

WAPH-Web Application Programming and Hacking

Instructor: Dr. Phu Phung

Mini Facebook

Team – 01

Team members

1. Tulasiram Nakkanaboina, nakkantm@mail.uc.edu
2. Vihasith Rasala, rasalavh@mail.uc.edu
3. Grahika Rampudi, Rampudga@mail.uc.edu
4. Sai Sandeep Pasham, pashamsp@mail.uc.edu

Team members respective headshots



Project Management Information

Source code repository (private access): [Click here](#)

Project homepage (public): [Click here](#)

Revision History

Date	Version	Description
26/03/2024	0.0	Initial draft
31/03/2024	0.1	Drafting template overview
01/04/2024	1.0	Sprint 1 update

Overview

The primary focus of the WAPH project revolves around creating an all-encompassing web application named "MiniFacebook." The development process of this web application incorporates comprehensive full-stack web development techniques and cybersecurity principles to ensure its resilience against potential vulnerabilities exploited by hackers. Our team, comprised of four classmates, collaboratively worked on the website's development. We primarily utilized WhatsApp for communication, supplemented by in-person meetings for

discussions. Additionally, we employed collaborative note-taking tools accessible to all team members, facilitating project discussions, idea generation, and progress tracking.

System Analysis

During the initial phase that is for the sprint 0 of the project, we constructed the website utilizing the system configuration file, which was utilized and modified for the WAPH-team project file. The hosts file in the etc folder was updated with the public URL: waph-team01.minifacebook.com alongside the IP address.

Demo screenshots

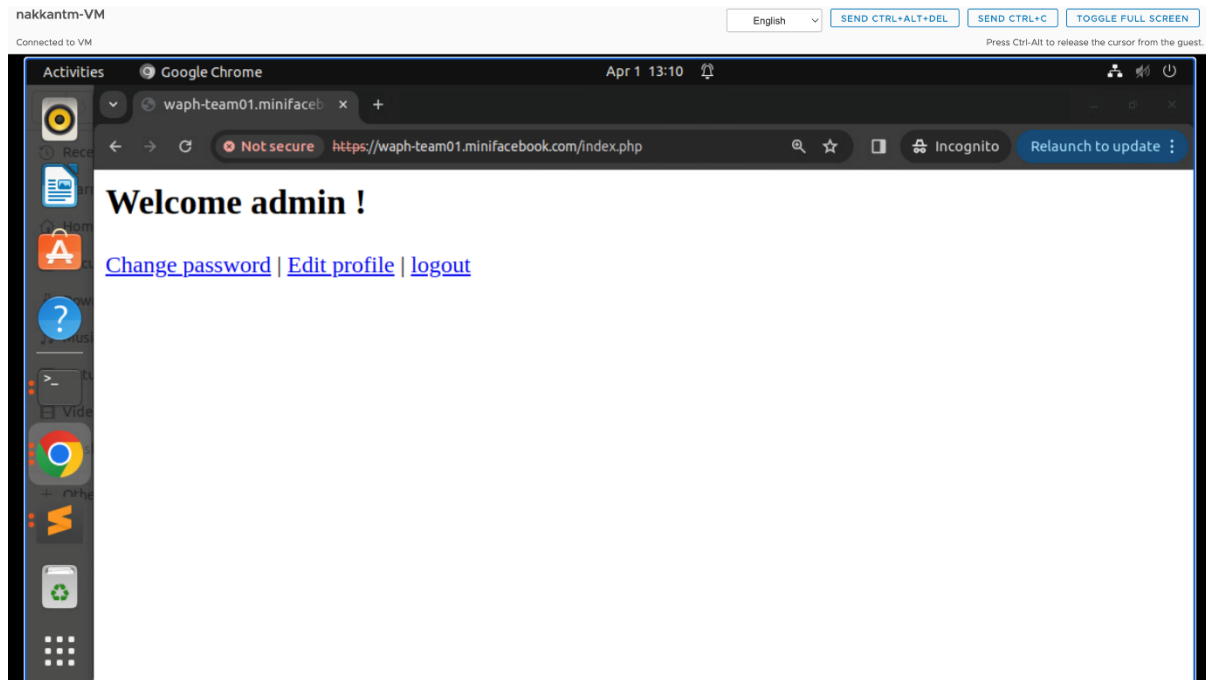


Figure 1 login admin page

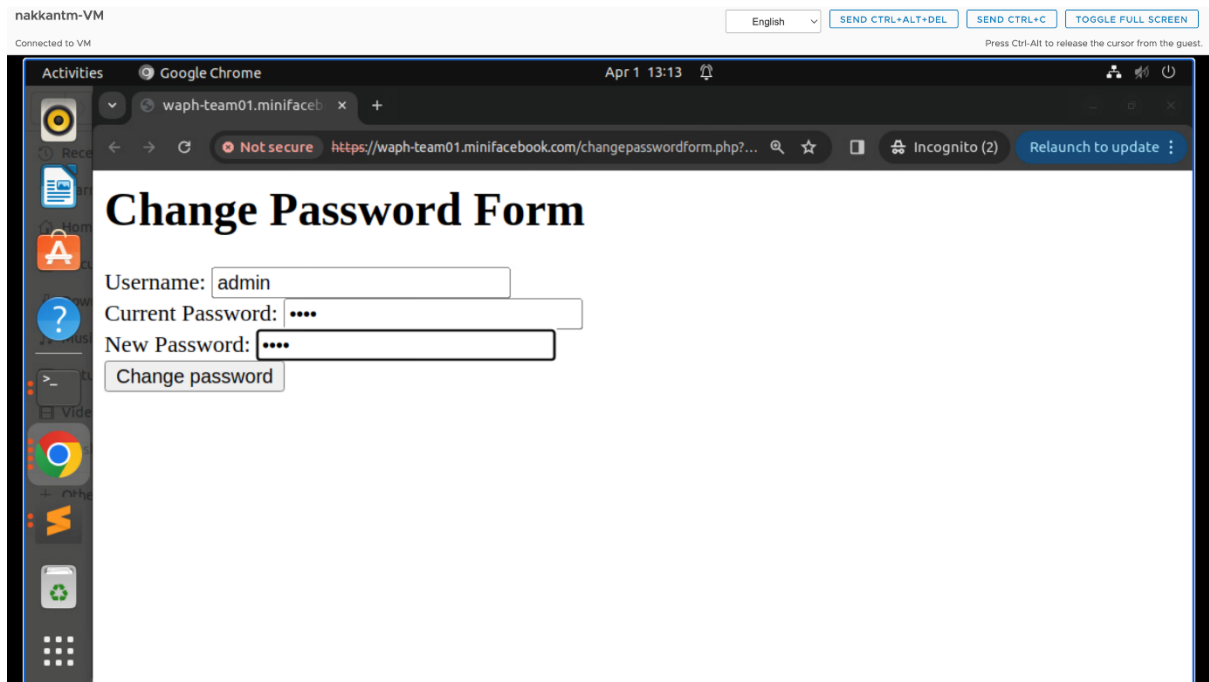


Figure 2 change password form

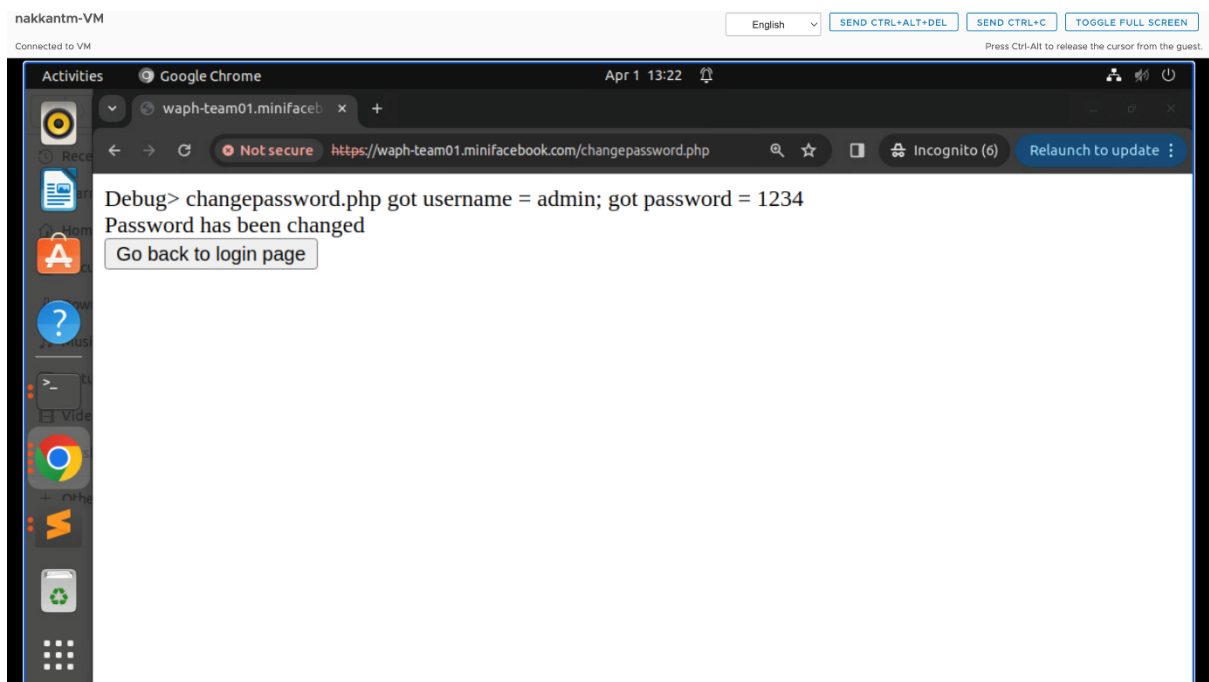


Figure 3 password changed successfully

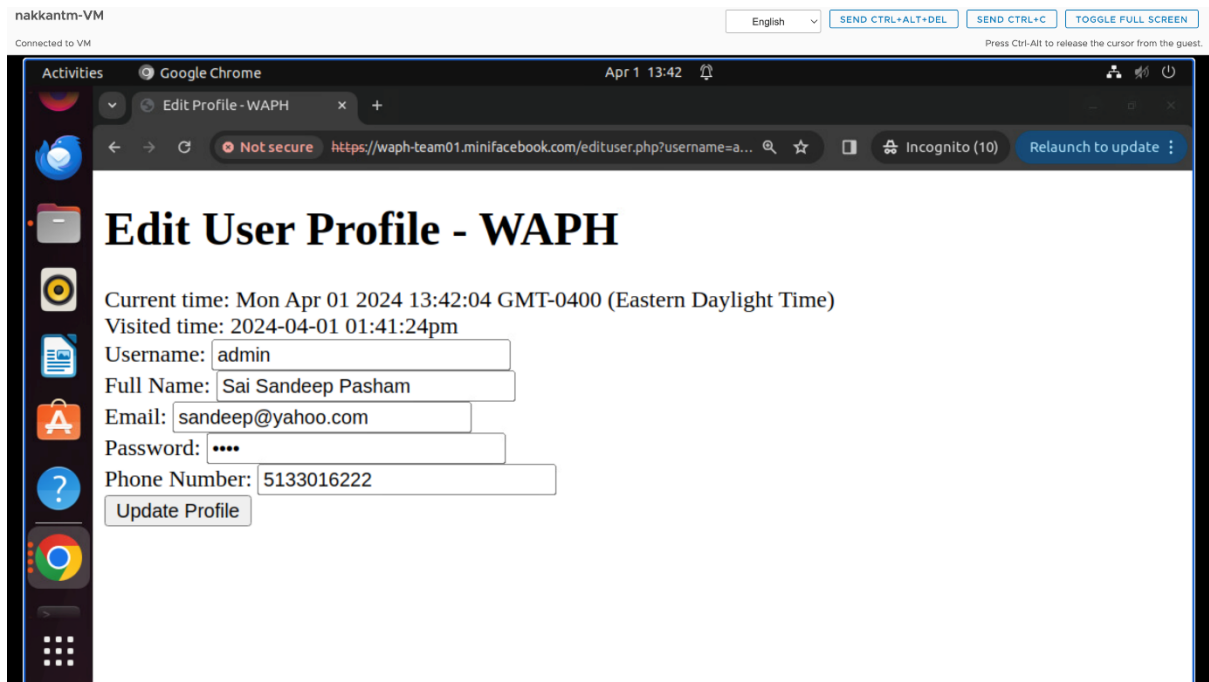


Figure 4 edit user profile

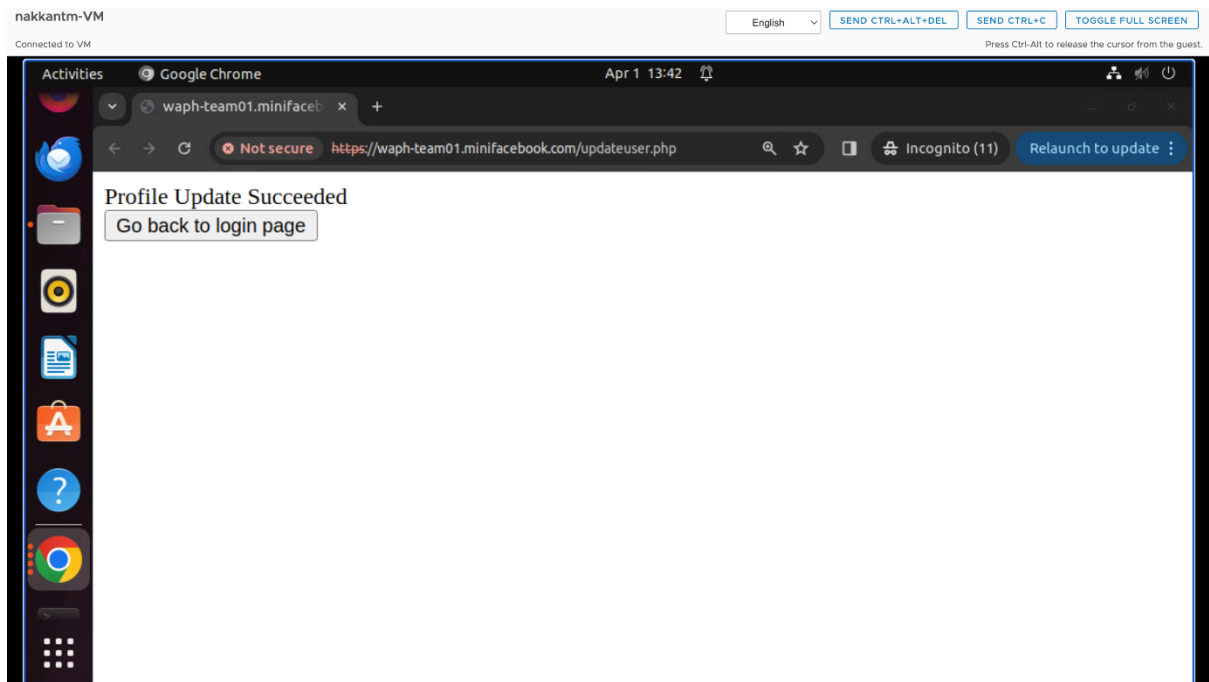


Figure 5 update profile

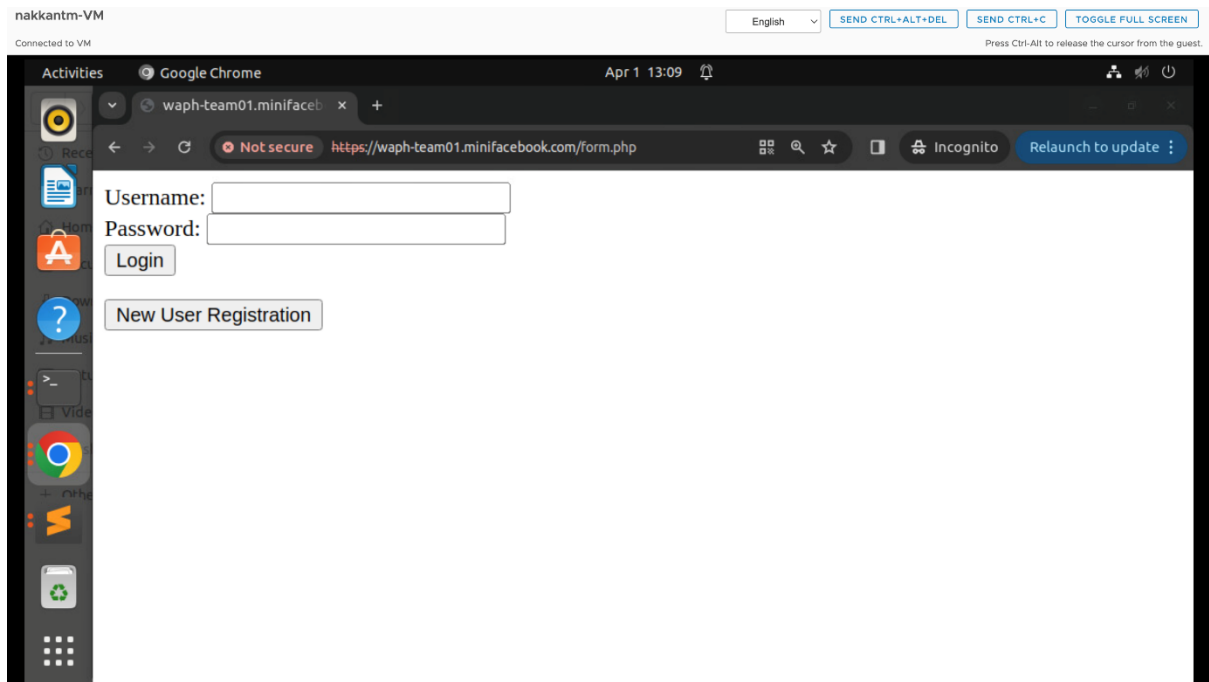


Figure 6 login page

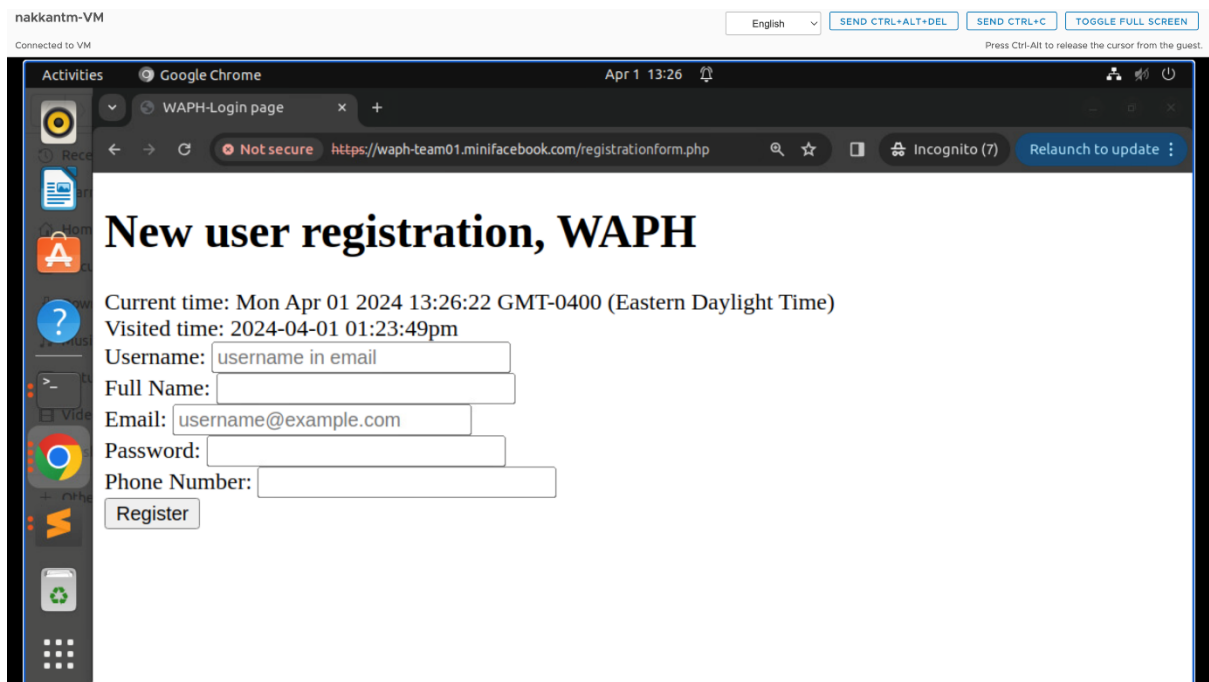


Figure 7 new user registration form

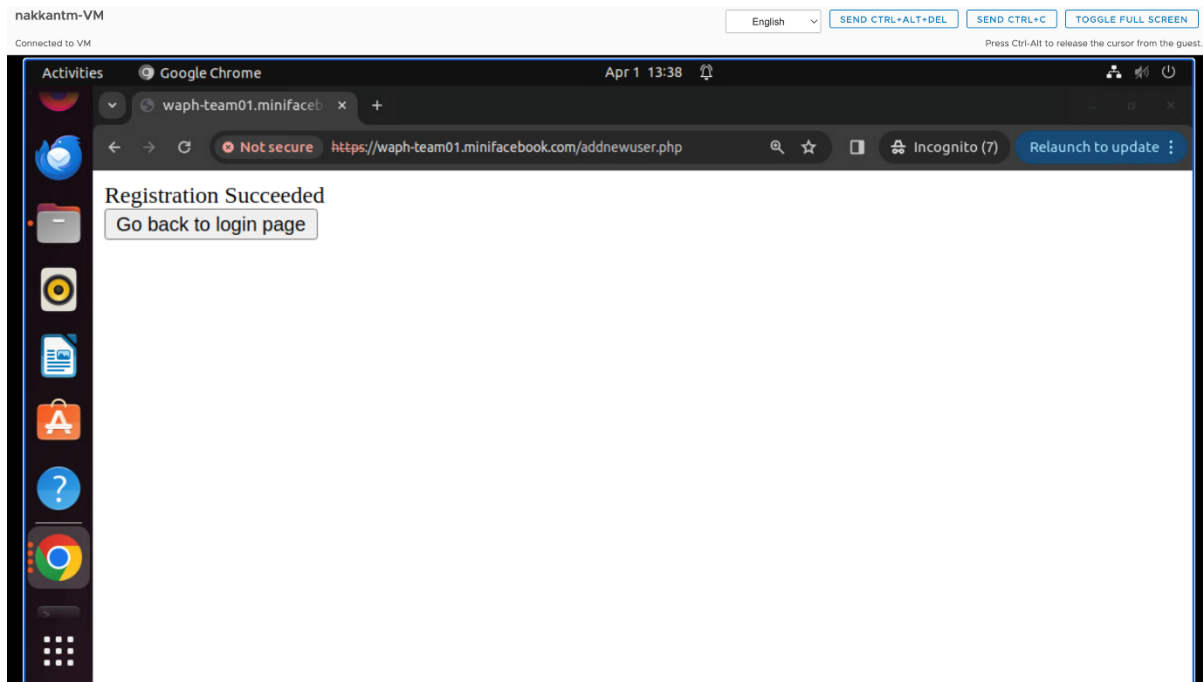


Figure 8 succesful registation form

Functional Requirements

- **Features for posting like images and texts:** Enabling users to create and share posts with images/texts.
- **Database setup:** Configuring database system for storing application data securely.
- **Notification add on:** Implementing notification system for relevant user alerts.
- **Forms for authentication and user registration:** Implementing secure login and registration forms for user access.
- **Search functionality:** Enabling users to search for specific content within application.
- **Messaging features:** Integrating private messaging functionalities for user communication.

Non - Functional Requirements

- **Error handling:** Managing errors for smoother user experience and application stability.
- **Implementing security features:** Strengthening application security through implementation of various security measures.
- **Cookies:** Utilizing cookies for managing session information and enhancing user experience.
- **Session information:** Handling session data to maintain user context and interactions.

Database

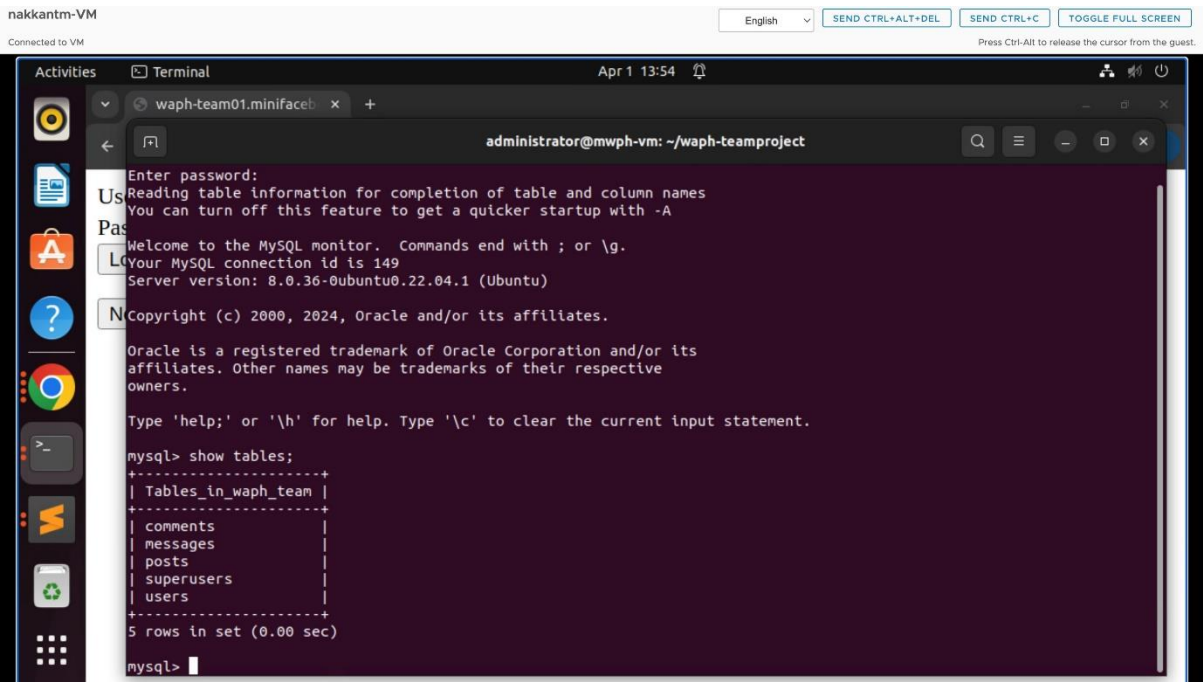


Figure 9 tables which stores the data

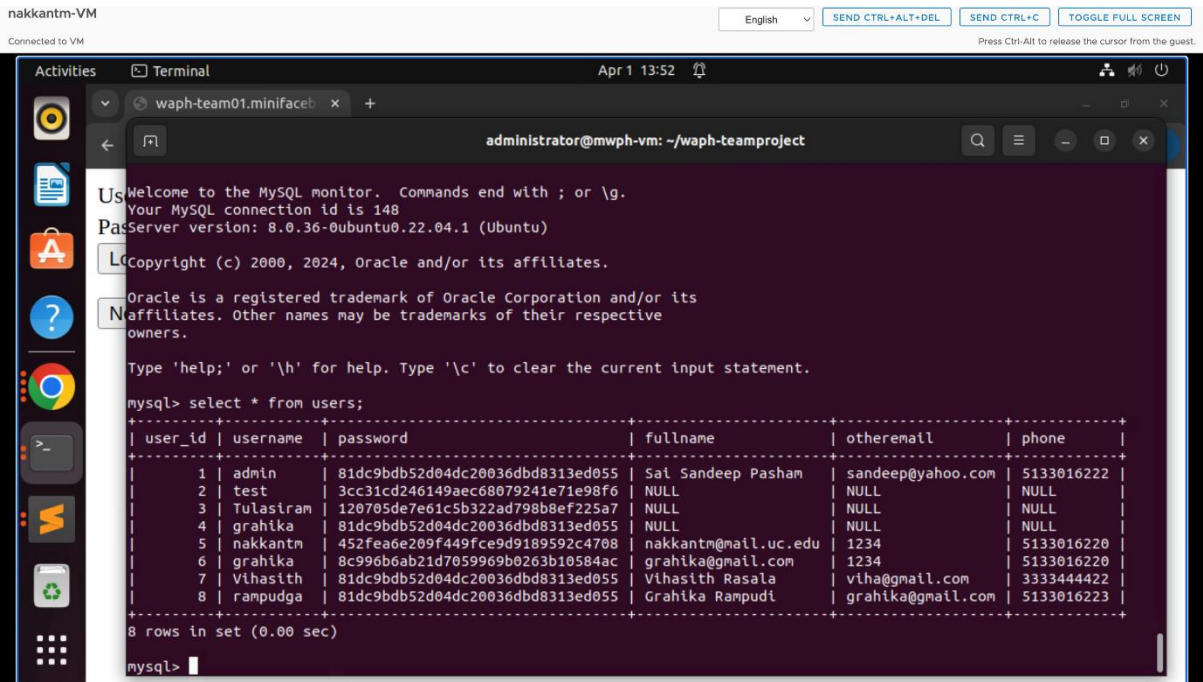


Figure 10 user tables

User Interface

Front-End:

Edit Profile Feature: Users can modify their full name, email, and phone number.

Change Password Option: Users can change their password if facing any issues.

Posts Visibility Setting: Upon logging in, users can view site's posted content.

Back-End:

Database Creation: Established a database with essential tables for the application.

Users Table: Stores user information including full name, username, password, email, and phone number.

Posts Table: Records posts from various users, containing fields like Post ID, Title, Content, Post Date, and Owner.

Database Integration: Connected the database with the front end to save user inputs.

Implementation

Security Analysis

Scrum Process

Sprint 0

Duration: 22/03/2024-03/31/2024

Completed Tasks:

1. Our team has successfully established both public and private repositories named "waph-teamproject" and "waph-team01.github.io" correspondingly.
2. SSL certificates and keys for the team project have been generated.
3. HTTPS has been configured within the local domain name to ensure secure communication.
4. A database has been set up for the system.
5. Individual home pages have been developed for all team members, and their respective lab files have been transferred to the team directory.
6. Testing of the index.html page has been carried out, yielding the anticipated performance and functionality as observed during the tests.

Contributions:

1. Tulasiram Nakkanaboina, 11 commits, 7 hours, contributed in generating

the SSL keys and certificates.

2. Vihasith Rasala, 8 commits, 6 hours, contributed in documentation of README file and organizing the data.

5

3. Grahika Rampudi, 8 commits, 7 hours, contributed in creating the personal page of our team which includes details of our team members.

4. Sai Sandeep Pasham, 7 commits, 6 hours, contributed in creating the database-data.sql, index.html page which includes the details of admin login.

Sprint 1

Duration: 03/28/2024-04/01/2024

Completed Tasks:

1. Established the database according to the outlined criteria.
2. Developed both login and registration forms.
3. Executed fundamental features for authenticated users, such as password modification.

Contributions:

1. Tulasiram Nakkanaboina, 11 commits, 7 hours, contributed in creating user registration and profile edit php files and further developed index.php file

2. Vihasith Rasala, 15 commits, 6 hours, contributed in documentation of README file and organizing the data. Created and developed logout php file.

3. Grahika Rampudi, 8 commits, 7 hours, contributed in creating and organizing the database and its implementation in back end of web application.

4. Sai Sandeep Pasham, 7 commits, 6 hours, contributed in creating the change password and login php files.

Spring Retrospection

- The main problem we faced for a team call is unmatched schedule of team members. We are usually doing our team meetings in late nights.
- Online calls didn't meet the expected progress of the project, so we are making ourselves available for physical sessions which is difficult.
- Proper collaborative dashboards require premium subscriptions for third party applications.
- Sandbox doesn't work properly which is a dot in a white page.
- Team members use different OS like MacOS or windows where it's difficult to work in virtual machine setup if it's a Macbook.

Appendix

Form.php:

waph-teamproject / form.php

nakkantm-uc Update form.php

Code

Blame

10 lines (10 loc) · 414 Bytes

Your organization can pay for GitHub Copilot

```
1  <form action="index.php" method="POST" class="form login">
2      Username: <input type="text" class="text_field" name="username" /><br>
3      Password: <input type="password" class="text_field" name="password" /><br>
4      <button class="button" type="submit">
5          Login
6      </button>
7  </form>
8  <button class="button" onclick="window.location.href='registrationform.php';">
9      New User Registration
10 </button>
```

Index.php:

nakkantm-uc Update index.php

e0eccf4 · 9 hours ago History

Code

Blame

37 lines (33 loc) · 1.37 KB

Your organization can pay for GitHub Copilot

Raw

Download

Copy

Share

```
1  <?php
2  session_set_cookie_params(15*60, "/", "waph-team01.mini.facebook.com", TRUE, TRUE);
3  session_start();
4  require "database.php";
5
6  if (isset($_POST["username"]) && isset($_POST["password"])) {
7      $username = $_POST["username"];
8      $password = $_POST["password"];
9
10     if (checklogin_mysql($username, $password)) {
11         $_SESSION["authenticated"] = TRUE;
12         $_SESSION["username"] = $username;
13         $_SESSION["browser"] = $_SERVER["HTTP_USER_AGENT"];
14     } else {
15         session_destroy();
16         echo "<script>alert('Invalid Username or password please recheck');window.location='form.php';</script>";
17         die();
18     }
19 }
20
21 if (!isset($_SESSION["authenticated"]) || $_SESSION["authenticated"] != TRUE) {
22     session_destroy();
23     echo "<script>alert('You have not logged in. Please login first');</script>";
24     header("Refresh:0; url=form.php");
25     die();
26 }
27
28 if ($_SESSION["browser"] != $_SERVER["HTTP_USER_AGENT"]) {
29     session_destroy();
30     echo "<script>alert('Session hijack detected');</script>";
31     header("Refresh:0; url=form.php");
32     die();
33 }
34 ?>
35 <h2> Welcome <?php echo htmlentities($_SESSION['username']); ?> !</h2>
36 <a href="changepasswordform.php?username=<?php echo urlencode($_SESSION['username']); ?>">Change password</a> | <a href="edituser.php?username=<?php echo urlencode($_SESSION['username']); ?>">Edit user</a>
37 </a>
```

Changepasswordform.php:

waph-teamproject / changepasswordform.php

 nakkantm-uc Update changepasswordform.php

1db3196 · 10 hours

CodeBlame6 lines (6 loc) · 497 Bytes


Your organization can pay for GitHub Copilot

Raw

```
1 <h1>Change Password Form</h1>
2 <form action="changepassword.php" method="POST" class="form login">
3 Username: <input type="text" class="text_field" name="username" value="<?php echo isset($_GET['username']) ? htmlentities($_GET['username']) : ''; ?>" readonly /> <br>
4 Current Password: <input type="password" class="text_field" name="password" /> <br>
5 New Password: <input type="password" class="text_field" name="newpassword" /> <br>
6 <button class="button" type="submit"> Change password </button>
```

Edituser.php:

waph-teamproject / edituser.php

 nakkantm-uc Update edituser.php

0.

CodeBlame47 lines (44 loc) · 1.82 KB

Your organization can pay for GitHub Copilot

Raw

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4 <meta charset="utf-8">
5 <title>Edit Profile - WAPH</title>
6 <script type="text/javascript">
7     function displayTime() {
8         document.getElementById('digit-clock').innerHTML = "Current time: " + new Date();
9     }
10    setInterval(displayTime, 500);
11
12    function validateForm() {
13        var email = document.forms["editProfileForm"]["email"].value;
14        var phone = document.forms["editProfileForm"]["phone"].value;
15
16        // Email validation
17        var emailPattern = /^[\\w.-]+@[\\w.-]+(\\.[\\w.-]+)*$/;
18        if (!emailPattern.test(email)) {
19            alert("Please enter a valid email address");
20            return false;
21        }
22
23        // Phone number validation
24        var phonePattern = /^[\\d]{10}$/;
25        if (!phonePattern.test(phone)) {
26            alert("Please enter a valid 10-digit phone number");
27            return false;
28        }
29    }
30 </script>
<body>
<h1>Edit User Profile - WAPH</h1>
<div id="digit-clock"></div>
<?php
echo "Visited time: " . date("Y-m-d h:i:sa");
?>
<form name="editProfileForm" action="updateuser.php" method="POST" class="form edit-profile" onsubmit="return validateForm();">
Username: <input type="text" class="text_field" name="username" value="<?php echo isset($_GET['username']) ? htmlentities($_GET['username']) : ''; ?>" readonly /> <br>
Full Name: <input type="text" class="text_field" name="fullName" required><br>
Email: <input type="email" class="text_field" name="email" required placeholder="username@example.com"><br>
Password: <input type="password" class="text_field" name="password" required><br>
Phone Number: <input type="tel" class="text_field" name="phoneNumber" required pattern="[0-9]{10}" title="Please enter a 10-digit phone number"><br>
<button class="button" type="submit">Update Profile</button>
</form>
</body>
</html>
```

registrationform.php:

waph-teamproject / registrationform.php

nakkantm-uc Update registrationform.php

6b8966a · 20 hours ago

Code Blame 51 lines (48 loc) · 1.94 KB

Your organization can pay for GitHub Copilot

Raw

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4 <meta charset="utf-8">
5 <title>WAPH-Login page</title>
6 <script type="text/javascript">
7     function displayTime() {
8         document.getElementById('digit-clock').innerHTML = "Current time: " + new Date();
9     }
10    setInterval(displayTime, 500);
11
12    function validateForm() {
13        var email = document.forms["registrationForm"]["email"].value;
14        var phone = document.forms["registrationForm"]["phone"].value;
15
16        // Email validation
17        var emailPattern = /^[\\w.-]+@[\\w-]+(\\.\\w-)+*$/;
18        if (!emailPattern.test(email)) {
19            alert("Please enter a valid email address");
20            return false;
21        }
22
23        // Phone number validation
24        var phonePattern = /^[0-9]{10}$/;
25        if (!phonePattern.test(phoneNumber)) {
26            alert("Please enter a valid 10-digit phone number");
27            return false;
28        }
29    }
30 </script>
31 </head>
32 <body>
33 <h1>New user registration, WAPH</h1>
34 <div id="digit-clock"></div>
35 <?php
36
37 echo "Visited time: " . date("Y-m-d h:i:sa");
38 ?>
39
40 <form name="registrationForm" action="addnewuser.php" method="POST" class="form login" onsubmit="return validateForm();">
41     Username: <input type="text" class="text_field" name="username" required
42     pattern="^[\\w.-]+@[\\w-]+(\\.\\w-)+*$"
43     title="Email address is required as username"
44     placeholder="username in email"
45     onchange="this.setCustomerValidity(this.validity.patternMismatch?this.title: '');"/> <br>
46     Full Name: <input type="text" class="text_field" name="fullName" required><br>
47     Email: <input type="email" class="text_field" name="email" required placeholder="username@example.com"><br>
48     Password: <input type="password" class="text_field" name="password" required><br>
49     Phone Number: <input type="tel" class="text_field" name="phoneNumber" required pattern="[0-9]{10}" title="Please enter a 10-digit phone number"><br>
50     <button class="button" type="submit">Register</button>
51 </form>
52 </body>
53 </html>
```