

Thomas Hoerger

Project 2

3/27/2025

1. Create a Login Page:

```
tehergs@csci-411-linux:~$ sudo nano /var/www/html/login.php
```

2. Write Login Page Code:

```
GNU nano 4.8 /var/www/html/login.php
<!DOCTYPE html>
<html>
<head>
  <title>Skyview Login</title>
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <style>
    body {
      font-family: Arial, sans-serif;
      background: #e0f7fa;
      margin: 0;
      padding: 0;
    }

    .login-box {
      max-width: 400px;
      margin: 10vh auto;
      background: white;
      padding: 20px;
      border-radius: 8px;
      box-shadow: 0px 4px 10px rgba(0,0,0,0.1);
    }

    input, button {
      width: 100%;
      padding: 10px;
      margin: 8px 0;
      font-size: 16px;
      border-radius: 5px;
      border: 1px solid #ccc;
    }

    button {
      background-color: #007BFF;
      color: white;
      border: none;
    }

    button:hover {
      background-color: #0056b3;
    }

    @media screen and (max-width: 480px) {
      .login-box {
        margin: 5vh 10px;
      }
    }
  </style>
</head>
<body>
  <div class="login-box">
    <h2>Login to Skyview Weather</h2>
    <form method="POST" action="auth.php">
      <input type="email" name="email" placeholder="Email" required><br>
      <input type="password" name="password" placeholder="Password" required><br>
      <button type="submit">Login</button>
    </form>
  </div>
  <p style="margin: 15px 0; font-weight: bold; text-align: center;">— OR —</p>
  <a href="google-login.php"><button>Login with Google</button></a>
</body>
</html>
```

3. Create auth.php to Handle Logins:

```
tehergs@csci-411-linux:~$ sudo nano /var/www/html/auth.php
```

4. Write Code for auth.php:

```
GNU nano 4.8 /var/www/html/auth.php Modified
<?php
session_start();
include 'db_connect.php'; // Reuse your existing DB connection

// Get email and password from the form submission
$email = $_POST['email'];
$password = $_POST['password'];

// In production, hash and verify passwords securely
$sql = "SELECT * FROM Users WHERE email = '$email' AND password = '$password'";
$result = $conn->query($sql);

// Check if login is valid
if ($result->num_rows > 0) {
    $row = $result->fetch_assoc();
    $_SESSION['user_email'] = $row['email'];
    $_SESSION['user_name'] = $row['username'];
    header("Location: weather_ui.php");
} else {
    echo "Invalid login. <a href='login.php'>Try again</a>";
}
?>
```

5. Set Up Google OAuth 2.0 Login:

OAuth 2.0 Client IDs

<input type="checkbox"/>	Name	Creation date ↓	Type	Client ID	Actions
<input type="checkbox"/>	WeatherLoginClient	Mar 27, 2025	Web application	913995445558-eop3...	

Google Auth Platform / Clients / Client: 913995445558-eop3jnicnggj46n14mgfheqo33jlfqdh.apps.googleusercontent.com

← Client ID for Web application Delete

Name *
WeatherLoginClient
The name of your OAuth 2.0 client. This name is only used to identify the client in the console and will not be shown to end users.

The domains of the URIs you add below will be automatically added to your [OAuth consent screen](#) as [authorized domains](#).

Authorized JavaScript origins ②

For use with requests from a browser

+ Add URI

Authorized redirect URIs ②

For use with requests from a web server

URIs 1 *
http://skyview.work.gd/google-callback.php

+ Add URI

Note: It may take 5 minutes to a few hours for settings to take effect

Save Cancel

Additional information

Client ID	913995445558-eop3jnicnggj46n14mgfheqo33jlfqdh.apps.googleusercontent.com
Creation date	March 27, 2025 at 6:48:59 PM GMT-5

Client secrets

If you are in the process of changing client secrets, you can manually rotate them without downtime. [Learn more](#)

Client secret	GOCSPX-IoXf5sJjnr4NJaZ2Ck5b4Gnc4Ql
Creation date	March 27, 2025 at 6:48:59 PM GMT-5
Status	Enabled

+ Add secret

6. Install Google OAuth PHP Library:

```
tehergs@csci-411-linux:~$ sudo apt install composer
```

```
tehergs@csci-411-linux:/var/www/html$ sudo composer require google/apiclient:^2.0
Do not run Composer as root/super user! See https://getcomposer.org/root for details
Warning from https://repo.packagist.org: Support for Composer 1 will be shutdown on August 1st 2025. You should
upgrade to Composer 2. See https://blog.packagist.com/shutting-down-packagist-org-support-for-composer-1-x/
./composer.json has been updated
Loading composer repositories with package information
Warning from https://repo.packagist.org: Support for Composer 1 will be shutdown on August 1st 2025. You should
upgrade to Composer 2. See https://blog.packagist.com/shutting-down-packagist-org-support-for-composer-1-x/
Updating dependencies (including require-dev)
Package operations: 18 installs, 0 updates, 0 removals
 - Installing ralouphie/getallheaders (3.0.3): Downloading (100%)
 - Installing psr/http-message (2.0): Downloading (100%)
 - Installing psr/http-factory (1.1.0): Downloading (100%)
 - Installing guzzlehttp/psr7 (2.7.0): Downloading (100%)
 - Installing symfony/deprecation-contracts (v2.5.4): Downloading (100%)
 - Installing psr/http-client (1.0.3): Downloading (100%)
 - Installing guzzlehttp/promises (2.0.4): Downloading (100%)
 - Installing guzzlehttp/guzzle (7.9.2): Downloading (100%)
 - Installing paragonie/random_compat (v9.99.100): Downloading (100%)
 - Installing paragonie/constant_time_encoding (v2.7.0): Downloading (100%)
 - Installing phpseclib/phpseclib (3.0.43): Downloading (100%)
 - Installing psr/log (1.1.4): Downloading (100%)
 - Installing monolog/monolog (2.10.0): Downloading (100%)
 - Installing firebase/php-jwt (v6.10.0): Downloading (100%)
 - Installing google/apiclient-services (v0.355.0): Downloading (100%)
 - Installing psr/cache (1.0.1): Downloading (100%)
 - Installing google/auth (v1.37.2): Downloading (100%)
 - Installing google/apiclient (v2.16.1): Downloading (100%)
guzzlehttp/psr7 suggests installing laminas/laminas-httphandler (Emit PSR-7 responses)
guzzlehttp/guzzle suggests installing ext-intl (Required for Internationalized Domain Name (IDN) support)
paragonie/random_compat suggests installing ext-libsodium (Provides a modern crypto API that can be used to gene
rate random bytes.)
phpseclib/phpseclib suggests installing ext-libsodium (SSH2/SFTP can make use of some algorithms provided by the
```

7. Create google-callback.php (Handles Google Login Response):

```
tehergs@csci-411-linux:~$ sudo nano /var/www/html/google-callback.php
```

```
GNU nano 4.8 /var/www/html/google-callback.php Modifi
<?php
require_once 'vendor/autoload.php';
session_start();

$client = new Google_Client();
$client->setClientId('913995445558-eop3jnicnggj46n14mgfhego33jlfqdh.apps.googleusercontent.com');
$client->setClientSecret('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');
$client->setRedirectUri('http://skyview.work.gd/google-callback.php'); // MUST match Google OAuth settings
$client->addScope("email");
$client->addScope("profile");

if (isset($_GET['code'])) {
    $token = $client->fetchAccessTokenWithAuthCode($_GET['code']);
    $client->setAccessToken($token);

    // Get user info from Google
    $oauth = new Google_Service_Oauth2($client);
    $user = $oauth->userinfo->get();

    // Store user info in session
    $_SESSION['user_email'] = $user->email;
    $_SESSION['user_name'] = $user->name;

    // Redirect to your weather app UI
    header('Location: weather_ui.php');
    exit();
} else {
    echo "Error: No authorization code received.";
}
?>
```

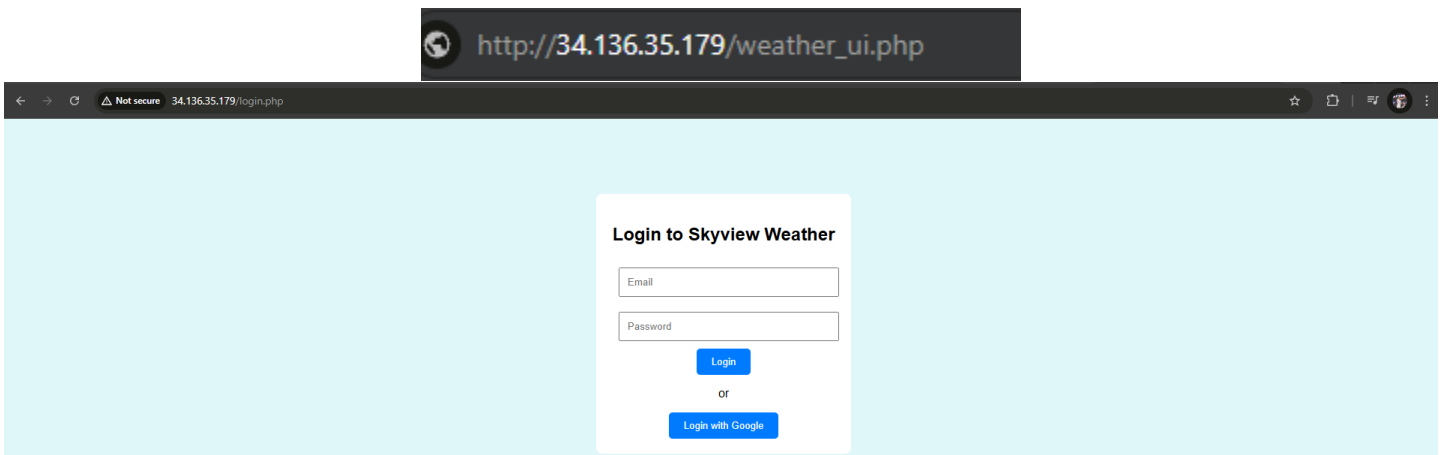
8. Update weather_ui.php to Require Login:

```
GNU nano 4.8 /var/www/html/weather_ui.php
<?php
session_start(); // Add this line first to enable session tracking

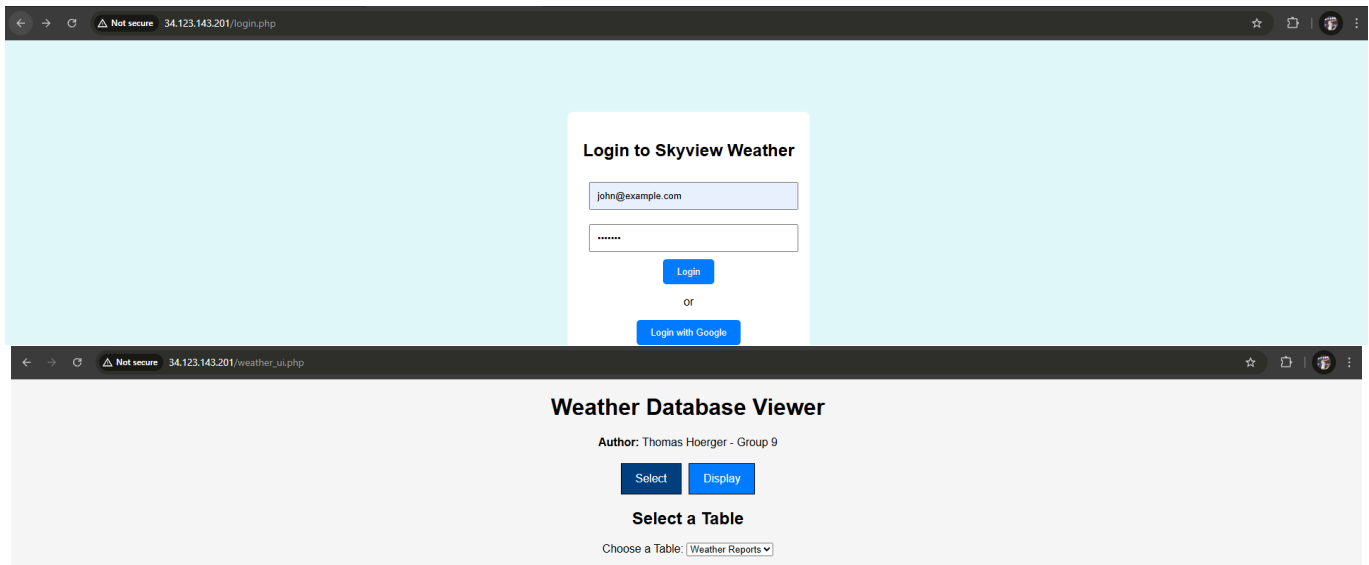
// Redirect user to login page if they are not logged in
if (!isset($_SESSION['user_email']) && !isset($_SESSION['user'])) {
    header("Location: login.php");
    exit();
}

/**
 * Weather Database Interactive Viewer
 * Author: Thomas Hoerger - Group 9
 *
 * Description:
 * This program allows users to dynamically select a table from the Weather Database
 * and display its data in the "Display" tab without needing a submit button.
 */
```

9. Test to Make Sure it Redirects You to login.php if You're Not Logged in:



10. Test a Test User:



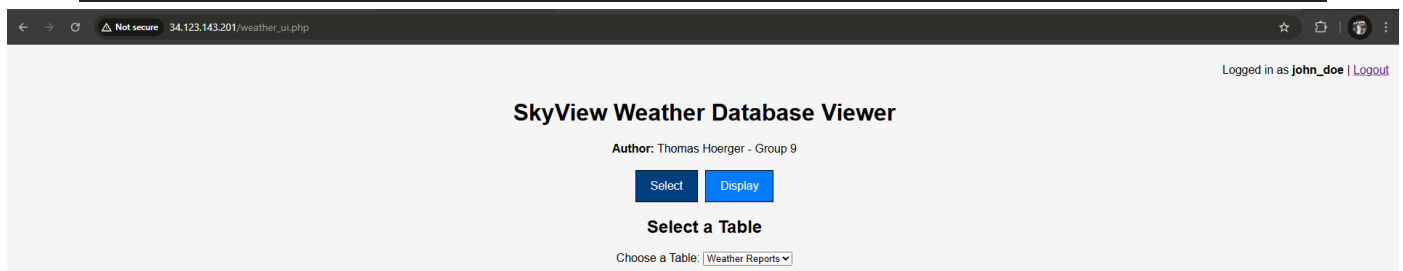
11. Create a Logout Option:

```
tehergs@csci-411-linux:~$ sudo nano /var/www/html/logout.php

GNU nano 4.8 /var/www/html/logout.php
<?php
session_start();           // Start the current session
session_destroy();         // Destroy all session data
header("Location: login.php"); // Redirect back to login page
exit();
?>
```

12. Add a Logout Link to weather_ui.php:

```
<p style="text-align: right; padding: 10px;">
  Logged in as <?= htmlspecialchars($_SESSION['user_name']) ?? $_SESSION['user_email'] ?? 'User' ?></strong> |
  <a href="logout.php">Logout</a>
</p>
```



13. Have your HCI/GUI/UI/UX running on at least 2 platforms: laptop and mobile (Edit Style Sheet):

```
GNU nano 4.8 /var/www/html/weather_ui.php
<meta name="viewport" content="width=device-width, initial-scale=1.0">
GNU nano 4.8 /var/www/html/login.php
<!DOCTYPE html>
<html>
<head>
<title>Skyview Login</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<style>
  body {
    font-family: Arial, sans-serif;
    background: #e0f7fa;
    margin: 0;
    padding: 0;
  }

  .login-box {
    max-width: 400px;
    margin: 10vh auto;
    background: white;
    padding: 20px;
    border-radius: 8px;
    box-shadow: 0px 4px 10px rgba(0,0,0,0.1);
  }

  input, button {
    width: 100%;
    padding: 10px;
    margin: 8px 0;
    font-size: 16px;
    border-radius: 5px;
    border: 1px solid #ccc;
  }

  button {
    background-color: #007BFF;
    color: white;
    border: none;
  }

  button:hover {
    background-color: #0056b3;
  }

  @media screen and (max-width: 480px) {
    .login-box {
      margin: 5vh 10px;
    }
  }
</style>
```

```
<title>Weather Database - Group 9</title>
<style>
  body {
    font-family: Arial, sans-serif;
    text-align: center;
    background-color: #f5f5f5;
    margin: 0;
    padding: 0;
  }

  h1, h2 {
    font-weight: bold;
    color: black;
    margin: 10px;
  }

  .tab-container {
    display: flex;
    flex-wrap: wrap;
    justify-content: center;
    margin: 10px;
  }

  .tab {
    padding: 12px 20px;
    cursor: pointer;
    background-color: #007bff;
    color: white;
    border: 1px solid black;
    margin: 5px;
    border-radius: 6px;
  }

  .tab.active {
    background-color: #003f7f;
  }

  .content {
    display: none;
    margin: 10px auto;
    width: 95%;
  }

  .content.active {
    display: block;
  }

  table {
    width: 100%;
    border-collapse: collapse;
    background-color: white;
    margin-top: 15px;
    font-size: 14px;
  }

  th, td {
    padding: 10px;
    border: 1px solid #ccc;
    text-align: center;
  }
}
```

```

th {
    background-color: #007BFF;
    color: white;
}

tr:nth-child(even) {
    background-color: #f9f9f9;
}

tr:hover {
    background-color: #ddd;
}

select {
    padding: 8px;
    font-size: 16px;
    margin-top: 10px;
    width: 90%;
    max-width: 300px;
}

@media screen and (max-width: 600px) {
    .tab {
        width: 90%;
        font-size: 16px;
    }

    table, th, td {
        font-size: 12px;
        padding: 8px;
    }
}
</style>

```

```

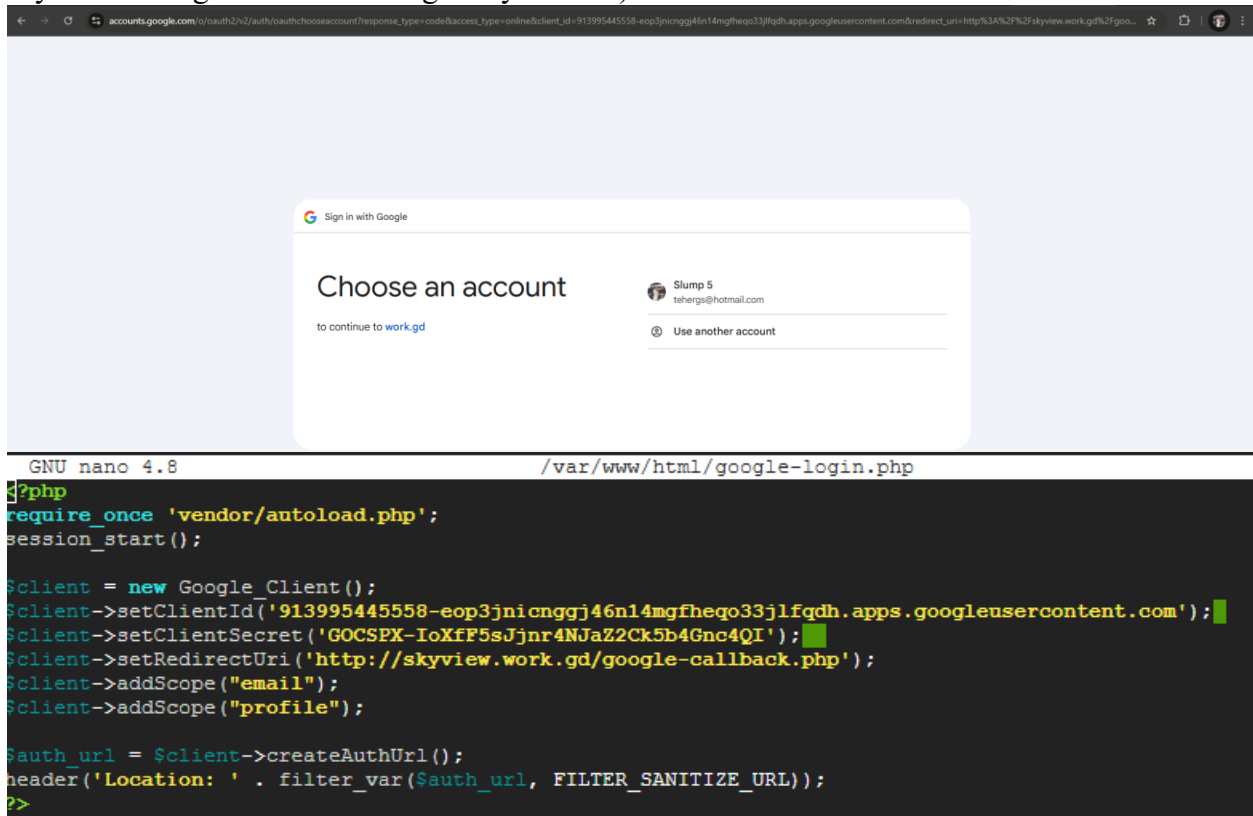
<div class="table-wrapper">
    <table>
        <tr>
            <?php foreach ($headers as $header): ??>
                <th><?= htmlspecialchars($header); ?></th>
            <?php endforeach; ??>
        </tr>
        <?php while ($row = $result->fetch_assoc()): ??>
            <tr>
                <?php foreach ($headers as $header): ??>
                    <td><?= htmlspecialchars($row[$header]); ?></td>
                <?php endforeach; ??>
            </tr>
        <?php endwhile; ??>
    </table>
</div>

```



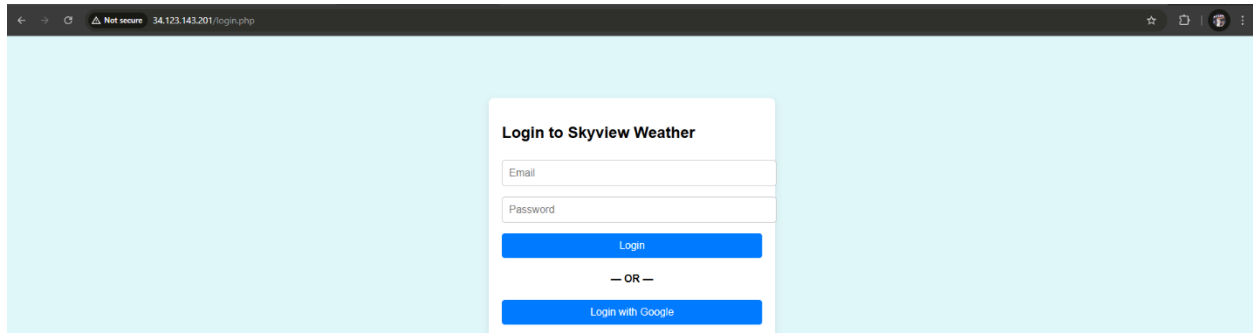
```
.table-wrapper {
    width: 100%;
    overflow-x: auto;
    display: flex;
    justify-content: center;
    margin-top: 10px;
}
```

14. Test Google Login (Works just need to wait for DNS propagation delay to use skyview.work.gd because I changed my VPS IP)

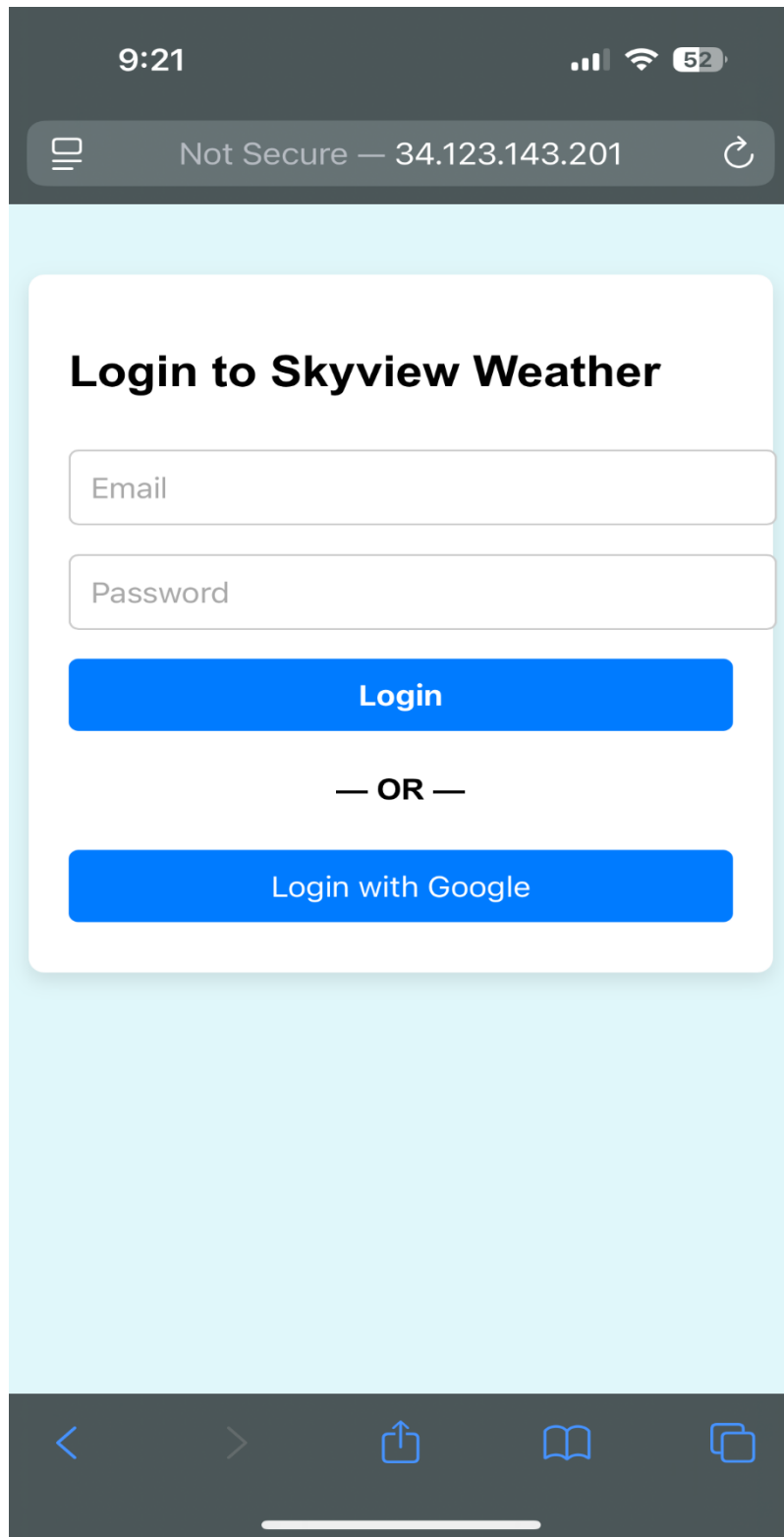


15. Test Website on PC and Mobile

PC:



Mobile:



The image shows a mobile application interface for logging into 'Skyview Weather'. At the top, a dark status bar displays the time '9:21', signal strength, Wi-Fi, and a 52% battery level. Below this is a browser-like address bar with a hamburger menu icon, the text 'Not Secure — 34.123.143.201', and a refresh icon. The main content area has a light blue background. A white rounded rectangle contains the login form. The form has a title 'Login to Skyview Weather', followed by two input fields labeled 'Email' and 'Password'. Below these is a blue 'Login' button. A separator '— OR —' is centered below the button. At the bottom of the form is a blue button labeled 'Login with Google'. The very bottom of the screen features a dark navigation bar with five icons: a back arrow, a forward arrow, a share icon, a bookmark icon, and a tabs icon. A white horizontal line is visible at the bottom of the navigation bar.

9:21

Not Secure — 34.123.143.201

Login to Skyview Weather

Email

Password

Login

— OR —

Login with Google

9:17

53

Not Secure — 34.123.143.201

Logged in as john_doe | [Logout](#)

SkyView Weather Database Viewer

Author: Thomas Hoerger - Group 9

Select

Display

Forecasts

forecast_id	location_id	forecast_date	expected_temp	expected_condition
1	1	2025-02-25	27	Sunny
2	2	2025-02-26	31.5	Cloudy
3	3	2025-02-27	23	Rainy
4	4	2025-02-28	19.5	Stormy
5	5	2025-03-01	28	Sunny
6	6	2025-03-02	16	Snowy
7	7	2025-03-03	36	Sunny
8	8	2025-03-04	22	Cloudy

<

>

9:17

53

Not Secure — 34.123.143.201

Logged in as john_doe | [Logout](#)

SkyView Weather Database Viewer

Author: Thomas Hoerger - Group 9

Select

Display

Forecasts

location_id	forecast_date	expected_temp	expected_condition
1	2025-02-25	27	Sunny
2	2025-02-26	31.5	Cloudy
3	2025-02-27	23	Rainy
4	2025-02-28	19.5	Stormy
5	2025-03-01	28	Sunny
6	2025-03-02	16	Snowy
7	2025-03-03	36	Sunny
8	2025-03-04	22	Cloudy

<

>