**COLLEGE TRANSPORT MANAGEMENT**

**ABSTRACT**

This abstract outlines the development of a comprehensive College Transport Management Website (CTMW) employing a robust stack of HTML, CSS, Bootstrap for frontend development, and PHP coupled with MySQL for backend functionalities. The CTMW serves as a centralized platform to streamline and enhance transportation services within the college campus.

The website's frontend, crafted using HTML, CSS, and Bootstrap, ensures a visually appealing and responsive interface accessible across devices. It offers an intuitive user experience, facilitating easy navigation and interaction. Leveraging Bootstrap's framework, the design prioritizes responsiveness, adapting seamlessly to various screen sizes, optimizing accessibility for all users.

On the backend, PHP acts as the server-side scripting language, enabling dynamic functionalities and seamless data processing. Integrated with MySQL, a robust relational database management system, the website securely stores and manages transportation-related data such as schedules, routes, user information, and vehicle details. The PHP-MySQL synergy facilitates efficient data retrieval, storage, and manipulation, ensuring the reliability and scalability of the system.

**INTRODUCTION**

In the modern landscape of higher education, effective transportation management within college campuses plays a pivotal role in facilitating seamless connectivity and convenience for students, faculty, and staff. This introduction sets the stage for the development and implementation of a dynamic College Transport Management Website (CTMW) integrating HTML, CSS, Bootstrap, PHP, and MySQL, aimed at revolutionizing the transportation experience within our institution.

The CTMW emerges as a solution designed to streamline and optimize the complexities of managing campus transportation systems. Leveraging the collective strengths of HTML, CSS, and Bootstrap, the website's frontend embodies an aesthetically pleasing and user-centric design. It ensures a visually engaging interface coupled with responsive elements, adapting seamlessly across a spectrum of devices, thus fostering accessibility and user interaction.

Complementing the frontend prowess, the website's backend is fortified with PHP and MySQL, constituting the backbone of a powerful and functional system. PHP, as the server-side scripting language, orchestrates dynamic content generation and efficient data handling, while MySQL, a robust relational database management system, stores, manages, and retrieves transportation-related information securely and reliably.

**SYSTEM ANALYSIS**

**EXISTING SYSTEM**

The existing system for college transport management often involves a mix of manual processes, basic software solutions, or outsourced transportation services. These systems typically face challenges in efficiency, communication, and data management. Here's an overview of the common aspects of an existing college transport management system:

1. **Manual Booking and Scheduling:** Students, faculty, or staff members may have to physically visit an office or make phone calls to book transport services. Schedules and routes might be displayed on notice boards or through printed materials, leading to potential miscommunications or outdated information.
2. **Paper-based Record-keeping:** The management of transportation-related data, such as passenger information, routes, vehicle maintenance records, and schedules, often relies on manual paperwork or basic spreadsheets. This manual process can be prone to errors, data loss, or inefficiencies.
3. **Communication Challenges:** Communication regarding route changes, delays, or cancellations might be disseminated through traditional methods like emails or notice boards. This method might not reach all stakeholders promptly, leading to confusion or inconvenience.

The existing system's limitations often result in inefficiencies, communication gaps, and suboptimal experiences for the college community using transportation services. These challenges highlight the need for a more streamlined, technology-driven, and centralized Transport Management System to enhance the overall transport experience within the college campus.

**PROPOSED SYSTEM**

The proposed Transport Management System (TMS) for a college aims to revolutionize the existing transport infrastructure by leveraging advanced technologies such as HTML, CSS, Bootstrap, PHP, and MySQL to address the limitations of the current system. Here's an outline of the proposed system:

1. **Online Booking and Scheduling:** The TMS will offer an intuitive web-based platform allowing students, faculty, and staff to book transport services online. Users can view schedules, select routes, and reserve seats conveniently through a user-friendly interface.
2. **Centralized Information Hub:** The proposed system will act as a centralized repository for all transport-related information. This includes details on routes, schedules, vehicle availability, driver information, and maintenance records, ensuring data accuracy and easy access.
3. **User Authentication and Profiles:** Secure user authentication will be implemented, enabling personalized profiles for users. This ensures authorized access to specific services and allows for tailored experiences based on user roles (e.g., students, faculty, administrators).
4. **Dynamic Administrative Dashboard:** An administrative dashboard will provide authorized personnel with insights into operations, allowing them to manage routes, update schedules, track vehicles, and generate reports for informed decision-making.

The proposed Transport Management System represents a paradigm shift from the manual and fragmented approach of the existing system to an integrated, technology-driven solution. By combining HTML, CSS, Bootstrap, PHP, and MySQL, the system aims to provide a seamless, efficient, and user-centric transport experience for all members of the college community.

**MODULE DESCRIPTION**

1. **User Authentication Module:**
   * Manages user registration, login, and logout functionalities for students, drivers, and administrators.
   * Implements secure password storage and retrieval.
   * Differentiates between different user roles such as students, drivers, and administrators.
2. **Route Management Module:**
   * Enables administrators to create, modify, and manage predefined transport routes.
   * Allows for the addition or removal of stops along routes based on demand.
   * Provides a visual representation of transport routes on a map.
3. **Vehicle Tracking and Assignment Module:**
   * Tracks the real-time location of college transport vehicles using GPS technology.
   * Allows administrators to assign vehicles to specific routes and manage the availability of vehicles.
   * Provides an interface for drivers to update their availability status and report any issues.
4. **Driver Management Module:**
   * Manages information about drivers, including their contact details, license information, and schedules.
   * Allows administrators to assign drivers to specific routes and vehicles.
   * Provides a platform for drivers to view their assigned routes and update their availability.
5. **Vehicle Maintenance Module:**
   * Tracks the maintenance history of each transport vehicle, including service dates and repairs.
   * Sends notifications for upcoming maintenance schedules or inspections.
   * Enables administrators to manage and schedule routine maintenance tasks.
6. **Attendance Tracking Module:**
   * Allows drivers to take attendance of students boarding and alighting from the transport vehicles.
   * Provides administrators with access to attendance records for monitoring and reporting.
   * Generates reports on attendance patterns and trends.
7. **Security Module:**
   * Implements robust security measures to protect user data, transport schedules, and vehicle information.
   * Manages access control to ensure that users have appropriate permissions.
   * Regularly updates security protocols to address potential vulnerabilities.
8. **Database Management Module:**
   * Manages the storage and retrieval of transport-related data using MySQL databases.
   * Ensures data integrity, reliability, and security.
   * Implements backup and recovery mechanisms to prevent data loss.

This modular breakdown provides a foundation for the development of a comprehensive College Transport Management System, ensuring efficient and transparent management of transportation services for students and administrators. The use of HTML, CSS, Bootstrap, PHP, and MySQL ensures a responsive and dynamic web application tailored to the needs of the college transport system.

**SYSTEM SPECIFICATION**

**HARDWARE SPECIFICATION**

|  |  |
| --- | --- |
| System | HP 15s |
| Processor | Ryzen 5 2.1 GHz |
| Storage | 512 GB SSD |
| RAM | 16 GB |
| Monitor | Integrated Monitor |
| Mouse | Integrated Trackpad |
| Keyboard | Integrated Keyboard |

**OPERATING SYSTEM**

|  |  |
| --- | --- |
| Operating System | Windows 11 |
| Front End | PHP Version 8 |
| Back End | MySQL Version 8 |
| Server | XAMPP |

**SOFTWARE SPECIFICATION**

**SOFTWARE DESCRIPTION**

**XAMPP:**

XAMPP is an [open-source](https://en.wikipedia.org/wiki/Free_software) [cross-platform](https://en.wikipedia.org/wiki/Cross-platform) [web server](https://en.wikipedia.org/wiki/Web_server) [solution stack](https://en.wikipedia.org/wiki/Solution_stack) package developed by Apache Friends, consisting mainly of the [Apache HTTP Server](https://en.wikipedia.org/wiki/Apache_HTTP_Server), [Maria DB](https://en.wikipedia.org/wiki/MariaDB) [database](https://en.wikipedia.org/wiki/Database), and [interpreters](https://en.wikipedia.org/wiki/Interpreter_%28computing%29) for scripts written in the [PHP](https://en.wikipedia.org/wiki/PHP) and [Perl](https://en.wikipedia.org/wiki/Perl) [programming languages](https://en.wikipedia.org/wiki/Programming_language). XAMPP stands for Cross-Platform (X), Apache (A), Maria DB (M), PHP (P), and Perl (P). It is a simple, lightweight Apache distribution that makes it extremely easy for developers to create a local web server for testing and deployment purposes.

Everything needed to set up a web server – server application (Apache), database (Maria DB), and scripting language (PHP) – is included in an extractable file. XAMPP is also cross-platform, which means it works equally well on Linux, Mac, and Windows.

XAMPP's designers intended it for use only as a development tool, to allow website designers and programmers to test their work on their computers without any access to the Internet.

**CROSS-PLATFORM**

Cross-platform software is a type of software application that works on multiple operating systems or devices, which are often referred to as platforms. A platform means an operating system such as Windows, Mac OS, Android, or iOS. When a software application works on more than one platform, the user can utilize the software on a wider choice of devices and computers.

**BENEFITS OF CROSS-PLATFORM**

The benefit of a cross-platform software app or program is that you can use the same program whether you’re on a Windows PC or whether you’re logging in from your laptop or smartphone. The Microsoft Office suite of applications, which includes Word, Excel, and PowerPoint, is available on Windows, Mac OS, iOS (iPhone/iPad), and Android. While there are differences based on how the platforms work, you’ll have a similar experience within the application between all of your devices.

Having a similar experience across any platform means there’s a much smaller learning curve if one even exists at all, so you’ll be more productive and be able to use a software product you’re familiar with regardless of the operating system or device you choose. In addition, your files can be moved much more easily between your devices so you can use the software with whatever device you have with you at the time. And there’s a way to keep all of your work in sync across all of your devices, by using the cloud.

**EXAMPLES OF CROSS-PLATFORM**

**Unity 3D**

First, let’s talk about Unity3D. I think the game engine should be preferred by people who want to write mobile games.  
You can develop games on 17 platforms using multiple languages, including Linux. Of course, iOS, Android, and Windows Phone is also the most ideal game engine to develop games.

You can develop your application using C #, JS, and C ++.

Link to: [https://unity3d.com](https://unity3d.com/)

# Xamarin

Xamarin Some time ago, it was purchased by Microsoft and is a perfect fit for developers using C #.

Because it is a C # language, it has a lot of documentation, and because of Microsoft support, Xamarin is the choice for C # developers.

In addition, you can do everything you can do in Objective-C, Swift, and Java with the Xamarin library.

Link to: [https://xamarin.com](https://xamarin.com/)

# React Native

React Native is an open-source JavaScript library developed by the new generation of React — Facebook, which was open to Github in 2013. Native application creation means writing applications only for a specific operating system. React Native helps developers reuse their code over the web and on mobile. Developers will not have to create the same app from scratch for iOS and Android. They will be able to reuse the code in each operating system. The great thing about React Native is that there is little difference between a finished application in Objective-C or Java and an application built using React Native. Android and iOS code development environments are very different. So it takes time to remove the application to two different platforms. However, with React Native, only one developer can write on different mobile operating systems.

**APACHE:**

The Apache HTTP Server, colloquially called Apache is a [free and open-source](https://en.wikipedia.org/wiki/Free_and_open-source) [cross-platform](https://en.wikipedia.org/wiki/Cross-platform) [web server](https://en.wikipedia.org/wiki/Web_server) software, released under the terms of [Apache License](https://en.wikipedia.org/wiki/Apache_License) 2.0. Apache is developed and maintained by an open community of developers under the auspices of the [Apache Software Foundation](https://en.wikipedia.org/wiki/Apache_Software_Foundation).

The vast majority of Apache HTTP Server instances run on a [Linux distribution](https://en.wikipedia.org/wiki/Linux_distribution), but current versions also run on [Microsoft Windows](https://en.wikipedia.org/wiki/Microsoft_Windows), [OpenVMS](https://en.wikipedia.org/wiki/OpenVMS),  and a wide variety of [Unix-like](https://en.wikipedia.org/wiki/Unix-like) systems. Past versions also ran on [NetWare](https://en.wikipedia.org/wiki/NetWare), [OS/2](https://en.wikipedia.org/wiki/OS/2), and other operating systems,  including ports to mainframes.

Originally based on the HTTP server, the development of Apache began in early 1995 after work on the NCSA code stalled. Apache played a key role in the initial growth of the [World Wide Web](https://en.wikipedia.org/wiki/World_Wide_Web), quickly overtaking NCSA HTTP as the dominant [HTTP](https://en.wikipedia.org/wiki/HTTP) server. In 2009, it became the first web server software to serve more than 100 million [websites](https://en.wikipedia.org/wiki/Website). As of January 2021, [Netcraft](https://en.wikipedia.org/wiki/Netcraft) estimated that Apache served 24.63% of the million busiest websites, while [Nginx](https://en.wikipedia.org/wiki/Nginx) served 23.21% and Microsoft is in third place at 6.85% (for some of Netcraft's other stats Nginx is ahead of Apache), while according to W3Techs, Apache is ranked first at 35.0% and Nginx second at 33.0% and Cloudflare Server third at 17.3%.

**LANGUAGE SPECIFICATION**

**PHP**

**INTRODUCTION OF PHP**

PHP started as a small open-source project that evolved as more and more people found out how useful it was. Rasmus Lerdorf unleashed the first version of PHP way back in 1994.

* PHP is a recursive acronym for "PHP: Hypertext Preprocessor".
* PHP is a server-side scripting language that is embedded in HTML. It is used to manage dynamic content, databases, session tracking, and even build entire e-commerce sites.
* It is integrated with several popular databases, including MySQL, PostgreSQL, Oracle, Sybase, Informix, and Microsoft SQL Server.
* PHP is pleasingly zippy in its execution, especially when compiled as an Apache module on the Unix side. The MySQL server, once started, executes even very complex queries with huge result sets in record-setting time.
* PHP supports a large number of major protocols such as POP3, IMAP, and LDAP. PHP4 added support for Java and distributed object architectures (COM and CORBA), making n-tier development a possibility for the first time.
* PHP is forgiving: PHP language tries to be as forgiving as possible.
* PHP Syntax is the same as C language.

**What is a PHP File?**

* PHP files can contain text, HTML, CSS, JavaScript, and PHP code.
* PHP code is executed on the server, and the result is returned to the browser as plain HTML.
* PHP files have the extension ".php".

**What Can PHP Do?**

* PHP can generate dynamic page content and it can create, open, read, write, delete, and close files on the server and it can collect form data.
* PHP can send and receive cookies it can add, delete, and modify data in your database and it can be used to control user-access and encrypt data.

**Why PHP?**

* PHP runs on various platforms (Windows, Linux, Unix, Mac OS X, etc.).
* PHP is compatible with almost all servers used today (Apache, IIS, etc.).
* PHP supports a wide range of databases.
* PHP is free.
* PHP is easy to learn and runs efficiently on the server side.

## **What is a Database?**

* A database is a separate application that stores a collection of data. Each database has one or more distinct APIs for creating, accessing, managing, searching, and replicating the data it holds.
* Other kinds of data stores can be used, such as files on the file system or large hash tables in memory but data fetching and writing would not be so fast and easy with those types of systems.
* Nowadays, we use relational database management systems (RDBMS) to store and manage huge volumes of data. This is called a relational database because all the data is stored in different tables and relations are established using primary keys or other keys known as foreign keys.

**MySQL Database**

* MySQL is released under an open-source license. So you have nothing to pay to use it. MySQL is a very powerful program in its own right. It handles a large subset of the functionality of the most expensive and powerful database packages.
* MySQL uses a standard form of the well-known SQL data language. MySQL works on many operating systems and with many languages including PHP, PERL, C, C++, JAVA, etc. MySQL works very quickly and works well even with large data sets.
* MySQL is very friendly to PHP, the most appreciated language for web development. MySQL supports large databases, up to 50 million rows or more in a table.
* The default file size limit for a table is 4GB, but you can increase this (if your operating system can handle it) to a theoretical limit of 8 million terabytes (TB). MySQL is customizable.
* The open-source GPL license allows programmers to modify the MySQL software to fit their specific environments.

**TABLE CREATION**

* Name of the table
* Names of fields
* Definitions for each field
* Field Attribute **NOT NULL** is being used because we do not want this field to be NULL. So if the user tries to create a record with a NULL value, then MySQL will raise an error.
* Field Attribute **AUTO\_INCREMENT** tells MySQL to go ahead and add the next available number to the id field.
* Keyword **PRIMARY KEY** is used to define a column as the primary key. You can use multiple columns separated by a comma to define a primary key.

## **ADMINISTRATIVE MYSQL COMMAND**

* **USE DATABASE NAME**: This will be used to select a particular database in the MySQL work area.
* **SHOW DATABASES:** Lists the databases that are accessible by the MySQL DBMS.
* **SHOW TABLES:** Shows the tables in the database once a database has been selected with the use command.
* **SHOW COLUMNS FROM Table name:** Shows the attributes, types of attributes, key information, whether NULL is permitted, defaults, and other information for a table.
* **SHOW INDEX FROM Table name:** Presents the details of all indexes on the table, including the PRIMARY KEY

## **CREATING TABLES USING PHP SCRIPT:**

To create a new table in any existing database you would need to use PHP function **mysqli\_query()**.

## **Dropping Tables Using PHP Script:**

Drop an existing table in any database, you would need to use the PHP function **mysqli\_query()**.

## **INSERTING DATA USING PHP SCRIPT:**

**CREATE**

Create table statement is used to create a table in MySQL.

**SELECT**

The SELECT statement is used to select data from one or more tables.

**UPDATE**

The UPDATE statement is used to update existing records in a table:

## **DELETE**

The DELETE statement is used to delete records from a table:

**DATABASE DESIGN:**

The data in the system has to be stored and retrieved from the database. Designing the database is part of system design. Data elements and data structures to be stored have been identified at the analysis stage.

They are structured and put together to design the data storage and retrieval system. A database is a collection of interrelated data stored with minimum redundancy to serve many users quickly and efficiently.

The general objective is to make database access easy, quick, inexpensive, and flexible for the user. Relationships are established between the data items and unnecessary data items are removed.

Normalization is done to get an internal consistency of data and to have minimum redundancy and maximum stability. This ensures minimizing data storage required, minimizing chances of data inconsistencies, and optimizing for updates.

**INPUT DESIGN**

The Input design is the main feature of the system. Input design determines the format and validation criteria for data entering the system. Inputs originate with end-users; human factors play a significant role in input design. The input design is designed to control the input, avoid delay, and errors in data, avoid extra steps, to keep the process simple. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps, and keeping the process simple. The input is designed in such a way that it provides security and ease of use while retaining privacy.

The following are the general principles, that are considered in designing inputs,

* + - Enter only variable data
    - Do not input data that can be calculated
    - List of values
    - Sequence entry

**OUTPUT DESIGN**

Designing the output is more important than working up with a few layout charts and reports. The outputs are designed based on the issue encountered. It will also take care of who will receive the output, what for it is produced how many details are needed, when it is needed, and by what method.

The outputs designed in this system are easy to use and useful for their jobs. The outputs are simple to read and interpret. The outputs obtained from this system are designed by using a few guidelines, which are given below. The information should be clear and accurate, yet concise and restricted to relevant data. Reports should have titles, data, and descriptive headings for columns of data, numbered pages, and so on.

**SYSTEM TESTING**

System testing is the process of exercising software with the intent of finding and ultimately correcting errors. This fundamental philosophy does not change for web applications, because Web-based systems and applications reside on a network and interoperate with many different operating systems, browsers, hardware platforms, and communication protocols; the search for errors represents a significant challenge for web applications.

The distributed nature of client/server environments, the performance issues associated with transaction processing, the potential presence of several different hardware platforms, the complexities of network communication, the need to serve multiple clients from a centralized database, and the requirements imposed on the server all combine to make testing of client\server architectures.

Testing issues

* Client GUI considerations
* Target environment and platform diversity considerations
* Distributed database considerations
* Distributed processing considerations

**TYPES OF TESTING**

1. Unit Testing

2. Integration Testing

3. Validation Testing

4. User Acceptance Testing

5. System Testing

**Unit Testing**

All modules were tested and individually as soon as they were completed were checked for their correct functionality. Unit testing is carried out by verifying and recovering errors within the boundary of the smallest unit or a module. In this testing step, each module was found to be working satisfactorily per the expected output of the module. In the package development, each module is tested separately after it has been completed and checked with valid data.

**Integration Testing**

The entire project was split into small programs; each of these single programs gives a frame as an output. These programs were tested individually; at last, all these programs were combined by creating another program where all these constructions were used. It causes a lot of problems by not functioning in an integrated manner.

The user interface testing is important since the user has to declare that the arrangements made in the frames are convenient and it is satisfied. When the frames are tested, the end user gives suggestions. Since they were much exposed to do the work manually.

**Validation Testing**

At the culmination of the black box testing software is completely assembled as a package. Interfacing errors have been uncovered and corrected and a final series of tests i.e., validation succeeds when the software functions in a manner that can be reasonably accepted by the customer.

**User Acceptance Testing**

User acceptance testing of the system is the key factor in the success of any system. The system under consideration is tested for user acceptance by constantly keeping in touch with prospective systems at the time of development and making changes whenever required. This is done concerning the input screen design and output screen design.

**System Testing**

This is to verify that all the system elements have been properly integrated and perform allocated functions. Testing is executing a program to test the logic changes made in it to find errors. Tests are also conducted to find discrepancies between the system and its original objective, current specifications, and documents.

**SYSTEM IMPLEMENTATION**

Implementation is the stage in the project where the theoretical design is turned into a working system. The most crucial stage is achieving a successful new system & and giving the user confidence that the new system will work efficiently & and effectively in the implementation stage.

The stage consists of

* + - Testing the developed program with simple data.
    - Detections and correction of errors.
    - Creating whether the system meets user requirements.
    - Testing whether the system.
    - Making necessary changes as desired by the user.
    - Training user personnel.

**Implementation Procedures**

The implementation phase is less creative than the system design. A system project may be dropped at any time before implementation, although it becomes more difficult when it goes to the design phase.

The final report to the implementation phase includes procedural flowcharts, record layouts, report layouts, and a workable plan for implementing the candidate system design into an operational one. Conversion is one aspect of implementation.

**System Maintenance**

Maintenance is the implementation of the review plan. As important as it is, many programmers and analysts are to perform or identify themselves with the maintenance effort. There are psychological, personality, and professional reasons for this. Analysts and programmers spend far more time maintaining programs than they do writing them. Maintenance accounts for 50-80 percent of total system development.

Maintenance is expensive. One way to reduce maintenance costs is through maintenance management and software modification audits.

* Maintenance is not as rewarding or exciting as developing systems. It is perceived as requiring neither skill nor experience.
* Users are not fully cognizant of the maintenance problem or its high cost.
* Few tools and techniques are available for maintenance.
* A good test plan is lacking.
* Standards, procedures, and guidelines are poorly defined and enforced.
* Programs are often maintained without care for structure and documentation.
* There are minimal standards for maintenance.
* Programmers expect that they will not be in their current commitment by the time their programs go into the maintenance cycle.

**SYSTEM DESIGN**

System design is "the process of studying a procedure or business to identify its goals, purposes and create systems and procedures that will efficiently achieve them". Another view sees system analysis as a problem-solving technique that breaks down a system into its component pieces for the study of how well those parts work and interact to accomplish their purpose.

The field of system analysis relates closely to requirements analysis or operations research. It is also "an explicit formal inquiry carried out to help a decision maker identify a better course of action and make a better decision than they might otherwise have made."

* **DESIGN NOTATION**

Design notations are used when planning and should be able to communicate the purpose of a program without the need for formal code. Commonly used design notations are:

* DFD
* ERD
* **DFD (DATA FLOW DIAGRAM):**

A data flow diagram (DFD) is a graphical representation of the "flow" of data through an [information system](https://en.wikipedia.org/wiki/Information_system), modeling its process aspects. A DFD is often used as a preliminary step to create an overview of the system without going into great detail, which can later be elaborated. DFDs can also be used for the [visualization](https://en.wikipedia.org/wiki/Data_visualization) of [data processing](https://en.wikipedia.org/wiki/Data_processing) (structured design). A DFD shows what kind of information will be input to and output from the system, how the data will advance through the system, and where the data will be stored. It does not show information about the timing of the process or information about whether processes will operate in sequence or parallel, unlike a [flowchart](https://en.wikipedia.org/wiki/Flowchart) which also shows this information.

Data flow diagrams were popularized in the late 1970s, arising from the book Structured Design, by computing pioneers Ed Yourdon and Larry Constantine. They based it on the “data flow graph” computation models by David Martin and Gerald Estrin. The structured design concept took off in the software engineering field, and the DFD method took off with it. It became more popular in business circles, as it was applied to business analysis than in academic circles.

**DFD SYMBOLS**

The process that transforms data flow

Source or Destination of Data

Data Flow

Data source

**ENTITY RELATIONSHIP DIAGRAM**

The relation upon the system is structured through a conceptual ER-Diagram, which not only specifies the existential entities but also the standard relations through which the system exists and the cardinalities that are necessary for the system state to continue. The Entity Relationship Diagram (ERD) depicts the relationship between the data objects. The ERD is the notation that is used to conduct the data modeling activity The attributes of each data object noted in the ERD can be described resign a data object description.

The set of primary components that are identified by the ERD are

* + Data object
  + Relationships
  + Attributes
  + Various types of indicators

The primary purpose of the ERD is to represent data objects and their relationships.

**ER-DIAGRAM SYMBOL**

Entity

Relationship

Flow

**INPUT DESIGN**

The input design is the link between the information system and the user. It comprises the developing specifications and procedures for data preparation and those steps are necessary to put transaction data into a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps, and keeping the process simple. The input is designed in such a way that it provides security and ease of use while retaining privacy.

Input Design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system. It is achieved by creating user-friendly screens for the data entry to handle large volumes of data.

The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data manipulations can be performed. It also provides record viewing facilities. When the data is entered it will check for its validity. Data can be entered with the help of screens.

**DATABASE DESIGN**

The database is designed to manage large bodies of information. The management of data involves both the definitions of structures for the storage of information. In addition, the database system must provide for the safety of the information solved, despite system crashes or attempts at unauthorized access. For developing an efficient database users have to fulfill certain conditions such as controlled redundancy.

* Defining the data
* Inputting the data
* Locating the data
* Accessing the data
* Communicating the data

Revising the data

**Objectives of Database Design**

For designing a database design several objectives have to be met as follows:

* Ease of use
* Control of data integrity
* Control of redundancy
* Control of security
* Data independence (logical & physical)
* Data storage protection
* System performance

**OUTPUT DESIGN**

A quality output is one, which meets the requirements of the end user and presents the information. In any system results of processing are communicated to the users and other systems through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source of information to the user. Efficient and intelligent output design improves the system’s relationship to help user decision-making.

Output design generally refers to the results and information that are generated by the system for many end-users; output is the main reason for developing the system and the basis on which they evaluate the usefulness of the application.

In this Online Repository System project output is to view customer details, employee lists, order tracking details, and attendance percentage results.

**SYSTEM FLOW DIAGRAM**

**DATA FLOW DIAGRAM**

Student

Admin

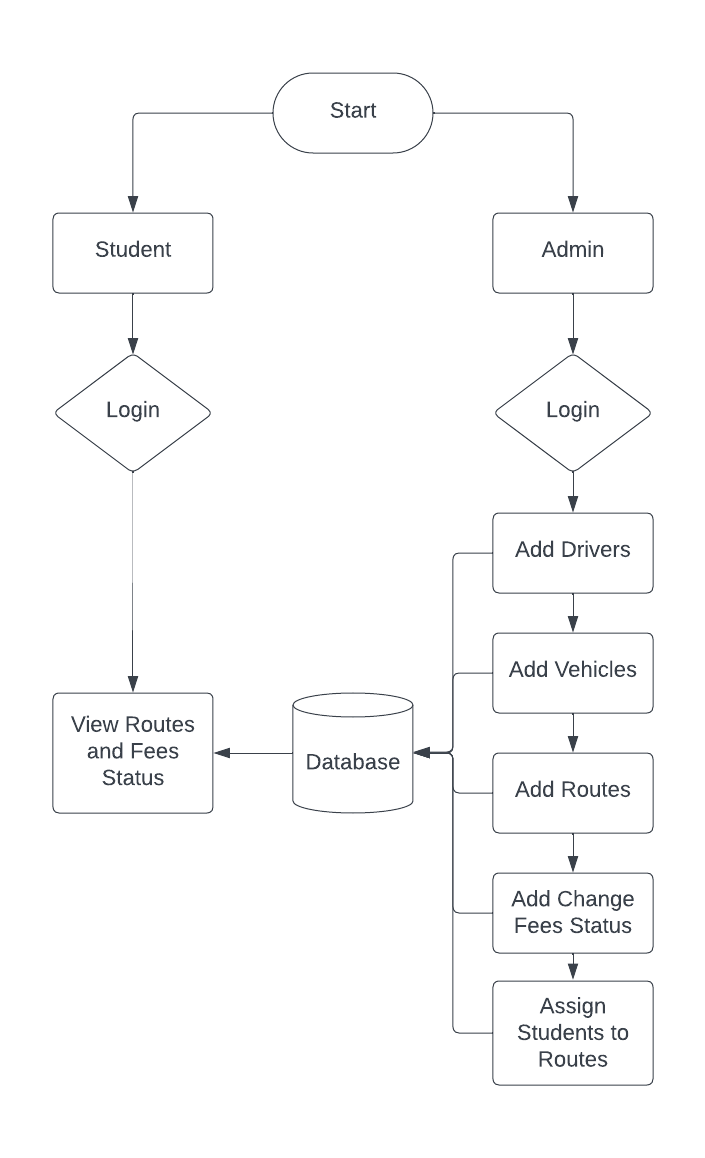
Adds vehicles, drivers, routes and assigns students

Gets info on transport details

Looks and manages vehicles, routes, students and drivers

Enters roll number

**ER DIAGRAM**



**DATABASE DESIGN**

**Table name: admin**

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD NAME** | **DATA TYPE** | **LENGTH** | **DESCRIPTION** |
| id | int | 11 | Primary key |
| username | varchar | 255 | Username |
| email | varchar | 255 | Email ID |
| pswd | varchar | 255 | Password |
| status | int | 11 | Status |

**Table name: route**

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD NAME** | **DATA TYPE** | **LENGTH** | **DESCRIPTION** |
| id | int | 11 | primary key |
| route | varchar | 255 | Route name |
| bus | int | 11 | Bus ID |
| driver | int | 11 | Driver ID |
| status | int | 11 | Status |

**Table name: vehicle**

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD NAME** | **DATA TYPE** | **LENGTH** | **DESCRIPTION** |
| Id | int | 11 | Unique ID |
| name | varchar | 255 | Bus Name |
| category | varchar | 255 | Category |
| seats | int | 11 | Seats |
| plate | varchar | 255 | Number Plate |
| number | int | 11 | Bus Number |
| year | int | 11 | Bought Year |
| owner | varchar | 255 | Ownership |
| service | int | 11 | Last Service |
| status | int | 11 | Status |

**CONCLUSION**

n conclusion, the proposed Transport Management System (TMS) marks a significant advancement over the limitations of the existing transport infrastructure within the college campus. The integration of HTML, CSS, Bootstrap, PHP, and MySQL offers a comprehensive solution aimed at enhancing the efficiency, accessibility, and safety of transportation services for students, faculty, and staff.

By leveraging technology, the TMS addresses several critical aspects of transport management, including real-time tracking, online booking, centralized data management, and robust communication channels. This holistic approach ensures a user-friendly interface for commuters while empowering administrators with tools to streamline operations and make data-driven decisions.

The envisioned system's emphasis on user authentication, personalized profiles, and responsive communication channels fosters a tailored and secure experience for all stakeholders. Moreover, safety measures such as driver monitoring, emergency alerts, and proactive maintenance schedules prioritize the well-being of passengers and the security of transport assets.

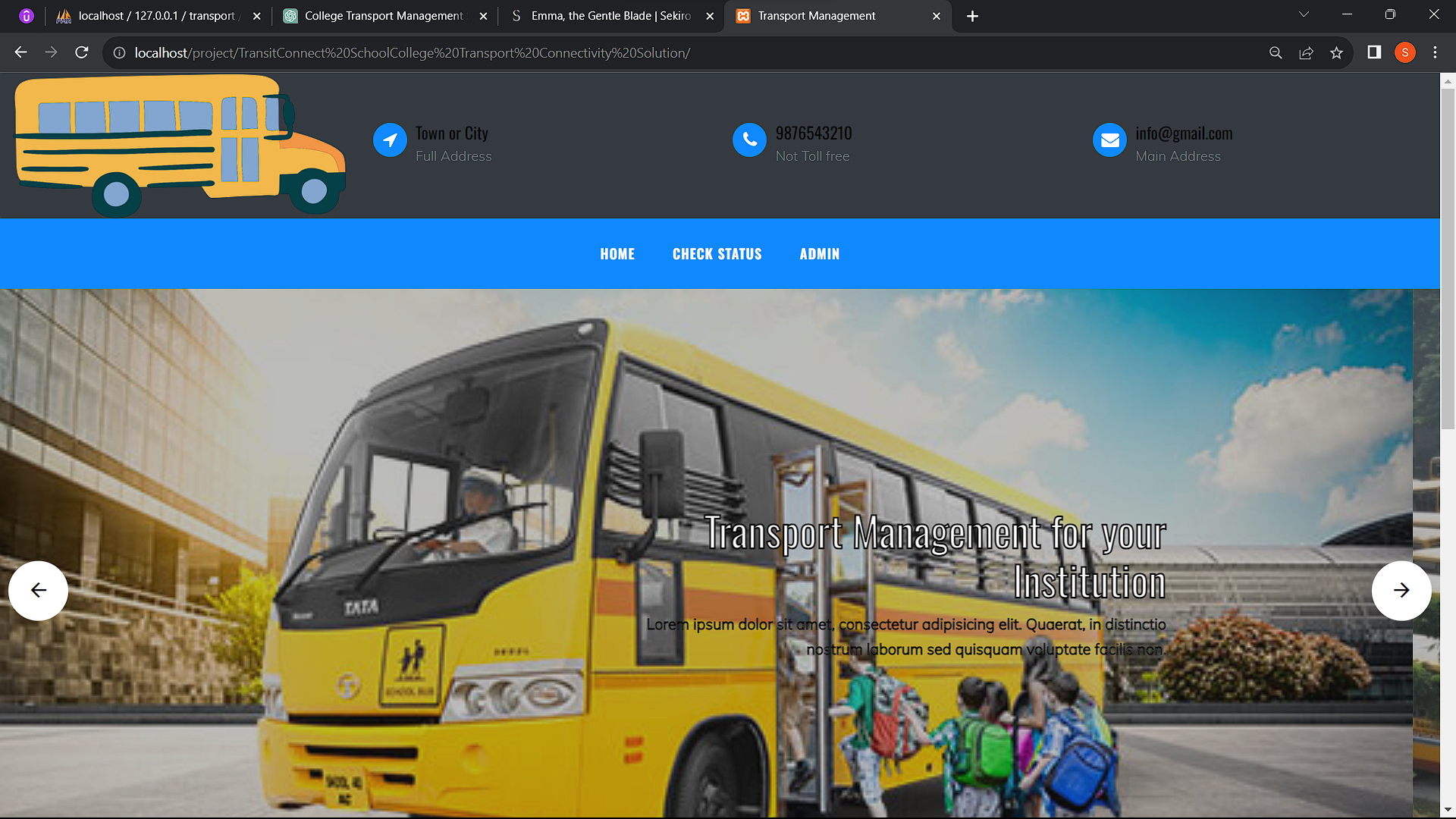
Furthermore, the proposed TMS's integration of data analytics capabilities using MySQL facilitates continuous improvement by analyzing usage patterns, optimizing routes, and generating valuable insights for service enhancements and resource allocation.

In essence, the proposed Transport Management System represents a paradigm shift towards a more efficient, technology-enabled, and user-centric approach to college transportation. It seeks to transform the way transport services are managed, ensuring convenience, safety, and reliability for everyone associated with the college community.

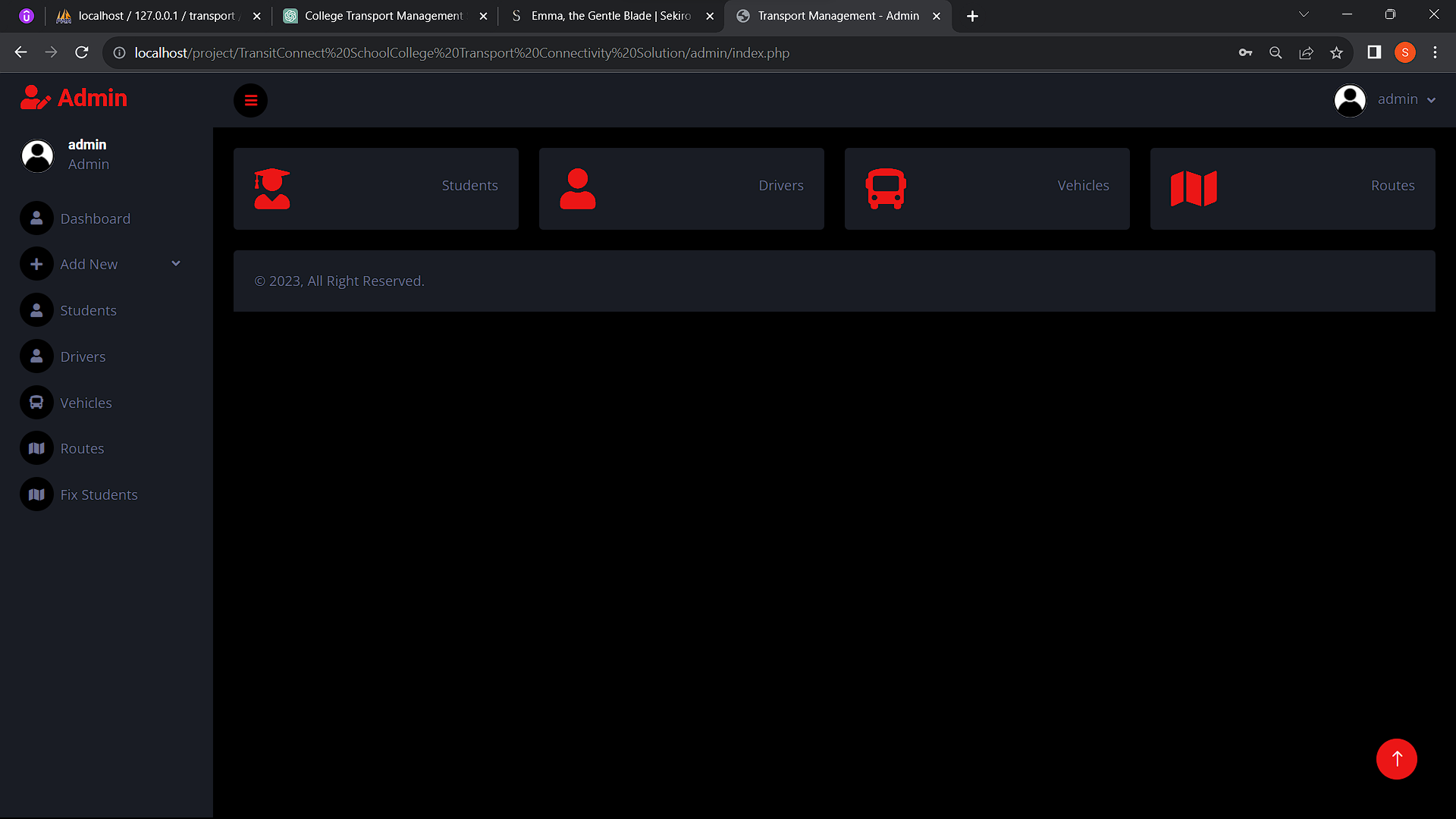
By embracing this innovative system, the college can significantly elevate the transportation experience, fostering a conducive environment for learning, working, and thriving within the campus. Overall, the implementation of this TMS is poised to revolutionize the college's transport landscape, providing a model for efficient transport management systems in educational institutions.

**SCREENSHOTS**

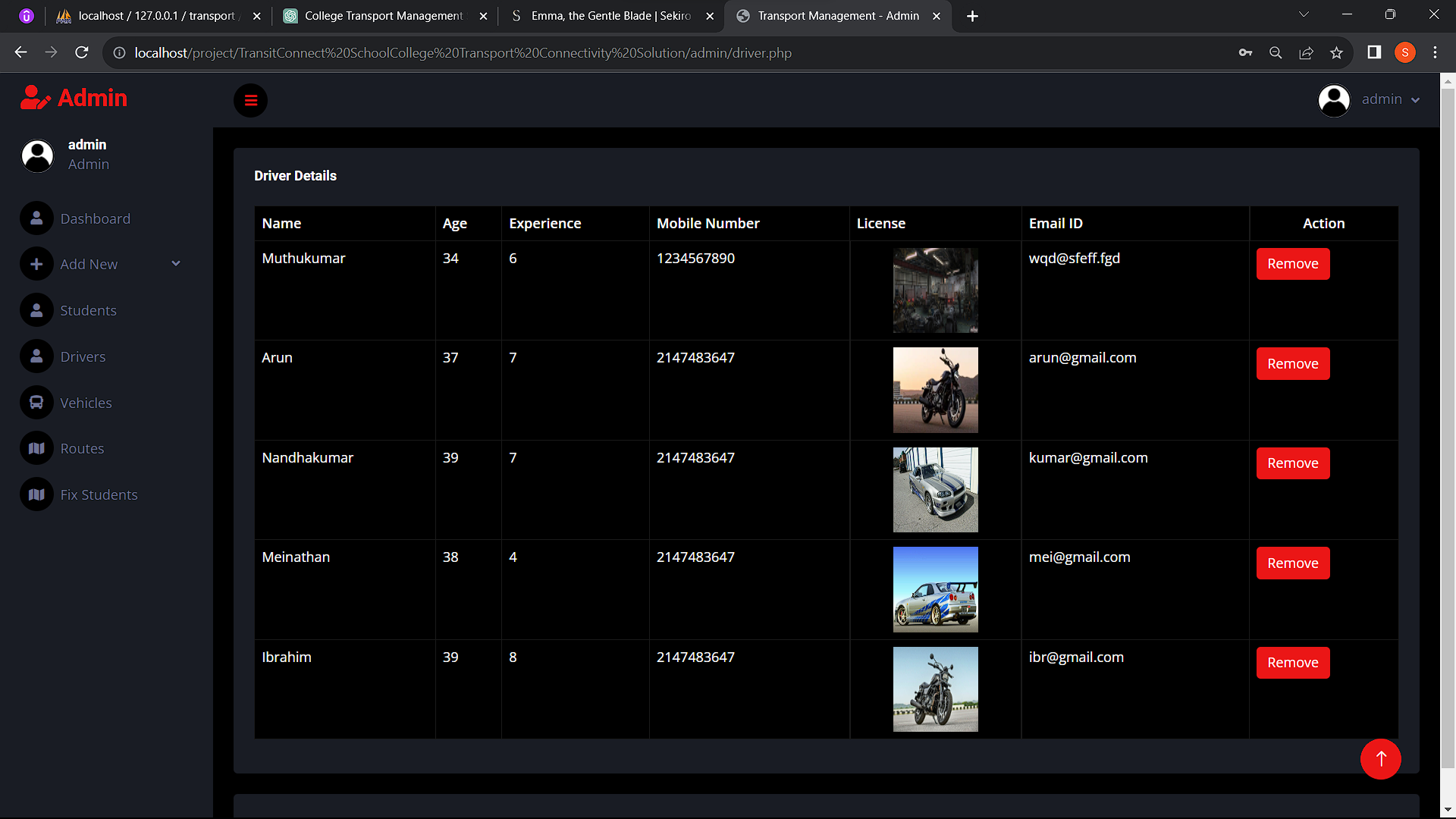
**Main Index page**



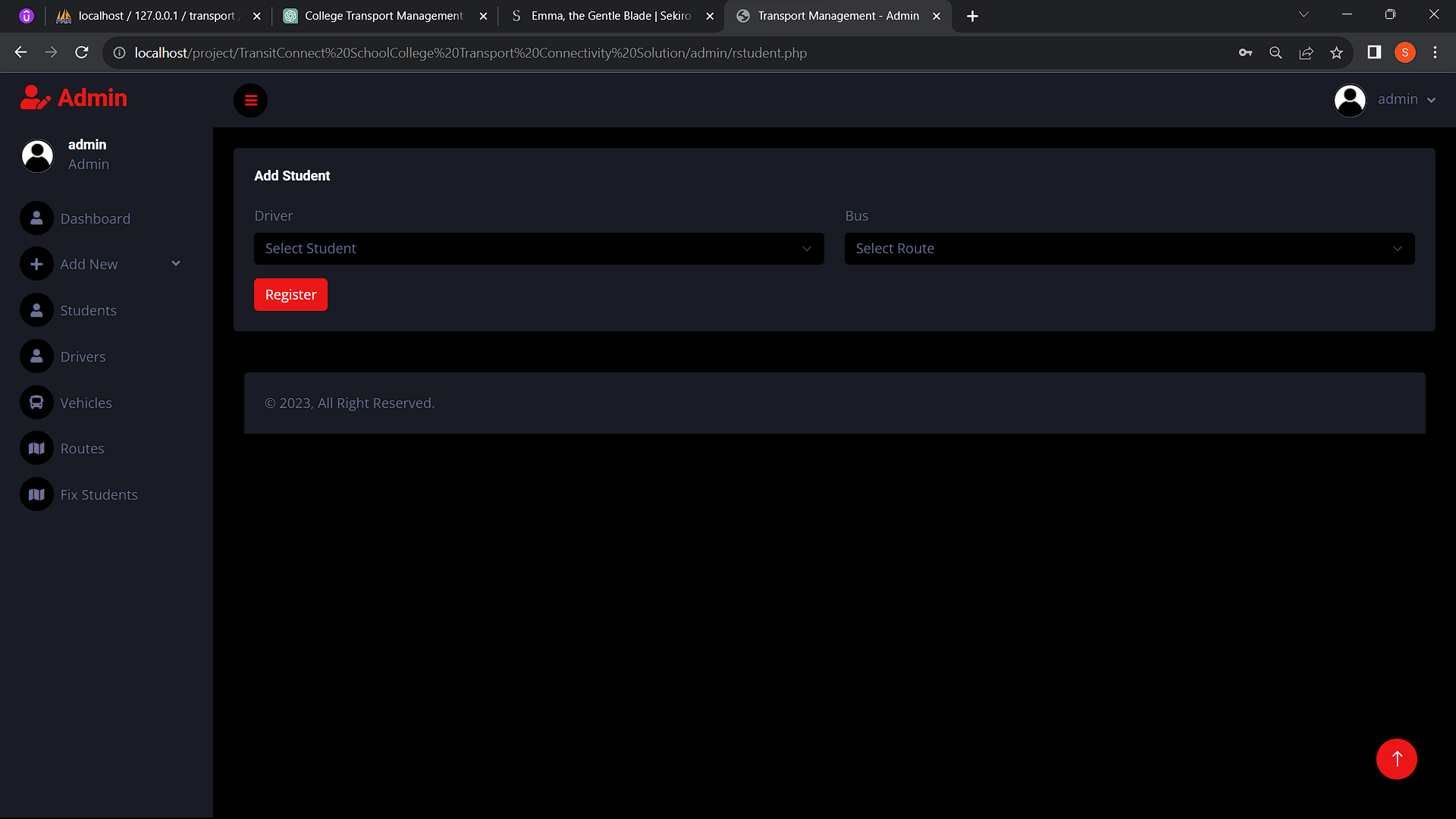
**Admin dashboard**

****

**Driver List**

****

**Route Fix page**

****

**SAMPLE CODE**

**<?php**

**require\_once('config.php');**

**?>**

**<!DOCTYPE html>**

**<html lang="en">**

**<head>**

**<title>Transport Management</title>**

**<meta charset="utf-8">**

**<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">**

**<link href="https://fonts.googleapis.com/css?family=Oswald:400,700|Dancing+Script:400,700|Muli:300,400" rel="stylesheet">**

**<link rel="stylesheet" href="fonts/icomoon/style.css">**

**<link rel="stylesheet" href="css/bootstrap.min.css">**

**<link rel="stylesheet" href="css/jquery-ui.css">**

**<link rel="stylesheet" href="css/owl.carousel.min.css">**

**<link rel="stylesheet" href="css/owl.theme.default.min.css">**

**<link rel="stylesheet" href="css/owl.theme.default.min.css">**

**<link rel="stylesheet" href="css/jquery.fancybox.min.css">**

**<link rel="stylesheet" href="css/bootstrap-datepicker.css">**

**<link rel="stylesheet" href="fonts/flaticon/font/flaticon.css">**

**<link rel="stylesheet" href="css/aos.css">**

**<link href="css/jquery.mb.YTPlayer.min.css" media="all" rel="stylesheet" type="text/css">**

**<link rel="stylesheet" href="css/style.css">**

**</head>**

**<body data-spy="scroll" data-target=".site-navbar-target" data-offset="300">**

**<div class="site-wrap">**

**<div class="site-mobile-menu site-navbar-target">**

**<div class="site-mobile-menu-header">**

**<div class="site-mobile-menu-close mt-3">**

**<span class="icon-close2 js-menu-toggle"></span>**

**</div>**

**</div>**

**<div class="site-mobile-menu-body"></div>**

**</div>**

**<div class="header-top bg-dark">**

**<div class="container-fluid">**

**<div class="row align-items-center">**

**<div class="col-6 col-sm-3">**

**<a href="index.php">**

**<img src="images/logo.png" alt="Image" class="img-fluid">**

**</a>**

**</div>**

**<div class="col-lg-3 d-none d-lg-block">**

**<div class="quick-contact-icons d-flex">**

**<div class="icon align-self-start">**

**<span class="icon-location-arrow text-primary"></span>**

**</div>**

**<div class="text">**

**<span class="h4 d-block">Town or City</span>**

**<span class="caption-text">Full Address</span>**

**</div>**

**</div>**

**</div>**

**<div class="col-lg-3 d-none d-lg-block">**

**<div class="quick-contact-icons d-flex">**

**<div class="icon align-self-start">**

**<span class="icon-phone text-primary"></span>**

**</div>**

**<div class="text">**

**<span class="h4 d-block">9876543210</span>**

**<span class="caption-text">Not Toll free</span>**

**</div>**

**</div>**

**</div>**

**<div class="col-lg-3 d-none d-lg-block">**

**<div class="quick-contact-icons d-flex">**

**<div class="icon align-self-start">**

**<span class="icon-envelope text-primary"></span>**

**</div>**

**<div class="text">**

**<span class="h4 d-block">info@gmail.com</span>**

**<span class="caption-text">Main Address</span>**

**</div>**

**</div>**

**</div>**

**<div class="col-6 d-block d-lg-none text-right">**

**<a href="#" class="d-inline-block d-lg-none site-menu-toggle js-menu-toggle text-black"><span**

**class="icon-menu h3"></span></a>**

**</div>**

**</div>**

**</div>**

**<div class="site-navbar py-2 js-sticky-header site-navbar-target d-none pl-0 d-lg-block" role="banner">**

**<div class="container">**

**<div class="d-flex align-items-center">**

**<div class="mx-auto">**

**<nav class="site-navigation position-relative text-right" role="navigation">**

**<ul class="site-menu main-menu js-clone-nav mr-auto d-none pl-0 d-lg-block">**

**<li>**

**<a href="index.php" class="nav-link text-left">Home</a>**

**</li>**

**<li>**

**<a href="contact.php" class="nav-link text-left">Check Status</a>**

**</li>**

**<li>**

**<a href="admin/signin.php" class="nav-link text-left">Admin</a>**

**</li>**

**</ul>                                                                                                                                                                                                                                                                                          </ul>**

**</nav>**

**</div>**

**</div>**

**</div>**

**</div>**

**</div>**

**REFFERENCES**

**BIBLIOGRAPHY**

**Comprehensive Web Development Textbook References**

**General Web Development:**

**• MDN Web Docs: (https://developer.mozilla.org/) - The authoritative source from Mozilla, offering in-depth documentation, tutorials, and references for various web technologies.**

**• W3Schools: (https://www.w3schools.com/) - A well-established website with interactive tutorials, references, and examples for a wide range of web development topics.**

**• The Odin Project: (https://theodinproject.com/) - A free, full-stack web development curriculum with a strong focus on practical projects.**

**• FreeCodeCamp: (https://www.freecodecamp.org/) - Another free coding platform with interactive lessons, projects, and a supportive community for web development learning.**

**Specific Technologies:**

**• HTML:**

**o HTML Dog: (https://www.htmldog.com/) - Offers interactive tutorials, challenges, and references for learning HTML.**

**• CSS:**

**o CSS-Tricks: (https://css-tricks.com/) - A popular website with articles, tips, tricks, and best practices for mastering CSS.**

**• JavaScript:**

**o JavaScript30: (https://javascript30.com/) - Provides 30 JavaScript coding challenges for practicing and improving your JavaScript skills.**

**o Eloquent JavaScript Website: (https://eloquentjavascript.net/) - Interactive tutorials and resources aligned with the book "Eloquent JavaScript" by Marijn Haverbeke.**

**• Bootstrap:**

**o Get Bootstrap: (https://getbootstrap.com/) - The official Bootstrap website with comprehensive documentation, examples, and tutorials for learning and using the framework.**

**o Start Bootstrap: (https://startbootstrap.com/) - Offers free Bootstrap templates to use as a starting point for your projects.**

**• PHP:**

**o PHP.net: (https://www.php.net/manual/en/index.php) - The official PHP website with comprehensive documentation, tutorials, and a reference manual.**

**o Laracasts: (https://laracasts.com/) - Features video tutorials and screencasts for learning PHP, Laravel (a popular PHP framework), and other web development topics.**

**• MySQL:**

**o MySQL Documentation: (https://dev.mysql.com/doc/) - Official documentation from MySQL, providing detailed information on using the database management system.**

**o SQLBolt: (https://sqlbolt.com/) - An interactive platform where you can practice writing and running SQL queries, the language used with MySQL.**

**Community Resources:**

**• Stack Overflow: (https://stackoverflow.com/) - A question-and-answer website for programmers, where you can search for solutions to your coding problems or ask questions related to web development.**

**• GitHub: (https://github.com/) - A version control system for code hosting and collaboration. GitHub also offers a wealth of open-source web development projects that you can explore and learn from.**

**Comprehensive Web Development Online References**

**General Web Development:**

**• MDN Web Docs: (https://developer.mozilla.org/) - The authoritative source from Mozilla, offering in-depth documentation, tutorials, and references for various web technologies.**

**• W3Schools: (https://www.w3schools.com/) - A well-established website with interactive tutorials, references, and examples for a wide range of web development topics.**

**• The Odin Project: (https://theodinproject.com/) - A free, full-stack web development curriculum with a strong focus on practical projects.**

**• FreeCodeCamp: (https://www.freecodecamp.org/) - Another free coding platform with interactive lessons, projects, and a supportive community for web development learning.**

**Specific Technologies:**

**• HTML:**

**o HTML Dog: (https://www.htmldog.com/) - Offers interactive tutorials, challenges, and references for learning HTML.**

**• CSS:**

**o CSS-Tricks: (https://css-tricks.com/) - A popular website with articles, tips, tricks, and best practices for mastering CSS.**

**• JavaScript:**

**o JavaScript30: (https://javascript30.com/) - Provides 30 JavaScript coding challenges for practicing and improving your JavaScript skills.**

**o Eloquent JavaScript Website: (https://eloquentjavascript.net/) - Interactive tutorials and resources aligned with the book "Eloquent JavaScript" by Marijn Haverbeke.**

**• Bootstrap:**

**o Get Bootstrap: (https://getbootstrap.com/) - The official Bootstrap website with comprehensive documentation, examples, and tutorials for learning and using the framework.**

**o Start Bootstrap: (https://startbootstrap.com/) - Offers free Bootstrap templates to use as a starting point for your projects.**

**• PHP:**

**o PHP.net: (https://www.php.net/manual/en/index.php) - The official PHP website with comprehensive documentation, tutorials, and a reference manual.**

**o Laracasts: (https://laracasts.com/) - Features video tutorials and screencasts for learning PHP, Laravel (a popular PHP framework), and other web development topics.**

**• MySQL:**

**o MySQL Documentation: (https://dev.mysql.com/doc/) - Official documentation from MySQL, providing detailed information on using the database management system.**

**o SQLBolt: (https://sqlbolt.com/) - An interactive platform where you can practice writing and running SQL queries, the language used with MySQL.**

**Community Resources:**

**• Stack Overflow: (https://stackoverflow.com/) - A question-and-answer website for programmers, where you can search for solutions to your coding problems or ask questions related to web development.**

**• GitHub: (https://github.com/) - A version control system for code hosting and collaboration. GitHub also offers a wealth of open-source web development projects that you can explore and learn from.**