Software 6

4/16/18

Milestone 4 Beta Launch

and Reviews

Principles of Software Engineering, Spring 2018

CEN 4010

Team Info

Team name: Software 6

Project: Access Control Device

Team Number: Milestone Group 6

Team Members:

Timothy Duncan (Front/Back-end Developer) – [tdunca11@fau.edu](mailto:tdunca11@fau.edu)

Bentialy Saint Julien (Front-end Developer) – [bsaintju@fau.edu](mailto:bsaintju@fau.edu)

Jonathan Giger (Project Owner) – [jgiger2013@fau.edu](mailto:jgiger2013@fau.edu)

Jonathan Parreira (Back-end Developer) – [jparreir@fau.edu](mailto:jparreir@fau.edu)

Mihail Sandor (Scrum Master) [-msandor2014@fau.edu](mailto:-msandor2014@fau.edu)

Revision History Table:

Project revised on 4/16/2018

2.2 Product summary

1. Name of the product

Access Control System

1. Explicit list of ALL major committed functions.
2. The system is design to grant access to users that would be using lab equipment or workstation
3. The system is designed to keep track of all available workstations and lab equipment
4. The system shall grant a two-level access, students and administrators
5. The system shall grant access to users by using their credentials, student Z-number and password
6. The system shall only allow access to users who hold a valid student ID number (student Z-number)
7. The system shall allow the admin to add, delete or block a user from using the site
8. The system shall allow student users only to use the site for lab equipment or workstation.
9. The system shall assign randomly an available workstation to the user
10. The system shall keep track of the time allotted to each student using a workstation
11. The system will periodically remind the user of their remaining time and will be warned when the time is approaching 0
12. The system shall support different type of workstations (drill press, soldering station)
13. The system will be accessed via an internet browser on a mobile device (smartphone, tablets, etc.)
14. The system shall keep track of multiple devices such as lab tools, soldering workstations, etc.
15. Describe unique features in your product (if any)

The site will provide the user a selection of two choices to choose from, one if bench is clean and operable and the other if is not. If the bench is not in good shape the user will be asked to take a picture of the workstation which will then be send to EE management team, while the student will be assigned a new available workstation.

1. URL to your product accessible to instructors, on deployment server

http://lamp.cse.fau.edu/~CEN4010\_S2018g06/

2.3 Usability test plan – maximum 2 pages

Select ONE major function (NOT login or registration) to be tested for usability. We recommend search or upload/post.

Write a usability test plan for this selected function. Please consult class material on developing usability test plan and questionnaire. This test plan is to contain:

1) Test objectives: 0.5 page

2) Test plan: System setup, starting point, task to be accomplished, who is the intended user, completion criteria, URL of the system to be tested. 3/4 page

3) Questionnaire form: 3 Lickert scale questions, in a form easy to be used by reviewer (check class slides). 3/4 page

Your test plan must be formatted to be easy to read and use by usability testers, including the questionnaire.

You can also ask your friends or team members to do the usability test.

2.4 QA test plan – maximum 2.5 pages

For the same function you chose for the usability test, write AND execute a QA test plan (check class slides)

a) Create formal QA test plan (consult QA class material). Basically, it has to contain: 1) Test objectives: max 0.5 pages

2) Hardware and software setup: max 0.5 page

3) Feature to be tested: max 0.5 page 4) Actual test cases: 3 test cases and results of testing them on your system: 1 page

You must provide test plan and test summary in the format (e.g. form) allowing easy reading and analysis by management e.g. in a table format like presented in the lecture.

Suggested format for QA Table columns are: test #; test title; test description; test input; expected correct output; test results (PASS or FAIL for each tested browser) 5) Perform the testing as per plan above and record the results in a form above.

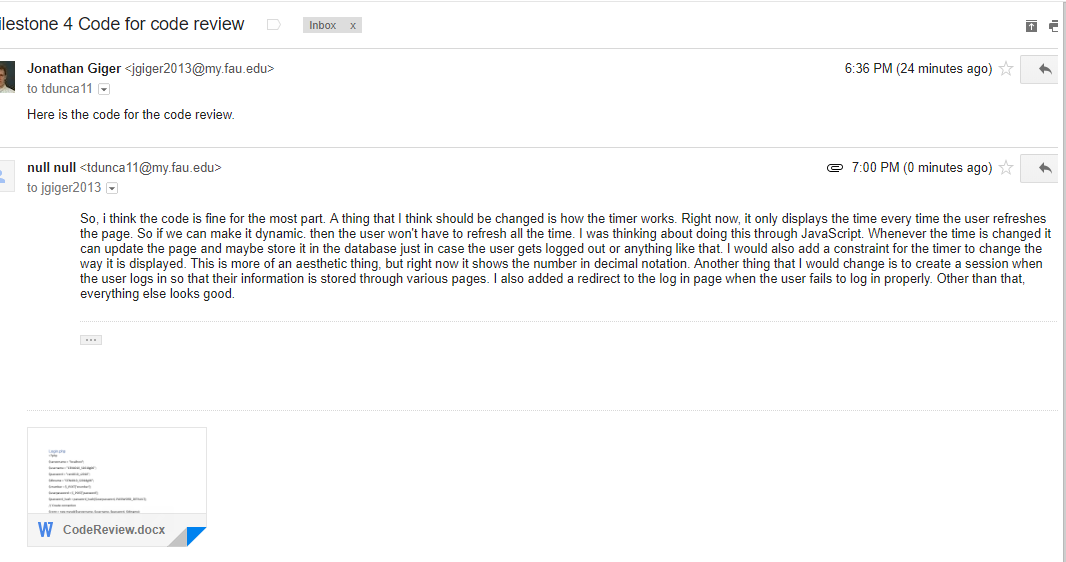
6) Apply the above test on 2 browsers of different type and record it in the above table

2.5 Code Review

By now you should have chosen a coding style. In the report state what coding style you chose.

Chose the code (substantial portion of it) related to the feature you used for QA and usability test. You need to submit an example of the code (or part of it – 2 pages or so MAX) for its function to be peer reviewed, and document this as follows:

1) One team member should submit code to other team member(s) for peer review. 2) Peer review should be performed by other group member(s) (1 review is OK).

3) Peer review is to be done by e-mail and comments are to be included in the code 4) Submit listing containing the peer review and commented code and communication related to this in your Milestone 4 document 

[Code for Timer](#Code)

2.6 Self-check on best practices for security – ½ page

The only assets that are being protected are the user credentials. This includes their z-number and password, more so their password. The password is encrypted using the PHP password\_hash function. The function takes in the user’s password as a parameter and passes it through the crypt function. The crypt function generates a random salt and uses that in the algorithm to encrypt the password. The resulted encrypted password is then stored in a variable that is then passed into the database. Along with the encrypted password, the salt that is generated and the algorithm used is also stored in the database so that it can be used to decrypt the password when validating a user’s credentials. The only information that is being validate is the user’s z-number and their password.

The z-number is validated through the input field in the html file. The field only takes numbers as inputs and the input must be 8 characters long as this is the length of a z-number. If the input is less than 8 characters, the user is notified with a message saying that it must be 8 characters long. If the input is anything other than a number, it is not registered. Passwords are validated through the input field in the html file also. The password is hidden when typed and it cannot exceed a length of 25 characters. This prevents any malicious code from being inputted into the system.

2.7 Self-check: Adherence to original Non-functional specs

Copy all original non-functional specs as in high level application document published at the very beginning of the class and then for each say DONE if it is done (which is expected and required); ON TRACK if it is in the process of being done and you are sure it will be completed on time; or ISSUE meaning you have some problems and then explain it.

Note: you must adhere to all original non-functional specs as published in the original high-level specification document. Failure to do so may cause reduced grade.

Code for Timer

## Login.php

//Create Session for future use

<?php

$servername = "localhost";

$username = "CEN4010\_S2018g06";

$password = "cen4010\_s2018";

$dbname = "CEN4010\_S2018g06";

$znumber = $\_POST['znumber'];

$userpassword = $\_POST['password'];

$password\_hash = password\_hash($userpassword, PASSWORD\_DEFAULT);

// Create connection

$conn = new mysqli($servername, $username, $password, $dbname);

// Check connection

if ($conn->connect\_error) {

die("Connection failed: " . $conn->connect\_error);

}

$sql = "select name from users where znumber = '".$znumber."' ";

$sql2 = "select times from users where znumber = '".$znumber."'";

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

$result2 = $conn->query($sql2);

//ResetTimer

function resettime()

{

$sql3 = "update users set time = " . time() . " where znumber = '".$znumber."'";

$conn->query($sql3);

}

if ($result->num\_rows > 0) {

// output data of each row

$row2 = $result2->fetch\_assoc();

while($row = $result->fetch\_assoc()) {

echo "Login Successful!<br>Your Name: " . $row["name"]. "<br>";

echo "Current Time: ".time()."<br>MySQL Time: ".$row2["times"]."<br>";

if(time() > $row2["times"] + 1800)

{

echo "Session time is up Start new session?<br><form action='called.php' method='POST'><input type='submit' class='button' name='reset' value='resettime' /><input type='text' name='znumber' value='". $znumber ."' /></form>";

}

else {

echo (1800-(time() - $row2["times"]))/60 . " minutes remaining";

}

}

} else {

echo "Login Failed!";

echo "<script>setTimeout(\"location.href = '../index.html ';\",1500);</script>";

}

$conn->close();

?>

# Called.php

<?php

$servername = "localhost";

$username = "CEN4010\_S2018g06";

$password = "cen4010\_s2018";

$dbname = "CEN4010\_S2018g06";

$znumber = $\_POST['znumber'];

// Create connection

$conn = new mysqli($servername, $username, $password, $dbname);

// Check connection

if ($conn->connect\_error) {

die("Connection failed: " . $conn->connect\_error);

}

$sql3 = "update users set times = " . time() . " where znumber = ".$znumber;

$conn->query($sql3);

echo "Workstation reserved successfully, use this site again to check how much time is remaining."; //Maybe throw in a redirect to another page

$conn->close();

?>