CEN 4010 Principles of Software Engineering, Fall 2019

Milestone 4: Beta Launch and Final Product Reviews

Owl Gamers

Team number: 5

Monika Spasovska - Product Owner
mspasovska2017@fau.edu
Rolando Leiva - Scrum Master
rlevia2017@fau.edu
Stephanie Juan - Development Team
sjuan2017@fau.edu
Gandhi Lucate - Development Team
glucate2015@fau.edu

11/18/2019

2.2 Product Summary

1) Owl Gamers website featuring Captain Defender game
2) Committed functions:
    1. Game - The website will have a spinoff version of the Alien Invaders game, called Captain Defender. Every user on the platform that has created an account can play the game and track their score as they're playing.
    2. User profile - After storing user information in the database, the user will be given the opportunity to personalize their own profile. They can select their own avi picture and post a bio. Username will be also displayed on every users page.
    3. User friends - Users will be able to search for and add other users in the game. The user has to go to their profile page, click on the add friends icon and it will redirect them to a search page. Here they can type in a username, if valid, and add the user to their friend list.
    4. Score page - There will be a universal score page for top ranking players. Having a universal high score function will add a sense of competitiveness and connection between the players. It will also be a great talking point amongst friends.
    5. Home page - The home page will be in place for user interaction. Users can post statuses and see their friend's statuses.

3) The unique features of our website are user interactions. Not only did we create a new alien invaders game, with our own unique twist to it, now there is a capability to have user interactions. If you score high enough, you can be amongst the top 10 on the site! You can post whatever you want as a status, and keep up with other users on the platform.
4) https://lamp.cse.fau.edu/~cen4010fal19_g05/milestone3/login/
Once you create an account you can log in. There you will have a personal user profile page. You will be able to navigate to the game, search for other users or go to the home page. (all features are still not a 100% implemented)

2.3 Usability Test Plan

Test objectives for posting comments on homepage:

- When user x inputs text into the comments box and submits the text, user will be able to see their status on the home page.
- User will be able to add additional statuses and post to home page.

- Status/Statuses are displayed to every user that is friends with user x.
- All users that have friended each other will be able to see each others statuses posted on the home page.
- User can view other friend's statuses without having to post one

Test Plan for posting comments on homepage:

- Create a homepage.
- Create and include a comments box where all users can post status updates and comments on the homepage.
- The intended user is any Owl Gamer who has created an account will have access to the homepage where they can see other users' comments and post their own.
- Final design should be simple yet, welcoming and easy to navigate through.
- Users should be able to navigate to the homepage simply and begin posting status updates and public comments quickly.
- URL: "http://lamp.eng.fau.edu/~cen4010fal19_g05/milestone3/homepage/"

Usability Questionnaire:

1. Navigating the Owl Gaming homepage was easy:
    - Totally Agree
    - Agree
    - Neutral
    - Disagree
    - Strongly Disagree
2. Posting Comments on the homepage was easy:
    - Totally Agree
    - Agree
    - Neutral
    - Disagree
    - Strongly Disagree
3. Ability to see other users comments and status updates on homepage:
    - Yes
    - No
    - N/A
4. How satisfied were you with posting comments and navigating through the home page:
    - Completely Satisfied
    - Satisfied

- Neutral
- Not Satisfied
- Completely Unsatisfied

2.4 QA Test Plan

1) **Test Objectives:**
   - When user x inputs text into the comments box and submits the text, user will be able to see their status on the home page.
   - User will be able to add additional statuses and post to home page.
   - Status/Statuses are displayed to every user that is friends with user x.
   - All users that have friended each other will be able to see each others statuses posted on the home page.
   - User can view other friend's statuses without having to post one

2) **Hardware and Software setup:**
   No hardware needed for website.
   Software used is Cyber Duck, Lamp Server, Brackets and phpMyAdmin.
   Cyber Duck is an open source client for FTP AND SFTP, WebDAV, and cloud storage.
   Lamp is an archetypal model of web service stacks, named as an acronym of the names of its original four open-source components: the Linux operating system, the Apache HTTP Server, the MySQL relational database management system, and the PHP programming language.
   Brackets is a source code editor with a primary focus on web development.
   PhpMyAdmin is a free and open source administration tool for MySQL and MariaDB.

3) **Feature to be tested:**
   The feature tested is user statuses. User can post a status allowing his/her friends to view it. Statuses are in chronological order. Whatever status is posted last, that is the one that should be first on the home page. User has to type in the comment box and click submit in order to post status. User does not have to post status to view other statuses. If user visits home page a display of all statuses posted by friends should be available on the home page

4) Actual Test cases:

Test #	Test Title	Test description	Test input	Expected correct output	Test result	Browser
1	Status post	User goes to the home page. User types in the status box and presses submit and status is submitted.	User inputs text into comment box.	Status is submitted and stored in database.	True	Google Chrome
1	Status post	User goes to the home page. User types in the status box and presses submit and status is submitted.	User inputs text into comment box.	Status is submitted and stored in database.	True	Mozilla FireFox

Test #	Test Title	Test description	Test input	Expected correct output	Test result	Browser
2	Status post visible to users	Status is posted by user and appears on the home page	User inputs status in the comment box.	Status is visible to user and friends.	True	Google Chrome
2	Status post visible to others	Status is posted by user and appears on the home page	User inputs status in the comment box.	Status is visible to user and friends.	True	Mozilla FireFox

Test #	Test Title	Test description	Test input	Expected correct output	Test result	Browser
3	Viewing friends statuses	User does not have to post a status to view other statuses. User goes to the home page and can see friend's statuses.	Going to the home page.	Friend's statuses are visible to user	False	Google Chrome
3	Viewing friends statuses	User does not have to post a status to view other statuses. User goes to the home page and can see friend's statuses.	Going to the home page.	Friend's statuses are visible to user	False	Mozilla FireFox

2.5 Code Review

```
//PHP code to connect to the database
// a php and an html index is repetetive
//It might be better to have one file to connect to the database
<?php

$dbServerName = "lamp.cse.fau.edu";
$dbUsername = "cen4010fal19_g05 ";
$dbPassword = "tbynSnKO38";
$dbName = "cen4010fal19_g05";

$conn = mysqli_connect($dbServerName, $dbUsername, $dbPassword, $dbName);
?>

<html>
  <head>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/jquery/3.3.1/jquery.min.js">
    </script>
    <link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.c
ss">
//The css should be in another file not in main index file
<style>
.card {
  box-shadow: 0 4px 8px 0 rgba(0, 0, 0, 0.2);
  max-width: 150px;
  margin: auto;
  text-align: center;
  font-family: arial;
}

.title {
  color: grey;
  font-size: 18px;
}

button {
```

```
  border: none;
  outline: 0;
  display: inline-block;
  padding: 8px;
  color: white;
  background-color: black;
  text-align: center;
  cursor: pointer;
  width: 100%;
  font-size: 18px;
}

a {
  text-decoration: none;
  font-size: 22px;
  color: white;
}

button:hover, a:hover {
  opacity: 0.7;
}

ul {
  list-style-type: none;
  margin: 0;
  padding: 0;
  overflow: hidden;
  background-color: #333;
}

li {
  float: left;
}

li a {
  display: block;
  color: white;
  text-align: center;
  padding: 14px 16px;
```

```
  text-decoration: none;
}

li a:hover {
  background-color: #111;
}
      </style>
  </head>
  <body>

// toolbar change to match the rest of the website
<ul>
  <li><a class="http://lamp.eng.fau.edu/~cen4010fal19_g05/milestone3/homepage/"
href="#home">Home</a></li>
  <li><a href="../game">Game</a></li>
  <li><a href="../scores">High Scores</a></li>
  <li><a href="../profile">Your Profile</a></li>
  <li><a href="../login">Log Out</a></li>
</ul>

// change up source code to match our website
<div class="card">
  <img src="img/avatar.png" alt="John" style="width:100%">
  <h1>Username</h1>
  <p class="title">Bio..</p>
  <p>Florida Atlantic University</p>
  <p><button class="btn"><a target="_blank"
href="http://lamp.eng.fau.edu/~cen4010fal19_g05/milestone3/profile/">Visit
Profile</a></button></p>
</div>

    <div class="info">
      <label for="example">Post a status
      </label>
      <input id="example" type="text"
            name="Ntext" id="Ntext" size="20"value="Status">
     <input type="submit" name="submit" id="submit" class="form-submit submit"
value="Submit"/>
```

```
<?php
        echo "Entering php";
        if(isset($_POST['submit']))
        {

                $status = $_POST['Ntext'];
                $sql = "INSERT INTO Status(Stat, Email) VALUES (?,?)";
                $stmt = mysqli_stmt_init($conn);
                $result = mysqli_query($conn, $sql);
                mysqli_stmt_prepare($stmt,$sql);
      mysqli_stmt_bind_param($stmt,"ss",$status,$_SESSION['userID']);
                mysqli_stmt_execute($stmt);



        }
?>
 </div>
   <p id="para">
   </p>
 </body>
   <script>
      window.onclick = function(e)
      {   var id =  e.target.id;
          if (id === 'sent')
          { var txtbox = document.getElementById('example');
           var txt = txtbox.value;
           $( "#para" ).append( txt + '<br>');
           $( txtbox ).val('');
          }

      }
      var a = [];
      a.push(txt);

   </script>

</html>
```

2.6 Self-check on best practices for security

The passwords in the database are encrypted to make sure that the user's information is protected and that a hacker can not access the information. When the user enters the information during the sign-up page, the password that is entered is encrypted by using the hash function in PHP. The function creates a string with a length of 60 to ensure that the encrypted password can not be guessed by someone trying to steal the user's information. When the user enters their login credentials on the log in page, the password that the user has entered will be one parameter of the PHP function password verify. With this function, the password that the user is compared with the password stored in the database. Since the password in the database is encrypted and the password the user entered is not encrypted, the function will encrypt the password and check to see if the password matches the password stored in the database. If the passwords match then the user is taken the home page. The password is the most important piece of information stored in the database so the password is the only information that is encrypted. The search bar will be validated through the database to make sure the information the user entered is actually in the database. If there is a match then the results will be displayed to the user on the website.

2.7 Check on non-functional specs

1. **ON TRACK:** Usability - The Owl Gaming website shall be a gaming platform that welcomes new users and shall be easy to navigate through.

   1.1 Once users get a look and feel for the platform they can easily begin playing the game, seeing their high score on their profiles, and be able to view the Top 10 Scores of the game amongst all the players of Captain Defender.
2. **ON TRACK:** Performance - The Owl Gaming websites response time should be 2 seconds or less starting with the login and applies to viewing each individual updated high score and viewing the Top 10 players scores.

2.1 The website shall be able to support up to 50 players on the server at any given time to be able to maintain response time.

3. **ON TRACK:** Security - The Owl Gaming website shall not execute commands embedded in data provided by users that force the application to manipulate the database tables in unintended ways.

   3.1 No user shall be able to view any other users' personal information unless posted publicly by the user.

4. **DONE:** Maintainability - The Owl Gaming website and the game shall have a 90% probability that any errors encountered will be repaired within an hour.

   4.1 The coding shall be simple enough that future maintenance can be edited quickly and efficiently, which will also make the coding simple and adaptable to any new requirements.

5. **ON TRACK:** Accessibility - The Owl Gaming website users shall be able to access their profile and update their information and icons at any time.

   5.1 No other users aside from yourself shall be able to change or view any of your personal information that is not public.

6. **DONE:** Data Storage - The Owl Gaming website shall store any and all personal information input to the website into a database such as a Name, and Email.

   6.1 Individual players' high scores shall be publicly viewable on their profiles.

7. **DONE:** Availability - The Owl Gaming website and game Captain Defender shall be available to any user who has created an account at any time in the day.

Peer Evaluation:

Monika Spasovska:	37.5/25
Rolando Leiva:	37.5/25
Stephanie Juan:	25/25
Gandhi Lucate:	0/25