

Web assignment-Yan Zhang

by Yan Zhang

FILE	12042_YAN_ZHANG_WEB_ASSIGNMENT-YAN_ZHANG_155469_2069834586.DOCX (1.42M)		
TIME SUBMITTED	23-JAN-2019 10:59AM (UTC+0000)	WORD COUNT	2360
SUBMISSION ID	99673068	CHARACTER COUNT	13338



WEB TECHNOLOGIES FOR ECOMMERCE

Interactive Website using a range of web technologies

Table of content

Abstract	1
Introduction.....	1
Overall structure.....	1
Database design	2
Website issues.....	2
Solutions and applications	3
1. PHP Sessions	3
2. Captcha	4
3. Hidden.....	5
4. Prepared statements	5
5. Data validation.....	6
6. Password	9
7. Other methods.....	10
Conclusion	10
References.....	10

Abstract

The author adopts the combination of PHP and MySQL to develop a convenient and practical vehicle information management website, which stores vehicle information and owner information. This article will mainly discuss the issues and solutions that the author considered when developing this live website.

Introduction

With the widespread of internet technology, websites have become an essential means for people to store and obtain information. The combination of PHP and MySQL is most common in web development. PHP has strong cross-platform features and supports most of the current databases [1], and MySQL is a very good security permission management system[2]. However, PHP websites may face various problems due to its defects and insufficient security awareness of developers, such as security, scalability, availability, accessibility and so on. Therefore, when building a dynamic information management system based on the database, we should fully consider the issues that may affect dynamic websites.

Overall structure

A successful website system should be safe and reliable, easy to operate, and easy to search. In the design of the website, the developer must plan the functions to be implemented by the website in advance, so that to have a better overall view of the developed website. Therefore, it is necessary to draw a structure block diagram before programming the code.

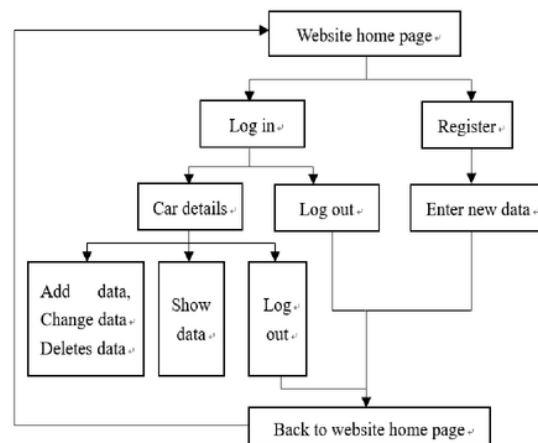


Figure 1 overall structure of the website

According to figure1, we need to implement five functions:

- The first function is used to implement user login.
- The second function is used to implement user registration.

- The third function is used to add, modify, and delete data.
- The fourth function is used to display user information.
- The fifth function is used to log out.

Through the above figure, we can know how to proceed to the next website design, which will help us to have a general understanding of the design of the required website.

Database design

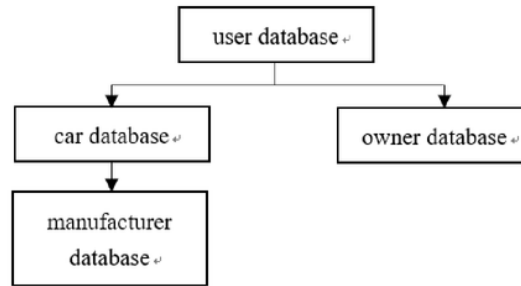


Figure 2 database structure

According to figure2, we need five databases:

- 1) The user database is to link car database and owner database. It contains Uid (user id), Userdetails (owner information) and Cardetails (car information). When using the MySQL statement to query the database for data, the owner database, and car database are connected by Userdetails and Cardetails. All data's type is INT.
- 2) The owner information database is to store owner information. It includes Oid (owner id), Firstname (First name), Surname, Username, Password, Time, Qusetion, Answer. The Oid's type is INT, and the others type is VARCHAR.
- 3) The car database is to link the manufacturer database. It contains the Carid(car id), Carmanufacturer (car manufacturer id), Carpig (car picture). When using the MySQL statement to query the database for data, the manufacturer database and the car database are connected by the Mid and the Carmanufacturer. The Carpig's type is VARCHAR, and the others type is INT.
- 4) The manufacturer database is used to store car manufacturer information. It contains Mid(car manufacturer id), Type(car type) and Mame(manufacturer name). The Mid's type is INT, and the others type is VARCHAR.

It is important to note that the data in the database is not case sensitive by default, so when the database query is performed, the query results may be affected. Therefore, we can set the order of each column in the data table to latin1_general_cs, so that when data query, the data in the database can be recognized in upper and lower case.

Website issues

A website may face the following issues[3]:

- The developer non-maliciously exposes portions of the user's data, which may be exploited by unauthorised persons.
- The user's sensitive data may be lost or damaged, which may lead to unauthorised access to sensitive data.
- Use inter-site scripts for malicious code injection, such as entering some scripts in the form that can be executed on the client side, resulting in varying degrees of user information loss, or incorrectly guiding the user's operation without the user's knowledge.
- Denial of service attack will make the server stop working for more than hours so that the site visits within a short period sharply reduced.
- The illegal user breaks into the server, and obtains the system access permission, carries on the arbitrary control to the server which breaks into, the illegal execution command, closes the server or installs the illegal software and so on.

Solutions and applications

1. PHP Sessions

Some unauthorised users may bypass the login page to enter each submodule and modify the information. For example, when the user logs on the website and enters the personal information management page, the URL of this page will be displayed, as shown in figure 3. Once an unauthorised user obtains this URL, he can skip the login page and modify the data, so this is dangerous and dramatically affects the privacy of the user.

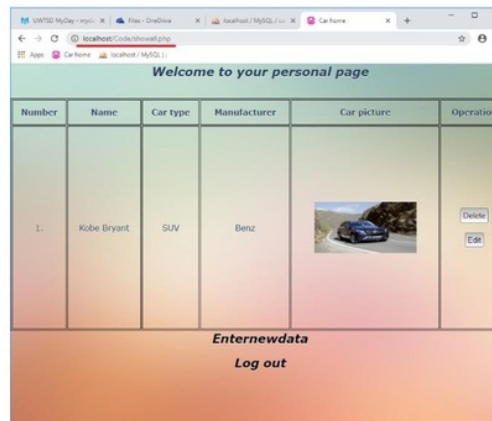


Figure 3

In order to strengthen the security of system information management and prevent unauthorised users from bypassing the login page to enter each information submodule and modify the information, it is necessary to start a legal user login session process

with `session_start()` command, which is used for user judgment in each management page. It can close a session and destroy all data in the current session, but it does not reset global variables in the session. When the user logs out of the website, use the `session_destroy()` command to close a session and reset all global variables, which can intercepts illegal user attempts to log in to the information management system. Therefore, once an unauthorised user wants to use this kind of URL(such as shown in figure 3), the website will remind him that this website page not accessible. As shown in figure 4.

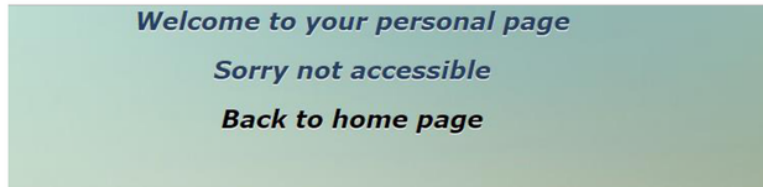


Figure 4

2. Captcha

Although web forms are useful, they may also be attacked by website robots. For example, if criminals use a username to log in to the site, he may continue to try the password through the robot, which may lead to password leakage. Therefore, we can solve this problem by adding a captcha to the login page. Moreover, the purpose of this method is to prove that this user is a person, not a robot. There are many advantages to using captchas.

- Prevent malicious password cracking. For example, some hackers, in order to obtain user information, send data to the server through different means to verify the accuracy of guessing user information.
- Prevent malicious requests. For example, when a user submits a form of information, it will waste server resources and cause malicious attacks by constantly requesting data information from the background.
- Become interesting. When we submit forms, such as encountered some interesting verification code way, can enhance the user's love of the site.

When programming code for captcha, the main steps are as follows:

- 1) Use the `srand()` function to broadcast the random number generator seed.
- 2) Use the `rand()` function to generate a non-repeating random number between 0-15, and then use the `dechex()` function to convert the decimal number to a hexadecimal number. Therefore, the result contains both the number 0-9 and the letter a-f.
- 3) Use the `imagecolorallocate()` function and `imagepng()` function to generate the captcha image, and finally use the `imagedestroy()` function to release the memory occupied by the image.

Through the above steps, the captcha design is completed, which can significantly improve the security of the website. The designed login page is shown in figure 5.

Figure 5

3. Hidden

The "hidden" type of `<input>` can be used when a form submits data that will not be seen or modified by the user. As a result, hidden input is entirely invisible in the rendered page and cannot be displayed in the content of the page. Figure 6 shows the hidden type input box used in the design of this website. It can help developer pass some data that we do not want to be seen externally.

```
<form action="changedata.php" method="post">
<input type="hidden" value="<?php echo $r['Uid'];?>" name="uid">
<input type="hidden" value="<?php echo $r['Userdetails'];?>" name="oid">
<input type="hidden" value="<?php echo $r['Carid'];?>" name="cid">
<input type="hidden" value="<?php echo $type[$i];?>" name="thetype">
<input type="hidden" value="<?php echo $mname[$i];?>" name="themname">
<input type="hidden" value="<?php echo $carpic[$i];?>" name="carpic">
```

Figure 6

4. Prepared statements

A problem facing database-enabled sites is that of malicious commands being sent to the server in place of legitimate data. Different data can be retrieved by making some simple changes to what is entered in the form. For example, the string is created as follows, effectively bypassing the username checking mechanism. As shown in figure 7, this string can retrieve all user data whose password is test1. Besides, the developer can use other SQL injection commands, including "DROP tables," which would be

disastrous. Therefore, we must carefully construct the code to avoid SQL injection.

Enter your username and password:

Username:

Password:

Thank you for logging in. Here are your details:

1 Gaynor Thomas 1 The Road, Swansea gaynor test1
2 Kevin Palmer 1 The Street, Pontlliw kevin test1

Figure 7

Prepared Statements are much like stored procedures. They are a collection of SQL statements running in the background. We can get many benefits from using prepared statements, whether it is performance issues or security issues.

- Only do one query, but the developer can use the same or different parameters multiple times, and the process will significantly reduce the speed of the application. In short, the prepared statement takes up fewer resources and runs faster.
- The parameters supplied to the prepared statement do not need to be enclosed in quotes, and the driver will handle it automatically.
- The prepared statement is beneficial for SQL injection because the parameter values are sent after using different protocols to ensure the legality of the data.

The code usage of prepared statements is shown in figure 8. This example is to query user data based on username and user password.

```
$query="SELECT * FROM owner WHERE $sb=? AND Password=?";  
$stmt=mysqli_stmt_init($link);  
mysqli_stmt_prepare($stmt,$query);  
mysqli_stmt_bind_param($stmt,"ss",$nm,$pw);  
mysqli_stmt_execute($stmt);  
$result=mysqli_stmt_get_result($stmt);  
$r=mysqli_fetch_array($result);
```

Figure 8

5. Data validation

When developing a web project, it is inevitable to encounter various processing of web page data, such as verifying whether the user input in the form is a null value, password length and so on. In order to avoid meaningless access to the database, the developer can use Javascript technology and use regular expressions to perform some necessary validation on the form before it is submitted, such as checking whether the input is null,

string length, string format, and so on.

In order to verify these basic data, the author writes Javascript program. As is shown in Figure 9 (This figure only shows the data validation of the login page, but user registration, modification of user data and other pages of the data validation program are not shown).

```
function Login_validation() //Form validation
{
    var nm = LTrim(document.forms["myForm"]["nm"].value); //trim whitespace
    var pw = LTrim(document.forms["myForm"]["pw"].value);
    var cpa = LTrim(document.forms["myForm"]["cpa"].value);
    if(nm == null || nm == "" || pw == null || pw == "" || cpa == null || cpa == "") {
        if((nm == null || nm == "") && (pw == null || pw == "") && (cpa == null || cpa == "")) {
            alert("Please enter your name and password and captcha");
        }
        else if((nm == null || nm == "") && (pw == null || pw == "") && (cpa != null || cpa != "")) {
            alert("Please enter your name and password");
        }
        else if((nm == null || nm == "") && (cpa == null || cpa == "") && (pw != null || pw != "")) {
            alert("Please enter your name and captcha");
        }
        else if((pw == null || pw == "") && (cpa == null || cpa == "") && (nm != null || nm != "")) {
            alert("Please enter your password and captcha");
        }
        else if((nm == null || nm == "") && (pw != null || pw != "") && (cpa != null || cpa != "")) {
            alert("Please enter your name");
        }
        else if((pw == null || pw == "") && (nm != null || nm != "") && (cpa != null || cpa != "")) {
            alert("Please enter your password");
        }
        else if((cpa == null || cpa == "") && (nm != null || nm != "") && (pw != null || pw != "")) {
            alert("Please enter your captcha");
        }
        return false;
    }

    if(!preg_match("/^[a-zA-Z]*$/",nm) || !preg_match("/^[a-zA-Z]*$/",pw)) //regular expression
    {
        alert("Please check the input format");
        return false;
    }
}
```

Figure 9

Onsubmit="JSCode". This code means that Javascript can be executed when the form is submitted. Therefore, when the user submits the login form, the data from that form is sent to Login_validation for validation. As is shown in Figure 10.

```
<form action="checklogin.php" onsubmit="return Login_validation()" method="post" name="myForm">
Name:<input type="text" name="nm"><br><br>
<input type="radio" name="searchbut" value="Username" checked="checked">Username
<input type="radio" name="searchbut" value="Surname">Surname
<input type="radio" name="searchbut" value="Firstname">Firstname<br><br>
Password:<input type="password" name="pw" id="pass" class="pass">
<input type="button" name="" id="btn" value="show"><br><br>
```

Figure 10

For example, when a new user registers, if a user forgets to fill in a username, a dialogue box will appear to remind the user to fill in the username when the form is submitted. Besides, in order to improve security, the length of the password filled in by the new user should be limited. If the length of the meter is too short, a dialogue box will pop up to remind the user (It is worth noting that there's no need to force users to upload photos when new users sign up). The effect is shown in figure 11.

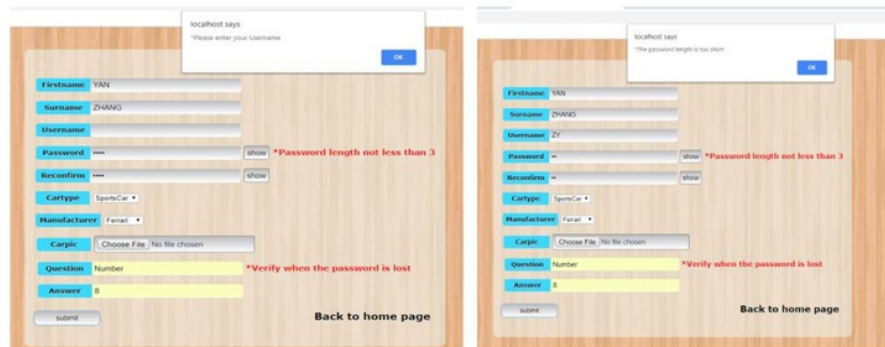


Figure 11

In addition to basic validation of the data, the website also needs to connect to the database for validation. For example, when a user logs in, verify that the user name and password whether they are correct and when a new user registers, verify that the user name whether it has been registered. As shown in figure 12, it is to check the registered user name of the new user.

```
$query="SELECT * FROM owner";
$result=mysqli_query($link,$query) or die("didn't work");
while($r=mysqli_fetch_array($result)) //check username
{
    if($un==$r["Username"])
    {
        $unerror=1;
    }
}
//check for password and confirm password
$pw=trim($pw);//trim whitespace
$cpw=trim($cpw);//trim whitespace
$pcperror=0;
if($pw!=$cpw)
{
    $pcperror=1;
}
if($unerror!=0 || $pcperror!=0)//must be error
{
    header("location:register.php?&unerror=$unerror &pcperror=$pcperror");
}
```

Figure 12

If the user name is occupied, an error value will be returned to the registration page. The registration page will use the `$_GET[]` method to get the error value and remind the user that the user name has been registered according to the error value. As shown in figure 13.

Register:

Firstname: YAN

Surname: ZHANG

Username: KB *The username has been used

Figure 13

6. Password

When a user enters a password, if the password is not encrypted, there is a possibility that others see it, so the developer must ensure the security of the password. Thus, the `<input>` type selects the password type so that the password is encrypted and not seen by others. What is more, in order to facilitate the user to check the entered password, a password display button is added to the right of the password input box, so that the user selects to view the password entered. As is shown in figure 14.

Log in

Name: KB

* Username Surname Firstname

Password: ***** show

Captcha: 6 b 2

Forget password?

submit Back to home page

Figure 14

If we forget our password, we can click the link to retrieve it. This link requires us to fill in some verification information, when we complete the verification of relevant information, we can retrieve the password. As is shown in figure 15.

Retrieve password

Firstname

Surname

Question

Answer

Captcha: c 7 6 c

submit Back to home page

Figure 15

7. Other methods

In order to increase the observability of the website, the author used CSS style sheets to enhance the overall layout of the webpage during the development of this website, which makes the website look comfortable.

The web page adds an administrator function that allows users to modify data, delete data, add data.

Another noteworthy thing is that when the new user registers, the new user's registration time will be added to the database so that we can know the user's registration time and facilitate the management of the user in the future. For example, the user level can be divided according to the registration time of the user.

Conclusion

The author adopts the combination of PHP and MySQL to develop a convenient and practical vehicle information management website and lists the problems encountered in the process of website development and the solutions. Although the author encountered many problems in the process of web development but also learned a lot of skills, which is very conducive to the author's mastery of web development technology.

References

- [1] Wikipedia, "PHP", Online source: <https://en.wikipedia.org/wiki/PHP>.(accessed 5/1/2019)
- [2] Wikipedia, "Mysql", Online source: <https://en.wikipedia.org/wiki/MySQL>. (accessed 5/1/2019)
- [3] Sebastian, "Security 1:1 - Part 3 - Various types of network attacks", Online source: <https://www.symantec.com/connect/articles/security-11-part-3-various-types-network-attacks>. (accessed 5/1/2019)

Web assignment-Yan Zhang

GRADEMARK REPORT

FINAL GRADE

65/100

GENERAL COMMENTS

Instructor

WEBSITE

- Range of PHP coding demonstrated - A good range demonstrated. Of particular note was the user of add/edit/delete and the user of populated pulldown menus and preventing duplicate usernames. Also, the login system was very good, and the registration system was extensively validated.

You have not demonstrated any search facilities which was a shame, nor can you do anything on the site without logging in. A website should have at least some content for anonymous users, to give them an incentive to become a registered user. Also, be careful on the login system - it is not appropriate to use firstname or surname to log in as they are not unique values.

23/30

- Database structure and design – 4 tables were provided, and all have been suitably normalised. Nothing to fault here, well done!

10/10

- Overall Integration of Website - Front end layout is functional, but a bit basic!

5/10

Knowledge and Understanding of Subject – A good understand was demonstrated, other than the issues with the unique login etc.

16 /20

REPORT

Research-informed Literature – Some really good discussion of security issues and the main security point have been covered. However, you have written very little about usability, accessibility and scalability

6/15

Analysis – A reasonable analysis of how what you have researched applied to your site, but is not complete because the research section is also not complete.

5 /15

Please note that Wikipedia is not a suitable reference for an academic report as it can be edited by anyone. Try to find the original reference instead.

Overall = 65%

PAGE 1

PAGE 2

PAGE 3

PAGE 4

PAGE 5

PAGE 6

PAGE 7

PAGE 8

PAGE 9
