Milestone 5
MyCUPool

Team Members:

Gustav Solis

Michael Whitlock

Sam Bennetts

Yi Wu

Yuxuan Luo

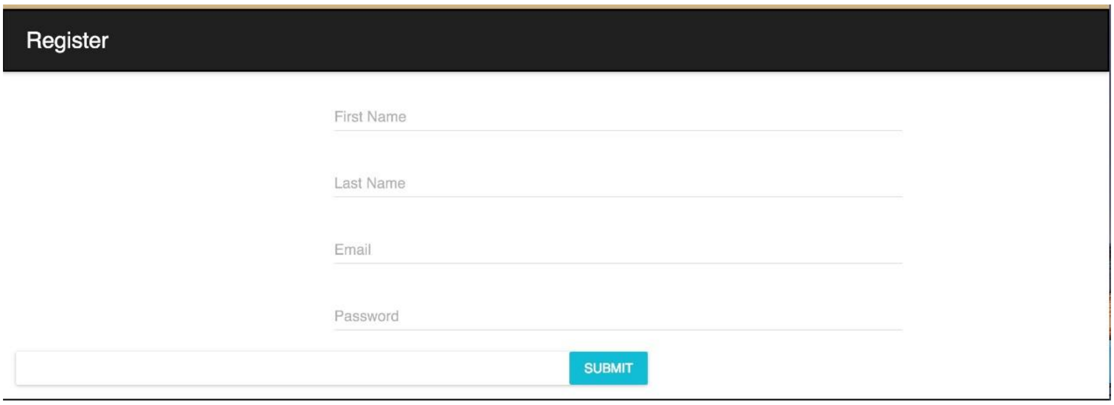
Zhiren Chen

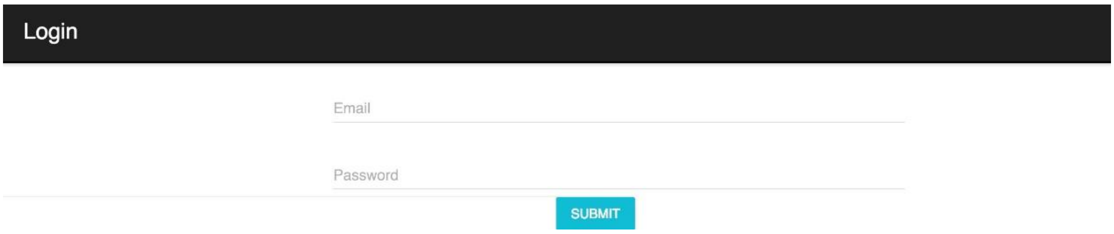
User Acceptance Tests:

The goal of user acceptance testing is to make sure that our web app matches user expectations.

UAT #1: **User can access the SignUp page and LogIn page.**

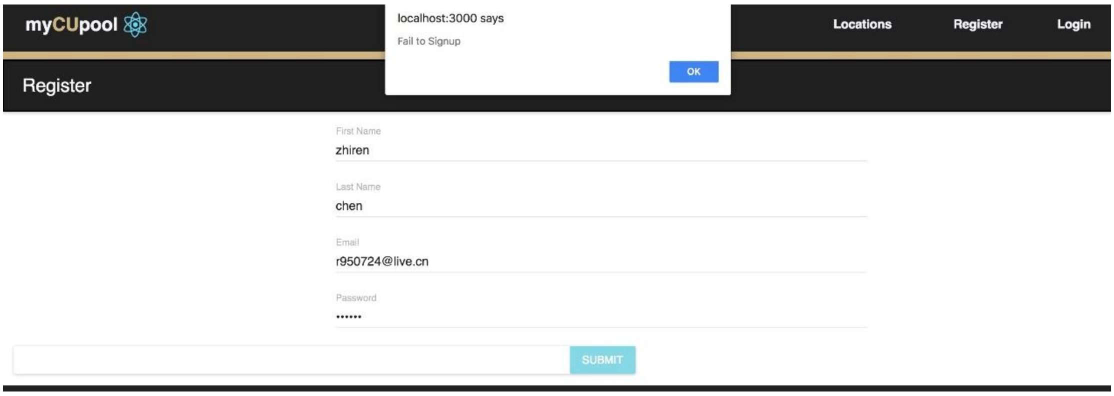
In this test, what we need is users are able to open the signup and login page. The test would be complete once the login and signup pages show up.





UAT #2: **User can create an account through the SignUp page, only if they input correct format of Email address.**

This test case would help users to create their account. Users can do the signup by entering their name, email address and password. If the email address was used before (That means this email address is already in the database), there will be an alert says: “ Fail to Signup”. If the user enters a invalid email address, another alert will be showed and says: “Need valid email address!”. Once the user enters a valid Email address, they will create an account, and after click the “submit” button, there will be a alert says: “Successfully Registered!”.



localhost:3000 says

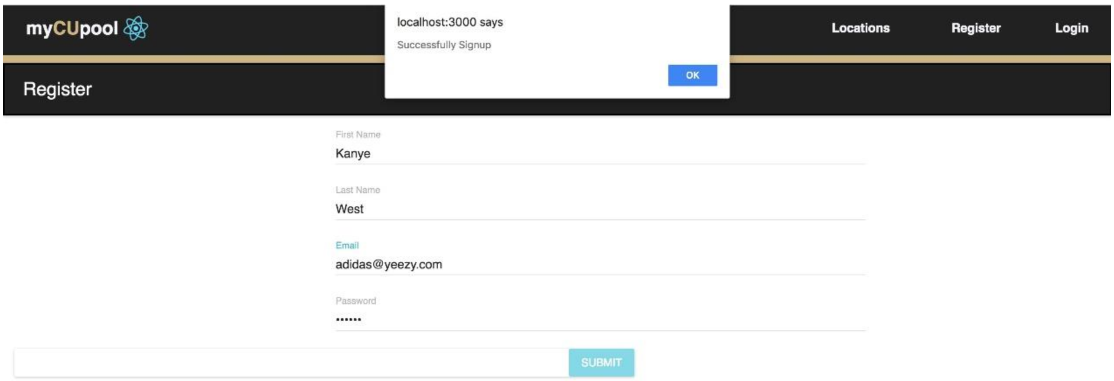
Fail to Signup

First Name
zhiren

Last Name
chen

Email
r950724@live.cn

Password
••••••



localhost:3000 says

Successfully Signup

First Name
Kanye

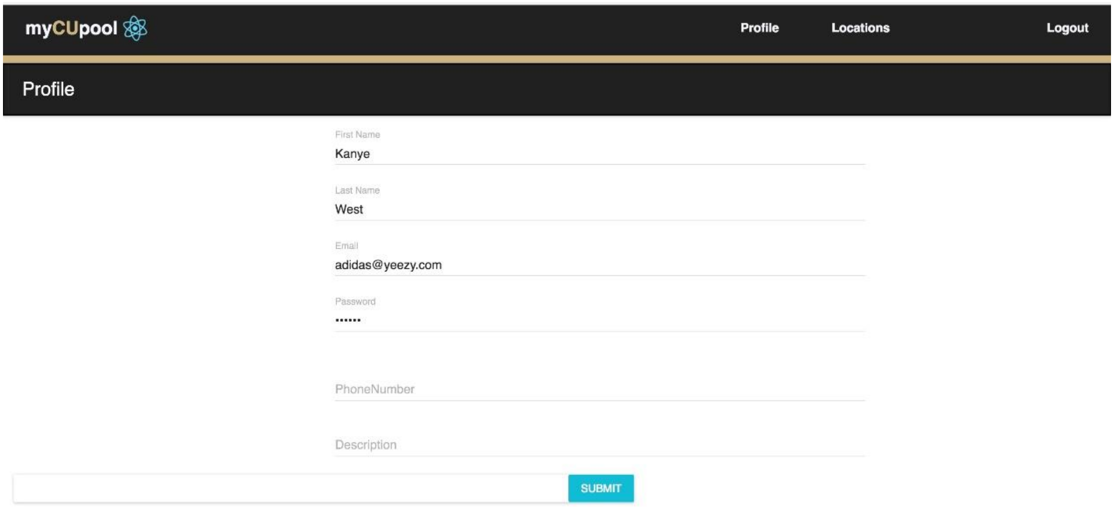
Last Name
West

Email
adidas@yeezy.com

Password
••••••

UAT #3: **Users will be redirected to the profile page after they login.**

After user enters a valid email address and a correct password, he/she will be redirected to the profile page, which shows all the user’s information.



Profile

First Name
Kanye

Last Name
West

Email
adidas@yeezy.com

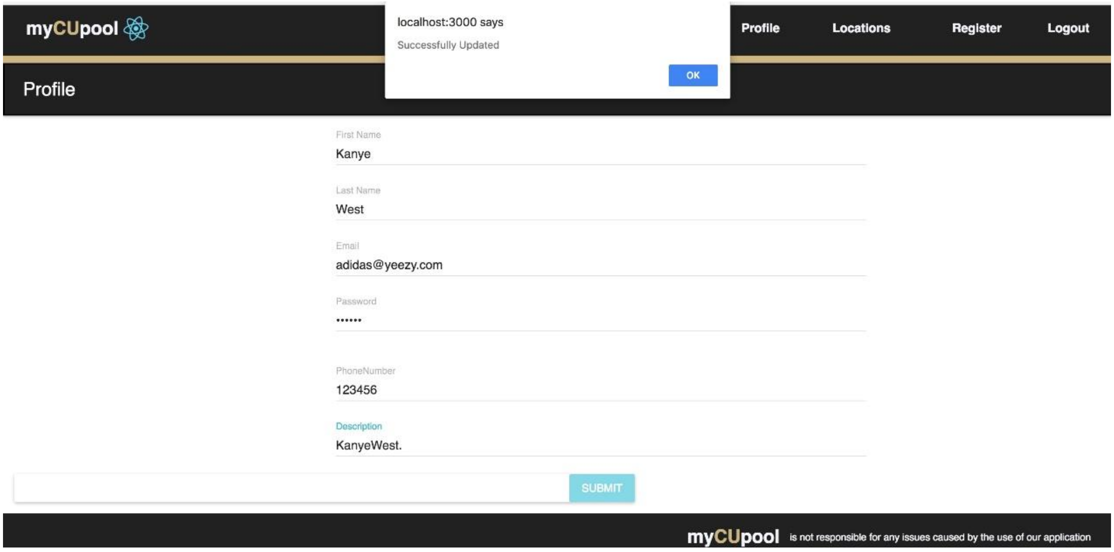
Password
••••••

PhoneNumber

Description

UAT #4: **Users can update their information through the profile page.**

In the profile page, users can change their password and personal information. They will also be able to add their phone number and personal description. After updating information, the website will show an alert says: “Successfully Updated!”.



localhost:3000 says
Successfully Updated

Profile

First Name
Kanye

Last Name
West

Email
adidas@yeezy.com

Password
••••••

PhoneNumber
123456

Description
KanyeWest.

myCUpool is not responsible for any issues caused by the use of our application

Automated Testing:

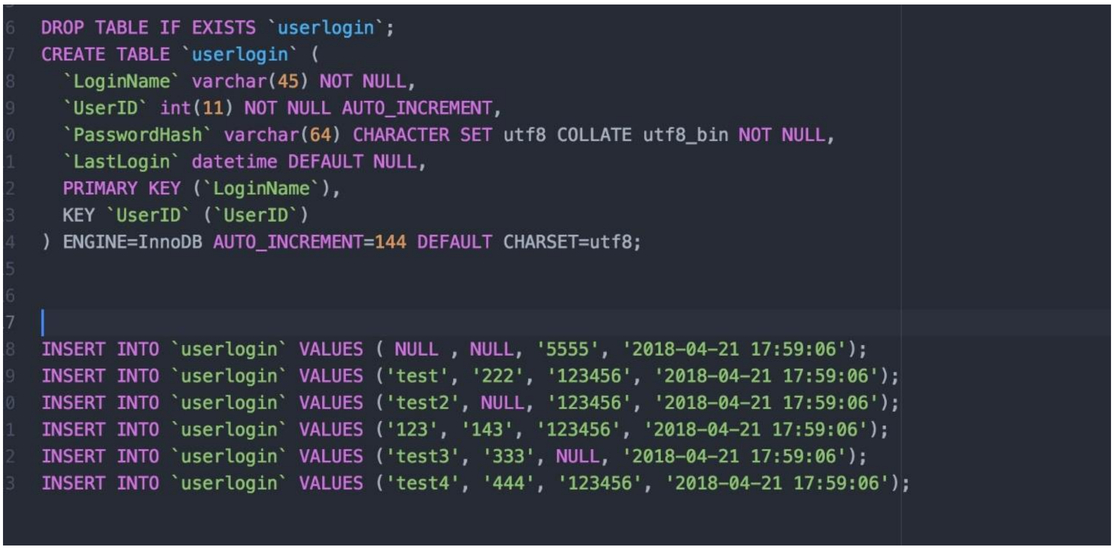
The goal of automated testing is to make sure our app is working fine and if there is any error, we'll throw/fix them.

(We decided to make an appointment with our TA, to demonstrate our automated tests)

Automated Test #1: **The database testing(MySql).[TOOL: MySql Command Client]**

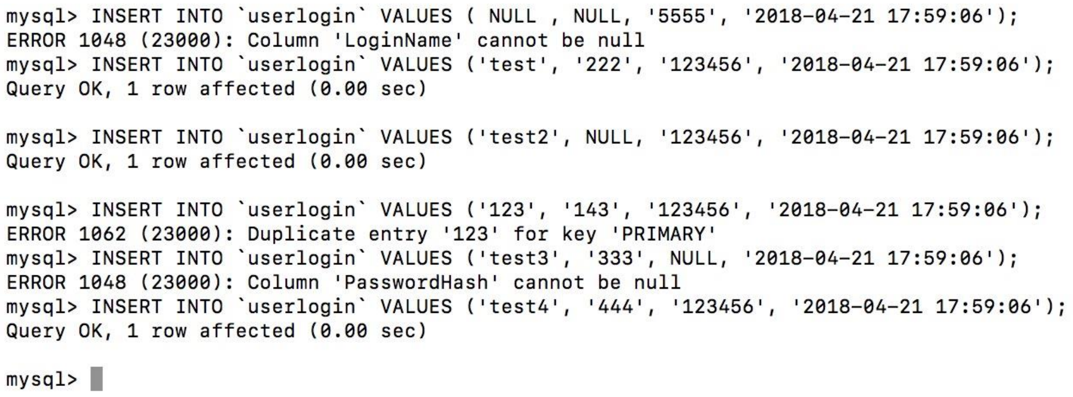
In this test case, we wrote some code that contains some incorrect MySql queries, and by putting these queries into our userlogin database, we received some error returns for the incorrect queries and we also received some passed returned for those correct queries.

The test query:



```sql
DROP TABLE IF EXISTS `userlogin`;
CREATE TABLE `userlogin` (
  `LoginName` varchar(45) NOT NULL,
  `UserID` int(11) NOT NULL AUTO_INCREMENT,
  `PasswordHash` varchar(64) CHARACTER SET utf8 COLLATE utf8_bin NOT NULL,
  `LastLogin` datetime DEFAULT NULL,
  PRIMARY KEY (`LoginName`),
  KEY `UserID` (`UserID`)
) ENGINE=InnoDB AUTO_INCREMENT=144 DEFAULT CHARSET=utf8;



INSERT INTO `userlogin` VALUES ( NULL , NULL, '5555', '2018-04-21 17:59:06');
INSERT INTO `userlogin` VALUES ('test', '222', '123456', '2018-04-21 17:59:06');
INSERT INTO `userlogin` VALUES ('test2', NULL, '123456', '2018-04-21 17:59:06');
INSERT INTO `userlogin` VALUES ('123', '143', '123456', '2018-04-21 17:59:06');
INSERT INTO `userlogin` VALUES ('test3', '333', NULL, '2018-04-21 17:59:06');
INSERT INTO `userlogin` VALUES ('test4', '444', '123456', '2018-04-21 17:59:06');
```

Console logs from MySql:

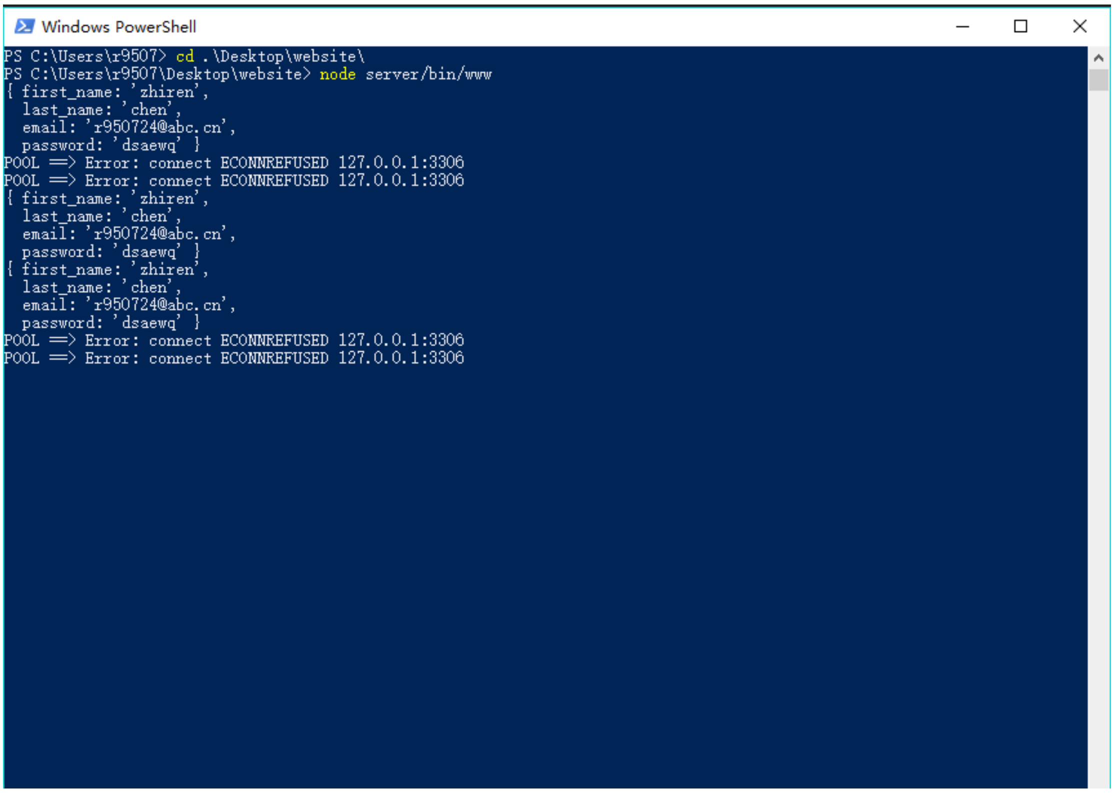


```
mysql> INSERT INTO `userlogin` VALUES ( NULL , NULL, '5555', '2018-04-21 17:59:06');
ERROR 1048 (23000): Column 'LoginName' cannot be null
mysql> INSERT INTO `userlogin` VALUES ('test', '222', '123456', '2018-04-21 17:59:06');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO `userlogin` VALUES ('test2', NULL, '123456', '2018-04-21 17:59:06');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO `userlogin` VALUES ('123', '143', '123456', '2018-04-21 17:59:06');
ERROR 1062 (23000): Duplicate entry '123' for key 'PRIMARY'
mysql> INSERT INTO `userlogin` VALUES ('test3', '333', NULL, '2018-04-21 17:59:06');
ERROR 1048 (23000): Column 'PasswordHash' cannot be null
mysql> INSERT INTO `userlogin` VALUES ('test4', '444', '123456', '2018-04-21 17:59:06');
Query OK, 1 row affected (0.00 sec)

mysql> ▌
```

Automated Test #2: **The program testing(MySql).[TOOL: React.js and Express.js]**

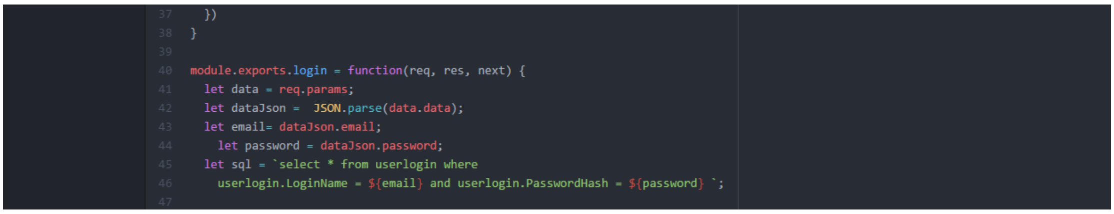
In this test case, we are testing the database's responses. We are sending requests and data from frontend to backend, and by observing the console logs, we found some incorrect statement in our code(/server/controller/userinfo.js):

The errors in backend:



```
PS C:\Users\r9507> cd .\Desktop\website\
PS C:\Users\r9507\Desktop\website> node server/bin/www
{ first_name: 'zhiren',
  last_name: 'chen',
  email: 'r950724@abc.cn',
  password: 'dsaewq' }
POOL ==> Error: connect ECONNREFUSED 127.0.0.1:3306
POOL ==> Error: connect ECONNREFUSED 127.0.0.1:3306
{ first_name: 'zhiren',
  last_name: 'chen',
  email: 'r950724@abc.cn',
  password: 'dsaewq' }
{ first_name: 'zhiren',
  last_name: 'chen',
  email: 'r950724@abc.cn',
  password: 'dsaewq' }
POOL ==> Error: connect ECONNREFUSED 127.0.0.1:3306
POOL ==> Error: connect ECONNREFUSED 127.0.0.1:3306
```

Incorrect statement was found:



```
  })
}

module.exports.login = function(req, res, next) {
  let data = req.params;
  let dataJson =  JSON.parse(data.data);
  let email= dataJson.email;
    let password = dataJson.password;
  let sql = `select * from userlogin where
    userlogin.LoginName = ${email} and userlogin.PasswordHash = ${password} `;

```