**Registration Page Specifications**

Wei Guang Yan

**Contents**

[**1.** **File List** 1](#_Toc6238876)

[**2.** **Function Details of some files** 1](#_Toc6238877)

[**2.1** **Register Page: "register.php"** 1](#_Toc6238878)

[**2.2** **Validation File: "regprocess.php"** 1](#_Toc6238879)

[**2.3** **Database Query File: "dbfunctions.php"** 2](#_Toc6238880)

[**3.** **Technology Problem and Solutions** 3](#_Toc6238881)

# **File List**

**Files for registration function in Travel Expert website**

|  |  |  |
| --- | --- | --- |
| **File Name** | **File to support** | **Functions** |
| **Main File** | | |
| register.php | (index.php) | Provide form for customer to create account and profile. |
| **Supporting files** | | |
| regvalidchk.js | register.php | Utilize OOP to make validation on each input. |
| registerstyle.css | register.php | Setting layout and style of main file “register.php”. |
| regprocess.php | register.php | Proceed data from main file, check whether user ID exist, write data to database. |
| customerclass.php | regprocess.php | Define two classes for OOP in “regprocess.php”. |
| dbfunctions.php | regprocess.php | Define database operation functions, and write log file as record during database visiting. |
| dbvariables.php | dbfunctions.php | Set log in information for link database. |

# **Function Details of some files**

## **Register Page: "register.php"**

Provide a form to customer to fill out his(her) information, validate all input's information (with javascript) before sending to "regprocess.php". The validation processes is written in "regvalidchk.js".

## **Validation File: "regprocess.php"**

**2.2.1 Step 1: Data pre-processing**

2.2.1.1 Import data from "$\_POST" into an associative array;

2.2.1.2 Slice the array into two parts:

(1) First array includes User Id and password;

(2) Second array include user name and contact information.

2.2.1.3 Declare two objects: "$Custlgin" and "$Cust", assign values during declaration.

(1) First object:

Set value Cust ID and password, which will be sent to table "customer\_login";

(2) Second object:

Set cust detailed information, which will be sent to table "customers".

**2.2.2 Step 2: Check whether the new customer user ID has existed**

It will use two table, “customer\_login” and “customers”, to save customer log in information and profile (contact information), respectively.

(1) Check whether table "customer\_login" exist?

(2) If did not find the table, create the table first.

(3) Check whether the user ID has been used (exist in table),

(4) If it exists, redirect back to register page and show warning message.

**2.2.3 Step 3: Write data to database**

2.2.3.1 pre-process array. Seperate array from $\_POST into 2 objects.

2.2.3.2 call relative function in "dbfunctions.php" to make query to database.

(1) Write customer profile first and get the “customerID” of the rocord.

(2) Hash password, and write log in information into table.

## **Database Query File: "dbfunctions.php"**

(1) Connect to database;

(2) Check whether a table exist;

(3) Create a table;

(4) Insert row(s) in table;

(5) Select special cell(s) in a table;

(6) Update special cell(s) in special user ID row.

# **Technology Problem and Solutions**

During the web design process, I met technical requirements and solved them as list below:

|  |  |  |
| --- | --- | --- |
| Location | Requirements (Problems) | Solutions |
| register.php  regprocess.php | Redirect to one page and redirect back to previous page | Use “$\_SESSION” to save and transfer page url. |
| register.php  regprocess.php | Message cannot be shown before page redirection and redirection error (not work and show error message) | Use PHP operaton ‘header(”location: $pgfrom”);’ or javascript operation ‘window.location.replace(“index.php”);’ depends on situations. |
| regprocess.php | Hash one element (password ) of an array before write to database | Convert array to object. Utilize OPP method (“get” and “set”) to change the value of a property. |
| regprocess.php | Write data to a table (log in table) not exist | Check whether table “customer\_login” exist. If not, create it and set field “CustomerId” as foreign key to reference to table “customers”. |
| register.php | Page layout design | Link to .css file (registerstyle.css) to setting position, fonts, color, etc. |
| regvalidchk.js | Form validation | Link to .js file (regvalidchk.js) to set validation regulars |
| register.php  regvalidchk.js | Dynamic show message on validation | Utilize OOP method, change content and color of <p>tag |
| register.php  regvalidchk.js | Event operation on focus/un-focus on Input cells and buttons | Define “onFucus”, “onBlur” and “onclick” events to input and buttons. |
| dbfunctins.php | Read/write database | Utilize database statements (such as “mysqli\_connect”, “mysqli\_query”) to visit to database. |
| dbfunctions.php | Record operation and error information during database visiting | Utilize file read/write statement to write log file to record operation. |
| regvalidchk.js | Create friendly and easy step to instruct customer fill out the form | Set a series logical decisions to check error during input and specialize the error details to customer. |