CEN 4010 Principles of Software Engineering Fall 2019

Milestone 4: Beta Launch and Final Projects Reviews

Project Title: Campus Times

Team Name: THE DREAM TECHS
Team/Group Number: 9

Group Members:

Samuel Parker(Scrum Master) - Sparke25@fau.edu

Abdullah Shafique(Front End Lead) - ashafique2014@fau.edu

Donny Stephens II(Back End Lead) - dsteph18@fau.edu

Joshua Vayntrub(Product Owner) - jvayntrub2015@fau.edu

Date: November 18, 2019

History Table:
1. 10-28-2019 Database setup.
2. 10-30-2019 Updated GUI.
3. 11-1-2019 Web Pages Created.
4. 11-2-2019 Website Connected to Database.
5. 11-12-2019 Rewrite Data Definition.
6. 11-12-2019 Prioritized the Functional Requirements.
7. 11-12-2019 System architecture and UML diagram updated.
8. 11-12-2019 Revised Risks and Mitigation Strategies.
9. 11-12-2019 Vertical Prototype Video Demo Created.
10. 11-12-2019 Team Member Participation Form Revised.

Product summary

Campus Times

1. **Create Account:** Users will create an account by entering their Email, Username, Password, and Confirmation Password. After this the User will select Student, Staff or Teacher. After the account has been created successfully, the user will be directed to the Members page.

2. **Signing In to Existing Account:** Once an account has been created Users will be able to sign in whenever visiting the website. To sign into Existing Account the User will enter Username and Password, the same Username and Password used to Create Account.

3. **Members Page/Status Update:** Once signed in the user will be directed to the members page where other members post can be seen. In addition to that feature Users will also have the option of posting a status update of their own. These updates can range in subject from events happening around campus to complaints about the libraries leaking toilet.

Unique Feature(s): One unique feature of our product will be the ability for users to submit a photo along with their status update. We wanted to add more expression to the members page by giving the users the option to post pictures along with their written post.

http://lamp.cse.fau.edu/~cen4010fal19_g09/project/

Usability Test Plan

Test Plan Goals and Objectives

- These tests are being performed on the Campus Snapshot system to ensure that the system can execute the uploading of pictures and texts correctly, and to ensure that the uploads are posted on the website, visible to all users that are signed in, and stored on the database. We are also performing these tests to make sure that the uploading and posting functionalities are easy for the average user to understand without having to possess a skill set beyond that of your standard internet surfer.
- By performing these tests, we hope to learn if there are any bugs in the uploading and posting functionality, and if found, to correct them accordingly. We also hope to learn through testing if there are any performance issues when users upload images and text, (e.g. the system taking too long to load under a specific condition or the system crashes altogether), if the system will accept different types of image file without causing the system to behave irregularly, (e.g. if a jpeg file or gif file will cause the system not to execute properly), and if the system's set parameters on image file size and word count maximum are holding, and if these limits are exceeded, will the system function correctly, or if it will display any irregularities.

System Setup

- The system setup will consist of a PC or mobile device, and an internet browser.

Intended Users

- The intended users will be someone who accesses the internet daily, who is also a campus goer, consisting of students, teachers, campus staff, and security.
- Users of the system will only be required to have the skill set of your average internet surfer, who has access to the internet, and does not have to possess any higher-level technical ability.

Starting Point

- The user will access the system by logging onto the website via a web browser. User will navigate to the login page, and log onto the system with the login credentials of a registered test account, previously set up specifically for this test.

Tasks

- Users commonly post images and text on the members page.
- Users will post images and text to test the system for any irregularities.
- Users will post images of different file types to check that system allows submission of multiple image types without irregular behavior, and to see if submission of an invalid file will cause irregular behavior. Users will submit images of varying sizes to test whether the system will support large image files. User will submit text of varying character count to test that the system will accept large texts without displaying irregular behavior.

Completion Criteria

- User can post an image and text onto the member page successfully. When posted, the submission is visible to the user, and appears on top of all previous user submissions, indicating that it is the most recent submission.
- The user can post multiple submissions, changing image file type and size, as well as text character count without out system lags or crashes.

URL of the system to be tested

- lamp.cse.fau.edu/~cen4010fal19_g09/project/login.php

		Strongly disagree	Neither	Strongly Agree
	User Questions			
1.	When logged in to the system, the members page allow you to upload an image and text of varying file type and size without lagging, crashing, or giving an error message?		●	
2.	Is the text and image that you submitted visible to you once it is posted, and does it show as the most reason items one the page, (e.g. is the		●	

	image and text as the top of the page, above all other postings)?			
3.	The system easy to use with minimal difficult to understand when posting your image and text?			●

QA test plan

QA Test Objectives

The overarching goal for the quality assurance test of the website is to cover a broad testing overview of the system to ensure the product is meeting all development requirements set forth. It is important to test each functionality multiple times in various ways to be certain there does not exist any unforeseen errors that could cause problems with the website that would hinder the products usability. What is hoped to be achieved is that errors are found during QA testing so that the testing can be considered a success.

QA Hardware/Software Setup

The setup for the system based on hardware will include a PC and a mobile device. The software setup will contain the internet browsers being tested on.

Features to be Tested

- Signing up a new user that does not already exist
- Signing up a new user but with information from a user that already exists (i.e. username and/or email)
- Logging in with a user that already exists
- Logging in with a user that does not exist
- Posts with images will show the correct image
- Posts without images won't show any images
- Posts with various image file types and sizes
- Posts will reflect the user correctly (e.g. will show the correct username, account type, and date of the post)
- New posts are added to the top of the page
- The UI is intuitive and structured/formatted correctly
- Logging out brings user to homepage
- Links on the page bring you to the correct area (e.g.login link brings to login)

Test Cases and Results

Test #	72456
Title	User Interface Functionality
Description	Ensure that the design of the website is formatted nicely and is pleasurable to the user's view
Input	N/A as this is a test case of visibility
Expected Output	The design of the website shows no UI errors visually and looks even and appealing to the user
Actual Output (Results): Browser Chrome Desktop	FAIL. The textbox for user input is not formatted properly in the surrounding box lines and overflows it
Actual Output (Results): Browser Safari Mobile	FAIL. On the main page where the placeholder image is showing, it is cut-off. Trying to zoom out causes the page to seem indented to the left. The textbox for user input is not formatted properly in the surrounding box lines and overflows it. The text for the file is also overflowing out of the box.

Test #	23971
Title	File types and sizes
Description	Insert images of various sizes and file types as a student, staff, and teacher account. See if there exists any errors that comes about because of the size of the image and/or the file type.
Input	JPG, PNG, and GIF files from desktop and mobile device.
Expected Output	Image file will be viewable and in an acceptable format for the user and will appear in the correct order
Actual Output (Results): Browser Chrome Desktop	PASS. The posts are there and the images are viewable. Posts are in order on the page.
Actual Output (Results): Browser Safari Mobile	PASS. The posts are there and the images are viewable. Posts are in order on the page.

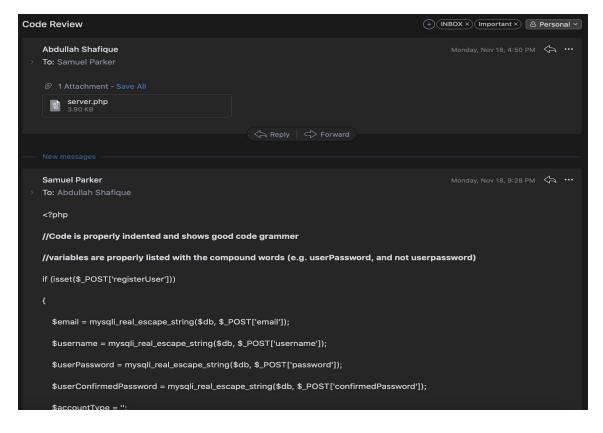| Test #	03813
Title	User reflection by post
Description	For each type of user, when they make a post, the post will include the username and account type of that user as well as the date the post was made
Input	Text posts
Expected Output	The correct text will be posted along with the correct username, account type, and date connected to the post.
Actual Output (Results): Browser Chrome Desktop	FAIL. Username is shown, but date and account type does not show

Actual Output (Results): Browser Safari Mobile	FAIL. Username is shown, but date and account type does not show

Code Review

For this project, we have chosen to use the object oriented style of coding. This style was chosen given the form data and submissions required to perform updating and input tasks for the website.

```php
1    <?php
2
3    //Code is properly indented and shows good code grammer
4
5    //variables are properly listed with the compound words (e.g. userPassword, and not userpassword)
6
7    if (isset($_POST['registerUser']))
8
9    {
10
11       $email = mysqli_real_escape_string($db, $_POST['email']);
12
13       $username = mysqli_real_escape_string($db, $_POST['username']);
14
15       $userPassword = mysqli_real_escape_string($db, $_POST['password']);
16
17       $userConfirmedPassword = mysqli_real_escape_string($db, $_POST['confirmedPassword']);
18
19       $accountType = '';
20
21       $accountTypeButtons = $_POST['accountType'];
22
23    //Good use of a conditional statement
24
25       switch($accountTypeButtons)
26
27       {
28
29          case "Student":
30
31          $accountType = "Student";
32
33          break;
34
35          case "Staff":
36
37          $accountType = "Staff";
38
39          break;
40
41          case "Teacher":
42
43          $accountType = "Teacher";
44
45          break;
46
47          default: die("Invalid type!");
48
49       }
50
```

```php
51         //Variable names correctly identify the purpose of the variable
52
53      $user_check_query = "SELECT * FROM User WHERE Email='$email' OR Username='$username' LIMIT 1";
54
55      $result = mysqli_query($db, $user_check_query);
56
57      $user = mysqli_fetch_assoc($result);
58
59   //Allot of unnecessary spacing below
60
61 ▼           {
62
63                 array_push($error, "Username/password is incorrect.");
64
65             }
66
67         }
68
69         else
70
71 ▼       {
72
73                             array_push($error, "Username/password is incorrect.");
74
75         }
76
77   }
78
79 }
80
81   //coded properly and functions correctly
82
83   if (isset($_POST['upload']))
84
85 ▼ {
86
87      $image = $_FILES['image']['name'];
88
89             $image_text = mysqli_real_escape_string($db, $_POST['image_text']);
90
91   //Below variable does not follow proper code grammer
92
93    $currentuser = $_SESSION['username'];
94
95             $target = "images/".basename($image);
96
97             $sql = "INSERT INTO Posts (IMGpath, Text, Username) VALUES ('$image', '$image_text', '$currentuser')";
98
99             mysqli_query($db, $sql);
100
101            if (move_uploaded_file($_FILES['image']['tmp_name'], $target)) { $msg = "Image uploaded successfully"; }
```

Code Review

Abdullah Shafique   Monday, Nov 18, 4:50 PM   ↩ ⋯
To: Samuel Parker

📎 1 Attachment - Save All

    📄 server.php
       3.90 KB

            ↩ Reply | ⇨ Forward

— New messages

Samuel Parker   Monday, Nov 18, 9:28 PM   ↩ ⋯
To: Abdullah Shafique

```php
<?php

//Code is properly indented and shows good code grammer

//variables are properly listed with the compound words (e.g. userPassword, and not userpassword)

if (isset($_POST['registerUser']))

{

   $email = mysqli_real_escape_string($db, $_POST['email']);

   $username = mysqli_real_escape_string($db, $_POST['username']);

   $userPassword = mysqli_real_escape_string($db, $_POST['password']);

   $userConfirmedPassword = mysqli_real_escape_string($db, $_POST['confirmedPassword']);

   $accountType = '';
```

Self-check on best practices for security

Major assets being protected are the users email and password. The password is encrypted using the Bcrypt hashing method on the database and the email is only used as a means to sign up and is not shown at all on the pages. When a user logs on the website, their credentials are checked on the database to make sure the information they are inputting is correct. If they enter invalid information on login, they will be notified that the username and/or password is incorrect. It works similarly for the signup page. If a user inputs certain information that is already being used by another user, the page will say that they cannot use those credentials as they have already been used by someone else. The images are stored in a folder within the website and when a user decides to delete them the image is also deleted from that folder.

Self-check: Adherence to original Non-functional specs

1. The application should run on two latest versions of all major web browsers: Mozilla, Safari, Chrome, IE, Firefox, etc. **DONE**

2. Data shall be stored in the database on the server **ON TRACK**

3. Privacy of users shall be protected, all privacy policies will be appropriately communicated to the users. **ON TRACK**

4. The language used shall be English. **DONE**

5. The site shall be very easy to use, intuitive (no prior training required). **DONE**

6. Site shall be attractive and media-rich in appearance. **ON TRACK**

7. Security of the site shall require users to register and log in (students to decide what functions require registration and login). **DONE**

8. Effort shall be made to make the site easily searchable by internet search. **ISSUE**

We are currently unsure how we are going to make the site searchable by internet search. We are currently researching a solution to this non-functional requirement.

Team member participation form:

Team Members:

Samuel Parker = 25pts, Abdullah Shafique = 25pts, Donny Stephens = 25pts, Joshua Vayntrub = 25pts