Jot: Team 4

CSCI 3308: Milestone 5

Alec Fiala, Bryan Geltz, Girish Narayanswamy,
Hussain Quadri, Jason Savath

18 April 2017

1 User Acceptance Test Plans

There are thee specific unit tests of the software at this point in development. These tests cover the basic functionality of the the web pages interacting with the database as well as the database itself.

1.1 Unit Test 1: User Account Sign-In

One of the fundamental features of the site is the ability of users to sign in to be able to access and upload materials. This feature is written in PHP code and interfaces with an user account information database with the relevant user name and password information. This sign in page should fundamentally recognize whether or not the information the user enters into the username and password fields is already in the database of use information, and if so log the user into additional parts of the website. If the user is not in the existing database, the login function should refuse the user entry to these aspects of the site. If a user does not have an account, and therefore an entry in the database, and wants to create an account, this site function should forward the user to page that allows the user to input new account information into the database. To test this function, test inputs will be created to verify all these requirements. These test inputs must verify that the page can accept correct inputs, i.e. inputs that exist in the database, and correctly respond, as well as incorrect inputs. Incorrect inputs consist of both inputs that simply do not exist in the database as well as adversarial inputs that test the error handling ability of the code. These adversarial inputs consist of inputs to long or incorrectly formatted for the database.

1.2 Unit Test 2: User Account Database

The next major code feature is the management of the user account database. This database maintains the user accounts for the website, and is what is searched through in the login process. This database must contain user names, passwords, as well as associated account information such as student ID number. To test this database, both good and bad inputs will be inserted into the database to determine how the software reacts to a given input. At a low level functionality, the database will be tested to ensure the correct number of entries are inserted into the database, as well as with the correct information in the correct portion of the table. Much of this testing can be done through visual inspection by checking the resulting entries after both good and bad inputs to the database. Adversarial inputs, checking that inputs such as SQL commands and other incorrect formatting are caught by the code's error handling features will be employed to test this aspect of code functionality. Integration testing will be completed by inputs to the website, verifying that the response of the website to both nominal and adversarial inputs.

1.3 Unit Test 3: Note Database

The third major feature of the website is the storage feature of student notes. This feature is the main aspect of the website, as it is what allows students to share their notes. This feature of the software must be able to accept user uploaded files of the correct format, store them within a database, and be able to retrieve them on request. Testing of this feature verifies the functionality of all three of these. For uploads, testing consists of uploading files with a range of file types and sizes to ensure that the site can register the user input. The second test ensures that the website is able to direct this user input to the correct database for storage, as well as to store the information correctly in the note database. The third part requires that the user be able to search for and retrieve notes that are contained within this database. This will be verified by testing database search functionality with sanitized and unsanitized inputs, and checking the response of the site and database.

2 Automated Test Documentation

2.1 SQL Testing

Testing within SQL was done with the following code. The output of the code is displayed in comments below the code.

```
/* This is a test case for sql to make sure it can set up a database, table, and
check for the correct types of input */

/* to run this please start mysql and path to this : /path/to/file.sql_test.sql; */

create database test_sql;

use test_sql;

create table if not exists `simple_table` (
 `student_Id` int(9) not null,
 `PassowordHash` varchar(40) not null,
 `first_Name` varchar(40) not null,
 `last_Name` varchar(40) not null,

 primary key (`student_Id`)
);

insert into simple_table Values ('987654321', 'test', 'testCase1', 'testCase1');

insert into simple_table Values ('!@#$%^&**', '7357', 'testCase2', 'testCase2');

insert into simple_table Values ('abcdefghij', 'test', 'testCase3', 'testCase3');

insert into simple_table Values ('987654312', '1234', 'testCase4', 'testCase4');

insert into simple_table Values ('987654213', 'test', 'testCae5', 'testCase5');

insert into simple_table Values ('987653214', 'test', 'testCase6', 'testCase6');

insert into simple_table Values ('987643215', 'test', 'testCase7', 'testCase7');

insert into simple_table values ('','','','');

select count(*) from simple_table; /*this will return 5 if the database is correctly config

delete from simple_table where student_Id='987654321';

delete from simple_table where student_Id='!@#$%^&*&';
```

```
select count(*) from simple_table; /* this will return 4 if the delete is working correctl

drop database test_sql;

/* a proper test will out put

Query OK, 1 row affected (0.00 sec)

Database changed
Query OK, 0 rows affected (0.01 sec)

Query OK, 1 row affected (0.01 sec)

ERROR 1366 (HY000): Incorrect integer value: '!@#$%^&**' for column 'student_Id' at row 1
ERROR 1366 (HY000): Incorrect integer value: 'abcdefghij' for column 'student_Id' at row 1
Query OK, 1 row affected (0.00 sec)

Query OK, 1 row affected (0.00 sec)

Query OK, 1 row affected (0.00 sec)

Query OK, 1 row affected (0.00 sec)

ERROR 1366 (HY000): Incorrect integer value: '' for column 'student_Id' at row 1
+----------+
| count(*) |
+----------+
|        5 |
+----------+
1 row in set (0.00 sec)

Query OK, 1 row affected (0.00 sec)

Query OK, 0 rows affected (0.00 sec)

+----------+
| count(*) |
+----------+
|        4 |
+----------+
1 row in set (0.01 sec)

Query OK, 1 row affected (0.01 sec)

mysql> source /var/www/html/sql_test.sql;
Query OK, 1 row affected (0.00 sec)

Database changed
Query OK, 0 rows affected (0.03 sec)

Query OK, 1 row affected (0.01 sec)

ERROR 1366 (HY000): Incorrect integer value: '!@#$%^&**' for column 'student_Id' at row 1
ERROR 1366 (HY000): Incorrect integer value: 'abcdefghij' for column 'student_Id' at row 1
```

```
Query OK, 1 row affected (0.00 sec)

Query OK, 1 row affected (0.01 sec)

Query OK, 1 row affected (0.01 sec)

Query OK, 1 row affected (0.00 sec)

ERROR 1366 (HY000): Incorrect integer value: '' for column 'student_Id' at row 1
+----------+
| count(*) |
+----------+
|        5 |
+----------+
1 row in set (0.00 sec)

Query OK, 1 row affected (0.00 sec)

Query OK, 0 rows affected (0.00 sec)

+----------+
| count(*) |
+----------+
|        4 |
+----------+
1 row in set (0.00 sec)

Query OK, 1 row affected (0.03 sec)

*/
```

2.2 PHP Testing

Testing with PHP code aspeccts was done with the following code.

```
<?php
  $connection = @mysqli_connect ('localhost', 'root',
'password', 'jot');

  if(mysqli_connect_errno())
{
 echo "<h4> Failed to connect to MySQL: </h4>" . mysqli_connect_error();
}
else
{
 echo "<h4> Successfully connected to MySQL: </h4>";
}

$query = "INSERT INTO simple_table VALUES ('192837465','test1','test1','test1');"; // Test

$resultset = mysqli_query($connection,$query);

if($resultset > 0)
{
    echo "Test␣1␣passed:␣Correct␣input␣test␣(should␣pass)<br>";
}
else
{
    echo "Test␣1␣failed␣:␣Check␣you␣table␣field␣for␣proper␣type<br>";
}

$query = "DELETE␣from␣simple_table;";
mysqli_query($connection,$query);

$query = "INSERT␣INTO␣simple_table␣VALUES␣('91@@@3423','test2','test2','test2');";// Test

$resultset = mysqli_query($connection,$query);
if($resultset > 0)
{
    echo "test␣2␣passed:␣Bad␣input␣test␣(should␣fail)<br>";
}
else
{
    echo "Test␣2␣failed␣:␣Bad␣input␣for␣student␣Id␣(should␣fail)<br>";
}
$query = "DELETE␣from␣simple_table;";
mysqli_query($connection,$query);

$query = "INSERT␣INTO␣simple_table␣VALUES␣('192abc465','test3','test3','test3');";// Test
with letters

$resultset = mysqli_query($connection,$query);
if($resultset > 0)
{
    echo "Test␣3␣passed:␣Bad␣input␣test␣(should␣Fail)<br>";
}
else
{
    echo "Test␣3␣failed␣:␣This␣should␣fail␣bad␣input␣letters␣in␣numbers␣for␣student␣Id<br>
}
$query = "DELETE␣from␣simple_table;";
mysqli_query($connection,$query);

$query = "INSERT␣INTO␣simple_table␣VALUES␣('192834765','','test4','test4');"; // Test for

$resultset = mysqli_query($connection,$query);
if($resultset > 0)
{
    echo "Test␣4␣passed:␣blank␣input␣test␣{password}␣(should␣fail)<br>";
}
else
{
    echo "Test␣4␣failed␣:␣should␣fail␣blank␣input␣for␣password<br>";
}
$query = "DELETE␣from␣simple_table;";
mysqli_query($connection,$query);

$query = "INSERT␣INTO␣simple_table␣VALUES␣('','test5','test5','test5');";// Test for wrong
```

```
$resultset = mysqli_query($connection,$query);
if($resultset > 0)
{
    echo "test␣5␣passed:␣blank␣input␣test␣{student_Id}␣(should␣fail)<br>";
}
else
{
    echo "Test␣5␣failed␣:␣Should␣fail␣blank␣student␣id<br>";
}
$query = "DELETE␣from␣simple_table;";
mysqli_query($connection,$query);

$query = "INSERT␣INTO␣simple_table␣VALUES␣('192873465','TEST6','TEST6','TEST6');"; // Test

$resultset = mysqli_query($connection,$query);
if($resultset > 0)
{
    echo "Test␣6␣passed:␣Correct␣input␣test␣with␣capitals␣(should␣pass)<br>";
}
else
{
    echo "Test␣6␣failed␣:␣Check␣you␣table␣field␣for␣proper␣type<br>";
}
$query = "DELETE␣from␣simple_table;";
mysqli_query($connection,$query);

$query = "INSERT␣INTO␣simple_table␣VALUES␣('19283','7357','7357','7357');"; // Test for wr

$resultset = mysqli_query($connection,$query);
if($resultset > 0)
{
    echo "Test␣7␣passed:␣Bad␣input␣test␣numbers␣insted␣of␣letter␣(should␣fail)<br>";
}
else
{
    echo "Test␣7␣failed␣:␣Should␣fail␣bad␣input␣numbers␣insted␣of␣letters<br>";
}
$query = "DELETE␣from␣simple_table;";
mysqli_query($connection,$query);

$query = "INSERT␣INTO␣simple_table␣VALUES␣('','','','');"; // Test for wrong input type :

$resultset = mysqli_query($connection,$query);
if($resultset > 0)
{
    echo "test␣8␣passed:␣All␣blank␣input␣test␣(should␣fail)<br>";
}
else
{
    echo "Test␣8␣failed␣:␣Should␣fail␣all␣blank␣input<br>";
}
$query = "DELETE␣from␣simple_table;";
mysqli_query($connection,$query);
```

```
$query = "INSERT␣INTO␣simple_table␣VALUES␣('192837465','test9','test9','');"; // Test for

$resultset = mysqli_query($connection,$query);
if($resultset > 0)
{
    echo "Test␣9␣passed:␣blank␣input␣test␣last-name␣(should␣fail)<br>";
}
else
{
    echo "Test␣9␣failed␣:␣should␣fail␣blank␣last-name<br>";
}
$query = "DELETE␣from␣simple_table;";
mysqli_query($connection,$query);

$query = "INSERT␣INTO␣simple_table␣VALUES␣('','test10','','');";
// Test for missing fields

$resultset = mysqli_query($connection,$query);
if($resultset > 0)
{
    echo "test␣10␣passed:␣only␣password␣input␣test␣(should␣fail)<br>";
}
else
{
    echo "Test␣10␣failed␣:␣should␣fail␣only␣a␣password␣was␣entered<br>";
}
$query = "DELETE␣from␣simple_table;";
mysqli_query($connection,$query);

$query = "INSERT␣INTO␣simple_table␣VALUES␣('','','test11','');";
// Test for missing fields

$resultset = mysqli_query($connection,$query);
if($resultset > 0)
{
    echo "test␣11␣passed:␣only␣first-name␣(should␣fail)<br>";
}
else
{
    echo "Test␣11␣failed␣:␣should␣fail␣only␣a␣first␣name␣was␣entered<br>";
}
$query = "DELETE␣from␣simple_table;";
mysqli_query($connection,$query);

$query = "INSERT␣INTO␣simple_table␣VALUES␣('','','','test12');";
// Test for missing fields

$resultset = mysqli_query($connection,$query);
if($resultset > 0)
{
    echo "test␣12␣passed:␣only␣last␣name␣(should␣fail)<br>";
}
else
```

```
{
    echo "Test␣12␣failed␣:␣should␣fail␣only␣a␣last␣name␣was␣entered<br>";
}
$query = "DELETE␣from␣simple_table;";
mysqli_query($connection,$query);

  ?>
```