TOURISM

MANAGEMENT SYSTEM

**A**

**PROJECT REPORT**

**ON**

**TOURISM MANAGEMENT SYSTEM**

**SUBMITTED BY**

**SACHIN CHAUDHARI**

**&**

**SRINIVAS KONKAL**

**(ROLL NO : 02 & 11)**

**DECEMBER-2020**

**FOR**

**THE EXAMINATION SYSTEM UNDER THE GUIDENCE OF**

**PROFESSORS: Mrs. SMITA DALVI, Mrs. VAISHALI KADAM &**

**Mrs. SUPRIYA SUDHIR**

**EXAMINATION (BSC.IT, SEMESTER V)**



**SHREE SHANKAR NARAYAN COLLEGE EDUCATION TRUSTS**

**SHANKAR NARAYAN COLLEGE**

**(DEPARTMENT OF BSC.IT)**

**ABSTRACT**

Online tours and travel is a web based project where a user may search and apply for a package. The system allows the user to check various tours packages and choose his package accordingly. The software system checks for user choice and then queries the database for various available packages to that destination. The system then loads all that data and puts those choices in front of the user. The user can now choose various packages as per his desire. When the user chooses the type of package option, the system also allows the user to book the package of the destination for the desire date. Thus this software system automates the working of a travel agency and allows users to check and book his holidays online through this website. This system also helps to keep the records of the tour agent. This system also provides the user to cancel the request.

**ACKNOWLEDGEMENT**

We take this opportunity to express our profound gratitude and indebtedness to our project guides Mrs. Smita Dalvi, Mrs. Vaishali Kadam, Mrs. Supriya Sudhir for giving us the opportunity to accomplish this project. We are very much thankful to Mrs. Smita Dalvi , Mrs.

Vaishali Kadam , Mrs. Supriya Sudhir for being very much resourceful, kind and helpful.

We are very much thankful to Mrs. Smita Dalvi, Mrs. Vaishali Kadam & Mrs. Supriya Sudhir for being very much resourceful, kind and helpful. Their positive attitude, unassailable optimism and unawaring faith in us assured that we come out of words whenever we encountered difficulties.

Finally we wish to thank all my friends and IT department to directly or indirectly help us in the completion of this project. Last but not the least we would thank our family without whose support, motivation and encouragement this would not have been possible.

**DECLARATION**

We here by declare that the project entitled, **“Tourism Management System”**, has not been in any case duplicated to submit to any other university for the award of any degree. To the best of my knowledge other than me, no one has submitted to any other university.

The project is done in partial fulfillment of the requirements for the award of degree of **BACHELOR OF SCIENCE (INFORMATION TECHNOLOGY)** to be submitted as final semester project as part of our curriculum.

**Sachin Chaudhari & Srinivas Konkal**

**Table of Contents**

# CHAPTER 1: INTRODUCTION……………………………………………………………3

1.1 Background……………………….…………………………………………………..3

1.2 Objectives…………………………………………………………………………….4

1.3 Purpose, Scope, and Applicability………………........................................................4

1.3.1 Purpose………………………………….….……………………………………5

1.3.2 Scope………………………..…………………………………………………...5

1.3.3 Applicability…………………………….……………………………………….6

1.4 Achievements………………………………….….……………………………….…..6

1.5 Organization of report………………………….……….………..................................6

CHAPTER 2: SURVEY OF TECHNOLOGIES………….…….…………………………...8

CHAPTER 3: REQUIREMENTS AND ANALYSIS………..……………………..………12

3.1 Problem Definition…………………………….……..…………………...…………12

3.2 Requirements Specification………………...…………………………..……………12

3.3 Planning and Scheduling………………….…………………………………………13

3.4 Software and Hardware Requirements…….….….……….…………………………14

3.5 Preliminary Product Description…………………….………………………………15 3.6 Conceptual Models…………………….………..……..……………………………16

CHAPTER 4: SYSTEM DESIGN…………..………………………………………………20 4.1 Module division………………………..……………………………………………20

4.2 Procedural design……………………………………………………………………21

4.3 Process flow chart……………………..……..…………………….……..…………24

4.4 User interface design………………. …..……………………………..……………25

4.5 Security Issues…………………………………………………………….…………26

4.6 Test Cases Design……………………..……. …………..…………………………27

CHAPTER 5: IMPLEMENTATION AND TESTING

5.1 Implementation Approaches……………………..……. ………………………..28

5.2 Coding Details and Code Efficiency……………………..……. ………………..29

5.2.1 Code Efficiency……………………..……. …………..…………………………31

5.3 Testing Approach……………………..……. …………..………………………32

5.3.1 Unit Testing……………………..……. …………..…………………………33

5.3.2 Integrated Testing……………………..……. …………..…………………34

5.3.3 Beta Testing……………………..……. …………..…………………………36

5.4 Modifications and Improvements……………………..……. …………………..37

5.5 Test Cases……………………..……. …………..…………………………39

CHAPTER 6: RESULTS AND DISCUSSION

6.1 Test Reports……………………..……. …………..…………………………40

6.2 User Documentation……………………..……. …………..…………………41

CHAPTER 7: CONCLUSIONS

7.1 Conclusion……………………..……. …………..…………………………42

7.1.1 Significance of the System……………………..……. ………………….43

7.2 Limitations of the System……………………..……. ……………………….45

7.3 Future Scope of the Project……………………..……. …………..…………..46

REFERENCES

GLOSSARY

**Table of figures**

Fig.1: Gantt chart…………………. …………………..……………………………………14

Fig.2: ER diagram……………………. …………………….………………………………16

Fig.3: Usecase diagram…………………… ……………………...…………………………17

Fig.4: DFD diagram for admin……………. …………………………………….…………18

Fig.5: DFD diagram for user…………………... ……………………………………………19

Fig.6: Process flow chart…………………………….………………………………………24

Fig.7: UI of registration page………….……………………………..………………………25

Fig.8: UI of login page…………..……….………..…………………………………………26

**CHAPTER 1: INTRODUCTION**

**1.1 Background:**

“The World is a Book and Those Who Do Not Travel Read Only a Page.”

- Saint Augustine.

Tourism has turned out to be an economic booster contributing to the economic development of many countries over the last few decades. People see holidays as a necessity, and not as luxury in the present scenario. Tourism calls for coordination and cooperation between travel agents, tour operators, and tourists. Tourism has a few major elements — destinations, attractions, sites, accommodation, and all ancillary services.

Tourism is among the economic and social sectors that are registering rapid growth in the world, and nowadays it has been found to be making contribution in supporting and accelerating national development. Tourism makes tremendous contribution serving as a source of foreign exchange promoting micro and small-scale enterprise, creating employment opportunities and ensuring sustainable development.

During recent years tourism has become one of the world's largest and most powerful industries. With the expansion in tourism, the application of information technology in that area has also developed rapidly. In today's fast-developing environment traditional management cannot meet the demand of updating speed, and it has significant limitations in saving and data modification and also data would easily be missing. All the shortcomings can lead to inaccurate information, as a basis for managers and decision makers. Tourism management system would be developed to meet these growing needs.

Experts point out that a travel decision-making of tourist, in fact, is a process including tourist information input, processing, and feedback. Therefore, this put forward a great demand on large amount of high-quality easy-to-get tourism information. Web-based tourism management system can well meet the demand, in which database is used to store large amount of tourism information, and web pages are used to present the information.

Tourism management system is a web based system which transforms the manual based tour and travel agency service to a computerized wet system by creating an easy way for the customer to plan their trip by including tourism site information and hotel information.

**1.2 Objectives:**

* To offer variety of travel services that are sure to match all the priorities of user.
* To provide users a web application that automates the process of tourist package booking.
* To globalize, organize, standardize and goal of journey towards perfectionism.
* To save the time of users and manual paper work required for booking a package.

**1.3 PURPOSE, SCOPE AND APPLICABILITY:**

**1.3.1 PURPOSE:**

* This web application provides the facility to book any tour package online.
* Users can choose from a variety of tourist destinations.
* Users can cancel the booking anytime and the refund will be given according to the terms & conditions of the web application.
* The administrator will have the right to manage user details, package details, bookings and various other things like- enquiries, issues, etc. He/She can add/delete/view any of these details at any time.

### 1.3.2 SCOPE:

* The website will provide stress-free, joyful and refreshing holidays with cost competitive and customized packages according to their requirements.
* As it is a web application, it can be used from any part of the world.
* It offer tour and travel services including ticket bookings and hotel reservations only in one click.
* It provides the most suitably designed as well as the customized travel packages to the customers.
* A traveler can find everything related to travelling services under one roof by this website.

**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.

**PROPOSED SYSTEM:**

The proposed system is a web based application and maintains a centralized repository of all related information. The system allows one to easily access the relevant information and make necessary travel arrangements. Users can decide about places they want to visit and make bookings online for travel and accommodation.

### 1.3.3 APPLICABILITY:

* This system can be used by tourism agencies:

The tourism agency which do work manually can use this system for bookings.

* It can be used by an individual to book a package:

An individual person can book any tourist package to any destination available on the website without going to any tourism agency office.

**1.4 ACHIEVEMENTS:**

The achievements made in coming up with Tourism Management System' were:

* Create a user-friendly tourism management website with which user will be able plan their trips from home.
* Learned how to face tense situations and meet the deadlines.
* Learned how to work in team.
* Gain necessary skills and experience by creating the website.
* Gaining practical knowledge of various programming languages by implementing them.
* Get to know about various databases for storing information from website.
* Get to know about front-end and back-end development of a website.

**1.5 ORGANIZATION OF REPORT:**

This research work is organized into five chapters.

Chapter one is concerned with the introduction of the research study and it presents the preliminaries, theoretical background, statement of the problem, aim or objectives of the study, significance of the study, scope of the study, project constraint and definition of terms.

Chapter two focuses on the review of related literature, the contributions of other scholars on the subject matter is discussed.

Chapter three is concerned with the system investigation and analysis. It presents the research methodology used in the development of the system; it analyzes the present system to identify the problems and provides information on the advantages and disadvantages of the proposed system. The system design is also presented in this chapter.

Chapter four presents the system implementation and documentation, the choice of programming language, analysis of modules, choice of programming language and system requirements for implementation.

# CHAPTER 2: SURVEY OF TECHNOLOGIES

**Php:**

PHP (or PHP Hypertext Preprocessor) is a server-side scripting language that is used to create dynamic web pages that can interact with databases. It is a widely-used open source language that is specifically used for web application development and can be embedded within HTML.

Why to use PHP?

1. It’s an open source language:

One of the major advantages of the PHP programming language is that it is accessible for free to web developers. It is executed at the server side that means it functions on the web server. Because of the open-source feature, PHP developer can learn about the scripting code in easily through online platforms.

1. Upgradation of php software is easier a lot:

Because of the effortless decoding of syntax, it is easy to alter the PHP codes. In other words, it is true that **PHP web development** tasks can be executed in very less time. This characteristic of PHP programming language helps the web developers to transform their websites into advanced apps. An open-source feature of this programming language helps the developers to incorporate new-fangled functions on a constant basis with minimal costing involved. The icing on the cake is that further maintenance in PHP codes can be carried out even by other developers, not necessary by the team which has completed the web development task.

1. Php web development is cost-efficient:

Another reason why the PHP programming language happens to be the first-preference of web developers is that it is available for free. There are no such licensing fees involved for reaping the benefits of a programming language by web developers. The original source code of PHP programming language is available for free and circulated under General Public License. Additionally, the international community eventually helps in making advancements in its functioning. There are numerous PHP-based web solutions, of which, the practical usage is thoroughly tested and this is the reason why there is not much requirement of time and effort for improving the language functionality.

**MySQL:**

Following are reasons for using MySQL over other databases:

1. Data Security:

MySQL is globally renowned for being the most secure and reliable database management system used in popular web applications like WordPress, Drupal, Joomla, Facebook and Twitter. The data security and support for transactional processing that accompany the recent version of MySQL, can greatly benefit any business especially if it is an eCommerce business that involves frequent money transfers.

1. On-Demand Scalability:

MySQL offers unmatched scalability to facilitate the management of deeply embedded apps using a smaller footprint even in massive warehouses that stack terabytes of data. On-demand flexibility is the star feature of MySQL. This open source solution allows complete customization to eCommerce businesses with unique database server requirements.

1. High Performance:

MySQL features a distinct storage-engine framework that facilitates system administrators to configure the MySQL database server for a flawless performance. Whether it is an eCommerce website that receives a million queries every single day or a high-speed transactional processing system, MySQL is designed to meet even the most demanding applications while ensuring optimum speed, full-text indexes and unique memory caches for enhanced performance.

1. Round-the-clock Uptime:

MySQL comes with the assurance of 24X7 uptime and offers a wide range of high availability solutions like specialized cluster servers and master/slave replication configurations.

1. Comprehensive Transactional Support:

MySQL tops the list of robust transactional database engines available on the market. With features like complete atomic, consistent, isolated, durable transaction support, multi-version transaction support, and unrestricted row-level locking, it is the go-to solution for full data integrity. It guarantees instant deadlock identification through server-enforced referential integrity.

**HTML:**

HTML or Hypertext Markup Language is a typical choice used for the development of websites, web pages and web based applications. Business stakeholders, project management and program developers prefer HTML over other alternative program development options due to the advantageous characteristics of HTML. A few of the notable advantages of HTML are ‘it

is light weight in structure‘, ‘it is easy to learn and use’, ‘it is an open source program that can be used for free of cost’, ‘it is supported in all kinds of browsers’, ‘effortless to create and edit’, ‘easy to integrate with other programming languages’, ‘allows to accommodate changes at any time as required for the requirements’, etc.

Advantages of using HTML:

* HTML is easy to learn and use.
* HTML is simple to edit.
* It is free and supported by all browsers.
* It can integrate with other programming languages easily.

**CSS:**

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

* Easier to maintain and update.
* Greater consistency in design.
* More formatting options.
* Lightweight code.
* Faster download times.
* Search engine optimization benefits.
* Ease of presenting different styles to different viewers.
* Greater accessibility.

**Apache Tomcat:**

Apache Tomcat is a web server that is an open source software implementation of the Java

Servlet, Java Server Pages, Java Expression Language and Java Web Socket technologies. The Java Servlet, Java Server Pages, Java Expression Language and Java Web Socket specifications are developed under the Java Community Process. Apache Tomcat is developed in an open and participatory environment and released under the Apache.

# CHAPTER 3: REQUIREMENTS AND ANALYSIS

## 3.1 Problem Definition:

* The problem is here to develop a web application which completely automates the process of package booking.
* To perform, the complete problem is divided into following sub-problems so that they can be solved easily and after that can be integrated to make it an integrated working application.
  + 1. Administrator module
    2. User module
    3. Package module
    4. Enquiry module
    5. Booking module

**3.2 Requirements Specification:**

**Functional requirements:**

The Functional Requirements of the project are as follows:

* To provide the functionality to user to book tour packages online.
* To provide login interface through which only registered users can pass by.
* To provide the functionality of booking cancellation any time.
* To functionally enable admin to manage tourist packages.
* To functionally enable admin to manage user details.
* To functionally enable admin to manage booking details.

**Non-functional requirements:**

The Non-Functional Requirements of the project are as follows:

* The system should be user-friendly.
* The system should be accurate.
* The system should be better than the existing system.
* The system should be completely consistent and secure.

**3.3 Planning and Scheduling:**

The major issue the project plan addresses are:

* Cost estimation
* Schedule and milestone
* Personal plan
* Software quality assurance
* Configuration management plan
* Project monitoring plans
* Risk management

➢ **Gantt chart:**

**Gantt chart data:**

|  |  |  |
| --- | --- | --- |
| Task name | Start date | Duration (in days) |
| Introduction | 10-09-2020 | 10 |
| Survey of technologies | 21-09-2020 | 18 |
| Requirements analysis | 10-10-2020 | 20 |
| System design | 31-10-2020 | 20 |
| Implementation and testing | 1-11-2020 | 30 |
| Results and discussions | 1-12-2020 | 40 |
| Conclusion | 9-1-2021 | 20 |

**Gantt chart:**

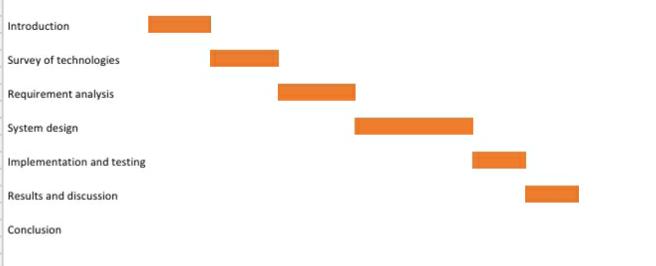


Fig.1: Gantt chart

**3.4 Software and hardware requirements:**

**Hardware requirements:**

* Pentium IV processor
* 20GB hard disk
* 1GB RAM
* 100-512 MBPS of network car

**Software requirements:**

* Operating system: windows 10
* Technologies: HTML, CSS, JavaScript
* Language: PHP
* Back-end: MySQL
* Xammp

**3.5 Preliminary product description:**

* The system will facilitate online booking of tour package.
* All one needs to do is change the database accordingly.
* This application has login module, we need username and password facility and credentials should be checked properly at the time of login for users and admin.
* The user can book as well as cancel packages.
* The user can view his tour history also.

**3.6 Conceptual models:**

**ER Diagram:**

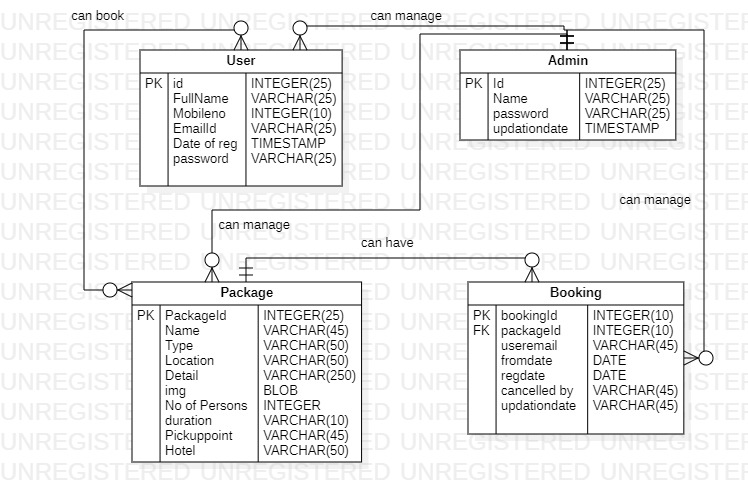


Fig.2: ER diagram

**Usecase diagram:**

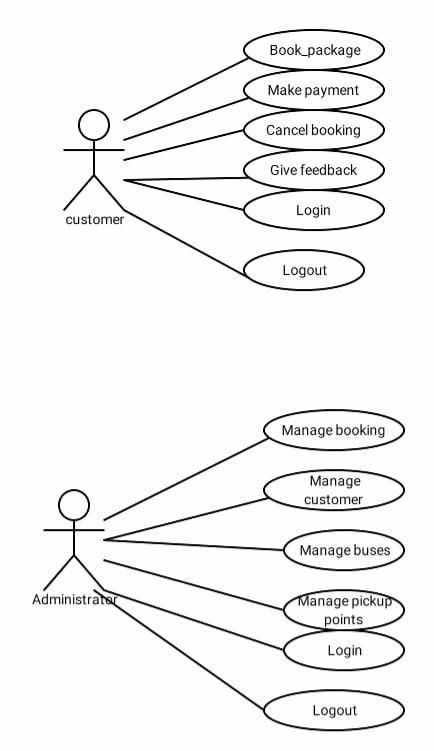


Fig.3: Usecase diagram

**Data flow diagram for admin:**

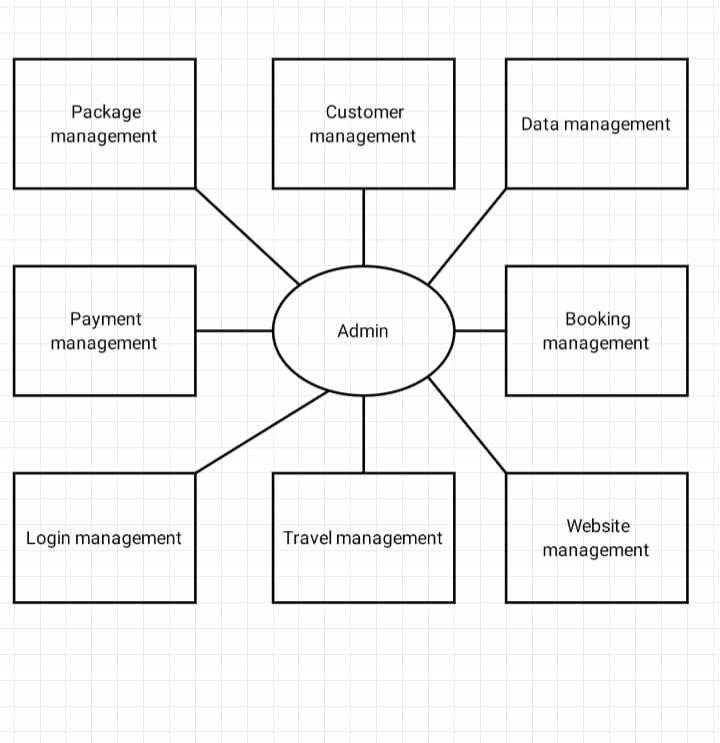


Fig.4: DFD for admin

**Data flow diagram for user:**

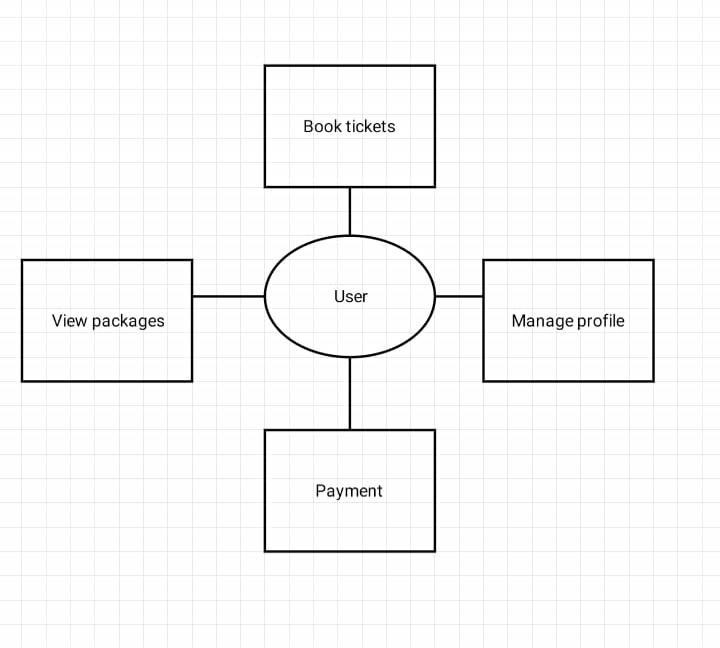


Fig.5: DFD for user

**Sequence Diagram:**

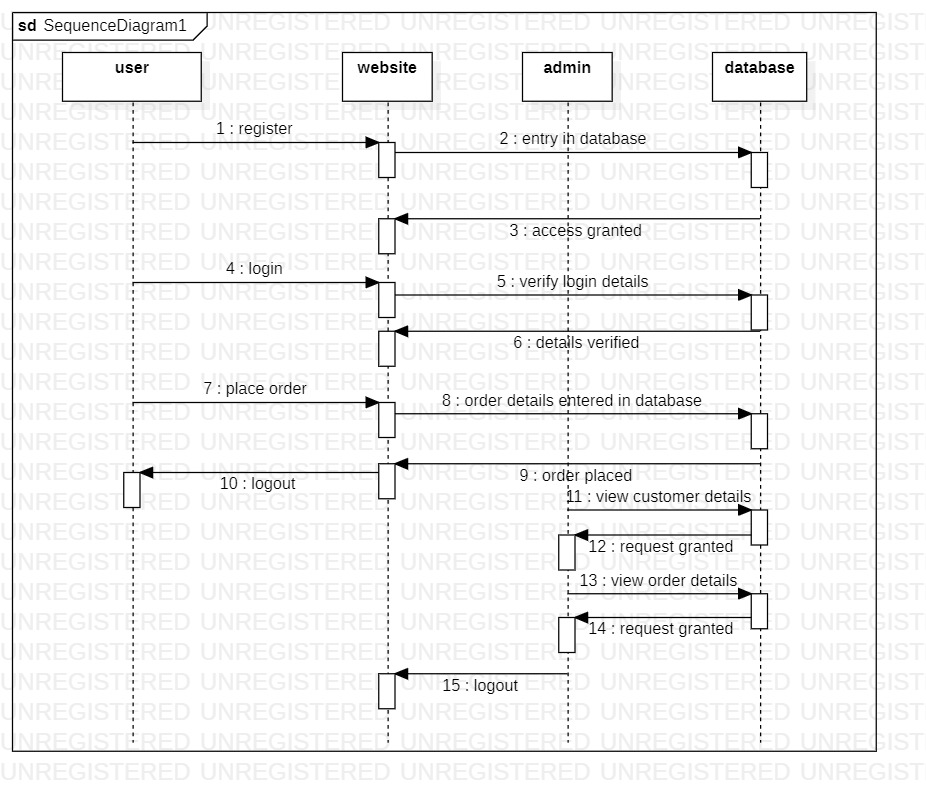


Fig 6 : Sequence Diagram

**CHAPTER 4: SYSTEM DESIGN**

**4.1 Module division:**

Our project has following modules:

* Administrator module
* User module
* Package module
* Payment module
* Transportation module
* Routes module
* Hotel module
* Feedback module

**Administrator module:**

The administrator can add, delete and view the data related to packages, transportation, routes, bookings, hotels etc.

**User module:**

This module will contain data related to user. The user can update personal information, book any package and cancel booking.

**Package module:**

This module will give information related to the tourist packages. It will contain description of the tourist place, photo gallery of tourist place, package price and discount on package (if available).

**Booking module:**

This module will have information about Package ID, Booking ID, Full name of User and price of the package.

**Enquiry module:**

This module will include the details of the enquiries submitted by the users about the services provided by our website.

**4.2 Procedural design:**

**4.2.1 Data structure:**

**Table name: login:**

|  |  |  |
| --- | --- | --- |
| **Field name** | **Data type** | **Key** |
| USERID | NUMBER(25) | Primary key |
| USERNAME | VARCHAR 2(20) | - |
| PASSWORD | VARCHAR 2(20) | - |

**Table name: User Registration**

|  |  |  |
| --- | --- | --- |
| **Field name** | **Data type** | **Key** |
| USERNAME | VARCHAR 2(20) | Primary key |
| NAME | VARCHAR 2(25) | - |
| AGE | NUMBER(5) | - |
| ADDRESS | VARCHAR 2(50) | - |
| PINCODE | NUMBER(6) | - |
| MOBNO | NUMBER(10) | - |
| EMAILID | VARCHAR 2(50) | - |

**4.2.2 Algorithms Design:**

**Login:-**

1. Start.
2. Sanitize inputs.
3. Check Username if exists.
4. Get username’s password if exists.
5. If using encryption, encrypt the entered password.
6. Compare the username’s password to the entered password.
7. Do some stuff.
8. End.

**Remember me:-**

1. User is logged on with Remember Me.
2. Login Cookie issued with token & Signature.
3. When is returning, Signature is checked.
4. If Signature is ok, then username & token is looked up in the database.
5. If not valid, return to login page.
6. If valid automatically login.

## 4.3 Process flow chart:

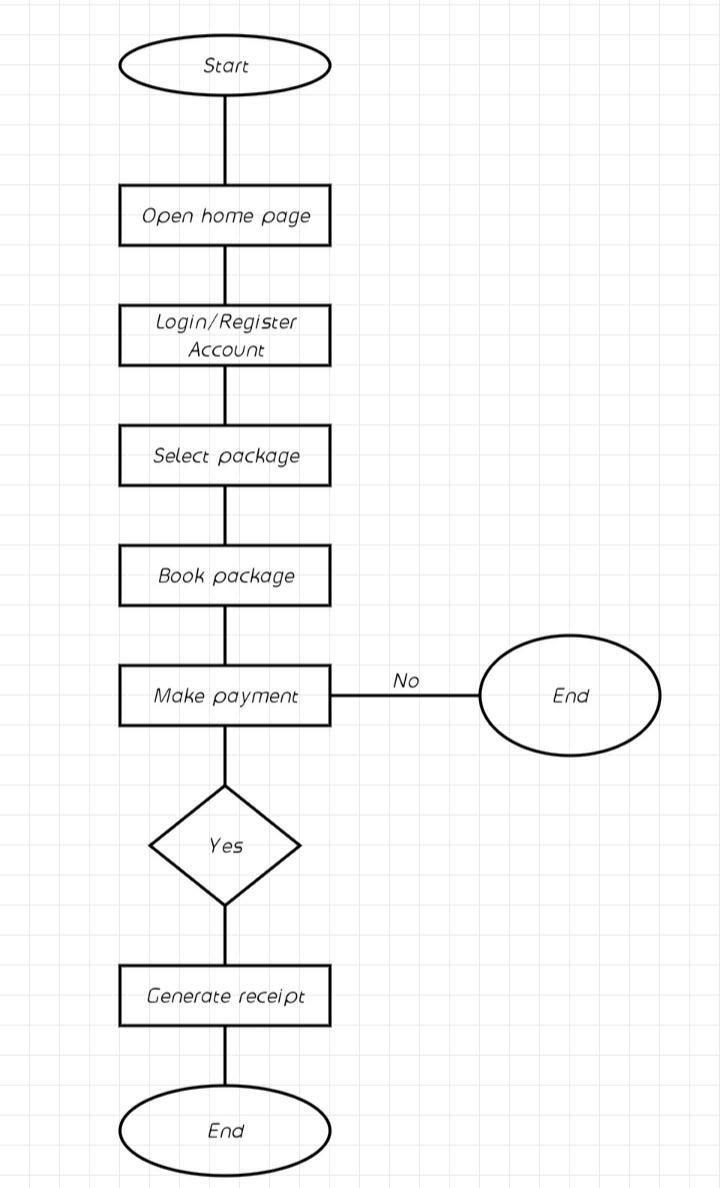


Fig.6: Process flow chart

## 4.4 User Interface design:

**Registration page:**

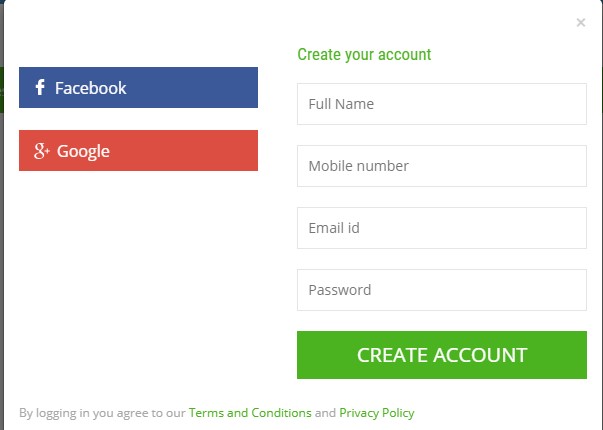


Fig.7: UI of registration page

**Login page:**

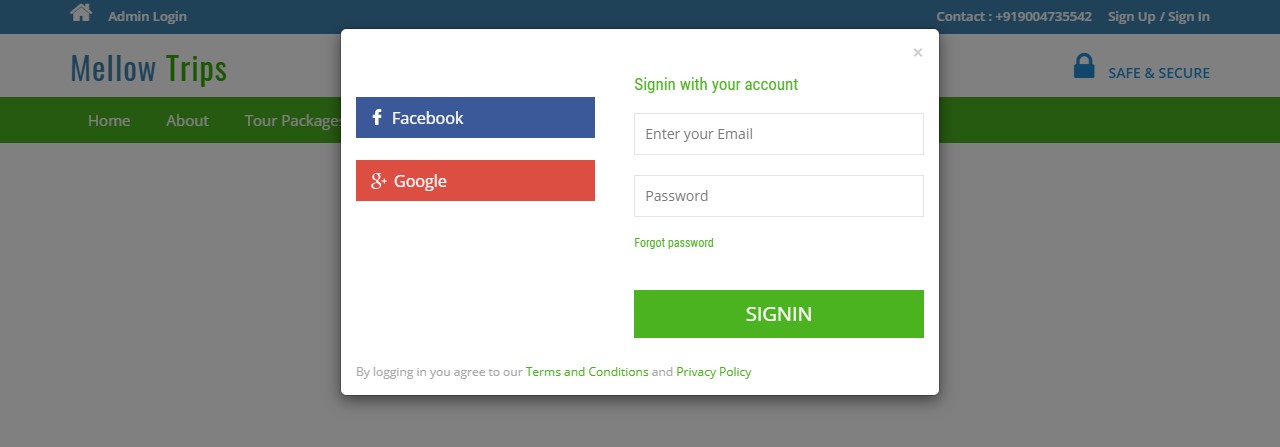


Fig.8: UI of login page

**4.5 Security Issues**

Security is an important issue for modern IT system. System administrator and security administrator are responsible for safeguard’s organization data and computing infrastructure.

We are using the ‘Authorization’ and ‘Authentication’ techniques to secure our “TOURISM MANAGEMENT SYSTEM” website.

**4.6 Test case design:**

|  |  |  |
| --- | --- | --- |
| Sr. No. | Test case description | Expected result |
| 1. | The Controller leaves the  Admin Name field blank. | Error message is displayed to the Controller and focus is set on the text box. |
| 2. | The Controller enters incorrect different password | Error message “Password doesn’t match!!” |

**Admin:**

**Registration**:

|  |  |  |
| --- | --- | --- |
| Sr. No. | Test case description | Expected result |
| 1. | All mandatory field must be filled, if it is blank then Error will generate. | Submission will not be done. Error will generate “Must Fill The Field”. |
| 2. | Confirm-password is different from password. | Error message “Wrong password”. |

**User**:

|  |  |  |
| --- | --- | --- |
| Sr. No. | Test case description | Expected result |
| 1. | The user leaves the user name or password or both blank before clicking OK. | Error message is displayed to the user and focus is set on the textbox. |
| 2. | The user enters incorrect username. | Error message “Invalid Username”. |
| 3. | The user enters incorrect password. | Error message “Invalid Password”. |
| 4. | The user clicks forget password. | Message “Password sent to registered e-mail id” is displayed. |

# Chapter 5: Implementation and Testing

## 5.1 Implementation Approaches

Implementation refers to the second last phase of the Software Development Lifecycle. There are many different approaches which can be used for development of software such as waterfall model, spiral model, big bang approach etc. Each of these approaches have their own advantages and disadvantages.

For developing this project we have used the waterfall model approach. We started with requirement gathering and analysis in which we analyzed the type of the project and decided on the requirements such as hardware requirements, software requirements, programming languages to be used etc. Then we moved on to design phase in which we designed the basic structure of the website such as user interface, different modules involved etc. After designing the basic skeleton of the project, we moved on to coding in which individual module were coded and implemented using the programming languages decided. After implementing all the modules, we tested the modules individually through unit testing and then tested the whole project through integration and system testing. Finally, the project was implemented and any errors solved through maintenance.

## 5.2 Coding details and code efficiency

Config.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());

}

?>

Package List.php

<?php

session\_start();

error\_reporting(0);

include('includes/config.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" />

<script type="applijewelleryion/x-javascript"> addEventListener("load", function() { setTimeout(hideURLbar, 0); }, false); function hideURLbar(){ window.scrollTo(0,1); } </script>

<link href="css/bootstrap.css" rel='stylesheet' type='text/css' />

<link href="css/style.css" rel='stylesheet' type='text/css' />

<link href='//fonts.googleapis.com/css?family=Open+Sans:400,700,600' rel='stylesheet' type='text/css'>

<link href='//fonts.googleapis.com/css?family=Roboto+Condensed:400,700,300' rel='stylesheet' type='text/css'>

<link href='//fonts.googleapis.com/css?family=Oswald' rel='stylesheet' type='text/css'>

<link href="css/font-awesome.css" rel="stylesheet">

<!-- Custom Theme files -->

<script src="js/jquery-1.12.0.min.js"></script>

<script src="js/bootstrap.min.js"></script>

<!--animate-->

<link href="css/animate.css" rel="stylesheet" type="text/css" media="all">

<script src="js/wow.min.js"></script>

    <script>

         new WOW().init();

    </script>

<!--//end-animate-->

</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;"> 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">

                    <img src="admin/pacakgeimages/<?php echo htmlentities($result->PackageImage);?>" class="img-responsive" alt="">

                </div>

                <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>INR <?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>

    </div>

</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>

Signin.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" aria-labelledby="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" aria-label="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>

                        </div>

                    </div>

                </div>

            </div>

## 5.3 Testing Approach

### 5.3.1 Unit Testing

In this all the modules are tested individually to see if they function correctly or not. The interface is tested to see if data is handled correctly by the module. For example, to check whether the data entered by the user is processes correctly or not. Testing of flow of data through a module is important and the most basic level of testing. If the tests fail at this level the entire project can be considered obsolete.

### 5.3.2 Integration Testing

After successful completion of unit tests the project was tested as a whole with all its modules in integration testing. Here integration with database was done to check whether both frontend and backend were working properly together. For example, to check whether data entered by the user is stored as it is in database without any modifications, check whether data can be retrieved from the database successfully, see if error messages were showing if incorrect data is entered by the user etc.

### 5.3.3 Modifications and Improvements

1) If… else block used to handle error statements and also to display messages.

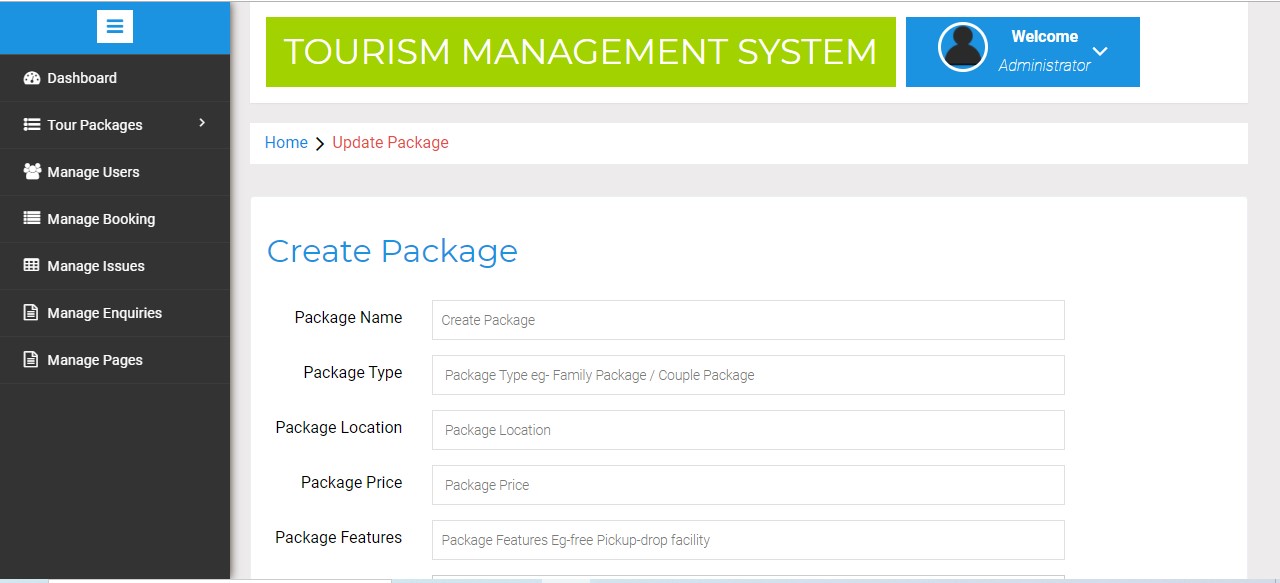
2) Biggest error was in configuring the XamppServer that we were using. We couldn’t access the database due to this but after configuring it is running perfectly. This was a major improvement for our project.

3) Minor fixes in database schema due to improper data handling issues.

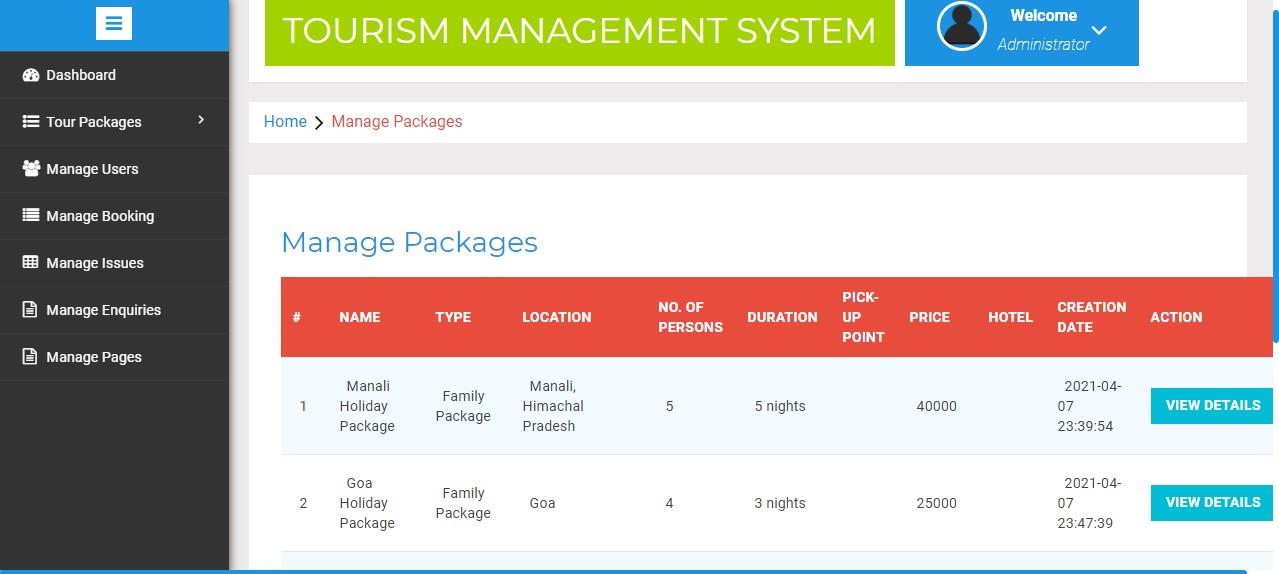
# Chapter 6: Results and Discussion

**ADMIN INTERFACE:**

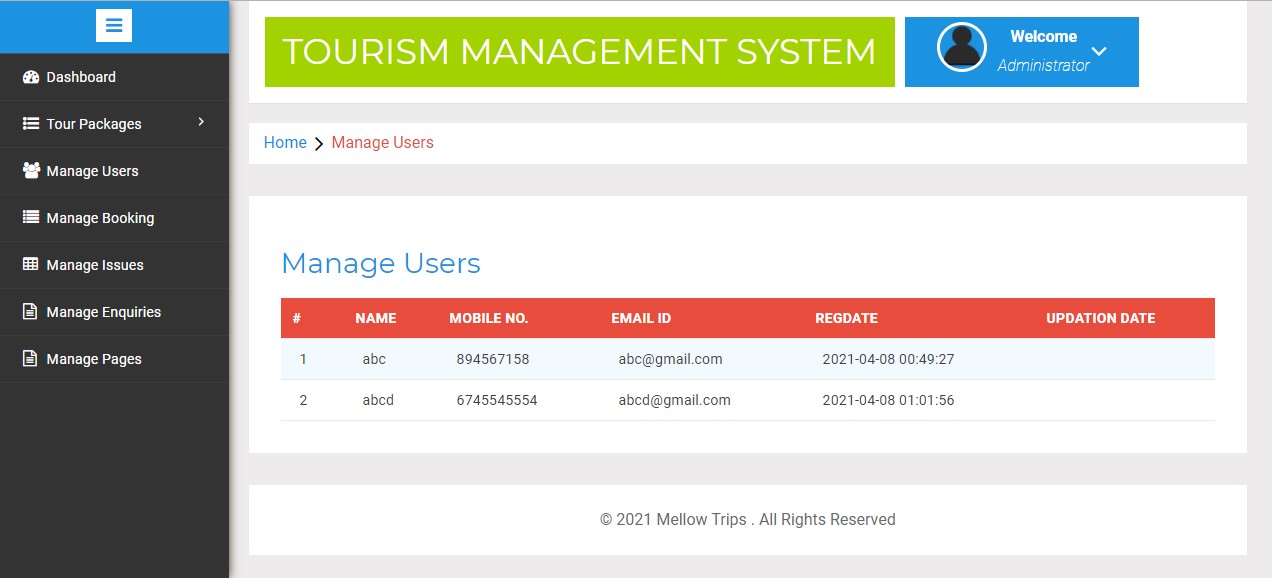
**Create Package:**



**Manage Package:**

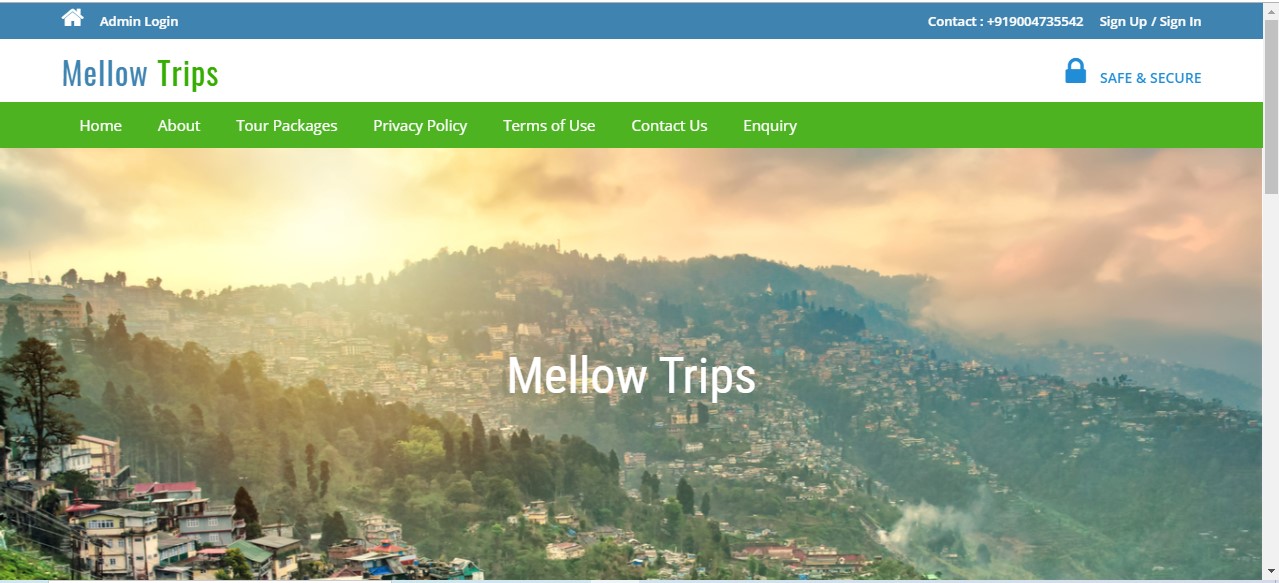


**Manage Users:**

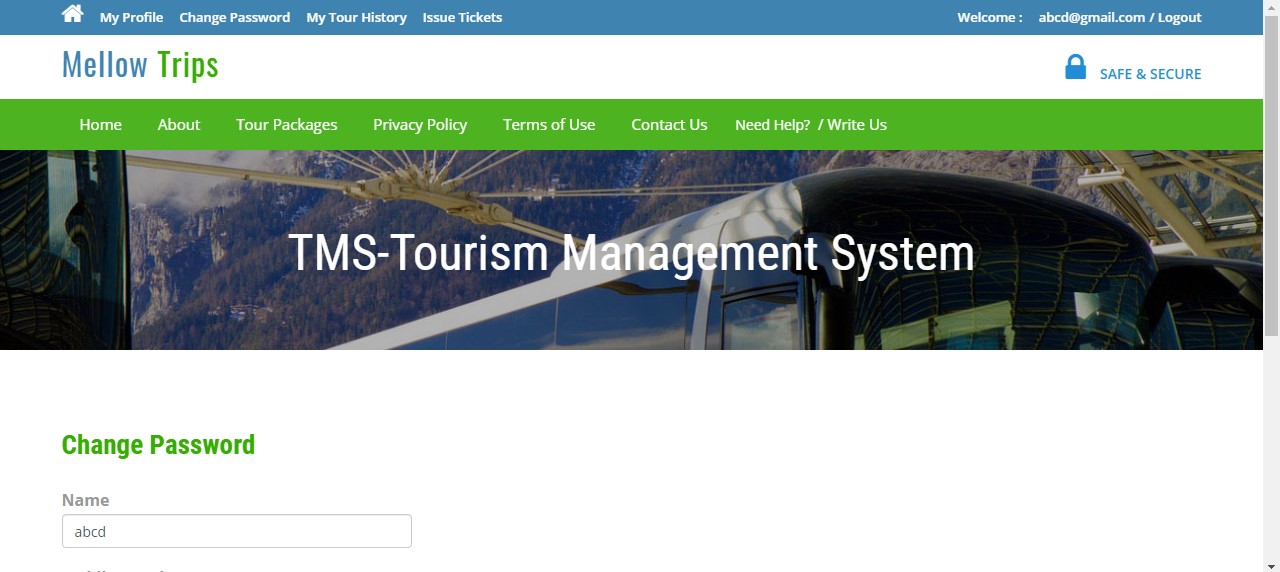


**USER INTERFACE:**

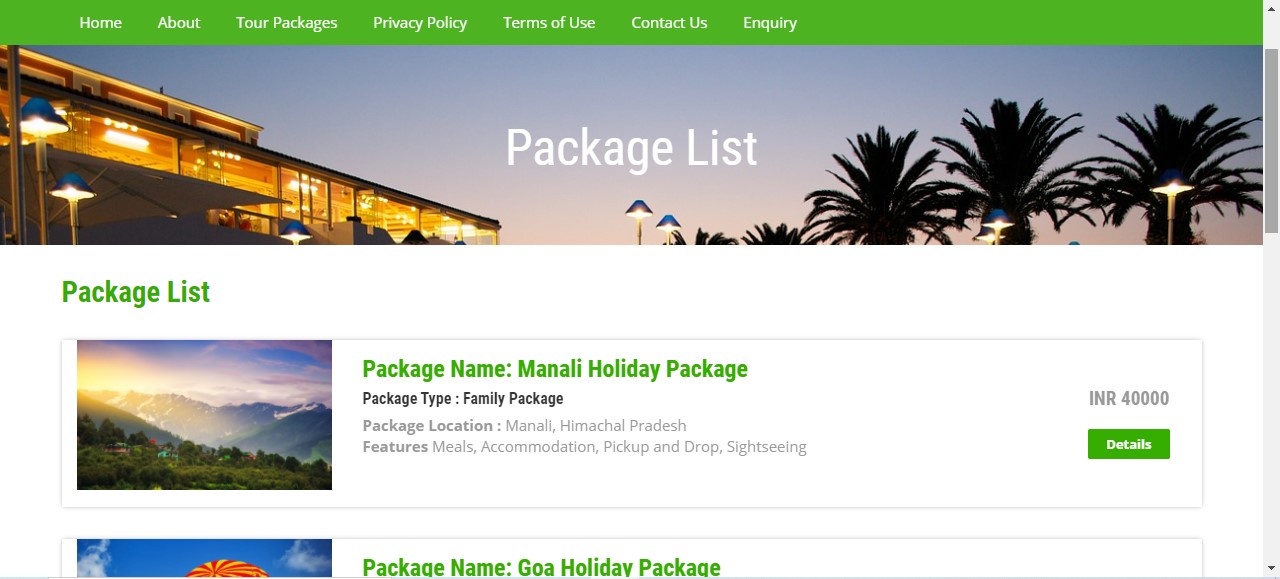
**Home page:**

****

**Change password page:**

****

**Package List Page:**

****

# Chapter 7: Conclusions

## 7.1 Conclusion

Website for Tourism Management is a user-friendly website which allows users to book tickets, accommodation,Tourist Packages or Destinations they from the comfort of their home. The system was tested and was found to function properly. But it can still be improved for higher use.

The website for Tourism Management is developed using PHP, HTML and CSS. MySQL server was used as the database and xampServer for hosting the website locally.

The following final conclusions were derived from the project:

* Automation of the entire system improves the efficiency.
* It provides a friendly graphical user interface which improves to be better when

compared to the existing system.

* Appropriate access given to users depending on permissions such as user and admin.
* Application can be modified and improved for further use.

## 7.2 Limitations of the system

* User verification is done only through emailid and password registered.
* Payment gateway is not implemented.
* Limited number of packages.
* No discounts on packages.

## 7.3 Future scope of project

Greater number of Packages can be added to the site. Additional features such as search option can be implemented. Discount and gift voucher schemes can be used. Chatbox can be used for automated handling of customer’s queries.

We can also add the feature of hotel selection and cab selection in addition to the package selection.