CEN 4010 Principles of Software Engineering

Milestone 4 Beta Launch and Reviews

**CoreShare**

Core5

Team Number: 25

November 9, 2021

**Team Members**

Scrum Master: Mac Consolabe

mconsolabe2020@fau.edu

-

Product Owner: Greydi Lora

Glora2019@fau.edu

-

GitHub Master: Dominic Robbins

Drobbins2019@fau.edu

-

Development Team: Victor Marinho Correa

jmarinhocorrea2021@fau.edu

-

Development Team: Michael Cevallos

Mcevallos2020@fau.edu

**History Revision Table**

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Revision | Actions | Performed by |
| 2021-10-19 | 1 | Made changes according to TA feedback | Group 25 |
| 2021-10-23 | 2 | Added diagrams | Group 25 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Contents

[Product Summary 4](#_Toc87726689)

[Usability Test Plan 6](#_Toc87726690)

[QA Test Plan 6](#_Toc87726691)

[Code Review 6](#_Toc87726692)

[Best practices for security 6](#_Toc87726693)

[Non-functional requirements 6](#_Toc87726694)

# Product Summary

CoreShare is a social media web application that will be geared towards connecting people across the world that are far away from each other, to remind them, and help them reminisce on old memories, with the ability to create new ones. The purpose of this application is to allow users the ability to connect with their loved ones by just sharing a song. All the experience, connection, and sharing will be made inside the platform, without having to leave and grab a song somewhere else and come back. Everything will be within your grasp, your CoreCircle and music, all in the same place. Connection, love, harmony is the fuel for the application, but giving you a good user experience is also in our top priority.

**CoreShare**

1. **Create an Account** – Users will be able to create an account using their first name, last name, password, email, birth date, and gender. This allows the user to access all the available features provided accessible to users with an account.
2. **Create a post** – Users can create posts to the feed.
3. **Search for music** – Users can search for music using their Spotify account right on the application.

**Unique Features** – The ability to create a post, search for a song from Spotify, search for other users, and visit their profile page is available for users on the application. Users can also send friend requests to other users and create posts to be private to friends only or public for everyone.

<https://lamp.cse.fau.edu/~cen4010_fa21_g25/login/index.php>

# Usability Test Plan

Test Objective

The objective is to test the usability of the posting feature in the application. This should allow users to post a text. After the user creates an account, they will have access to this portion of the application’s feature. The create post feature should be easily understandable to anyone that uses the application. The user interface is designed to duplicate most social media applications in the best and simplest form. This can help decrease the time it takes to use the feature. The create post feature includes a text box where the user can type a message of their choice to share with other users. After the user creates a post, it will be shown on the feed for others to view. The user has the option to select whether they want their post to be private or public by checking the ‘Public’ box.

Test Plan

The starting point will be when the user first arrives on the website and creates an account. They will sign up using their first name, last name, password, email, birth date, and gender. Their information will be stored in a database After the user clicks on the ‘Create Account’ button, they will be sent to the main page where the feed and ‘Make Post’ feature is displayed. The user will create a text post and click the ‘Post’ button to add the text to the feed. The text should be displayed on the feed which is located under the ‘Make Post’ feature. The ideal user for this application will include anyone knowledgeable of using any electronic device. The goal is for the user to create a post without any roadblocks and confusion.

URL of system to be tested: <https://lamp.cse.fau.edu/~cen4010_fa21_g25>

Questionnaire

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Strongly Disagree | Somewhat Disagree | Neutral | Somewhat Agree | Strongly Agree |
| The application is easy to understand and navigate. |  |  |  |  | x |
| After creating an account, the homepage is displayed. |  |  |  |  | x |
| The ‘Make Post’ feature allows you to type in a message. |  |  |  |  | x |
| When user selects the ‘Public’ checkmark box, their post is listed stating that it’s public. |  |  |  |  | x |
| When user leaves the ‘Public’ box unchecked, their post is listed stating that it’s private. |  |  |  |  | x |
| The post created is displayed on the feed. |  |  |  |  | x |
| The application ran without crashing. |  |  |  |  | x |
| Creating a post was responsive and quick. |  |  |  |  | x |

# QA Test Plan

Test objective

The objective of this application is to test the basic functionalities that allow the user to successfully use the features without any problems or errors. We will test the functionality of the posting feature that will display on the main feed of the application. The speed of the post creation and ease of use will be observed while testing. The user should be able to type in a message and select whether they want their post to be public or private without any errors.

Hardware and software setup

The hardware that the application will be tested on will be a desktop or laptop. The web browsers we will use to test the application will be on Chrome and Firefox. Visual Studios was used to write the code as well as phpMyAdmin, and lamp server.

Feature to be tested

* Creating an account for a new user.
* Typing mismatched passwords. Message will display and prevent user from proceeding with the registration.
* Signing in after creating an account and then logging out.
* Writing a message on the text box and posting it so it will display on the feed.
* The users’ name will be displayed with their post.
* Shows if the post is private or public.

Actual test cases

|  |  |
| --- | --- |
| Test # | 1 |
| Title | Create an account |
| Description | User will create an account so they can access the application and have their posts saved in their profile page. They can sign back in after logging out. |
| Input | First name, last name, password, email, birth date, and gender. |
| Expected Output | An account will be created for the user and stored in the database. |
| Results - Chrome | PASS – Account was created and stored in the database. |
| Results - Firefox | PASS - Account was created and stored in the database. |

|  |  |
| --- | --- |
| Test # | 2 |
| Title | Create a post with text |
| Description | After creating an account, user will type a message in the ‘Make Post’ feature. |
| Input | User will input any alphanumerical text into the textbox and select the post button. |
| Expected Output | The written message will be displayed on the feed underneath the ‘Make Post’ box. |
| Results - Chrome | PASS - Message is displayed on the feed. |
| Results - Firefox | PASS - Message is displayed on the feed. |

|  |  |
| --- | --- |
| Test # | 3 |
| Title | Create a private post |
| Description | User should be able to make their post private. |
| Input | User will write a message in the ‘Make Post’ feature and select the ‘Public’ checkmark box to make the post private. |
| Expected Output | The message that was entered is posted and states that it is private. |
| Results - Chrome | PASS - Post is private |
| Results - Firefox | PASS – Post is private |

# Code Review

For the coding style, PHP was used to create the search function for the application.

Text

Description automatically generated

Graphical user interface, text, application, email, Teams

Description automatically generated

# Best practices for security

The major assets we are protecting are the users’ personal email and password. When the user signs up for an account or logs in, the password is covered so it is not displayed to the screen. If the user inputs an incorrect password while signing in, they are not able to access the account until they get the correct password. A message will be displayed to inform the user that the password is incorrect. The password and email are encrypted and stored in the database and will not be displayed on the users’ profile or feed. We also validated the search bar and made sure we can search for other users. Php was used to create these functionalities.

# Non-functional requirements

Performance Requirements:

1. Availability: The very first requirement is that the service should be highly available. The users should be able to send songs and should be able to see other users’ songs posted on their home timeline without any downtime. ON TRACK
2. Execution Speed: Generating the timeline should be fast within half a second (at most) DONE
3. Consistency: The system does not need to be strongly consistent--eventual consistency is fine. DONE
4. Scalability: The system should be scalable with the increasing data load presented on the backend. DONE
5. Durability: User data (user information and song post/ and friends) should be durable. DONE

YouTube URL of Demo: <https://www.youtube.com/watch?v=6BFgofeKXXg>