

**DESIGN AND DEVELOPMENT OF A WEB BASED MULTIPURPOSE
ONLINE TICKET BOOKING SYSTEM**

BY

SH Aidur Rahman

ID: 191-15-2625

AND

Swarna Roy

ID: 191-15-2509

This Report Presented in Partial Fulfillment of the Requirements for the
Degree of Bachelor of Science in Computer Science and Engineering

Supervised By

NAZNIN SULTANA

Associate Professor

Department of CSE

Daffodil International University

Co-Supervised By

AL AMIN BISWAS

Lecturer

Department of CSE

Daffodil International University



DAFFODIL INTERNATIONAL UNIVERSITY

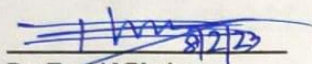
DHAKA, BANGLADESH

6 FEBRUARY 2023

APPROVAL

This Project/internship titled “**DESIGN AND DEVELOPMENT OF A WEB BASED MULTIPURPOSE ONLINE TICKET BOOKING SYSTEM**”, submitted by **Shaidur Rahman** 191-15-2625 and **Swarna Roy** 191-15-2509 to the Department of Computer Science and Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on 6 February, 2023

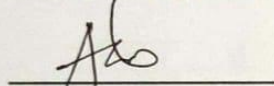
BOARD OF EXAMINERS



Dr. Touhid Bhuiyan
Professor and Head

Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

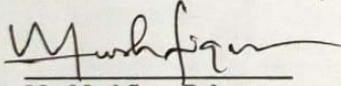
Chairman



Arif Mahmud
Assistant Professor

Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

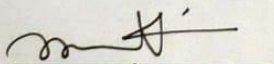
Internal Examiner



Mr. Mushfiqur Rahman
Senior Lecturer

Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner



Dr. Mohammad Shorif Uddin
Professor

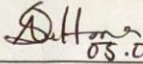
Department of Computer Science and Engineering
Jahangirnagar University

External Examiner

DECLARATION

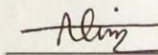
We hereby declare that, this project has been done by us under the supervision of **Naznin Sultana, Associate Professor, Department of CSE** Daffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree or diploma.

Supervised by:

 05.01.2023

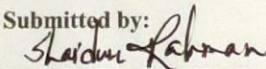
Naznin Sultana
Associate Professor
Department of CSE
Daffodil International University

Co-Supervised by:

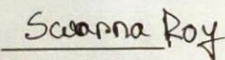
 05.1.2023

Al Amin Biswas
Lecturer
Department of CSE
Daffodil International University

Submitted by:



Shaidur Rahaman
ID: -191-15-2625
Department of CSE
Daffodil International University



Swarna Roy
ID: -191-15-2509
Department of CSE
Daffodil International University

ACKNOWLEDGEMENT

First we express our heartiest thanks and gratefulness to almighty God for His divine blessing makes us possible to complete the final year project/internship successfully.

We really grateful and wish our profound our indebtedness to **Naznin Sultana, Associate Professor**, Department of CSE Daffodil International University, Dhaka. Deep Knowledge & keen interest of our supervisor in the field of “*Web Based*” to carry out this project. His endless patience ,scholarly guidance ,continual encouragement , constant and energetic supervision, constructive criticism , valuable advice ,reading many inferior draft and correcting them at all stage have made it possible to complete this project.

We would like to express our heartiest gratitude to Pro. Dr. Touhid Bhuiyan, and Head, Department of CSE, for his kind help to finish our project and also to other faculty member and the staff of CSE department of Daffodil International University.

We would like to thank our entire course mate in Daffodil International University, who took part in this discuss while completing the course work.

Finally, we must acknowledge with due respect the constant support and patients of our parents.

ABSTRACT

The project's goal is to purchase bus, air, movie tickets online. Anyone with a web connection can access the ticket reservation system, which is an Internet-based program, from anywhere in the world. The tickets will be held using this application. Any internet user can access the website for a movie theater provided by this online ticket reservation system. A credit card is necessary to log into the system and to purchase tickets. At the counter, tickets can be obtained .One of the best forms of entertainment after a busy day is going to the movies with family and friends. However, all of this excitement quickly fades after waiting for hours in a line to get tickets. . The website offers comprehensive facts about movies that are currently playing on all screens, including show times and seat availability. Reservations for tickets are made with a credit card and are cancellable if necessary. One of the finest options for people who can't afford the time to reserve their tickets while waiting in long lines is our online system. Online ticket purchasing is available day or night.

TABLE OF CONTENTS

CONTENTS	PAGE
Approval	i
Board of examiners	ii
Declaration	iii
Acknowledgements	iv
Abstract	v

CHAPTER 1: INTRODUCTION

1. Introduction	1
2. Motivation	2
3. Objective	2
4. Expected Outcomes	3
5. Project Management and Finance	3
6. Report Layout	4

CHAPTER 2: BACKGROUND

2.1 Preliminaries / Terminologies	5
2.2 Comparative Analysis	5
2.3 Scope of The Problem	5

CHAPTER 3: REQUIREMENT SPECIFICATION

3.1 Requirement Management	6
3.2 Requirement Collection	6
3.3 Visual Studio Code	6
3.4 PHP	7
3.5 Xampp	7
3.6 Use Case Modeling and Description	8
3.7 Use Case Diagram	8
3.8 Design Requirements	9

CHAPTER 4: DESIGN SPECIFICATION

4.1 Front-end design	10
4.2 Back-end design	10
4.3 User Login & registration	11
4.4 Homepage	12
4.5 User Sign In	13
4.6 Show User Profile	14
4.7 Update User Profile	14
4.8 Interacting Design and User Experience (UX)	22

CHAPTER 5: IMPLEMENTATION AND TESTING

5.1 Introduction	23
5.2 Implementation of database	29
5.3. Testing Implementation	30
5.4 Implementation of Front-End Design	31
5.5 Test Results and Reports	31

CHAPTER 6: IMPACT ON SOCIETY, ENVIRONMENT AND SUSTAINABILITY

6.1 Influence of Society	32
6.2 Effects on the Environment	32
6.3 Ethical Aspects	32
6.4 Sustainability	32

Chapter 7: CONCLUSION AND FUTURE SCOPE

7.1 Discussion and Conclusion	33
7.2 Scope for Future Development	33

REFERENCES	34
-------------------	----

PLAGIARISM REPORT

LIST OF FIGURES

FIGURES NAME	PAGE NO
Figure 3.7 Use Case Diagram	8
Figure 4.3 User Login & registration	11
Figure 4.4. Homepage	12
Figure 4.5. User Sign in	13
Figure 4.6 Show User Profile	13
Figure 4.7 Update User Profile	14
Figure 4.8.1 Nevbar/menubar	15
Figure 4.8.2 Booking page	15
Figure 4.8.3 Service Page	16
Figure 4.8.4 Contact Page	16
Figure 4.8.5 Order Page	17
Figure 4.8.6 Payment Page	19
Figure 4.8.7 Manage Order Page	20
Figure 4.8.8 Manage Payment Page	21
Figure 4.8.9 Manage Contact Page	22
Figure 5.2 User Profile Controller Page	23
Figure 5.2 All Order Page	29
Figure 5.2 All Payment Page	29

LIST OF TABLES

TABLE	PAGE
Log in Pre-condition/Register	30

CHAPTER 1

INTRODUCTION

4.4 Introduction

In the contemporary technological era, the Internet of Things (IOT) has experienced tremendous global growth in popularity. Almost all “smart” devices have internet connections, allowing them to access information from anywhere in the world. Once upon a time, people would waste their valuable time attempting to learn anything. Modern technology is more advanced than technology from prior eras. Technology has blessed us with web applications. Users can access the system from any location as long as they have an internet connection. Typically, the ticket counters for buses, flights, and movie theaters preserve paper records of their customers’ information and reservations. Any customer who wants to reserve a seat must call or visit the counter, which is considered to be a waste of their important time. Sometimes consumers can’t reserve seats over the phone because it’s always busy. In addition, it is difficult for them to maintain track of payment concerns because they must preserve paper records of consumer payments. We now have plans to replace their outdated booking system with a new, web-based one. We therefore intend to put in place an online web-based bus, flight ticketing system that will make it simpler for clients to make reservations from home and abroad and for them to run their company generally. The system I’m going to talk about in this instance is called “Web Based Online Ticket Booking System” and it is entirely a web application. This web application can connect to the appropriate servers for data access since, as we previously established above, the internet has made it simpler for users to interface with the system. This will undoubtedly let users buy tickets or book seats online without waiting in line. In addition, more recently, people have begun to travel to get away from their mundane lifestyles. As a result, they want to travel without any hesitations. They are looking for a solution that will enhance portability, usability, and accessibility in the modern technological era. As previously stated, we will build a web system with all the capabilities necessary to increase its usability and accessibility.

We'll outline the underlying presumptions, key characteristics, and the UML design process for the system below (Uniform Modelling Language). Along with that, we'll discuss the system's advantages and disadvantages (limitations) as well as some screenshots of my projects.

1.2 Motivation

The development of information technology and the widespread use of the internet have greatly enhanced many business operations and relationships between companies (service providers) and their clients, who are not excluded from the manual ticket booking system sector. This online ticket ordering system was created to offer the following services:

1. Enhance business processes: must have the ability to digitally project the organization. Rental companies are expanding their return on investment (ROI) to the worldwide market rather than limiting their services to their local region solely.
2. Online Vehical Booking : a process that allows consumers to reserve available cars online before their planned date or time of pickup.
3. Restriction Of the Customer : a platform for registration that stores user information, tracks their transactions, and uses it to improve services and grow the customer base.

1.3. Objective

1. The project staff and employees of the ticket booking agency with low throughput will be able to book tickets more effectively thanks to this method.
2. This system project has been designed to be as user-friendly as feasible so that anyone with a basic understanding of system computers can use it.

3. By storing all the project information for bus ticket booking and ticket cancellation in the form database on the computer's hard drive, the ticket booking project will lessen the laborious task of system paperwork related to ticket booking.
4. Information about the current state of the system's performance and other inquiries.
5. We provide up to date information that is not possible manually.
6. The goal of my project is to make the Ticket Booking Agency's ticket booking system straightforward, dependable, approachable, and corrective. Additionally, it takes less time than physical labor.
7. Different components of the ticket booking report can be produced, including the seats that are available, the booking's status right now, cancellation information, a monthly status report, etc.

Expected Outcome

It will be simpler for customers to book through our website. With the use of this system, customers can confirm their bookings by making manual or online payments. They will gain time as a result. The main objective of this project is to automatically complete the manual processes needed to reserve an airline, bus, or movie ticket for any trip made via this website.

Project Management and Finance

As a business owner, it might be difficult to run a lodging operation. To keep our business running well, in addition to interacting with our personnel and clients, we also need to take care of administrative responsibilities. Taking and handling reservations for our firm is one of the most crucial of these tasks. However, it might be difficult to keep track of all business appointments and bookings.

Report Layout

Chapter 1: The project's introduction, aim, purpose, and projected consequences have all been made explicit up front.

Chapter 2: This chapter provides background information.

Chapter 3: This chapter displays a flowchart and use case modeling.

Chapter 4: The requirements for the front-end, back-end, and implementation may be found.

Chapter 5: The front-end design, test, and database implementation were all describe

Chapter 6: Sustainability and the Social Impact on Society were discussed in this chapter.

Chapter 7: The extent of our project and its potential for growth to purchase our api.

CHAPTER 2

BACKGROUND

2.1 Preliminaries/Terminologies

We chose Laravel [1] over rival web frameworks because of its powerful features and development tools, which speed up the construction of online applications. We made an effort to make the entire product responsive and user-friendly so that it could be utilized on any device.

2.2 Comparative Analysis

The online application is designed so that even those without prior computer experience can use it without difficulty. This online application is simple to use and doesn't need a technical expert to utilize.

2.3 Scope of the Problem

- Adaptive demand
- Visibility is restricted
- Manual Records
- Complexity of the Supply Chain

CHAPTER 3

REQUIREMENT SPECIFICATION

3.1 Requirement Management

A system or component's functional requirements define it. The user determines the functionality that is required. It is essential to follow these guidelines. It is included in the use case. The following are the functional criteria for the "Online Booking System": Registration, logging in, choosing a bus, a bus date, a seat, a ticket date, adding payment, and entering all relevant information are required.

3.2 Requirement Collection

- Visual Studio Code for development.
- JavaScript, PHP for language.
- Html, CSS and Bootstrap for as styling.
- Xampp for database.

3.3 Visual Studio Code

On Windows, Linux, and macOS, you can utilize the source code editor called Visual Studio Code by Microsoft. A few of the features include integrated Git, syntax highlighting, intelligent code completion, snippets, code refactoring, and debugging tools. Users can change the extension's additional functionality, keyboard shortcuts, preferences, and theme.

3.4 PhP

The acronym PHP stands for “Hypertext Pre-processor.” HTML incorporates the Web programming language PHP. This suggests that a Web page’s HTML may contain PHP code. The server that hosts a PHP page reads, or “parses,” the PHP code each time a user browses the page. Usually, the website’s PHP functions result in HTML code that the browser can understand. Before a website loads, PHP code is converted into HTML, rendering it hard for people to read. PHP pages may therefore access databases and other sensitive information without compromising their security. The syntax of other languages like C, Java, and Perl had a big impact on PHP. PHP, on the other hand, has a variety of unique characteristics.

3.5 Xampp

The free and open-source XAMPP package, developed by Apache Friends, includes the Apache HTTP Server, MariaDB, and interpreters for PHP and Perl scripts. The majority of commercial web server installations use the same parts as XAMPP, therefore switching from a local test server to a live server is not difficult. A developer can quickly and efficiently build a WAMP or LAMP stack on an operating system with the help of well-known add-on programs like WordPress and Joomla thanks to the straightforward implementation of XAMPP. They can also be set up using Bitnami with the same ease.

3.6 Use Case Modeling and Description

In order to solve a problem, different user types interact with a technology, as shown by a use-case[3] model. As a result, it describes the users' goals, how they interact with the system, and the steps the system needs to take to achieve these goals.

3.7 Use Case Diagram

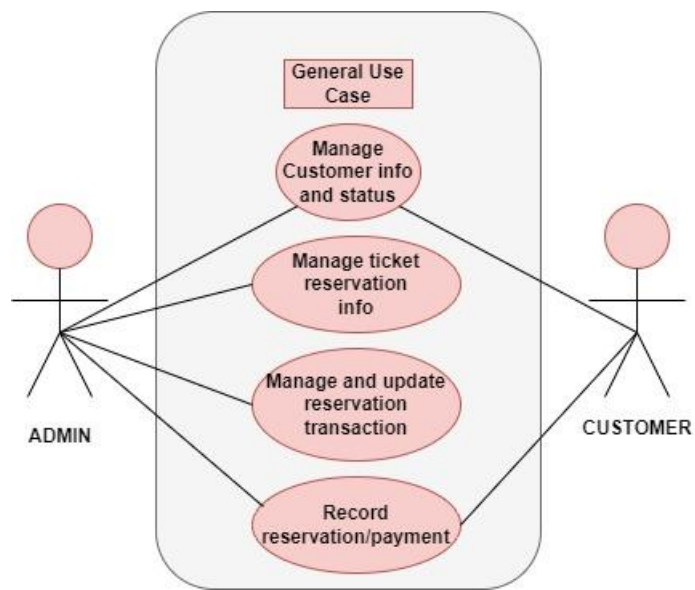


Figure 3.7 Use Case Diagram

3.8 Design Requirement

We utilized Visual Studio Code as our main code editor for this project. These days, Visual Studio Code is a well-liked code editor. We utilized the open-source database MySQL for my personal needs. It aids in the development of a responsive design. We used the PHP-based Laravel framework for our project's backend, and we utilize Livewire, a full-stack framework for Laravel that facilitates the creation of dynamic interfaces without requiring Laravel to stray from its strengths. It is a package that makes using Laravel Blade as a templating language to create contemporary, reactive, dynamic interfaces straightforward. we've used several Laravel packages that have enabled us to build out a range of functionalities for our project. Moreover, we utilized JavaScript to carry out some operations.

CHAPTER 4

DESIGN SPECIFICATION

4.1 Front-End Design

HTML : Each page contains a number of hyperlinks, or connections to other pages. HTML offers a helpful explanation of how a web page is organized. The most essential component of the Web is (HyperText Markup Language). It explains how online material is structured and used. Technology other than HTML is frequently used to describe the presentation and functionality of a web page (CSS and JS) (JavaScript). For example, the HTML element “p” identifies a paragraph. Front-end [2] developers use the HTML element to specify the type of information that each item on a webpage possesses. Developers also build HTML language code to explain the connections between different elements in a page’s overall structure or document structure.

CSS: The language used to describe how Web pages are presented, including their colors, design, and fonts, is called CSS. It enables the presentation to be adjusted for use on many sorts of devices, including printers, tiny screens, and huge screens. CSS can be used with any XML-based markup language and is not dependant on HTML. We also use bootstrap.

4.2 Back-End Design

PHP: PHP is a server-side programming language with HTML integration. It may be used to build full e-commerce websites and manage dynamic content, databases, and session monitoring. It is connected to many well-known databases, including Microsoft SQL Server, MySQL, PostgreSQL, Oracle, Sybase, and Informix. The most popular open source server-side programming language for creating dynamic websites and mobile apps is called PHP [5]. PHP stands for hypertext preprocessor.

Framework for LARAVEL: The Taylor Otwell-created and Symfony-based free and open-source Model-view-controller (MVC) web applications should be made using the Laravel [4] framework.

Laravel promises to make development easier by automating typical activities required in the majority of online projects, such as authentication, routing, sessions, and caching. Laravel makes an effort to improve the developer experience while maintaining the application's functionality.

4.3 User Login & registration : This is a screenshot of the registration and login screens for users and customers. First User must register, log in, and then access to view home page.

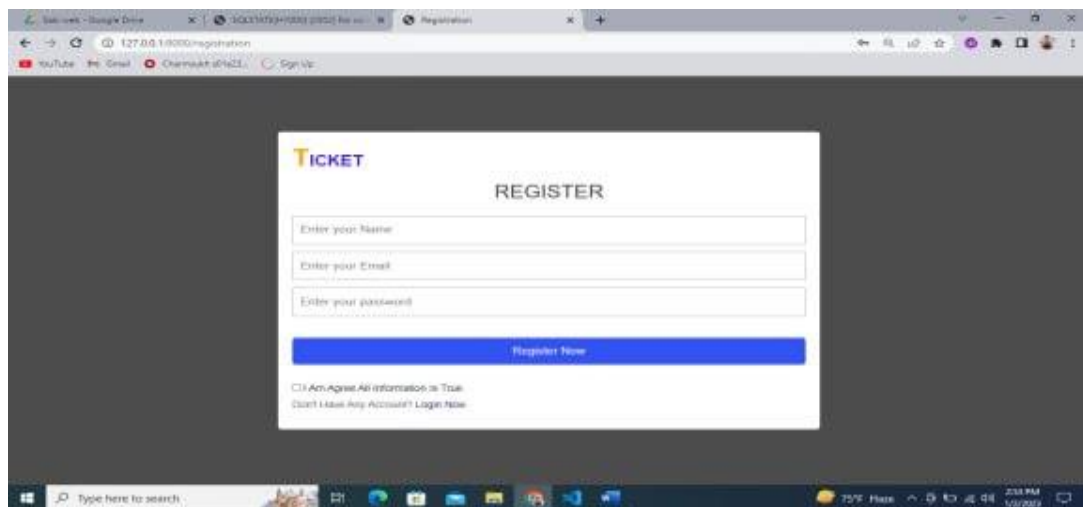


Figure 4.3 User Login & registration

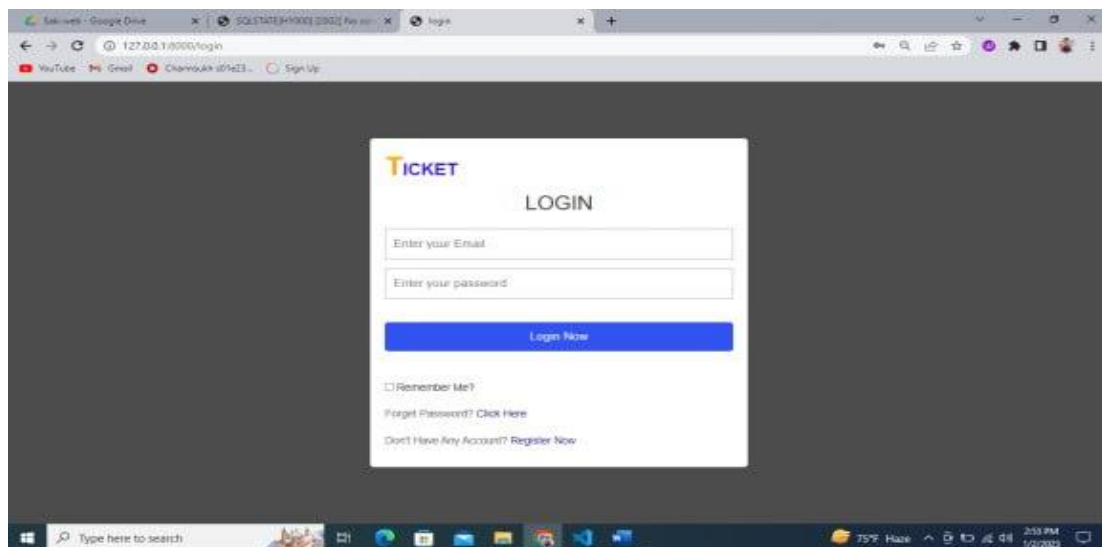


Figure 4.3 Login page

4.4 Homepage Here, after logging in, the user can access their home page. On the homepage, a succinct summary of our website is provided.

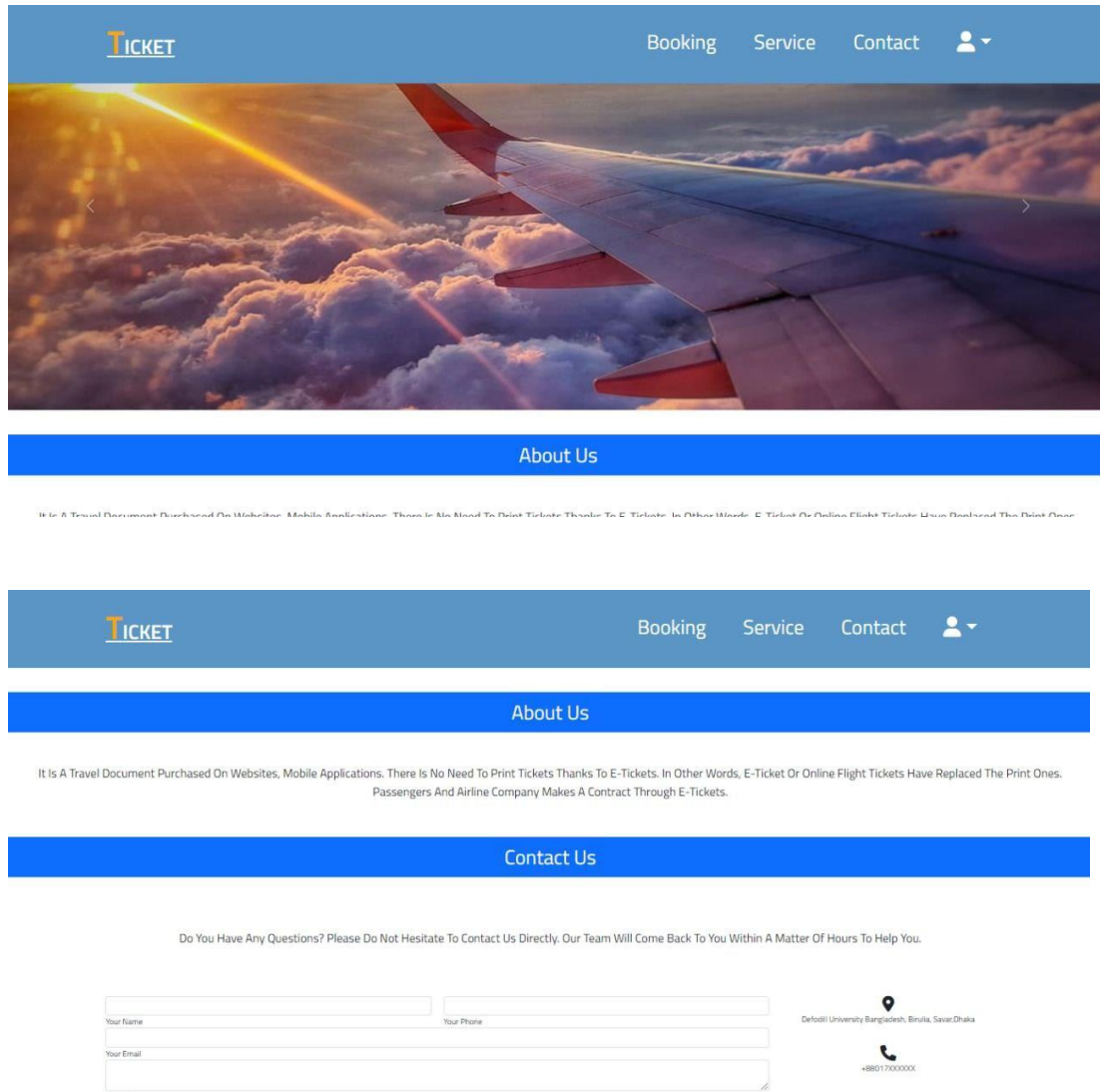
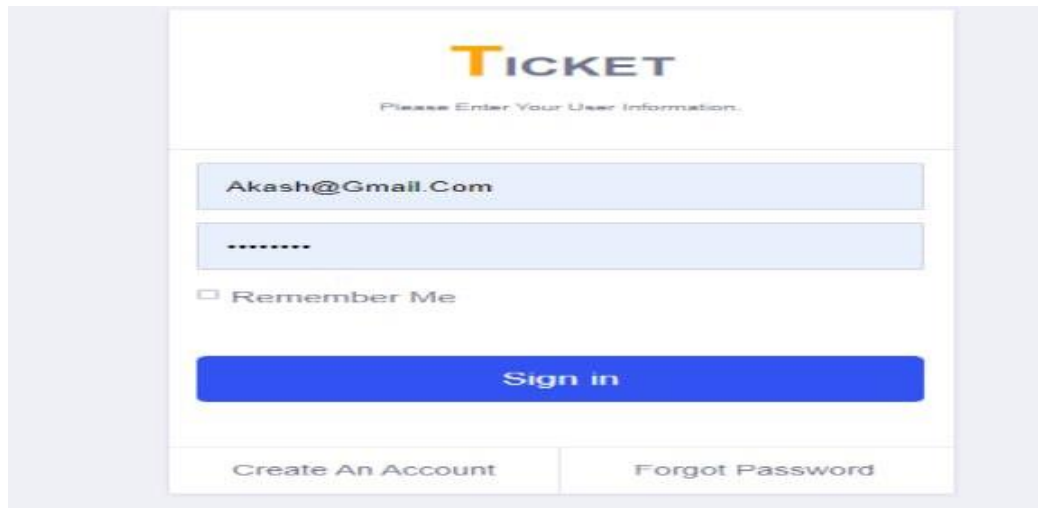


Figure 4.4 Homepage

4.5 User Sign In

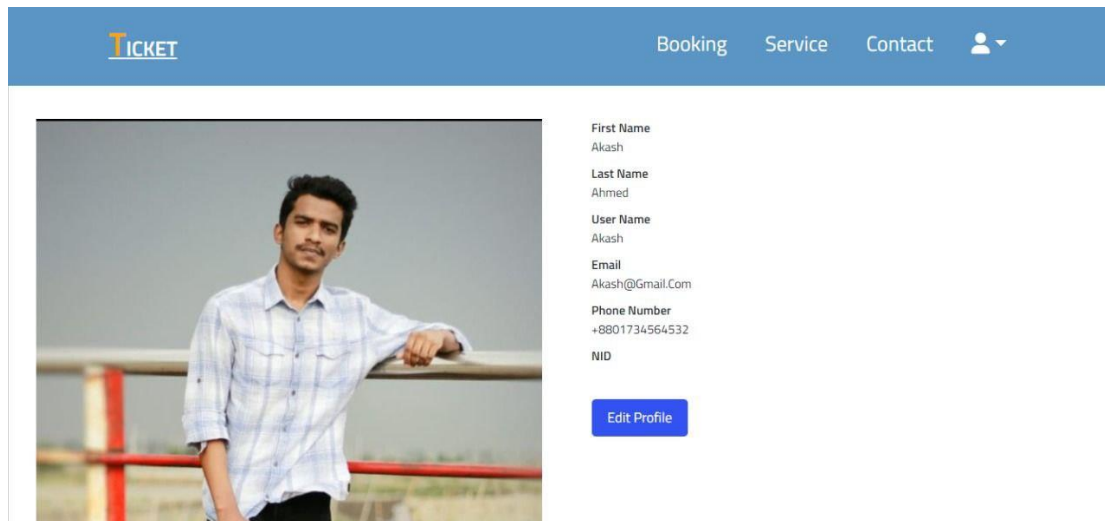
The user sign-in page is located here. Here, the administrator will sign in. Fix for administrative data kept in databases. As a result, registration is not required, and access is not permitted without it.



The screenshot shows a sign-in form for the 'TICKET' system. At the top, the word 'TICKET' is displayed in a large, bold, orange font. Below it, the text 'Please Enter Your User Information.' is centered. The form contains two input fields: the first is for the email address, containing 'Akash@Gmail.Com', and the second is for the password, shown as a series of dots. Below the password field is a checkbox labeled 'Remember Me'. A prominent blue button labeled 'Sign in' is positioned below the form. At the bottom of the form, there are two links: 'Create An Account' and 'Forgot Password'.

Figure 4.5 User Sign In

4.6 Show User Profile



The screenshot displays the user profile page for the 'TICKET' system. The page has a blue header with the 'TICKET' logo on the left and navigation links for 'Booking', 'Service', and 'Contact' on the right, along with a user profile icon. The main content area features a profile picture of a man in a light blue plaid shirt leaning on a railing. To the right of the photo, the user's details are listed: First Name (Akash), Last Name (Ahmed), User Name (Akash), Email (Akash@Gmail.Com), Phone Number (+8801734564532), and NID. A blue 'Edit Profile' button is located below the user information.

Figure 4.6 Show User Profile

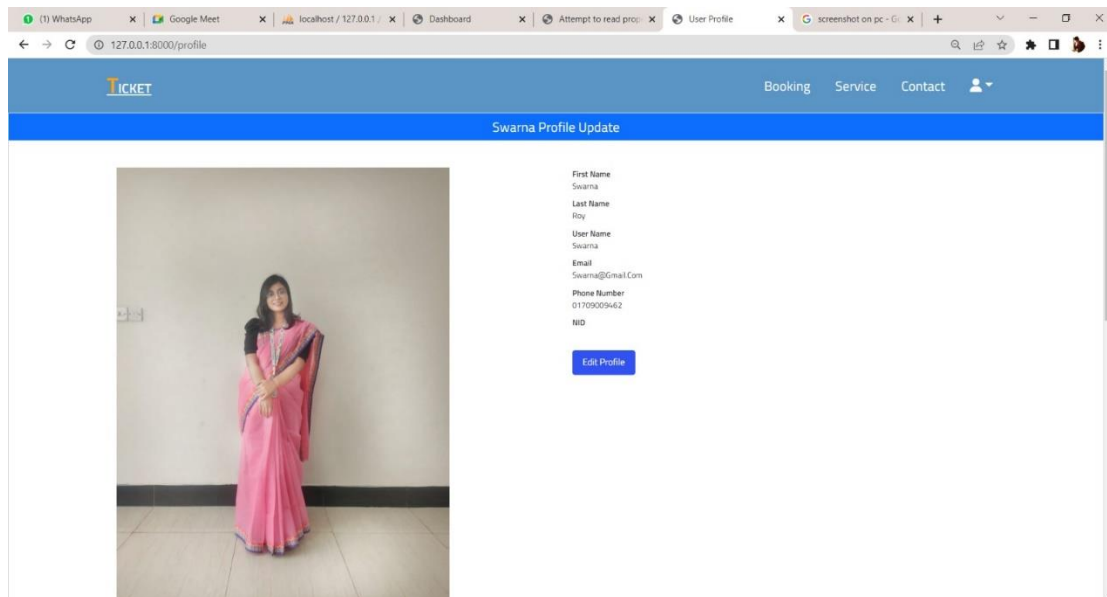


Figure 4.6 Show User Profile

4.7 Update User Profile

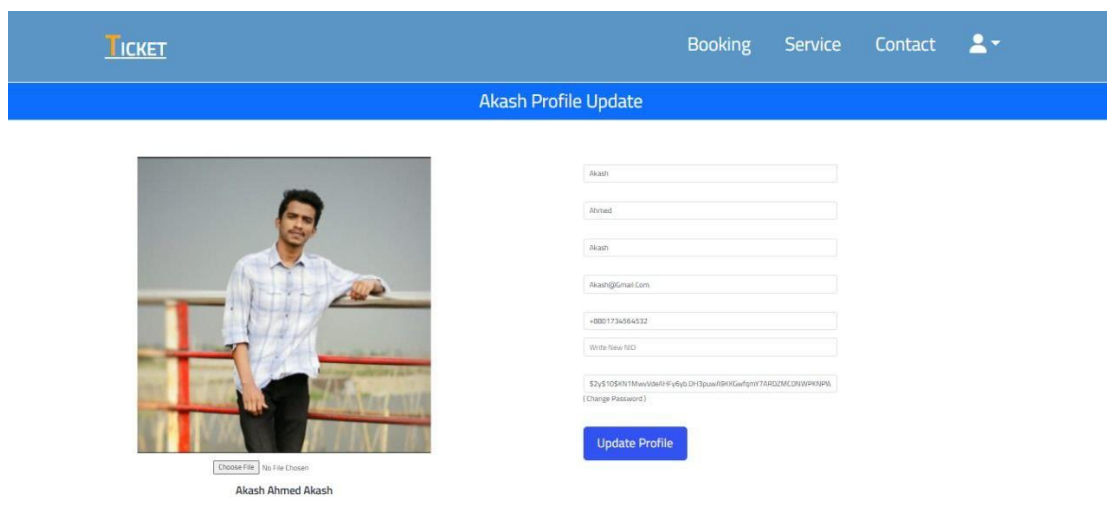


Figure 4.7 Update User Profile

4.8 Interacting Design and User Experience (UX)

Navbar/Menu bar: There is a navigation bar on our home page. Bookings, services, contacts, and users are all listed on this menu.

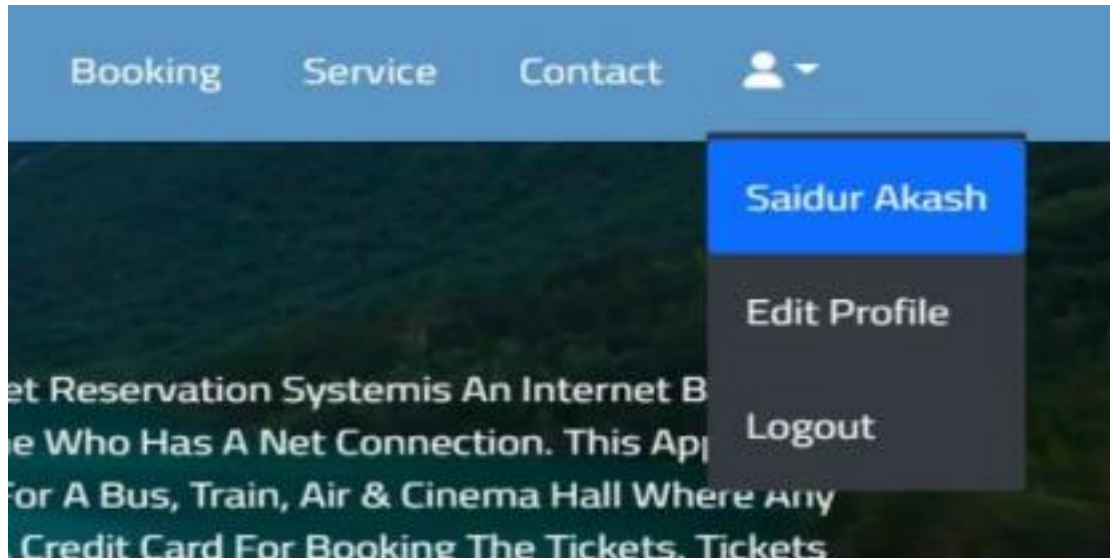


Figure 4.8.1 Navbar/Menubar

Booking : This will take you to our reservations page. The user can select their destination right here.



Figure 4.8.2 Booking Page

Service : The user can select his services after selecting the Service menu. Hovering the cursor over any service, such as a bus, an aircraft, or a movie, allows users to see the sort details. Any service's card can be clicked to access the order page.



Figure 4.8.3 Service Page

Contact Page : We can reach us through that page. Users can provide feedback and grievances here



Figure 4.8.4 Contact Page

Order Page : There are three order pages available. There are booking options for buses, flights, and movies. The order page for each specific service is distinct.

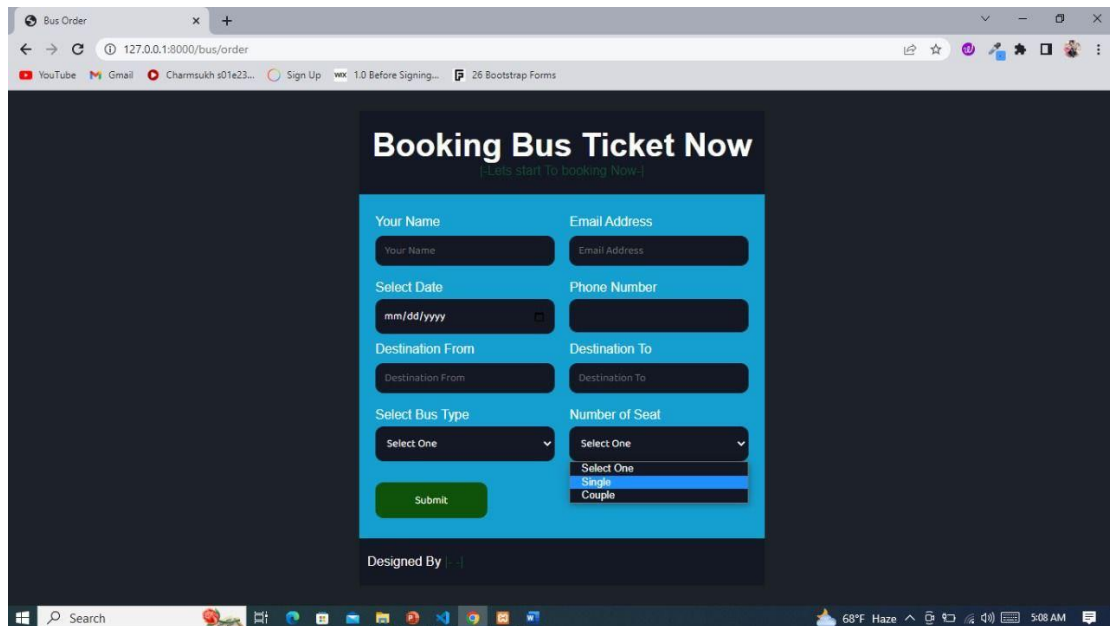


Figure 4.8.5 Order Page



Figure 4.8.5 Order Page

Payment : There are three payment options for the three orders. Payments for buses, flights, and movies are what they are.

The screenshot shows a payment form for a bus ticket purchase. It is divided into two main sections: 'BUS TICKET PURCHASE' on the left and 'PAYMENT' on the right. The 'BUS TICKET PURCHASE' section includes input fields for 'Full name', 'Email', 'Address', 'City', 'State' (with a dropdown menu), and 'Zip code'. The 'PAYMENT' section includes 'Accepted Card' with logos for VISA, Mastercard, and American Express, a 'Total Purchase Amount' field, 'Credit card number', 'Exp month', 'Exp year' (with a dropdown menu), and 'CVV'. A green 'Proceed to Checkout' button is located at the bottom right of the form.

Figure 4.8.6 Payment Page

The screenshot shows a payment form for a flight ticket purchase. It is divided into two main sections: 'FLIGHT TICKET PURCHASE' on the left and 'PAYMENT' on the right. The 'FLIGHT TICKET PURCHASE' section includes input fields for 'Full Name', 'Email', 'Address', 'City', 'State' (with a dropdown menu), and 'Zip code'. The 'PAYMENT' section includes 'Accepted Card' with logos for VISA, Mastercard, and American Express, a 'Total Purchase Amount' field, 'Credit card number', 'Exp month', 'Exp year' (with a dropdown menu), and 'CVV'. A green 'Proceed to Checkout' button is located at the bottom right of the form.

Figure 4.8.6 Payment Page

Figure 4.8.6 Payment Page

Dashboard : Here is our dashboard. Here, all information is kept current. The administrator can see orders and payments made by users here.

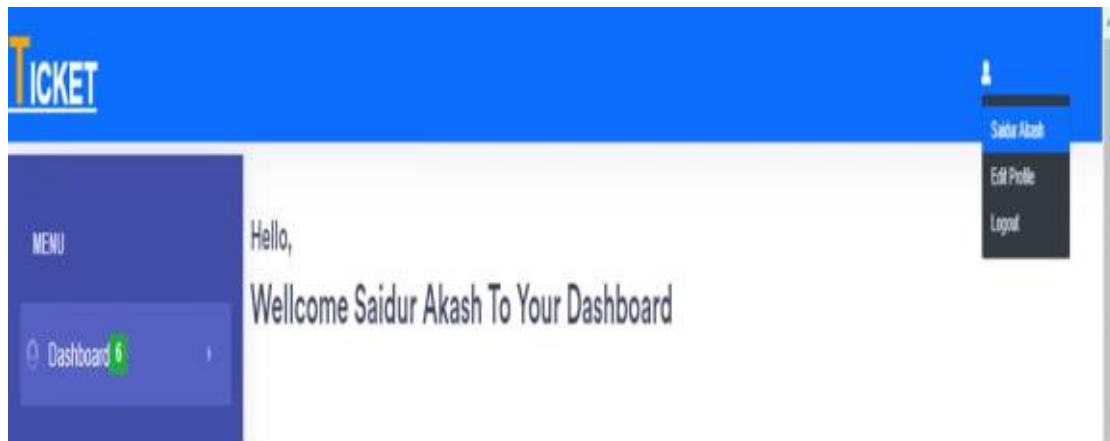


Figure 4.8.7 Dashboard

Manage Order : It manages the hierarchy. Manage Air Order, Manage Bus Order, and Manage Movie Order are the three manage orders that are offered in our manage order. We now have the option to review, change, and remove orders.

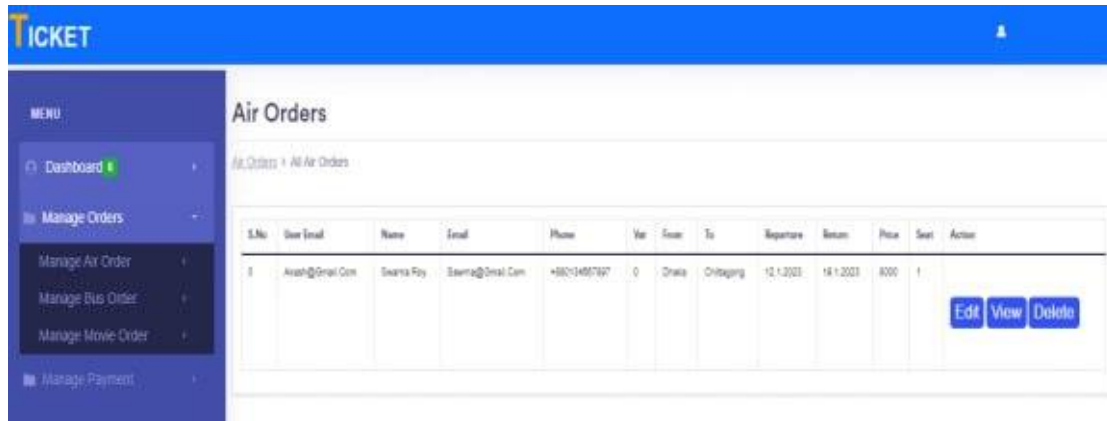


Figure 4.8.8 Manage Order Page

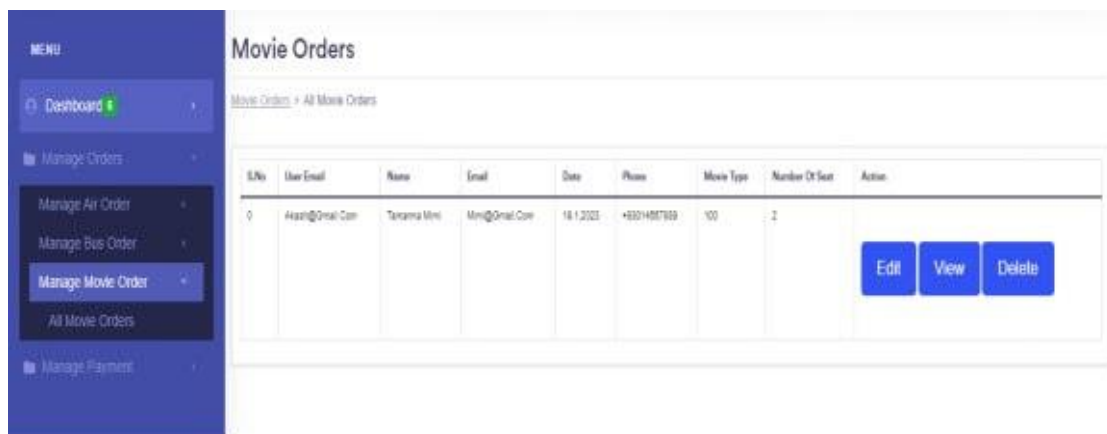


Figure 4.8.8 Manage Order Page

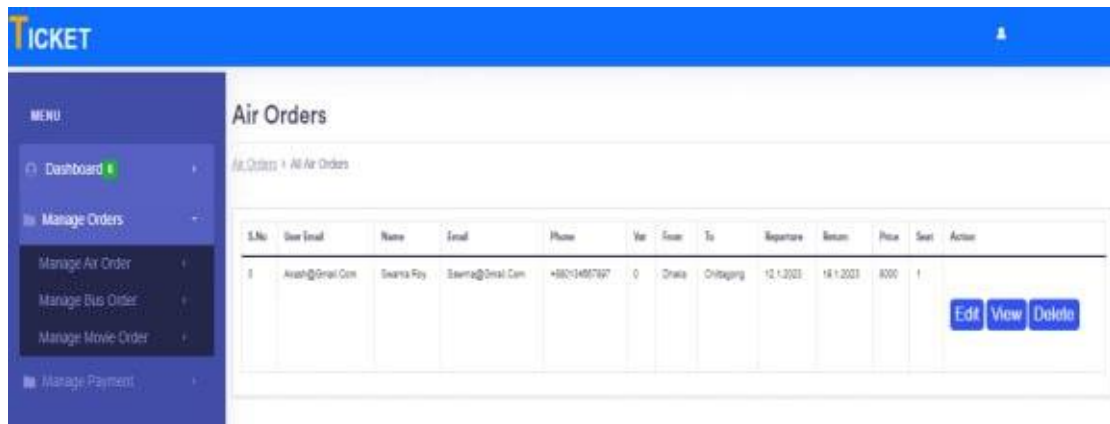


Figure 4.8.8 Manage Order Page

Manage Payment : We manage payments here. There are three manageable payments available in manage payment system: manage air, manage bus, and manage movie. From here, we may view, edit, and remove orders

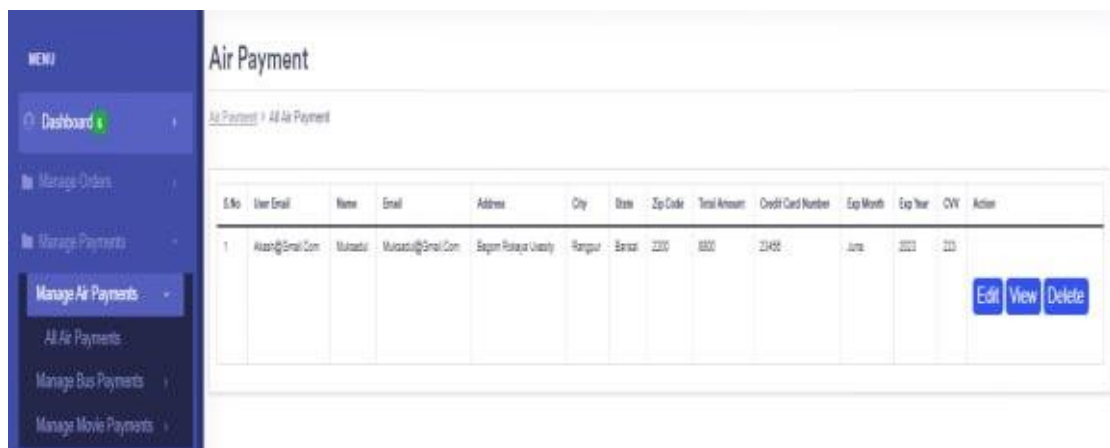


Figure 4.8.9 Manage Payment Page

S.No	User Email	Name	Email	Address	City	State	Zip Code	Total Amount	Credit Card Number	Exp Month	Exp Year	CVV	Action
1	kean@gmail.com	Muhammad	Muhammad@gmail.com	Bogor-Telaga Ulu	Pangar	Jepara	2001	200	0000	June	2004	567	Edit View Delete

Figure 4.8.9 Manage Payment Page

Manage Contact : Manage contacts right here.

S.No	User ID	Name	Phone Number	Email	Manage	Action
------	---------	------	--------------	-------	--------	--------

Figure 4.8.10 Manage Contact Page

CHAPTER 5

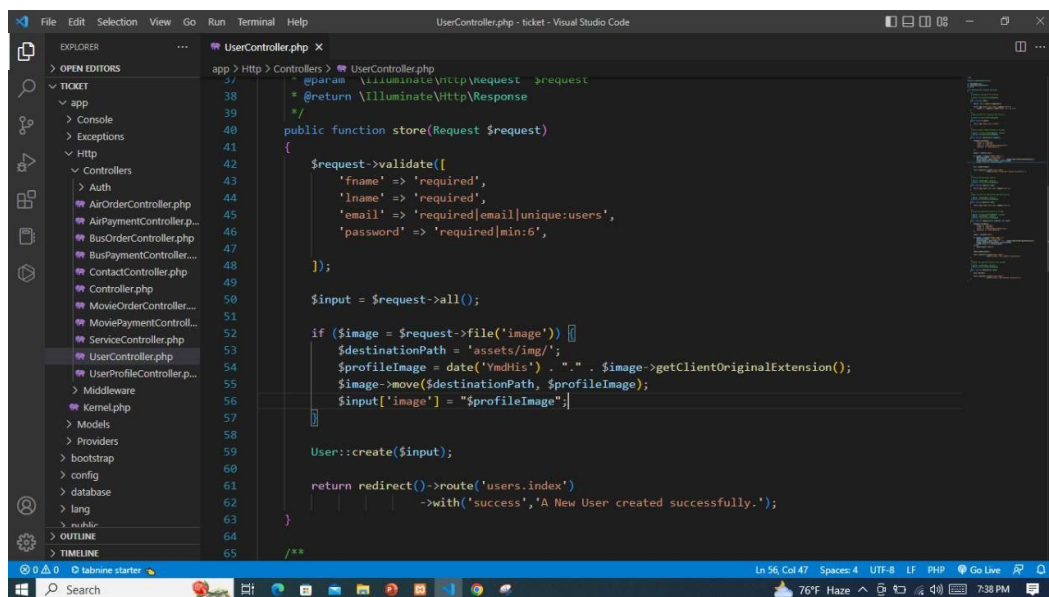
IMPLEMENTATION AND TESTING

5.1 Introduction

We'll discuss the project's implementation plan in this chapter. This web-based program has both a front-end and a back-end component because it is web-based. These two concepts will each be shown using the proper screenshots.

5.2 Implementation of Database

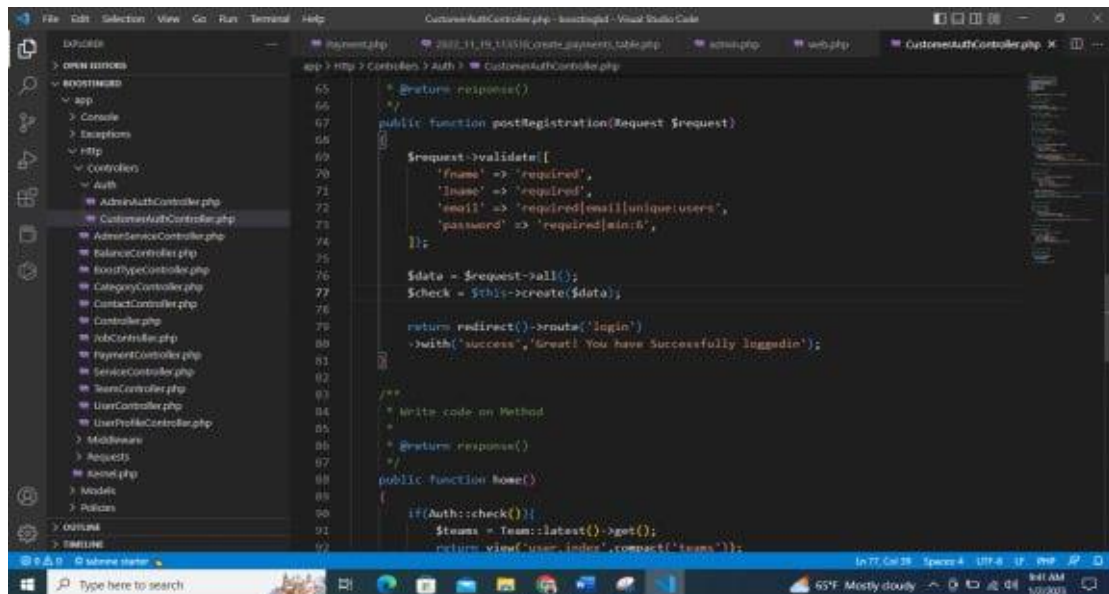
1 User Profile Controller :



```
File Edit Selection View Go Run Terminal Help UserController.php - ticket - Visual Studio Code
EXPLORER
  OPEN EDITORS
  TICKET
    app
      Http
        Controllers
          Auth
            AirOrderController.php
            AirPaymentController.p...
            BusOrderController.php
            BusPaymentController...
            ContactController.php
            Controller.php
            MovieOrderController...
            MoviePaymentControll...
            ServiceController.php
            UserController.php
            UserProfileController.p...
          Middleware
          Kernel.php
        Models
        Providers
        bootstrap
        config
        database
        lang
        public
        OUTLINE
        TIMELINE
  UserController.php X
  app > Http > Controllers > UserController.php
  37 * @param \Illuminate\Http\Request $request
  38 * @return \Illuminate\Http\Response
  39 */
  40 public function store(Request $request)
  41 {
  42     $request->validate([
  43         'fname' => 'required',
  44         'lname' => 'required',
  45         'email' => 'required|email|unique:users',
  46         'password' => 'required|min:6',
  47     ]);
  48     $input = $request->all();
  49     if ($image = $request->file('image')) {
  50         $destinationPath = 'assets/img/';
  51         $profileImage = date('YmdHis') . "." . $image->getClientOriginalExtension();
  52         $image->move($destinationPath, $profileImage);
  53         $input['image'] = "$profileImage";
  54     }
  55     User::create($input);
  56     return redirect()->route('users.index')
  57         ->with('success','A New User created successfully.');
```

Figure 5.2 User Profile Controller

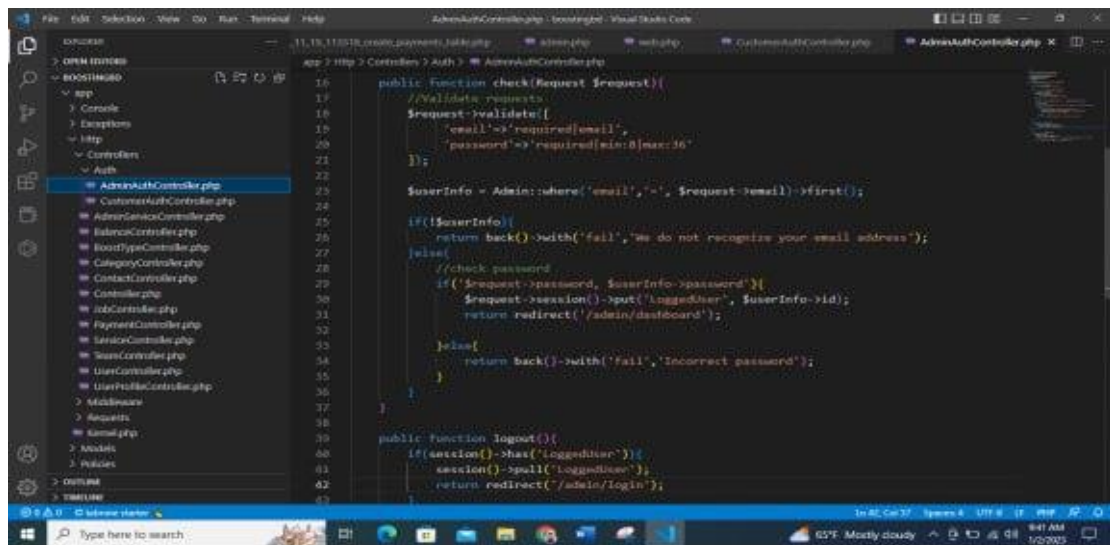
2. Customer Auth Controller: The customer Auth Controller is in charge of managing the customer login registration authentication.



```
CustomerAuthController.php - basistogist - Visual Studio Code
File Edit Selection View Go Run Terminal Help
CustomerAuthController.php - basistogist - Visual Studio Code
app > Http > Controllers > Auth > CustomerAuthController.php
65 @return response()
66 /
67 public function postRegistration(Request $request)
68 {
69     $request->validate([
70         'name' => 'required',
71         'image' => 'required',
72         'email' => 'required|email|unique:users',
73         'password' => 'required|min:6',
74     ]);
75     $data = $request->all();
76     $check = $this->create($data);
77
78     return redirect()->route('login')
79     ->with('success','Great! You have Successfully loggedin');
80 }
81 /**
82  * Write code on Method
83  * @return response()
84  */
85 public function home()
86 {
87     if(Auth::check()){
88         $teams = Team::latest()->get();
89         return view('user.index',compact('teams'));
90     }
91 }
92
93
94
```

Figure 5.2 Customer Auth Controller

3. Admin Auth Controller: Crols login authentication for administrators.



```
public function check(Request $request){
    //validate requests
    $request->validate([
        'email'=>'required|email',
        'password'=>'required|min:8|max:36'
    ]);

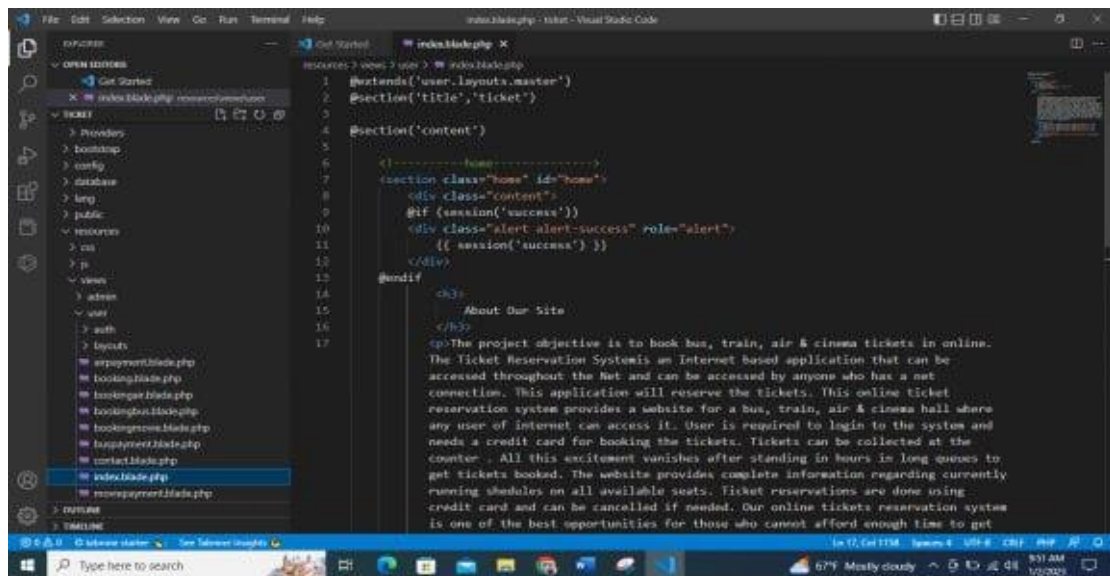
    $userInfo = Admin::where('email','=', $request->email)->first();

    if(!$userInfo){
        return back()->with('fail','We do not recognize your email address');
    }else{
        //check password
        if($request->password, $userInfo->password){
            $request->session()->put('loggedUser', $userInfo->id);
            return redirect('/admin/dashboard');
        }else{
            return back()->with('fail','Incorrect password');
        }
    }
}

public function logout(){
    if(session()->has('loggedUser')){
        session()->pull('loggedUser');
        return redirect('/admin/login');
    }
}
```

Figure 5.2 Admin Auth Controller

4. Index Page: The index page is the main page of our project.



```
@extends('user.layouts.master')
@section('title','ticket')
@section('content')
<!-- Home -->
<section class="home" id="home">
<div class="content">
@if (session('success'))
<div class="alert alert-success" role="alert">
{{ session('success') }}
</div>
@endif
<h3>
    About Our Site
</h3>
<p>The project objective is to book bus, train, air & cinema tickets in online. The Ticket Reservation System is an Internet based application that can be accessed throughout the Net and can be accessed by anyone who has a net connection. This application will reserve the tickets. This online ticket reservation system provides a website for a bus, train, air & cinema hall where any user of internet can access it. User is required to login to the system and needs a credit card for booking the tickets. Tickets can be collected at the counter. All this excitement vanishes after standing in hours in long queues to get tickets booked. The website provides complete information regarding currently running schedules on all available seats. Ticket reservations are done using credit card and can be cancelled if needed. Our online tickets reservation system is one of the best opportunities for those who cannot afford enough time to get
```

Figure 5.2 Index Page

5. Dashboard: On the dashboard page, you may find our admin panel or dashboard.

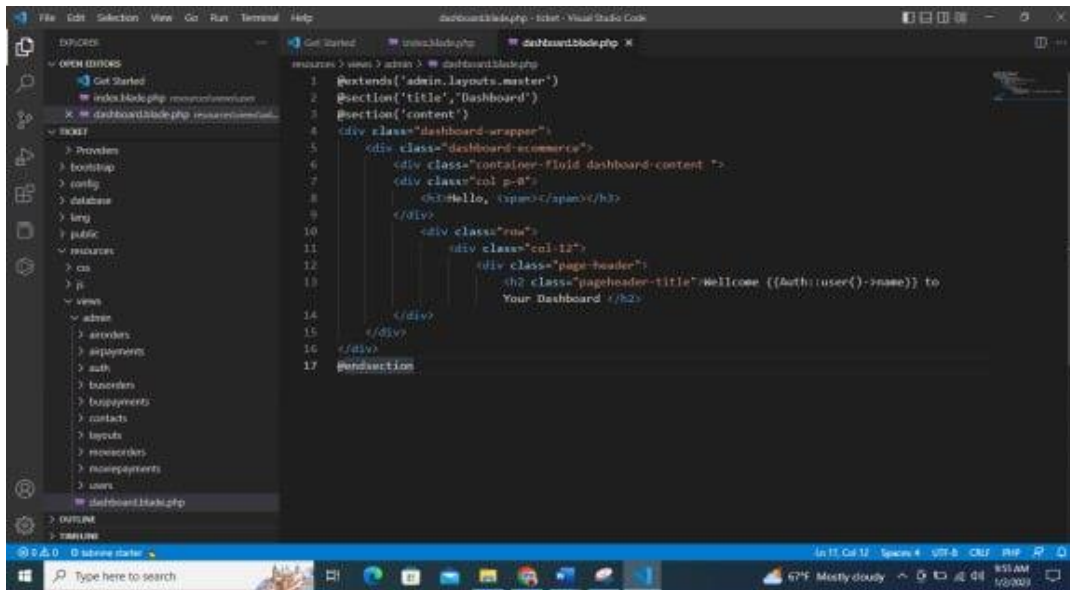


Figure 5.2 Dasboard

6. Service Controller: The service controller has control over the service page.

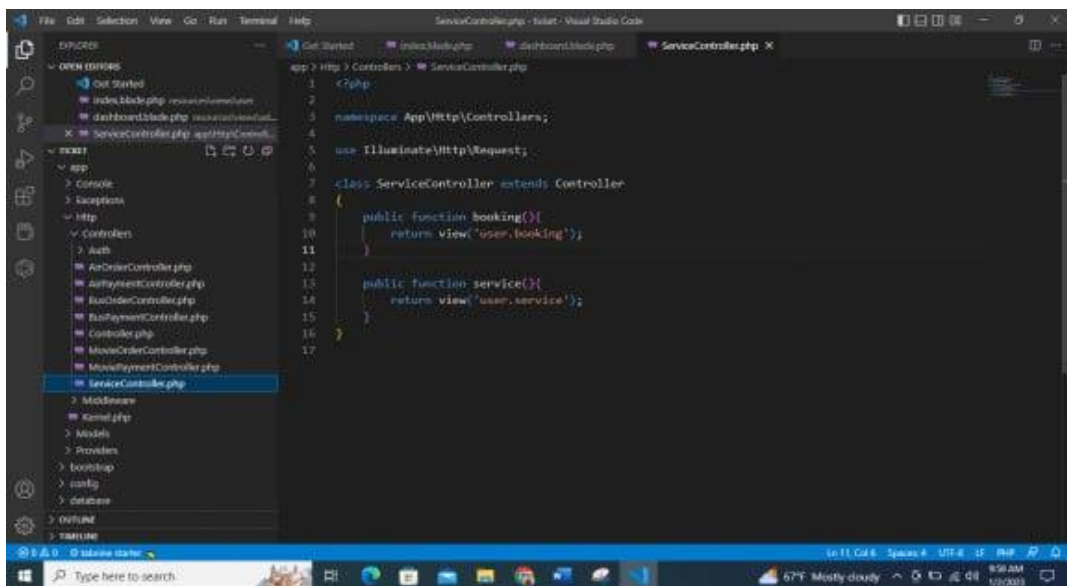


Figure 5.2 Service Controller

7. Service Blade: The service blade is a page that lists services.

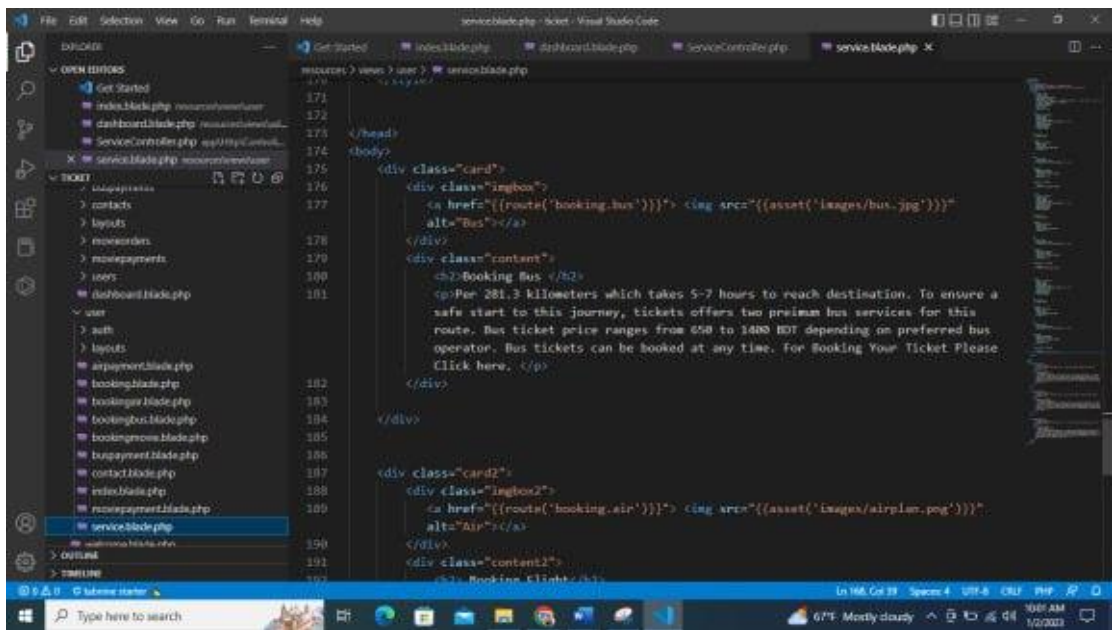


Figure 5.2 Service Blade

8. Order Controller: There are three order controllers on duty. Based on user requests, the controller managed orders for buses, flights, and movies.

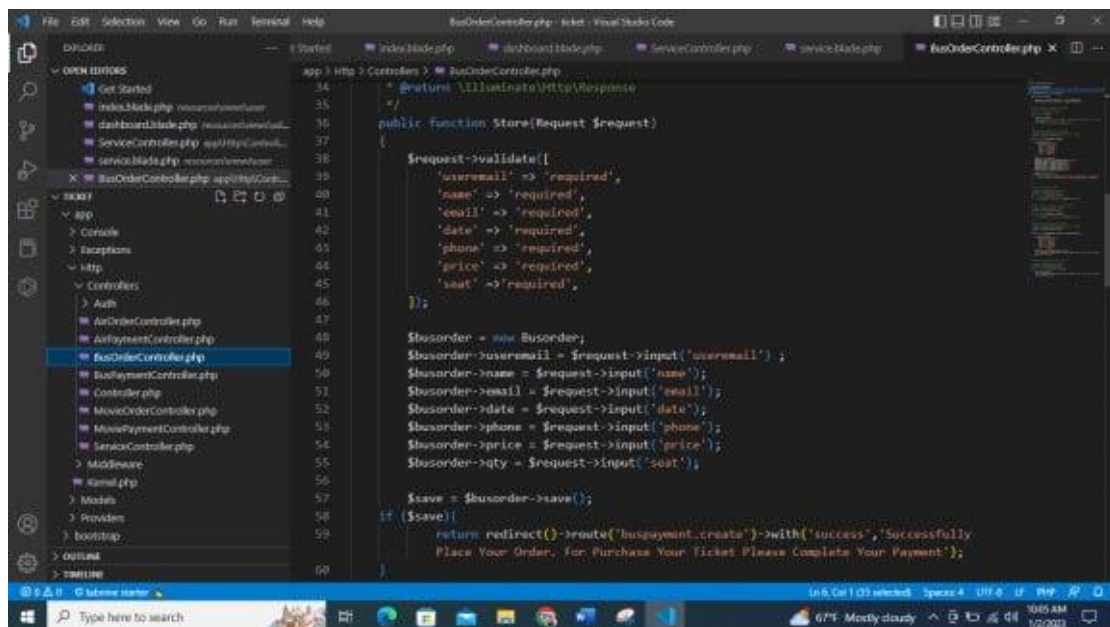


Figure 5.2 Order Controller

9. Order View pages: Three order pages are available. Booking a bus, a flight, or a movie. To request an order, users can enter their information here.

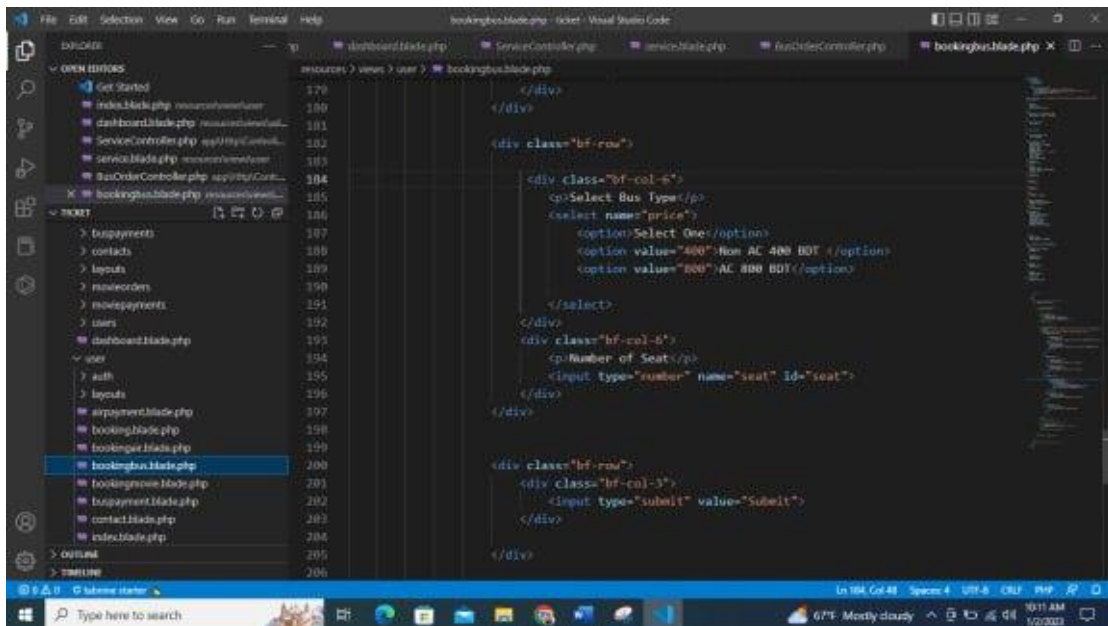


Figure 5.2 Order View Page

10. Payment Controller : The three payment processors are the bus payment controller, the air payment controller, and the movie payment controller. They run their business with the help of three controllers

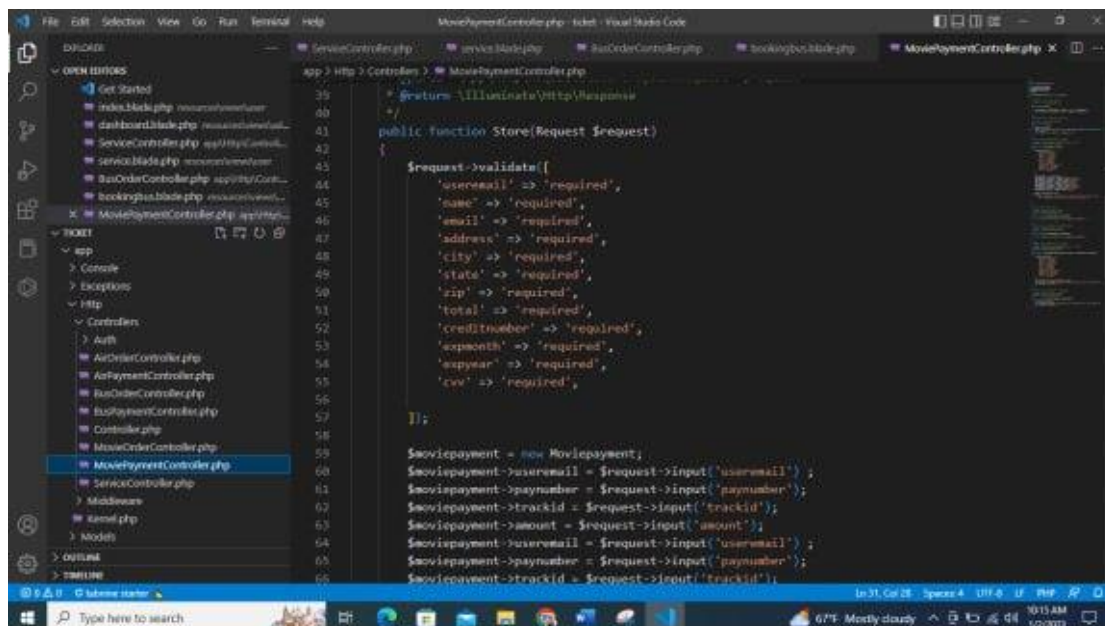


Figure 5.2 Payment Controller Page

11.All Order: The admin panel's code, which manages all orders for bus tickets, is provided here. There are additional movies and air that are comparable to it.

```

<tr>
<th>S.No</th>
<th>User Email</th>
<th>Name</th>
<th>Email</th>
<th>Date</th>
<th>Phone</th>
<th>Bus Type</th>
<th>Number Of Seat</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<?php $i=0; ?>
@foreach ($busorders as $busorder)
<tr>
<td>{{ $i++ }}</td>
<td>{{ $busorder->useremail }}</td>
<td>{{ $busorder->name }}</td>
<td>{{ $busorder->email }}</td>
<td>{{ $busorder->date }}</td>
<td>{{ $busorder->phone }}</td>
<td>{{ $busorder->price }}</td>
<td>{{ $busorder->seat }}</td>
<td>
<form action="{{ route('busorders.destroy', $busorder->id) }}"
method="Post">
<a class="btn btn-primary" href="{{ route('busorders.edit',

```

Figure 5.2 All Order Page

12. All Payment : This is the code for the administrative interface for purchasing and booking bus tickets. A movie and air both exist in the same way.

```

<th>State</th>
<th>Zip code</th>
<th>Total Amount</th>
<th>Credit card number</th>
<th>Exp month</th>
<th>Exp year</th>
<th>CVV</th>
<th>Action</th>
</tr>
</thead>
<tbody>
@foreach ($buspayments as $buspayment)
<tr>
<td>{{ $buspayment->id }}</td>
<td>{{ $buspayment->useremail }}</td>
<td>{{ $buspayment->name }}</td>
<td>{{ $buspayment->email }}</td>
<td>{{ $buspayment->address }}</td>
<td>{{ $buspayment->city }}</td>
<td>{{ $buspayment->state }}</td>
<td>{{ $buspayment->zip }}</td>
<td>{{ $buspayment->total }}</td>
<td>{{ $buspayment->creditnumber }}</td>
<td>{{ $buspayment->expmonth }}</td>
<td>{{ $buspayment->expyear }}</td>
<td>{{ $buspayment->cvv }}</td>
<td>
<form action="{{ route('buspayments.destroy', $buspayment->id) }}"

```

Figure 5.2 All Payment Page

5.3 Testing Implementation

Log in Pre-condition/Register

SL No	Statement	Test Data	Result	Pass/Fail
1	If email is not correctly inserted	akash@gmail.com Password:12345	Please enter correct email	fail
2	If email password is not correctly inserted	akash@gmail.com Password:12345	Please enter correct email	fail
3	If email and password is correctly inserted	swarna@gmail.com Password:12345	Your registration successfully	pass
4	If email and password is correctly inserted	swarna@gmail.com Password:12345	Your registration successfully	pass

5.4 Implementation of Font-End Design

The front end of this type of shopping platform is essential in order to draw in more customers. As a result, the front-end has received a lot of attention. I've worked hard to make the project responsive so that it can be used on any device in an effort to expand my clientele. It made use of the bootstrap framework. Many contemporary websites and web applications are made using this front-end framework. It is a free front-end framework made to hasten and make the creation of websites easier. For forms, typography, buttons, navigation, tables, modals, picture carousels, and a range of other features, it also supports additional optional HTML and CSS-based [5] design templates.

5.5 Test Results and Reports

The above experiments were successful, and the results were encouraging. It's easy to make mistakes because Laravel [6] was used throughout the entire project's development and because there are over 40 files and thousands of lines of code. We used unit testing to find and fix our mistakes. The development of the application and the integration test were done simultaneously. Function testing is used to test user interfaces, APIs, and databases. Carousels and many other website elements passed testing, which was successful.

CHAPTER 6

IMPACT ON SOCIETY, ENVIRONMENT AND SUSTAINABILITY

6.1 Influence of Society

After performing the aforementioned tests, we got positive results. Considering the full project .Small business owners can easily manage their operations from my website without spending any money. A very small portion of each order's sales will go to the administrator. We are confident that our idea will be extremely beneficial to small business owners as well as customers, managers, and our society as a whole.

6.2 Effects on the Environment

The management approach is more carbon-efficient than conventional retail locations, according to a global study by generation. The company's carbon-efficient activities span industries like freight transportation.

6.3 Ethical Aspects

The assessment must be given by a qualified person, and this must be confirmed. We make sure students can only apply through the website. We chose the well-known programming platform Laravel, which is renowned for its security.

6.4 Sustainability

For all of our data, the model we utilized appears to provide a takeable accuracy of over 80%. Our society is constantly changing as we speak. Although a more accurate model may exist, the working model and procedures may not change because the principle obeying project is utilized for a variety of applications, the bulk of which are handled by the server.

CHAPTER 7

CONCLUSION AND FUTURE SCOPE

7.1 Discussion and Conclusion

Our project was created using PHP and Laravel, and it is based on the analysis of current systems and the specification of user requirements. Future improvements are also possible. In conclusion, Inventory Management System is a straightforward desktop application. program primarily suitable for small enterprises. It has every single tool required for a small business. Our team has succeeded in developing a program that enables us to change, add, and remove objects as necessary. A daily report that details the tool's daily sales and purchases is also available. Small enterprises with few available IoDs should use this application. Despite some of its shortcomings, our team is convinced that the organization will profit from the installation of this technology.

7.2 Scope for Future Development

Since we had very little prior understanding of the Inventory Management System when we began this project, we learned about the improvement capacity as we built it. The following are some of the areas of focus we might broaden for improvement and effectiveness:

Design of an interactive user interface.

Control stock Down wisely.

Oracle is used as the database.

A system of online payments could be added.

Making any form of system adaptable.

To enable product returns, a sales and buy return mechanism will be developed.

Inaccurate and broken

REFERENCES

- [1] Learn About Laravel, available at <<<https://www.laravel.com>>>, last accessed on 4-12-2022 at 9:00 PM
- [2] Front end discussion< <https://airfocus.com/glossary/what-is-a-front-end/>> last accessed on 3-7-2022
- [3] Use Case diagram < <https://www.lucidchart.com/pages/uml-use-case-diagram> > last accessed on 11-9-2022 at 10.20 AM
- [4] Learn About Laravel, available at <<<https://www.laravel.com>>>, last accessed on 4-12-2022 at 9:00 PM
- [5]Jon Duckett, HTML and CSS, CSS Basic, 2rd Edition, The Ebook Press, 2018,
- [6] Learn About Laravel, available at <<<https://en.wikipedia.org/wiki/Laravel>>>, last accessed on 15-12-2022 at 5:00 PM.

ORIGINALITY REPORT

28%

SIMILARITY INDEX

22%

INTERNET SOURCES

3%

PUBLICATIONS

23%

STUDENT PAPERS

PRIMARY SOURCES

1

dspace.daffodilvarsity.edu.bd:8080

Internet Source

15%

2

Submitted to Daffodil International University

Student Paper

4%

3

Submitted to Asia Pacific International College

Student Paper

2%

4

www.coursehero.com

Internet Source

1%

5

Submitted to De Montfort University

Student Paper

1%

6

Submitted to University of Strathclyde

Student Paper

1%

7

Submitted to Universiti Tenaga Nasional

Student Paper

1%

8

Submitted to Morgan Park High School

Student Paper

1%

9

Submitted to tup

Student Paper

1%
