CEN 4010 Summer 2020 Milestone 4 Beta Launch and Reviews

COVID-19 News Website

Group 08

Scrum Master: Kyle Costela

Product Owner: Jordan Barrios

Project/Development Team: Muhammad Asmar, Parker Hawkins

July 28, 2020

Product summary

Product Name: COVID-19 News Website

Major committed functions (Priority 1):

1. Discussion Board/Comment Section
2. Account Creation
3. Pandemic data

Our COVID-19 News Website is intended to keep people not only informed, but also to keep them connected. The website always brings you the latest news from a wide range of news resources across the globe, dedicated to the ongoing pandemic. Included with each displayed article comes a comment section, in which you can freely express your opinions on the given article and have meaningful discussions with other users. Even while in social isolation, you can still be connected to others. Additionally, the website provides up-to-date statistics on the COVID-19 pandemic, both global and by country. By simply typing the name of your desired country, you can be informed instantly with all the latest data. Stay inside while staying informed with the COVID-19 News Website.

URL: https://lamp.cse.fau.edu/~cen4010s2020_g08/

Usability test plan

Test objectives:

Main objective is to verify that the comment section is working as intended. Some of the objectives are to make sure that the comments properly get stored in the database table. Also, that the comments are properly pulled and displayed to the user. Another objective is to validate that the comments that get published are properly connected to the article that the user intended to comment on.

Test plan:

To verify that the overall functionality of the comment section is working as intended, the following will be tested: comments are published correctly into database tables, comments are correctly loaded from tables to the right articles and overall display of the comment section is good. To verify that the login and create account is functioning properly, a dummy username and password can be created using the create account section of the index.html file. Using those credentials, the corresponding username and password can be entered into the login portion of the index.html file. If the login is successful, then the user will then be escorted to the coronavirus home page. Another way to verify that the system is functioning properly, when the user creates an account, shortly after by refreshing the phpMyAdmin that the developers have access to should be the username that was just entered when creating an account. If the username does not appear in the database, it either means that the username already exists and cannot be added again or that the connection to the database has failed. If the username already exists, an error message is displayed that explains that there cannot be a duplicate username. Using the database to confirm that a username and password has gone through the system can be used to validate the system. If the password on the database is a long string that is not the original password that the user has entered, this means that the password was encrypted properly before it was stored on the server.

Url to be tested(any article can be chosen):

https://lamp.cse.fau.edu/~cen4010s2020_g08/home.php

Starting point - Article that has no comments associated with it will first be tested. The user will comment a few times and then check database table with article comments. First validate that the comments were associated with the right article. After, refresh the pages to validate that the same comments are being loaded from the database and displayed to the end user. Last test after main functionality testing is done is to verify that overall design and feel of the product meets user needs.

Questionnaire:

The website and its functionalities have a user-friendly interface.

- Strongly agree
- Agree
- Neutral
- Disagree

- Strongly disagree

I would use this website frequently to stay up-to-date on the COVID pandemic.

- Strongly agree
- Agree
- Neutral
- Disagree
- Strongly disagree

QA test plan

Test objectives:

The objective of this test is to ensure the functionality of the comment section. That is to say, that comments submitted by the user are properly stored in the database. Additionally, when the associated article is pulled up, the proper comments, along with the names of the users who wrote them, are retrieved and appear under the given article. Comments from other articles should not appear under this given article.

Hardware and software setup:

As far as hardware is concerned, the only requirement is a desktop or mobile device with a modern which has internet connectivity. As for software, the chosen device needs to have a modern browser (Google Chrome, Firefox, Microsoft Edge, etc.) which can handle HTML5 and ES6 Javascript.

Feature to be tested:

We are testing the comment section. On each article view page, there should be a comment section displayed underneath the article, along with a textarea to write a comment. The user should be able to write a comment on the desired article and see it in the comment section on page reload. Although, the comment section is not yet fully implemented.

Actual test cases:

1. After browsing the articles on the home page, the user clicks “Read More” and is brought to the article view page. On this page, all the associated comments, if any, should be displayed along with the commenters’ names underneath the article.
2. Starting from the article view page, the user decides to leave a comment. After typing out their comment in the textarea, they submit it. After a page reload, their comment should appear under the associated article.
3. Starting from this same article view page, the user returns to the home page and clicks on another article. On this different article view page, the comment the user just made should not appear underneath the article, as it is not associated with it.

Test #	Title	Description	Input	Expected Output	Result (Chrome)	Result (Firefox)
1	Viewing Comment Section	Testing that the comment section can be viewed	Click “Read More”	The article, along with the comments are displayed.	FAIL (saved in database but not displayed)	FAIL (saved in database but not displayed)
2	Posting Comment	Testing the functionality of the comment post	Type comment in textarea and Click	The new comment is viewable on page reload.	FAIL (saved in database but not displayed)	FAIL (saved in database but not displayed)

			“submit”			
3	Viewing different Comment Section	Testing that comments are properly associated	Click “Read More” on different article	Comments made on different articles are not viewable.	FAIL (saved in database but not displayed)	FAIL (saved in database but not displayed)

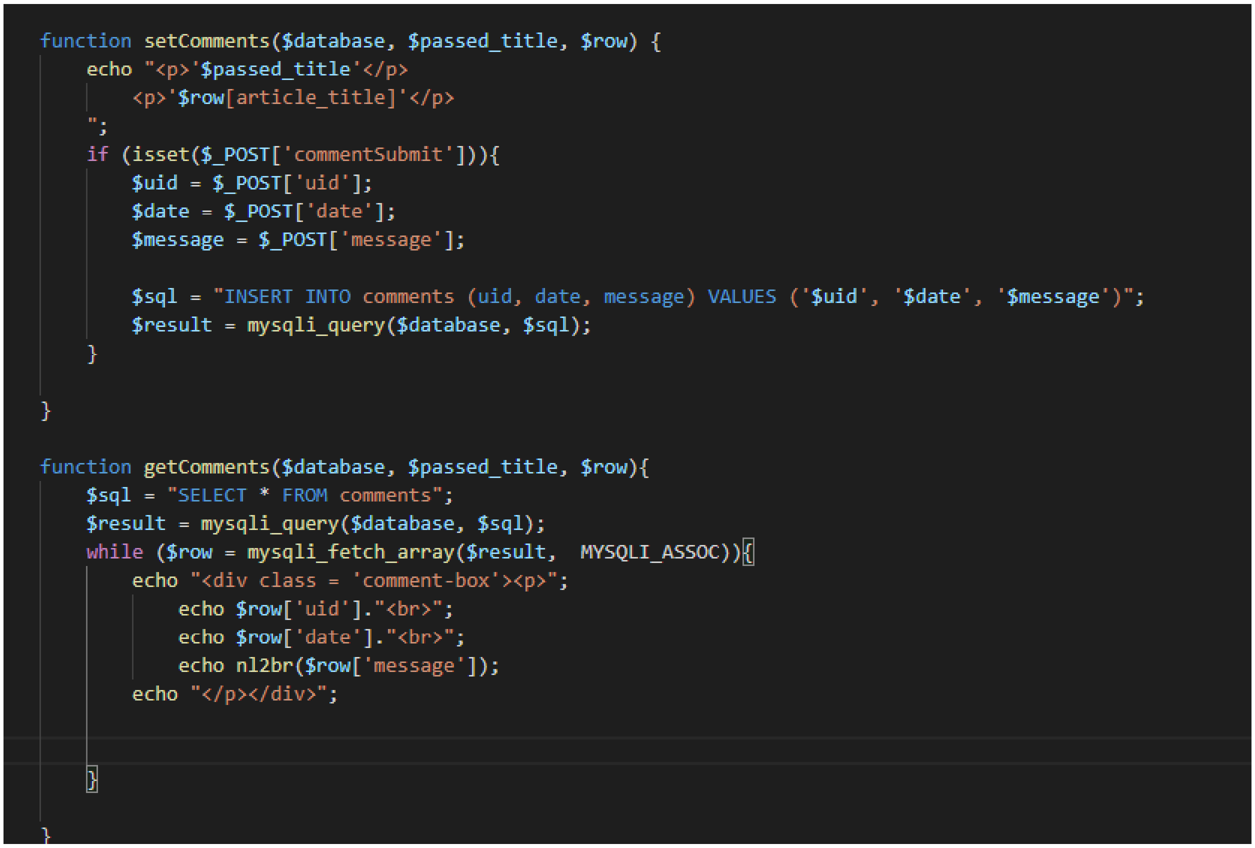
Self-check on best practices for security

The major assets that we are protecting are the username and password that the user provides when they create their account or are trying to login to the coronavirus system. The username is not encrypted when it is stored on the server; however, the password is encrypted. The password is encrypted using the PASSWORD_BCRYPT function that is defined in the php library. The encryption adds salt when hashing the password and the result is stored on the server. Because the username is set to “unique” on phpMyAdmin, if the user enters a username that has already been chosen by another user when creating an account, an error message is returned to the user. If the user enters a unique username and any password, a message is displayed that reads that the user has successfully created an account and a link is put on the screen that directs the user to the login page again to put their login information. When the user logs, in the username and password is retrieved when the user clicks on the submit button. If the user enters a username that does not match any usernames on the database, a message is displayed that the user has not entered a valid username. The password is compared to the one stored on the database by using the password_verify php function. The function compares a string to a salted, encrypted string. If there is a match, then the user is directed to the coronavirus service. If there is not a match, a message is displayed that indicates that the user has entered an incorrect password. On successful login, a session is created while they are logged in, and will expire after some time.

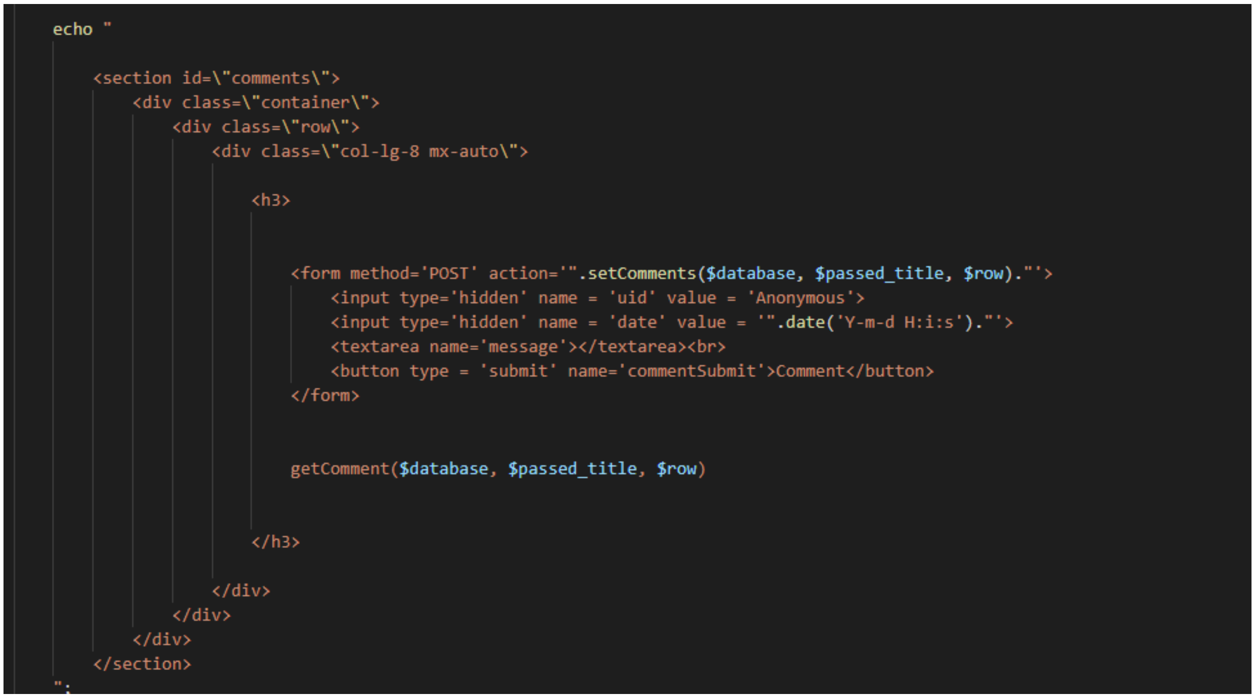
However, additional security measures must be taken to ensure that the user cannot hack into the system. For example, when users enter in information, such as " OR" into the login form in the password field, it is possible to bypass the account validation process and head straight into the coronavirus service. To circumvent this issue, user information when they are logging into or creating an account is tested before any validation or database storage takes place. This is done by stripping the user input of all slashes, special characters, and removing any white spaces in the user input. To do this, the trim, stripslashes, and htmlspecialchars functions are used to change user input, if necessary. This is done to ensure that our service is free from any script attacks or sql injection.

Code Review

Comment Section Code Example



```
function setComments($database, $passed_title, $row) {
    echo "<p>'$passed_title'</p>
        <p>'$row[article_title]'</p>
    ";
    if (isset($_POST['commentSubmit'])){
        $uid = $_POST['uid'];
        $date = $_POST['date'];
        $message = $_POST['message'];

        $sql = "INSERT INTO comments (uid, date, message) VALUES ('$uid', '$date', '$message')";
        $result = mysqli_query($database, $sql);
    }

}

function getComments($database, $passed_title, $row){
    $sql = "SELECT * FROM comments";
    $result = mysqli_query($database, $sql);
    while ($row = mysqli_fetch_array($result,  MYSQLI_ASSOC)){
        echo "<div class = 'comment-box'><p>";
            echo $row['uid']."<br>";
            echo $row['date']."<br>";
            echo nl2br($row['message']);
        echo "</p></div>";


    }

}
```



```
echo "

    <section id=\"comments\">
        <div class=\"container\">
            <div class=\"row\">
                <div class=\"col-lg-8 mx-auto\">

                    <h3>


                        <form method='POST' action='".setComments($database, $passed_title, $row)."'>
                            <input type='hidden' name = 'uid' value = 'Anonymous'>
                            <input type='hidden' name = 'date' value = '".date('Y-m-d H:i:s')."'>
                            <textarea name='message'></textarea><br>
                            <button type = 'submit' name='commentSubmit'>Comment</button>
                        </form>


                        getComment($database, $passed_title, $row)


                    </h3>

                </div>
            </div>
        </div>
    </section>
";
```

Self-check: Adherence to original Non-functional specs

Security - DONE: the coronavirus website will allow the user to create a username and password before being able to use our service. Upon creating an account, the password will then be encrypted before it is stored on a database. When the user logs in, the username and password will be compared to the database if it exists before allowing the user to enter the website. Security can be quantified by the percentage of users that do not have to change their username or password after account creation.

Usability - DONE: the website will straight-forward to a person to use. When the website opens, the user will be presented with a login and create account form. They will enter in information for the corresponding form they wish to fill out. When successfully logged in, the user will be taken to a main html page that will have a form with only one entry. In the text input form, the user will give a country name. When the submit button is clicked, coronavirus information for that country will be displayed. Underneath the coronavirus data, there will be recent news with hyperlinks for the user to read more on coronavirus news.

Manageability - DONE: The login portion of the website will be managed solely by phpMyAdmin to store username and password information. To retrieve coronavirus and recent news information the API's listed below will be used to request information.

Portability - DONE: In order to ensure that our software is portable, our website must work across multiple web browsers including: Google Chrome, Internet Explorer, Mozilla Firefox, Microsoft Edge, and Safari on desktop and mobile devices. Using Bootstrap to develop our website will help because the templates are created to prioritize mobile devices. In order to ensure our service works across all major web browsers, it will be imperative that we adhere to all HTML5 standards and not use any deprecated practices in web development.

Modifiability - DONE: With each modification of our website, the LAMP server will need to be updated with the changed files so that our website will be accessed with the latest changes and features. Also, each version of our website will need to be uploaded to github so that the changes can have pull requests to each developer on the team.

Performance - DONE: the performance of our website will depend mostly on response time to retrieve JSON data, parse it, and display it directly in an HTML file. Another performance aspect that can be measured will be the time it takes to create an account or validate login credentials by collecting user form data and inserting or comparing the information with username and passwords that are currently on the database.

Platform constraints - DONE: the account creating and login functionality will depend if there is a limit on the amount of usernames and passwords that can be stored in the accounts table on phpMyAdmin. There should be no platform constraints of any web browser that the user uses when on the website or if the user is using a desktop or a mobile device. Another platform constraint will be the use of the recent news API in our website. This is because to use the recent news API, the API only lets requests be made if the articles being requested are at most one month old.