

Three Level Password Authentication System

A project report submitted

to

MANIPAL ACADEMY OF HIGHER EDUCATION

For Partial Fulfillment of the Requirement for the

Award of the Degree

of

Bachelor of Technology

in

Information Technology

By

Name: Abhay Patel (225811230)

Name: Anshul Aryan(225811088)

Under the guidance of

Under the guidance of

Dr. Abhijit Das

Assistant Professor - Senior Scale

Department of I &T

Manipal Institute of Technology

Bengaluru, India



MANIPAL INSTITUTE OF TECHNOLOGY

MANIPAL

(A constituent unit of MAHE, Manipal)

Nov 2024

DECLARATION

I hereby declare that this project work entitled **Three Level Password Authentication System** is original and has been carried out by me in the Department of Information and Communication Technology of Manipal Institute of Technology, Manipal, under the guidance of Dr.Abhijit Das Designation of internal guide ,Department of Information and Communication Technology, M.I.T ,Bengaluru. No part of this work has been submitted for the award of a degree or diploma either to this University or to any other Universities.

Place:Bengaluru

Date :07/11/2024

Abhay Patel

Anshul Aryan



MANIPAL INSTITUTE OF TECHNOLOGY

MANIPAL
(A constituent unit of MAHE, Manipal)

CERTIFICATE

This is to certify that this project entitled **Three Level Password Authentication System** is a bonafide project work done by **Mr. Abhay Patel (Reg.No.:225811230)** and **Mr Anshul Aryan (Reg.No.:225811088)** at Manipal Institute of Technology, Manipal, independently under my guidance and supervision for the award of the Degree of Bachelor of Technology in Information Technology.

Dr.Abhijit Das

Designation of guide

Department of I &T

Manipal Institute of Technology

TechnologyBengaluru, India

Dr. Dayananda P

Professor & Head

Department of I & T

Manipal Institute of

Bengaluru, India

Table of contents:

1. Abstract
2. Introduction
3. Objective
4. Scope
5. Methodology
6. Implementation
7. Snapshots
8. Conclusion

Abstract:

The project is aimed at developing a multi-level authentication system for a web application to enhance security. It focuses on protecting user data through a layered approach that combines email/password authentication, security question verification, and PIN validation. By leveraging Firebase for authentication and Firestore for secure data storage, the system offers robust protection against unauthorized access while ensuring a seamless user experience. The authentication process begins with traditional email and password entry, followed by a user-defined security question and PIN validation. The implementation of these multiple layers adds an extra barrier to malicious actors, making it significantly harder for them to breach user accounts. This project ensures a high level of security without compromising ease of use, providing users with a safe and reliable method of accessing their sensitive data

Introduction:

In today's digital world, security has become a fundamental concern for both users and developers of web applications. As cyber threats such as hacking, identity theft, and phishing attacks become increasingly common, securing users' personal and financial data is of paramount importance. Traditional single-factor authentication systems, such as password-only logins, are often insufficient to protect sensitive information from unauthorized access.

This project addresses the need for an enhanced authentication mechanism that combines multiple layers of security. The goal is to build a multi-level authentication system that integrates email/password authentication, security question validation, and a PIN verification system, all while maintaining ease of use for the end user. Firebase Authentication and Firestore have been chosen as the core technologies for this application. Firebase ensures reliable and secure user authentication, while Firestore is used to store user-related data, including security questions, answers, and PINs. The system aims to provide an additional layer of security by requiring users to pass through multiple authentication checks before gaining access to their accounts, making it far more difficult for unauthorized users to breach the system.

Objective:

The primary objective of this project is to develop a secure and user-friendly multi-level authentication system for web applications. Below are the specific objectives that guide the development of this project:

1. **Secure User Authentication:** To build a web application that enhances user security by implementing multi-level authentication. This includes the integration of email/password authentication, security questions, and PIN verification to safeguard sensitive data.
2. **Email and Password Authentication:** To integrate Firebase Authentication for secure registration and login using email and password, ensuring that user credentials are protected with modern security standards.
3. **Security Question and Answer Verification:** To implement an additional layer of security where users are required to answer a pre-configured security question to verify their identity.
4. **PIN Validation:** To use PIN validation as a third layer of security, adding an extra barrier to entry by requiring users to enter a PIN known only to them.
5. **Password Recovery System:** To provide users with the ability to recover their account credentials by resetting their password through a secure email-based link sent via Firebase Authentication.
6. **Firestore Integration for Data Storage:** To store user-specific data, such as answers to security questions and PINs, securely in Firestore, ensuring data privacy and protection.
7. **Responsive and User-Friendly Interface:** To design a user interface that is easy to navigate and works seamlessly on different devices, including desktops, tablets, and smartphones, enhancing the overall user experience.

Scope:

The scope of this project includes the design, development, and implementation of a multi-level authentication system for a web application. The project covers both frontend and backend development, with a focus on security and user experience. Below are the key areas covered by the project:

1. **User Authentication:** The system is designed to handle all aspects of user authentication, starting from the sign-up process to account recovery. It integrates multiple security layers, including email/password authentication, security question validation, and PIN entry.
2. **Data Security:** Firebase Authentication ensures that user credentials are securely handled during the registration and login processes. Firestore is used for securely storing sensitive information, including answers to security questions and user-specific PINs. This system is designed to protect data from unauthorized access and ensure privacy.
3. **Multi-Level Authentication:** The project covers the integration of three layers of security: Email/Password Authentication, Security Question and Answer Validation, and PIN Verification. Each layer is designed to strengthen the security of the user's account and protect sensitive data from unauthorized access.
4. **Compatibility:** The application is designed to work seamlessly across modern web browsers such as Google Chrome, Mozilla Firefox, and Safari, as well as mobile devices, ensuring that it is accessible to a wide audience.
5. **Password Reset:** A password recovery mechanism has been integrated into the project, allowing users to securely reset their passwords through an email-based link sent via Firebase Authentication.
6. **Authentication Data Handling:** All user-specific data, such as answers to security questions and PINs, are securely stored in Firestore, ensuring that sensitive data is handled with the highest level of security.

Methodology:

The methodology used to develop the multi-level authentication system was a step-by-step approach, starting from the planning phase through to testing and deployment. The key phases of the development process are outlined below:

1. **Planning Phase:** In this phase, the requirements for the project were identified, including the need for multi-layered security and secure storage of user data. Firebase Authentication and Firestore were selected due to their scalability, ease of use, and robust security features.
2. **Design Phase:** During the design phase, the user interface (UI) and user experience (UX) were carefully planned. The system was designed to ensure ease of use while maintaining a high level of security. The layout was created with responsiveness in mind, ensuring compatibility across devices.
3. **Development Phase:** The frontend of the application was developed using HTML, CSS, and JavaScript. Firebase Authentication was integrated to handle user sign-up and login operations securely, while Firestore was used to store user data, including security question answers and PINs. The multi-level authentication logic was implemented, ensuring users passed through each security check.
4. **Testing Phase:** Thorough testing was conducted to ensure all features worked as expected. This included testing the registration process, login process, security question and PIN validation, password recovery, and Firestore data storage. Any issues identified during testing were addressed and resolved.
5. **Deployment:** After successful testing, the web application was deployed on a server, making it accessible to users. Continuous monitoring and bug fixes were implemented to ensure smooth operation after deployment.

Implementation:

The implementation of the multi-level authentication system involved the creation of several key webpages and the integration of Firebase Authentication and Firestore. Each page was developed to handle a specific part of the authentication process. Below is a summary of the key pages and their functionalities:

1. **Sign-Up Page (index.html):** This page allows new users to create an account by providing their email, password, security question, and answer. Once submitted, the system uses Firebase Authentication to register the user, and the data is stored in Firestore.
2. **Login Page (login.html):** This page allows existing users to log in using their email and password. After successful authentication, the user is redirected to the next step in the authentication process.
3. **Security Question and PIN Setup Page (questionSetup.html):** On this page, users are asked to provide a security question and answer. They also set a PIN, which will be used as an additional security layer during login.
4. **PIN Verification Page (pin.html):** Users are required to enter the PIN they previously set. This step adds another layer of security before granting access to the user's account.
5. **Password Recovery Page (forgotPassword.html):** If a user forgets their password, they can use this page to recover it. Firebase Authentication sends a password reset link to the user's email, allowing them to reset their password securely.

Index Page

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>3 Level Pass Auth System</title>
  <link rel="stylesheet" href="/static/style.css">
</head>
<body>
  <div class="container">
    <div class="right-section">
      <h1>3 Level Password Authentication System</h1>
      <div class="navigation">
        <a href="signup.html"><button>Signup</button></a>
        <a href="login.html"><button>Login</button></a>
      </div>
    </div>
  </div>
</body>
</html>
```

Sign Up Page

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Sign Up</title>
  <link rel="stylesheet" href="/static/style.css">
</head>
<body>
  <div class="container">
    <h2>Sign Up</h2>
    <form id="signup-form">
      <label for="email">Email:</label>
      <input type="email" id="signup-email" placeholder="Enter your email"
required><br><br>

      <label for="password">Password:</label>
      <input type="password" id="signup-password" placeholder="Enter a strong password"
required><br><br>

      <label for="question">Security Question:</label>
      <input type="text" id="signup-question" placeholder="Enter your custom security
question" required><br><br>

      <label for="answer">Answer:</label>
      <input type="text" id="signup-answer" placeholder="Enter the answer to your question"
required><br><br>

      <label for="birthdate">Birthdate:</label>
      <input type="date" id="signup-birthdate" required><br><br>

      <button type="submit">Sign Up</button>
    </form>

    <div class="google-signup">
      <button id="google-sign-up" type="button">
        
        Signup with Google
      </button>
    </div>

    <p>Already have an account? <a href="login.html">Login</a></p>
  </div>

  <script src="/config/signup.js" type="module"></script>
</body>
</html>
```

Question SignUp Page

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Security Setup</title>
  <link rel="stylesheet" href="/static/style.css">
</head>
<body>
  <div class="container">
    <h2>Security Setup</h2>
    <form id="q-a-form">
      <label for="question">Security Question:</label>
      <input type="text" id="security-question" placeholder="Enter your security question"
required><br><br>

      <label for="answer">Answer:</label>
      <input type="text" id="security-answer" placeholder="Enter the answer to your question"
required><br><br>

      <label for="birthdate">Birthdate:</label>
      <input type="date" id="birthdate" required><br><br>

      <button type="submit">Save</button>
    </form>
  </div>
  <script src="/config/questionsignup.js" type="module"></script>
</body>
</html>
```

Login Page

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Login</title>
  <link rel="stylesheet" href="/static/style.css">
</head>
<body>
  <div class="container">
    <h2>Login</h2>
    <br><br>
    <form id="login-form">
      <input type="email" id="login-email" placeholder="Email" required>
      <input type="password" id="login-password" placeholder="Password" required>

      <div class="navigation">
        <button type="submit">Login</button>
      </div>
      <div class="google-sign-up">
        <button id="google-sign-up" type="button">
          
          Login with Google
        </button>
      </div>
      <p><a href="forgotpassword.html">Forgot Password?</a></p>
    </form>
    <p>Don't have an account? <a href="signup.html">Sign up</a></p>
  </div>
  <script src="/config/login.js" type="module"></script>
</body>
</html>
```

Forgot Password Page

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Forgot Password</title>
  <link rel="stylesheet" href="/static/style.css">
</head>
<body>
  <div class="container">
    <h2>Forgot Password</h2>
    <p>Please enter your email address. You will receive a link to create a new password via
email.</p>
    <form id="forgot-password-form">
      <input type="email" id="forgot-email" placeholder="Email" required>
      <div class="navigation">
        <button type="submit">Send Reset Link</button>
      </div>
    </form>
    <p>Remembered your password? <a href="login.html">Login</a></p>
  </div>
  <script src="/config/forgotpassword.js" type="module"></script>
</body>
</html>
```

Question Page

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Security Question</title>
  <link rel="stylesheet" href="/static/style.css">
</head>
<body>
  <div class="container">
    <h2>Answer Security Question</h2>
    <br><br>
    <form id="security-question-form">
      <label for="question">Security Question:</label>
      <input type="text" id="security-question" placeholder="Your question will appear here"
readonly><br><br>

      <label for="answer">Answer:</label>
      <input type="text" id="answer" placeholder="Your answer" required><br><br>

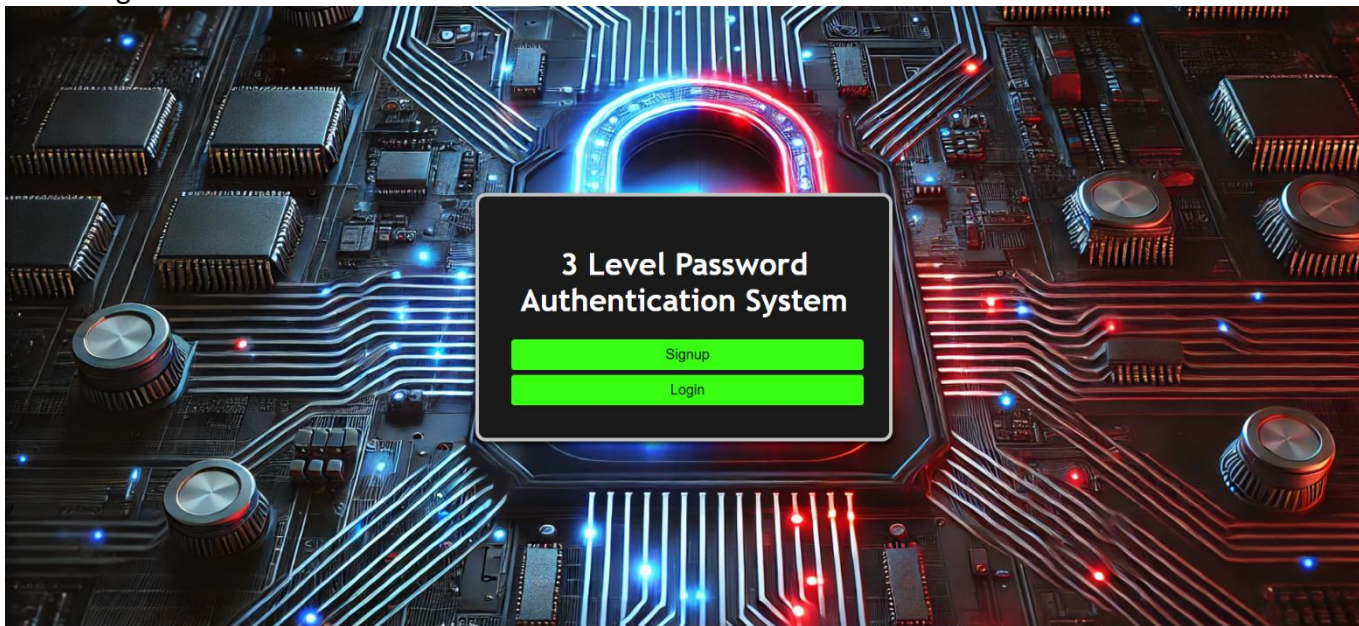
      <div class="navigation">
        <button type="submit">Verify Answer</button>
      </div>
    </form>
  </div>
</div>
<script src="/config/questions.js" type="module"></script>
</body>
</html>
```

Pin Password Page

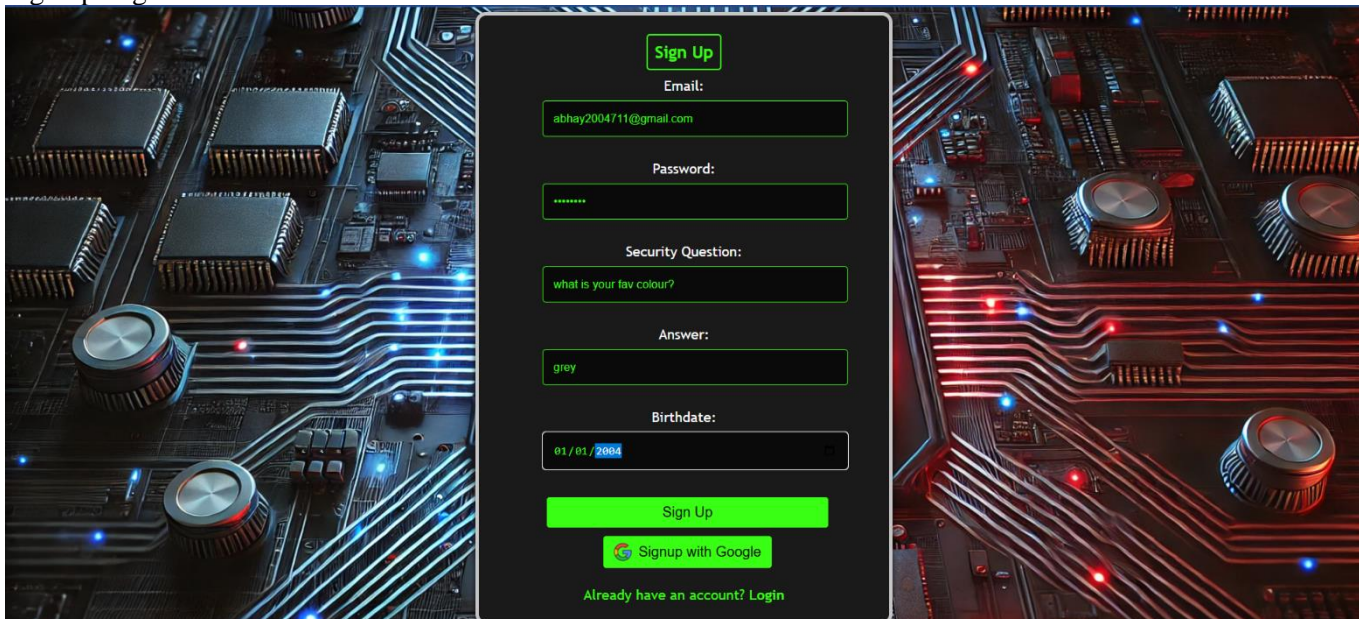
```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Enter PIN</title>
  <link rel="stylesheet" href="/static/style.css">
</head>
<body>
  <div class="container">
    <h2>Enter PIN</h2>
    <form id="pin-form">
      <label for="pin">PIN (first 4 digits of your birth year):</label>
      <input type="password" id="pin" placeholder="Enter PIN" required minlength="4"
maxlength="4" pattern="\d{4}">
      <button type="submit">Submit</button>
    </form>
  </div>
  <script src="/config/pin.js" type="module"></script>
</body>
</html>
```


Snapshot:

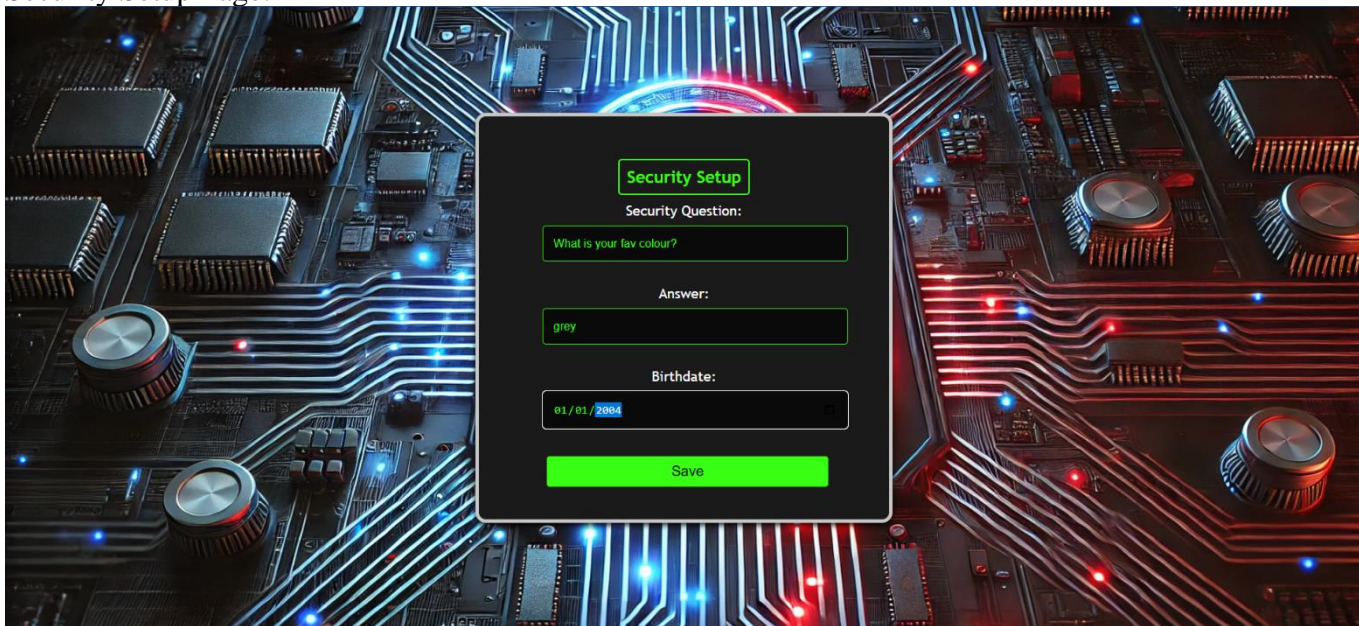
Index Page:



Sign up Page:



Security Setup Page:



The Security Setup page is displayed over a background of a glowing circuit board. The page has a dark theme with green accents. It contains a title 'Security Setup', a 'Security Question' field with the text 'What is your fav colour?', an 'Answer' field with the text 'grey', and a 'Birthdate' field with the text '01/01/2004'. A green 'Save' button is at the bottom.

Security Setup

Security Question:

What is your fav colour?

Answer:

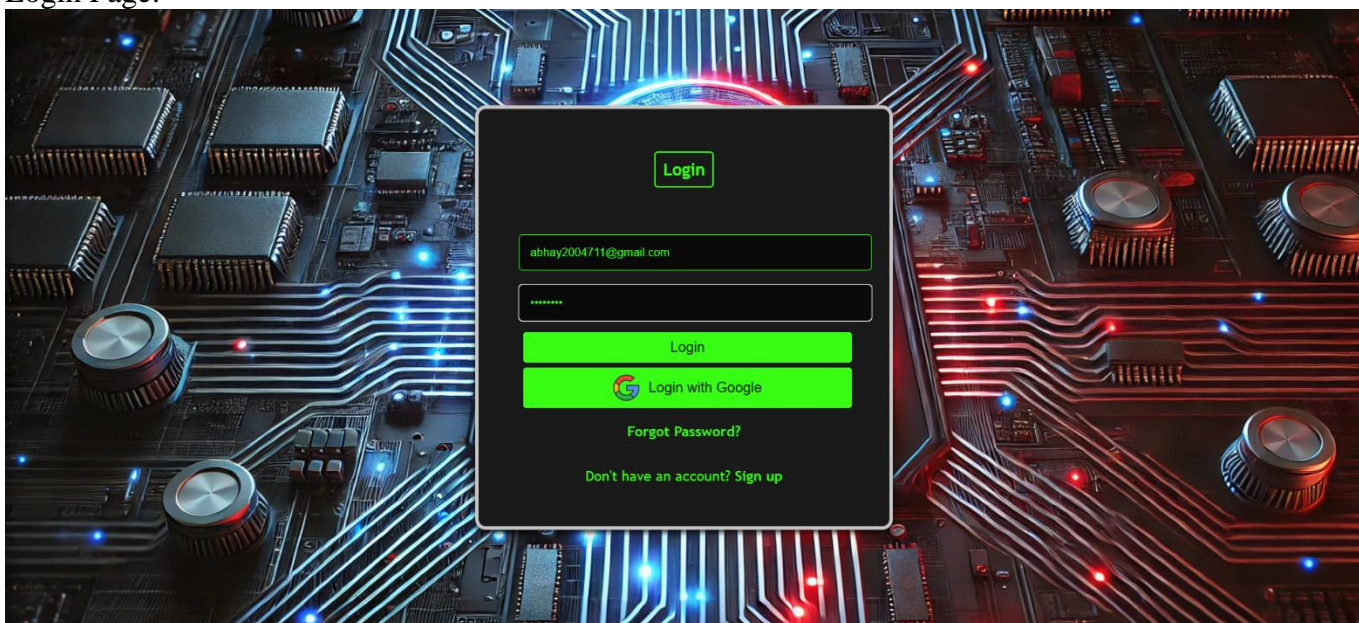
grey

Birthdate:

01/01/2004

Save

Login Page:




The Login page is displayed over the same glowing circuit board background. It has a dark theme with green accents. It contains a 'Login' title, an email input field with 'abhay2004711@gmail.com', a password input field with masked characters, a green 'Login' button, a 'Login with Google' button with the Google logo, a 'Forgot Password?' link, and a 'Don't have an account? Sign up' link.

Login

abhay2004711@gmail.com

password

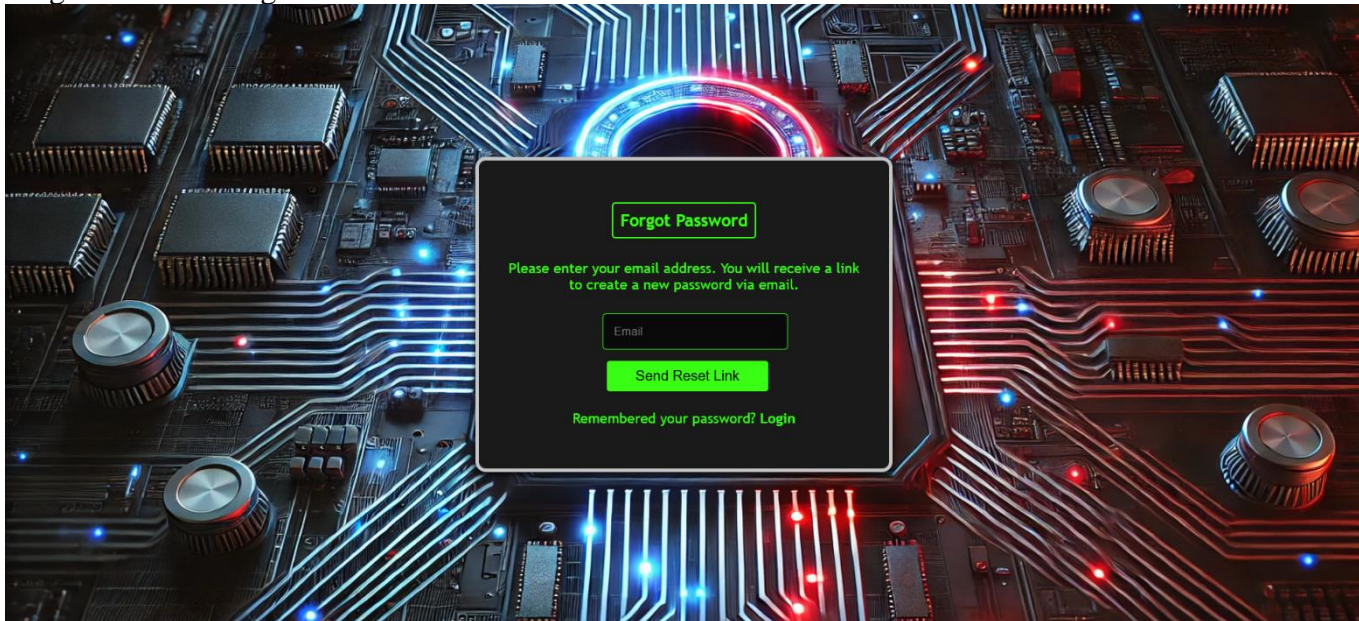
Login

 Login with Google

[Forgot Password?](#)

[Don't have an account? Sign up](#)

Forgot Password Page:

A futuristic digital interface for a password reset. The background is a dark, glowing circuit board with blue and red light trails. A central dark grey panel contains the form. At the top of the panel is a green-bordered box with the title "Forgot Password". Below it, green text reads "Please enter your email address. You will receive a link to create a new password via email." There is a white text input field for the email. Below the input field is a green button labeled "Send Reset Link". At the bottom of the panel, green text says "Remembered your password? Login".

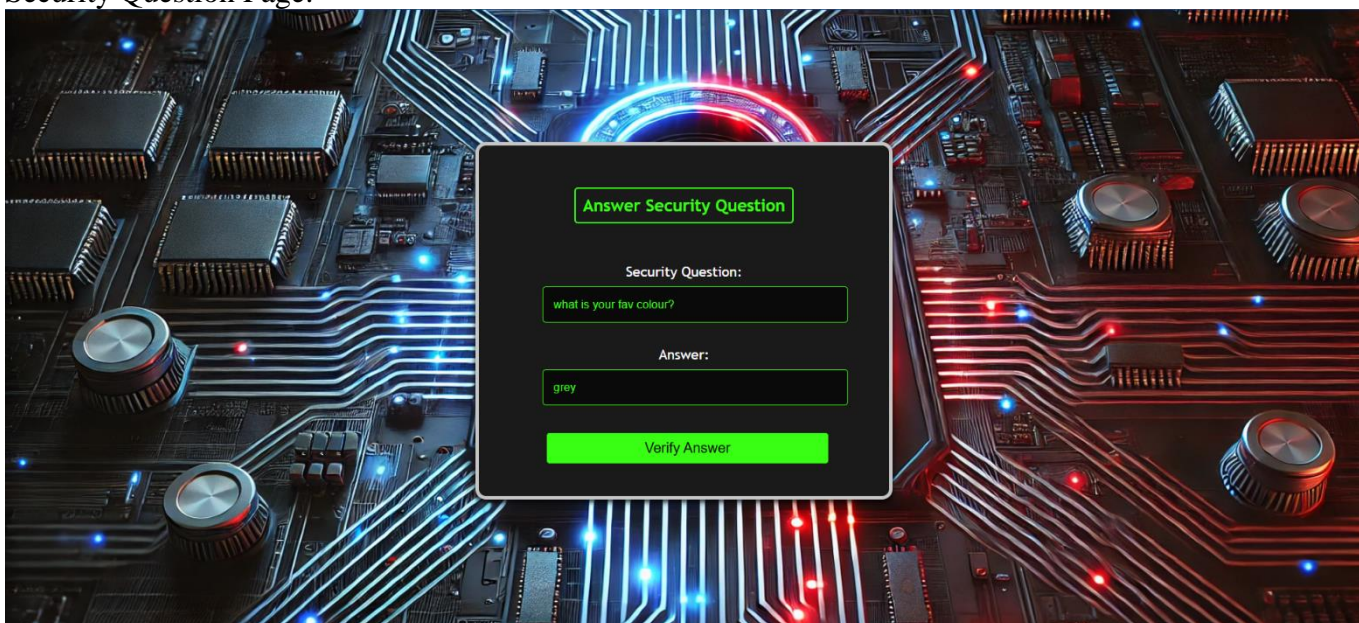
Forgot Password

Please enter your email address. You will receive a link to create a new password via email.

Send Reset Link

Remembered your password? [Login](#)

Security Question Page:

A futuristic digital interface for a security question verification. The background is a dark, glowing circuit board with blue and red light trails. A central dark grey panel contains the form. At the top of the panel is a green-bordered box with the title "Answer Security Question". Below it, green text reads "Security Question:". There is a white text input field containing the text "what is your fav colour?". Below the input field, green text reads "Answer:". There is a white text input field containing the text "grey". Below the input field is a green button labeled "Verify Answer".

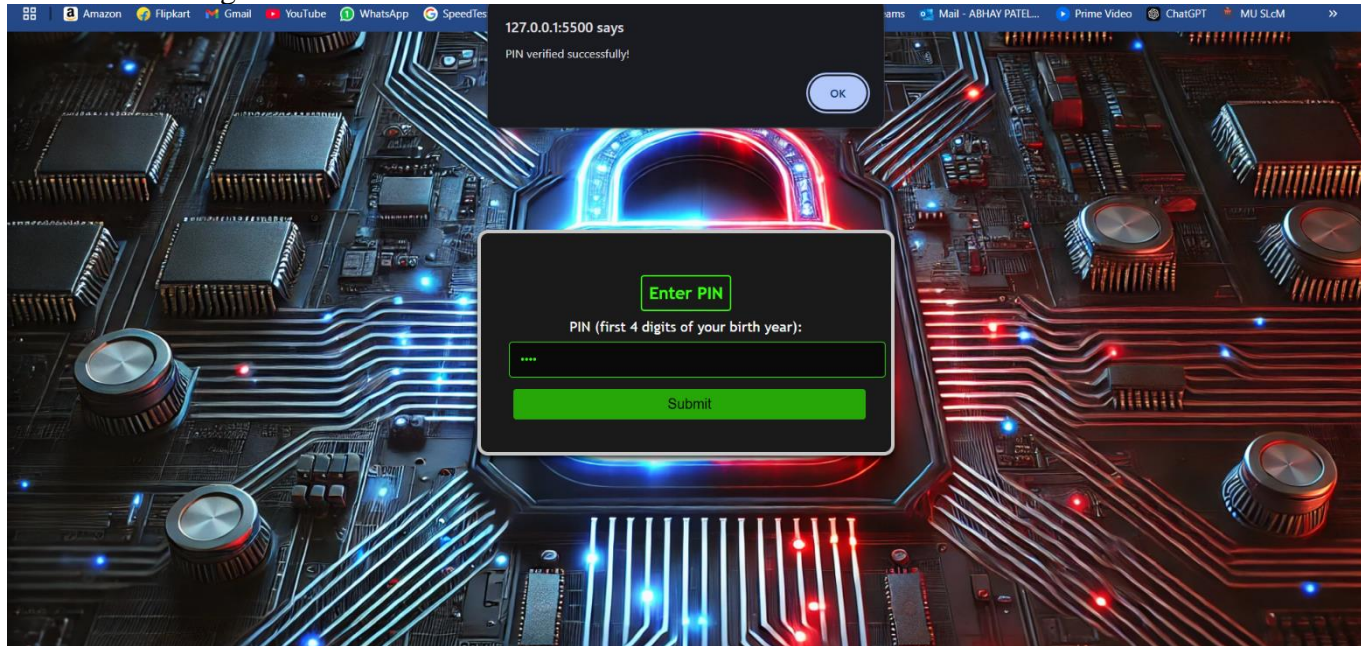
Answer Security Question

Security Question:

Answer:

Verify Answer

Pin Password Page:



Conclusion:

The project successfully implements a robust multi-level authentication system using Firebase Authentication, Firestore, and a combination of email/password, security questions, and PIN verification. This multi-layered approach provides enhanced security to protect user data from unauthorized access. By integrating Firebase for authentication and Firestore for secure data storage, the system offers a reliable and scalable solution for modern web applications. The user interface is designed to be responsive, ensuring that users can access their accounts securely across different devices.

With cyber threats becoming more sophisticated, systems like the one developed in this project are crucial for ensuring that sensitive user data is protected. This project serves as a foundation for building secure and scalable authentication systems for future web applications, and it provides a practical demonstration of how to implement modern security practices in web development.