ABSTRACT

Our DBMS project is based on Complete Online Tourism System Management. It provides various information about the various activities such as studying tour destination, Planning the tour, Making Travel arrangements and providing accommodation.

As the name specifies "TOURISM MANAGEMENT SYSTEM" is a software developed for managing tour booking.
Identification of the drawbacks of the existing system leads to the designing of computerized system that will be compatible to the existing system with the system Which is more user friendly and more GUI oriented.
We can improve the efficiency of the system, thus overcome the drawbacks of the existing system.

- **Less human error.**
- **Strength and strain of manual labour can be reduced.**
- **High security.**
- **Data redundancy can be avoided to some extent.**
- **Data consistency.**
- **Easy to handle.**
- **Easy data updating.**
- **Easy record keeping.**
- **Backup data can be easily generated.**

CONTENTS

CHAPTER1

INTRODUCTION

The Tourism Management System is a stand-alone based application and maintains a centralized repository of all related information. The objective of this project is to develop a **system** that automates the processes and activities of a travel agency and customer details. The purpose is to design a **system** using which one can perform all operations related to traveling and sight-seeing.

1.1 DBMS

Database Management System (DBMS) is a software for storing and retrieving users' data while considering appropriate security measures. It consists of a group of programs which manipulate the database. The DBMS accepts the request for data from an application and instructs the operating system to provide the specific data. In large systems, a DBMS helps users and other third-party software to store and retrieve data. DBMS allows users to create their own databases as per their requirement. The term "DBMS" includes the user of the database and other application programs. It provides an interface between the data and the software application.

1.2 PHP

PHP is a general-purpose scripting language that is especially suited to server-side web development, in which case PHP generally runs on a web server. Any PHP code in a requested file is executed by the PHP runtime, usually to create dynamic web page content or dynamic images used on websites or elsewhere. PHP originally stood for Personal Home Page, but it now stands for the recursive backronym PHP. Hypertext Pre-processor. PHP code may be embedded into HTML code, or it can be used in combination with various web template systems, web content management system and web frameworks.

PHP Developer

PHP developers develop programs, applications, and web sites using the dynamic scripting language PHP. PHP is known for web development and business applications. Depending on job function, PHP developers may be classified as software developers or web developers.

Tags Description

<? php to open PHP sections
?> to close PHP sections
ECHO prints the lines

1.3 Mysql

MySql is a powerful database. It's very good and free of charge. Many developers in the world selected mysql and php for developing their website.
The MySQL database has become the world's most popular open source database because of its consistent fast performance, high reliability and ease of use. It's used in more than 6 million installations ranging from large corporations to specialized embedded applications on every continent in the world. (Yes, even Antarctica!)
Not only is MySQL the world's most popular open source database, it's also become the database of choice for a new generation of applications built on the LAMP stack (Linux, Apache, MySQL, PHP / Perl / Python.) MySQL runs

on more than 20 platforms including Linux, Windows, OS/X, HP-UX, AIX, Netware, giving you the kind of flexibility that puts you in control.
MySQL is very friendly to PHP, the most appreciated language for web development.

1.4 Xampp Server

XAMPP is a free and open-source cross-platform web server solution stack package developed by Apache Friends, consisting mainly of the Apache HTTP Server, MariaDB database, and interpreters for scripts written in the PHP and Perl programming languages. and it allows you to build WordPress site offline, on a local web server on your computer. This simple and lightweight solution works on Windows, Linux, and Mac.

XAMPP helps a local host or server to test its website and clients via computers and laptops before releasing it to the main server. It is a platform that furnishes a suitable environment to test and verify the working of projects based on Apache, Perl, MySQL database, and PHP through the system of the host itself.

ADVANTAGES

- In comparison to other web servers such as WAMP, it is simple to set up.
- It is Multi Cross-Platform, which implies it works on both Windows and Linux.
- Both a full and a standard version of XAMPP are available.
- It has a control panel that you can see contains start and stop buttons for specific mechanisms, such as Apache, which is running through its Control Panel.
- It also includes OpenSSL, phpMyAdmin, MediaWiki, Joomla, WordPress, and a lot of additional modules.

Components of XAMPP

The components that are included in the XAMPP are given below:

- Apache: Apache is a cross-platform HTTP web server. It is used to transport web material all over the world. If someone requests files, photos, or documents using their browser, HTTP servers will serve such assets to clients.
- PHP: The full form of PHP is Hypertext Preprocessor. PHP is a backend programming language that is most commonly used in web development. Users can use PHP to build dynamic websites and applications. It supports a variety of database management systems and may be installed on any platform. It was written in the C programming language.
- phpMyAdmin: It is a database administration tool for MariaDB.
- XAMPP Control Panel: The XAMPP Control Panel is a panel that assists in the operation and regulation of other XAMPP components.

CHAPTER2

SYSTEM SPECIFICATION

2.1 EXISTING SYSTEM

- In the present system a customer has to approach various agencies to find details of places and to book tickets.
- This often requires a lot of time and effort.
- A customer may not get the desired information from these offices and often the customer may be misguided.
- It is tedious for a customer to plan a particular journey and have it executed properly.

2.2 STUDY OF THE SYSTEM

To provide flexibility to the users, the interfaces have been developed that are accessible through a browser. The GUI'S at the top level have been categorized as

1. Administrative user interface
2. The operational or generic user interface

The 'Administrative user interface' concentrates on the consistent information that is practically, part of the organizational activities and which needs proper authentication for the data collection. Theseinterfaces help the administrators with all the transactional states like Data insertion, Data deletion and Date updation along with the extensive data search capabilities.

The 'Operational or generic user interface' helps the end users of the system in transactions through the existing data and required services. The operational user interface also helps the ordinary users in managing their own information in a customized manner as per the included flexibilities.

2.3 FUNCTIONAL REQUIREMENTS

NUMBER OF MODULES

AFTER CAREFUL ANALYSIS THE SYSTEM HAS BEEN IDENTIFIED TO HAVE THE FOLLOWING MODULES:

- ADMINISTRATOR MODULE
- USER (TRAVELER) MODULE
- GUEST USER

1. ADMINISTRATOR MODULE:

This module provides administrator related functionality. Administrator manages all information and has access rights to add, delete, edit and view the data related to places, travels, routes, bookings, Enquiries etc.

Packages Admin will create the packages and Manage the packages (Create, Update, delete)

Users-- Admin view all Information of all users.

Booking-- Admin will responsible for manage booking. Admin can confirm and cancel a booking of traveller.

Manage issues/ Complaints Admin can take action on any issue /complaint raised by user(traveller) and Put remark.

Manage Enquiries--admin can manage all enquiries raised by users(traveller).

Manage pages-- Admin can edit the info of all pages that are display on the website,

Dashboard-- Here admin can view all count of booking, issues, Enquiries and Users.

Change password--- Admin can change own password.

2.USER(TRAVELLER) MODULE:

Signup- User can register themselves for booking.

Sign-in- Here user can login with valid username and password.

Forgot Password-User can recover his/her own password.

My Profile- user can update own profile.

Tour history-After login user can book any tour that will show in Tour history. User can cancel his/her booking before 24 hr of travelling.

Change Password---- User can own Password.

Write-use-Here user can raise any issue related to booking. Cancelation etc.

3.Guest MODULE:

Guest user can visit the website and view the all content of website. Guest user can also Enquiry.

2.4 SYSTEM SPECIFICATIONS

REQUIREMENT SPECIFICATION

A Software requirements definition is an abstract description of the service which the system should provide, and the constraints and which system must operate. It should only specify the external behaviour of the system.

SOFTWARE & HARDWARE CONFIGURATIONS:

SOFTWARE CONFIGURATION

- Operating system: WINDOWS 11
- Programming language: HTML/CSS, PHP
- Front-End: HTML/CSS
- Back-End: MYSQL
- Web Server: XAMPP SERVER

HARDWARE CONFIGURATION

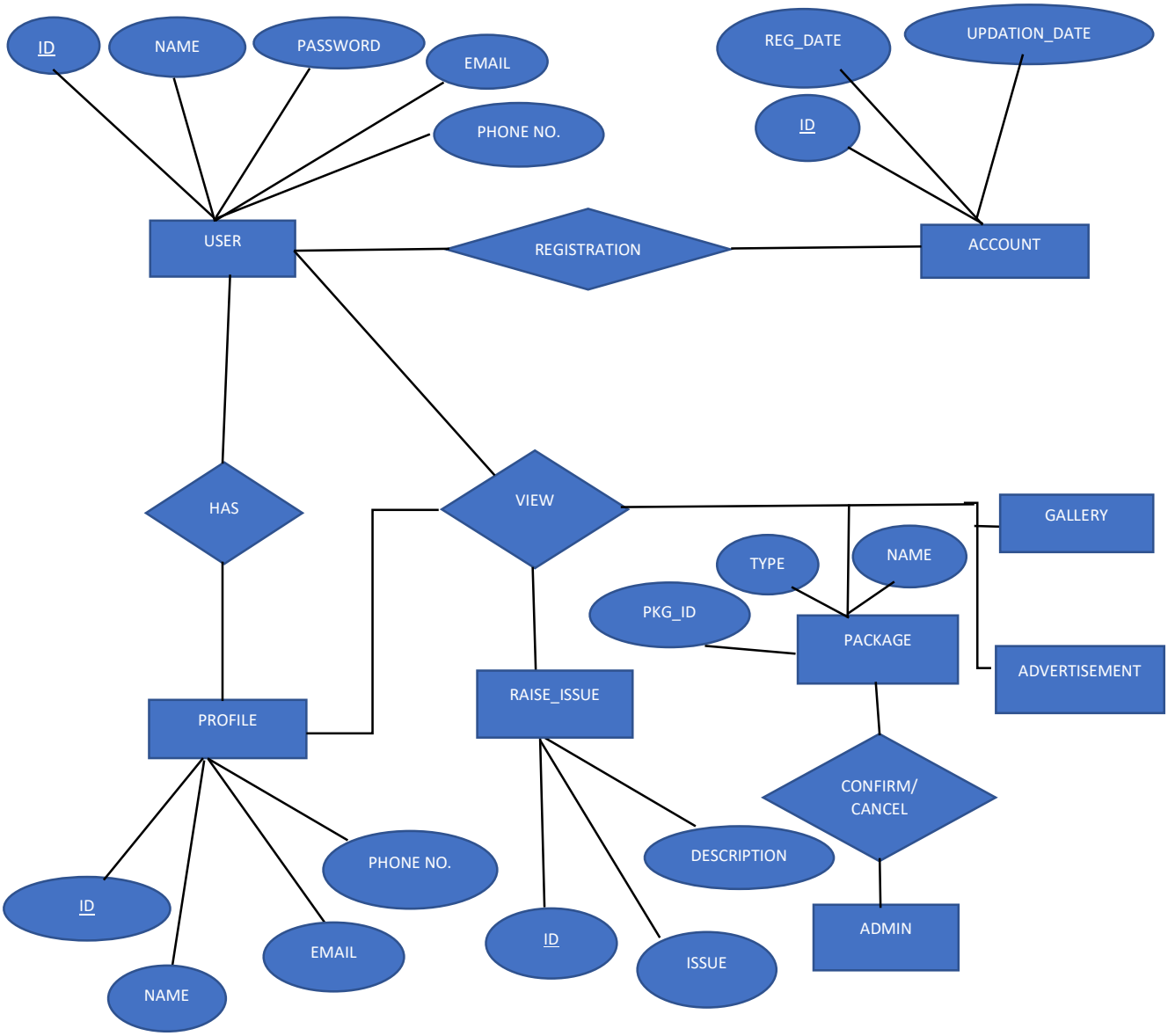
- Intel CORE i5(Processor)
- 8GB Ram
- Hard Disk 1TB
- Microsoft Compatible 101

CHAPTER3

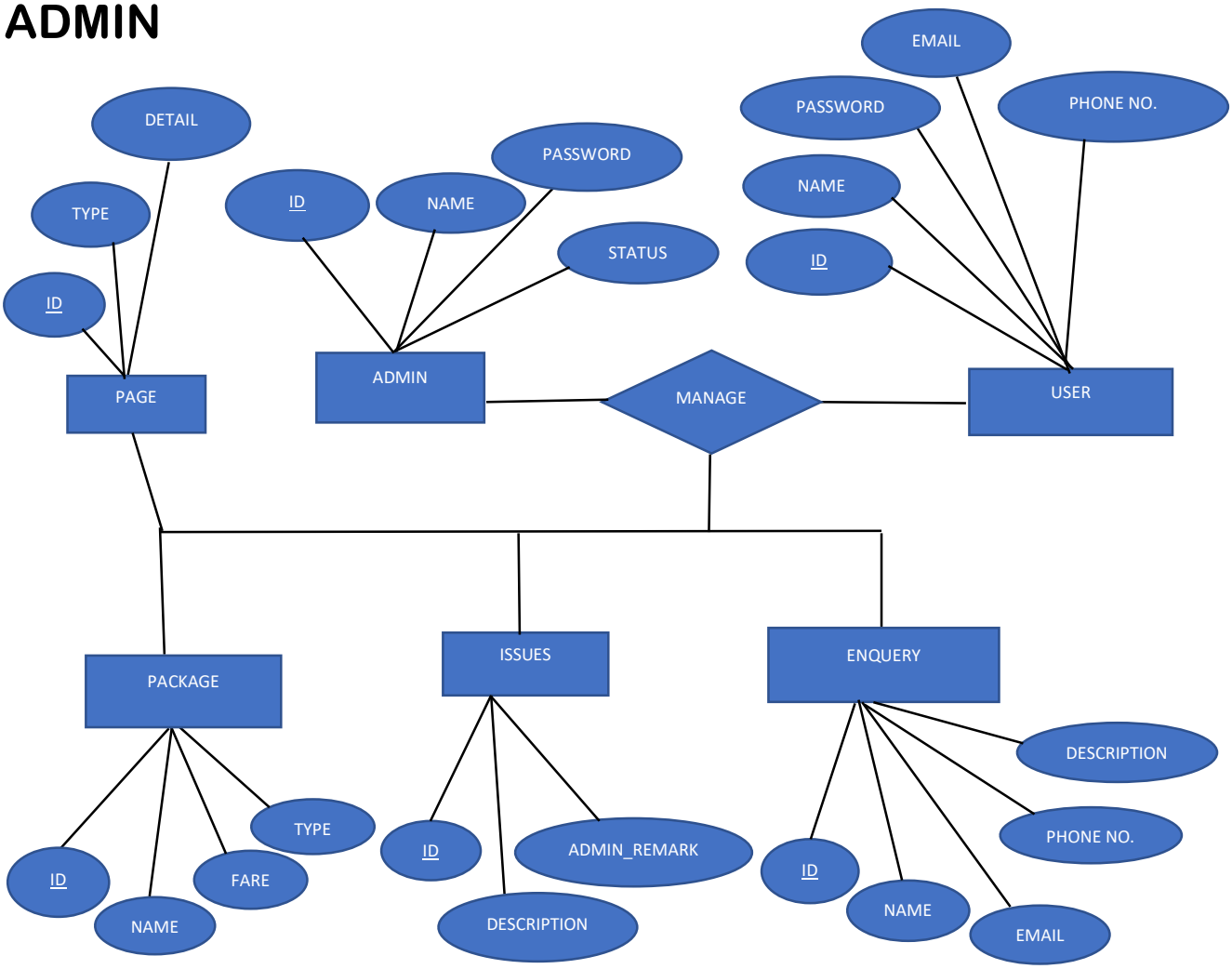
DESIGN APPROACH

3.1 E-R DIAGRAMS

USERS

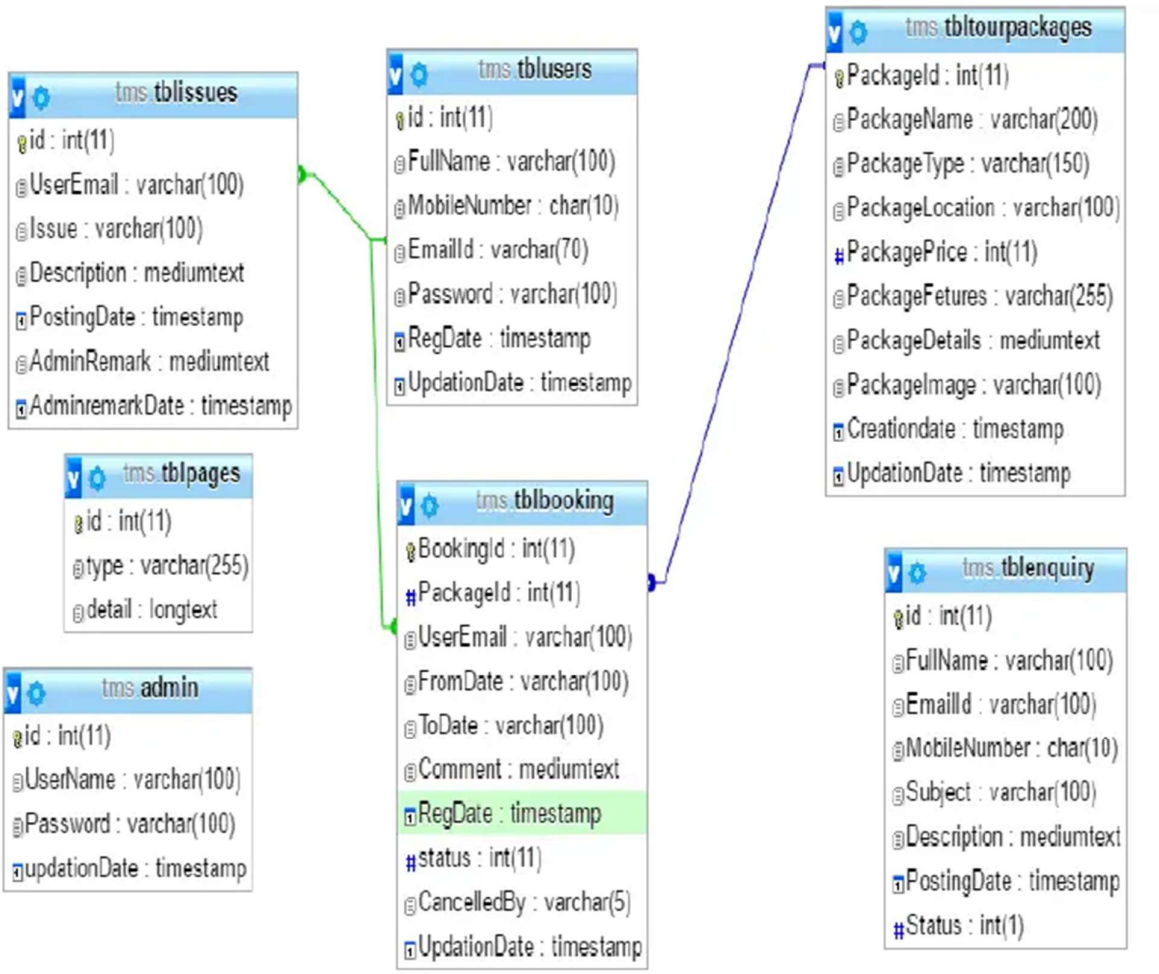


ADMIN


EMAIL
PASSWORD
PHONE NO.
DETAIL
PASSWORD
NAME
TYPE
ID
NAME
STATUS
ID
ID
ADMIN
MANAGE
USER
PAGE
PACKAGE
ISSUES
ENQUERY
DESCRIPTION
TYPE
ID
ADMIN_REMARK
PHONE NO.
ID
FARE
ID
NAME
DESCRIPTION
ID
NAME
EMAIL

3.2 SCHEMATIC DIAGRAM

3.3 DATABASE DESIGN



3.4 CODES

SQL CODE

```
-- Database: `tms`
--
-- --------------------------------------------------------
-- Table structure for table `admin`
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=latin1;
-- Dumping data for table `admin`
INSERT INTO `admin` (`id`, `UserName`, `Password`, `updationDate`) VALUES
(1, 'admin', 'f925916e2754e5e03f75dd58a5733251', '2017-05-13 11:18:49');

-- --------------------------------------------------------
-- Table structure for table `tblbooking`
CREATE TABLE `tblbooking` (
  `BookingId` int(11) NOT NULL,
  `PackageId` int(11) DEFAULT NULL,
  `UserEmail` varchar(100) DEFAULT NULL,
  `` FromDate varchar(100) DEFAULT NULL,
  `ToDate` varchar(100) DEFAULT NULL,
  `Comment` mediumtext DEFAULT NULL,
  `RegDate` timestamp NULL DEFAULT current_timestamp(),
  `status` int(11) DEFAULT NULL,
  `CancelledBy` varchar(5) DEFAULT NULL,
  `UpdationDate` timestamp NULL DEFAULT NULL ON UPDATE current_timestamp()
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
-- Dumping data for table `tblbooking`
```

```
INSERT INTO `tblbooking` (`BookingId`, `PackageId`, `UserEmail`, `FromDate`, `ToDate`, `Comment`, `RegDate`, `status`, `CancelledBy`, `UpdationDate`) VALUES

(2, 1, 'anuj@gmail.com', '05/18/2017', '05/31/2017', '\"Lorem ipsum dolor sit amet, cpariatur. Excepteur sint ', '2017-05-13 19:01:10', 2, 'u', '2017-05-13 21:30:23'),

(3, 2, 'anuj@gmail.com', '05/16/2017', '05/31/2017', 'casf esd sg gd gdfh df', '2017-05-13 20:20:01', 2, 'a', '2017-05-13 23:04:40'),

(4, 1, 'anuj@gmail.com', '05/16/2017', '05/31/2017', 'dwef  fwe', '2017-05-13 20:32:54', 2, 'a', '2017-05-13 21:36:39'),

-- --------------------------------------------------------
```

-- Table structure for table `tblenquiry`

```
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 NULL DEFAULT current_timestamp(),
  `Status` int(1) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

-- Dumping data for table `tblenquiry`

INSERT INTO `tblenquiry` (`id`, `FullName`, `EmailId`, `MobileNumber`, `Subject`, `Description`, `PostingDate`, `Status`) VALUES

-- --------------------------------------------------------
```

-- Table structure for table `tblissues`

```
CREATE TABLE `tblissues` (
  `id` int(11) NOT NULL,
  `UserEmail` varchar(100) DEFAULT NULL,
  `Issue` varchar(100) DEFAULT NULL,
  `Description` mediumtext DEFAULT NULL,
  `PostingDate` timestamp NULL DEFAULT current_timestamp(),
  `AdminRemark` mediumtext DEFAULT NULL,
  `AdminremarkDate` timestamp NULL DEFAULT NULL ON UPDATE current_timestamp()
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

-- Dumping data for table `tblissues`

INSERT INTO `tblissues` (`id`, `UserEmail`, `Issue`, `Description`, `PostingDate`, `AdminRemark`, `AdminremarkDate`) VALUES
```

(5, 'sarita@gmail.com', 'Cancellation', 'tbt 3y 34y4 3y3hgg34t', '2017-05-14 05:12:14', 'sg sd gs g sdgfs ', '2017-05-14 07:52:07'),

(6, 'demo@test.com', 'Refund', 'demo test.com demo test.comdemo test.comdemo test.comdemo test.com', '2017-05-14 07:45:37', NULL, '0000-00-00 00:00:00'),

(7, 'abc@g.com', 'Refund', 'test test ttest test ttest test ttest test ttest test ttest test t', '2017-05-14 07:56:46', 'vetet ert erteryre', '2017-05-14 07:58:43');

-- Table structure for table `tblpages`

CREATE TABLE `tblpages` (

`id` int(11) NOT NULL,

`type` varchar(255) DEFAULT '',

`detail` longtext DEFAULT NULL

) ENGINE=MyISAM DEFAULT CHARSET=latin1;

-- Dumping data for table `tblpages`

INSERT INTO `tblpages` (`id`, `type`, `detail`) VALUES

-- Table structure for table `tbltourpackages`

CREATE TABLE `tbltourpackages` (

`PackageId` int(11) NOT NULL,

`PackageName` varchar(200) DEFAULT NULL,

`PackageType` varchar(150) DEFAULT NULL,

`PackageLocation` varchar(100) DEFAULT NULL,

`PackagePrice` int(11) DEFAULT NULL,

`PackageFetures` varchar(255) DEFAULT NULL,

`PackageDetails` mediumtext DEFAULT NULL,

`PackageImage` varchar(100) DEFAULT NULL,

`Creationdate` timestamp NULL DEFAULT current_timestamp(),

`UpdationDate` timestamp NULL DEFAULT NULL ON UPDATE current_timestamp()

ENGINE=InnoDB DEFAULT CHARSET=latin1;

-- Dumping data for table `tbltourpackages`

INSERT INTO `tbltourpackages` (`PackageId`, `PackageName`, `PackageType`, `PackageLocation`, `PackagePrice`, `PackageFetures`, `PackageDetails`, `PackageImage`, `Creationdate`, `UpdationDate`) VALUES

-- --

-- Table structure for table `tblusers`

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 NULL DEFAULT current_timestamp(),
  `UpdationDate` timestamp NULL DEFAULT NULL ON UPDATE current_timestamp()
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
-- Dumping data for table `tblusers`
INSERT INTO `tblusers` (`id`, `FullName`, `MobileNumber`, `EmailId`, `Password`, `RegDate`, `UpdationDate`) VALUES
(1, 'Anuj kumar', '1111111111', 'anuj@gmail.com', 'f925916e2754e5e03f75dd58a5733251', '2017-05-10 10:38:17', '2019-07-20 20:18:18'),
(3, 'sarita', '9999999999', 'sarita@gmail.com', '5c428d8875d2948607f3e3fe134d71b4', '2017-05-10 10:50:48', '2017-05-14 07:40:19'),
(7, 'test', '7676767676', 'test@gm.com', 'f925916e2754e5e03f75dd58a5733251', '2017-05-10 10:54:56', '0000-00-00 00:00:00'),
(8, 'Anuj kumar', '9999999999', 'demo@gmail.com', 'f925916e2754e5e03f75dd58a5733251', '2017-05-14 07:17:44', '0000-00-00 00:00:00'),
(9, 'XYZabc', '3333333333', 'xyz@gmail.com', 'f925916e2754e5e03f75dd58a5733251', '2017-05-14 07:25:13', '2017-05-14 07:25:42'),
(10, 'Anuj Kumar', '4543534534', 'demo@test.com', 'f925916e2754e5e03f75dd58a5733251', '2017-05-14 07:43:23', '2017-05-14 07:46:57'),
(11, 'XYZ', '8888888888', 'abc@g.com', 'f925916e2754e5e03f75dd58a5733251', '2017-05-14 07:54:32', '2017-05-14 07:55:17');
-- Indexes for dumped tables
-- Indexes for table `admin`
ALTER TABLE `admin`
  ADD PRIMARY KEY (`id`);
-- Indexes for table `tblbooking`
ALTER TABLE `tblbooking`
  ADD PRIMARY KEY (`BookingId`);
-- Indexes for table `tblenquiry`
ALTER TABLE `tblenquiry`
  ADD PRIMARY KEY (`id`);
-- Indexes for table `tblissues`
ALTER TABLE `tblissues`
  ADD PRIMARY KEY (`id`);
-- Indexes for table `tblpages`
ALTER TABLE `tblpages`
  ADD PRIMARY KEY (`id`);
-- Indexes for table `tbltourpackages`
```

```
ALTER TABLE `tbltourpackages`
  ADD PRIMARY KEY (`PackageId`);
-- Indexes for table `tblusers`
ALTER TABLE `tblusers`
  ADD PRIMARY KEY (`id`),
  ADD KEY `EmailId` (`EmailId`),
  ADD KEY `EmailId_2` (`EmailId`);
-- AUTO_INCREMENT for dumped tables
-- AUTO_INCREMENT for table `admin`
ALTER TABLE `admin`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=2;
-- AUTO_INCREMENT for table `tblbooking`
ALTER TABLE `tblbooking`
  MODIFY `BookingId` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=12;
-- AUTO_INCREMENT for table `tblenquiry`
ALTER TABLE `tblenquiry`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=5;
-- AUTO_INCREMENT for table `tblissues`
ALTER TABLE `tblissues`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=8;
-- AUTO_INCREMENT for table `tblpages`
ALTER TABLE `tblpages`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=22;
-- AUTO_INCREMENT for table `tbltourpackages`
ALTER TABLE `tbltourpackages`
  MODIFY `PackageId` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=7;
-- AUTO_INCREMENT for table `tblusers`
ALTER TABLE `tblusers`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=12;
COMMIT;
```

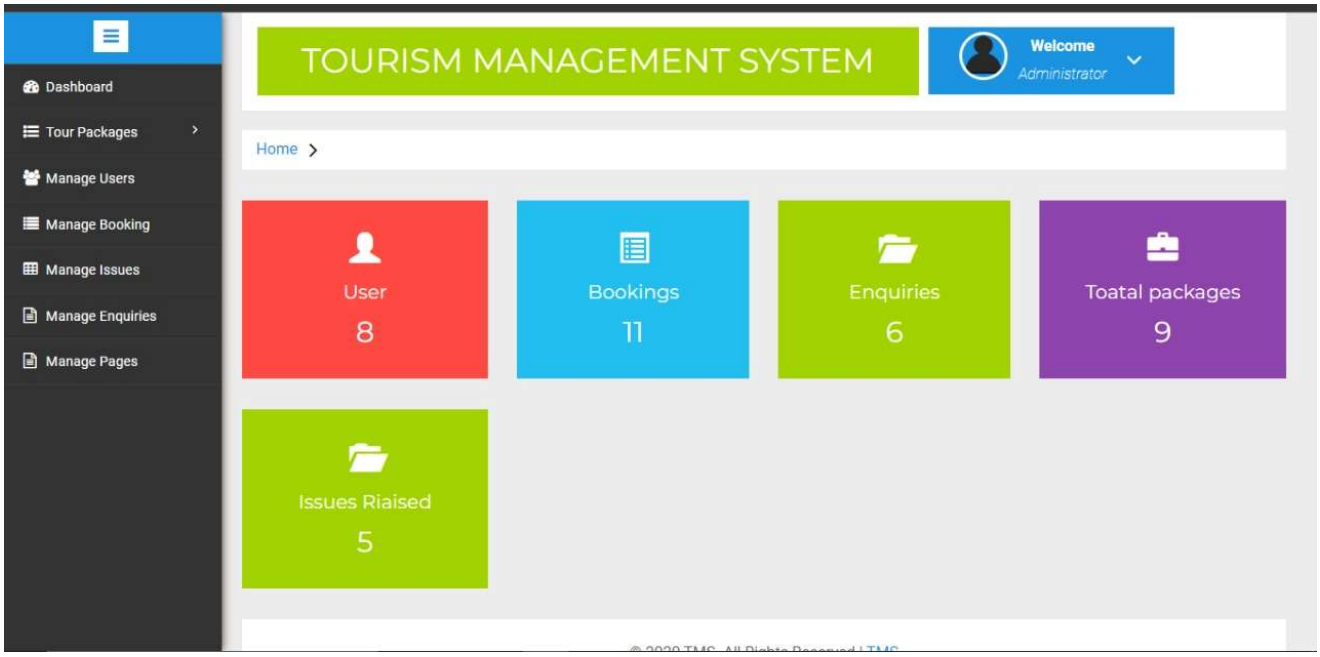
CHAPTER4

OUTPUT SCREENS

4.1 ADMIN OUTPUT SCREEN

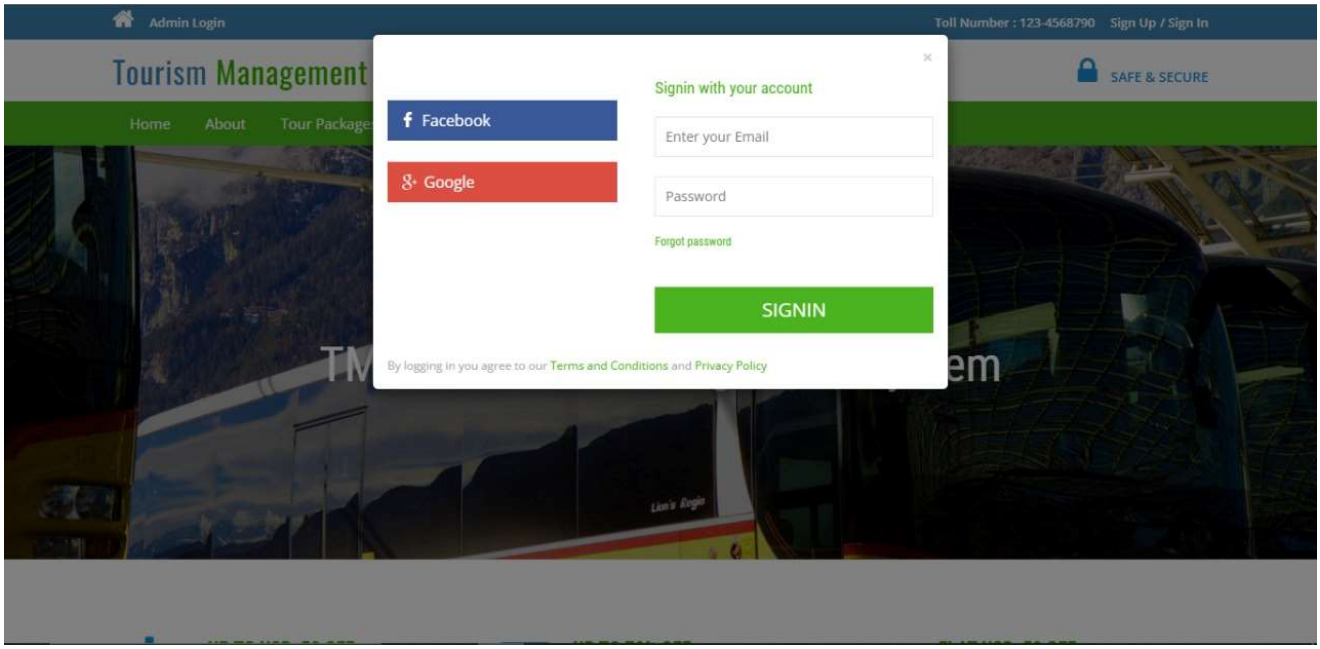
ADMIN LOGIN

ADMINISTRATOR PAGE



4.2 USER OUTPUT SCREEN

USER LOGIN



PACKAGE LISTS