



CS-GY 6083 B

Principles of Database Systems

Project Part 2

Xinyu Zhao: xz2671, N12924775

Tieliu Zhou: tz1430, N14973953

December 20, 2020

Part 1: Execute Summary of the Business Case.....	3
Part 2: Brief details of software, language, and database used	6
Part3: List of tables, and total number of records of each table.....	7
Part4: Screenshots of some sessions, pages, menus of your Web Application	9
Part 5: Details of security features that We have implemented on Web Application development	24
Part 6: Journey of Project and Summary.....	26
Part 7: Business analysis with 4 SQLs	27
Part 8: Extra Features	30

Part 1: Execute Summary of the Business Case

In our design, we helped the WOW (World On Wheels), a car rental company which has many locations at various airports, towns, and cities in the United States, to convert its file system base data management to sophisticated centralized database system and to create a web-based user interface for the database schema. So in our designed web-based user interface, the customer can register, login, create a service, pay for the car rent fee and see the history of his/her rent history. Meanwhile, the administrator and employee of the WOW can perform business activities (CURD – Create, Update, Read, Delete records).

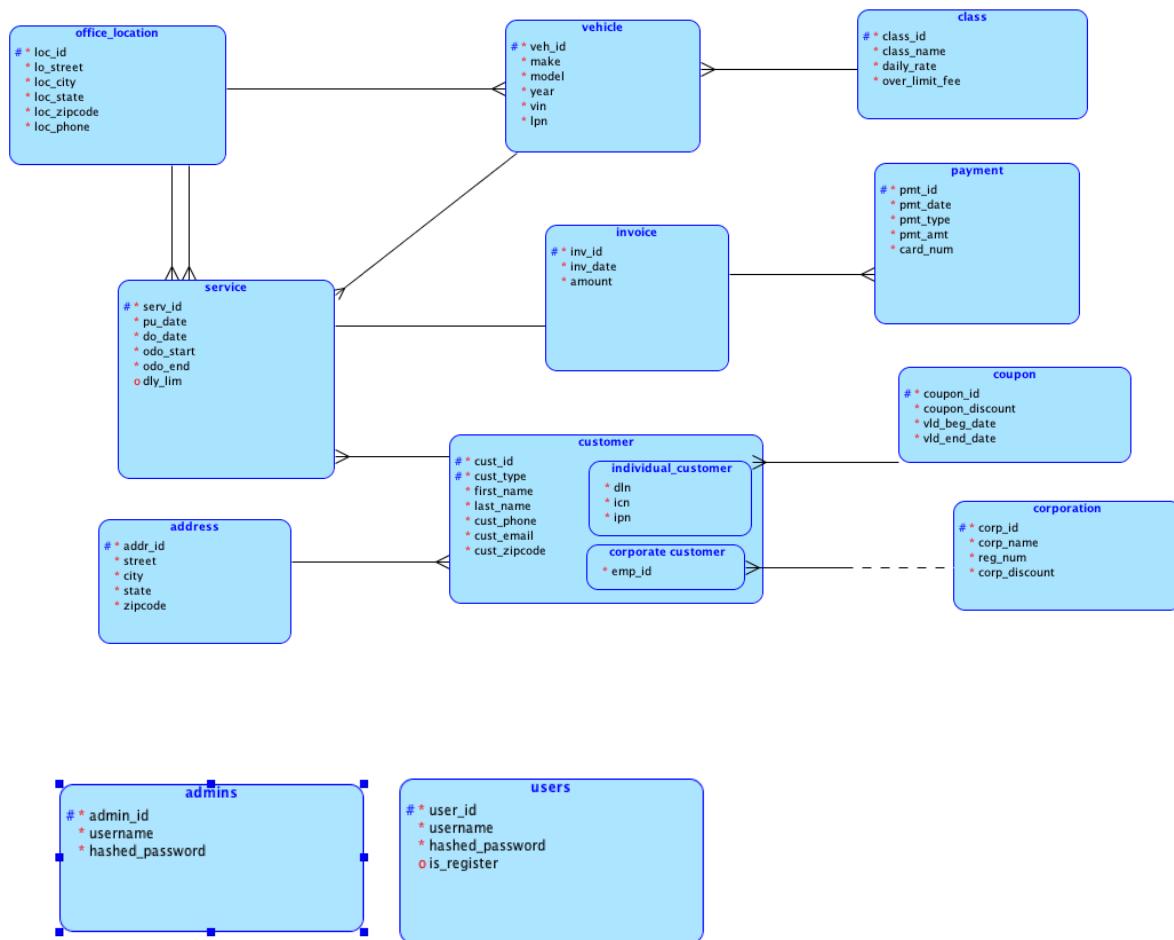
Based on part 1 of this project, we have designed the database by different entities based on each one's own function and in every entity there are different attributes. This can improve the WOW centralized database system performance because we can make some small changes by entity or attribute without affecting the whole database system.

There are some approaches that we have used towards business solution. The buttons on different login user interfaces are different and the operability is different. Administrator users can delete and modify customer users. When customer users log in to their accounts, they can search for the available cars at their wished time and location. And they can estimate a total mileage to get the estimated total amount for their rent fee. When they finished their trip, they can see the invoice for this trip and pay for it with different payment methods. They can view historical order records and view various invoices and payments, and if an invoice is paid in full, it cannot be paid again, the pay button will not be displayed and the word “PAID” will be displayed.

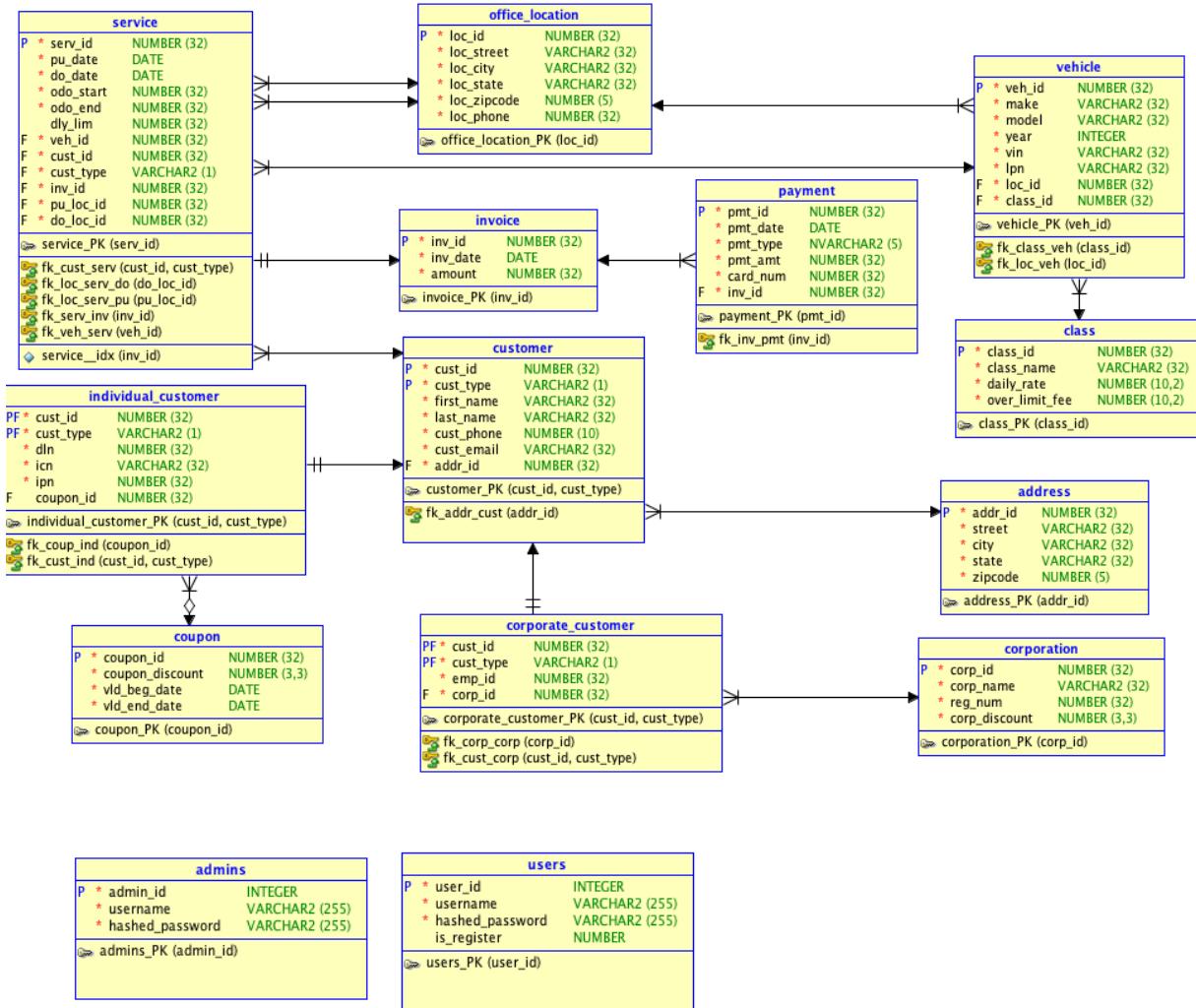
Our assumptions of the WOW database are: One customer can have multiple rental service.

Each customer has one address associated with it. A vehicle can be used in one or many rental services. One coupon can be given to multiple customers. A customer may or may not have one or many coupons. For each rental service there is one invoice. Amount of an invoice can be paid in multiple payments.

Logical Model Design



Relational Model Design



```

CREATE TABLE history(
    hist_id INT NOT NULL AUTO_INCREMENT,
    inv_id INT NOT NULL,
    serv_id INT NOT NULL,
    PRIMARY KEY(hist_id)
);

ALTER TABLE history
    ADD CONSTRAINT fk_hist_inv FOREIGN KEY ( inv_id )
        REFERENCES invoice ( inv_id );
ALTER TABLE history
    ADD CONSTRAINT fk_hist_serv FOREIGN KEY ( serv_id )
        REFERENCES service ( serv_id );
ALTER TABLE history ADD INDEX index_inv_id (inv_id);
ALTER TABLE history ADD INDEX index_serv_id (serv_id);

CREATE TABLE hist_cust(
    hist_id INT NOT NULL,
    cust_id INT NOT NULL,
    cust_type VARCHAR(1) NOT NULL
);
ALTER TABLE hist_cust
    ADD CONSTRAINT fk_hist_cust_hist FOREIGN KEY ( hist_id )
        REFERENCES history ( hist_id );
ALTER TABLE hist_cust
    ADD CONSTRAINT fk_hist_cust_cust FOREIGN KEY ( cust_id,cust_type )
        REFERENCES customer ( cust_id,cust_type );

```

Part 2: Brief details of software, language, and database used

We use the Oracle Data Modeler to draw the logical model and the relation model. Oracle Data Modeler is a new, graphical data modeling tool that facilitates and enhances communication between data architects, database administrators, application developers and users, and simplifies the data modeling development process itself. We have been using this software at the beginning of this semester so we are really familiar with the process. Then we generated the DDL code,

For the database part, we choose to use MySQL database, because it is a relational database of backend and it is recommended by professor. And also, it is a free software. For the frameworks, we use PHP to connect our backend design. PHP is a scripting language used for server-side execution program interpretation. The PHP code is interpreted, executed, and converted into normal HTML page content on the server side and sent to the browser side. This mode allows us to use it to complete quite complex functions.

The PHP and MySQL design is perfect for web design because this kind of combination has good security, good portability and strong scalability. PHP is inherently anti-injection. PHP is open source software. Everyone can see all the source code of PHP. The code has been tested in the hands of many engineers. At the same time, the way it is compiled with Apache also allows it to have flexible security settings. PHP has Recognized safety performance.

Part3: List of tables, and total number of records of each table

```

35
36 • select count(*) from address;
37
100% 1:37 | 1 error found

Result Grid Filter Rows: Search

count(*)
▶ 20

42 • select count(*) from corporate_customer;
43
100% 1:43 | 1 error found

Result Grid Filter Rows: Search Export: CSV

count(*)
▶ 10

46 • select count(*) from coupon;
47
100% 1:47 | 1 error found

Result Grid Filter Rows: Search

count(*)
▶ 11

48 • select count(*) from customer;
49
100% 1:49 | 1 error found

Result Grid Filter Rows: Search

count(*)
▶ 20

52 • select count(*) from history;
53
100% 1:53 | 1 error found

Result Grid Filter Rows: Search

count(*)
▶ 1

54 • select count(*) from individual_customer;
55
100% 1:55 | 1 error found

Result Grid Filter Rows: Search Export: CSV

count(*)
▶ 10

```

```

38 • select count(*) from admins;
39
100% 1:39 | 1 error found

Result Grid Filter Rows: Search

count(*)
▶ 1

44 • select count(*) from corporation;
45
100% 1:45 | 1 error found

Result Grid Filter Rows: Search

count(*)
▶ 11

40 • select count(*) from class;
41
100% 1:41 | 1 error found

Result Grid Filter Rows: Search

count(*)
▶ 15

50 • select count(*) from hist_cust;
51
100% 1:51 | 1 error found

Result Grid Filter Rows: Search

count(*)
▶ 1

56 • select count(*) from invoice;
57
100% 1:57 | 1 error found

Result Grid Filter Rows: Search

count(*)
▶ 21

58 • select count(*) from office_location;
59
100% 1:59 | 1 error found

Result Grid Filter Rows: Search Export: CSV

count(*)
▶ 10

```

```
60 • select count(*) from payment;
61
100% 1:61 1 error found
Result Grid Filter Rows: Search
count(*)
▶ 1
```



```
64 • select count(*) from users;
65
100% 1:65 1 error found
Result Grid Filter Rows: Search
count(*)
▶ 6
```

```
62 • select count(*) from service;
63
100% 1:63 1 error found
Result Grid Filter Rows: Search
count(*)
▶ 11
66 • select count(*) from vehicle;
67
100% 1:67 1 error found
Result Grid Filter Rows: Search
count(*)
▶ 15
```

Part4: Screenshots of some sessions, pages, menus of your Web Application

The screenshot shows a file browser interface on the left and a code editor on the right.

Project Structure:

- Project
 - ProjHttp
 - index.php
 - phpinfo.php
 - private
 - database.php
 - db_credentials.php
 - functions.php
 - initialize.php
 - shared
 - public
 - admins
 - delete_admin.php
 - delete.php
 - edit_admin.php
 - edit.php
 - index.php
 - new_admin.php
 - new.php
 - show_admin.php
 - show.php
 - admins_login.php
 - customer_register.php
 - images
 - index.php
 - login.php
 - logout.php
 - pages
 - history.php
 - index.php
 - invoice.php
 - pay.php
 - payment.php
 - service.php
 - test.php
 - vehicle_query.php
- register.php
- stylesheets
 - staff.css

```

1 <?php require_once('../private/initialize.php'); ?>
2
3 <?php
4 require_login();
5 if(is_post_request()){
6     $inv_date = date('Y-m-d');
7     $inv_amount = $_POST['amount'];
8     $inv_id = create_invoice($inv_date,$inv_amount);
9     $service = [];
10    $service['pu_date'] = $_POST['pu_date'];
11    // $service['pu_date'] = strtotime($service['pu_date']);
12    $service['do_date'] = $_POST['do_date'];
13    // $service['do_date'] = strtotime($service['do_date']);
14    $service['odo_start'] = $_POST['odo_start'];
15    $service['odo_end'] = $_POST['odo_end'];
16    $service['dly_lim'] = $_POST['dly_lim'];
17    $service['veh_id'] = $_POST['veh_id'];
18    $service['cust_id'] = $_POST['cust_id'];
19    $service['cust_type'] = $_POST['cust_type'];
20    $service['inv_id'] = $inv_id;
21    $service['pu_loc_id'] = $_POST['pu_loc_id'];
22    $service['do_loc_id'] = $_POST['do_loc_id'];
23    $service['serv_id'] = create_service($service);
24    /*if(isset($service['serv_id'])) {
25        echo $Service['serv_id']."<br />";*/
26
27    $history = create_history($inv_id,$Service['serv_id']);
28    /*if(isset($history)) {
29        echo $history; */
30    $hist_cust = create_hist_cust($_SESSION['user_id'],$_SESSION['cust_type'],$history);
31 }
32 else{
33     $inv_date = date('Y-m-d');
34     if(isset($_GET['inv_id'])){
35         $inv_id = $_GET['inv_id'];
36         $result = find_by_inv_id($inv_id);
37         if(!$result)
38             redirect_to(url_for('/pages/index.php'));
39         else{
40             $res = $result->fetch_assoc();
41             $inv_date = $res['inv_date'];
42             $inv_amount = $res['amount'];
43         }
44     }
45     else
46         redirect_to(url_for('/pages/index.php'));
47 }
48
49 ?>
50
51 <?php $page_title = 'Show Invoice'; ?>
52 <?php include(SHARED_PATH . '/proj_header.php'); ?>
53
54 <div id="content">
55
56     <a class="back-link" href="php echo url_for('/pages/index.php'); ?&gt;"&lquo; Back to List</a>
57
58     <div class="invoice show">
59
60         <h1>Invoice: </h1>
61
62         <div class="attributes">
63             <dl>
64                 <dt>Invoice ID</dt>
65                 <dd><?php echo h($inv_id); ?></dd>
66             </dl>
67             <dl>
68                 <dt>Invoice Created Date</dt>
69                 <dd><?php echo h($inv_date); ?></dd>
70             </dl>
71             <dl>
72                 <dt>Amount</dt>
73                 <dd><?php echo h($inv_amount); ?></dd>
74             </dl>
75             <dl>
76                 <dt>Left Amount</dt>
77                 <dd><?php echo h($inv_amount); ?></dd>
78             </dl>
79         </div>
80
81     </div>
82
83 </div>
84 <?php require (SHARED_PATH . '/proj_footer.php');?>
85
86     <?php require_once ('../private/initialize.php');?>
87     <?php require_login();?>
88
89     /*$option = 1;
90     if(isset($_SESSION['admin_id'])){
91         $id = $_SESSION['admin_id'];
92         $option = 0;
93     }
94     else {
95         $id = $_SESSION['user_id'];
96     }*/
97
98     <?php $page_title = 'Public Page'; ?>
99     <?php require (SHARED_PATH . '/proj_header.php');?>
100    <div id="content">
101        <!--<?php echo $_SERVER['SCRIPT_NAME'];?-->
102        <!-- user login register logout admins-->
103        <div style="text-align: center;">
104            <h2>
105                <a class="action" href="php echo url_for('/pages/index.php'); ?&gt;">Order Now</a>
106            </h2>
107            <a href="php echo url_for('/index.php'); ?&gt;">
108                Home</a> <a href="#">Logout</a> <a href="#">Admins Page</a> |          |                      |                      |                        |  |
| <b>Admins</b>                                                           |          |                      |                      |                        |  |
| <a href="#">Create New Admin</a>                                        |          |                      |                      |                        |  |
| ID                                                                      | Username | <a href="#">View</a> | <a href="#">Edit</a> | <a href="#">Delete</a> |  |
| 10                                                                      | root     | <a href="#">View</a> | <a href="#">Edit</a> | <a href="#">Delete</a> |  |

## Users

[Create New User](#)

| ID | Username    | <a href="#">View</a> | <a href="#">Edit</a> | <a href="#">Delete</a> |  |
|----|-------------|----------------------|----------------------|------------------------|--|
| 1  | XinyuZhao   | <a href="#">View</a> | <a href="#">Edit</a> | <a href="#">Delete</a> |  |
| 3  | TieLiuZhou  | <a href="#">View</a> | <a href="#">Edit</a> | <a href="#">Delete</a> |  |
| 4  | Qwertyuiop  | <a href="#">View</a> | <a href="#">Edit</a> | <a href="#">Delete</a> |  |
| 5  | Qgewgwwerqt | <a href="#">View</a> | <a href="#">Edit</a> | <a href="#">Delete</a> |  |
| 6  | Poiuytrewq  | <a href="#">View</a> | <a href="#">Edit</a> | <a href="#">Delete</a> |  |
| 7  | Lkjhgfdsa   | <a href="#">View</a> | <a href="#">Edit</a> | <a href="#">Delete</a> |  |

Meanwhile, administrator can create, view, edit or delete a user/administrator.



[« Back to List](#)

## Create Admin

|                     |                          |
|---------------------|--------------------------|
| Username            | <input type="text"/>     |
| Password            | <input type="password"/> |
| Confirm<br>Password | <input type="password"/> |

Passwords should be at least 12 characters and include at least one uppercase letter, lowercase letter, number, and symbol.



[« Back to List](#)

## User: XinyuZhao

[Edit Delete](#)

|          |           |
|----------|-----------|
| Username | XinyuZhao |
|----------|-----------|



[« Back to List](#)

## Edit User

|                     |                                        |
|---------------------|----------------------------------------|
| Username            | <input type="text" value="XinyuZhao"/> |
| Password            | <input type="password"/>               |
| Confirm<br>Password | <input type="password"/>               |

Passwords should be at least 12 characters and include at least one uppercase letter, lowercase letter, number, and symbol.



[« Back to List](#)

## Delete User

Are you sure you want to delete this user?

XinyuZhao

If you are a customer of the WOW car rental system, you can login as user.

**WOW Car Rental System**

[Home](#) [Logout](#)

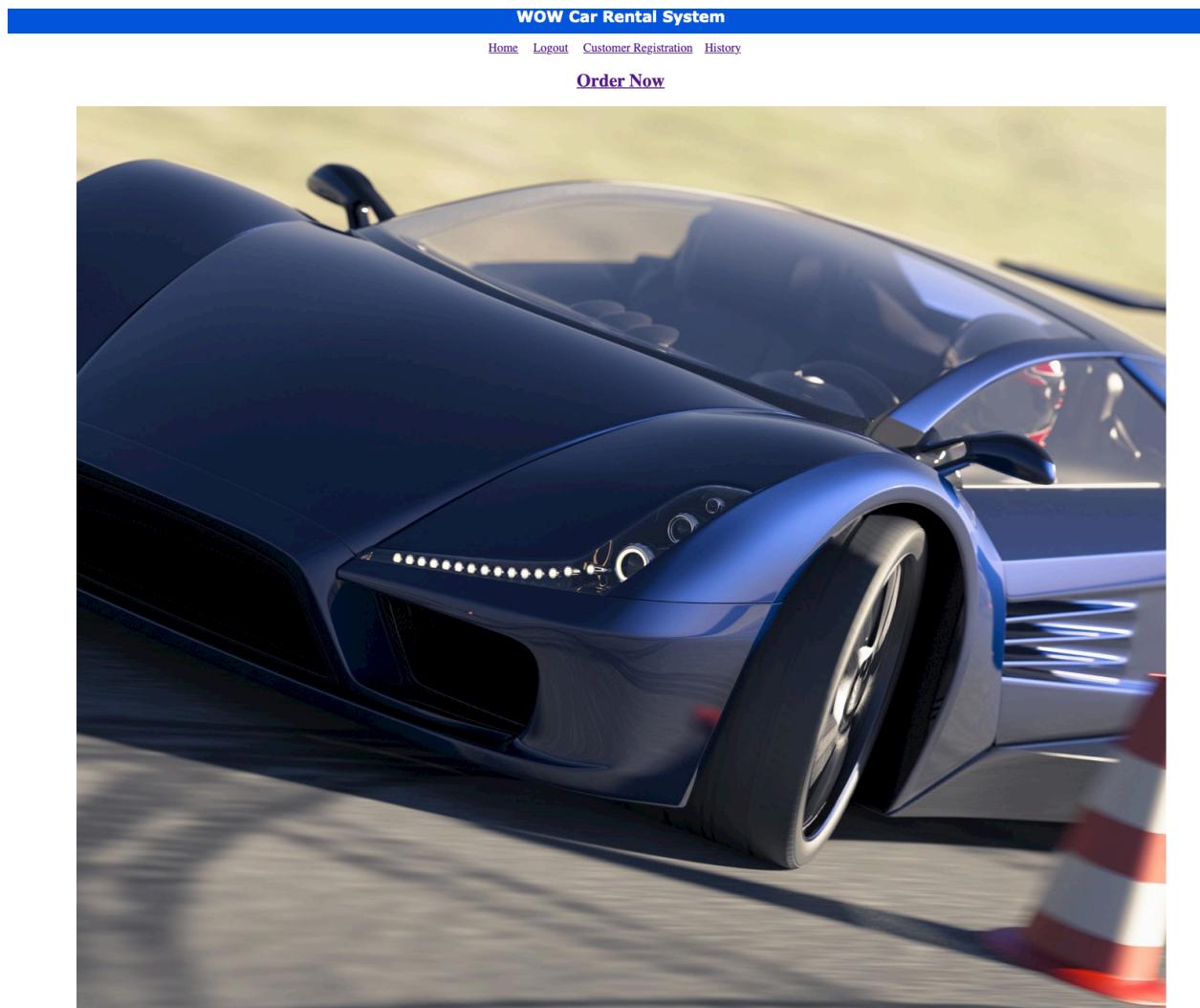
## Log in

Username:

Password:

Individual or Corporation?(I/C):

[Log in](#) [Register](#) [Admins](#)



Before you can order a car, you need to have a customer registration.

**WOW Car Rental System**

[Home](#) [Logout](#) [Customer Registration](#) [History](#)

[« Back to List](#)

### Fill in your information

|                               |                                            |
|-------------------------------|--------------------------------------------|
| individual or corporate?(I/C) | <input type="radio"/>                      |
| First Name                    | Damian                                     |
| Last Name                     | Ocaniel                                    |
| Phone Number                  | 2571226954 <input type="button" value=""/> |
| Email                         | damiano@gmail.com                          |
| Street                        | 27 Laurence St                             |
| City                          | Berkeley                                   |
| State                         | CA                                         |
| zipcode                       | 79810 <input type="button" value=""/>      |

If you are individual, please fill in the following:

|                         |                                              |
|-------------------------|----------------------------------------------|
| Driver License Number   | 686492146 <input type="button" value=""/>    |
| Insurance Company       | SunRise Inc. <input type="button" value=""/> |
| Insurance Policy Number | 88653 <input type="button" value=""/>        |
| Coupon ID(if you have)  | <input type="button" value=""/>              |

If you are corporate customer, please fill in the following:

|              |                                 |
|--------------|---------------------------------|
| Employ ID    | <input type="button" value=""/> |
| Corporate ID | <input type="button" value=""/> |

[Register](#)

If you are registered as an individual customer, you will need to fill your driver license number, your insurance company, your insurance policy number and your coupon ID if you have any.

If you are registered as a corporation customer, you will need to input your employ ID and your corporate ID.

Then you can input your need to the system and see if there are any available cars.

**WOW Car Rental System**

[Home](#) [Logout](#) [Customer Registration](#) [History](#)

[« Back to List](#)

### Enter your requirement

|                               |                                                                                                            |
|-------------------------------|------------------------------------------------------------------------------------------------------------|
| Pick Up City                  | Tullahoma <input type="button" value=""/>                                                                  |
| Drop Off City                 | Tullahoma <input type="button" value=""/>                                                                  |
| Pick Up State                 | TN <input type="button" value=""/>                                                                         |
| Drop Off State                | TN <input type="button" value=""/>                                                                         |
| Pick Up Date(Year/Month/Day)  | 2020 <input type="button" value=""/> 12 <input type="button" value=""/> 21 <input type="button" value=""/> |
| Drop Off Date(Year/Month/Day) | 2020 <input type="button" value=""/> 12 <input type="button" value=""/> 21 <input type="button" value=""/> |
| Class of Car                  | COUPE <input type="button" value=""/>                                                                      |
| Estimate Distance /miles      | 35 <input type="button" value=""/>                                                                         |

© 2020 CS-GY 6083 B

Based on your estimate distance, you can have an estimate fee of this rental service.

**WOW Car Rental System**

[Home](#) [Logout](#) [Customer Registration](#) [History](#)

[« Back to Requirement](#)

### Available Vehicle

| Vehicle ID | Made | Model    | Year | Vehicle Identification Number | License Plate Number | Location                                 | Total Price | Order                                                                                                                  |
|------------|------|----------|------|-------------------------------|----------------------|------------------------------------------|-------------|------------------------------------------------------------------------------------------------------------------------|
| 4001       | BMW  | 1 Series | 2020 | WP0CA29912S622816             | 824-NHT              | 30 Winchester Ave., Tullahoma, TN, 37388 | 140         | Drop off location: <input type="text" value="id:1 street:30 Winchester Ave."/><br><input type="button" value="Order"/> |

© 2020 CS-GY 6083 B

Click order, then your order will be made with a invoice showing up.

**WOW Car Rental System**

[Home](#) [Logout](#) [Customer Registration](#) [History](#)

[« Back to List](#)

**Invoice:**

|                      |            |
|----------------------|------------|
| Invoice ID           | 22         |
| Invoice Created Date | 2020-12-20 |
| Amount               | 140        |
| Left Amount          | 140        |

© 2020 CS-GY 6083 B

When you have returned your car, you can find your order in the History section.

**WOW Car Rental System**

[Home](#) [Logout](#) [Customer Registration](#) [History](#)

[« Back to Query](#)

**History Page**

| History ID | Invoice ID | Service ID | Payment              | Pay                 |
|------------|------------|------------|----------------------|---------------------|
| 2          | 22         | 789000012  | <a href="#">View</a> | <a href="#">Pay</a> |

© 2020 CS-GY 6083 B

You can check your invoice and your service info.

**WOW Car Rental System**

[Home](#) [Logout](#) [Customer Registration](#) [History](#)

[« Back to List](#)

**Invoice:**

|                      |                     |
|----------------------|---------------------|
| Invoice ID           | 22                  |
| Invoice Created Date | 2020-12-20 00:00:00 |
| Amount               | 140                 |
| Left Amount          | 140                 |

© 2020 CS-GY 6083 B

**WOW Car Rental System**

[Home](#) [Logout](#) [Customer Registration](#) [History](#)

[« Back to History](#)

**Service ID: 789000012**

|                      |                     |
|----------------------|---------------------|
| Pick Up Date         | 2020-12-21 00:00:00 |
| Drop Off Date        | 2020-12-21 00:00:00 |
| The Start Odometer   | 42368               |
| The End Odometer     | 42403               |
| Daily Limit          | 50                  |
| Vehicle ID           | 4001                |
| Customer ID          | 1                   |
| Customer Type        | I                   |
| Invoice ID           | 22                  |
| Pick Up Location ID  | 1                   |
| Drop Off Location ID | 1                   |

© 2020 CS-GY 6083 B

Click on view under Payment, you can see that there are still 140\$ you have not paid yet.

[Home](#) [Logout](#) [Customer Registration](#) [History](#)

### Payment History

[« Back to History](#)

| Payment ID | Payment Time | Payment Type | Payment Amount | Card Number |
|------------|--------------|--------------|----------------|-------------|
|            |              |              |                |             |

Left Amount: 140 \$ [Pay](#)

© 2020 CS-GY 6083 B

Click Pay, then you need to input your card info.

**WOW Car Rental System**

[Home](#) [Logout](#) [Customer Registration](#) [History](#)

### Payment

[« Back to History](#)

|                                         |                                                                 |
|-----------------------------------------|-----------------------------------------------------------------|
| Payment Date                            | 2020-12-20                                                      |
| Left To Pay                             | 140 \$                                                          |
| Invoice ID                              | 22                                                              |
| Payment Type(Credit = 'C', Debit = 'D') | <input type="text" value="C"/>                                  |
| Card Number                             | 62621894613164981 <input type="button" value=""/>               |
| Payment Amount                          | <input type="text" value="80"/> <input type="button" value=""/> |

[Pay](#)

© 2020 CS-GY 6083 B

You can pay the bill separately. You can have different payment to one invoice.

## WOW Car Rental System

[Home](#) [Logout](#) [Customer Registration](#) [History](#)

### Payment History

[« Back to History](#)

| Payment ID | Payment Time        | Payment Type | Payment Amount | Card Number       |
|------------|---------------------|--------------|----------------|-------------------|
| 2          | 2020-12-20 17:00:02 | C            | 80             | 62621894613184981 |

Left Amount: 60 \$

[Pay](#)

© 2020 CS-GY 6083 B

After paid 80\$, you can pay the remaining amount by using a different payment method.

## WOW Car Rental System

[Home](#) [Logout](#) [Customer Registration](#) [History](#)

[« Back to History](#)

### Payment

|                                         |                                             |
|-----------------------------------------|---------------------------------------------|
| Payment Date                            | 2020-12-20                                  |
| Left To Pay                             | 60 \$                                       |
| Invoice ID                              | 22                                          |
| Payment Type(Credit = 'C', Debit = 'D') | <input type="text" value="D"/>              |
| Card Number                             | <input type="text" value="84946131489494"/> |
| Payment Amount                          | <input type="text" value="60"/>             |

[Pay](#)

© 2020 CS-GY 6083 B

## WOW Car Rental System

[Home](#) [Logout](#) [Customer Registration](#) [History](#)

### Payment History

[« Back to History](#)

| Payment ID | Payment Time        | Payment Type | Payment Amount | Card Number       |
|------------|---------------------|--------------|----------------|-------------------|
| 2          | 2020-12-20 17:00:02 | C            | 80             | 62621894613184981 |
| 3          | 2020-12-20 17:01:53 | D            | 60             | 84946131489494    |

Left Amount: 0 \$      [Paid](#)

And you have paid your rental service. Once you have paid your bill, then the status of this rental service history will change from Unpaid to Paid.

## WOW Car Rental System

[Home](#) [Logout](#) [Customer Registration](#) [History](#)

### History Page

[« Back to Query](#)

| History ID | Invoice ID         | Service ID                | Payment              | Pay                  |
|------------|--------------------|---------------------------|----------------------|----------------------|
| 2          | <a href="#">22</a> | <a href="#">789000012</a> | <a href="#">View</a> | <a href="#">Paid</a> |

© 2020 CS-GY 6083 B

## Part 5: Details of security features that We have implemented on Web Application development

The customer and manager have different authorization to use the App. Manager can use more functions than customer. The sensitive data, like passwords, need to be encrypted and then stored in the database. We used password\_hash from PHP to implement this encryption.

A WOW employee can login as a administrator

Here shows an administrator username root and with an encrypted password “root”.

| admin_id | username | hashed_password                                                 |
|----------|----------|-----------------------------------------------------------------|
| 10       | root     | \$2y\$10\$IIQnp38FT8jEISE.MwRNtubLJd3sbC80SUwzNRw305YE3hiana8fy |

A customer can register as a user.

The screenshot shows the 'Register' page of the 'WOW Car Rental System'. At the top, there is a blue header bar with the system name. Below it, a navigation bar includes links for 'Home' and 'Logout', and a link to '« Back to List'. The main content area is titled 'Register' in bold. It contains three input fields labeled 'Username:', 'Password:', and 'Confirm Password:' respectively. Below these fields is a note stating: 'Passwords should be at least 12 characters and include at least one uppercase letter, lowercase letter, number, and symbol.' At the bottom of the form is a single button labeled 'Register'.

© 2020 CS-GY 6083 B

Passwords should be at least 12 characters and include at least one uppercase letter, lowercase letter, number, and symbol. The more complicated the password is, the more secure it will be.

## **Part 6: Journey of Project and Summary**

We did this project as a team. There are two great challenges. First is that we are in different time zone, with 13 hours difference, every time we communicate to each other is a great challenge. Second is that none of us has designed a web page before and we have never learned the PHP. So we have to do our best to learn this language from the very beginning and make it into use.

We watched the video series on LinkedIn that professor recommended. Then we started to set up the environment, and tried to start coding for the web page little by little. There are many bugs at the first, but we patiently go through the logic of the code and checked every line of code to debug. Finally, we got a breakthrough. Then we did not stop and added some extra features into our design.

From this project, we have learned much more than just knowledge of the database system. We learned how to teamwork, how to quickly learn a new programming language, how to manage time to have a high efficient communication, and more importantly, we have learned to design a web page based on what we have learned.

After doing about one week, our WOW car rental system are finished. We are really proud of this web application. There are no errors in the final web page, every function works great.

We increased the index for the key value frequently used in the query, mainly the foreign key index to facilitate the search. As for the security part, we applied two methods, prepare statement and escape string, to avoid SQL injection attacks. The htmlspecialchars is used to avoid xss attacks, and urlencode is used to avoid url attacks.

## Part 7: Business analysis with 4 SQLs

1. table joins: Query the location of all BMW vehicles .

```
SELECT o.loc_street, o.loc_city, o.loc_state, o.loc_zipcode
FROM office_location o, vehicle v
WHERE v.make = 'BMW' AND v.loc_id = o.loc_id;
```

```
79
80 • SELECT o.loc_street, o.loc_city, o.loc_state, o.loc_zipcode
81 FROM office_location o, vehicle v
82 WHERE v.make = 'BMW' AND v.loc_id = o.loc_id;

100% 1:80 | 1 error found
Result Grid Filter Rows: Search Export:

loc_street	loc_city	loc_state	loc_zipcode
30 Winchester Ave.	Tullahoma	TN	37388
853 Race Street	Cary	NC	27511
853 Race Street	Cary	NC	27511
334 E. Argyle Ave.	Doylestown	PA	18901


```

2. Subquery: List all the vehicles that was rented on 2020-10-15 and returned after this date with mileage farther than 120 miles.

```
SELECT se.veh_id ,(s.odo_end-s.odo_start) "Total Odometers"
FROM service se
INNER JOIN (SELECT * FROM service WHERE (odo_end-odo_start) > 120) s
WHERE se.pu_date = '2020-10-15'
AND se.do_date >= '2020-10-16'
ORDER BY veh_id;
```

```

87 • SELECT se.veh_id , (s.odo_end-s.odo_start) "Total Odometers"
88 FROM service se
89 INNER JOIN (SELECT * FROM service WHERE (odo_end-odo_start) > 120) s
90 WHERE se.pu_date = '2020-10-15'
91 AND se.do_date >= '2020-10-16'
92 ORDER BY veh_id;

```

17:92 | 1 error found

**Result Grid** Filter Rows: Search Export:

| veh_id | Total Odometers |
|--------|-----------------|
| 4003   | 125             |

3. Top-N queries: Find three people who drives farthest during the rental process no matter if she/he is individual customer or corporate customer.

```

SELECT cus.first_name, cus.last_name,
(s.odo_end-s.odo_start) "Total Odometers"

FROM customer cus INNER JOIN service s

ON cus.cust_id = s.cust_id

ORDER BY (s.odo_end-s.odo_start)

DESC LIMIT 3;

```

14:99 | 1 error found

**Result Grid** Filter Rows: Search

| first_name | last_name | Total Odometers |
|------------|-----------|-----------------|
| Ho         | Alston    | 125             |
| Tahmina    | Barajas   | 112             |
| Ralph      | Avila     | 103             |

4. Business Information: average mileage of each car rent.

```
SELECT ROUND (SUM(odo_end-odo_start)/count(cust_id),0)
AS "Average Mileage of Each Car Rent"
FROM service;
```

The screenshot shows a database query editor interface. The query is displayed in a code editor with line numbers 103 through 106. Lines 103, 104, and 105 are highlighted in blue, while line 106 is grayed out. The query itself is:  
103 • **SELECT ROUND (SUM(odo\_end-odo\_start)/count(cust\_id),0)**  
104 **AS "Average Mileage of Each Car Rent"**  
105 **FROM service;**  
106  
A status bar at the bottom indicates "1 error found". Below the editor is a toolbar with buttons for "Result Grid", "Filter Rows", "Search", and "Export". The result grid shows one row of data:  
Average Mileage of Each Car Re...  
▶ 57

## Part 8: Extra Features

1. Introduce the indexes, to increase the index for the key value frequently used in the query, mainly the foreign key index to facilitate the search, including function, procedure, trigger, etc.

```
drop trigger if exists arc_fkarc_individual_customer_update;
delimiter $
CREATE TRIGGER arc_fkarc_individual_customer_update BEFORE UPDATE
 ON individual_customer
#INSERT OR UPDATE OF cust_id, cust_type ON individual_customer
 FOR EACH ROW
BEGIN
 DECLARE d VARCHAR(1);
 SELECT
 a.cust_type
 INTO d
 FROM
 customer a
 WHERE
 a.cust_id = new.cust_id AND a.cust_type = new.cust_type;
 IF (d IS NULL OR d <> 'I') THEN
 SIGNAL SQLSTATE '42S22' SET MESSAGE_TEXT = 'FK in Table INDIVIDUAL_CUSTOMER violates
constraint on Table CUSTOMER CUST_TYPE doesn''t have value ''I'''';
 END IF;
END$
delimiter ;

drop trigger if exists arc_fkarc_individual_customer_insert;
delimiter $
CREATE TRIGGER arc_fkarc_individual_customer_insert BEFORE INSERT
 ON individual_customer
 FOR EACH ROW
BEGIN
 DECLARE d VARCHAR(1);
 SELECT
 a.cust_type
 INTO d
 FROM
 customer a
 WHERE
 a.cust_id = new.cust_id AND a.cust_type = new.cust_type;
 IF (d IS NULL OR d <> 'I') THEN
 SIGNAL SQLSTATE '42S22' SET MESSAGE_TEXT = 'FK in Table INDIVIDUAL_CUSTOMER violates
constraint on Table CUSTOMER CUST_TYPE doesn''t have value ''I'''';
 END IF;
END$
delimiter ;

drop trigger if exists arc_fkarc_corporate_customer_insert;
delimiter $
CREATE TRIGGER arc_fkarc_corporate_customer_insert BEFORE INSERT
 ON corporate_customer
#INSERT OR UPDATE OF cust_id, cust_type ON individual_customer
 FOR EACH ROW
BEGIN
 DECLARE d VARCHAR(1);
 SELECT
 a.cust_type
 INTO d
 FROM
 customer a
 WHERE
 a.cust_id = new.cust_id AND a.cust_type = new.cust_type;
 IF (d IS NULL OR d <> 'C') THEN
 SIGNAL SQLSTATE '42S22' SET MESSAGE_TEXT = 'FK in Table CORPORATE_CUSTOMER violates
constraint on Table CUSTOMER CUST_TYPE doesn''t have value ''C'''';
 END IF;
END$
delimiter :
```

## 2. Concurrency-control :

Concurrency control schemes tradeoff between the amount of concurrency they allow and the amount of overhead that they incur. The goal of concurrency control is to develop concurrency control protocols that will assure serializability, and thus ensure data consistency.

In our code, we use `$db->begin_transaction()` to start transaction and finally commit or rollback.

```
function user_register($user,$option=1){
 global $db;

 $errors = validate_user($user);
 if(!empty($errors)){
 return $errors;
 }

 //Encrypted password
 $hashed_password = password_hash($user['password'],PASSWORD_BCRYPT);

 $db->begin_transaction();
 try{
 if($option==0){
 $sql = "INSERT INTO admins";
 }
 else{
 $sql = "INSERT INTO users";
 }

 $sql .= "(username, hashed_password, is_register) ";
 $sql .= "VALUES (";
 $sql .= "'" . $db->real_escape_string($user['username']) . "',";
 $sql .= "'" . $db->real_escape_string($hashed_password) . "',";
 $sql .= "'" . $db->real_escape_string(false) . "'')";
 $result = $db->query($sql);
 $db->commit();
 }
}
```

```
function create_invoice($date,$amount){
 global $db;

 $db->begin_transaction();
 try{
 $sql = "INSERT INTO invoice(inv_date,amount) VALUES(";
 $sql .= "".$db->real_escape_string($date)."','".$$db->real_escape_string($amount)."'");

 $result_set = $db->query($sql);
 $new_id = $db->insert_id;
 $db->commit();
 return $new_id;
 }
 catch (mysqli_sql_exception $exception){
 $db->rollback();
 throw $exception;
 }
 return 0;
}
```

3. The sensitive data, like passwords, need to be encrypted and then stored in the database.

We used password\_hash from PHP to implement this encryption.

107 • `select * from users`

20:107 | 1 error found

Result Grid Filter Rows: Search Edit:

|   | user_id | username    | hashed_password                                 | is_register |
|---|---------|-------------|-------------------------------------------------|-------------|
| ▶ | 1       | XinyuZhao   | \$2y\$10\$IIQnp38FT8jEISE.MwRNtubLJd3sbC80...   | 1           |
|   | 3       | TieliuZhou  | \$2y\$10\$IIQnp38FT8jEISE.MwRNtubLJd3sbC80...   | 1           |
|   | 4       | Qwertyuiop  | \$2y\$10\$1otgzqdoQam/xgfrwN8jl.YlblbdDyCTRr... | 0           |
|   | 5       | Qgewgwwerqt | \$2y\$10\$IIQnp38FT8jEISE.MwRNtubLJd3sbC80...   | 1           |
|   | 6       | Poiuytrewq  | \$2y\$10\$IIQnp38FT8jEISE.MwRNtubLJd3sbC80...   | 1           |
|   | NULL    | NULL        | NULL                                            | NULL        |

74 • `select * from admins`

100% 21:74 | 1 error found

Result Grid Filter Rows: Search Edit: Export/Imp

|   | admin_id | username | hashed_password                                                 |
|---|----------|----------|-----------------------------------------------------------------|
| ▶ | 10       | root     | \$2y\$10\$IIQnp38FT8jEISE.MwRNtubLJd3sbC80SUwzNRw305YE3hiana8fy |
|   | NULL     | NULL     | NULL                                                            |

4. When customers finished their trip, they can see the invoice for this trip and pay for it with different payment methods. They can view historical order records and view various invoices and payments, and if an invoice is paid in full, it cannot be paid again, the pay button will not be displayed and the word “PAID” will be displayed.

**WOW Car Rental System**

[Home](#) [Logout](#) [Customer Registration](#) [History](#)

[« Back to List](#)

**Invoice:**

|                      |                     |
|----------------------|---------------------|
| Invoice ID           | 22                  |
| Invoice Created Date | 2020-12-20 00:00:00 |
| Amount               | 140                 |
| Left Amount          | 140                 |

**WOW Car Rental System**

[Home](#) [Logout](#) [Customer Registration](#) [History](#)

[« Back to History](#)

**Payment**

|                |                            |
|----------------|----------------------------|
| Payment Date   | 2020-12-20                 |
| Left To Pay    | 140 \$                     |
| Invoice ID     | 22                         |
| Payment Type   | Credit = 'C', Debit = 'D') |
| Card Number    | 62621894613184981          |
| Payment Amount | 80 \$                      |

[Pay](#)

© 2020 CS-GY 6083 B  
**WOW Car Rental System**

[Home](#) [Logout](#) [Customer Registration](#) [History](#)

**Payment History**

[« Back to History](#)

| Payment ID | Payment Time        | Payment Type | Payment Amount | Card Number       |
|------------|---------------------|--------------|----------------|-------------------|
| 2          | 2020-12-20 17:00:02 | C            | 80             | 62621894613184981 |

Left Amount: 60 \$ [Pay](#)

© 2020 CS-GY 6083 B

## WOW Car Rental System

[Home](#) [Logout](#) [Customer Registration](#) [History](#)

### Payment History

[« Back to History](#)

| Payment ID | Payment Time        | Payment Type | Payment Amount | Card Number       |
|------------|---------------------|--------------|----------------|-------------------|
| 2          | 2020-12-20 17:00:02 | C            | 80             | 62621894613184981 |
| 3          | 2020-12-20 17:01:53 | D            | 60             | 84946131489494    |

Left Amount: 0 \$ Paid

## WOW Car Rental System

[Home](#) [Logout](#) [Customer Registration](#) [History](#)

### History Page

[« Back to Query](#)

| History ID | Invoice ID | Service ID | Payment              | Pay                  |
|------------|------------|------------|----------------------|----------------------|
| 2          | 22         | 789000012  | <a href="#">View</a> | <a href="#">Paid</a> |

© 2020 CS-GY 6083 B

5. We applied two methods, prepare statement and escape string, to avoid SQL injection attacks. The htmlspecialchars is used to avoid xss attacks, and urlencode is used to avoid url attacks.

```
/* For Security */
function u($string=""){ // urlencode function for query
 //?后 转换特殊字符、保留字符 空格 = +
 return urlencode($string);
}

function raw_u($string=""){ // rawurlencode function for path
 //?"前部分语句 空格 = %20
 return rawurlencode($string);
}

function h($string=""){
 //避免URL注入攻击，夹杂html指令或者javascript命令
 //将"<", ">"等转换成别的字符"<", ">"
 return htmlspecialchars($string);
}
```

6. We implemented the history function on the web page. Customers can see their rental history.

The screenshot shows the 'History Page' of the 'WOW Car Rental System'. At the top, there is a navigation bar with links for Home, Logout, Customer Registration, and History. Below the navigation bar, the title 'History Page' is displayed. A link '« Back to Query' is present. The main content area contains a table with the following data:

| History ID | Invoice ID | Service ID | Payment | Pay  |
|------------|------------|------------|---------|------|
| 2          | 22         | 789000012  | View    | Paid |

At the bottom of the page, there is a footer bar with the copyright notice '© 2020 CS-GY 6083 B'.