VISVESVARAYA TECHNOLOGICAL UNIVERSITY
"Jnanasangama", Belagavi-590018, Karnataka

BANGALORE INSTITUTE OF TECHNOLOGY
K.R. Road, V.V.Puram, Bangalore-560 004

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

DATABASE MANAGEMENT SYSTEM MINI
PROJECT 18CSL58

"TRAVEL MANAGEMENT SYSTEM"

Submitted By

SUDHANSHU ARYA-1BI20CS171
SYED RUMAAN-1BI20CS179

for the academic year 2022-23

Department of Computer Science & Engineering Bangalore Institute of Technology
K.R. Road, V.V.Puram, Bangalore-560 004

BANGALORE INSTITUTE OF TECHNOLOGY

K.R. Road, V.V.Puram, Bangalore-560 004

Department of Computer Science & Engineering

Certificate

This is to certify that the implementation of **DBMS MINI PROJECT** entitled "**TRAVEL MANAGEMENT**" has been successfully completed by

USN: 1BI20CS179	NAME: SYED RUMAAN
1BI20CS171	SUDHANSHU ARYA

of V semester B.E. for the partial fulfillment of the requirements for the Bachelor's degree in Computer Science & Engineering of the Visvesvaraya Technological University during the academic year 2022-2023.

Lab In-charge :

Dr. SAVITHA S K   **Dr. J GIRIJA**
Associate Professor   Professor and Head
Dept. of CS&E   Department of CS&E
Bangalore Institute of Technology   Bangalore Institute of
TechnologyBangalore   Bangalore

Examiners: 1)   2)

ACKNOWLEDGEMENT

The knowledge & satisfaction that accompany the successful completion of any task would be incomplete without mention of people who made it possible, whose guidance and encouragement crowned my effort with success. I would like to thank all and acknowledge the help I have received to carry out this Mini Project.

I would like to convey my thanks **Dr. J Girija.** HoD, Dept. Of CS&E for being kind enough to provide the necessary support to carry out the mini project.

I am most humbled to mention the enthusiastic influence provided by the Faculty in-charge **Dr. Savitha S K** Associate Professor, Dept. Of CS&E on the project for their ideas, time to time suggestions for being a constant guide and co-operation showed during the venture and making this project a great success.

I would also take this opportunity to thank my friends and family for their constant support and help. I am very pleased ton express my sincere gratitude to the friendly co-operation showed by all the members of **Computer Science & Engineering Department, BIT.**

CONTENT

1. INTRODUCTION

1.1 Overview

Web design refers to the design of websites that are displayed on the internet. It usually refers to the user experience aspects of website development rather than software development. Web design used to be focused on designing websites for desktop browsers; however, since the mid-2010s, design for mobile and tablet browsers has become ever-increasingly important.

A web designer works on the appearance, layout, and, in some cases, content of a website. Appearance, for instance, relates to the colors, font, and images used. Layout refers to how information is structured and categorized. A good web design is easy to use, aesthetically pleasing, and suits the user group and brand of the website. Many webpages are designed with a focus on simplicity, so that no extraneous information and functionality that might distract or confuse users appears. As the keystone of a web designer's output is a site that wins and fosters the trust of the target audience, removing as many potential points of user frustration as possible is a critical consideration.

Two of the most common methods for designing websites that work well both on desktop and mobile are responsive and adaptive design. In responsive design, content moves dynamically depending on screen size; in adaptive design, the website content is fixed in layout sizes that match common screen sizes. Preserving a layout that is as consistent as possible between devices is crucial to maintaining user trust and engagement. As responsive design can present difficulties in this regard, designers must be careful in relinquishing control of how their work will appear. If they are responsible for the content as well, while they may need to broaden their skillset, they will enjoy having the advantage of full control of the finished product.

1.2 Problem Statement

This project gives an overview about the aim, objectives, background and operation environment of a tourism webpage, which is an online website for tourism package. It is a software built for the users to

view and gather information about

the available tour packages and the

travel destinations that are trending.

Here in this website, the admin is given the complete control over the architecture of the webpage. The objective and scope of this project is to provide a simple yet effective platform which provides information based on the user needs. During implementation every admin has a login for the authentication. The admin has privileges to make changes to any part of the code and the ability to set rules for the users.
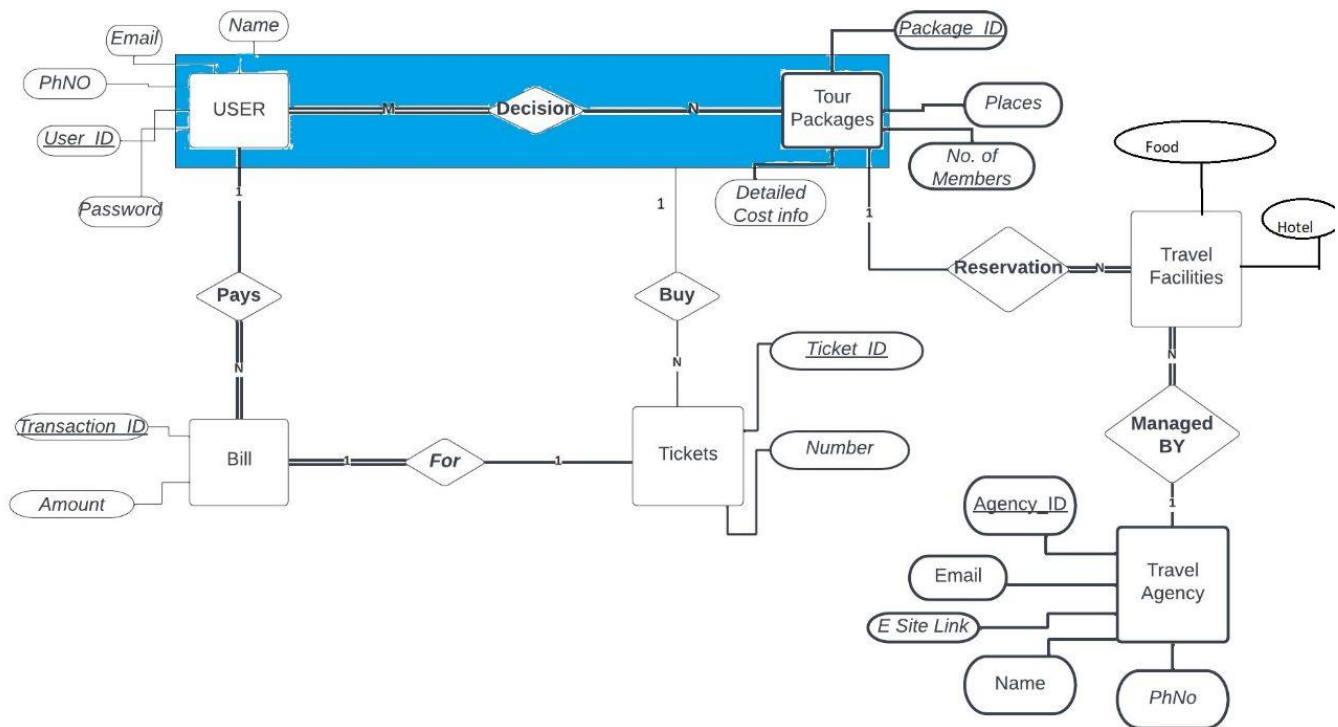
2. BACK END DESIGN

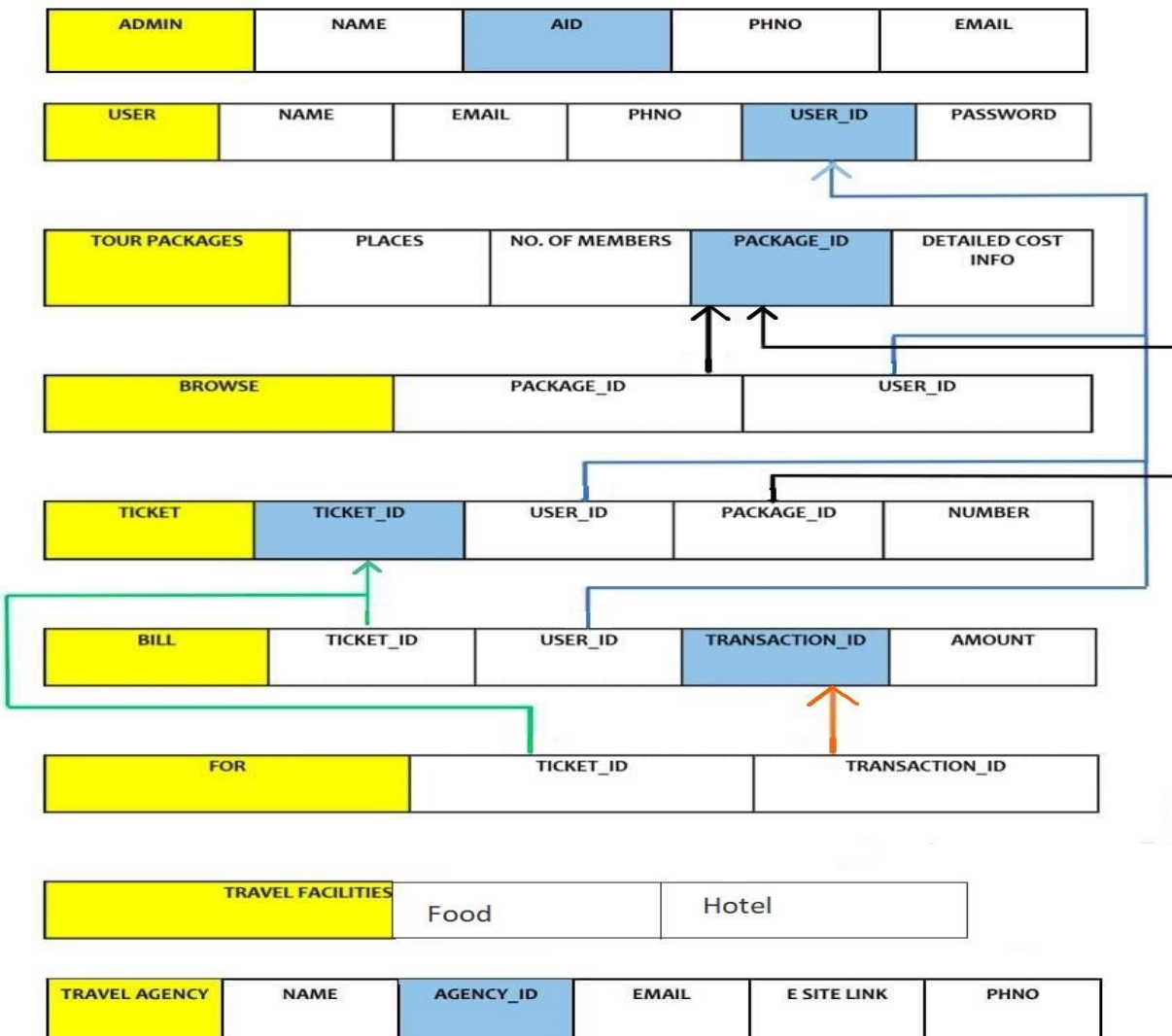## 2.1 Conceptual Database Design

Figure 2.1.1 ER-Diagram

2.2 Logical Database Design

RELATIONAL MAPPING:

ADMIN	NAME	AID	PHNO	EMAIL

USER	NAME	EMAIL	PHNO	USER_ID	PASSWORD

TOUR PACKAGES	PLACES	NO. OF MEMBERS	PACKAGE_ID	DETAILED COST INFO

BROWSE	PACKAGE_ID	USER_ID

TICKET	TICKET_ID	USER_ID	PACKAGE_ID	NUMBER

BILL	TICKET_ID	USER_ID	TRANSACTION_ID	AMOUNT

FOR	TICKET_ID	TRANSACTION_ID

TRAVEL FACILITIES	Food	Hotel

TRAVEL AGENCY	NAME	AGENCY_ID	EMAIL	E SITE LINK	PHNO

Figure 2.2.1 Relational Mapping

ADMIN(ID , UserName , Password , UpdationDate)

TblTourPackages(Pid , state , PName , Ptype , Plocation ,Pprice , Pfeatures, ,Pdetails , PCreationDate , PupdationDate)

tblUsers(Id , FullName , MobNum , EmailId , Password , RegDate , UpdationDate)

tblBooking(Bid , Pid , Uemail , PeopleCount , TotalPrice , FromDate , ToDate , RegDate , Status , Comment , CancelledBy , UpdationDate)

tblEnquiry(id , FullName , EmailId , MobNum , Subject , Description , PostingDate) **tblIssues**(idUserEmail , Issue , Description , PostingDate , AdminRemark)

tblPages(id, type , detail)

2.3 Normalization

Database Normalization is a technique of organizing the data in the database. Normalization is a systematic approach of decomposing tables to eliminate data redundancy and undesirable characteristics like Insertion, Update and Deletion Anomalies. It is a multi- step process that puts data into tabular form by removing duplicated data from the relation tables. Normalization is used for mainly two purposes,

- Eliminating redundant(useless) data.
- Ensuring data dependencies make sense i.e. data id logically stored.

First Normal Form (1NF) :

As per First Normal Form

- There are no duplicated rows in the table.
- Each attribute is single valued or atomic.

Second Normal Form (2NF) :

As per Second Normal Form, a table is in 2NF if every non prime attribute is not partially dependent on any key of the table.

Third Normal Form (3NF) :

Third Normal Form applies that every non-prime attribute of table must be dependent on primary key, or we can say that, there should not be the case that a non-prime attribute is determined by another nonprime attribute. So this transitive functional dependency should be removed from the table and also the table must be in the Second Normal Form.

Relational schema:

R(AId , AUserName,Apassword,AdminupdationDate, PId , state, Pname, Ptype , Plocation , Pprice ,Pfeatures, , Pdetails , PCreationDate , PUpdationDate , BId , BPid , PeopleCount ,BEmail , TotPrice , FromDate ,ToDate , RegDate , Status , BComment, CancelledBy ,BUpdationDate , UId , UFullName , UMobNum , UEmail , UPassword URegDate ,UserUpdationDate , EnqId , EFullName , EnqEmail , EMobNum , Subject , EDescription , IssueId , IEmail , Issue , IssueDescription , IPostingDate , AdminRemark)

Functional Dependencies :

FD1 = {AId ->AUserName ,APassword , AdminUpdationDate }

FD2 = {PId ->state ,PName , PType , Plocation , Pprice , Pfeatures, Pdetails ,
PCreationDate ,PUpdationDate , BPId }

FD3 = {BId ->BPId ,PeopleCount , BEmail , TotPrice , FromDate ,ToDate , RegDate,
Status ,BComment, CancelledBy , BUpdationDate }

FD4 = {UId ->UFullName ,UMobNum , UEmail , UPassword , URegDate , UserUpdationDate }

FD5 = {EnqId ->EFullName ,EnqEmail , EMobNum , Subject , EDescription }

FD6 = {IssueId ->IEmail , Issue , IssueDescription , IPostingDate , AdminRemark}

FD7 = {UEmail ->BEmail ,EnqEmail , IEmail }

FD8 = {UFullName ->EFullName }

FD9 = {UMobNum ->EMobNum }

Candidate key = { Aid , UserId , BookingId , PackageId , EnqId , IssueId , PageId)

Dependency Graph - a dependency graph is a directed graph representing dependencies of several

objects towards each other.

2.3.1 Dependency Graph

Figure 2.3.1 Dependency graph

2.3.2 Normalization of the tables

Normalization of ADMIN Table :

AId	UserName	Password	UpdationTime
1	admin	qwerty	2020-11-07 11:18:49
2	root	server123	2021-01-12 10:00:00

First Normal Form
 In ADMIN, all the attributes are atomic and there cannot be duplicate rows. Hence, it is in 1NF.

Second Normal Form
This is already in 2NF since every non key attribute is fully dependent on primary key.

Third Normal Form
Since there is no transitive functional dependency, therefore table is already in 3NF.

Normalization of tblUserstable :

UId	FullName	MobNum	EmailId	Password	RegDate	UpdationDate
1	Anuj kumar	1111111111	anuj@gmail.com	anuj2k	2021-01-09 10:38:17	2021-01-12 11:59:42
3	Sarita	9999999999	sarita@gmail.com	sarita	2021-01-09 10:50:48	2021-01-13 09:26:30
12	Sahana	9998880002	sahana@gmail.com	qwerty2k	2021-01-12 09:29:11	2021-01-12 18:50:29

First Normal Form
 In ADMIN, all the attributes are atomic and there cannot be duplicate rows. Hence, it is in 1NF.

Second Normal Form
This is already in 2NF since every non key attribute is fully dependent on primary key.

Third Normal Form

Since there is no transitive functional dependency, therefore table is already in 3NF.

Normalization of tblTourPackages :

Package Id	state	PName	PType	PLoc	PPrice	PFeatures	PDetails	CreationDate	UpdationDate
3	Karnataka	Treasures of karnataka tour	Historical	Hampi	1000	Excepteur sint occaeca	velit esse cillum dolore	2021-01-07 16:00:58	2021-01-10 21:23:55
4	Kerla	Kerala	Familty and Couple	Kelara	2000	velit esse cillum dolore eu fugiat nulla pariatur.	Excepteur sint occaecat cupidatat non proident	2021-01-07 22:39:37	2021-01-10 21:23:11
7	Rajasthan	Rajasthan	General	Rajasthan	3000	Free wifi, pickup and drop etc	Test Test Test Test	2021-01-07 16:00:58	2021-01-10 21:35:47
8	Tamilnadu	Tamil nadu trip	General	Tamil Nadu	1000	Air Conditioning ,Balcony ,Fan , hairdryer	Lorem ipsum dolor sit amet	2021-01-07 14:23:44	2021-01-10 21:42:43

First Normal Form
In ADMIN, all the attributes are atomic and there cannot be duplicate rows. Hence, it is in 1NF.

Second Normal Form
This is already in 2NF since every non key attribute is fully dependent on primary key.

Third Normal Form
Since there is no transitive functional dependency, therefore table is already in 3NF.

Normalization of tblBooking :

BookingId	PackageId	UserEmail	PeopleCount	TotalPrice	Comment
2	1	anuj@gmail.com	0	0	"Loremipsum dolorsitamet, cpariatur."
12	3	sahana@gmail.com	3	12000	velitesse cillumdolore eufugiat nulla
13	4	sahana@gmail.com	2	24000	

Extension of tblBooking(due to no space)

BId	FromDate	ToDate	RegDate	Status	CancelledBy	UpdationDate
2	2021-01-18	2021-01-31	2021-01-07 19:01:10	2	u	2020-01-10 21:30:23
12	2021-01-15	2021-01-19	2021-01-12 09:34:19	1	NULL	2021-01-12 15:14:13
13	2021-01-20	2021-01-26	2021-01-12 18:51:22	0	NULL	NULL

First Normal Form
In ADMIN, all the attributes are atomic and there cannot be duplicate rows. Hence, it is in 1NF.

Second Normal Form
This is already in 2NF since every non key attribute is fully dependent on primary key.

Third Normal Form
Since there is no transitive functional dependency, therefore table is already in 3NF.

Normalization of tblEnquiryTable :

EnqId	FullName	EmailId	MobNum	Subject	Description	PostingDate
5	Anuj	anuj@gmail.com	1111111111	Nemo enimipsamvoluptatemquia vola	Lorem ipsum dolorsit amet	2021-01-17 15:41:46
6	Sahana	sahana@gmail.com	9998880002	ventore veritatis et	sed do eiusmod	2021-01-17 15:42:27
				quasi architecto	tempor incididuntut laboreetdo	

First Normal Form
 In ADMIN, all the attributes are atomic and there cannot be duplicate rows. Hence, it is in 1NF.

Second Normal Form
This is already in 2NF since every non key attribute is fully dependent on primary key.

Third Normal Form
 Since there is no transitive functional dependency, therefore table is already in 3NF.

Normalization of tblIssuestable :

IssueId	UserEmail	Issue	Description	PostingDate	AdminRemark
4	anuj@gmail.com	Cancellation	Loremipsum dolorsitamet, consectetur adipiscingelit	2021-01-07 22:03:33	Sedut perspiciatis undeomnis istenatus
5	sarita@gmail.com	Refund	Utenim ad minim veniam, quis	2021-01-09 01:12:14	totam rem aperiam, eaqueipsa quae

First Normal Form
In ADMIN, all the attributes are atomic and there cannot be duplicate rows. Hence, it is in 1NF.

Second Normal Form
This is already in 2NF since every non key attribute is fully dependent on primary key.

Third Normal Form
Since there is no transitive functional dependency, therefore table is already in 3NF.

3. FRONT END DESIGN

3.1 Screen layout Design

One of the most important for any application is the design of the graphical user interface (GUI) and the layout of the screen. Many of the applications have become popular because of the good and ease of use of the application because of the good user interface and attractive design of the GUI(Graphical user interface). The basic web page is divided into sections as shown below :- Header, Navigation(is not there for some pages that require full screen), Main Content and footer
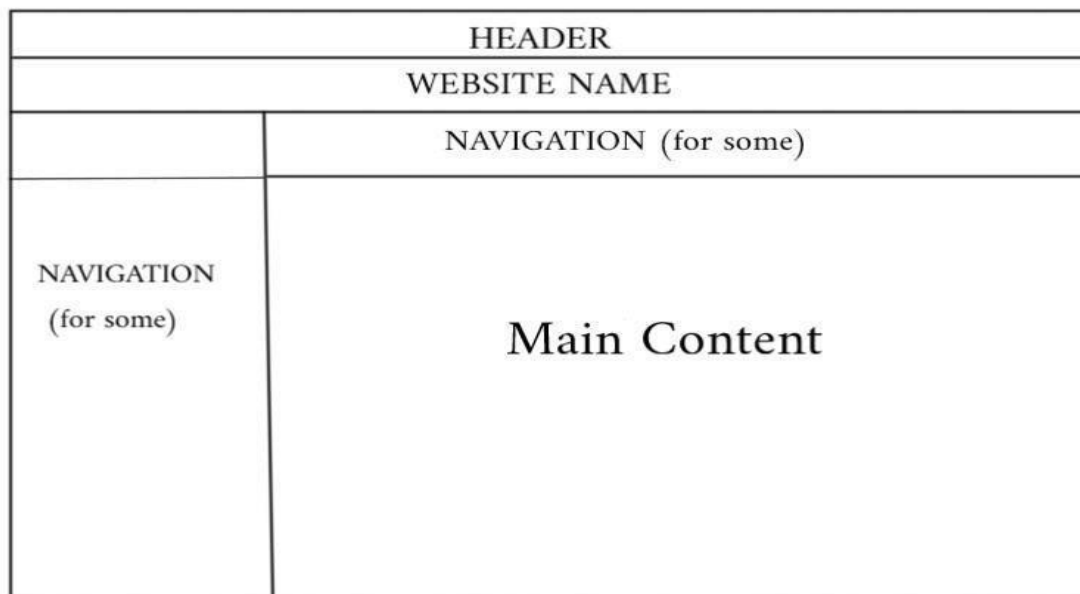
Figure 3.1.1 Screen Layout

3.2 System Requirements

A software requirements specification (SRS) is a description of asoftware system to bedeveloped. The software requirements specification lays outfunctionalandnon-functional requirements, and it may include a set ofuse cases that describe user interactions that the software must provide to the user for perfect interaction. Software requirements specification establishes the basis for an agreement between customers and contractors or suppliers on how the software product should function. Software requirements specification is a rigorous assessment of requirements before the more specific system design stages, and its goal is to reduce later redesign. It should also provide a realistic basis for estimating product costs, risks, and schedules.

3.2.1 Hardware and Software Requirements
Software requirements

Front End:

  • Packages: XAMPP CONTROL PANAL V3.2.2

  • HTML 5 (Hypertext Mark-up Language)

  • CSS3 (Cascading style sheet)

  • PHP 5.5 (Hypertext Preprocessor) Sublime Text Back End:

  • phpMyAdmin- A tool for administration of MySQL Databases over the web.

Hardware requirements

  • Dual Core Processor

  • 2 GB RAM

  • 2 GB Hard Disk Space

Department of CS&E, 2022-23

3.3 Connectivity

In this application I am using phpMyAdmin which is an open source software tool and a config.php to connect the localhost server to the database.
There are three types of methods in PHP to connect MySQL database through backend:

☐MySQL

☐MySQLi

☐PDO an overview of connecting Database using

PDO.

1.Connect MySQL using Localhost Server

2.Connect MySQL using PDO

3.Connect MySQL using Remote MySQL

The below is an example how we connect to the database.

Config.php :

```php
<?php
// DB credentials.
define('DB_HOST','localhost');
define('DB_USER','root');
define('DB_PASS','');
define('DB_NAME','tms'); //
Establish database connection.
try {
$dbh = new PDO("mysql:host=".DB_HOST.";dbname=".DB_NAME,DB_USER,
DB_PASS,array(PDO::MYSQL_ATTR_INIT_COMMAND => "SET NAMES 'utf8'"));
}
catch (PDOException $e)
{
exit("Error:                    "                    .                    $e->getMessage());}?>
```

Department of CS&E, 2022-23

4. DESCRIPTION OF FUNCTIONALITIES

4.1 Modules that user can access

1.   Home page -This page is the landing page for the "Travel Management System" web page where the user gets an overview of different options that can assessed in the web page. Also, this page provides the admin a login option.

2.   Login form - Any admin/owner can login using this button in the home page, to modify the webpage based on the tours and tour packages also for upgrading user interface. He has the credentials to add and delete the various travel destinations.

3.   Sign up/Sign in form – This is used by users of the website. The user can create an account if one already hasn't and view his profile , change his password , view the packages available on the website and his tour history and obtain the ticket in pdf format.

4.   Tour Package - This is used by both the users and the admin of the website. The admin can add new travel destinations, delete tour packages if there are any inconvenience in those tour packages. He can add details regarding the various tour packages. The user can view the details about the various tour packages, select his travel destination and book tickets. The tickets will be generated in pdf format.

4.2 Modules that admin can access

1. User module - The admin is allowed to add a new user, remove existing user, reset the password of a particular user.

2. Database module -Refactor the records (Delete or edit the existing Packages and create new packages).

5. IMPLEMENTATION

5.1 SQL (Database code)

SQL is Structured Query Language, which is a computer language for storing, manipulating and retrieving data stored in a relational database.SQL is the standard language for Relational Database System.

5.1.1 Table Creation

CREATE DATABASE tms;

CREATE TABLE `admin` (
 `id` int(11) NOT NULL,
 `UserName` varchar(100) DEFAULT NULL,
 `Password` varchar(100) DEFAULT NULL,
 `updationDate` timestamp NULL DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

CREATE TABLE `tblusers` (
 `id` int(11) NOT NULL,
 `FullName` varchar(100) DEFAULT NULL,
 `MobileNumber` char(10) DEFAULT NULL,
 `EmailId` varchar(70) DEFAULT NULL,
 `Password` varchar(100) DEFAULT NULL,
 `RegDate` timestamp NOT NULL DEFAULT current_timestamp(),
 `UpdationDate`   timestamp   NULL DEFAULT   NULL ON   UPDATE
current_timestamp()) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

CREATE TABLE `tbltourpackages` (
 `PackageId` int(11) NOT NULL,
 `state` varchar(20) NOT NULL,
 `PackageName` varchar(200) DEFAULT NULL,
 `PackageType` varchar(200) DEFAULT NULL,
 `PackageLocation` varchar(100) DEFAULT NULL,
 `PackagePrice` int(11) DEFAULT NULL,
 `PackageFeatures` varchar(255) DEFAULT NULL,
 `PackageDetails` mediumtext DEFAULT NULL,
 `PackageImage` varchar(100) DEFAULT NULL,
 `CreationDate` timestamp NOT NULL DEFAULT current_timestamp(),

`UpdationDate` timestamp NULL DEFAULT NULL ON UPDATE current_timestamp()
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

CREATE TABLE `tblbooking` (
 `BookingId` int(11) NOT NULL,
 `PackageId` int(11) DEFAULT NULL,
 `UserEmail` varchar(100) DEFAULT NULL,
 `PeopleCount` int(10) NOT NULL,
 `TotalPrice` int(10) NOT NULL,
 `FromDate` varchar(100) DEFAULT NULL,
 `ToDate` varchar(100) DEFAULT NULL,
 `RegDate` timestamp NOT NULL DEFAULT current_timestamp(),
 `status` int(11) DEFAULT NULL,
 `comment` varchar(100) NOT NULL,
 `CancelledBy` varchar(5) DEFAULT NULL,
 `UpdationDate` timestamp NULL DEFAULT NULL ON UPDATE current_timestamp()
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

CREATE TABLE `tblenquiry` (
 `id` int(11) NOT NULL,
 `FullName` varchar(100) DEFAULT NULL,
 `EmailId` varchar(100) DEFAULT NULL,
 `MobileNumber` char(10) DEFAULT NULL,
 `Subject` varchar(100) DEFAULT NULL,
 `Description` mediumtext DEFAULT NULL,
 `PostingDate` timestamp NOT NULL DEFAULT current_timestamp(),
 `Status` int(1) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

CREATE TABLE `tblissues` (
 `id` int(11) NOT NULL,
 `UserEmail` varchar(100)   DEFAULT NULL,
`Issue` varchar(100) DEFAULT NULL,
 `Description` mediumtext DEFAULT NULL,
 `PostingDate` timestamp NOT NULL DEFAULT current_timestamp(),
 `AdminRemark` mediumtext DEFAULT NULL,
 `AdminRemarkDate`   timestamp   NULL DEFAULT   NULL ON   UPDATE
current_timestamp()) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

CREATE TABLE `tblpages` (
 `id` int(11) NOT NULL,
 `type` varchar(255) NOT NULL DEFAULT " ",
 `detail` longtext DEFAULT NULL

) ENGINE=MyISAM DEFAULT CHARSET=utf8mb4;

5.1.2 Stored Procedures

1. With the stored procedure we will get the values of the users table without passing parameters :

```
CREATE PROCEDURE 'GetUsers'()
NOT DETERMINISTIC
CONTAINS SQL SQL SECURITY DEFINER
BEGIN
    Select * from tblusers;
END
```

2. With the stored procedure we will get the values of the users table with passing parameters :

```
CREATE PROCEDURE 'GetUsersByEmail'(IN 'uEmail' VARCHAR(255))
NOT DETERMINISTIC
CONTAINS SQL SQL SECURITY DEFINER
BEGIN
    Select * from tblusers where emailId = uEmail;
END
```

3. With the stored procedure we will get the values of the booking table with passing parameters :

```
CREATE PROCEDURE 'GetBookingByEmail'(IN 'uEmail' VARCHAR(100))
NOT DETERMINISTIC
CONTAINS SQL SQL SECURITY DEFINER
BEGIN
    Select * from tblbooking where emailId = uEmail;
END
```

5.1.3 Triggers

1. This trigger generates an exception if a duplicate entry is made by the user :

```
DELIMITER $$
CREATE TRIGGER `TRIGON_DUPLICATE_EMAIL` BEFORE INSERT ON `tblusers` FOR
EACH ROW BEGIN
    IF EXISTS (SELECT EmailId FROM tblusers WHERE tblusers.EmailId = NEW.EmailId) THEN
SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'This EmailId already exists, please
```

enter a different EmailId!';   END IF;
   END
$$
DELIMITER ;

2. This trigger generates an exception if the email that is to be inserted is not in the right format
    :

DELIMITER $$
CREATE TRIGGER `email_user` BEFORE INSERT ON `tblusers` FOR EACH ROW BEGIN
  IF NEW.emailid NOT LIKE '_%@_%.__%' THEN
    SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'Email field is not valid';
END IF;
END
$$
DELIMITER ;

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

Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

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

CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple web pages to share formatting by specifying the relevant CSS in a separate .css file, and reduce complexity and repetition in the structural content.

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

Alongside HTML and CSS, JavaScript is one of the core technologies of the World Wide Web. JavaScript enables interactive web pages and is an essential part of web applications. The vast majority of websites use it, and major web browsers have a dedicated JavaScript engine to execute it.

As a multi-paradigm language, JavaScript supports event-driven, functional, and imperative (including object-oriented and prototype-based) programming styles. It has APIs for working with text, arrays, dates, regular expressions, and the DOM, but the language itself does not include any I/O, such as networking, storage, or graphics facilities. It relies upon the host environment in which it is embedded to provide these features.

PHP: Hypertext Preprocessor (or simply **PHP**) is ageneral-purpose programming language originally designed forweb development. It was originally created byRasmusLerdorf in 1994; the PHPreference implementation is now produced by The PHP Group. PHP originally stood for *Personal Home Page*, but it now stands for therecursiveinitialism*PHP: Hypertext Preprocessor*.

PHP code may be executed with acommand line interface (CLI), embedded intoHTML code, or used in combination with variousweb template systems, webcontent management systems, andwebframeworks. PHP code is usually processed by a PHPinterpreter implemented as amodule in a web server or as aCommon Gateway Interface (CGI) executable. The web server outputs the results of the interpreted and executed PHP code, which may be any type of data, such as generated HTML code or binary image data. PHP can be used for many programming tasks outside of the web context, such asstandalonegraphical applications and roboticdrone control.

5.2 PHP (Front End Code)

HOMEPAGE
<!DOCTYPE HTML>

```html
<html>
<head>
<title>TMS | Travel Management System</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<meta name="keywords" content="Travel Management System In PHP" />
<style>
.errorWrap
{
padding: 10px;
margin: 0 0
20px 0;
background
: #fff;      border-left: 4px solid
#dd3d36;
   -webkit-box-shadow: 0 1px 1px 0 rgba(0,0,0,.1);       box-
shadow: 0 1px 1px 0 rgba(0,0,0,.1);
}
.succWrap{
padding: 10px;
margin: 0 0
20px 0;
background
: #fff;      border-left: 4px solid
#5cb85c;
   -webkit-box-shadow: 0 1px 1px 0 rgba(0,0,0,.1);       box-
shadow: 0 1px 1px 0 rgba(0,0,0,.1);
}
               </style>
</head>
<body>
<!-- top-header -->
<div class="top-header">
<?php include('includes/header.php');?>
<div class="privacy">
        <div class="container">
<?php
$pagetype=$_GET['type'];
$sql = "SELECT type,detail from tblpages where type=:pagetype";
$query = $dbh -> prepare($sql);
$query->bindParam(':pagetype',$pagetype,PDO::PARAM_STR);
$query->execute();
$results=$query->fetchAll(PDO::FETCH_OBJ);
```

```php
$cnt=1;
if($query->rowCount() > 0)
{ foreach($results
as $result) {
?>
</div>
<!--- /privacy ---->
<!--- footer-top ---->
<!--- /footer-top ---->
<?php include('includes/footer.php');?>
<!-- signup -->
<?php include('includes/signup.php');?>
<!-- //signu -->
<!--signin -->
<?php include('includes/signin.php');?>
<!-- //signin -->
<!-- write us -->

<?php include('includes/write-us.php');?>

</body>

</html>
```

LOGIN
```php
<?php
session_start();
if(isset($_POST['signin']))
{
$email=$_POST['email'];
$password=md5($_POST['password']);
$sql ="SELECT EmailId,Password FROM tblusers WHERE EmailId=:email and
Password=:password";
$query= $dbh -> prepare($sql);
$query->bindParam(':email', $email, PDO::PARAM_STR);
$query->bindParam(':password', $password, PDO::PARAM_STR);
$query->execute();
$results=$query->fetchAll(PDO::FETCH_OBJ);
if($query->rowCount() > 0)
{
$_SESSION['login']=$_POST['email'];          echo          "<script
type='text/javascript'>document.location = 'package-list.php';
</script>"; } else{ echo "<script>alert('Invalid
Details');</script>";
```

```
}
}
?>
<div    class="modal fade"    id="myModal4"        tabindex="-1" role="dialog"
arialabelledby="myModalLabel">
<div class="modal-dialog" role="document">
<div class="modal-content modal-info">
<div    class="modal-header"><button type="button"    class="close"    data-dismiss="modal"
arialabel="Close"><span aria-hidden="true">×</span></button></div>
<div class="modal-body modal-spa">
<div class="login-grids">
<div class="login">
<div class="login-left">
<ul>
<li><a class="fb" href="#"><i></i>Facebook</a></li>
<li><a class="goog" href="#"><i></i>Google</a></li>
</ul>
</div>
<div class="login-right">
<form method="post">
<h3>Signin with your account </h3>
<input type="text" name="email" id="email" placeholder="Enter your Email"
required=""><input type="password" name="password" id="password"
placeholder="Password" value="" required="">
<h4><a href="forgot-password.php">Forgot password</a></h4>
<input type="submit" name="signin" value="SIGNIN">
</form>
</div>
<div class="clearfix"></div>
</div>
<p>By logging in you agree to our <a href="page.php?type=terms">Terms and Conditions</a>
and <a href="page.php?type=privacy">Privacy Policy</a></p></div>
```

ADD TOUR PACKAGES

```
<?php include('server1.php'); ?>
<!DOCTYPE html>
<html>
<head>
        <title>tourpackages</title>
        <link        rel="stylesheet"        type="text/css"
        href="tourpackages.css"></head>
<body>
```

```html
<form method="post" action="tour.php">
        <div class="image">
        </div>
        <div class="header">
        <h1>Add Tour Package</h1> </div>
<div class="input-group">
        <label>id      </label>
        <input type="text" name="id">
        </div>
        <div class="input-group">
            <label>name     </label>
            <input type="text" name="name">
        </div>
        <div class="input-group">
            <label>Location</label>
            <input type="text" name="location">
        </div>
        <div class="input-group">
            <label>Package type </label>
            <input type="text" name="course">
        </div>


        </div>
        <div class="input-group">
            <label>Contact   </label>
            <input type="text" name="contact">
        </div>
        <div>
            <div class="input-group">
        <button type="submit" name="add" class="btn">Add</button>

        </div>
</form>
</body>
</html>
```

DELETE TOUR PACKAGES
```php
<?php
if(isset($_POST['Delete']))
{
        $db = mysqli_connect('localhost','root','','test');
```

```php
        $id=$_POST['id'];
        $query="DELETE FROM devents WHERE `id` = $id";
        $result=mysqli_query($db,$query);
        if($result)
        {       echo'<p><font       color=white           size=12pt>Data
                Deleted</font></p>';
        } else{ echo'Data
        Not Deleted'; }
        mysqli_cl ose($db);

}
?>
<html>
<head>

        <title>deletede</title>
        <link rel="stylesheet" type="text/css" href="deletedestyle.css">
</head>

<body>
        <form method="POST" action="deletede.php">

                <div class="header">
                    <h2>Delete Tour Packages</h2>
                </div>

                <div class="input-group">
                        <label>Id</label>
                    <input type="text" name="id">
                </div>

                <div class="input-group">
                                <button type="submit" name="Delete"
class="btn">Delete</button>
                </div>
        </form>
</body>
</html>
```

DISPLAY TOUR PACKAGES

```php
<?php
session_start();
error_reporting(0);
```

include('includes/confi
g.php'); ?>

```
<!DOCTYPE HTML>
<html>
<head>
<title>TMS  | Package List</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
</head>
<body>
<?php include('includes/header.php');?>
<!--- banner ---->
<div class="banner-3">
        <div class="container">
                <h1 class="wow zoomIn animated animated" data-wow-delay=".5s"
style="visibility: visible; animation-delay: 0.5s; animation-name: zoomIn;"> TMS- Package
List</h1>
        </div>
</div>
<!--- /banner ---->
<!--- rooms ---->
<div class="rooms">
        <div class="container">

                <div class="room-bottom">
                        <h3>Package List</h3>


<?php $sql = "SELECT * from tbltourpackages";
$query = $dbh->prepare($sql);
$query->execute();
$results=$query->fetchAll(PDO::FETCH_OBJ);
$cnt=1;
if($query->rowCount() > 0)
{
foreach($results as $result)
{            ?>
        <div class="rom-btm">
<div class="col-md-3 room-left wow fadeInLeft animated" data-wow-delay=".5s">
<imgsrc="admin/pacakgeimages/<?php    echo    htmlentities($result->PackageImage);?>"
class="imgresponsive" alt="">
</div>
```

Department of CS&E, 2022-23

```
<div class="col-md-6 room-midle wow fadeInUp animated" data-wow-delay=".5s">
<h4>Package Name: <?php echo htmlentities($result->PackageName);?></h4>
<h6>Package Type :<?php echo htmlentities($result->PackageType);?></h6>
<p><b>Package Location :</b><?php echo htmlentities($result->PackageLocation);?></p><p><b>Features</b><?php echo htmlentities($result>PackageFetures);?></p></div>
<div class="col-md-3 room-right wow fadeInRight animated" data-wow-delay=".5s">
<h5>Rs. <?php echo htmlentities($result->PackagePrice);?></h5>
<a href="package-details.php?pkgid=<?php echo htmlentities($result->PackageId);?>" class="view">Details</a></div>
<div class="clearfix"></div>
        </div>

<?php }} ?>
</div>
<!--- /rooms ---->

<!--- /footer-top ---->
<?php include('includes/footer.php');?>
<!-- signup -->
<?php include('includes/signup.php');?>
<!-- //signu -->
<!--signin -->
<?php include('includes/signin.php');?>
<!-- //signin -->
<!-- write us -->
<?php include('includes/write-us.php');?>
<!-- //write us -->
</body>
</html>
```

SEARCH PACKAGE TOUR
```
<?php
session_start();
error_reporting(0); include('includes/config
.php'); if(isset($_POST['submit2']))
{
$pid=intval($_GET['pkgid']);
$useremail=$_SESSION['login'];
$fromdate=$_POST['fromdate'];
$todate=$_POST['todate'];
$comment=$_POST['comment'];
```

```php
$status=0;
$sql="INSERT INTO tblbooking(PackageId,UserEmail,FromDate,ToDate,Comment,status)
VALUES(:pid,:useremail,:fromdate,:todate,:comment,:status)";
$query = $dbh->prepare($sql);
$query->bindParam(':pid',$pid,PDO::PARAM_STR);
$query->bindParam(':useremail',$useremail,PDO::PARAM_STR);                    $query-
>bindParam(':fromdate',$fromdate,PDO::PARAM_STR);
$query->bindParam(':todate',$todate,PDO::PARAM_STR);
$query->bindParam(':comment',$comment,PDO::PARAM_STR);
$query->bindParam(':status',$status,PDO::PARAM_STR);
$query->execute();
$lastInsertId    =       $dbh->lastInsertId();
if($lastInsertId)
{
$msg="Booked Successfully";
} else
{
$error="Something went wrong. Please try again";
}

}
?>
```

```html
<!DOCTYPE HTML>
<html>
<head>
<title>TMS | Package Details</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<script src="js/jquery-ui.js"></script>
<script>
        $(function() {
        $(          "#datepicker,#datepicker1"
        ).datepicker(); });
</script>
<style>
        .error
Wrap {    padding:
10px;    margin: 0
0
20px 0;    background:
#fff;    borderleft:
4px solid
#dd3d36;
```

```
   -webkit-box-shadow: 0 1px 1px 0 rgba(0,0,0,.1);      box-shadow: 0
1px 1px 0 rgba(0,0,0,.1);
}
.succWrap{    padding:
10px;    margin: 0 0
20px 0;
background
: #fff;       border-left: 4px solid
#5cb85c;
   -webkit-box-shadow: 0 1px 1px 0 rgba(0,0,0,.1);      box-shadow: 0
1px 1px 0 rgba(0,0,0,.1);
}
                    </style>
</head>
<body>
<!-- top-header -->
<?php include('includes/header.php');?>
<div class="banner-3">
<div class="container">
<h1 class="wow zoomIn animated animated" data-wow-delay=".5s" style="visibility:
visible; animation-delay: 0.5s; animation-name: zoomIn;"> TMS Package
Details</h1></div>
</div>
<!--- /banner ---->
<!---selectroom ---->
<div class="selectroom">
<div class="container">
<?php if($error){?><div      class="errorWrap"><strong>ERROR</strong>:<?php
     echo htmlentities($error); ?></div><?php }
else    if($msg){?><div     class="succWrap"><strong>SUCCESS</strong>:<?php    echo
htmlentities($msg); ?></div><?php }?>
<?php
$pid=intval($_GET['pkgid']);
$sql = "SELECT * from tbltourpackages where PackageId=:pid";
$query = $dbh->prepare($sql);
$query ->bindParam(':pid', $pid, PDO::PARAM_STR);
$query->execute();
$results=$query->fetchAll(PDO::FETCH_OBJ);
$cnt=1;
if($query->rowCount() > 0)
{
foreach($results as $result)
{           ?>
```

```
<form name="book" method="post">
<div class="selectroom_top">
<div class="col-md-4 selectroom_left wow fadeInLeft animated" data-wow-delay=".5s">
<imgsrc="admin/pacakgeimages/<?php    echo    htmlentities($result->PackageImage);?>"
class="imgresponsive" alt="">
</div>
<div class="col-md-8 selectroom_right wow fadeInRight animated" data-wow-delay=".5s">
<h2><?php echo htmlentities($result->PackageName);?></h2>
<p class="dow">#PKG-<?php echo htmlentities($result->PackageId);?></p>
<p><b>Package Type :</b><?php echo htmlentities($result->PackageType);?></p>
<p><b>Package Location :</b><?php echo htmlentities($result->PackageLocation);?></p>
<p><b>Features</b><?php echo htmlentities($result->PackageFetures);?></p>
<div class="ban-bottom">
<div class="bnr-right">
<label class="inputLabel">From</label>
<input class="date"   id="datepicker"      type="text"   placeholder="dd-mm-yyyy"
name="fromdate" required="">
</div>
<div class="bnr-right">
<label class="inputLabel">To</label>
<input class="date" id="datepicker1" type="text" placeholder="dd-mm-yyyy" name="todate"
required="">
</div>
</div>
<div class="clearfix"></div>
<div class="grand">
<p>Grand Total</p>
<h3>Rs.800</h3>
</div>
</div>
<h3>Package Details</h3>
<p    style="padding-top:    1%"><?php    echo    htmlentities($result-
>PackageDetails);?></p><div class="clearfix"></div>
</div>
<div class="selectroom_top">
<h2>Travels</h2>
<div class="selectroom-info animated wow fadeInUp animated" data-
wowduration="1200ms" datawow-delay="500ms" style="visibility: visible; animation-
duration: 1200ms; animation-delay: 500ms; animation-name: fadeInUp; margin-top: -70px">
<ul>
<li class="spe">
<label class="inputLabel">Comment</label>
```

```
<input class="special"        type="text"    name="comment"
required=""></li>
<?php if($_SESSION['login'])
{?>
<li class="spe" align="center">
<button        type="submit"        name="submit2"        class="btn-primary
btn">Book</button></li>
<?php } else {?>
<li class="sigi" align="center" style="margin-top: 1%">
<a  href="#"  data-toggle="modal"  data-target="#myModal4"  class="btn-primary  btn"  >
Book</a></li><?php } ?>
<div class="clearfix"></div>
</ul>
</div>
</div>
</form>
<?php }} ?>


        </div>
</div>
<!--- /selectroom ---->
<<!--- /footer-top ---->
<?php include('includes/footer.php');?>
<!-- signup -->
<?php include('includes/signup.php');?>
<!-- //signu -->
<!--signin -->
<?php include('includes/signin.php');?>
<!-- //signin -->
<!-- write us -->
<?php include('includes/write-us.php');?>
</body>
</html>
```

6. TESTING AND SNAPSHOTS

6.1 Testing

TestId	Test_Name	Input	Output	Result
1	Admin_login	Correct UserName and password	Login	Success
2	Denial	Incorrect UserName or password	Access denied	Failure
3	User_signIn	Correct EmailId and password	SignIn	Success
4	Denial	Incorrect EmailId or password , an account with the emailId doesn't exist.	Access denied	Failure
5	User_signUp	Enter Email, Full Name, Mobile Number, password	SignUp, Creates account	Success
6	Denial	Email not in format or duplicate Email	Error creating account	Failure
7	View Package	Choose state and click on submit and details of a package	View the package	Success
8	Book Package	Fill out the particulars and click on Book	Book a package	Success
9	Enquire	Fill the Email, Full name, Subject and description fields.	Post enquiry	Success
10	Post Issue	After signin, choose the issue and fill out the description	Post Issue	Success
11	Generate Ticket	After SignIn, enter the BookingId and click Download	The ticket is obtained in pdf format	Success
12	Error	Booking not confirmed by Admin	Error in printing the ticket	Failure
13	Create Package	All fields are filled and no primary key conflict	Package created	Success
14	Error	Not all the fields are filled or primary key conflict or server problem.	Error adding data to the database	Failure

15	View Tour History	After Signin click on tour history	View tour history of the user	Success
16	Change Password	The current password field is correct.	Password of the user is changed	Success

17	Error	Password fields are wrong.	Error in changing password	Failure

6.2 Snapshots

6.2.1 Home Page

Fig. 6.1

When we run the "Travel Management System" webpage first home page is displayed. The home page will appear as fig 6.1. If a user wants to access the different options of the website then he must use the explore button. Login button is for the admin.

6.2.2 About Us

Fig. 6.2

Fig 6.2 shows the information of our project Travel Management System. We can see the available options that the Travel Management System provides.

6.2.3 Package List
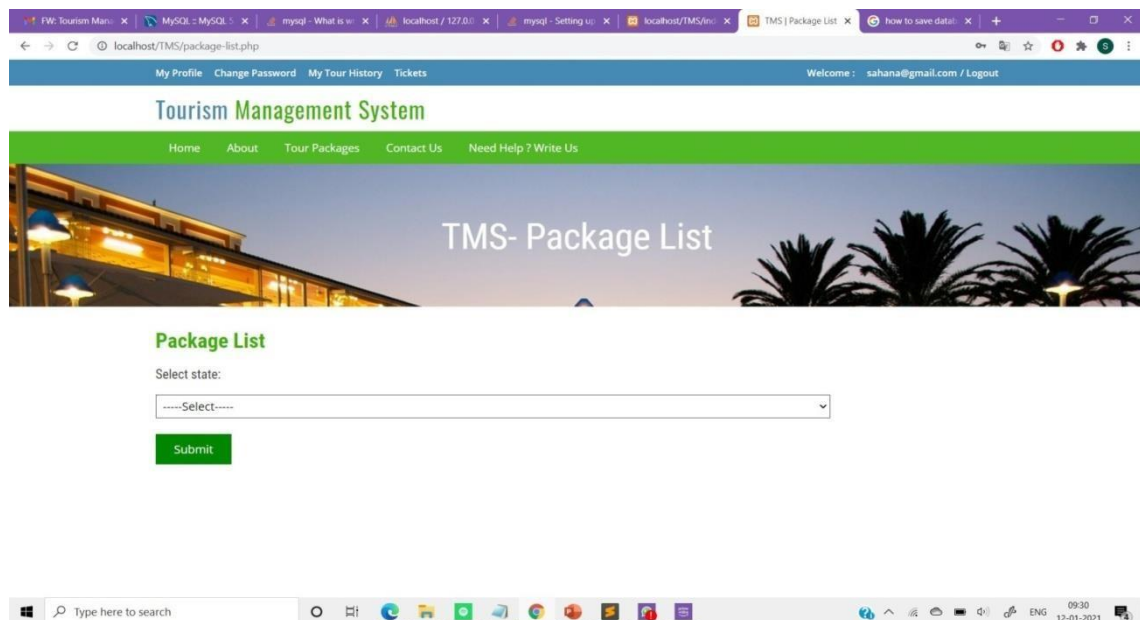
Fig 6.3

6.2.4 Package List by State
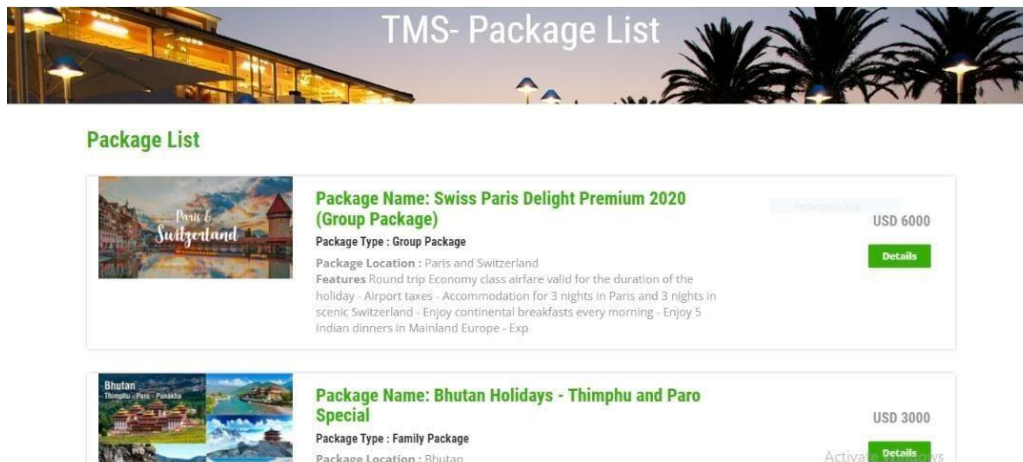
Fig 6.4

The package list shows the details about each tour packages in detail. This includes the price, facilities available, booking options, transportation facilities and so on.

Fig 6.4 shows the List of packages by state , it lists the name , location and cost of the package

6.2.5 Package Details
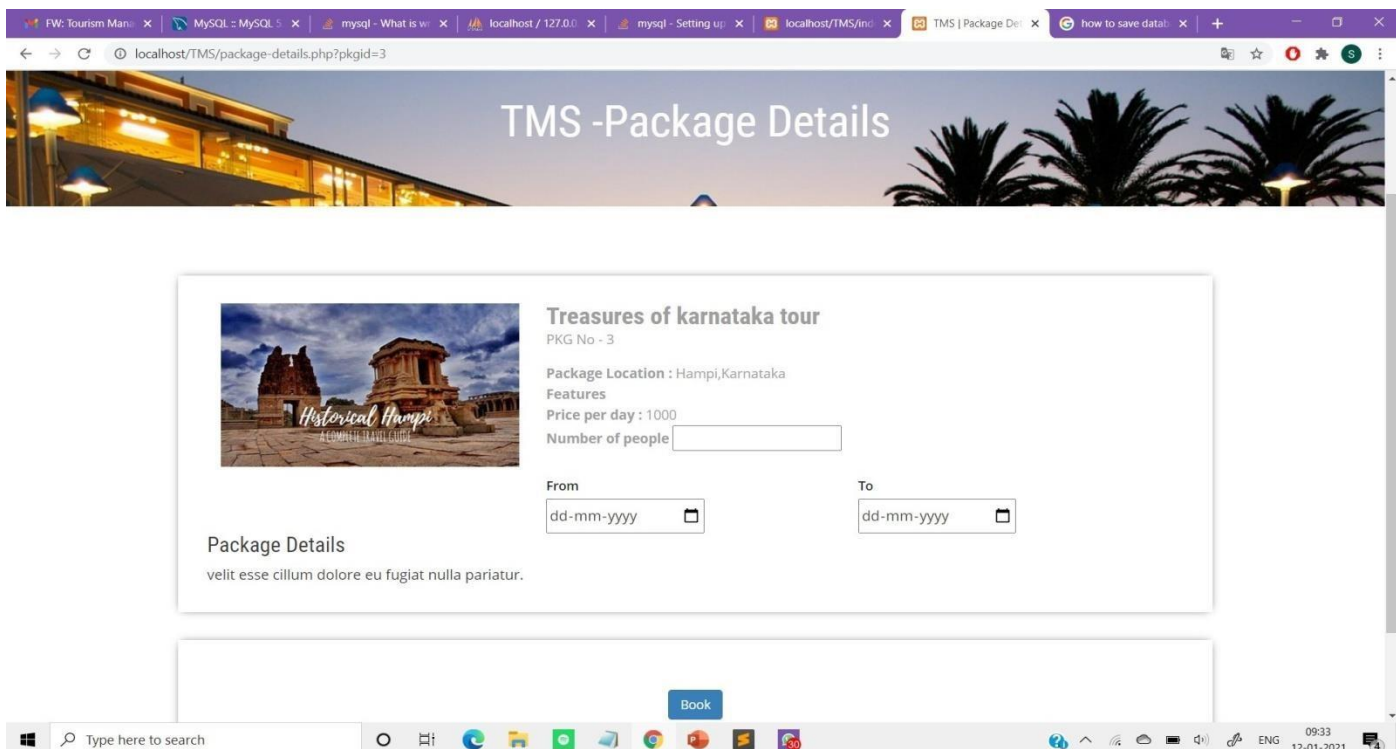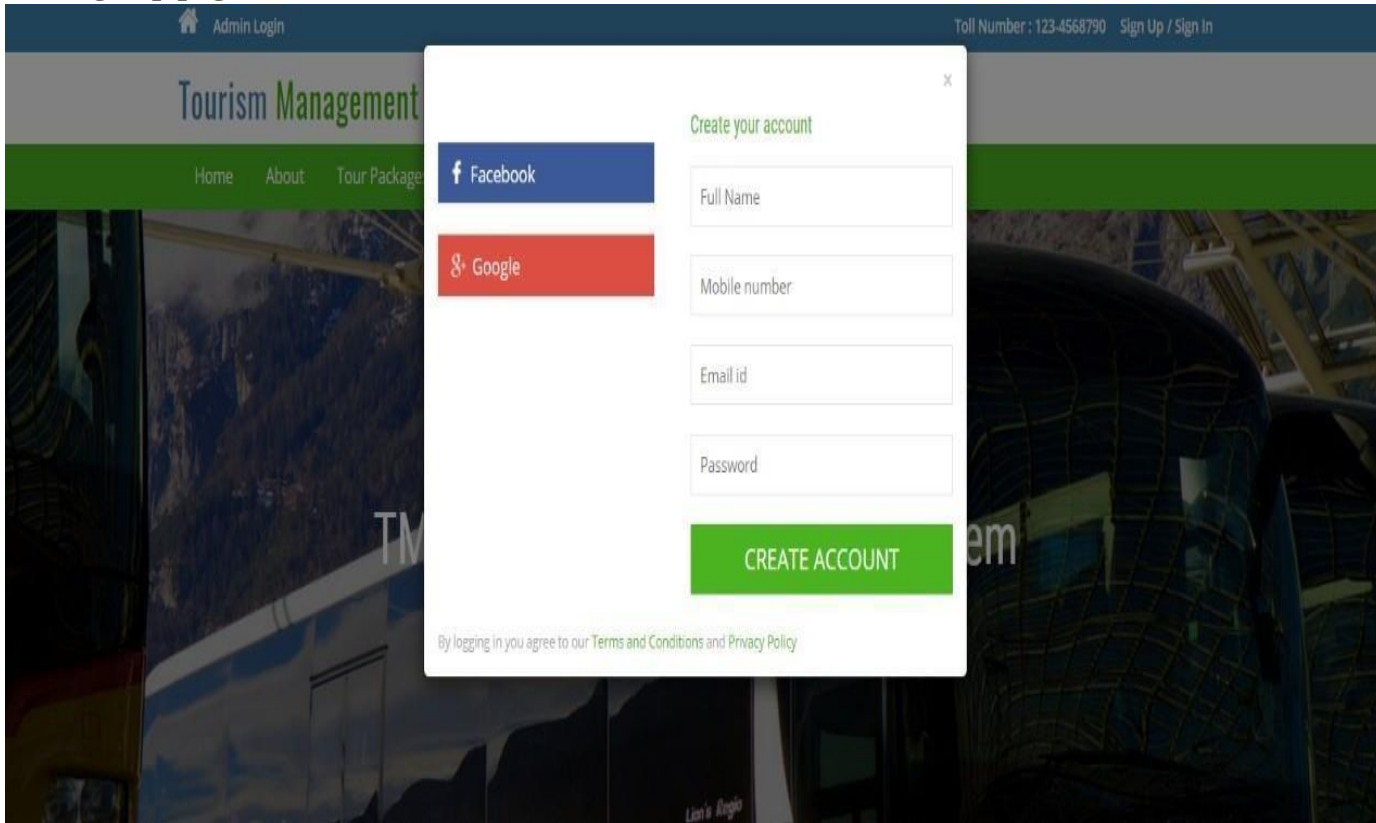
Fig 6.5

Fig 6.5 shows the detailed description and features of a selected package. The users should provide details such as number of people , from and to dates.

6.2.6 Sign up page

Fig 6.6

The sign up page is used for creating a new account for first time users. The users should provide their name, email-id, phone number and sign up.

6.2.7 Sign in page
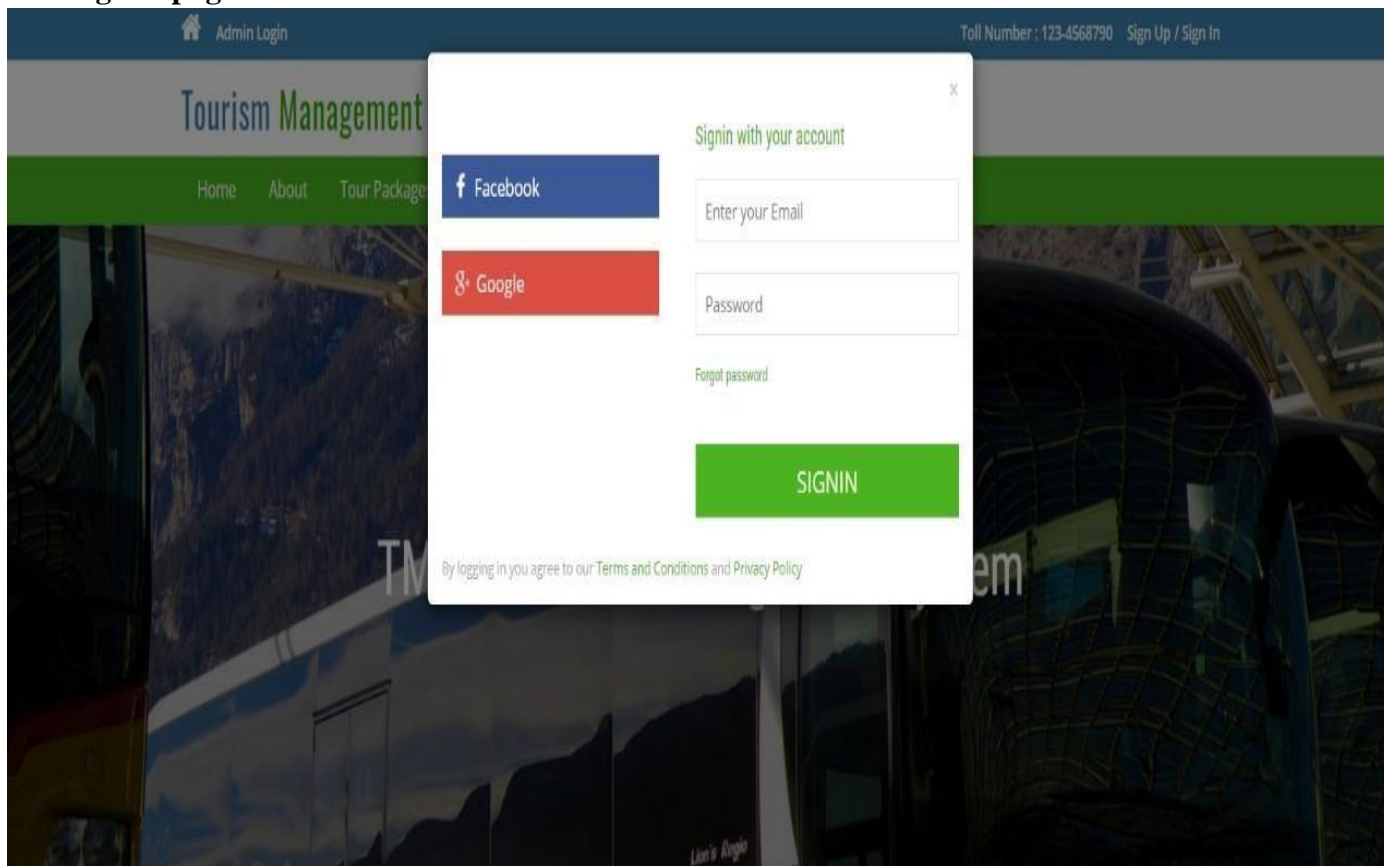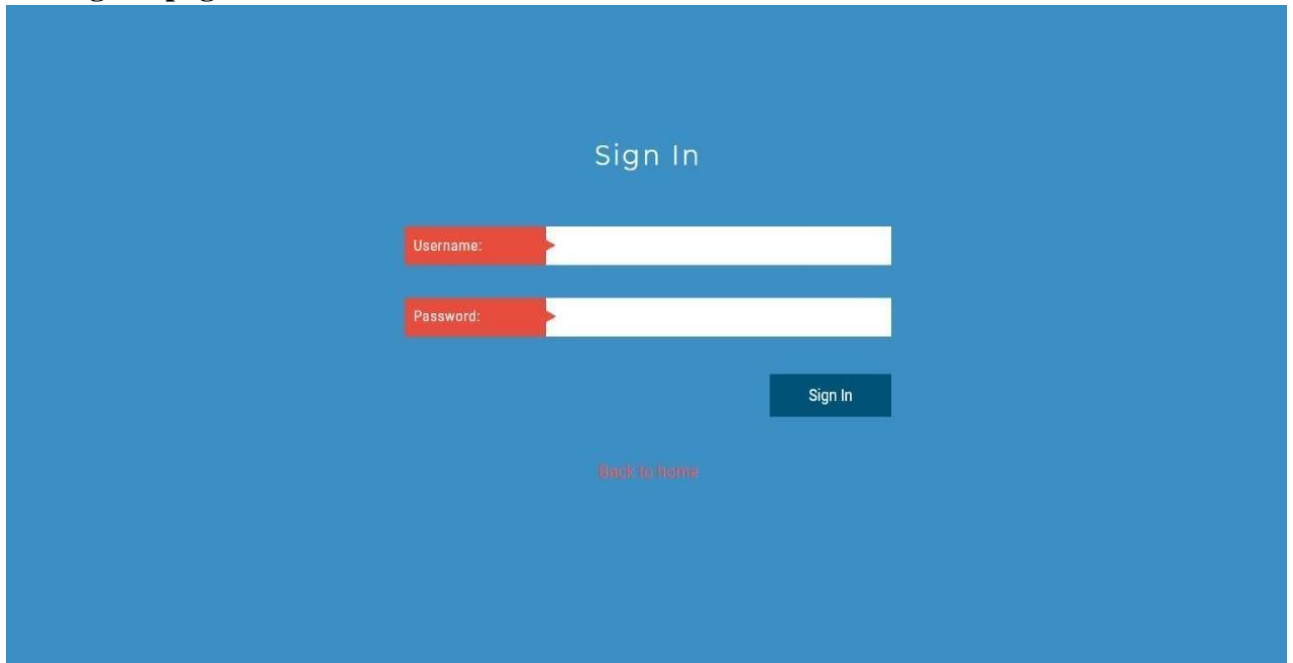
Fig 6.7

Once the user is signed up next time he visits the website he can directly sign in and use the

Travel Management System website.
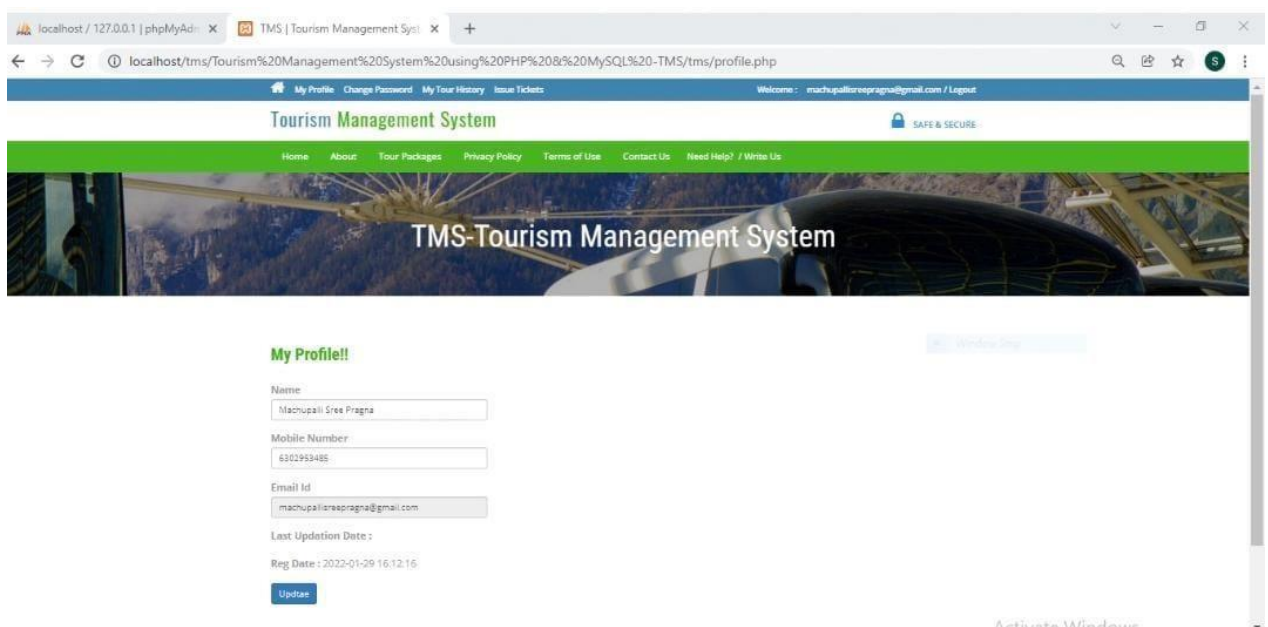
6.2.8 Sign in page

Fig 6.8

Fig 6.8 depicts the login page of the admin where after successful login the admin is redirected to the account.

6.2.9 My Profile Page

Fig 6.9

Fig 6.9 is the profile page of the user logged in which displays the information of the user who can also update details.

6.2.10 My Tour History Page

Fig 6.10

Fig 6.10 is the tour history of the user who has logged in. It displays information about the bookings done in the past and present.

6.2.11 Change Password Page

Fig 6.11

Fig 6.11 is the change password page where the user who has logged in can change password. **6.2.12 Issue Page**

Fig 6.12

Fig 6.12 is the issue page where the user who has logged in can select the type of issue and report to the admin
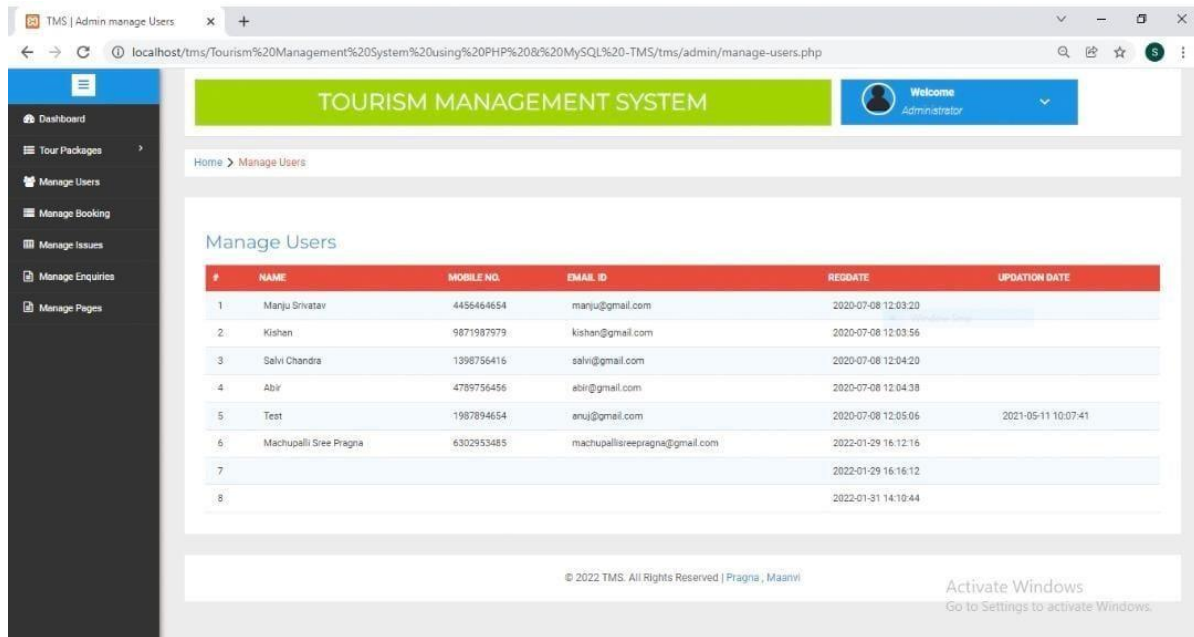
6.2.13 Manage users page

Fig 6.13

Fig 6.13 is the page where the admin can view the basic information of the users who has registered with the website.
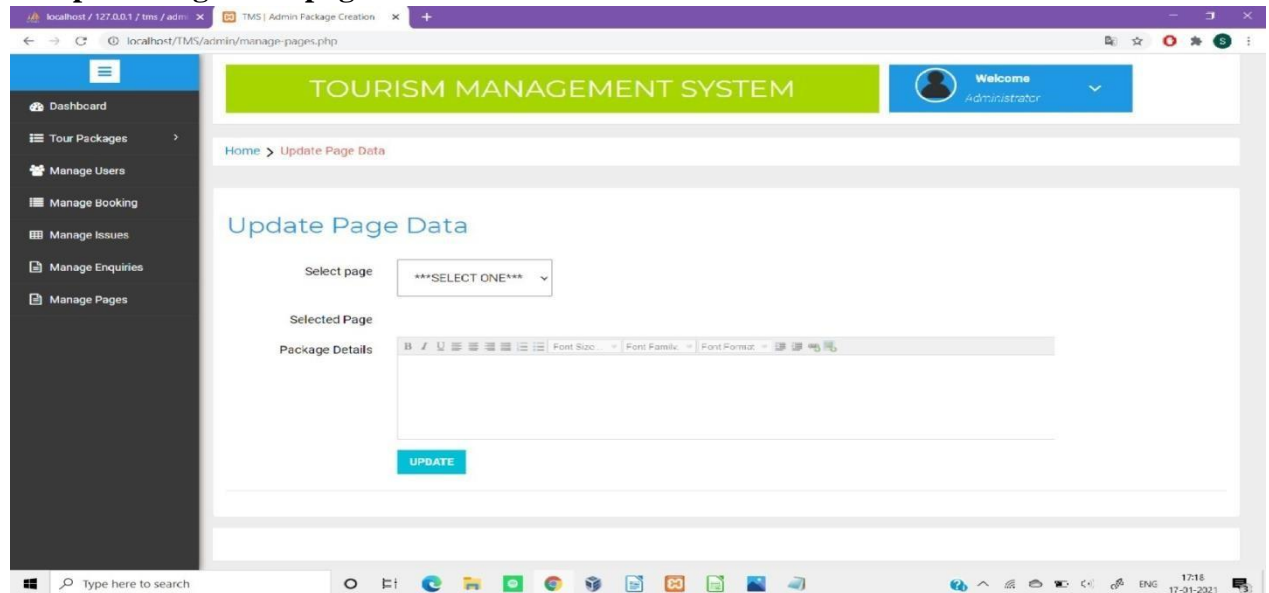
6.2.14 Update Page Data page

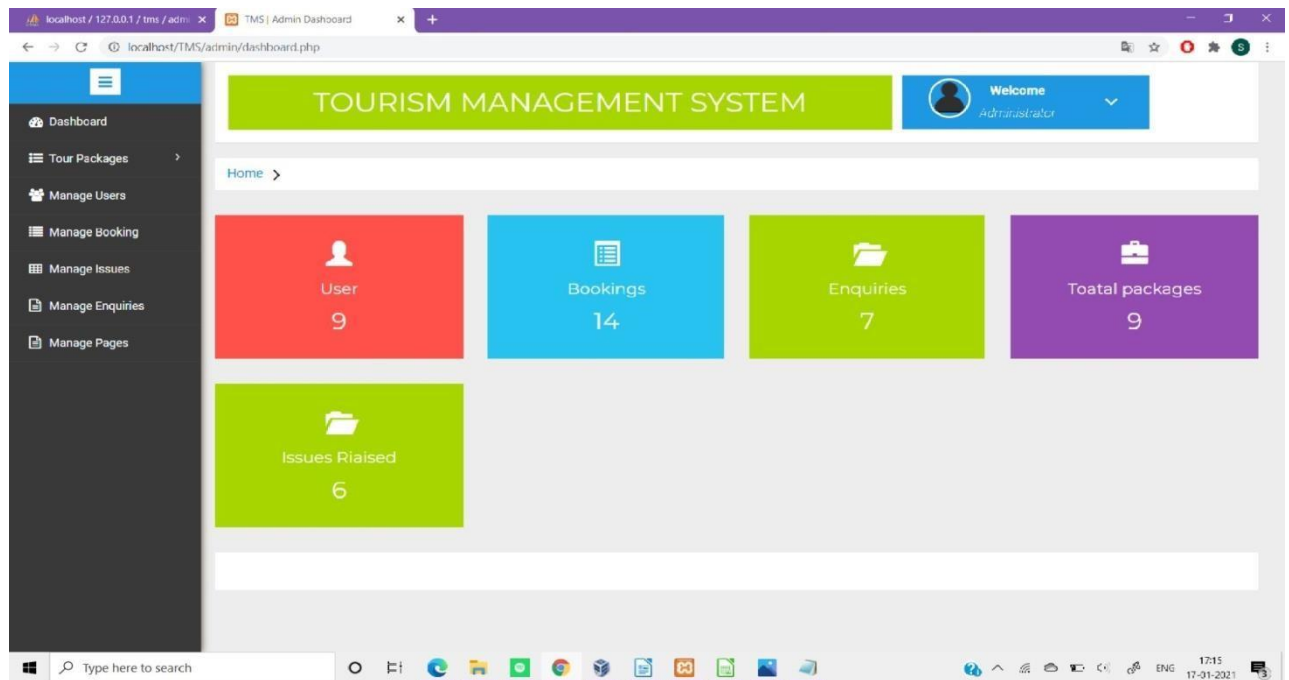

Fig 6.14

6.2.15 Admin Dashboard

Fig 6.15 Fig 6.15 is the Admin's dashboard.It gives
all the information about the different aspects of the website

7. APPLICATIONS

- This application will help in accessing the information related to the travel to the particular destination with great ease.

- The user can track the information related to their tours, tour history, profile through this application

- The user can obtain the tickets booked for a certain packages in pdf format with all the necessary information.

- Travel Management System would play a vital role in planning the perfect trip.

- The Travel Management System allows the user of the system access all the details such as location ,events , features etc.

- The main purpose is to help tourism companies to manage customer , packages etc.

- The system can also be used for both professional and business trips.

- The proposed system maintains centralized repository to make necessary travel arrangements and to retrieve information easily.

8. CONCLUSION

The development of Travel Management System website involved many phases. The approach used is a top-down approach, concentrating on what first, then how, later moving to the successive levels in detail. First phase started with the detailed study of the problems of the existing manual searching method to search different institutions and events nearby. These problems, information needs and activities were documented and later used as the basis for system design. The design phase was concerned primarily with the specification of the system elements in such a manner that best meets the requirements. To implement this design, a computer program was then written in HTML, PHP and CSS. It is hoped that effective implementation of this software product would eliminate the many problems of the existing system.

System allows users to browse through different categories such as various travel/tour packages, booking options. This is achieved through an easy to use graphical interface explore option. System allows users to view and search the items. The admin will have the privilege of maintaining the list of tour packages by adding new ones and deleting the unwanted items.