Software Design Document

For Minor Project

Travelogue

****

|  |  |
| --- | --- |
| Prepared by:  Aparna Tyagi (PCE15IT007)  Pooja Sharma(PCE15IT031) | Guide:  Ms. Sita Gupta  Asst. Professor, PCE |
| Department of Information Technology,  Poornima College of Engineering | |
| October, Session – 2018-19 | |

**Index**

|  |  |
| --- | --- |
| Topic | Page No. |
| 1. List of figures | 3 |
| 1. Introduction | 4 |
| 1. Use Case Diagram | 5 |
| 1. Architectural Design | 6 |
| 1. Flow Chart | 7 |
| 1. UML Diagram    1. Class Diagram    2. Deployment Diagram | 8  9 |
| 1. Database Diagrams | 10 |
| 1. GUI Design | 11-13 |
| 1. API’s | 14 |
| 1. References | 15 |
| 1. Appendices | 16 |
| 1. Guide’s Comments | 17 |

1. **List of Figures**

|  |  |  |
| --- | --- | --- |
| Fig Number | Figure Name | Page |
|  | USE CASE Diagram | 5 |
|  | Architectural Diagram | 6 |
|  | Flow Chart | 7 |
|  | Class Diagram | 8 |
|  | Deployment Diagram | 9 |
|  | Database Diagram | 10 |

1. **Introduction**

A lot of effort is required when one needs to decide or plan their holiday destination. But travelogue makes that work a lot easier and simpler. The user of the application just has to specify the type of destination he or she is looking for along with the number of people who will be travelling and the suitable destinations will be displayed on the screen.

Travelogue Trip Planner is a web-based project that makes it easy for users to plan their holiday in a matter of minutes. By pre-planning their dream holiday, the user can then proceed to book with ease. Travelogue is an online travel management system. It is aimed to offer a range of best-value services to ensure that tour runs smoothly and efficiently.

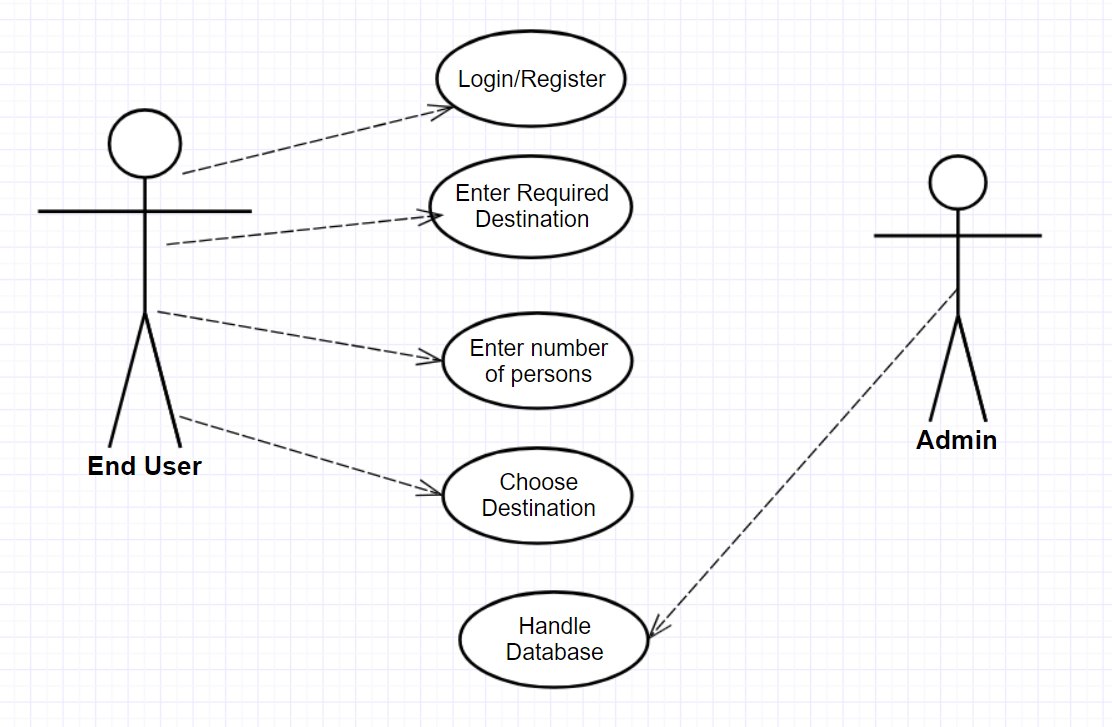
A user can login or create his or her own account and can navigate between multiple destinations depending upon what he or she likes best.

Now that the world is becoming a global village, the need to travel and explore has increased by a considerable amount. And this is where our application “Travelogue” comes into use. Its user-friendly and very easy to use. Just register, login and you are ready to go!

The application was developed keeping in mind the need of the travelers. Travelogue not only makes their journey comfortable but it also helps them to choose a location that is best suited according to their needs.

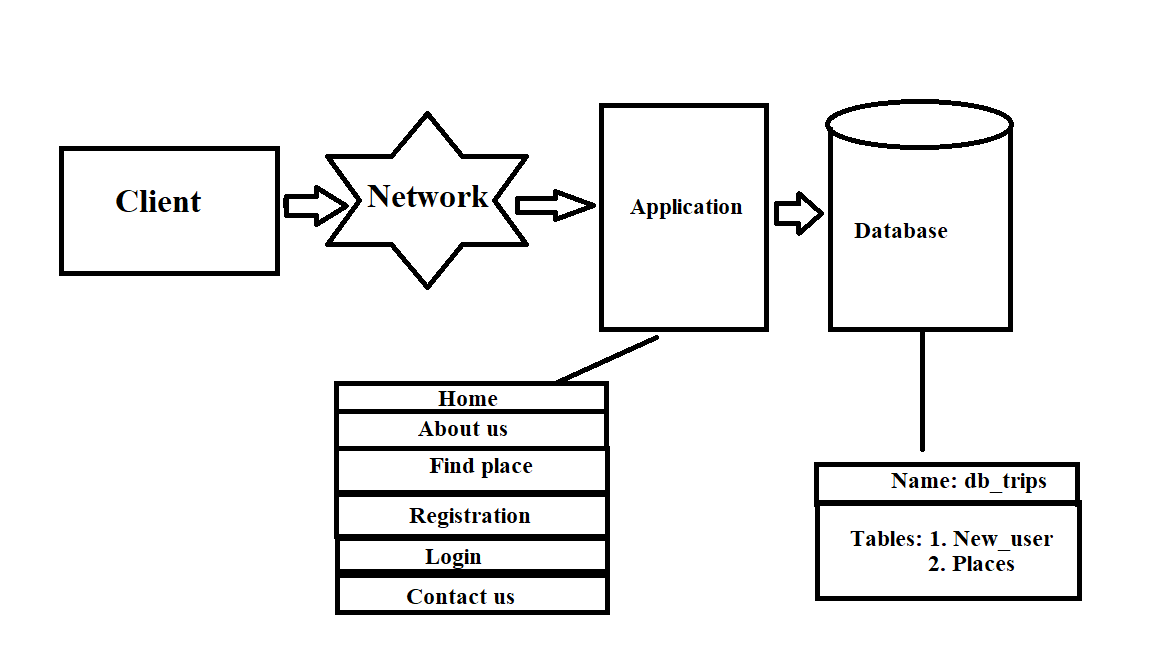
In case of any confusions or second thoughts, the end user can always confirm their destination using the application.

1. **Use Case Diagram**

****

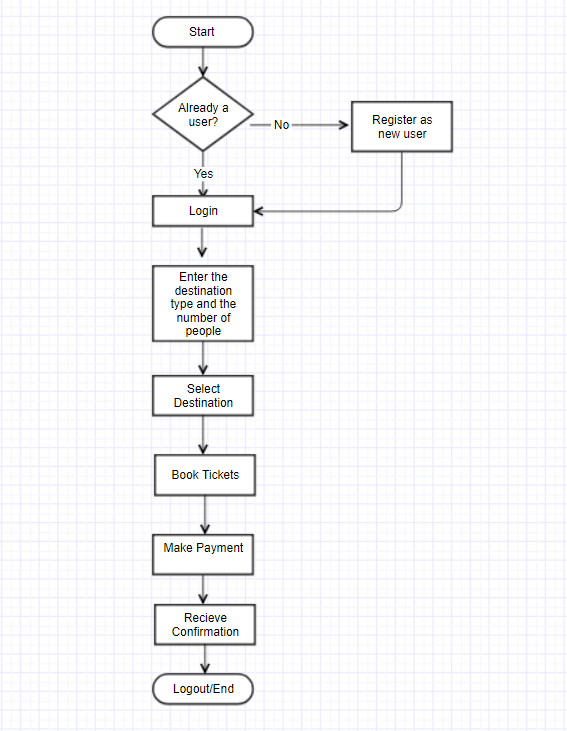
**FIGURE NO. 1**

1. **Architectural Diagram**

****

**Figure No. 2**

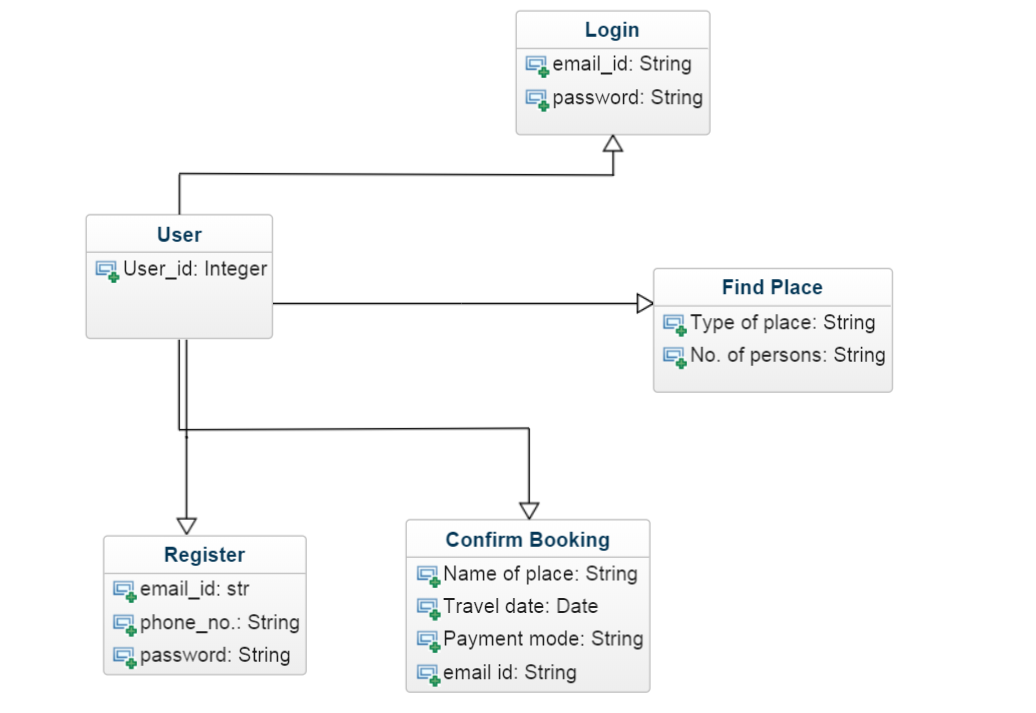
1. **Flow Chart**

****

**FIGURE NO. 3**

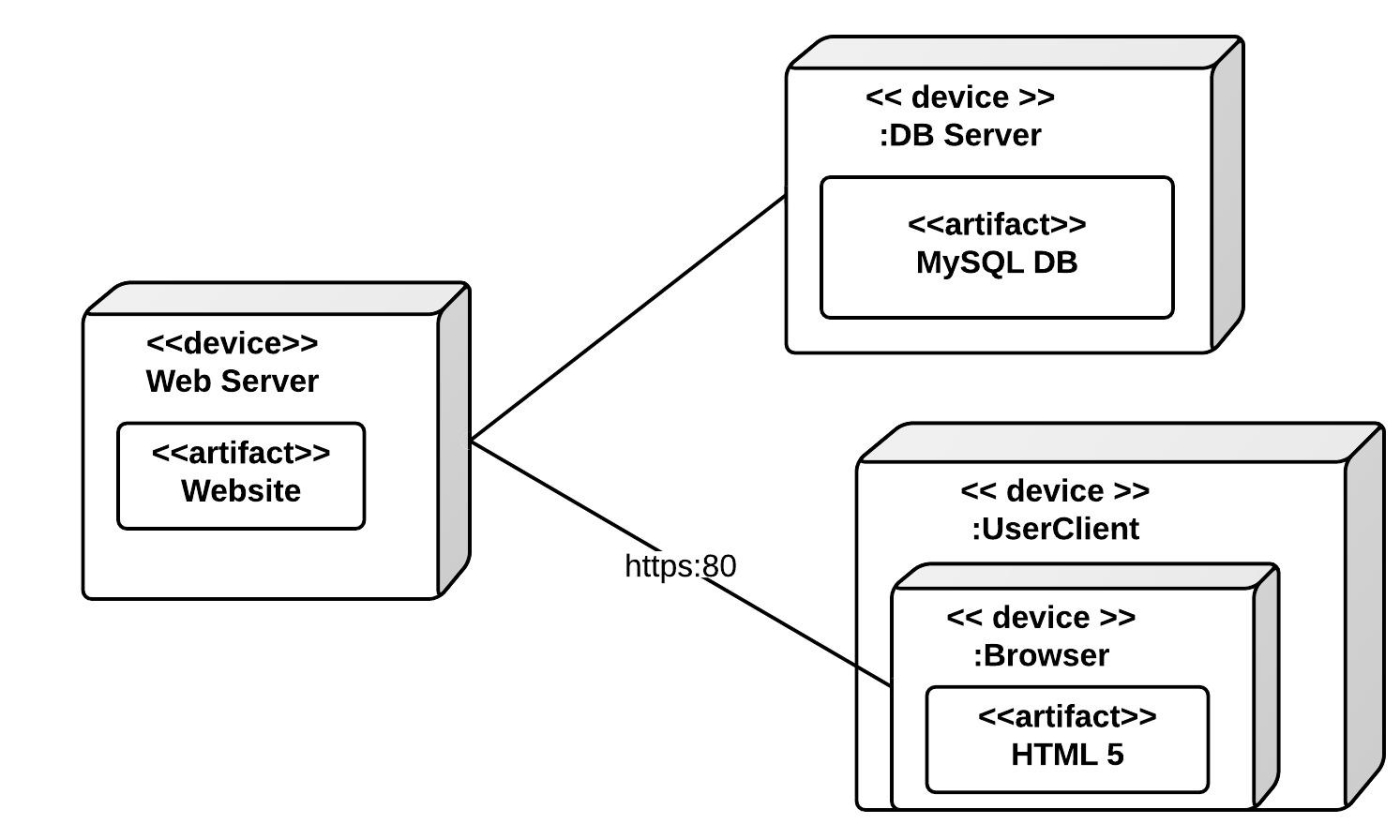
**6. UML Diagram**

**6.1 Class Diagram**

****

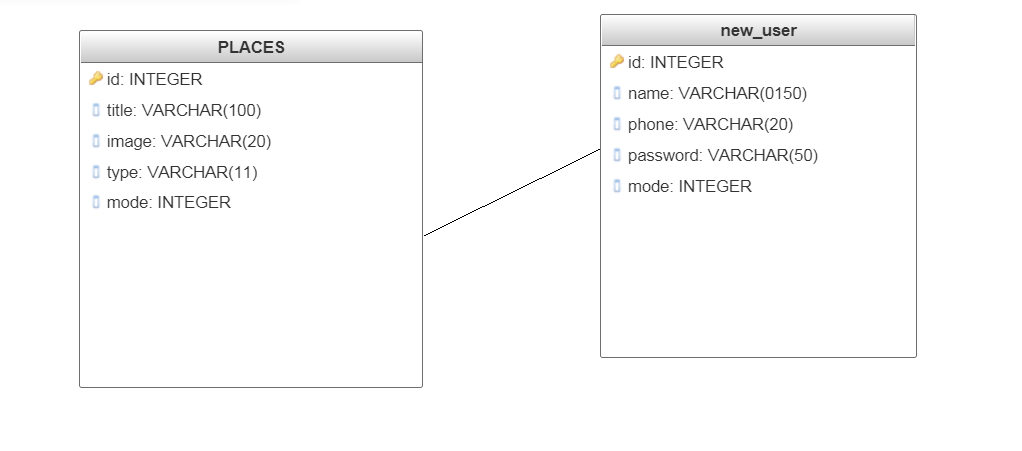
**FIGURE NO. 4**

**6.2 Deployment Diagram**

****

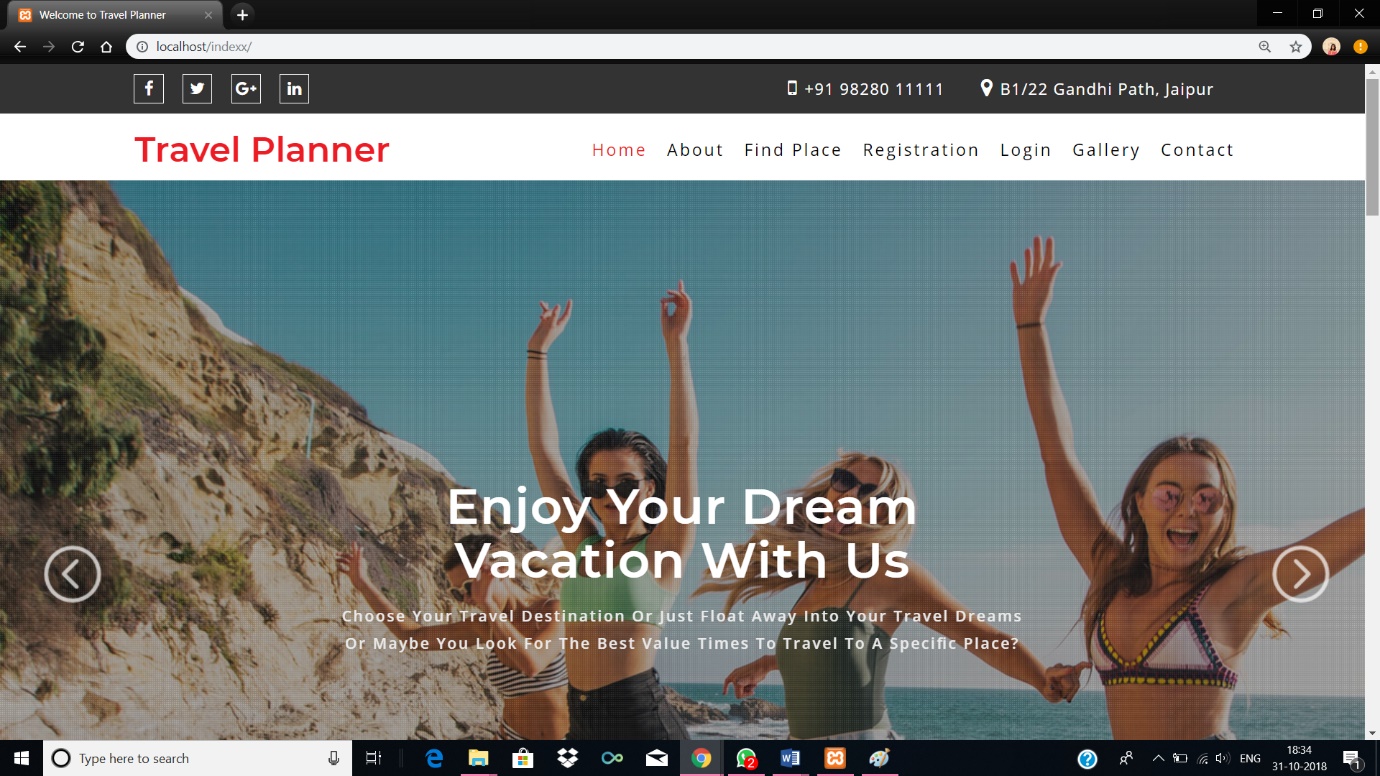
**FIGURE NO. 5**

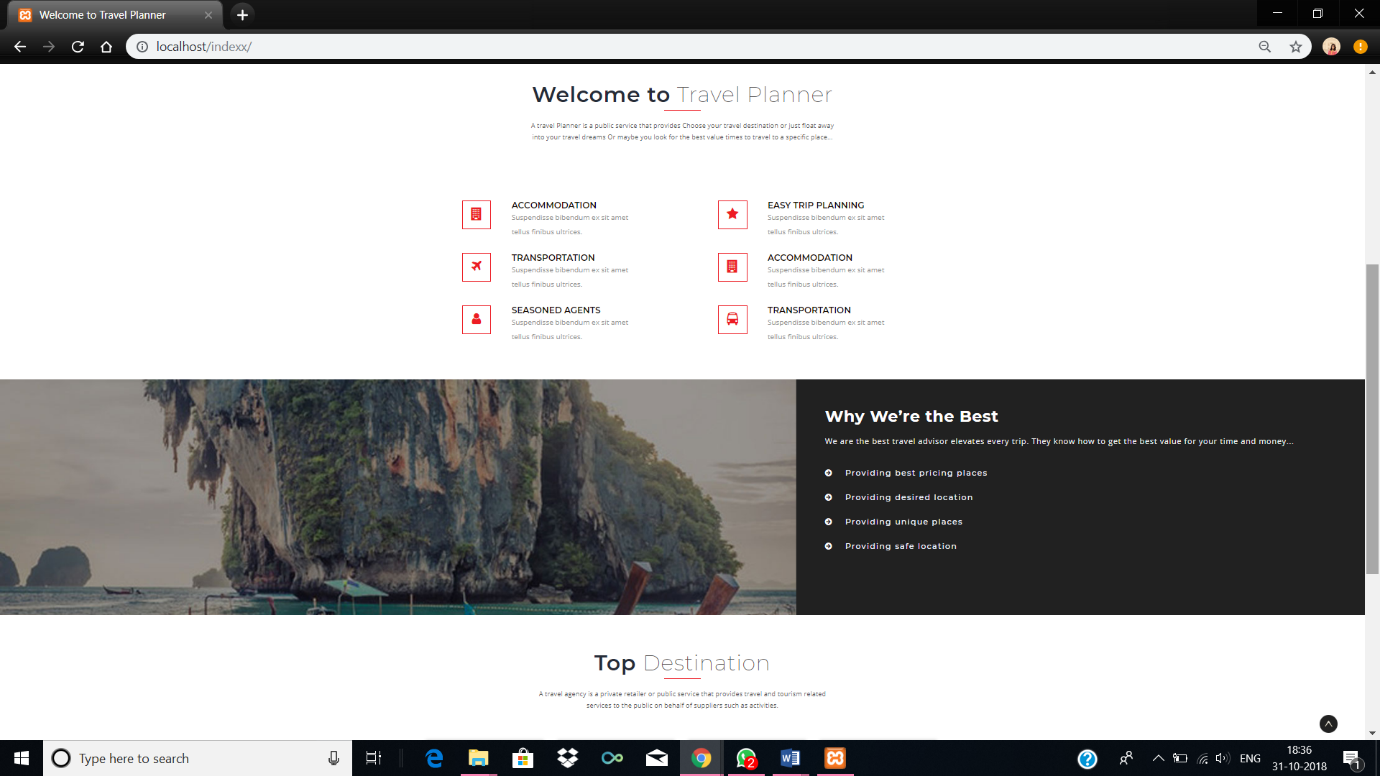
1. **Database Diagram**

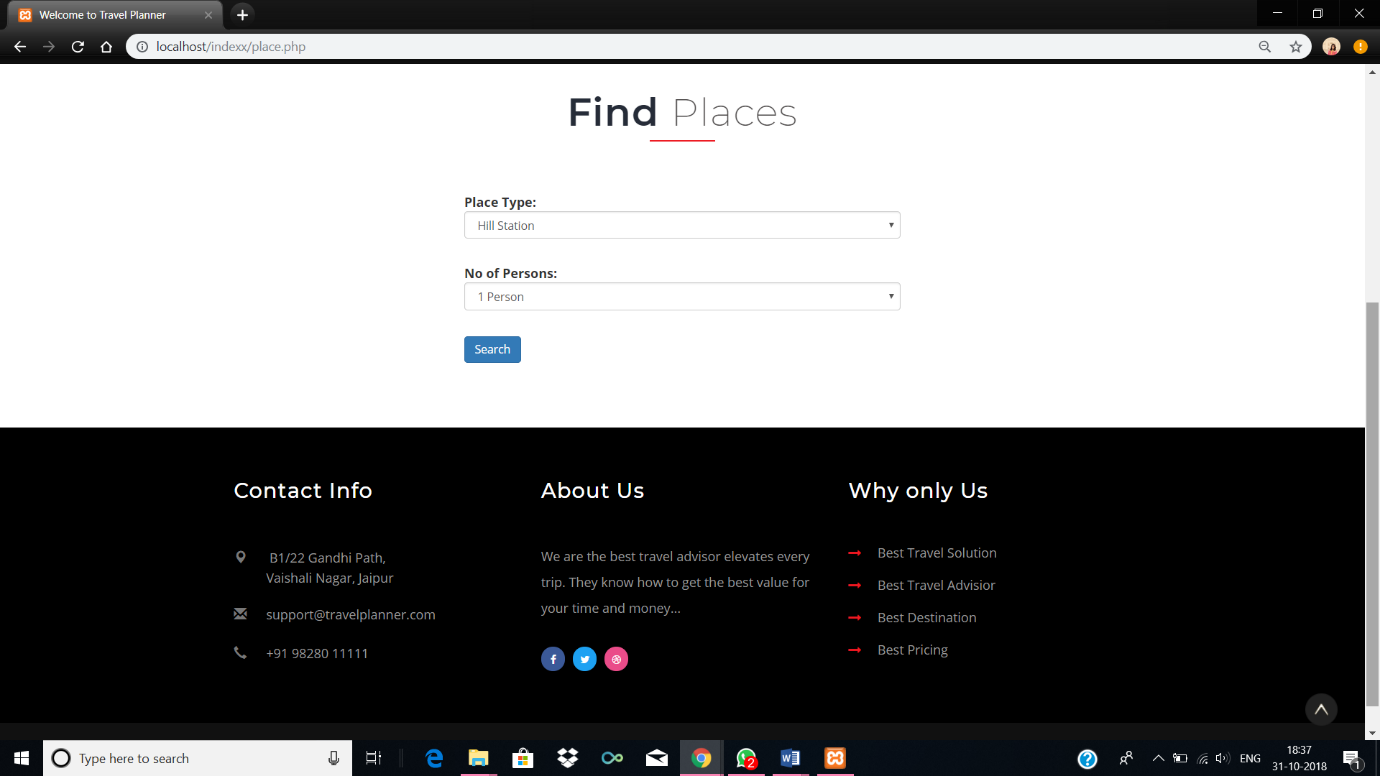
****

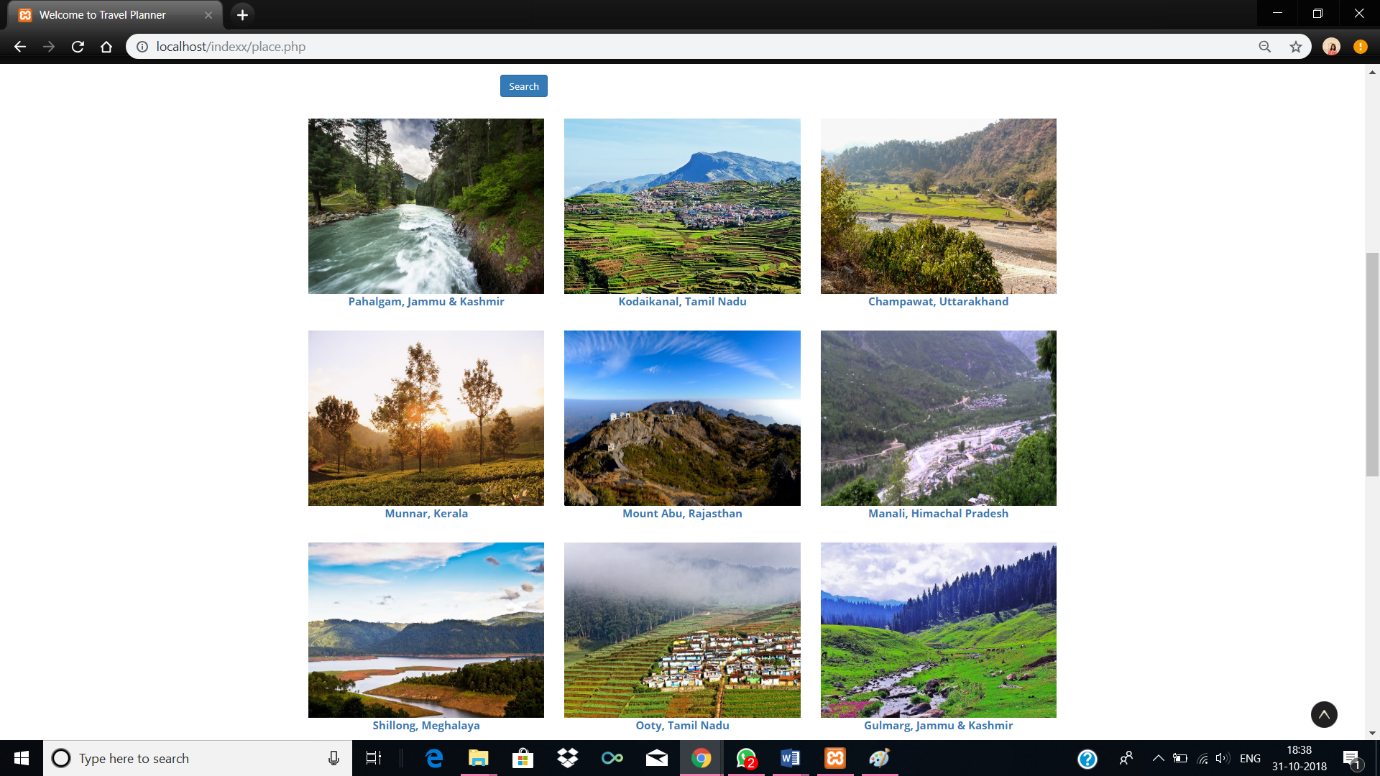
**FIGURE NO. 6**

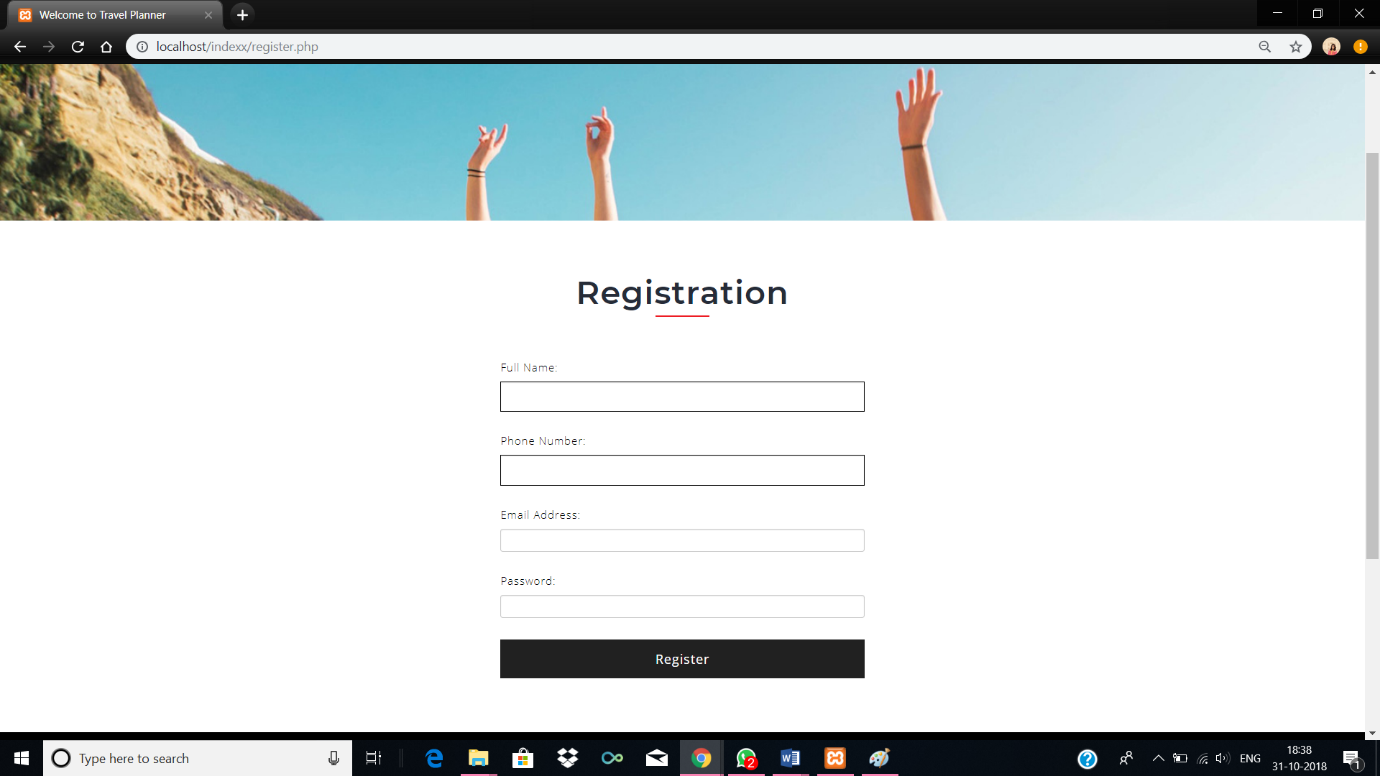
1. **GUI Design**

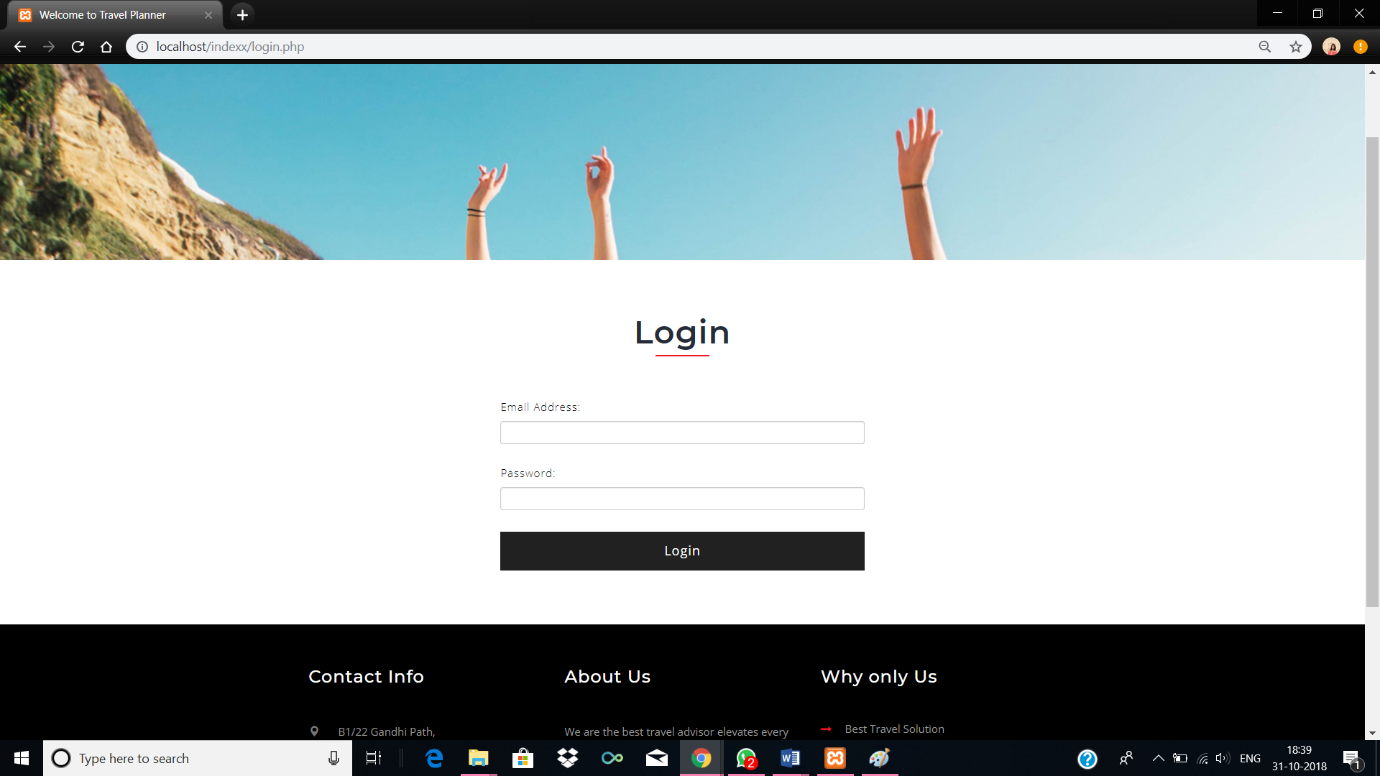
****

****

****

****

****

****

1. **Application Program Interface**

|  |  |  |
| --- | --- | --- |
| **Sr. Number** | **Name of library** | **Description** |
|  | pChart | Helps to generate text data in a visual form. |
|  | DISPATCH | This can define URL rules to better organize the site. |
|  | PHP text to image library | As the name suggests, it converts text into an image |
|  | PHP DB Class | Helps us to organize and help with the database connectivity. |
|  | PHP Errors | This PHP library helps in detecting errors along with Ajax works, code snipers and syntax highlights. |
|  | PHP PASSWORDLIB | This is meant for satisfying the crypto-graphic needs that may arise for the developer |
|  | WHOOPS | helps in easy management of errors and exceptions resulting in better management of the project. |

**References**

**Books**

1. The Joy of PHP Programming: A Beginner’s Guide – by Alan Forbes
2. PHP & MySQL Novice to Ninja – by Kevin Yank
3. Head First PHP & MySQL – by Lynn Beighley & Michael Morrison
4. Learning PHP, MySQL, JavaScript, and CSS: A Step-by-Step Guide to Creating Dynamic Websites – by Robin Nixon
5. PHP & MySQL Web Development – by Luke Welling & Laura Thompson

**Links**

1. <http://php.net/manual/en/index.php>
2. <http://www.tizag.com/phpT/>
3. <https://devzone.zend.com/6/php-101-php-for-the-absolute-beginner/>
4. <http://www.hackingwithphp.com/>
5. <https://www.killerphp.com/tutorials/object-oriented-php/>
6. <https://www.w3schools.com/php/default.asp>
7. <https://www.lynda.com/PHP-tutorials/PHP-MySQL-Essential-Training-1-Basics/587674-2.html>

**Appendices**

1. **https:**

Hypertext Transfer Protocol Secure (HTTPS) is an extension of the [Hypertext Transfer Protocol](https://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol) (HTTP) for [secure communication](https://en.wikipedia.org/wiki/Secure_communications) over a [computer network](https://en.wikipedia.org/wiki/Network_operating_system), and is widely used on the [Internet](https://en.wikipedia.org/wiki/Internet).[[1]](https://en.wikipedia.org/wiki/HTTPS#cite_note-1)[[2]](https://en.wikipedia.org/wiki/HTTPS#cite_note-2) In HTTPS, the [communication protocol](https://en.wikipedia.org/wiki/Communication_protocol) is [encrypted](https://en.wikipedia.org/wiki/Encrypted) using [Transport Layer Security](https://en.wikipedia.org/wiki/Transport_Layer_Security) (TLS), or, formerly, its predecessor, Secure Sockets Layer (SSL). The protocol is therefore also often referred to as HTTP over TLS,[[3]](https://en.wikipedia.org/wiki/HTTPS#cite_note-3)or HTTP over SSL.

The principal motivation for HTTPS is [authentication](https://en.wikipedia.org/wiki/Authentication) of the accessed [website](https://en.wikipedia.org/wiki/Website) and protection of the [privacy](https://en.wikipedia.org/wiki/Information_privacy) and [integrity](https://en.wikipedia.org/wiki/Data_integrity) of the exchanged data while in transit. It protects against [man-in-the-middle attacks](https://en.wikipedia.org/wiki/Man-in-the-middle_attack). The bidirectional [encryption](https://en.wikipedia.org/wiki/Block_cipher_mode_of_operation) of communications between a client and server protects against [eavesdropping](https://en.wikipedia.org/wiki/Eavesdropping) and [tampering](https://en.wikipedia.org/wiki/Tamper-evident#Tampering) of the communication.[[4]](https://en.wikipedia.org/wiki/HTTPS#cite_note-httpse-4) In practice, this provides a reasonable assurance that one is communicating without interference by attackers with the website that one intended to communicate with, as opposed to an impostor.

1. **Client-server model:**

The client–server model is a [distributed application](https://en.wikipedia.org/wiki/Distributed_application) structure that partitions tasks or workloads between the providers of a resource or service, called [servers](https://en.wikipedia.org/wiki/Server_(computing)), and service requesters, called [clients](https://en.wikipedia.org/wiki/Client_(computing)).[[1]](https://en.wikipedia.org/wiki/Client%E2%80%93server_model#cite_note-1) Often clients and servers communicate over a [computer network](https://en.wikipedia.org/wiki/Computer_network) on separate hardware, but both client and server may reside in the same system. A server [host](https://en.wikipedia.org/wiki/Host_(network)) runs one or more server programs which share their resources with clients. A client does not share any of its resources, but requests a server's content or service function. Clients therefore initiate communication sessions with servers which await incoming requests. Examples of computer applications that use the client–server model are [Email](https://en.wikipedia.org/wiki/Email), [network printing](https://en.wikipedia.org/wiki/Network_printing), and the [World Wide Web](https://en.wikipedia.org/wiki/World_Wide_Web)

**Guides Comments**