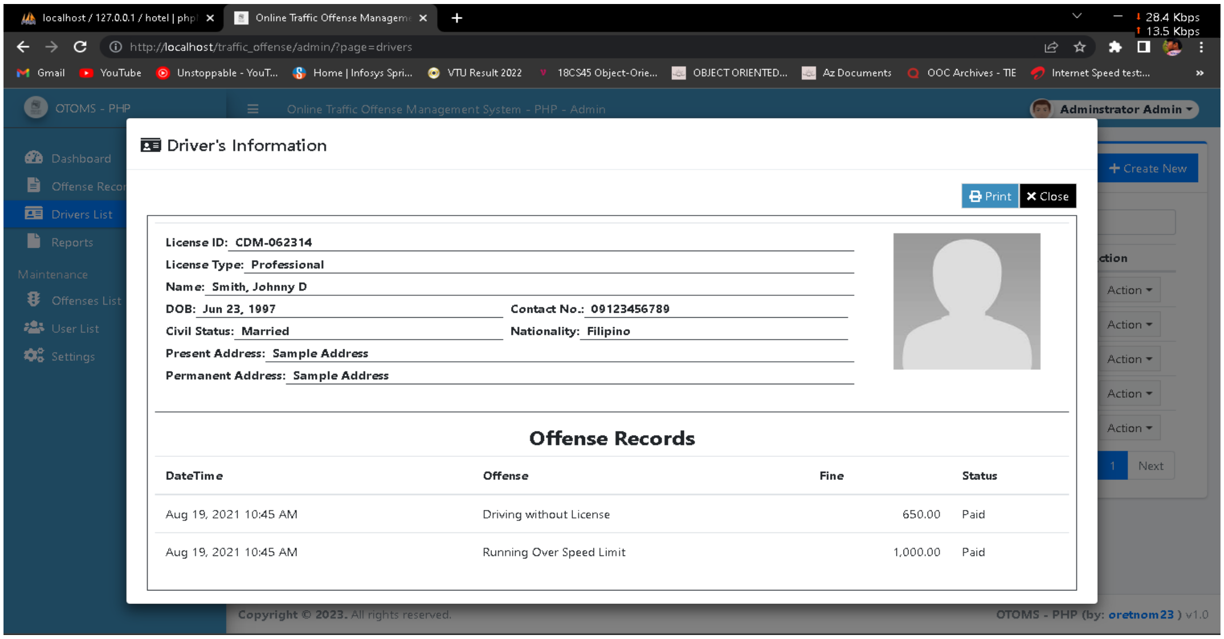
Show 10 entries Search:

#	DateTime	Ticket No.	License ID	Officer	Status	Action
1	2023-01-17 23:29 PM	2344322	gnd 982738	karthik	Pending	Action
2	2023-01-17 23:28 PM	2345532	gnd 982737	karthik	Pending	Action
3	2023-01-17 23:25 PM	234553	gnd 982737	karthik	Pending	Action
4	2021-08-18 15:00 PM	123456789	CDM-062314	George Wilson	Paid	Action

Showing 1 to 4 of 4 entries Previous 1 Next

Figure 5.7 List of Offense Record

Driver’s Information:


localhost / 127.0.0.1 / hotel | php
Online Traffic Offense Managem
http://localhost/traffic_offense/admin/?page=drivers
28.4 Kbps
13.5 Kbps
OTOMS - PHP
Online Traffic Offense Management System - PHP - Admin
Adminstrator Admin
Dashboard
Offense Recor
Drivers List
Reports
Maintenance
Offenses List
User List
Settings
Create New
Print
Close
Copyright © 2023. All rights reserved.
OTOMS - PHP (by: oretnom23) v1.0

Driver's Information

License ID: CDM-062314
License Type: Professional
Name: Smith, Johnny D
DOB: Jun 23, 1997 Contact No.: 09123456789
Civil Status: Married Nationality: Filipino
Present Address: Sample Address
Permanent Address: Sample Address

Offense Records

DateTime	Offense	Fine	Status
Aug 19, 2021 10:45 AM	Driving without License	650.00	Paid
Aug 19, 2021 10:45 AM	Running Over Speed Limit	1,000.00	Paid

Figure 5.8 Driver’s Information

Figure 5.8 shows a driver’s information if user click on view it will show the detail information ofdriver and their offense record .We can print the information if needed.

Chapter 6

CONCULSION

The following conclusions can be deduced from the development of the project

- Automation of the entire system improves the efficiency
- It gives appropriate access to the authorized users depending on their permissions.
- It effectively overcomes the delay in communications.
- Updating of information becomes so easier.
- System security, data security and reliability are the striking features.
- The System has adequate scope for modification in future if it is necessary.

‘Online Traffic Offense Management’ provide more efficient ways to manage a school it reduces manual paperwork in order to maintain traffic offense record . For implementing this system, PHP, HTML, CSS, JavaScript and MySql are used.

The system comprises of following features:

- Login Portal
- Secure Login/Logout
- Dashboard
- Manage Offense List
- Manage User List
- Manage Drivers List
- Manage Offense Ticket/Records
- Print Driver's Offense Ticket
- Print Driver's Information and Offense Records
- Generate a printable date-wise Report
- Update System Information

INDEX

LIST OF FIGURES

REFERENCES

1. Fundamentals of Database Systems, and B. Navathe, 7th Edition, 2017,Pearson.
2. Database management systems, , and Gehrke, 3rd Edition, 2014, McGraw Hill
3. https://en.wikipedia.org/wiki/Traffic_violations_reciprocity
4. https://www.w3schools.in/mysql/