**CHAPTER 1**

**INTRODUCTION**

This chapter gives us the complete information about introduction, problem statement,

proposed system, motivation and objectives of our project

**1.1 INTRODUCTION**

College Bus information system is software which will inform about the transport facility in college and Institute. Bus information will provide the Information about the student. The main objective of the system is to reduce the consumption of time during maintaining the records of college Transport management. The advantage of the system is that we can avoid manual paper work which was done previously for transport management.

The application also provides an interface for the admin to login into the application and add or update the student details. It also shows the type of bus pass and the valid duration of the pass. The system will be used by the administrator where he/she will be provided with a unique login id and password. The administrator will be allowed to update the details related to the particular student, the admin can also update the payments done by the student.

1.2 **PROBLEM STATEMENT**

For a college it would be difficult to keep record’s manually of the students who are travelling by the collage bus. College bus information application makes it easier for the college staff to keep the records without any difficulties, and it also allows the college staff to know details about the payment related to the bus which is used by the student according to his/her route. The application also allows the staff to add or remove a particular student.

**1.3 MOTIVATION AND OBJECTIVES OF THE PROJECT**

The main objective of this system is to reduce the consumption of time during maintaining the records of college bus information. Separate divisions are provide to maintain the records of Student, faculty, Roots, diversion, etc.

In other words the objective of the present college Transport management software are:

* Simple database is maintained.
* .Easy operation for the operator of the System.
* Easy maintaining the records.

**1.4 PROPOSED SOLUTION & ADVANTAGES**

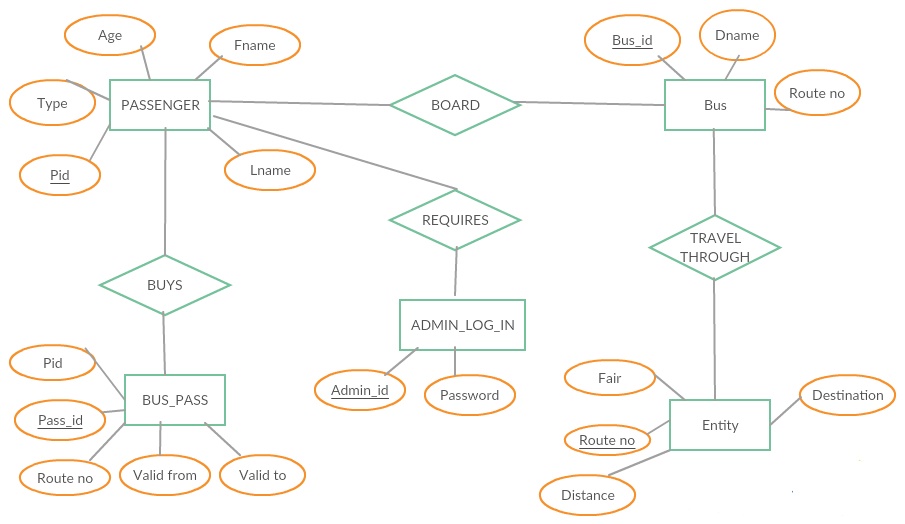
In our proposed system we have the provision for adding the details of the students. Another advantage of the system is that it is very easy to edit the details of the student and delete a student when it found unnecessary. Here is facility of find root, direction, individual profile facility is also provided.

We can attain the following facilities:

* Easy to handle and feasible.
* Easy to operate.
* Convenient.

**2.2 ER DIAGRAM**

An entity–relationship model (ER model for short) describes interrelated things of interest in a specific domain of knowledge. A basic ER model is composed of entity types (which classify the things of interest) and specifies relationships that can exist between instances of those entity types



**Fig 2.2.1 ER Diagram**

**3.2 SQL COMMANDS AND QUERIES**

**3.2.1 COMMANDS**

CREATE TABLE `admin\_log\_in` (

`AdminId` varchar(50) NOT NULL,

`Password` varchar(50) NOT NULL

);

CREATE TABLE `bus` (

`Busid` varchar(50) NOT NULL,

`Dname` varchar(50) NOT NULL,

`Routeno` int(11) NOT NULL,

`Passid` bigint(20) NOT NULL

);

CREATE TABLE `buspass` (

`Passid` bigint(50) NOT NULL,

`Pid` int(11) NOT NULL,

`Validfrom` date NOT NULL,

`Validto` date NOT NULL,

`Routeno` int(11) NOT NULL

);

CREATE TABLE `passenger` (

`Type` varchar(5) NOT NULL,

`Pid` int(11) NOT NULL,

`Fname` varchar(50) NOT NULL,

`Lname` varchar(50) NOT NULL,

`Age` int(11) NOT NULL,

`Gender` char(1) NOT NULL,

`Bloodtag` varchar(5) NOT NULL

);

CREATE TABLE `route` (

`Destination` varchar(50) NOT NULL,

`Distance` int(11) NOT NULL,

`Routeno` int(11) NOT NULL,

`Fair` int(11) NOT NULL

);

CREATE TABLE ` contactus` (

`Fname` varchar(50) NOT NULL,

`Lname` varchar(50) NOT NULL,

`Mailid` varchar(15)NOT NULL,

`Phone` int(11) NOT NULL

`Feedback` varchar(100)NOT NULL,

);

**3.2.1 QUERIES**

SELECT \*FROM passenger;

SELECT \* FROM bus;

SELECT \* FROM contactus;

CREATE TRIGGER `pass\_insert` AFTER INSERT ON `passenger` FOR EACH ROW BEGIN

INSERT INTO users\_log

VALUES(NEW.Pid, NOW());

DELIMITER $$

CREATE DEFINER=`root`@`localhost` PROCEDURE `myproc` (IN `Fn` VARCHAR(50), IN `Ln` VARCHAR(50))  NO SQL

BEGIN

(SELECT Passid from buspass where Pid in (SELECT Pid from passenger where Fname=Fn and Lname=Ln))union(

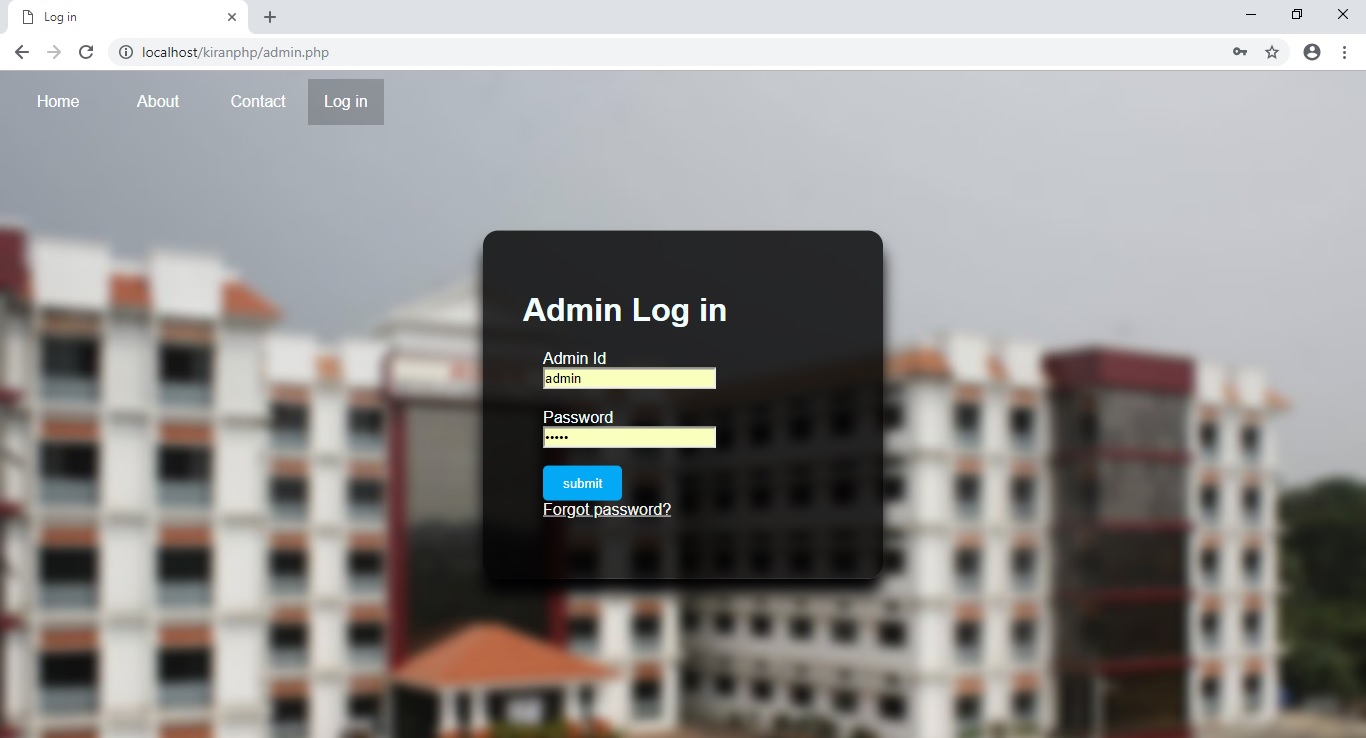
SELECT Pid FROM passenger WHERE Fname=Fn and Lname=Ln);

END$$

DELIMITER ;

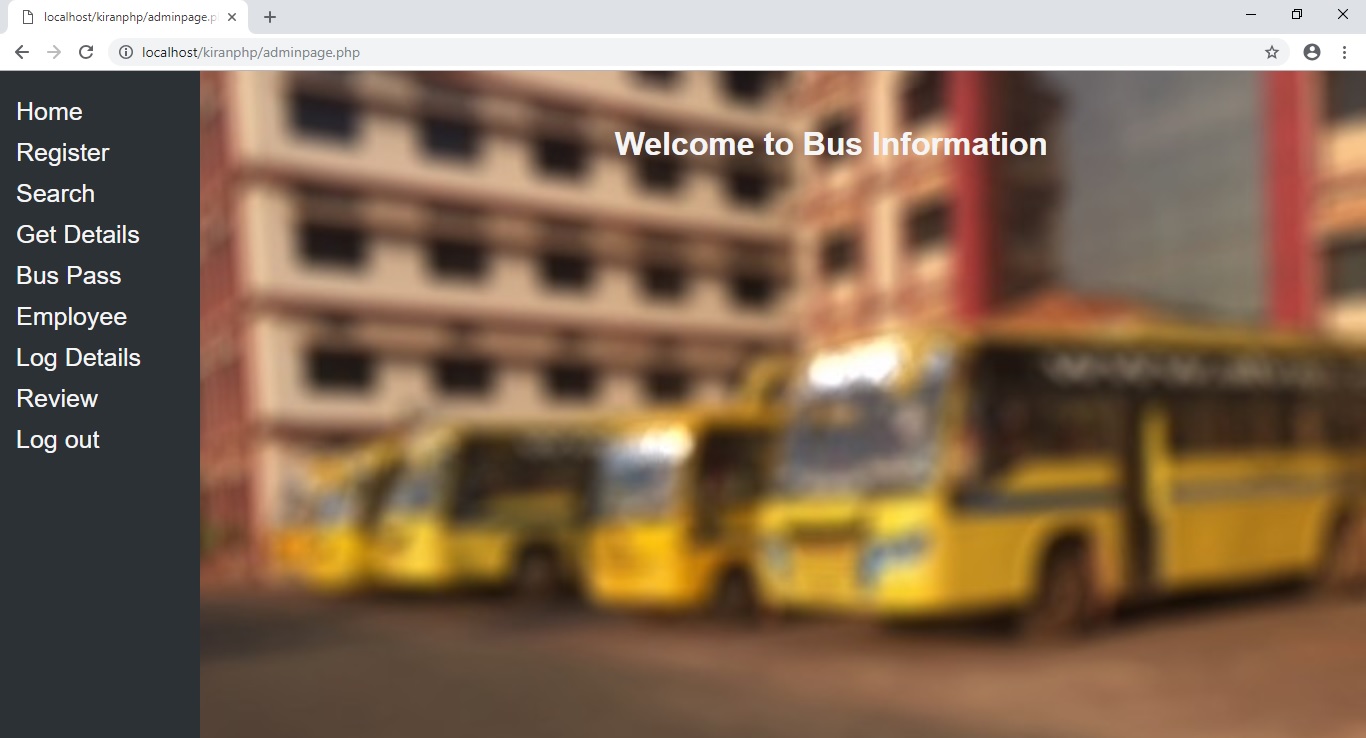
**3.4 OUTPUT TESTING**

* While executing phpmysqli connection code we were not able to make the connection of backend mysqli to front end php. So to solve this problem we had to create a new mysqli user with password. After this the connection was successful.
* The connection was successful but the data entered in front end was not storing in backend, since all the attributes data types in backend were not set to varchar. So we modified the php code and mysqli query accordingly.
* If we enter wrong password, local host says wrong password. If XAMP server is not started, then we cannot run local host

****

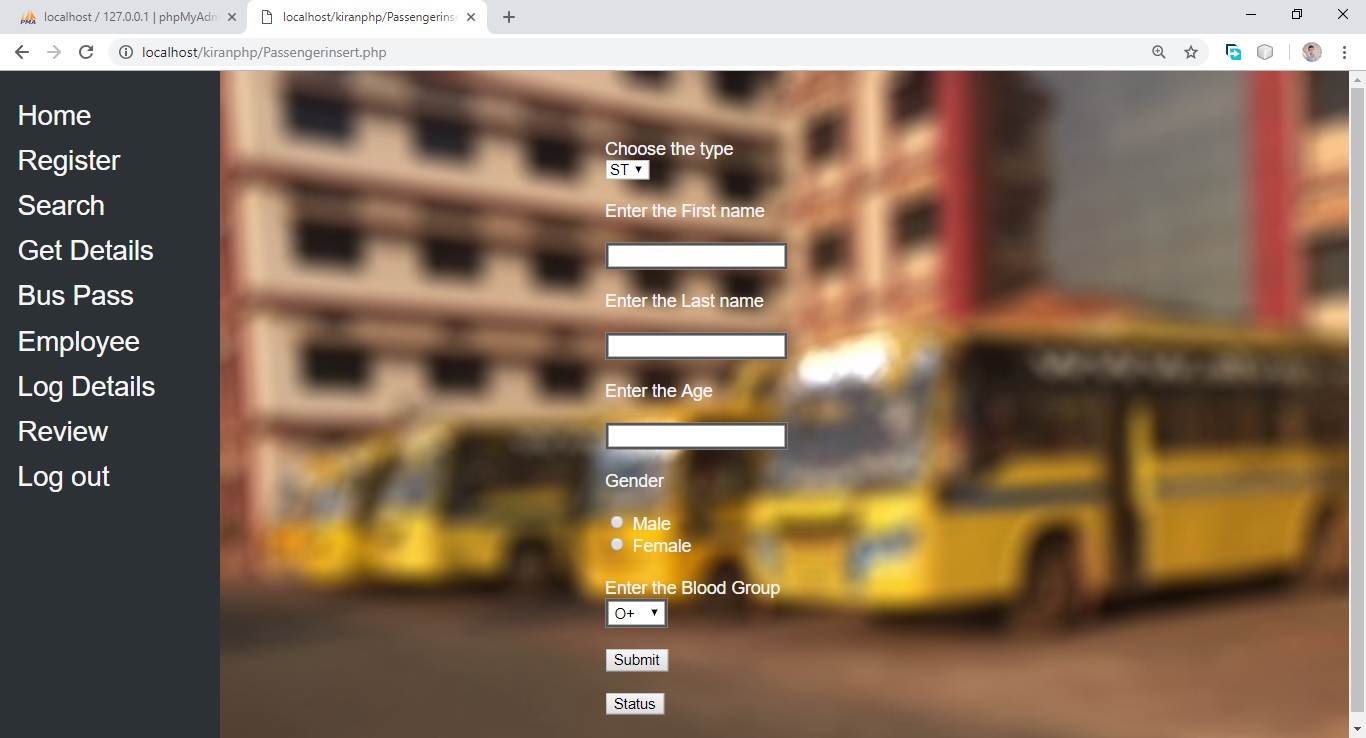
**Fig 4.1.2: Admin Login**

Here admin can logs in with Admin Id and password to add student details regarding bus.

****

**Fig 4.1.3: Admin Home Page**

This is the home page which user encounters after logging in.

****

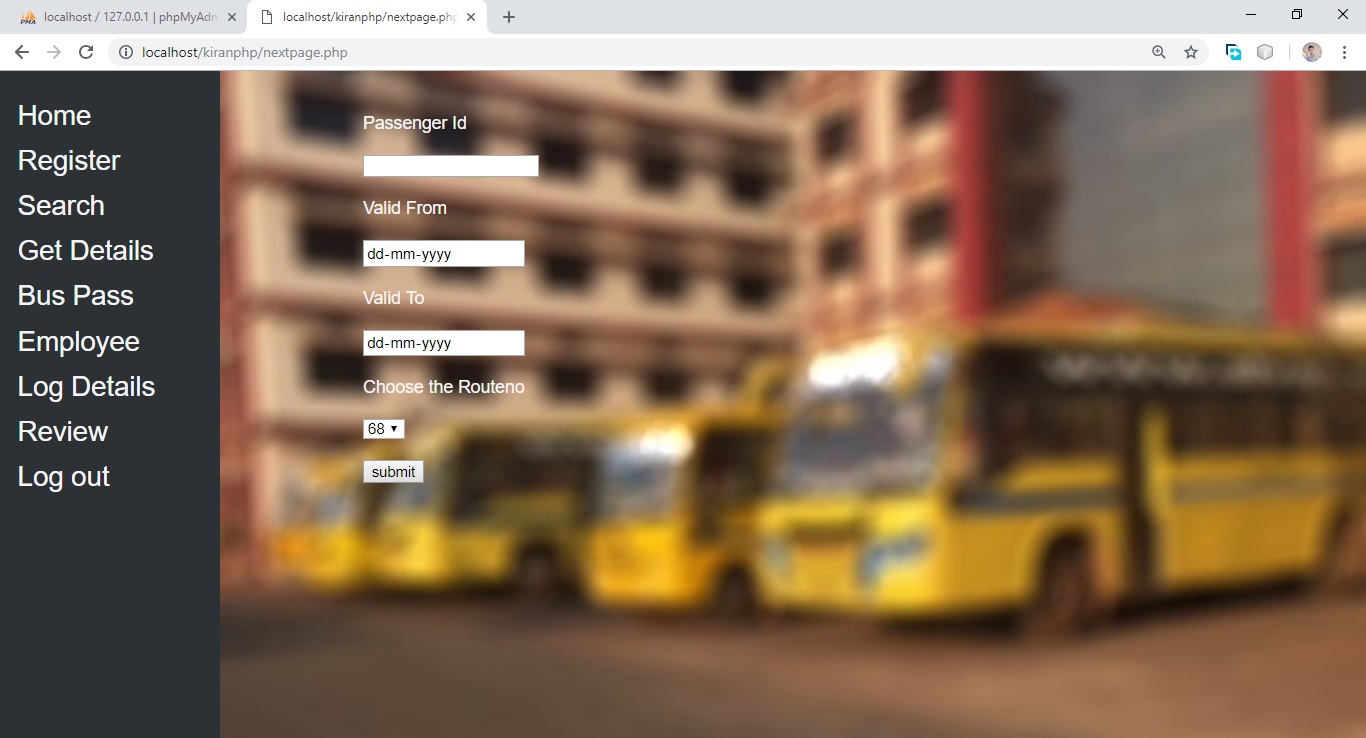
**Fig 4.1.4: Student Registration**

In this page the user enters the details of the student for registration.

****

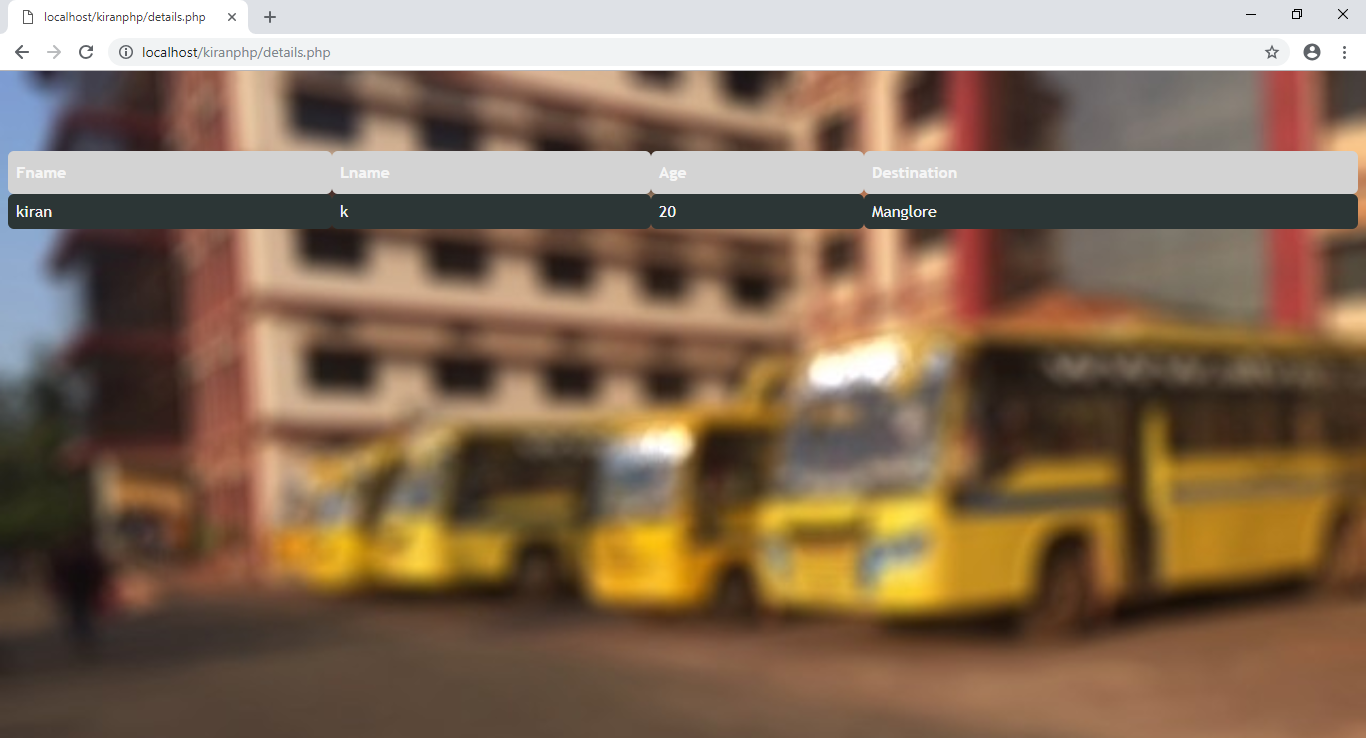
**Fig 4.1.5: Current Status of Passenger table**

This page shows the current status of the passenger table.

****

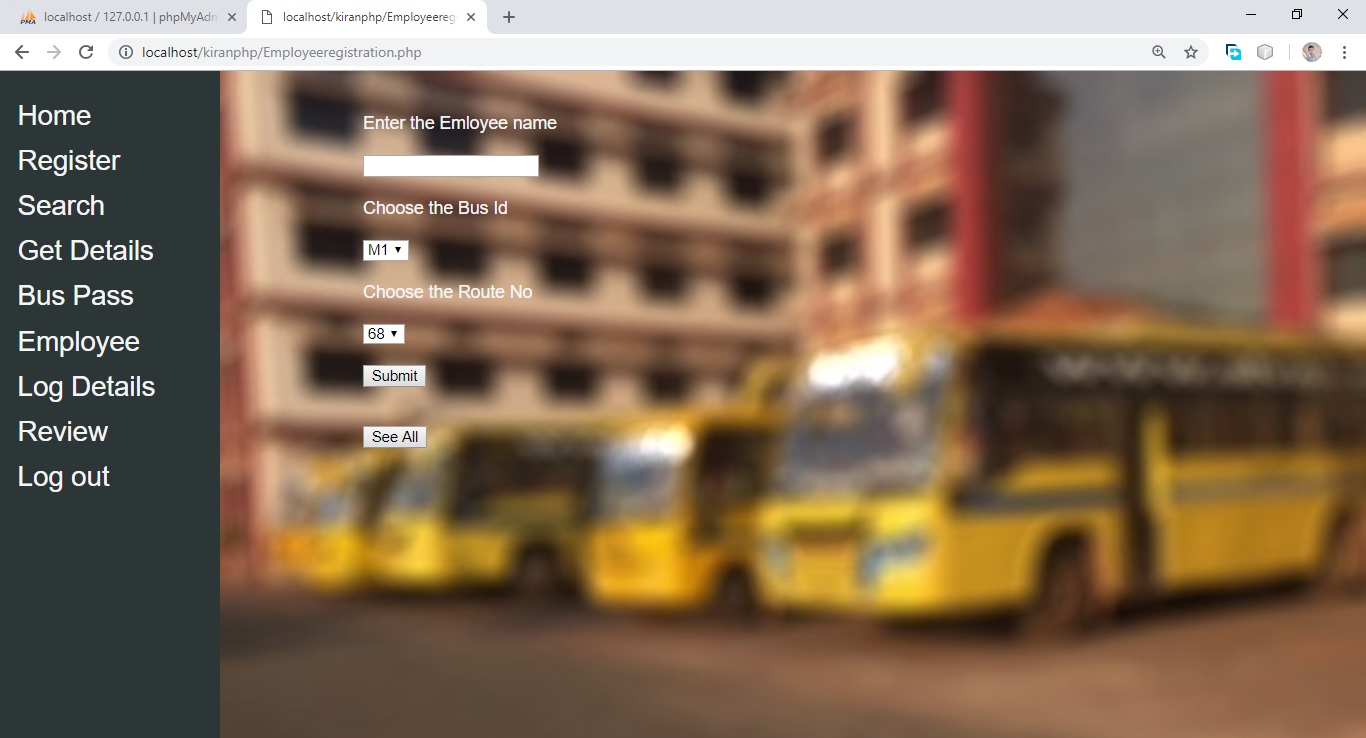
**Fig 4.1.6: Bus Pass**

This page generates the Bus pass and the user enters validity details of the buspass.



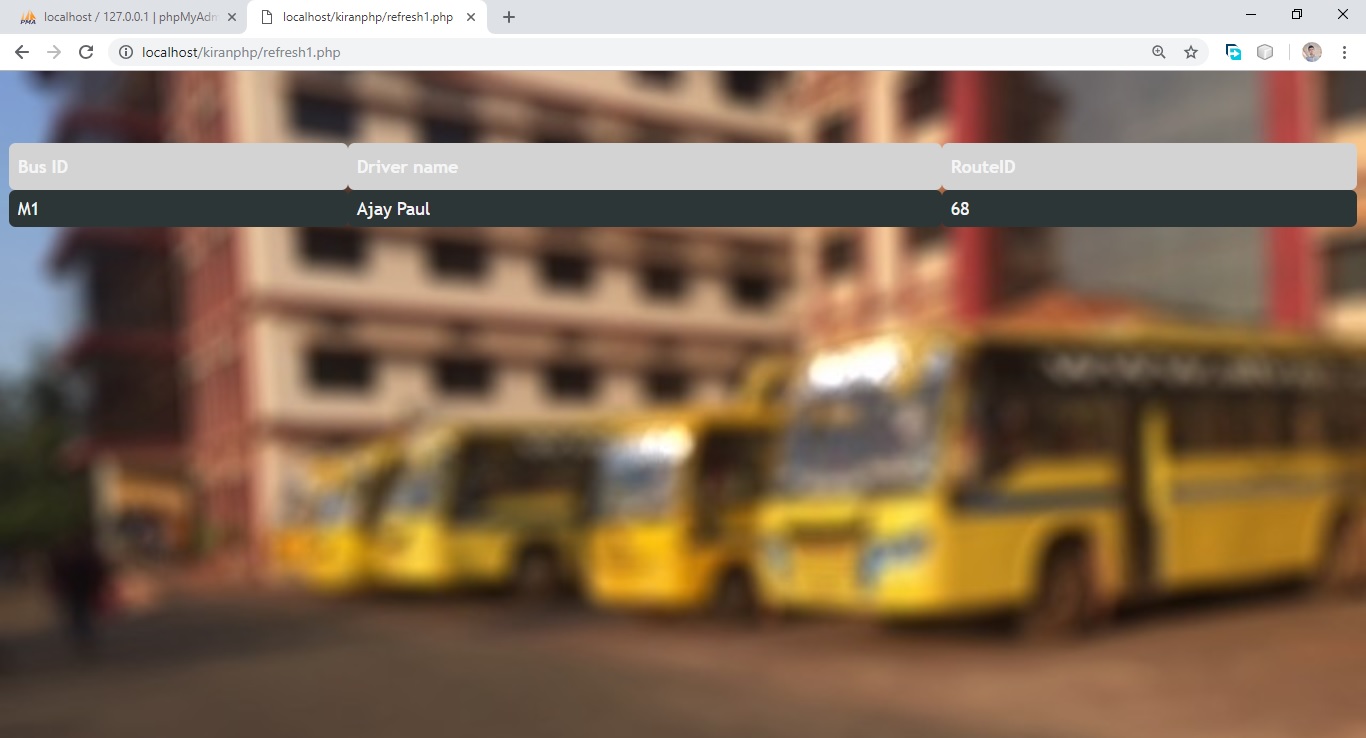
**Fig 4.1.7: Destination details of student**

This shows the details of the student after the get details button is clicked.

****

**Fig 4.1.8: Employee Registration**

Here the admin will enter the Employee details of the particular bus.

****

**Fig 4.1.9: Employee details**

This page shows the current Employee details of the bus

****

**Fig 4.1.10: Register details using trigger**

This page shows the registration time details which is done by trigger.

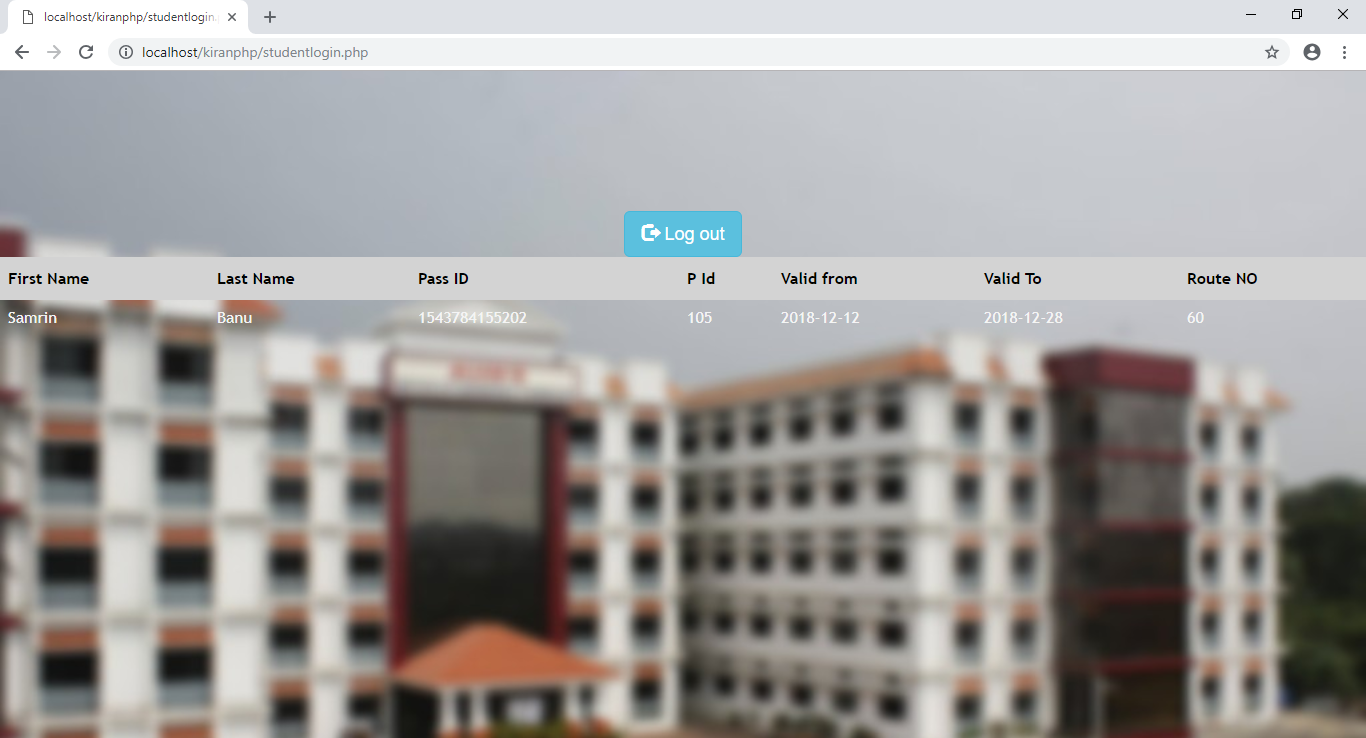
****

**Fig4.1.11: Feedback given by users**

In this page the admin can view the Feedback given by the user in the Contac us page.

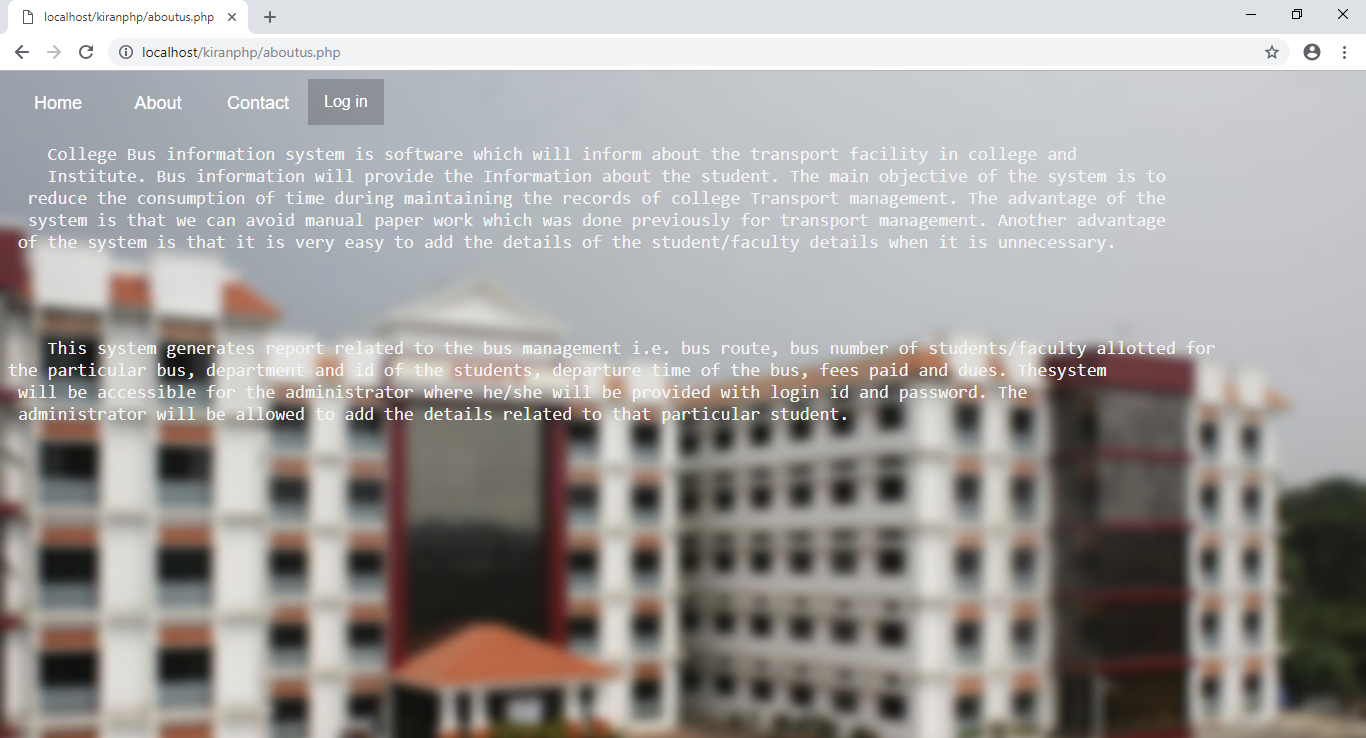
**Fig 4.1.12: Student Log in**

This is the page where the student will enter his/her passed to view the current details of that particular student.



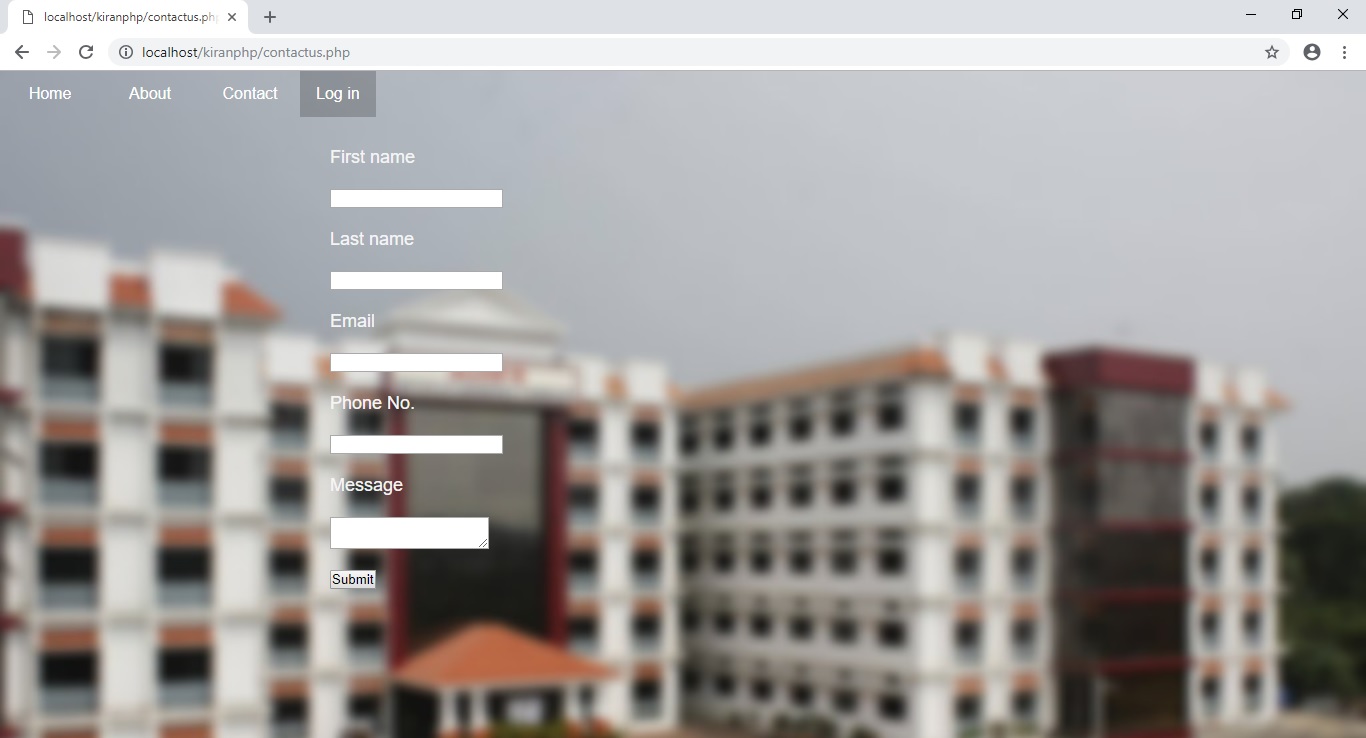
**Fig 4.1.13: Details after student Log in**

This page show the details of the student after logging in.



**Fig 4.1.14: About us page**

This is the about us page which describes the application.

****

**Fig 4.1.15: Feedback page**

The feedback page will allow the user to give the feedback of the application.

**REFERENCES**

* W3 Schools-https://www.w3schools.com/php/default.asp
* Stack Overflow-https://stackoverflow.com/
* Quara- https://www.quora.com/topic/Web-Development