BASELIOS POULOSE II CATHOLICOS COLLEGE

BASELIOS MOUNT, PIRAVOM

Re-accredited with 'A' Grade by

NAAC (Affiliated to Mahatma Gandhi University)

DEPARTMENT OF COMPUTER APPLICATIONS

2022-23

Project

Report on

BUS ENQUIRY SYSTEM

BASELIOS POULOSE II CATHOLICOS COLLEGE

BASELIOS MOUNT, PIRAVOM

Re-accredited with 'A' Grade by

PIRAVOM

2022-23

DEPARTMENT OF COMPUTER APPLICATIONS

Project Report

On

BUS ENQUIRY SYSTEM

Submitted in partial fulfilment of the requirements for the award of the degree of

BACHELOR OF COMPUTER APPLICATION

Guided by: Mrs. Leeja Mathew
(Dept. of Computer application)

Submitted by:
Sanjay Benoy

BASELIOS POULOSE II CATHOLICOS COLLEGE
BASELIOS MOUNT, PIRAVOM

Re-accredited with 'A' Grade by

PIRAVOM

2022-23

DEPARTMENT OF COMPUTER APPLICATIONS

Certificate

This is to certify that the project entitled "BUS ENQUIRY SYSTEM" submitted in partial fulfillment for the award of the degree of BACHELOR OF COMPUTER APPLICATION is a bonafide report of the project done by Sanjay Benoy (Reg no:200021096676) during the year 2022-23.

Internal Guide: **Head of the department:**

Mrs. Leeja Mathew **Dr. Kurian MJExaminer: 1**

College Seal **Department Seal**

DECLARATION

I hereby declare that the this project work entitled "BUS ENQUIRY SYSTEM " is a record of original work done by me under the guidance of **Mrs.Leeja Mathew**, *Assistant Professor, Department of Computer Applications and the work has not formed the basis for the awardof any degree or diploma or similar title to any candidate of any university subject.*

Internal Guide: **Signature of Student**

Mrs. Leeja Mathew

ACKNOWLEDGEMENT

At the outset, I thank God Almighty for making endeavor a success.

I express my gratitude to **Dr. Tiji Zacharia**, Principal, Baselios Poulose II Catholicos College, for providing me with adequate facilities, ways and means by which I was able to complete the project work. I express my sincere thanks to my internal guide **Mrs. Leeja Mathew**, who guide me properly from the beginning to the end of my project and examining the draft of this project ,suggestions and modifications .With immense pleasure I take this opportunity to record out sincere thanks to Head ofthe Department **Dr. Kurian M J**, Associate Professor, Department of Computer Applications for his motivation throughout this project .

Last but not the least, I also express my gratitude to all other members of the faculty and well wishers who assisted me in various occasions during the project work.

:- Sanjay Benoy

ABSTRACT

The Web application entitled “Bus Enquiry System” is an online platform for providing information about buses including their time of arrivals in bus stops. The system is specially designed to provide information about bus times to people without making them stand under stops for hours. The system consists of three users : Admin, Busowner User. The admin has the privilege control on the system. The admin can verify bus owners and accept or reject busowners. The admin can also verify buses accept or reject bus requests. Customers are general users who can simply register with basic details and login to search for various buses times. Bus owners are added by the admin. They could view their profiles ,add buses.

TABLE OF CONTENTS

ACKNOWLEGMENT

ABSTRACT

LIST OF TABLES

LIST OF FIGURES

1. INTRODUCTION **15**

1.1. BACKGROUND AND MOTIVATION 15

1.2. THE PROPOSED SYSTEM 15

1.3. PROJECT SCOPE 16

1.3.1 LIMITATIONS OF EXISTING SYSTEM 16

1.3.2 ADVANTAGES OF PROPOSED SYSTEM 17

2. SYSTEM ANALYSIS **19**

2.1. INTRODUCTION 19

2.2. STAKEHOLDERS OF THIS PROJECT 20

2.2.1. ADMINISTRATOR 20

2.2.2. BUS OWNER 20

2.2.3. USER 21

2.3. SOFTWARE REQUIREMENT SPECIFICATIONS 21

2.3.1. ADMIN 21

2.3.2. BUS OWNER 21

2.3.3. USER 21

2.4. FEASIBILITY STUDY 22

2.4.1 TECHNICAL FESIBILITY 23

2.4.2 OPERATIONAL FESIBILITY 23

2.4.3 ECONOMIC FESIBILITY 24

2.5. SOFTWARE DEVELOPMENT LIFECYCLE MODEL 25

2.6. HARDWARE AND SOFTWARE REQUIREMENTS 27

2.6.1. SOFTWARE SPECIFICATIONS 26
2.6.1.1. HTML 27
2.6.1.2. CSS 28
2.6.1.3. AJAX 28
2.6.1.4. JQUERY 29
2.6.1.5. BOOTSTRAP 29
2.6.1.6. JAVASCRIPT 29
2.6.1.7. PHP 29
2.6.1.8. WAMP SERVER 30
2.6.1.9. MYSQL 31
2.6.1.10. WINDOWS 10 33
2.6.1.11. MICROSOFT WORD 34
2.6.1.12. SMART DRAW 34
2.6.2. HARDWARE REQUIREMENTS 36
3. SYSTEM DESIGN 37
3.1. SYSTEM ARCHITECTURE 37
3.2. MODULE DESIGN 37
3.3. DATABASE DESIGN 38
3.3.1. NORMALIZATION 39
3.3.2. TABLE STRUCTURE 41
3.3.3. DATA FLOW DIAGRAM 47
3.3.3.1. INTRODUCTION TO DATA FLOW DIAGRAMS 47
3.3.3.2. DATAFLOW DIAGRAM 49
3.4. INTERFACE DESIGN 54
3.4.1. USER INTERFACE SCREEN DESIGN 54
3.4.2. OUTPUT DESIGN 56
4. IMPLEMENTATION 60
4.1. CODING STANDARD 60
4.2. SAMPLE CODE 61
5. TESTING 64
5.1. TEST CASES 64
5.1.1. UNIT TESTING 65

5.1.2. INTEGRATION TESTING 67
5.1.3. BLACK BOX TESTING 68
5.1.4. WHITE BOX TESTING 69
5.1.5. VALIDATION TESTING 70
5.1.6. USER ACCEPTANCE TESTING 72
5.2. TEST CASE DOCUMENTS 73
6. CONCLUSION 75
6.1. FUTURE ENHANCEMENTS 75
7. REFERENCES 76
8. APPENDIX 77
8.1 SCREENSHOTS 77

LIST OF TABLES

Sl. no.	Table name	Page number
1	Sign off table	22
2	tbl_admin	42
3	tbl_district	42
4	tbl _place	42
5	tbl_user	42
6	tbl_busowner	43
7	tbl_busdetails	43
8	tbl_busrate	43
9	tbl_bustiming	44
10	tbl_bustype	44
11	tbl_cnclbus	44
12	tbl_complaint	44
13	tbl_complainttype	45
14	tbl_dailycollection	45
15	tbl_feedback	45
16	tbl_notification	45
17	tbl_route	46
18	tbl_routestop	46
19	tbl_expense	46
20	tbl_expensetype	46
21	Data Flow Diagram Symbols	48
22	Test Case	73

LIST OF FIGURES

Sl. no.	Figure name	Page number
1.	Context level DFD for Bus Enquiry System	49
2	First level DFD for Bus Enquiry System	50
3	Second level DFD for Admin Login	51
4	Second level DFD for Bus Owner Login	51
5	Second level DFD for User Login	51
6	Second level DFD for Bus Owner Registration	52
7	Second level DFD for User Registration	52
8	Second level DFD for Adding Routes	52
9	Second level DFD for Adding Routestops	52
10	Second level DFD for Adding Buses	53
11	Second level DFD for Searching Buses	53
12	Second level DFD for Users Report	51
13	User Registration Form	55
14	Bus Owner Registration Form	56
15	Table format output based on date wise user registration	58
16	Pie chart report based on Bus collection	59
17	Bar graph based report on bus availability	59
18	Unit Testing	65
19	Unit Testing Result	66
20	Integration Testing	67
21	Validation Testing	70
22	Already Exist Validation Testing	71
23	Already Exist Validation Testing Result	71
24	Guest Homepage	77

25	Login Page	77
26	User Registration	78
27	User Home Page	79
28	User Profile Page	79
29	User Search Bus	80
30	Select Route Page	80
31	View Buses Page	81
32	View Bus Timings Page	81
33	Post Complaints Page	82
34	Post Feedbacks Page	82
35	Bus Owner Profile Page	83
36	Request for Cancelling Bus Page	83
37	Admin Home Page	84
38	Adding Routes Page	84
39	Accept/Reject Bus Owners	85
40	View and Reply Complaints	85
41	View Users Report	86
42	Registration Confirmation Email From User	86

1.INTRODUCTION

1.1 BACKGROUND AND MOTIVATION

A bus is a company that provides air transport services for traveling passengers and freight. Bus utilize aircraft to supply these services and may form partnerships or alliances with other bus for codeshare agreements. Generally, bus companies are recognized with an air operating certificate or license issued by a governmental aviation body. Bus vary in size, from small domestic bus to full-service international bus with double decker airplanes. Bus services can be categorized as being intercontinental, domestic, regional, or international, and may be operated as scheduled services or charters. The largest bus currently is American Bus Group.

Since most common public transport is bus, most of us face many issues while using buses such as entering wrong buses because of having not enough knowledge about buses routes and bus names. In addition to this, most of us especially old people are not aware about the timings of bus arrival so we may lose buses and have to wait for long period of time at public bus stops and bus stands. This project provides users information about buses and their timings, charges, routes so that they no longer have to wait for buses.

1.2 THE PROPOSED SYSTEM

The proposed system is developed to ensure that people will no longer lose the bus due to less knowledge about bus times at various places. Also if there is any problem with the bus, users will get notified about it so that it won't waste their time . The customers can register their details and after registering they can login using their unique ID and password.After log-in to the system, they can search for buses .He/she also have the provision to edit their profile.

The system admin has the provision to login to the system by using a unique

username and password . The a dministrator of the system is the person who have the permission to add busowners, to accept buses, cancel buses, product gallery images. He / she also has the provision to make necessary changes in the above details if necessary. The system administrator has the authoritative power to view the details of all the busowners and users who have registered into the system. He/she can also make them active/inactive. He/she has the right to accept or reject any bus owners and buses if necessary.

1.3 PROJECT SCOPE

1.3.1. Limitations of Existing System

- **Data storage:-** In manual system paper files require a huge amount of storage space and paper storage creates several problems like spoilage the deterioration by way of aging, humidity etc... Paper based systems are generally very bulky both to handle to store and office space are expensive.

- **Speed of processing:-** The speed of execution of data is slow in the existing system. Processing is slower where large volumes of data need to be dealt with. Slower processing means that some information that could be provided if computerized systems were used will not be provided at all, because there is no time.

- **Speed of retrieval of information:-** The speed of retrieval information is very slow in this system. Since all details are entered on registers, if we want to retrieve the information about an old customer, we want to go through all the past records until we find the right one.

- **Time and manpower consuming:-** A considerable amount of time is required for recording details into the system. Report generation of variousareas is done manually using great amount of manpower and

time. Erroneous records may lead to misleading information, which is more likely in manual system.

- **Accuracy:-** In the existing system the error rate is high and it is difficult to locate the errors and correct them. Calculations made on papers often leads to cash mismatch and inaccurate results.
- **Alternations:-** It is difficult to make corrections. If a manual document contains errors or need updating it is often necessary to recreate the whole document from scratch, rather than just a new version with the relevant details changed.
- **Redundancy:-** If a customer gives different works at different time, each time the customer arrives, the administrator want to store the personnel details repeatedly with each work.
- **User friendliness:-** In the existing system, the degree of user friendliness is considerably low. This system involves readability of the records and maintenance of different details. The technique used in the system is more complicated and there is a lack of technical background towards the system.
- **Back up:-** Back up of data cannot be done easily since all data are in different registers and are written on paper.

1.3.2 Advantages Of Proposed System

Data entry screens are designed such that they are very user friendly and minimum typing is required from the user,

- Not much training required.
- System provides various information's report quickly and accurately in easily understandable formats.
- The new Web application is more user friendly.
- It aims on paperless work.
- Fast access information.
- Efficient traceability.
- Talking into the speed of computer access, large data in less time and facilities

provided by the access.

- Duplication of data will be avoided.
- Menu driven interface provides ease to use.
- Availability of previous data for future reference.

2. SYSTEM ANALYSIS

2.1 INTRODUCTION

Software Engineering is the analysis, design, construction, verification and management of technical or social entities. To engineer software accurately, a software engineering process must be defined. System analysis is a detailed study of the various operations performed by the system and their relationship within and module of the system. It is a structured method for solving the problems related to the development of a new system.The detailed investigation of the present system is the focal point of system analysis. This phase involves the study of parent system and identification of system objectives. Information has to be collected from all people who are affected by or who use the system. During analysis,data are collected on the variable files, decision point and transactions handled by the present system. The main aim of system is to provide the efficient and user friendly automation.So the system analysis process should be performed with extreme precision, so that an accurate picture of existing system, its disadvantages and the requirements of the new system can be obtained.

System analysis involves gathering the necessary information and using the structured tool for analysis. This includes the studying existing system and its drawback,designing a new system and conducting cost benefit analysis. System analysis is a problem solving activity that requires intensive communication between the system users and system developers. The system is studied to the minute detail and analyzed. The system is viewed as a whole and the inputs to the system are identified. The outputs from the organization are traced through various phases of processing of inputs.

There are a number of different approaches to system analysis. When a computer based information system is developed, systems analysis(according to the Waterfall model) would consti tute the following steps:

- The development of a feasibility study, involving determining whether a project is economically, technologically and operationally feasible.
- Conducting fact-finding measures, designed to ascertain the requirements of the system's end-users. These typically span interviews, questionnaires, or visual observations of the work of the existing system.

- Gauging how the end-users would operate the system (in terms of general experience in using computer hardware or software), what the system would be used for and so on.

Techniques such as interviews, questionnaires etc. can be used for the detailed study of these processes. The data collected by these sources must be scrutinized to arrive at aconclusion. The conclusion is an understanding of how the system functions. This system is called the Existing System. The Existing system is then subjected to close observation and the problemareas are identified. The designer now functions as a problem solver and tries to sort out thedifficulties that the enterprise faces. The solutions are given as a proposal which is the Proposed System. The proposal is then weighed with the existing system analytically and thebest one is selected. The proposal is then presented to the user for an endorsement by the user. The proposal is reviewed on user request and suitable changes are made. This is a loop that ends as soon as the user is satisfied with the proposal.

2.2 STAKE HOLDERS OF THIS PROJECT

2.2.1 Administrator

Administrator is the person who manages the software. He is the person who focuses on the data and reports of the software. He/she is the person who add places, routes, bus stops, accept bus owners and their buses.

In the case of other stakeholders, they have only access to read or search the questions provided in the database. But in the case of administrator he is the top person, he can view and access all details.

2.2.2 Bus Owner

Bus owners can add buses, add notifications about the current status of buses.Bus owners have to update their daily collection and expense of buses

2.2.3 User

They are main users of this application.They can enquire about buses and their timings. Customers are also needed to register by providing their basic information such as name, email address,
phone number, address, etc., to order use the system. They can also view and update their information. They can also post complaints and feedbacks.

2.3 SOFTWARE REQUIREMENT SPECIFICATION

2.3.1 Admin

1. Admin can login to the system.
2. Admin should have the provision to add products based on category.
3. Admin also have the privilege to edit or update product details.
4. Admin have the privilege to assign staff for their work.
5. Admin should have the provision to logout.

2.3.2 Bus Owner

1. The system should have a provision to login the bus owner by entering bus owner Username and password.
2. After logging in the bus owner have the permission to view and edit their profile.
3. The bus owner should have the permission to edit the password
4. The bus owner can request to add buses through the available routes specified by admin

2.3.3 User

1. Customers can login to the system.
2. Customer should the privilege to search buses and view their details
3. The system should have the provision to change password.
4. The system should have the provision to logout

Table 2.1 Sign of table

Sl. No.	Name & Designation	Date	Accepted (Yes/No)
1	Dr. Leeja Regi Assistant Professor, Dept of Computer Application BPC College, Piravom		
2	Sanjay Benoy Developer		

2.4 FEASIBILTY STUDY

Feasibility is defined as the practical extent to which a project can be performed successfully. To evaluate feasibility, a feasibility study is performed, which determines whether the solution considered to accomplish the requirements is practical and workable in the software. Information such as resource availability, cost estimation for software development, benefits of the software to the organization after it is developed and cost to be incurred on itsmaintenance are considered during the feasibility study. The objective of the feasibility study is to establish the reasons for developing the software that is acceptable to users, adaptable to change and conformable to established standards. Various other objectives of feasibility studyare listed below.

- To analyse whether the software will meet organizational requirements.
- To determine whether the software can be implemented using the current technology and within the specified budget and schedule.
- To determine whether the software can be integrated with other existing software.
- When our project guide as well as our client Resmi K R told us regarding the mini project and about Word to the Wise for getting the desired product developed, it comes up with rough idea about what all functions the software must perform and which all features are expected from the software.
- Referencing to this information, we does a studies and discussions about whether the desired system and its functionality are feasible to develop and the output of this phase is a feasibility study report that should contained adequate comments and recommendations.
- Various types of feasibility that we checked include technical feasibility, operational feasibility,

and economic feasibility.

2.4.1 Technical Feasibility

Technical feasibility assesses the current resources (such as hardware and software) and technology, which are required to accomplish user requirements in the software within the allocated time and budget. For this, the software development team ascertains whether the current resources and technology can be upgraded or added in the software to accomplish specified user requirements. Technical feasibility also performs the following tasks.

- Analyses the technical skills and capabilities of the software developmentteam members.
- Determines whether the relevant technology is stable and established.
- Ascertains that the technology chosen for software development has a large number of users so that they can be consulted when problems arise or improvements are required.

From our perspective there are two languages PHP, HTML and database MySQL which are used to develop this web based applications. PHP is used in the front end and MySQL is used in the back end. The Word to the Wise is web based and thus can be accessed through any browsers. As we are using these latest technologies which are currently trending and used by a number of developers across the globe, we can say that our project is technically feasible.

2.4.2 Operational Feasibility

Operational feasibility assesses the extent to which the required software performsa series of steps to solve business problems and user requirements. This feasibility is dependent on human resources (software development team) and involves visualizing whether the software will operate after it is developed and be operative once it is installed. Operational feasibility also performs the following tasks.

- Determines whether the problems anticipated in user requirements are of high priority.
- Determines whether the solution suggested by the software development team is acceptable.
- Analyses whether users will adapt to a new software.
- Determines whether the organization is satisfied by the alternative solutions

proposed by the software development team.

We found that our project will be satisfied for the client since we were discussing every detail about the software with the client at every step. The most important part of operational feasibility study is the input from client. So the software is built completely according to the requirements of the client. We have used the current industry standards for the software. Hencewe can say that this software is operationally feasible.

2.4.3 Economic Feasibility

Economic feasibility determines whether the required software is capable of generating financial gains for an organization. It involves the cost incurred on the software development team, estimated cost of hardware and software, cost of performing feasibility study, and so on.For this, it is essential to consider expenses made on purchases (such as hardware purchase) andactivities required to carry out software development. In addition, it is necessary to consider thebenefits that can be achieved by developing the software. Software is said to be economically feasible if it focuses on the issues listed below.

- Cost incurred on software development to produce long-term gains foran organization.
- Cost required to conduct full software investigation (such as requirements elicitation and requirements analysis).
- Cost of hardware, software, development team, and training.

It is estimated that our project is economically feasible as development cost is very minimal since the tools and technologies used are available online. It's a group student project so thereare no personnel costs. Development time is well planned and will not affect other operations and activities of the individuals. Once the system has been developed, the companies purchasing the system will be providing with a manual for training purposes. There is no needto purchase new hardware since the existing computers can still be used to implement the newsystem.

2.5 SOFTWARE DEVELOPMENT LIFECYCLE MODEL

One of the basic notions of the software development process is SDLC models which stand for Software Development Life Cycle models. SDLC – is a continuous process, which starts fromthe moment, when it's made a decision to launch the project, and it ends at the moment of its full remove from the exploitation. Software development lifecycle (SDLC) is a framework thatdefines the steps involved in the development of software. It covers the detailed plan for building, deploying and maintaining the software. SDLC defines the complete cycle of development i.e. all the tasks involved in gathering a requirement for the maintenance of a Product.

Some of the common SDLC models are Waterfall Model, V-Shaped Model, Prototype Model, Spiral Model, Iterative Incremental Model, Big Bang Model, Agile Model. We used Agile Model for our Project.

Agile Model

Agile Model is a combination of the Iterative and incremental model. This model focuses moreon flexibility while developing a product rather than on the requirement. In the agile methodology after every development iteration, the client is able to see the result and understandif he is satisfied with it or he is not. Extreme programming is one of the practical use of the agilemodel. The basis of this model consists of short meetings where we can review our project. In Agile, a product is broken into small incremental builds. It is not developed as a complete product in one go. At the end of each sprint, the project guide verifies the product and after his approval, it is finalised. Client feedback is taken for improvement and his suggestions and enhancement are worked on in the next sprint. Testing is done in each sprint to minimize the risk of any failures.

Advantages of Agile Model:

- It allows more flexibility to adapt to the changes.
- The new feature can be added easily.
- Customer satisfaction as the feedback and suggestions are taken at every stage.
- Risks are minimized, thanks to the flexible change process

Disadvantages:

- Lack of documentation.
- If a customer is not clear about how exactly they want the product to be, then the project would fail.
- With all the corrections and changes there is possibility that the project will exceed expected time

2.6 HARDWARE AND SOFTWARE REQUIRMENTS

2.6.1 Software Specifications

This project is built upon the latest technology software.

Front End	:	HTML
Development tool	:	HTML, JavaScript, CSS, Bootstrap, Ajax, jQuery,PHP
Database	:	MySQL
Web server	:	Wamp server
Operating System	:	Windows 10

2.6.2 HTML

HTML is a computer language devised to allow Website creation. These Websites can then be viewed by anyone else connected to the Internet.It is relatively easy to learn,with the basics being accessible to most people in one sitting; and quite powerful in what it allows you to create.It is constantly undergoing revision and evolution to meet the demands and requirements of the growing Internet audience under the direction of the W3C, the organization charged with designing and maintaining the language.

HTML consists of a series of short codes typed into a text-file by the site author - these are the tags. The text is then saved as a HTML file, and viewed through a browser, like Internet Explorer.This browser reads the file and translates the text into a visible form, hopefully render ing the page as the author had intended.Writing your own HTML entails using tags correctly to create your vision. You can use anything from a rudimentary text- editor to a powerful graphical editor to create HTML pages.

2.6.3 CSS

Stands for "Cascading Style Sheet." Cascading style sheets are used to format the layout of Web pages. They can be used to define text styles, table sizes, and other aspects of Web pages that previously could only be defined in a page's HTML. CSS helps Web developers create a uniform look across several pages of a Website. Instead of defining the style of each table and each block of text within a page's HTML, commonly used styles need to be defined only once in a CSS document. Once the style is defined in cascading style sheet, it can be used by any page that references the CSS file. Plus, CSS makes it easy to change styles across several pages at once. For example, a Web developer may want to increase the default text size from 10pt to 12pt for fifty pages of a Web site. If the pages all reference the same style sheet, the text size only needs to be changed on the style sheet and all the pages will show the larger text. While CSS is great for creating text styles, it is helpful for formatting other aspects of Web page layout as well. For example, CSS can be used to define the cell padding of table cells, the style, thickness, and color of a table's border, and the padding around images or other objects. CSS gives Web developers more exact control over how Web pages will look than HTML does. This is why most Web pages today incorporate cascading style sheets.

2.6.4 Ajax

Ajax is a set of web development techniques using many web technologies on the client side to create asynchronous web applications. With Ajax, web applications can send and retrieve data from a server asynchronously(in the background) without inter fering with the display and behavior of the existing page.By decoupling the data inter change layer from the presentation layer, Ajax allows web pages and, by extension, web applications, to change content dynamically without the need to reload the entire page. In practice, modern implementations commonly utilize JSON instead of XML.

2.6.5 jQuery

jQuery is a JavaScript library designed to simplify HTML DOM tree traversal and manipulation, as well as event handling, CSS animation, and Ajax. It is free, open source software using the permissive MIT License. As of May 2019, jQuery is used by 73% of the 10 million most popular websites. Web analysis indicates that it is the most widely deployed JavaScript library by a large margin, having to 4 times more usage than any other JavaScript library.

2.6.6 Bootstrap

Bootstrap is a free and open-source CSS framework directed at responsive, mobile first front-end web development. It contains CSS-and (optionally) JavaScript-based design templates for typography, forms, buttons, navigation and other interface components.

2.6.7 JavaScript

JavaScript is a dynamic computer programming language.It is lightweight and most comm only used as a part of Web pages, whose implementations allow client-side script to interact with the user and make dynamic pages. It I s an interpreted programming lang uage with object-oriented capabilities. JavaScript was first known as LiveScript, but Nets cape changed its name to JavaScript, possibly because of the excitement being generated by Java. JavaScript made its first appearance in Netscape 2.0 in 1995 with the name LiveScript. The general-purpose core of the language has been embedded in Netscape, Internet Explorer, and other Web browsers.

2.6.8 PHP

PHP started out 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 MUST for students and working professionals to become a great Software Engineer specially when they are working in Web Development Domain. The key advantages of learning PHP are:

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,even build entire e-commerce sites.It is integrated with a number of popular databases, including MySQL, PostgreSQL, bOracle, 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 c omplex 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 C-Like.

2.6.9 Wamp Server

Stands for "Windows, Apache, MySQL, and PHP." WAMP is a variation of LAMP for Windows systems and is often installed as a software bundle (Apache, MySQL, and PHP). It is often used for web development and internal testing, but may also be used to serve live websites. The most important part of the WAMP package is Apache (or "Apache HTTP Server") which is used run the web server within Windows. By running a local Apache web server on Windows machine, a web developer can test webpages in a web browser without publishing them live on the Internet. WAMP also includes MySQL and PHP, which are two of the most common technologies used for creating dynamic websites. MySQL is a high-speed database, while PHP is a scripting language that can be used to access data from the database. By installing these two components locally, a developer can build and test a dynamic website before publishing it to a public web server. While Apache, MySQL, and PHP are open source components that can be installed Individually, they are usually installed together. One popular package is called"WampServer," which provides a user-friendly way to install and configure the "AMP" components on Windows.

2.6.10 MySQL

MySQL is the world's most popular open source database software, with over 100 million copies of its software downloaded or distributed throughout its history. With its superior speed, reliability, and ease of use, MySQL has become the preferred choice for Web, Web 2.0, SaaS, ISV, Telecom companies and forward-thinking corporate IT Managers because it eliminates the major problems associated with downtime, maintenance and administration for modern,online applications.Many of the world's largest and fastest-growing organizations use MySQL to save time and money powering their high-volume Web sites, critical business systems, and packaged software —including industry leaders such as Yahoo!, Alcatel-Lucent, Google, Nokia, YouTube, Wikipedia, and Booking.com.

The flagship MySQL offering is MySQL Enterprise, a comprehensive set of production-tested software, proactive monitoring tools, and premium support services available in an affordable annual subscription. MySQL is a key part of LAMP (Linux, Apache, MySQL, PHP / Perl / Python), the fast-growingopen source enterprise software stack. More and more companies are using LAMP as an alternativeto expensive proprietary software stacks because of its lower cost and freedom from platform lock-in.

MySQL was originally founded and developed in Sweden by two Swedes and a Finn: David Axmark, Allan Larsson and Michael "Monty" Widenius, who had worked together since the 1980's. MySQL, the most popular Open Source SQL database management system, is developed, distributed, and supported by Oracle Corporation.

MySQL is a database management system. A database is a structured collection of data. It may be anything from a simple shopping list to a picture gallery or the vast amounts of information in a corporate network. To add, access, and process data stored in a computer database, you need a database management system such as MySQL Server. Since computers are very good at handling large amounts of data, database management systems play a central role in computing, as standalone utilities, or as parts of other applications.

MySQL databases are relational. A relational database stores data in separate tables rather than putting all the data in one big storeroom. The database structures are organized into physical files optimized for speed. The logical model, with objects such as databases, tables, views, rows, and columns, offers a flexible programming environment. You set up rules governing the relationships between different data fields, such as one-to one, one-to-many, unique, required or optional, and ―pointers‖ between different tables. The database enforces these rules, so that with a well-designed database, your application never sees inconsistent, duplicate, orphan, out-of-date, or missing data.

The SQL part of ―MySQL‖ stands for ―Structured Query Language‖. SQL is themost common standardized language used to access databases. Depending on your programming environment,you might enter SQL directly (for example, to generate reports), embed SQL statements into codewritten in another language, or use a language-specific API that hides the SQL syntax.

SQL is defined by the ANSI/ISO SQL Standard. The SQLstandard has been evolving since 1986and several versions exist. In this manual, ―SQL-92‖ refers to the standard released in 1992,―SQL:1999‖ refers to the standard released in 1999, and ―SQL:2003‖ refers to the current version of the standard.

We use the phrase ―the SQL standard‖ to mean the current version of the SQL Standard at any time. MySQL software is Open Source. Open Source means that it is possible for anyone to useand modify the software. Anybody can download the MySQL software from the Internet and use it without paying anything. If you wish, you may study the source code and change it to suit your needs.

The MySQL software uses the GPL (GNU General Public License) ,http://www.fsf.org/licenses/, to define what you may and may not do with the software in different situations. If you feel uncomfortable with the GPL need to embed MySQL code into a commercial application, you canbuy a commercially licensed version from us.

The MySQL Database Server is very fast, reliable, scalable, and easy to use. If that is what you are looking for, you should give it a try. MySQL Server can run comfortably on a desktop or laptop, alongside your other applications, web servers, and so on, requiring little or no attention. If you dedicate an entire machine to MySQL, you can adjust the settings to take advantage of all the memory, CPU power, and I/O capacity available. MySQL can also scale up to clusters of machines, networked together.

MySQL Server was originally developed to handle large databases much faster than existing solutions and has been successfully used in highly demanding production environments for several years. Although under constant development, MySQL Server today offers a rich and useful set of functions. Its connectivity, speed, and security make MySQL Server highly suited for accessing databases on the Internet. MySQL Server works in client/server or embedded systems.

A large amount of contributed MySQL software is available. MySQL Server has a practical set of features developed in close cooperation with our users. It is very likely that your favourite application or language supports the MySQL Database Server.

2.6.11 WINDOWS 10

Operating System is defined as a program that manages the computer hardware. An operating system can be viewed as a scheduler, where it has resources for which it has charge. Resources include CPU, memory, I/O device and disk space. In another view, the operating system is a new machine. The third view is that operating system is a multiplexer which allows sharing of resources provides protection from interference and provides a level of cooperation between users. This project is developed using Windows 10 as the operating system and supports its latest versions. Windows 10 is a series of personal computer operating systems produced by Microsoft as part of its Windows NT family of operating systems. It is the successor to Windows8.1, and was released to manufacturing on July 15, 2015, and to retail on July 29, 2015. One ofWindows 10's most notable features is support for universal apps. Windows 10 also introduced the Microsoft Edge web browser, a virtual desktop system, a window and desktop management feature called Task View, support for fingerprint and face recognition login, new security features for enterprise environments, and DirectX12. Windows 10 received mostly positive reviews upon its original release in July 2015. Critics praised Microsoft's decision to provide a desktop-oriented interfacing line with previous versions of Windows, contrasting the tablet- oriented approach of 8, although Windows 10's touch-oriented user interface mode was criticized for containing regressions upon the touch-oriented interface of Windows 8. Critics also praised the improvements to Windows 10's bundled software over Windows 8.1, Xbox Live integration, as well as the functionality and capabilities of the Cortana personal assistant and the replacement of Internet Explorer with Microsoft Edge. However, media outlets have been critical of changes to operating system behaviours, including mandatory update installation, privacy concerns over data collection performed by the OS for Microsoft and its partners and

the adware-like tactics used to promote the operating system on its release.

2.6.12 Microsoft Word

Microsoft Word (or simply Word) is a word processor developed by Microsoft. It was first released on October 25, 1983 under the name *Multi-Tool Word* for Xenix systems. Subsequent versions were later written for several other platforms including IBM PCs running DOS (1983), Apple Macintosh running the Classic Mac OS (1985), AT&T Unix PC (1985), Atari ST (1988), OS/2 (1989), Microsoft Windows (1989), SCO Unix (1994), and macOS (formerly OS X; 2001).

Commercial versions of Word are licensed as a standalone product or as a component of Microsoft Office, Windows RT or the discontinued Microsoft Works suite. Unlike most MS- DOS programs at the time, Microsoft Word was designed to be used with a mouse. Advertisements depicted the Microsoft Mouse, and described Word as a WYSIWYG, windowed word processor with the ability to undo and display bold, italic, and underlinedtext, although it could not render fonts. It was not initially popular, since its user interface was different from the leading word processor at the time, WordStar. However, Microsoft steadily improved the product, releasing versions 2.0 through 5.0 over the next six years. In 1985, Microsoft ported Word to the classic Mac OS (known as Macintosh System Software at the time). This was made easier by Word for DOS having been designed for use with high- resolution displays and laser printers, even though none were yet available to the general public. Following the precedents of LisaWrite and MacWrite, Word for Mac OS added true WYSIWYG features. It fulfilled a need for a word processor that was more capable than MacWrite. After its release, Word for Mac OS's sales were higher than its MS-DOS counterpart for at least four years.

2.6.13 SmartDraw

SmartDraw is a diagram tool used to make flowcharts, organization charts, mind maps, project charts, and other business visuals. SmartDraw has two versions: an online edition and a downloadable edition for Windows desktop.

SmartDraw integrates with Microsoft Office products including Word, PowerPoint, and Excel and G Suite applications like Google Docs and Google Sheets. SmartDraw has apps for Atlassian's Confluence, Jira, and Trello. SmartDraw is compatible with Google Drive, Dropbox, Box, and OneDrive.

Since 1994, the mission of SmartDraw Software has been to expand the ways in which people communicate so that we can clearly understand each other, make informed decisions, and work

together to improve our businesses and the world. We accomplish this by creating software and services that make it possible for people to capture and present information as visuals, while being a pleasure to use. In 2019, we took this to the next level by launching VisualScript, whichmakes it easy to visualize data in relational formats like trees, flows, and timelines, automatically, without any human input. VisualScript is a relationship visualization platform that empowers organizations to visualize data across siloed ecosystems and gain critical insights in real-time. Today, SmartDraw Software is one of the most sophisticated digital marketing organizations in the world with over 90,000 unique visitors to our website each business day and in excess of 3,000,000 installations of our apps each year. SmartDraw is used by more thanhalf of the Fortune 500 and by over 250,000 public and private enterprises of all sizes around the world. Privately held, SmartDraw Software is headquartered in San Diego, California.

Hardware Requirments

The selection of hardware configuring is a very task related to the software development, particularly inefficient RAM may affect adversely on the speed and corresponding on the efficiency of the entire system. The processor should be powerful to handle all the operations

The hard disk should have the sufficient to solve the database and the application.

Minimum hardware requirement :

CPU : Pentium IV Processor

Memory : 256 MB Above

Cache : 512 KB Above

Hard disk : 20 GB Above

Monitor : Any

Keyboard : Any

Mouse : Any

3.SYSTEM DESIGN

3.1 SYSTEM ARCHITECTURE

A system architecture or system's architecture is the conceptual model that defines the structure, behaviour, and more views of a system. An architecture description is a formal description and representation of a system, organized in a way that supports reasoning about the structures of the system,

System architecture can comprise system components, the externally visible properties of those components, the relationships (e.g. the behaviour) between them. It can provide a plan from which products can be procured, and systems developed, that will work together to implement the overall system. There have been efforts to formalize languages to describe system architecture; collectively these are called architecture description languages (ADLs).

The system architecture can best be thought of as a set of representations of an existing (or to be created) system. It is used to convey the informational content of the elements comprising a system, the relationships among those elements, and the rules governing those relationships. The architectural components and set of relationships between these components that architecture describes may consist of hardware, software, documentation, facilities, manual procedures, or roles played by organizations or people. System architecture is primarily concerned with the internal interfaces among the system's components or subsystems, and the interface between the system and its external environment, especially the user.

The structural design reduces complexity, facilitates change and result in easier implementation by encouraging parallel development of different parts of the system. The procedural design transforms structural elements of program architecture into a procedural description of software components. The architectural design considers architecture as the most important functional requirement. The system is based on the three-tier architecture.

The first level is the user interface (presentation logic), which displays controls, receives and validates user input. The second level is the business layer (business logic) where the application specific logic takes place. The third level is the data layer where the application information is stored in files or database. It contains logic about to retrieve and update data. The important feature about the three-tier design is that the information only travels from one level to adjacent level.

3.2 MODULE DESIGN

1. **LOGIN**

   This module allows the users to login to the website using his/her unique faculty identity card number and password so that he/she can add questions and can view the questions that are already uploaded by other faculties in the same department in the website. Along with viewing the products they can buy those.

2. **REGISTRATION**

   This module helps the admin, busowners, users to register to our website. He/she can use the registration form for registering his/her account. While registering the faculty needs to provide some data for example email id, password etc.

3. **REPORTS**

   This module allows the admin of the site to view the details of the site through various reports. This includes pie chart and some table reports. All these reports provide information about the academics and their details such as departments, courses, semesters, subjects, etc. So that the administrator can easily manage the website.

4. **ROUTE MANAGEMENT**

   This module allows the admin to add routes and its associated stops.

5. **SALES**

   This modules includes in sales. Product booking and sales are managed in this module.
   Customers can view products and book them for buying.
   Payment can be added for the product in this module.

3.3 DATABASE DESIGN

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 information access easy, quick, inexpensive and flexible for the users. The general theme behind a database is to integrate all information. Database design is recognized as a standard of management information system and is available virtually for every computer system. In database design several specific

objectives are considered:

- Ease of learning and use
- Controlled redundancy
- Data independence
- More information at low cost
- Accuracy and integrity
- Recovery from failure
- Privacy and security
- Performance

A database is an integrated collection of data and provides centralized access to the data. Usually the centralized data managing the software is called RDBMS. The main significant difference between RDBMS and other DBMS is the separation of data as seen by the program and data has in direct access to stores device. This is the difference between logical and physical data.

3.3.1 Normalization

Designing a database is complete task and the normalization theory is a useful aid in the design process. The process of normalization is concerned with transformation of conceptual schema into computer representation form. There will be need for most databases to grow by adding new attributes and new relations. The data will be used in new ways. Tuples will be added and deleted. Information stored may undergo updating also. New association may also be added. In such situations the performance of a database is entirely depend upon its design. A bad database design may lead to certain undesirable things like:

- Repetition of information
- Inability to represent certain information
- Loss of information

To minimize these anomalies, Normalization may be used. If the database is in a normalized form, the data can be growing without, in most cases, forcing the rewriting application programs. This is important because of the excessive and growing cost of maintaining an organization's application programs and its data from the disrupting effects of database growth.

As the quality of application programs increases, the cost of maintaining the without normalization will rise to prohibitive levels. A normalized database can also encompass many related activities of an organization thereby minimizing the need for rewriting the applications of programs. Thus, normalization helps one attain a good database design and there by ensures continued efficiency of database.

Normalization theory is built around the concept of normal forms. A relation is said to be in normal form if it satisfies a certain specified set of constraints. For example, a relation is said to be in first normal form (1NF) if it satisfies the constraint that it contains atomic values only. Thus every normalized relation is in 1NF.Numerous normal forms have been defined. Codd defined the first three normal forms.

All normalized relations are in 1NF, some 1NF relations are also in 2NF and some 2NF relations are also in 3NF.2NF relations are more desirable than 1Nf and 3NF are more desirable than 2NF. That is, the database designer should prefer 3NF than 1NF or 2NF.Normalization procedure states that a relation that is in some given normal form can be converted into a set of relations in a more desirable form. We can define this procedure as the successive reduction of a given collection of relations to some more desirable form. This procedure is reversible. That is, it is always possible to take the output from the procedure and convert them back into input. In this process, no information is lost. So it is also called “no loss decomposition”.

First Normal Form

A relation is in first normal form (1NF) if and all its attributes are based on single domain. The objective of normalizing a table is to remove its repeating groups and ensure that all entries of the resulting table have at most single value.

Second Normal Form

A table is said to be second Normal Form (2NF), when it is in 1NF and every attribute in record is functionally dependent upon the whole key, and not just a part of the key.

Third Normal Form

A table is in third Normal Form (3NF), when it is in 2NF and every non-key attribute is functionally dependent on just the primary key.

3.3.2 Table Structure

Table is a collection of complete details about a particular subject. These data are saved in rows and Columns. The data of each Row are different units. Hence, rows are called RECORDS and Columns of each row are called FIELDS.

Data is stored in tables, which is available in the backend the items and data, which are entered in the input, form id directly stored in this table using linking of database. We can link more than one table to input forms. We can collect the details from the different tables to display on the output.

There are mainly 15 tables in the project. They are,

1. tbl_admin
2. tbl_district
3. tbl_place
4. tbl_user
5. tbl_busowner
6. tbl_busdetails
7. tbl_busrate
8. tbl_bustiming
9. tbl_bustype
10. tbl_cnclbus
11. tbl_complaint
12. tbl_complainttype
13. tbl_dailycollection
14. tbl_feedback
15. tbl_notification
16. tbl_route
17. tbl_routestop
18. tbl_expense
19. tbl_expensetype

1. Table: tbl_admin

Description: This table is used to store the details of administrator.

Slno	Field Name	Data type	Size	Constraints
1	admin_id	INT		Primary key
2	admin_mail	VARCHAR	30	
3	admin_pass	VARCHAR	50	
4	admin_name	VARCHAR	50	

2. Table: tbl_district

Description: This table is used to store districts.

Slno	Field Name	Data type	Size	Constraints
1	district_id	INT		Primary key
2	district_name	VARCHAR	30	

3. Table: tbl_place

Description: This table is used to store different places.

Slno	Field Name	Data type	Size	Constraints
1	place_id	INT		Primary key
2	place_name	VARCHAR	50	
3	district_id	INT		Foreign Key
4	place_pin	VARCHAR	10	

4. Table: tbl_user

Description: This table is used to store the details of users

Slno	Field Name	Data type	Size	Constraints
1	user_id	INT		Primary key
2	user_name	VARCHAR	30	
3	user_gender	VARCHAR	10	
4	user_contact	VARCHAR	15	
5	user_email	VARCHAR	50	
6	user_proof	VARCHAR	100	
7	user_photo	VARCHAR	100	
8	place_id	INT		Foreign key
9	reg_date	DATE		
10	user_address	VARCHAR	200	
11	user_pass	VARCHAR	20	

5. Table: tbl_busowner

Description: This table contains details of busowners

Slno	Field Name	Data type	Size	Constraints
1	owner_id	INT		Primary key
2	owner_name	VARCHAR	50	
3	owner_gender	VARCHAR	8	
4	owner_mail	VARCHAR	40	
5	owner_pass	VARCHAR	50	
6	owner_licenseno	VARCHAR	50	
7	owner_contact	VARCHAR	50	
8	owner_aadharno	VARCHAR	20	
9	owner_photo	VARCHAR	100	
10	owner_status	INT		
11	place_id	INT		Foreign Key

6. Table: tbl_busdetails

Description: This table contains details of buses.

Slno	Field Name	Data type	Size	Constraints
1	bus_id	INT		Primary key
2	bus_name	VARCHAR	30	
3	bus_regno	VARCHAR	40	
4	bus_photo	VARCHAR	100	
5	bus_status	INT		
6	bus_insno	INT		
7	bus_capacity	INT		
8	owner_id	INT		Foreign Key
9	route_id	INT		Foreign Key
10	bustype_id	INT		Foreign Key

7. Table: tbl_busrate

Description: This table is used to store bus charges

Slno	Field Name	Data type	Size	Constraints
1	busrate_id	INT		Primary key
2	bustype_id	INT		Foreign Key
3	busrate_minrate	INT		
4	busrate_minkm	INT		
5	busrate_addrate	INT		
6	busrate_addkm	INT		

8. Table: tbl_bustiming

Description: This table is used to store the timings of buses

Slno	Field Name	Data type	Size	Constraints
1	bustime_id	INT		Primary key
2	bus_id	INT		Foreign Key
3	routestop_id_	INT		Foreign Key
4	bus_time	TIME		

9. Table: tbl_bustype

Description: This table stores details about bus types

Slno	Field Name	Data type	Size	Constraints
1	bustype_id	INT		Primary key
2	bustype_name	VARCHAR	50	

10. Table: tbl_cnclbus

Description: This table is used to store details about cancelling buses

Slno	Field Name	Data type	Size	Constraints
1	cncl_id	INT		Primary key
2	bus_id	INT		Foreign key
3	cncl_reason	VARCHAR	400	

11. Table: tbl_complaint

Description: This table is used to store complaint details

Slno	Field Name	Data type	Size	Constraints
1	complaint_id	INT		Primary key
2	bus_id	INT		Foreign key
3	user_id	INT		Foreign key
4	ctype_id	DATE		Foreign key
5	complaint_status	INT		
6	complaint_reply	VARCHAR	400	
7	complaint_content	VARCHAR	400	
8	complaint_date	DATE		

12. Table: tbl_complaintype

Description: This table id used to store complaint type.

Slno	Field Name	Data type	Size	Constraints
1	ctype_id	INT		Primary key
2	ctype_name	VARCHAR	500	

13. Table: tbl_dailycollection

Description: This table is used to store daily collection of buses

Slno	Field Name	Data type	Size	Constraints
1	collec_id	INT		Primary key
2	collec_date	DATE		
3	bus_id	INT		Foreign key
4	collec_amt	VARCHAR	10	

14. Table: tbl_feedback

Description: This table is used to store the feedback.

Slno	Field Name	Data type	Size	Constarints
1	f_id	INT		Primary key
2	f_msg	VARCHAR	1000	
3	user_id	VARCHAR	100	Foreign key
4	bus_id	INT		Foreign key

15. Table: tbl_notification

Description: This table is used to store status of buses

Slno	Field Name	Data type	Size	Constraints
1	not_id	INT		Primary key
2	bus_id	INT		Foreign key
3	not_date	DATE		
4	not_msg	VARCHAR	200	

16. Table: tbl_route

Description: This table is used to store bus routes

Slno	Field Name	Data type	Size	Constraints
1	route_id	INT		Primary key
2	route_name	VARCHAR	40	
3	route_frmplace	INT		
4	route_toplace	INT		

17. Table: tbl_routestop

Description: This table is used to store stops in bus routes

Slno	Field Name	Data type	Size	Constraints
1	routestop_id	INT		Primary key
2	route_id	INT		Foreign key
3	routestop_name	VARCHAR	40	

18. Table: tbl_expense

Description: This table is used to store the expense daily expense details of buses

Slno	Field Name	Data type	Size	Constraints
1	exp_id	INT		Primary key
2	exp_date	DATE		
3	exp_amt	VARCHAR	20	
4	exptype_id	INT		
5	bus_id	INT		Foreign key

19. Table: tbl_expensetype

Description: This table is used to store the daily expense types of buses

Slno	Field Name	Data type	Size	Constraints
1	exptype_id	INT		Primary key
2	exptype_name	VARCHAR	20	

3.3.3 Data Flow Diagram

3.3.3.1 Introduction To Data Flow Diagram

Data Flow Diagram is a network that describes the flow of data and processes that change, or transform, data throughout the system. This network is constructed by use a set of symbols that do not imply a physical implementation. It is a graphical tool for structured analysis of the system requirements. DFD models a system by using external entities from which data flows to a process, which transforms the data and creates, output-data-flows which go to other processes or external entities or files. Data in files may also flow to processes as inputs.

There are various symbols used in a DFD. Bubbles represent the processes. Named arrows indicate the data flow. External entities are represented by rectangles. Entities supplying data are known as sources and those that consume data are called sinks. Data are stored in a data store by a process in the system. Each component in a DFD is labelled with a descriptive name. Process names are further identified with a number.

The Data Flow Diagram shows the logical flow of a system and defines the boundaries of the system. For a candidate system, it describes the input (source), outputs (destination), database (files) and procedures (data flow), all in a format that meet the user's requirements.

The main merit of DFD is that it can provide an overview of system requirements, what data a system would process, what transformations of data are done, what files are used, and where the results flow.

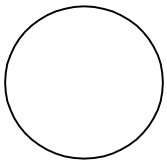
This network is constructed by use a set of symbols that do not imply a physical implementation. It is a graphical tool for structured analysis of the system requirements. DFD models a system by using external entities from which data flows to a process, which transforms the data and creates, output-data-flows which go to other processes or external entities or files. External entities are represented by rectangles. Entities supplying data are known as sources and those that consume data are called sinks. Data are stored in a data store by a process in the system. It

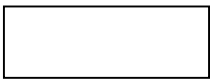
is a graphical tool for structured analysis of the system requirements. DFD models a system by using external entities from which data flows to a process, which transforms the data and creates, output-data-flows which go to other processes or external entities or files. Data in files may also flow to processes as inputs.

Rules for constructing a Data Flow Diagram

1. Arrows should not cross each other
2. Squares, circles and files must bear names.
3. Decomposed data flow squares and circles can have same time
4. Choose meaningful names for data flow
5. Draw all data flows around the outside of the diagram

Basic Data Flow Diagram Symbols

	A data flow is a route, which enables packets of data to travel from one point to another. Data may flow from a source to a process and from data store or process. An arrow line depicts the flow, with arrow head pointing in the direction of the flow.
	Circles stands for process that converts data in to information. A process represents transformation where incoming data flows are changed into outgoing data flows.

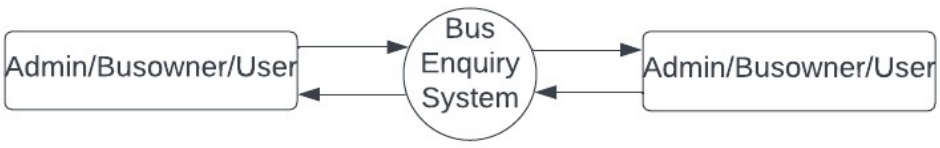
	A data store is a repository of data that is to be stored for use by a one or more process may be as simple as buffer or queue or sophisticated as relational database. They should have clear names. If a process merely uses the content of store and does not alter it, the arrowhead goes only from the store to the process. If a process alters the details in the store then a double-headed arrow is used.
	A source or sink is a person or part of an organization, which enters or receives information from the system, but is considered to be outside the contest of data flow model.

3.3.3.2 Data Flow Diagram

Each component in a DFD is labelled with a descriptive name. Process name are further identified with number. Context level DFD is draw first. Then the process is decomposed into several elementary levels and is represented in the order of importance. A DFD describes what data flow (logical) rather than how they are processed, so it does not depend on hardware, software, and data structure or file organization.

A DFD methodology is quite effective; especially when the required design.

Zeroth Level DFD for Bus Enquiry System



First Level DFD for Bus Enquiry System

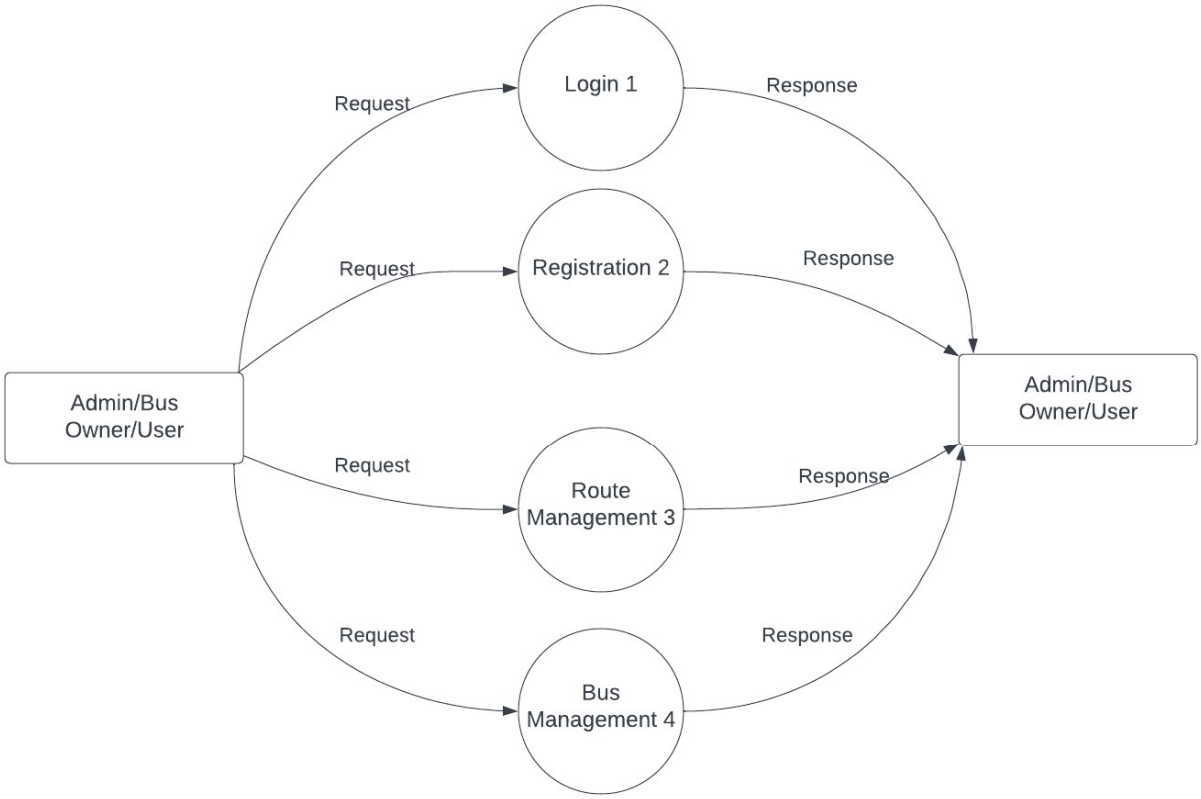


Fig 3.1 First Level DFD For Bus Enquiry

Second Level DFD for Bus Enquiry

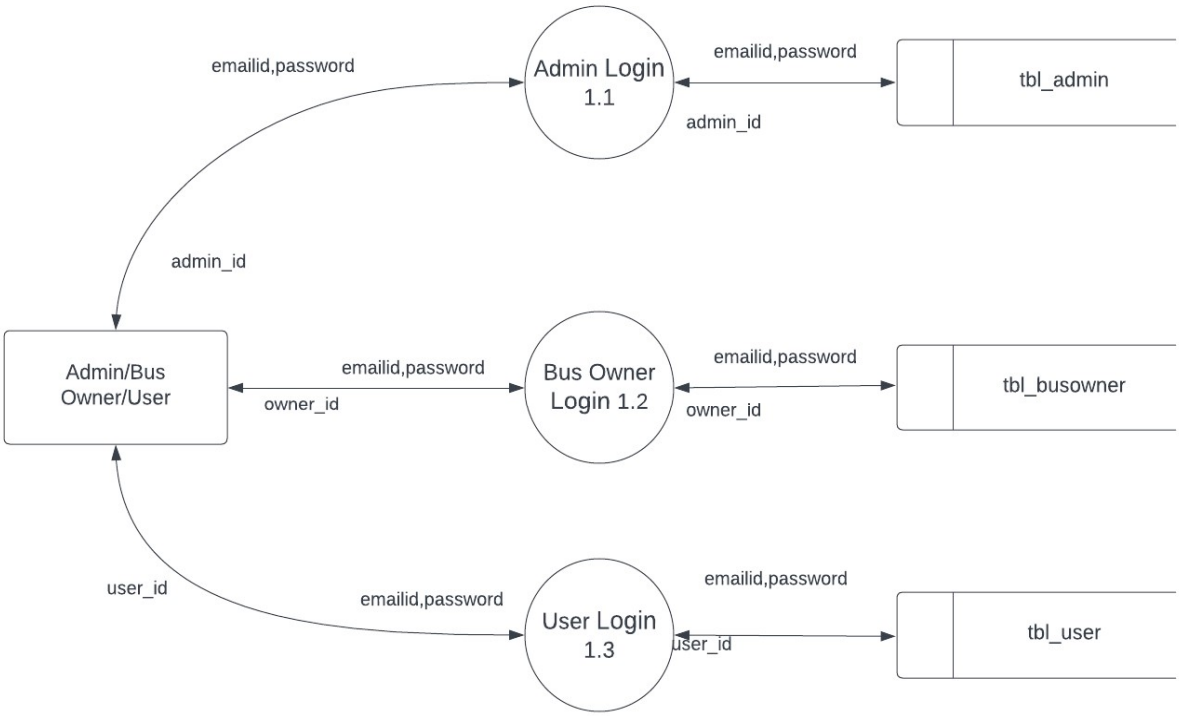


Fig 3.2 Second Level DFD For Admin/Bus Owner/User Login

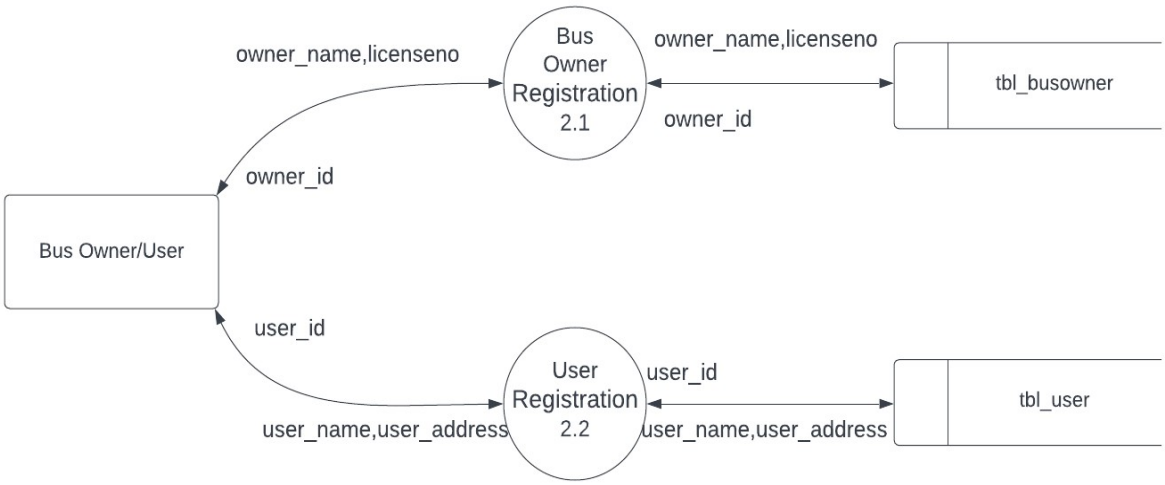


Fig 3.3 Second Level DFD For Bus Owner/User Registration

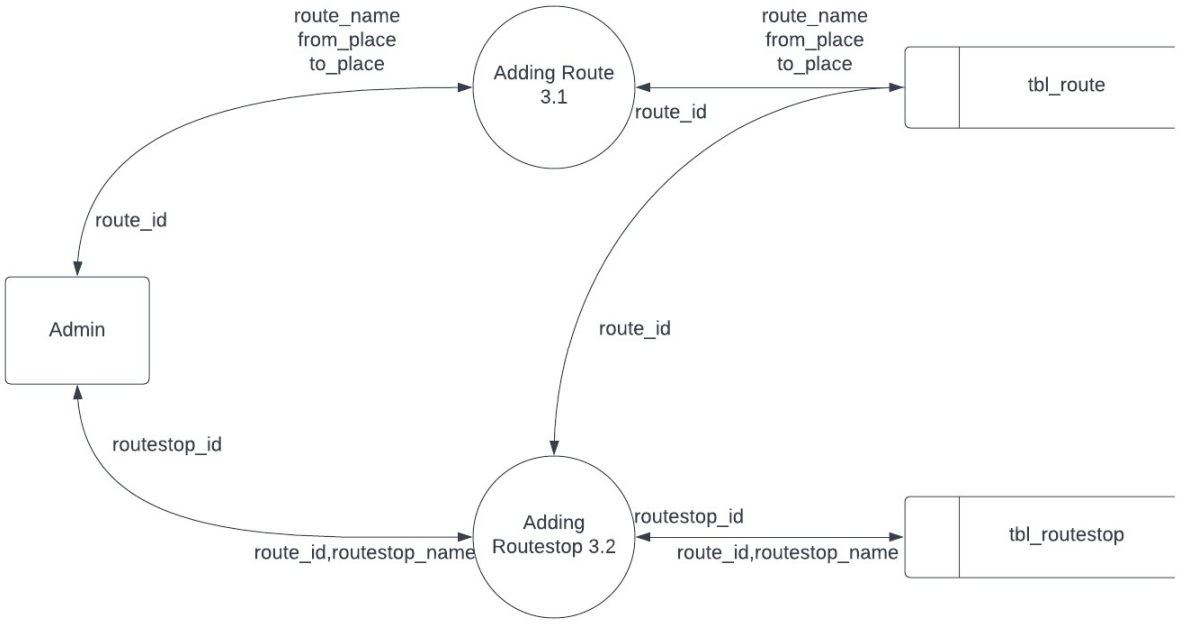


Fig 3.4 Second Level DFD for Adding Routes And Routestops

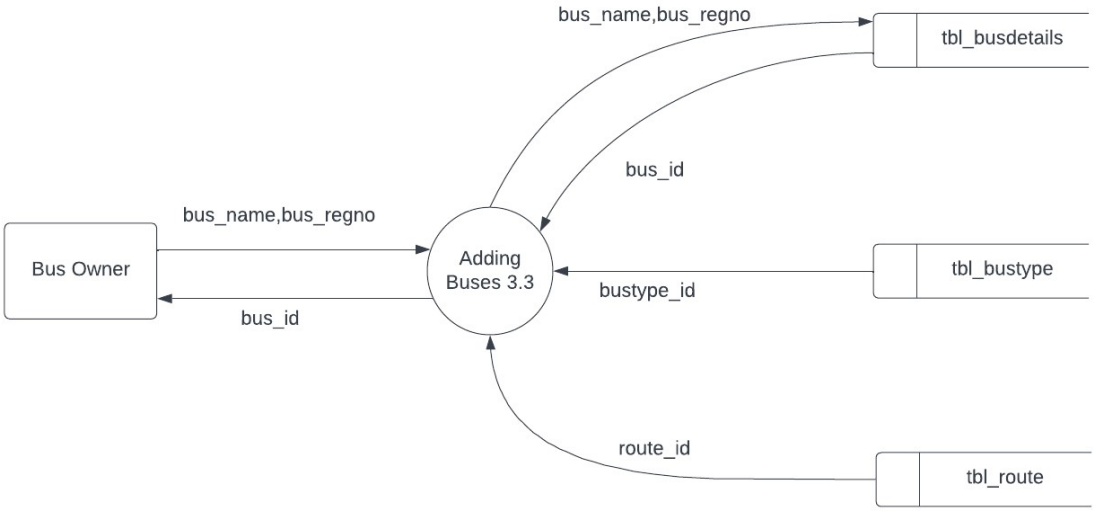


Fig 3.5 Second Level DFD For Adding Buses

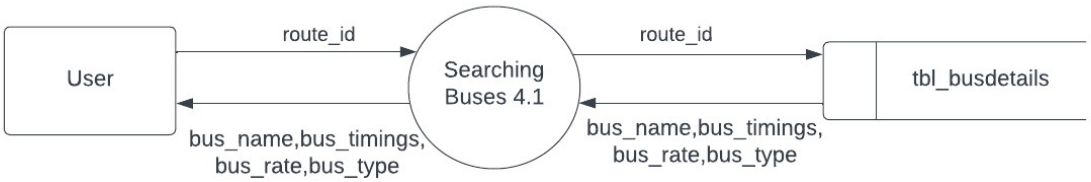


Fig 3.6 Second Level DFD For Searching Buses

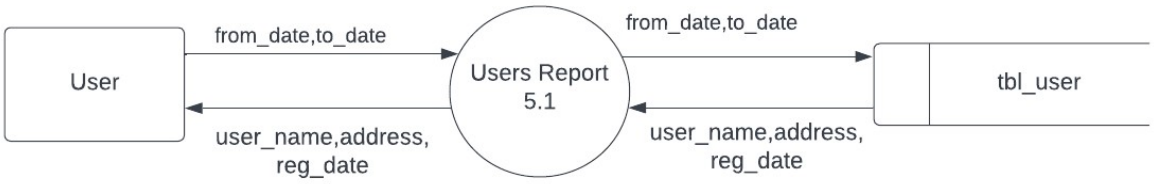


Fig 3.7 Second Level DFD For Users Report

3.4 INTERFACE DESIGN

These modules can apply to hardware, software or the interface between a user and a machine. An example of a user interface could include a GUI, a control panel for a nuclear power plant, or even the cockpit of an aircraft. In systems engineering, all the inputs and outputs of a system, subsystem, and its components are listed in an interface control document often as part of the requirements of the engineering project. The development of a user interface is a unique field.

3.4.1 User Interface Screen Design

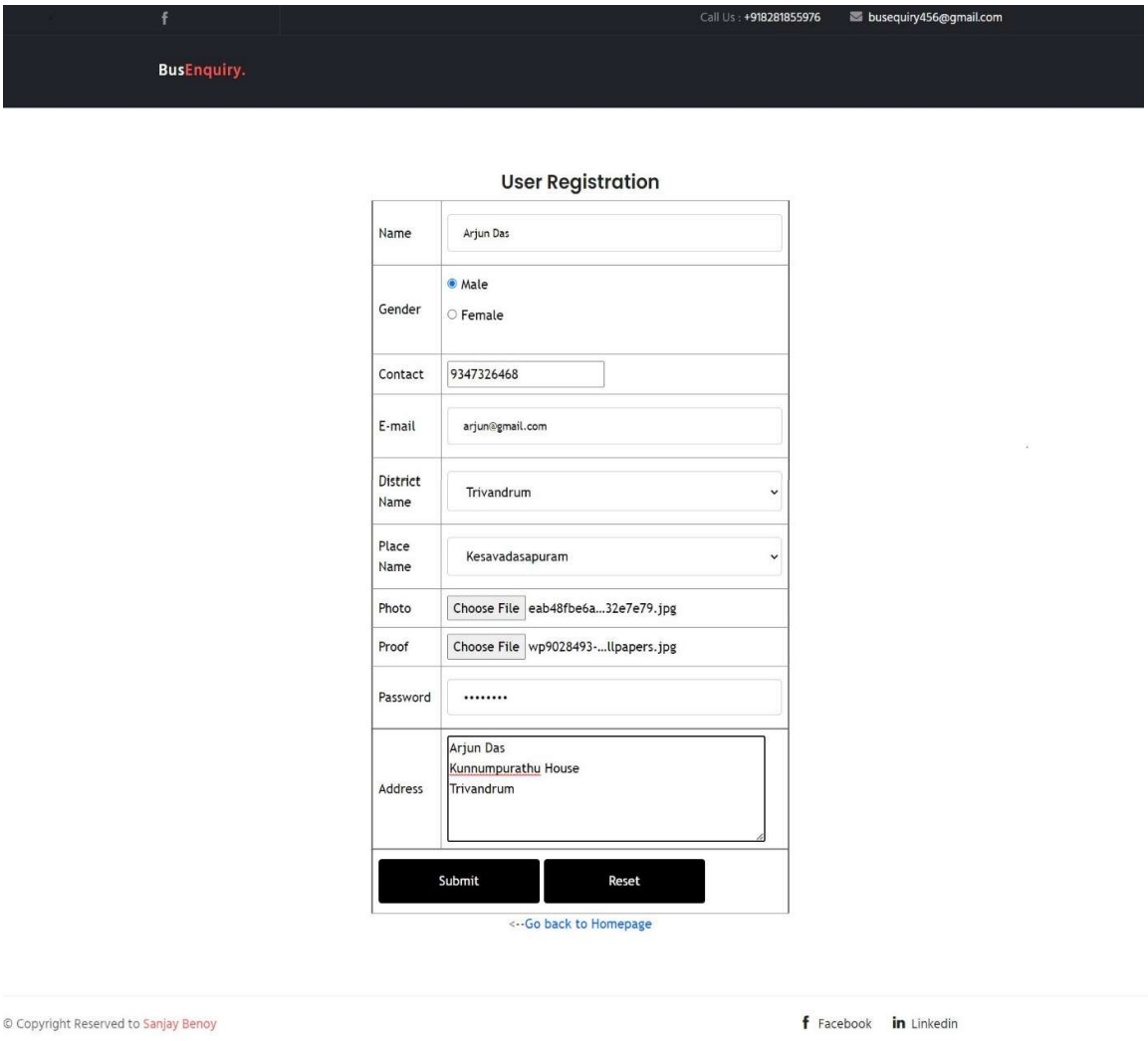
The user interface design is very important for any application. The interface design describes how the software communicates within itself, to system that interpreted with it and with humans who use it. The input design is the process of converting the user-oriented inputs into the computer based format. The data is fed into the system using simple inactive forms. The forms have been supplied with messages so that the user can enter data without facing any difficulty. They data is validated wherever it requires in the project. This ensures that only the correct data have been incorporated into system. The goal of designing input data is to make the automation as easy and free from errors as possible. For providing a good input design for the application easy data input and selection features are adopted. The input design requirements such as user friendliness, consistent format and interactive dialogue for giving the right messages and help for the user at right are also considered for development for this project.

Input Design is a part of the overall design. The input methods can be broadly classified into batch and online. Internal controls must be established for monitoring the number of inputs and for ensuring that the data are valid. The basic steps involved in input design are:

- Review input requirements.
- Decide how the input data flow will be implemented.
- Decide the source document.
- Prototype on line input screens.
- Design the input screens.

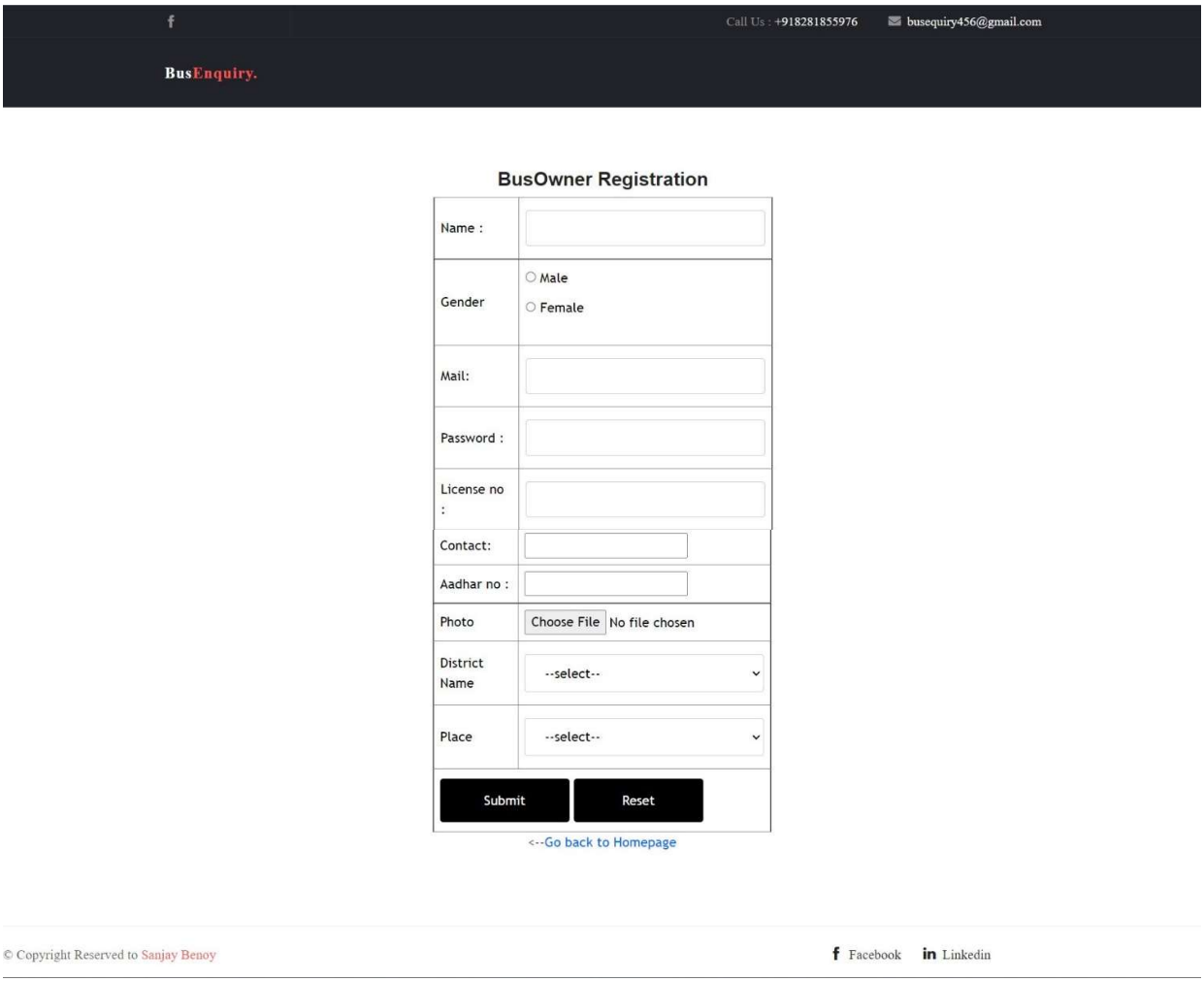
The quality of the system input determines the quality of the system output. Input specifications describe the manner in which data enter the system for processing. Input design features can ensure the reliability of the system and produce results from accurate data. The input design also determines whether the user can interact efficiently with the system.

These are the two sample input forms



User Registration Form

This input form is for the registration of new bus users. It contains textboxes for inputting name, email id, place, address and phone number of the user, a password box for giving the password. After clicking the submit button the bus owner details will be saved. And with this username and password he/she can login to the system.



This is the registration form of new bus owners. After registering, admin need to accept him to to start his bus service.

3.4.2 Output Design

A quality output is one, which meets the requirements of end user and presents the

information clearly. In any system result of processing are communicated to the user and to the other system through outputs. In the output design it is determined how the information is to be displayed for immediate need.

It is the most important and direct source information is to the user. Efficient and intelligent output design improves the system's relationships with the user and helps in decision - making. The objective of the output design is to convey the information of all the past activities, current status and to emphasis important events. The output generally refers to the results and information that is generated from the system. Outputs from computers are required primarily to communicate the results of processing to the users.

Output also provides a means of storage by copying the results for later reference in consultation. There is a chance that some of the end users will not actually operate the input data or information through workstations, but will see the output from the system.

Two phases of the output design are:

1. Output Definition
2. Output Specification

Output Definition takes into account the type of output contents, its frequency and its volume, the appropriate output media is determined for output. Once the media is chosen, the detail specification of output documents are carried out. The nature of output required from the proposed system is determined during logical design stage. It takes the outline of the output from the logical design and produces output as specified during the logical design phase.

In a project, when designing the output, the system analyst must accomplish the following:

- Determine the information to present.
- Decide whether to display, print, speak the information and select the output medium.
- Arrange the information in acceptable format.
- Decide how to distribute the output to the intended receipt.

Thus by following the above specifications, a high quality output can be generated.

In our projects outputs are generated as pie charts and in table format on based on date wise user registration.

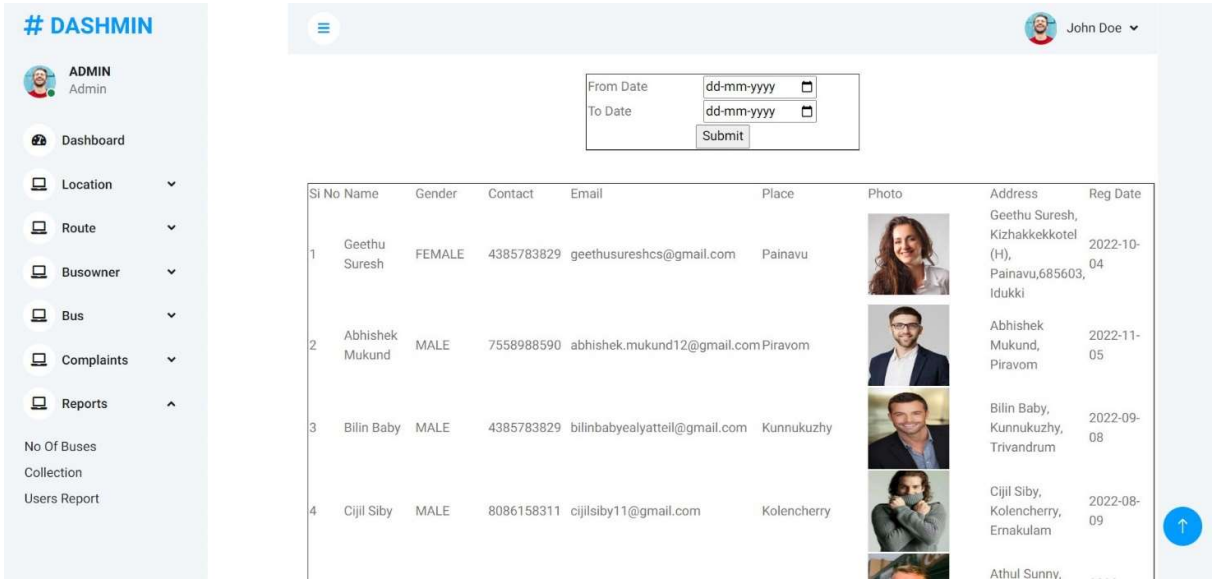
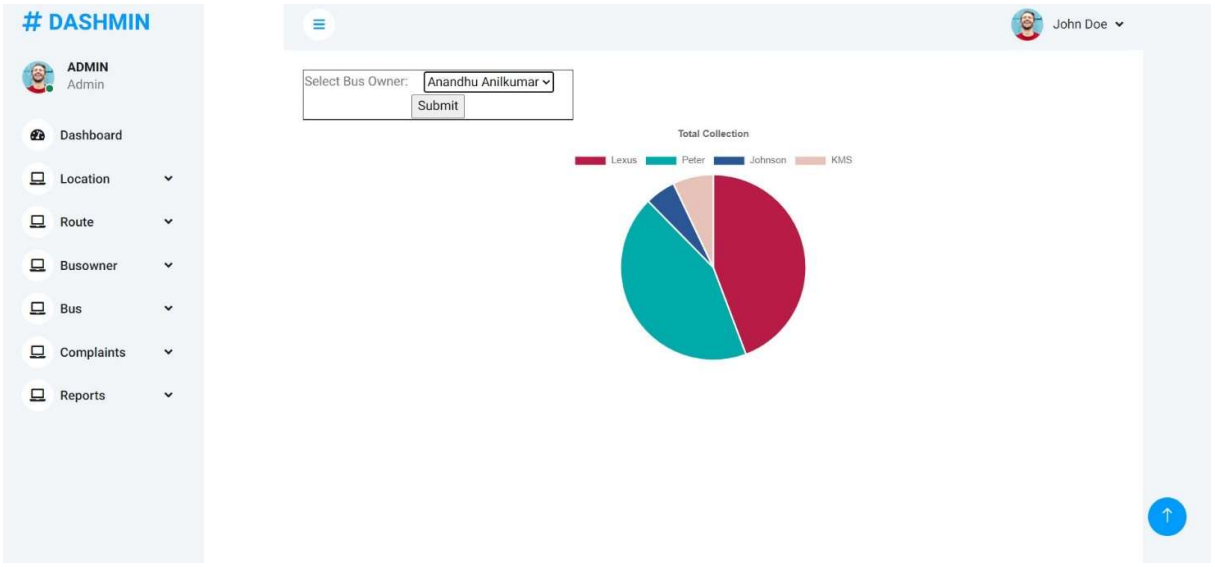
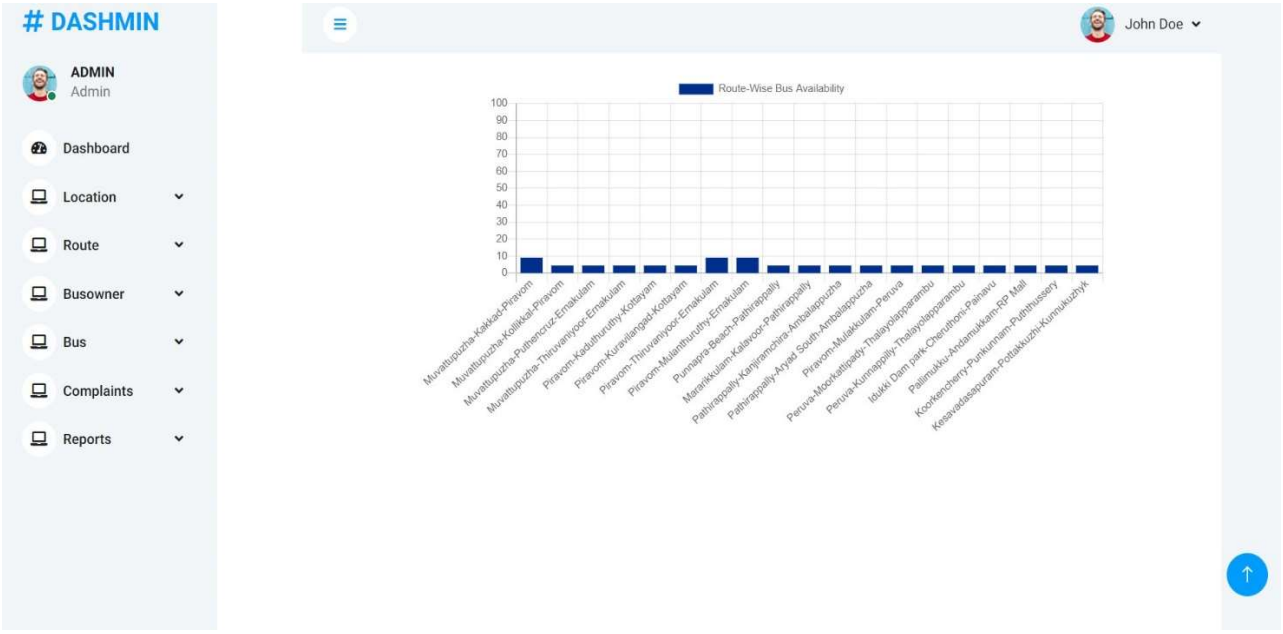


Table format output based on date wise user registration



Pie chart based on Bus collection



Bar graph based on Bus availability

4. IMPLEMENTATION

Implementation is the stage of the project when the theoretical design is turned into a working system. The implementation stage is a systems project in its own rig ht. It includes careful planning, investigation of current system and its constraints on implementation, design of methods to achieve the changeover, training of the staff in the changeover procedure and evaluation of changeover method.

4.1 CODING STANDARDS

Writing an efficient software code requires a thorough knowledge of programming. This knowledge can be implemented by following a coding style which comprises several guidelines that help in writing the software code efficiently and with minimum errors. These guidelines, known as coding guidelines**,** are used to implement individual programming language constructs, comments, formatting, and so on. These guidelines, if followed, help in preventing errors, controlling the complexity of the program, and increasing the readability and understandability of the program.

A set of comprehensive coding guidelines encompasses all aspects of code development. To ensure that all developers work in a harmonized manner (the source code should reflect a harmonized style as a single developer had written the entire code in one session), the developers should be aware of the coding guidelines before starting a software project. Moreover, coding guidelines should state how to deal with the existing code when the software incorporates it or when maintenance is performed.

Since there are numerous programming languages for writing software codes, each having different features and capabilities, coding style guidelines differ from one language to another. However, there are some basic guidelines which are followed in all programming languages. These include naming conventions, commenting conventions, and formatting conventions.

1. **File header comments** are useful in providing information related to a file as a whole and comprise identification information such as date of creation, Dame of

the creator, and a brief description of the software code.

2. **Trailing comments** are used to provide explanation of a single line of code. These comments are used to clarify the complex code. These also specify the function of the abbreviated variable names that are not clear. In some languages, trailing comments are used with the help of a double slash (//).
3. **Indentation:** This refers to one or more spaces left at the beginning of statements in the program. Indentation is useful in making the code easily readable. However, the spaces used for indentation should be followed in the entire program.
4. **Implementing coding guidelines:** If coding guidelines are used in a proper manner, errors can be detected at the time of writing the software code. Such detection in early stages helps in increasing the performance of the software as well as reducing the additional and unplanned costs of correcting and removing errors. Moreover, if a well-defined coding guideline is applied, the program yields a software system that is easy to comprehend and maintain.

4.2 SAMPLE CODE

- **Adding district**

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Bus Enquiry::District</title>
</head>
<?php
ob_start();
include("head.php");
include("../Assets/Connection/Connection.php");
if(isset($_POST["btnsubmit"]))
{
	$dist=$_POST["txt_dist"];
	$sql1="select * from tbl_district where district_name='".$dist."'";
	$rslt=$con->query($sql1);
	if(mysqli_num_rows($rslt)>0)
	{
		?>
		<script>
			alert("District already exist!!");
			 window.location("District.php");
```

```
                </script>
    <?php
        }else
        {
        $insQry="insert into tbl_district(district_name) VALUES('".$dist."')";
        if($con->query($insQry))
        {
         ?>
    <script>
          alert("Data inserted");
          window.location("District.php");
         </script>
  <?php
        }
   else
        {
   ?>
     <script>
          alert("Data Insertion Failed");
               window.location("District.php");
         </script>
    <?php
   }
        }
}
 if(isset($_GET["did"]))
                {
                        $did=$_GET["did"];
                        $delqry="delete from tbl_district where district_id='".$did."'";
                        if($con->query($delqry))
                        {
                                ?>
        <script>
                        alert("deleted..");
                        location.href="District.php";
                        </script>
        <?php
                        }
                        else
                        {
                                ?>
        <script>
                        alert("failed..");
                        location.href="District.php";
                        </script>
        <?php
                        }
                }
                  ?>
<body>
<div >
```

```
<form id="form1" name="form1" method="post" action="District.php">
  <table width="429" height="104" border="1" cellpadding="10">
    <tr>
      <td>District Name</td>
      <td><input type="text" name="txt_dist" id="txt_dist" /></td>
    </tr>
    <tr>
      <td colspan="2"><input type="submit" name="btnsubmit" id="btnsubmit" value="Submit" />
      <input type="reset" name="btncancel" id="btncancel" value="Cancel" /></td>
    </tr>
  </table>
  </form>
  <table border="black">
  <tr>
  <th>SI no</th>
  <th>District Name</th>
  <th>Action</th>
  </tr>

  <?php
  $selqry="select * from tbl_district order by district_name ASC";
  $rows=$con->query($selqry);
  $i=0;
  while($result=$rows->fetch_assoc())
  {
            $i++;
            ?>
  <tr>
   <td><?php echo $i;?></td>
   <td><?php echo $result["district_name"];?></td>
   <td><a href="District.php?did=<?php echo $result["district_id"]?>">delete</a></td>
  </tr>
  <?php
 }
  ?>
  </table>
</div>
</body>
<?php
ob_flush();
include("Foot.php");
?>
</html>
```

5.TESTING

Coding conventions are a set of guidelines for a specific programming language that recommend programming style, practices and methods for each aspect of a piece program written in this language. These conventions usually cover file organization, indentation, comments, declarations, statements, white space, naming conventions, programming practices, programming principles, programming rules of thumb, architectural best practices, etc. These are guidelines for software structural quality. Software programmers are highly recommended to follow these guidelines to help improve the readability of their source code and make software maintenance easier.

5.1 TEST CASES

The objective of system testing is to ensure that all individual programs are working as expected, that the programs link together to meet the requirements specified and to ensure that the computer system and the associated clerical and other procedures work together. The initial phase of system testing is the responsibility of the analyst who determines what conditions are to be tested, generates test data, produced a schedule of expected results, runs the tests and compares the computer produced results with the expected results with the expected results. The analyst may also be involved in procedures testing. When the analyst is satisfied that the system is working properly, he hands it over to the users for testing. The importance of system testing by the user must be stressed. Ultimately it is the user must verify the system and give the go-ahead.

During testing, the system is used experimentally to ensure that the software does not fail, i.e., that it will run according to its specifications and in the way users expect it to. Special test data is input for processing (test plan) and the results are examined to locate unexpected results. A limited number of users may also be allowed to use the system so analysts can see whether they try to use it in unexpected ways. It is preferably to find these surprises before the organization implements the system and depends on it. In many organizations, testing is performed by person other than those who write the original programs. Using persons who do not know how certain parts were designed or programmed ensures more complete and unbiased testing and more reliable software.

Parallel running is often regarded as the final phase of system testing. Since he parallel operation of two systems is very demanding in terms of user resources it should be embarked on only if the user is satisfied with the results of testing -- it should not be started if problems are known to exist. Testing is the major quality control measure during software development. Its basic function is to detect errors in the software. Thus the goal of testing is to uncover requirement design and coding errors in the program.

Testing is the process of correcting a program with intends of finding an error. Different types of testing are,

1. Unit Testing
2. Integrated Testing
3. Black Box Testing
4. White Box Testing
5. Validation Testing
6. User Acceptance Testing

5.1.1 Unit Testing

In computer programming, unit testing is a method by which individual units of source code, sets of one or more computer program modules together with associated control data, usage procedures, and operating procedures are tested to determine if they are fit for use. In this testing we test each module individual and integrated the overall system. Unit testing focuses verification efforts on the smaller unit of software design in the module. This is also known as module testing. The modules of the system are tested separately. The testing is carried out during programming stage itself. In this testing step each module is found to working satisfactory as regard to the expected output from the module. There are some validation checks for verifying the data input given by the user which both the formal and validity of the entered. It is very easy to find error debug the system.

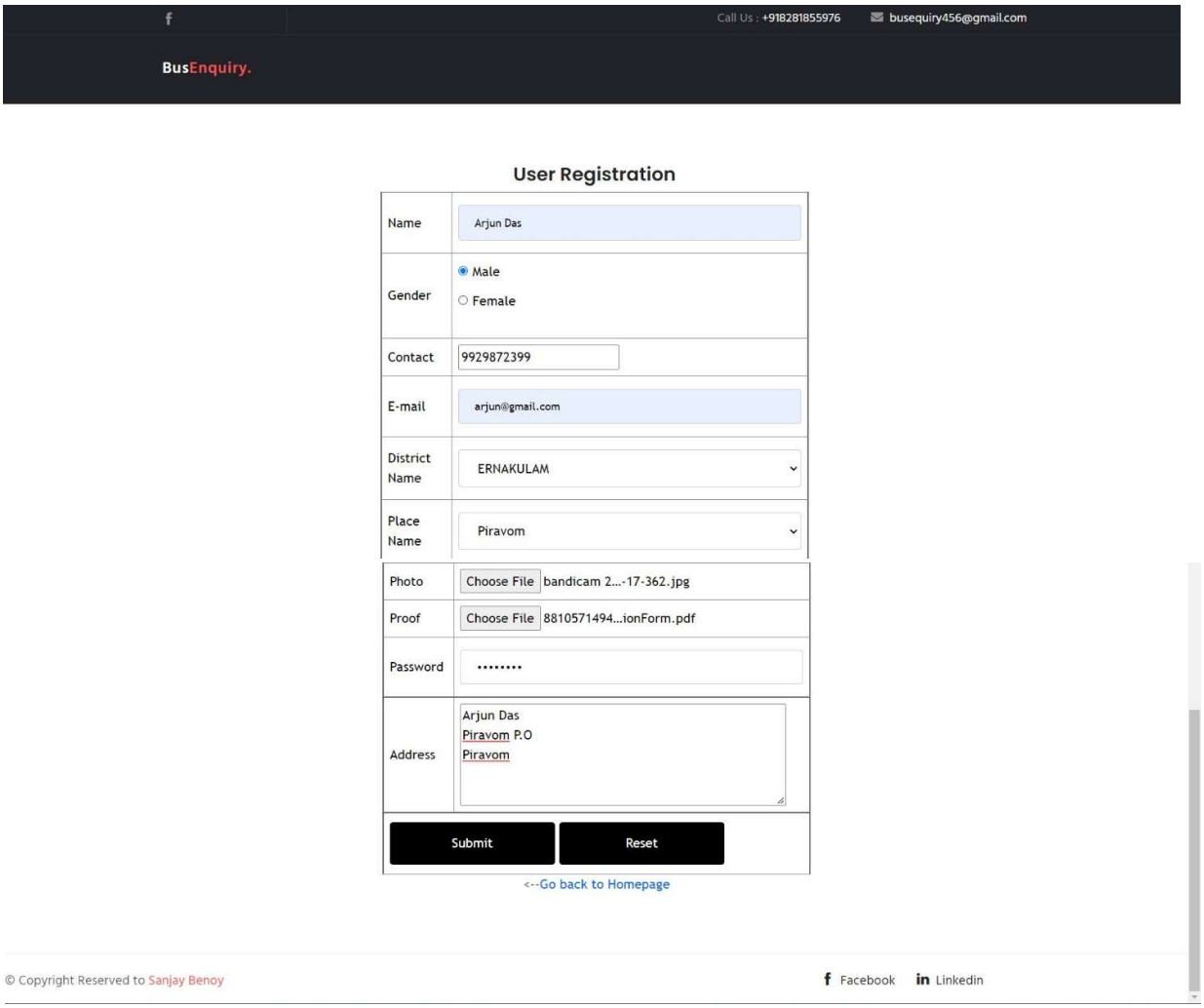


Fig 5.1 Unit Testing

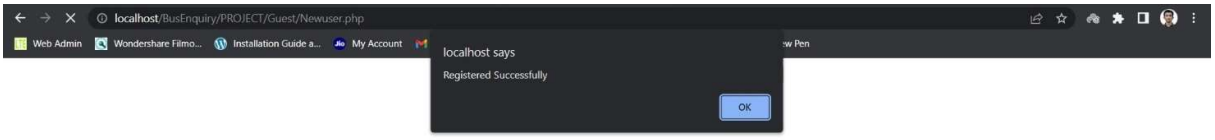


Fig 5.1 Unit Testing Result

I have continued Unit Testing from the starting of the coding phase itself. Whenever I completed one small sub module, some amount of testing was done based on the requirements to see if the functionality is aligned to the gathered requirements.

5.1.2 Integration Testing

Integration testing (sometimes called integration and testing, abbreviated I&T) is the phase in software testing in which individual software modules are combined and tested as a group. Software components may be integrated in an iterative way or all together ("big bang"). Normally the former is considered a better practices since it allows interface is uses to be located more quickly and fixed. Data can be lost across an interface; one module can have an adverse effort on the other sub functions when combined by, may not produce the desired major functions. Integrated testing is the systematic testing for constructing the uncover errors within the interface. This testing was done with sample data. The developed system has run successful for this sample data. The need for integrated test is to find the overall system performance.

Integration testing is a logical extension of unit testing. In its simplest form, two units that have already been tested are combined into a component and the interface between them is tested. A component, in this sense, refers to an integrated aggregate of more than one unit. Integration testing identifies problems that occur when units are combined. By using a test plan that requires you to test each unit and ensure the viability of each before combining units, you know that any errors discovered when combining units are likely related to the interface between units. This method reduces the number of possibilities to a far simpler level of analysis. Progressively larger groups of tested software components corresponding to elements of the architectural design are integrated and tested until the software works as a system.

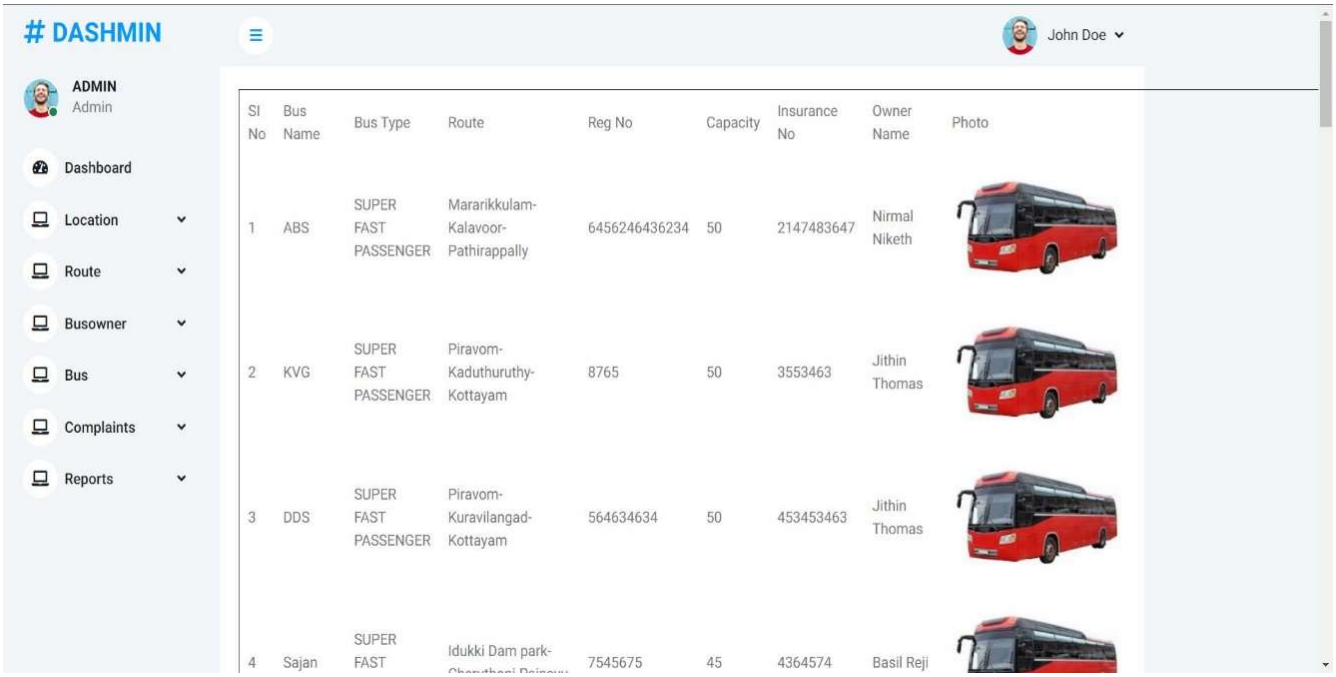


Fig 5.3 Integration Testing

I have performed integration testing whenever I have combined two modules together. When two modules are combined I have checked whether the functionality works correctly or not through integration testing.

5.1.3 Black Box Testing

Black-box testing is a method of software testing that examines the functionality of an application (e.g. what the software does) without peering into its internal structures or workings. This method of test can be applied to virtually every level of software testing: unit, integration, system and acceptance. It typically comprises most if not all higher level testing, but can also dominate unit testing as well. In black box testing the structure of the program is not considered. Test cases are decided solely on the basis of the requirements or the specification of the programmer module, and the internals of the module or program are not considered for selection of the test cases.

In the Black Box testing tester only knows the input that can be given to the system and what output the system should give. In other words, the basis of deciding test cases in functional testing is requirements or specifications of the system or module. This form of testing is also called functional or behavioural testing. One advantage of the black box technique is that no programming knowledge is required. Whatever biases the programmers may have had, the tester likely has a different set and may

emphasize different areas of functionality. On the other hand, black-box testing hasbeen said to be "like a walk in a dark labyrinth without a flashlight. Because they donot examine the source code, there are situations when a tester writes many test cases to check something that could have been tested by only one test case, or leaves some parts of the program untested.

5.1.4 White Box Testing

White-box testing (also known as clear box testing, glass box testing, and transparent box testing and structural testing) is a method of testing software that tests internal structures or workings of an application, as opposed to its functionality. In white-box testing an internal perspective of the system, as well as programming skills, are used to design test cases. The tester chooses inputs to exercise paths through the code and determine the appropriate outputs.This is analogous to testing nodes in a circuit, e.g. in-circuit testing (ICT).

While white-box testing can be applied at the unit, integration and system levels of the software testing process, it is usually done at the unit level. It can test paths within a unit, paths between units during integration, and between subsystems during a system–level test. Though this method of test design can uncover many errors or problems, it might not detect unimplemented parts of the specification or missing requirements. White Box testing is concerned with testing the implementation of the program. The intent of this testing is not to exercise all the different input or output conditions but to exercise the different programming structures and data structures used in the program.

White-box test design techniques include:

- Control flow testing
- Data flow testing
- Branch testing
- Path testing
- Statement coverage
- Decision coverage

5.1.5 Validation Testing

At the culmination of Black Box testing, software is completely assembled as a package, interface errors have been uncovered and corrected and final series of software tests, Validation tests begins. Validation testing can be defined many was but a simple definition is that validation succeeds when the software functions in a manner that can be reasonably accepted by the customer. After validation test has been conducted one of the two possible conditions exists.

1. The function or performance characteristics confirm to specification and are accepted.
2. A derivation from specification uncovered and a deficiency list is created.

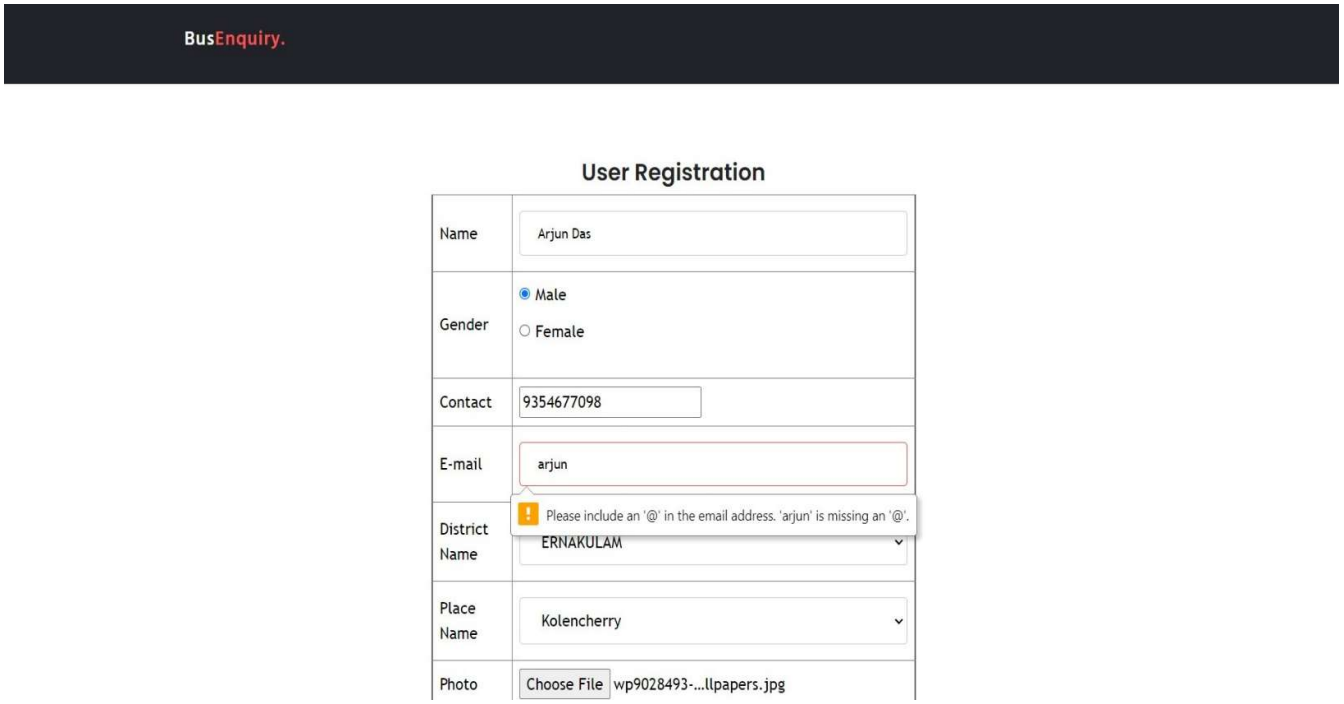


Fig 5.4 Validation Testing

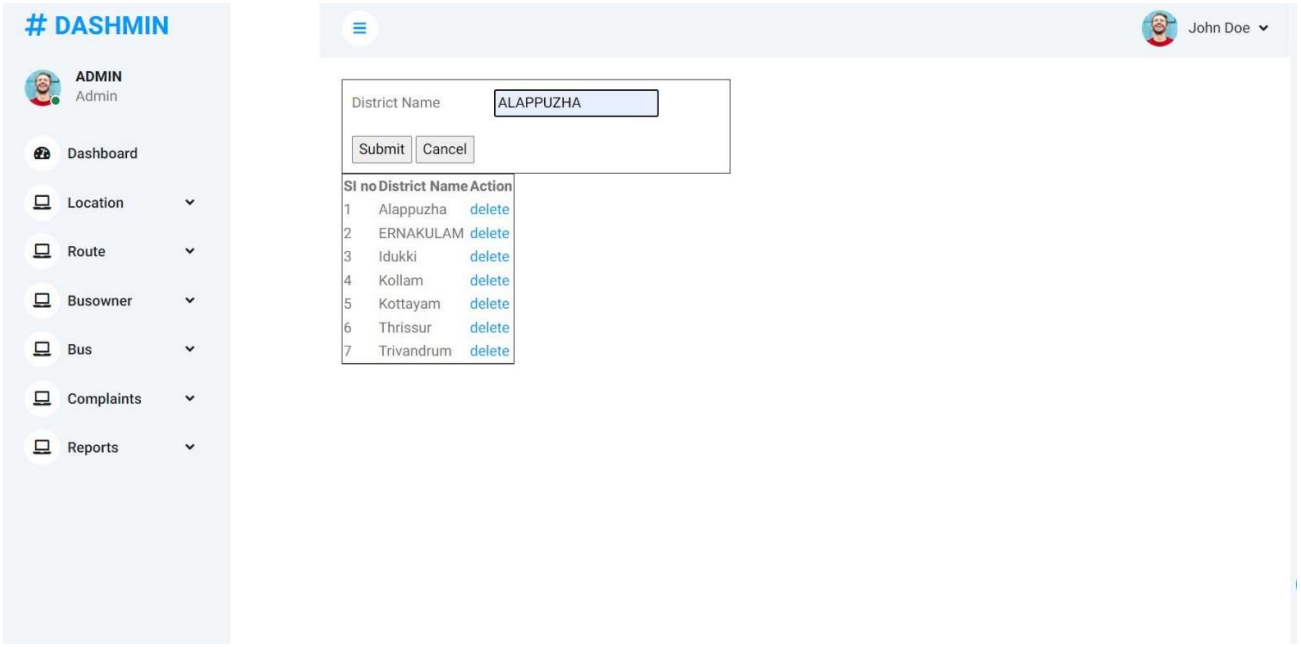


Fig 5.5 Already Exists Validation Testing

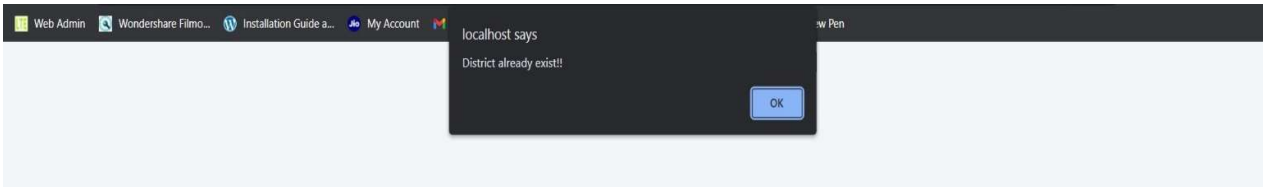


Fig 5.6 Already Exists Validation Testing Result

5.1.6 User Acceptance Testing

Acceptance Testing is a level of the software testing process where a system is tested for acceptability. User Acceptance testing is the software testing process where system tested for acceptability & validates the end to end business flow. Such type of testing executed by client in separate environment & confirms whether system meets the requirements as per requirement specification or not.

UAT is performed after System Testing is done and all or most of the major defects have been fixed. This testing is to be conducted in the final stage of Software Development Life Cycle(SDLC) prior to system being delivered to a live environment. UAT users or end users are concentrating on end to end scenarios & typically involves running a suite of tests on the completed system.

User Acceptance testing also known as Customer Acceptance testing (CAT), if the system is being built or developed by an external supplier. The CAT or UAT are the final confirmation from the client before the system is ready for production. The business customers are the primary owners of these UAT t ests. These tests are created by business customers and articulated in business domain languages. So ideally it is collaboration between business customers, business analysts, test ers and developers.It consists of test suites which involve multiple test cases & each test case contains input data (if required) as well as the expected output. The result of test case is either a pass or fail.

5.2 TEST CASE DOCUMENTS

A test case is a set of conditions or variables under which a tester will determine whether a system under test satisfies requirements or works correctly. The process of developing test cases can also help find problems in the requirements or design o an application. A sample of test case document format is given below.

Table 5.1 Test Case

TC No.	Test Steps	Expected Result	Actual Result	Status	Comment
1	Run application and navigate to login screen	Login screen is displayed. A filed for entering username, a field for entering password and a button to submit should be present	Login screen has been displayed, fields for entering email address and password together with a log in button is available.	Pass	
2	Enter an invalid username and invalid password and press the button	A message should be displayed stating that user name and password are invalid	A message has been displayed stating that user name and password are invalid	Pass	
3	Enter a valid username and password and press the button	User must successfully login to the webpages.	A message has been displayed stating that the login successful and navigate into home page	Pass	Using admin as the username and admin as the password

4	Enter a valid username and leave password and press the button	A message should be displayed stating that please enter the user name and password	A message has been displayed stating that please enter the user name and password	Pass	
5	Leave username and password and press the button	A message should be displayed stating that please enter the user name and password	A message has been displayed stating that please enter the user name and password	Pass	
6	Leave username and enter a valid password and press the button	A message should be displayed stating that please enter the user name and password	A message has been displayed stating that please enter the user name and password	Pass	

6. CONCLUSION

The project was successfully completed within the time span allotted. All the modules are tested separately and put together to form the main system. Finally, the modules are tested with real data and it worked successfully. Thus the system has fulfilled the entire objective defined.

This project will help the customer to reduce man power and consume less time purchasing smart gadgets without going to the shop. Our goal of developing this **"Bus Enquiry System"** has come to a good result without many defects.

The main motive for developing this system is for the welfare of the society by giving all time access to a set of smart gadgets.

6.1 FUTURE ENHANCEMENTS

The system has been designed in such a way that it can be modified with very little effort when such needs arise in the future. New features can be added with slight modifications of software which make it easy to expand the scope of this project. Though the system is working on various assumptions, it can be modified easily to any kind of requirements.

Even though we have tried our best to present the information effectively and efficiently, yet there can be further enhancement in the application. We have taken care of all the critical aspects, which were needed to be taken care of. Because of fast changes in the world of programming this system will gradually get outdated and less effective. For the time being it's possible to overcome problems by amendments and minor modifications to acknowledge the need of fundamental design. Though the new system provides base for improving the efficiency of operations, there are a lot of future enhancements that can be added to this project. Keeping this in view, a provision has been made in the system to facilities easy modification updating in the future. Any modification will not affect the normal working of the system. It can also be converted into a mobile application

7. REFERENCES

- Web Programming using PHP, Dr. Jeeva Jose, Kalyani Publications.
- www.stackoverflow.com
- www.w3schools.com
- www.gihub.com
- Pamkaj Jalote-An Integerated approach to Software Engineering ,Second Edition, Narosa Publishing Company

8. APPENDIX

8.1 SCREENSHOTS

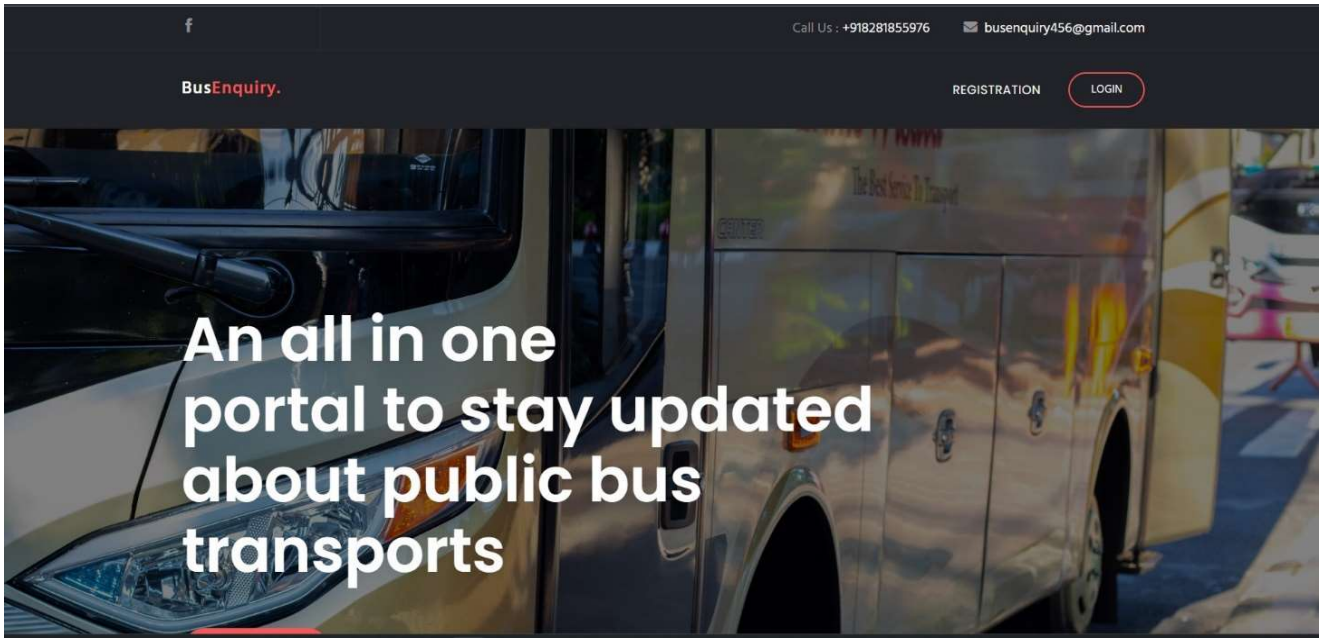


Fig 8.1 Guest Homepage

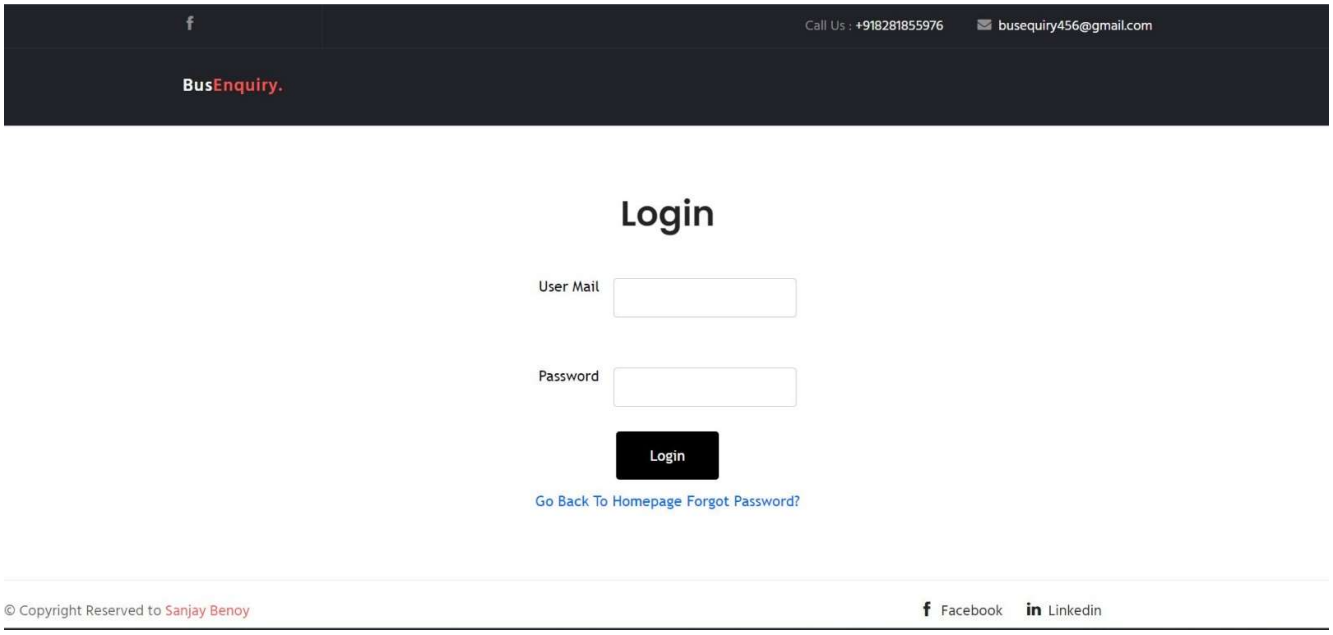


Fig 8.2 Login Page

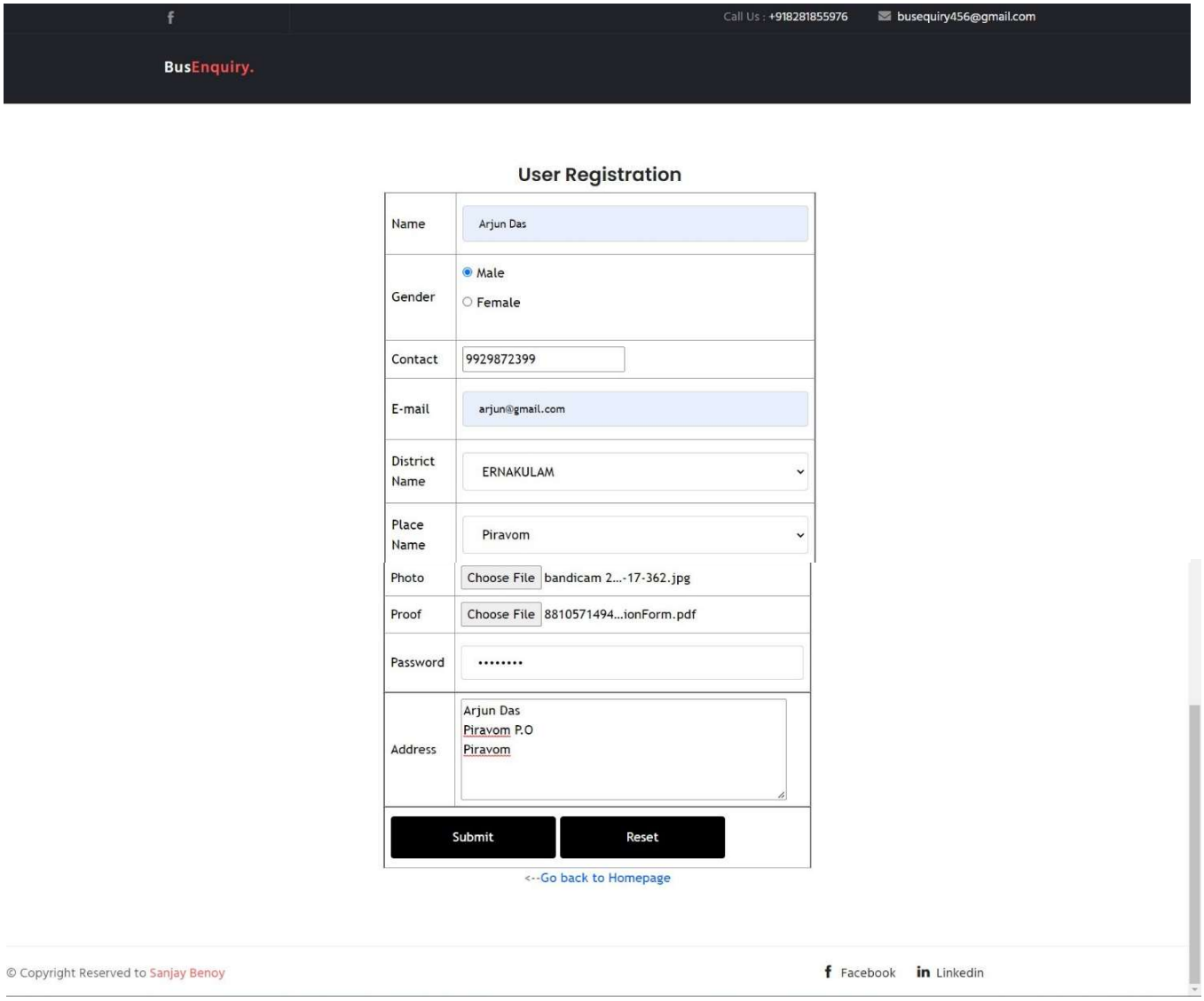


Fig 8.3 User Registration

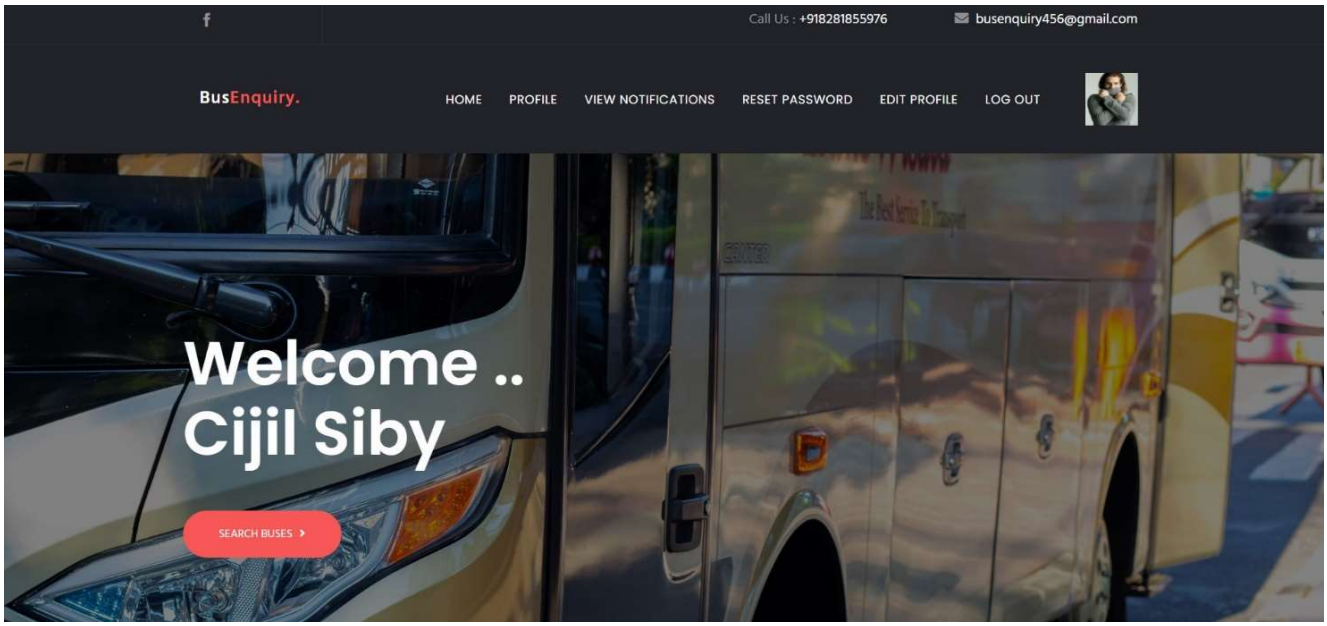


Fig 8.4 User Homepage

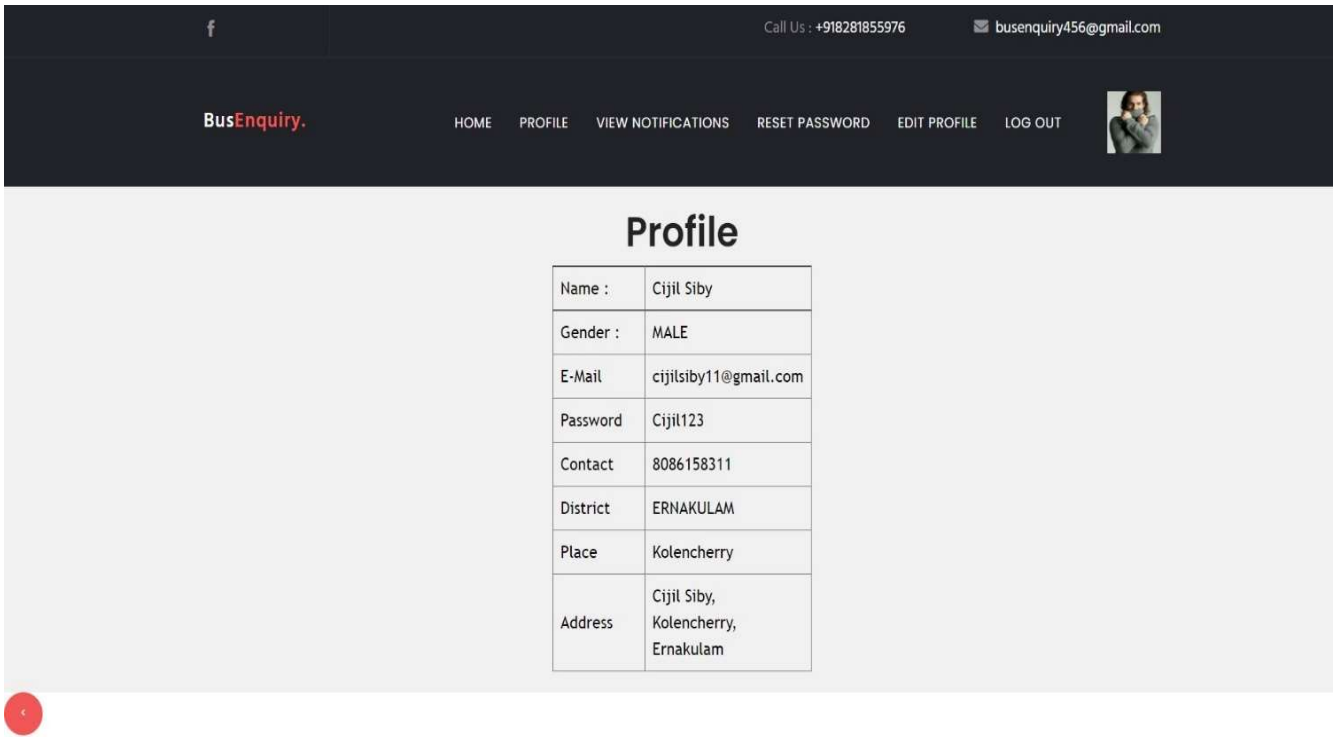


Fig 8.5 User Profile page

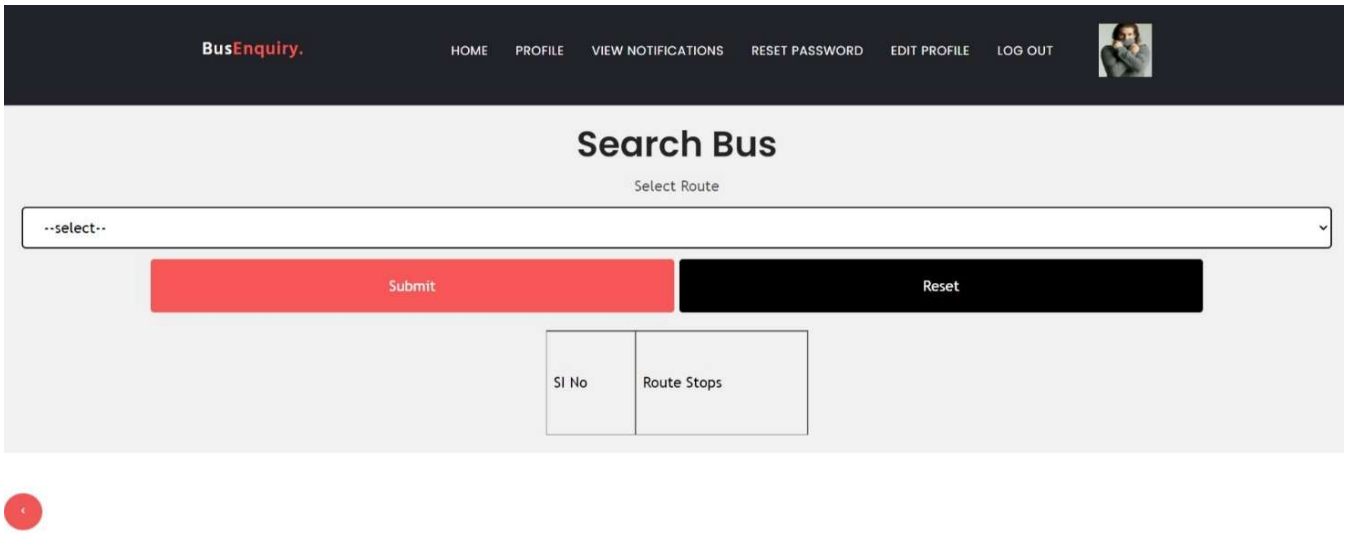


Fig 8.6 User Bus Search Page

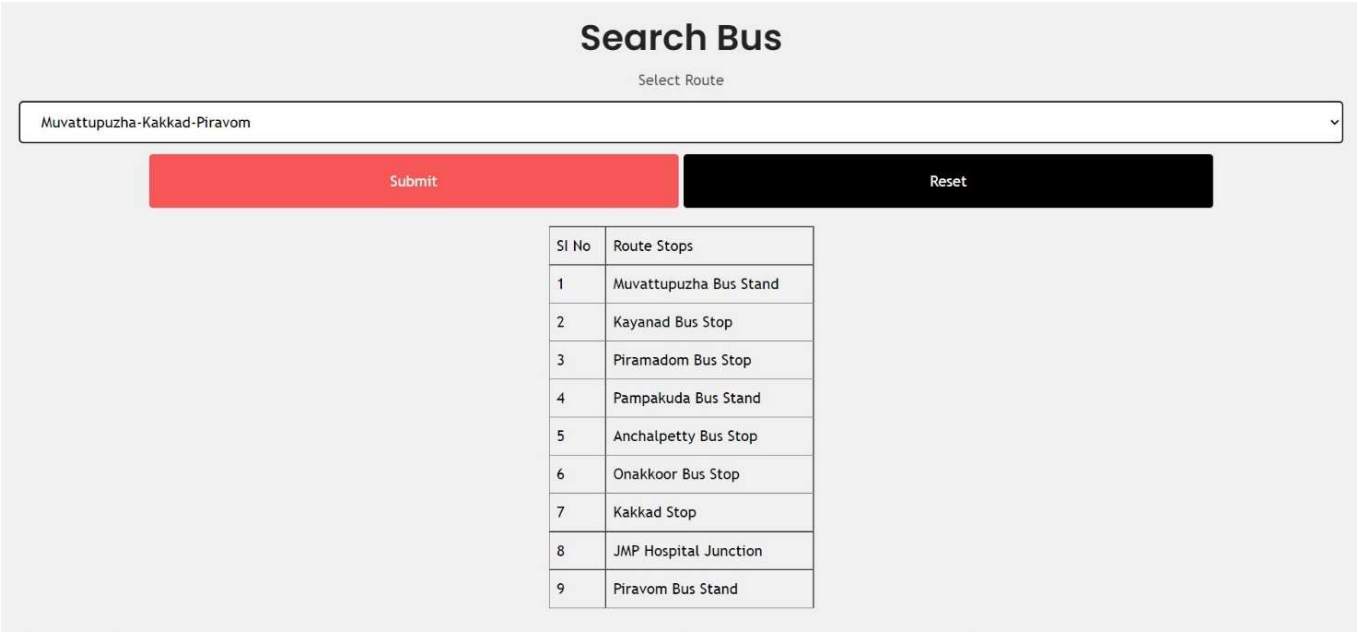


Fig 8.7 Select Route Page

Sl No	Bus Name	Bus type	Route	Reg No	Capacity	Insurance No	Owner Name	Photo	Timings
1	Grace	FAST PASSENGER	Muvattupuzha-Kakkad-Piravom	2235353453	60	2147483647	Sanjay Benoy		View Time & Charges
2	Jesus	FAST PASSENGER	Muvattupuzha-Kakkad-Piravom	534535346343	30	345353433	Sanjay Benoy		View Time & Charges

Fig 8.8 View Buses Page

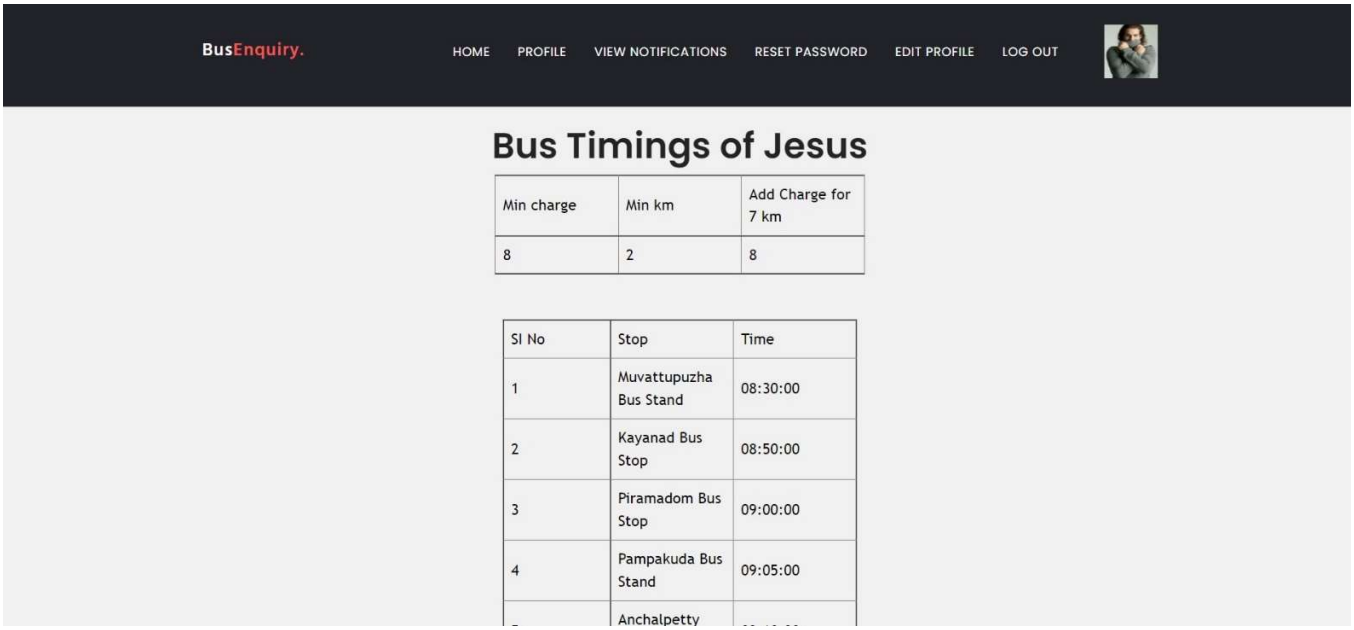


Fig 8.9 View Bus Timings Page

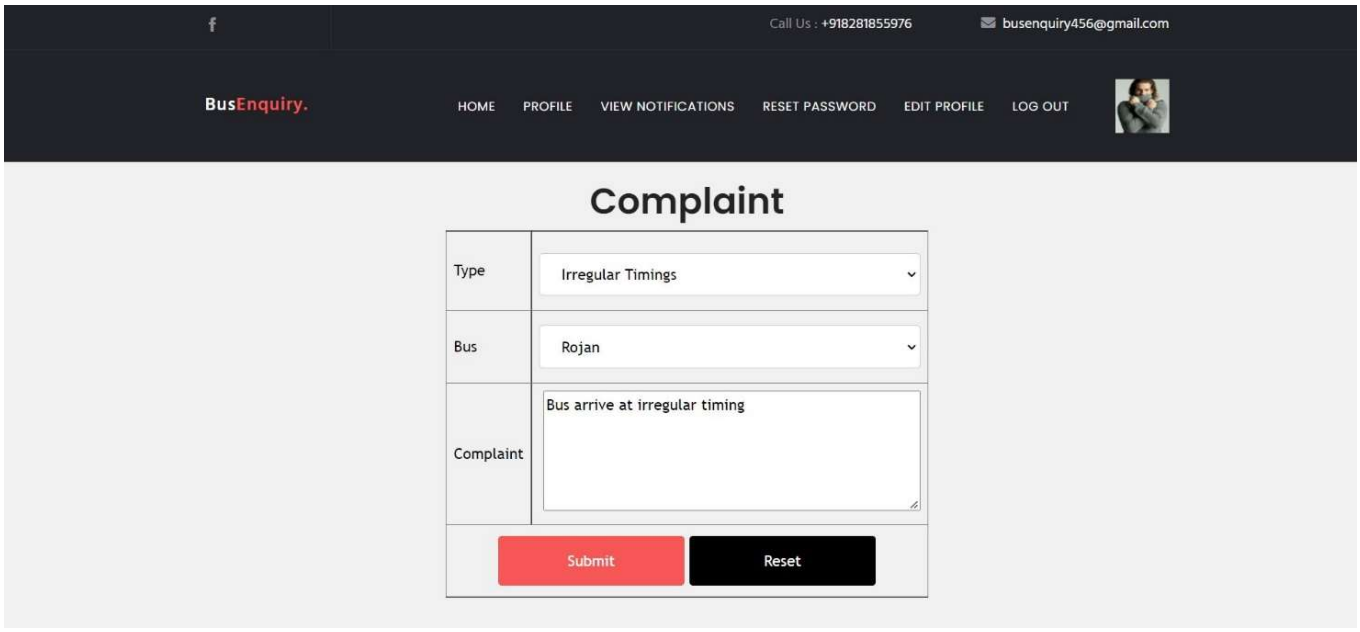


Fig 8.10 Post Complaints

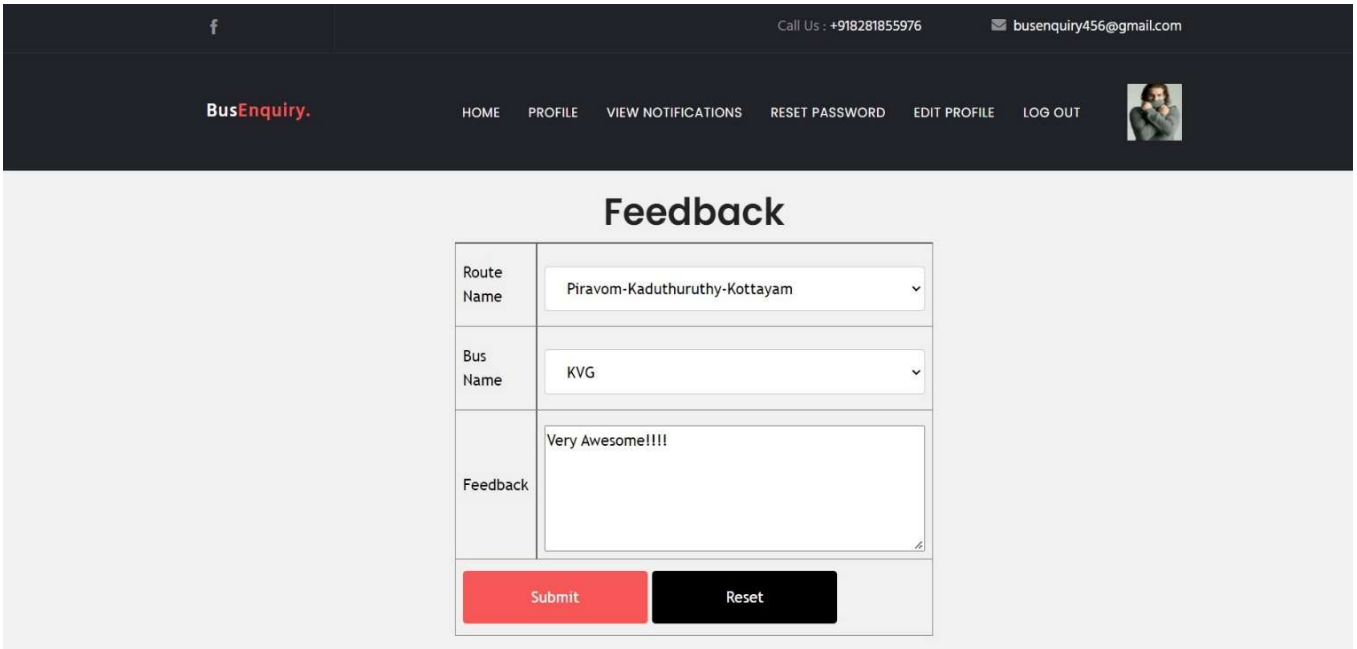


Fig 8.11 Post Feedback

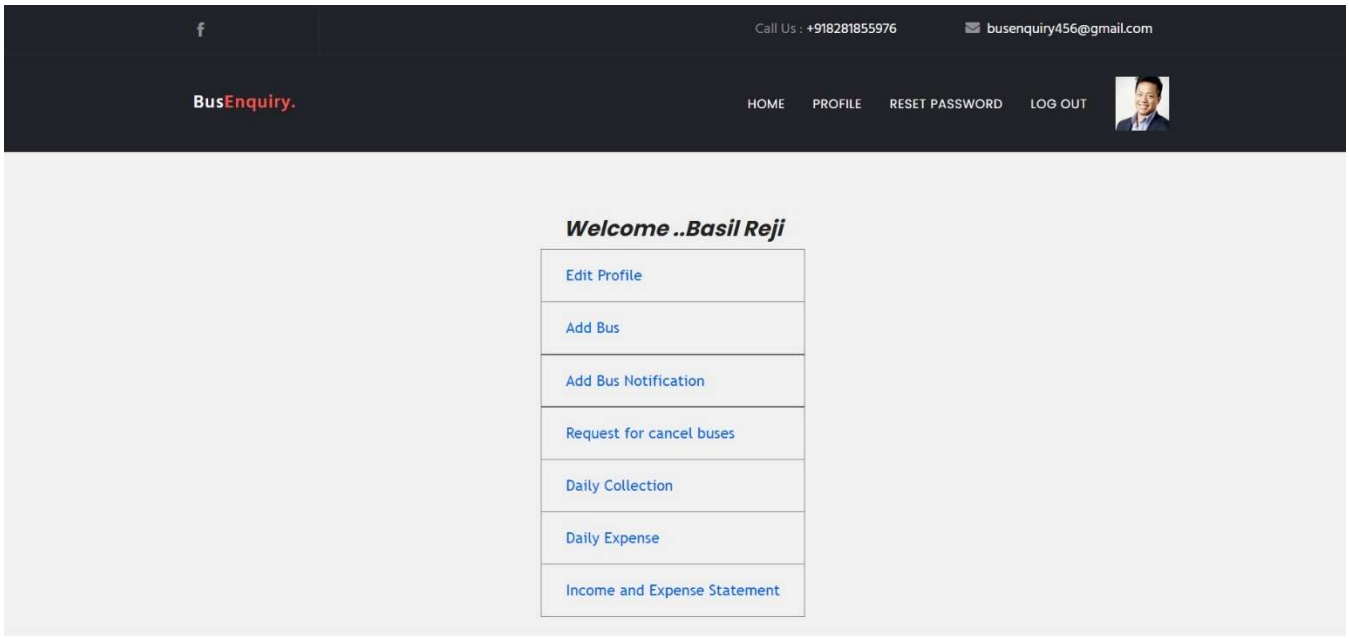


Fig 8.12 Bus Owner Profile Page

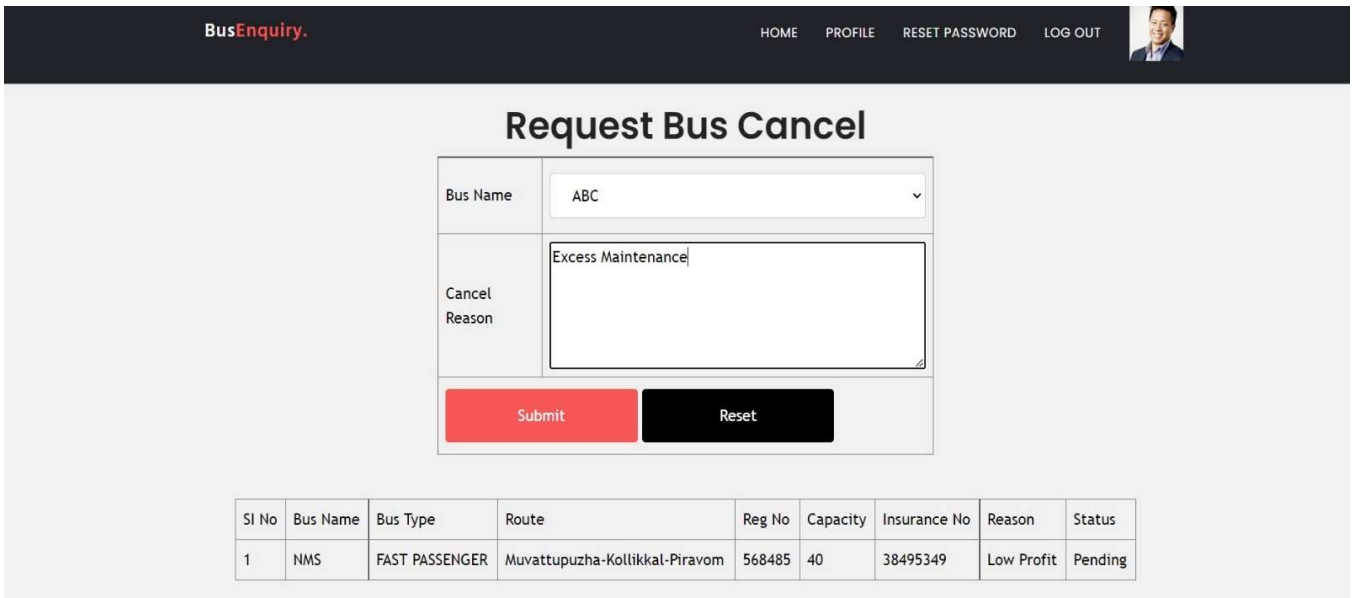


Fig 8.13 Request for Cancelling Buses

Fig 8.14 Admin Homepage

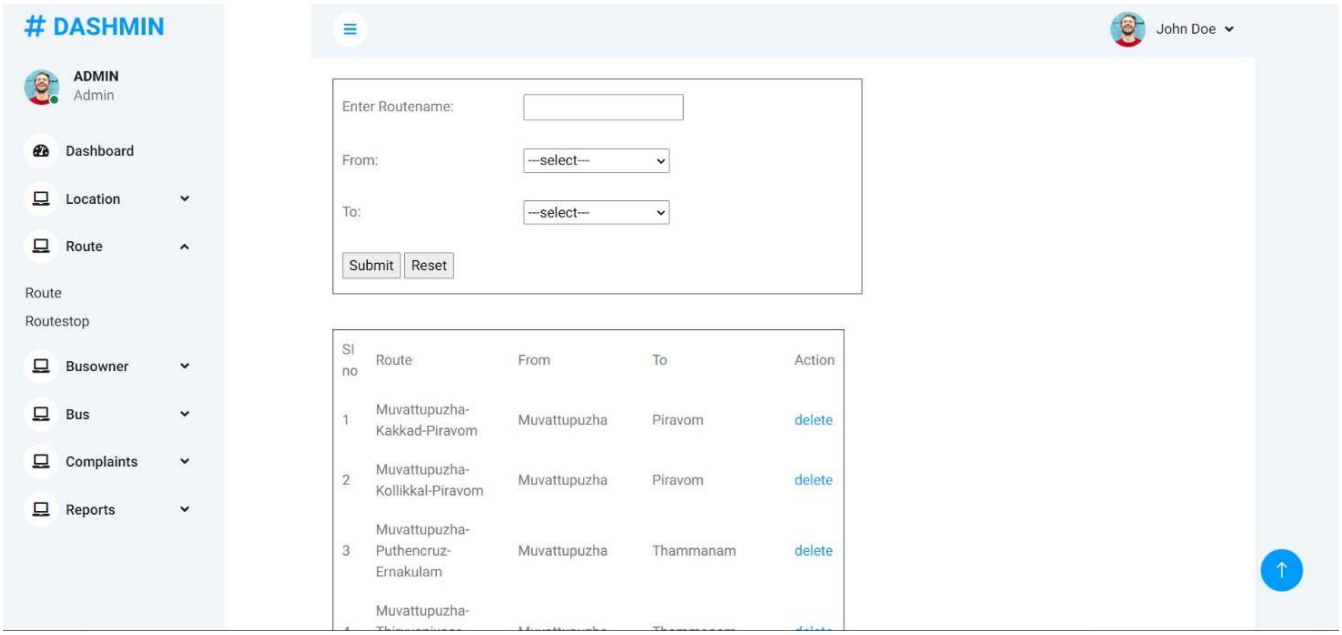


Fig 8.15 Adding Routes

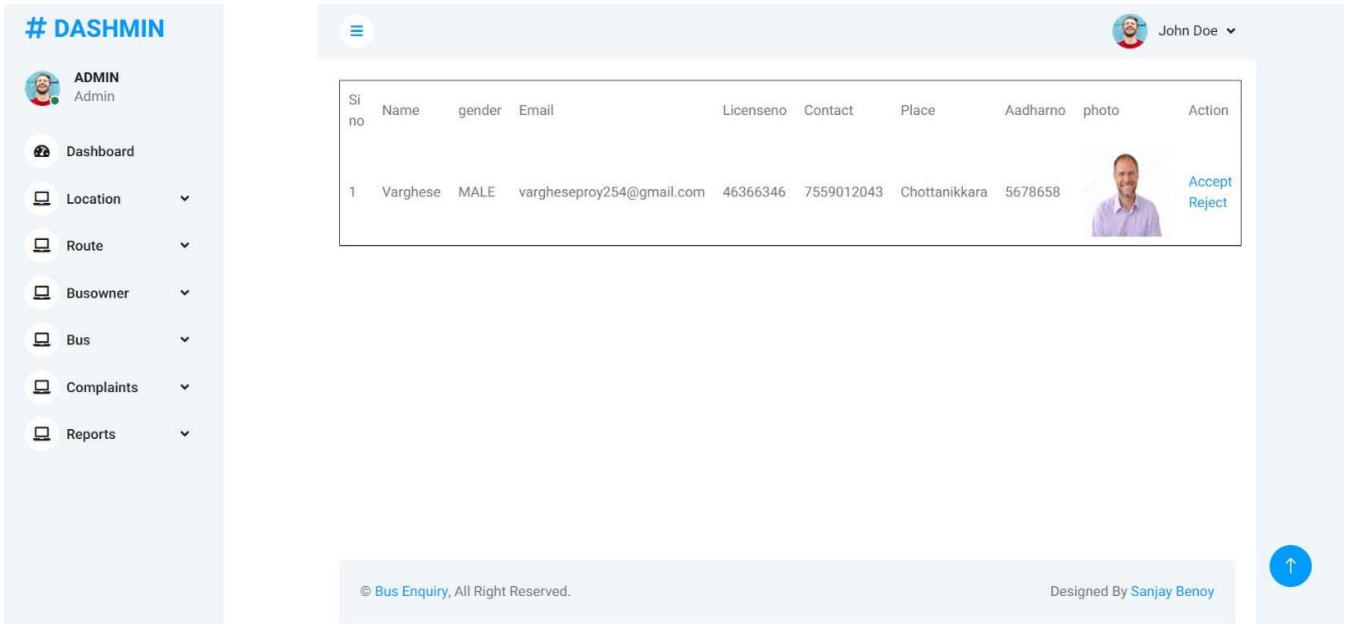


Fig 8.16 Accept/Reject Bus Owners

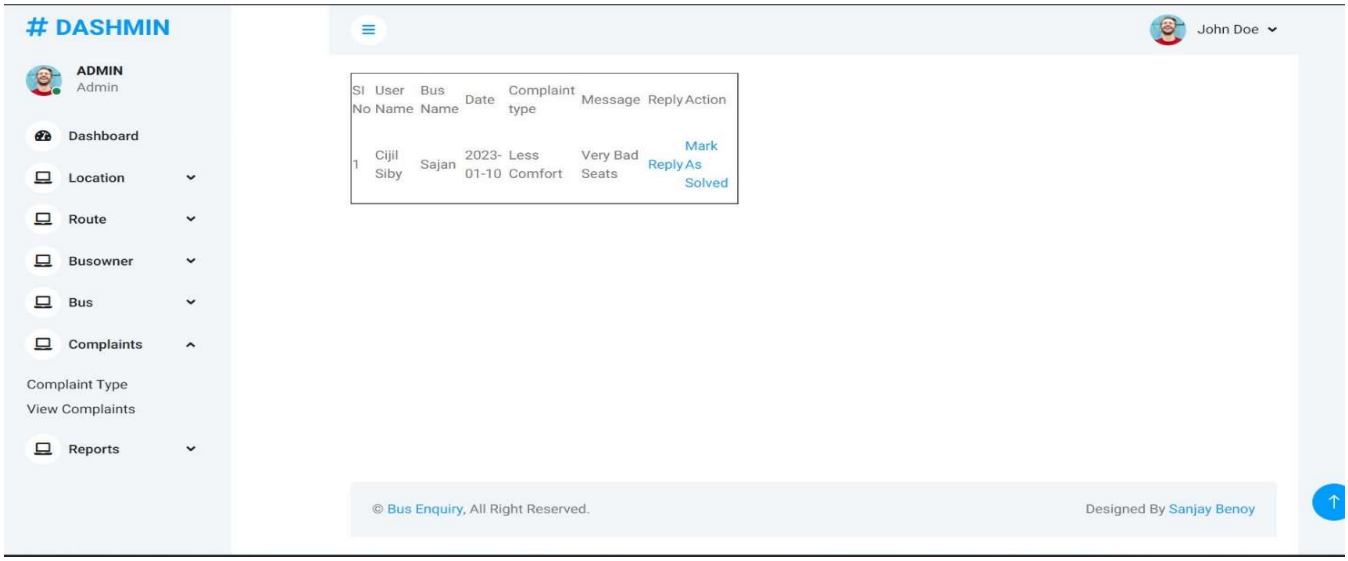


Fig 8.17 View Complaints

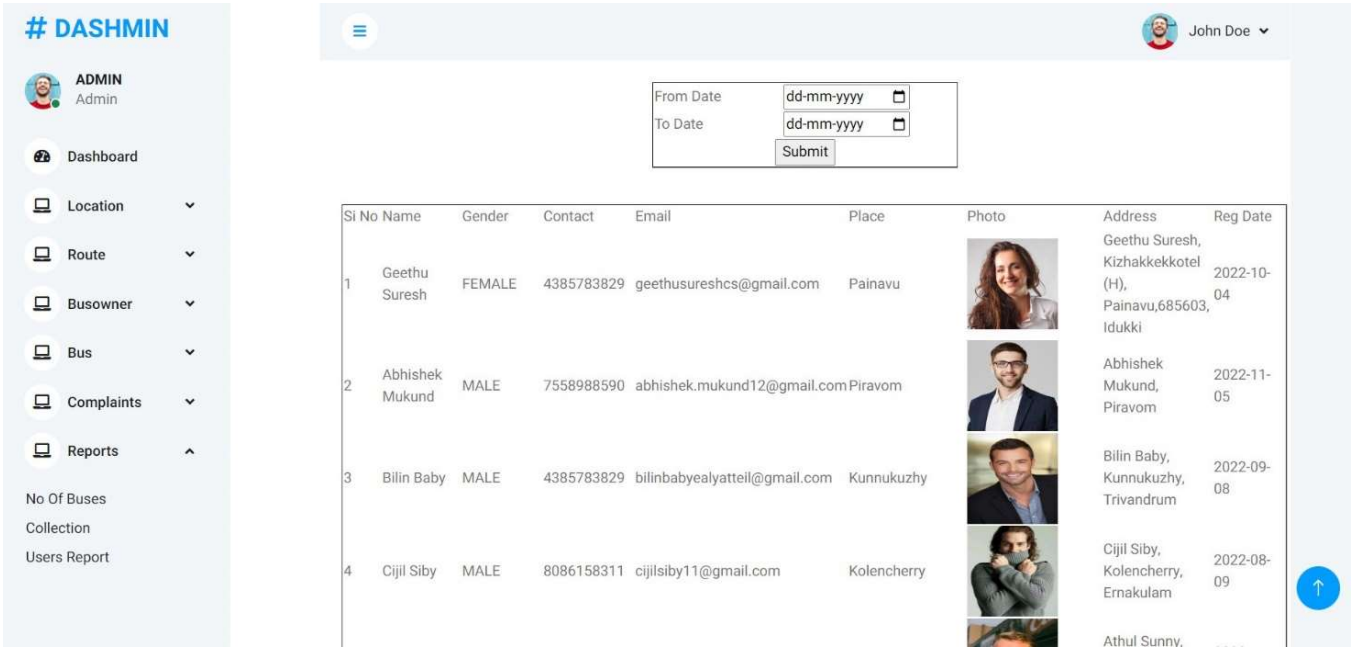


Fig 8.18 View Users report

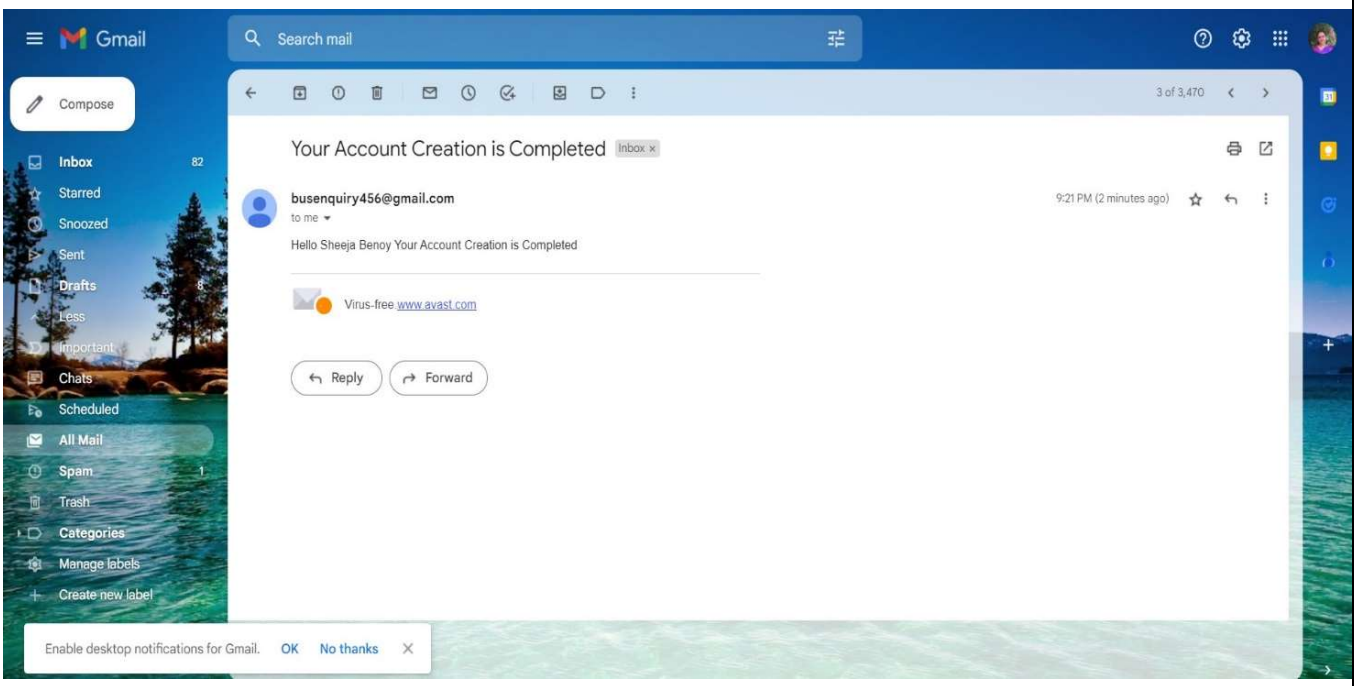


Fig 8.19 Registration Confirmation Mail from User