# Project1 Report
# CSCE-608

**QIFAN LI**

**127004551**

# Project Description

The database system I designed is called "Anime Information Management System". The aim of the system is to gather some most useful information about anime in one place, so that it will be very easy and convenient for people who love anime to get most information they want about their favorite anime.

As for the background of the application, I have great interest in anime and I watch anime every week when I am in my spare time. But every time I have interest in the information in anime, I have no choice but to search them through Google and it takes a lot of time to find the information I want. So I want to build this "Anime Information Management System" to manage information of the top anime and I think it is interesting to build something that I can use in my daily life. Also, this system can benefit the people whoever have interest in anime.

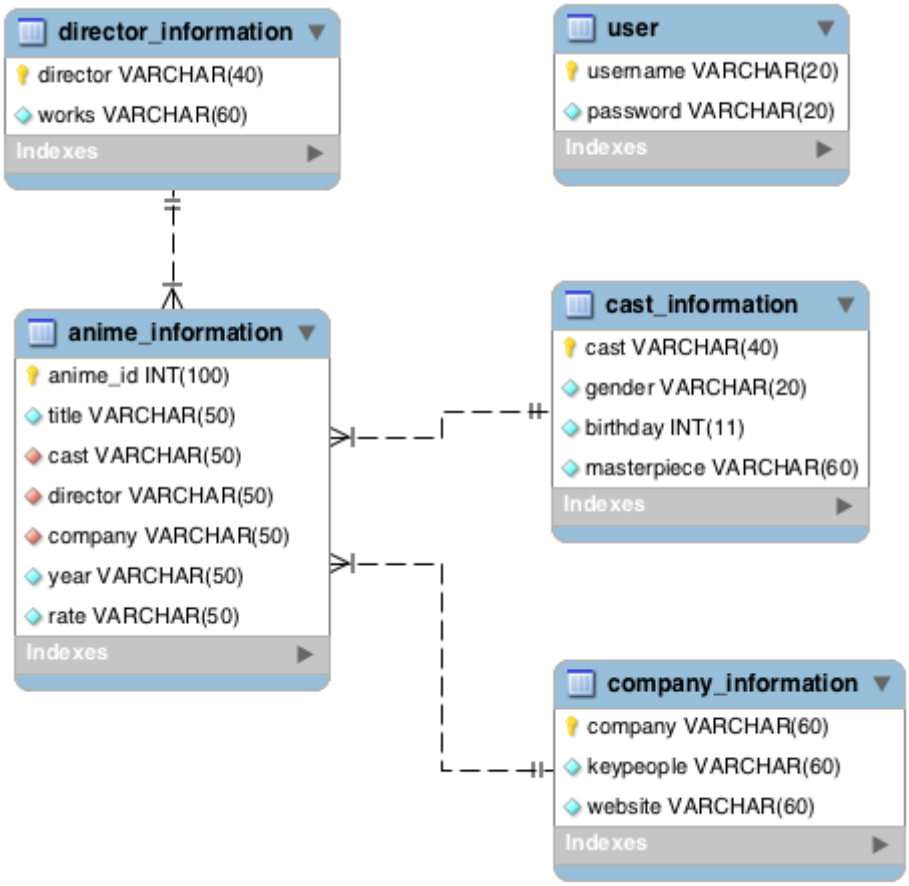
To fulfil the requirement of anime fans, the system is equipped with the functions including adding, deleting, updating and searching information of anime. As for the database, it has four tables: anime_information, cast_information, director_information, company_information and user. The main table anime_information contains the data of anime_id, cast, director, company, year and rate, which are necessary information that people are most interested in. When people are looking for some anime to watch, it is common for them to search for the casts they love, famous directors, and well-known anime producer companies. And there are three foreign keys in this table: cast, director and company. The cast attribute references the table cast_information. The director attribute references the table director_information. The company attribute references the table company_information. With the help of them, the user can not add or update information like cast, director or company that are nonexistent to the database. The cast_information contains data of cast, gender, birthday and masterpiece. The director_information contains data of director and works. The company_information contains data of company, key people and website. As for the user table, it stores username and password of the people who have signed up for this system. The username is primary key of the table, so you can not register with the username that is already in the database.

As for the functions of the system, at first, you need to sign up to get an account for this system and then you can get full access to the anime data. Duplicate usernames are not allowed and you can not leave your username or password blank while registering or logging in the system. After logging in the system, you will see the anime information home page. There are three icons at the right top of the page: Add, Search and Sign Out. You can press the "Add" icon to add information to the anime information list and press the "Search" Icon to search for the Anime ID to get the specific information you want. When you press the "Sign Out" icon, you will log out of the system. Then, you can see the main table in the middle of the home page that contains anime information and a manipulation icon at last column of the table. In this column, you can choose whether to update or delete this tuple. Remember that when you add or update tuple in anime information, you can only add or update the casts, directors or companies are included in the relevant tables. If you do not follow that rule, your adding or updating action will fail. In the cast, director and company columns, you can press the name of the attributes to get details of that column. For instance, if you press the cast icon, you will get detail information about cast, like gender, birthday and masterpiece. For the convenience of users, I add another search icon below the main table, so that there is no need for user to return top to search for anime information.

# Data Collection

All the data of anime is collected from the real world. I collect some most useful information about top anime in recent years from the website like My AnimeList, Wiki and so on. Download all the information from the network and use excels to put the information into four tables. After that, I convert the excel documents to csv documents and import all the data into the database using MySQL Workbench. Since it can be used in daily life, more data will be imported to that database in future.

# The Entity-Relationship diagram of database




# Table Normalization

After I finish the process of table normalization, it seems that the table structures do not change at all. I think the reason for it is that my database only contains four tables and three foreign keys, which makes the structure simple. Another reason for it is that I define the E-R diagram carefully, and identify all entities correctly, thus the relation schemas generated from diagram should not need much further normalization. If the database is built more complicated, the table normalization process may optimize the structure a lot.

# User Interface

I build the system user interface with HTML, CSS, PHP and Bootstrap. Since the goal is to design a user-friendly interface, I think we need to make the interface as simple as possible and there is no need for user to read manual before using the system. First, you need to sign up or sign in to the system if you already have an account. You are not allowed to leave the username or password blank. Then, you can view the whole anime information table in home page and you can use the icon on the top right of the page to add anime information to the system or search for the anime id. However, you are not allowed to add information that are nonexistent. (For this system, if you want to add information to the database successfully, the cast, director and company should already exist in the database) At the last column of each tuple, you can choose to update or delete the information from the database. The update function has the same restriction as the add function. And you can press the cast, director and company icons for details. After that, you can sign out through pressing the sign out icon on the top right of the page. In a nutshell, the anime information management system has four functions as select, update, insert and delete, (search, update, add, delete) and it is really easy for user to get started with without any knowledge about database.

# Project Source Code

Since there are many source code files, I will include part of the function code here and I attach all the source code to the submitting folder.

**index.php**

index.php

```php
<?PHP
    header("Content-Type: text/html; charset=utf8");

    $conn = new mysqli("127.0.0.1", "root", "liqifan6", "anime");//
    connect to database
    if(!$conn){
        echo 'can not connect: ' . mysql_error();
    }
    $uin = $_POST['username'];
    $pwd = $_POST['password'];
    $sql = "select password from user where username='{$uin}'";
    $result = $conn->query($sql);
    $n = $result->fetch_row();//这里的$result是一个对，存到变量$n里面。
    if($uin !="" && $pwd !="")
{
    if($n[0]==$pwd)
    {
        header("location:home.php");
    }
    else
    {
        echo "Wrong Username or Wrong Password.<br /><a href =
            'index.html'>Return</a>";
    }
}
    else
    {
    echo "Username or Password can not be null!<br /><a href =
        'index.html'>Return</a>";
    }
    $conn->close();//关闭数据库
?>
```

**home.php**

home.php

```
<div class="container-fluid bg-2 text-center">
  <h3 class="margin">Anime Information</h3>
  <?php
    #连接数据库
    $conn = new mysqli("127.0.0.1", "root", "liqifan6", "anime");

    if(!$conn){
        echo 'can not connect: ' . mysql_error();
    }//链接数据库

    //准备sql语句
    $sql = "select * from anime_information";

    //发送sql语句
    $obj = $conn->query($sql);

    echo "<center>";

    echo "<table border = 1 cellspacing = '0' cellpadding = '10'>";
    echo "<th>ID</th><th>Title</th><th><a href =
      'cast.php'>Cast</a></th><th><a href =
      'director.php'>Director</a></th><th><a href =
      'company.php'>Company</a></th><th>Year</th><th>Rate</th><th>
      Manipulation</th>";
    while($row = mysqli_fetch_assoc($obj)){
        echo "<tr>";
            echo '<td>'.$row['anime_id'].'</td>';
            echo '<td>'.$row['title'].'</td>';
            echo '<td>'.$row['cast'].'</td>';
            echo '<td>'.$row['director'].'</td>';
            echo '<td>'.$row['company'].'</td>';
            echo '<td>'.$row['year'].'</td>';
            echo '<td>'.$row['rate'].'</td>';
            echo '<td><a href = "delete.php?id='.$row['anime_id'].'
              ">Delete</a>/<a href = "update.php?id='.$row['anime_id'].'
              ">Update</a></td>';
        echo "</tr>";
    }

    echo "</table>";

    echo "<center>";

    //关闭连接
    $conn->close();
  ?>
```

**doregister.php**

doregister.php

```php
<?php

    $uin = $_POST['username'];
    $pwd = $_POST['password'];

    $conn = new mysqli("127.0.0.1", "root", "liqifan6", "anime");
    if(!$conn){
        echo 'can not connect: ' . mysql_error();
    }
    if ($uin == '' || $pwd == ''){
        echo "Username or Password can not be null!";
    }
    else{
    $sql = "insert into user(username,password) values('$uin','$pwd')";}

    $obj = $conn->query($sql);

    if($obj){
        echo "Register Success<a href = 'index.html'><br />Press to
            login</a>";
    }else{
        echo "Register Fail<a href = 'index.html'><br />Return</a>";
    }
    $conn->close();
?>
```

**doadd.php**

doadd.php

```php
<?php

    $id = $_GET['id'];
    $title = $_GET['title'];
    $cast = $_GET['cast'];
    $director = $_GET['director'];
    $company = $_GET['company'];
    $year = $_GET['year'];
    $rate = $_GET['rate'];

    $conn = new mysqli("127.0.0.1", "root", "liqifan6", "anime");
    if(!$conn){
        echo 'can not connect: ' . mysql_error();
    }

    $sql = "insert into anime_information(
        anime_id,title,cast,director,company,year,rate) values('$id','$
        title','$cast','$director','$company','$year','$rate')";

    $obj = $conn->query($sql);

    if($obj){
        echo "Add Success<a href = 'home.php'><br />Return</a>";
    }else{
        echo "Add Fail<a href = 'home.php'><br />Return</a>";
    }
    $conn->close();
?>
```

**dosearch.php**

dosearch.php

```php
<?php

    $id = $_GET['id'];


    $conn = new mysqli("127.0.0.1", "root", "liqifan6", "anime");

    if(!$conn){
        echo 'can not connect: ' . mysql_error();
    }

    $sql = "select * from anime_information where anime_id=$id";

    $result = $conn->query($sql);

    echo "<center>";

    echo "<table border = 1 cellspacing = '0' cellpadding = '10'>";
    echo "<th>ID</th><th>Title</th><th>Cast</th><th>Director</th><th>
      Company</th><th>Year</th><th>Rate</th>";
    while($row = mysqli_fetch_assoc($result)){
        echo "<tr>";
            echo '<td>'.$row['anime_id'].'</td>';
            echo '<td>'.$row['title'].'</td>';
            echo '<td>'.$row['cast'].'</td>';
            echo '<td>'.$row['director'].'</td>';
            echo '<td>'.$row['company'].'</td>';
            echo '<td>'.$row['year'].'</td>';
            echo '<td>'.$row['rate'].'</td>';
        echo "</tr>";
        echo "<a href = 'home.php'>Return</a>";
    }

    echo "</table>";

    echo "<center>";

    $conn->close();
?>
```

**doupdate.php**

doupdate.php

```php
<?php
    $id = $_GET['id'];
    $title = $_GET['title'];
    $cast = $_GET['cast'];
    $director = $_GET['director'];
    $company = $_GET['company'];
    $year = $_GET['year'];
    $rate = $_GET['rate'];

    $conn = new mysqli("127.0.0.1", "root", "liqifan6", "anime");
    if(!$conn){
        echo 'can not connect: ' . mysql_error();
    }

    $sql = "update anime_information set title = '$title', cast = '$cast
        ', director = '$director', company = '$company', year = '$year',
        rate = '$rate' where anime_id =$id";
    $obj = $conn->query($sql);
    if($obj && mysqli_affected_rows($conn)){
        echo "Update Success<a href = 'home.php'><br />Return</a>";
    }else{
        echo "Update Fail<a href = 'home.php'><br />Return</a>";
    }

    $conn->close();

?>
```

**delete.php**

delete.php

```php
<title>del.php</title>

<?php

    $id =$_GET['id'];
    $conn = new mysqli("127.0.0.1", "root", "liqifan6", "anime");
    if(!$conn){
        echo 'can not connect: ' . mysql_error();
    }

    $sql = "delete from anime_information where anime_id =$id";
    $result = $conn->query($sql);
    if($result && mysqli_affected_rows($conn)){
        echo "Delete Success<a href = 'home.php'><br />Return</a>";
    }else{
        echo "Delete Fail<a href = 'home.php'><br />Return</a>";
    }

    $conn->close();
?>
```

**cast.php**

cast.php

```php
<?php

    $conn = new mysqli("127.0.0.1", "root", "liqifan6", "anime");

    if(!$conn){
        echo 'can not connect: ' . mysql_error();
    }

    $sql = "select * from cast_information";

    $obj = $conn->query($sql);

    echo "<center>";

    echo "<table border = 1 cellspacing = '0' cellpadding = '10'>";
    echo "<th>Cast</th><th>Gender</th><th>Birthday</th><th>
      Masterpiece</th>";
    while($row = mysqli_fetch_assoc($obj)){
        echo "<tr>";
            echo '<td>'.$row['cast'].'</td>';
            echo '<td>'.$row['gender'].'</td>';
            echo '<td>'.$row['birthday'].'</td>';
            echo '<td>'.$row['masterpiece'].'</td>';

        echo "</tr>";
    }
    echo "<a href = 'home.php'>Return</a>";
    echo "</table>";

    echo "<center>";

    $conn->close();
?>
```

**director.php**

director.php

```php
<?php

    $conn = new mysqli("127.0.0.1", "root", "liqifan6", "anime");

    if(!$conn){
        echo 'can not connect: ' . mysql_error();
    }

    $sql = "select * from director_information";

    $obj = $conn->query($sql);

    echo "<center>";

    echo "<table border = 1 cellspacing = '0' cellpadding = '10'>";
    echo "<th>Director</th><th>Works</th>";
    while($row = mysqli_fetch_assoc($obj)){
        echo "<tr>";
            echo '<td>'.$row['director'].'</td>';
            echo '<td>'.$row['works'].'</td>';

        echo "</tr>";
    }
    echo "<a href = 'home.php'>Return</a>";
    echo "</table>";

    echo "<center>";

    $conn->close();
?>
```

**company.php**

company.php

```php
<h3 class="margin">Company Information</h3>
<?php

    $conn = new mysqli("127.0.0.1", "root", "liqifan6", "anime");

    if(!$conn){
        echo 'can not connect: ' . mysql_error();
    }

    $sql = "select * from company_information";

    $obj = $conn->query($sql);

    echo "<center>";

    echo "<table border = 1 cellspacing = '0' cellpadding = '10'>";
    echo "<th>Company</th><th>Keypeople</th><th>Website</th>";
    while($row = mysqli_fetch_assoc($obj)){
        echo "<tr>";
            echo '<td>'.$row['company'].'</td>';
            echo '<td>'.$row['keypeople'].'</td>';
            echo '<td>'.$row['website'].'</td>';

        echo "</tr>";
    }
    echo "<a href = 'home.php'>Return</a>";
    echo "</table>";

    echo "<center>";

    $conn->close();
?>
```

# Discussion

Before doing this project, I have never developed a real database myself. This project gives me a chance to have some experience to make use of the knowledge learnt from database class. Also, I become more familiar with SQL after building the database and learn a lot about the html, php while designing the user interface of this system. As for difficulties, I met several difficulties during the developing process and I will just list some of them below. First, it took me long time to build the local developing environment, that was, setting up Apache server and MySQL server correctly and connecting the database to the server using PHP. Second, after I finished the design of database schema and imported the data to the database, I found that not all the data was imported to the database. Then I checked if there were several tuples that were the same. Since nothing were looking up, I went on to check the foreign key restriction of the database and found that I forgot to add some necessary information to the table that it references. After I fixed it, I succeeded in importing all data to the database correctly. At last, there were a lot of bugs after I built the system and I tested my system several times to refine my code. After completing this project, I learnt a lot about how to build the frontend and backend of the website application using HTML, CSS and PHP, and how to connect them to the database and manipulate the database.