

**KH6005CEM Security**

**CW 2**

**DR AYMAN TAHA**

**SECURITY WEBSITE**

GITHUB link:

Introduction

In the realm of cybersecurity and information security, several key concepts play a crucial role in ensuring the confidentiality, integrity, and availability of data. Among these, Authentication, Multi-Factor Authentication (MFA), User Access Control, and Cryptography are fundamental components that contribute to creating secure and robust systems. This coursework aims to address the contemporary challenges of creating secure computer systems in a world heavily dependent on IT infrastructure. As information technology continues to evolve, the importance of security-by-design, compliance with legislation, and the deployment of associated products and services cannot be overstated.

Some of the key components of a secure computer website and system are having an easy (GUI) Graphical User Interface that reflects some functionalities and has a user-friendly interactive website and gets the job done. The security measures created through the user interface are forget password, login, registration, error messages, MFA, user access control, and many more. The suggested security devices are:

Firewall for Web Applications (WAF):

A web application firewall (WAF) is a software program or security device that guards against common online exploits. It assists in thwarting attacks like SQL injection and cross-site scripting (XSS) by filtering and monitoring HTTP traffic between a web application and the Internet.

TLS/SSL Certifications:

For data to be encrypted while it is being transferred between the user's browser and the web server, Secure Sockets Layer (SSL) or its successor Transport Layer Security (TLS) certificates are essential. This stops man-in-the-middle attacks and eavesdropping.

DDoS Defense Package:

Attacks known as Distributed Denial of Service (DDoS) can cause websites to become unavailable. DDoS mitigation tools and services assist in absorbing and reducing these attacks, ensuring that the website is still accessible even during periods of high traffic.

Intrusion Detection and Prevention Systems (IDPS):

These systems keep an eye on system and/or network activity to look for security policy violations or malicious exploits. They are able to recognize and stop suspicious activity on the website as well as unauthorized access.

Content Security Policy (CSP):

CSP is a security standard that aids in thwarting a number of attacks, including cross-site scripting attacks. It helps reduce the risk of injection attacks and gives web developers the ability to specify which resources are permitted to load.

Two-Factor Authentication (2FA):

By requiring users to provide two forms of identification before accessing an account, 2FA adds an extra layer of security, despite not being a device in the traditional sense. Using a smartphone app, hardware token, or SMS could be required for this.

Solutions for Disaster Recovery and Backup:

It's imperative to regularly backup the data on websites. Having a solid backup and recovery system in place guarantees that the website can be brought back to a secure state in the event of a security incident or data loss.

Headers for Security:

By regulating browser behavior, HTTP security headers like X-Content-Type-Options and Strict-Transport-Security (HSTS) help guard against a variety of online vulnerabilities.

Frequent penetration tests and security audits:

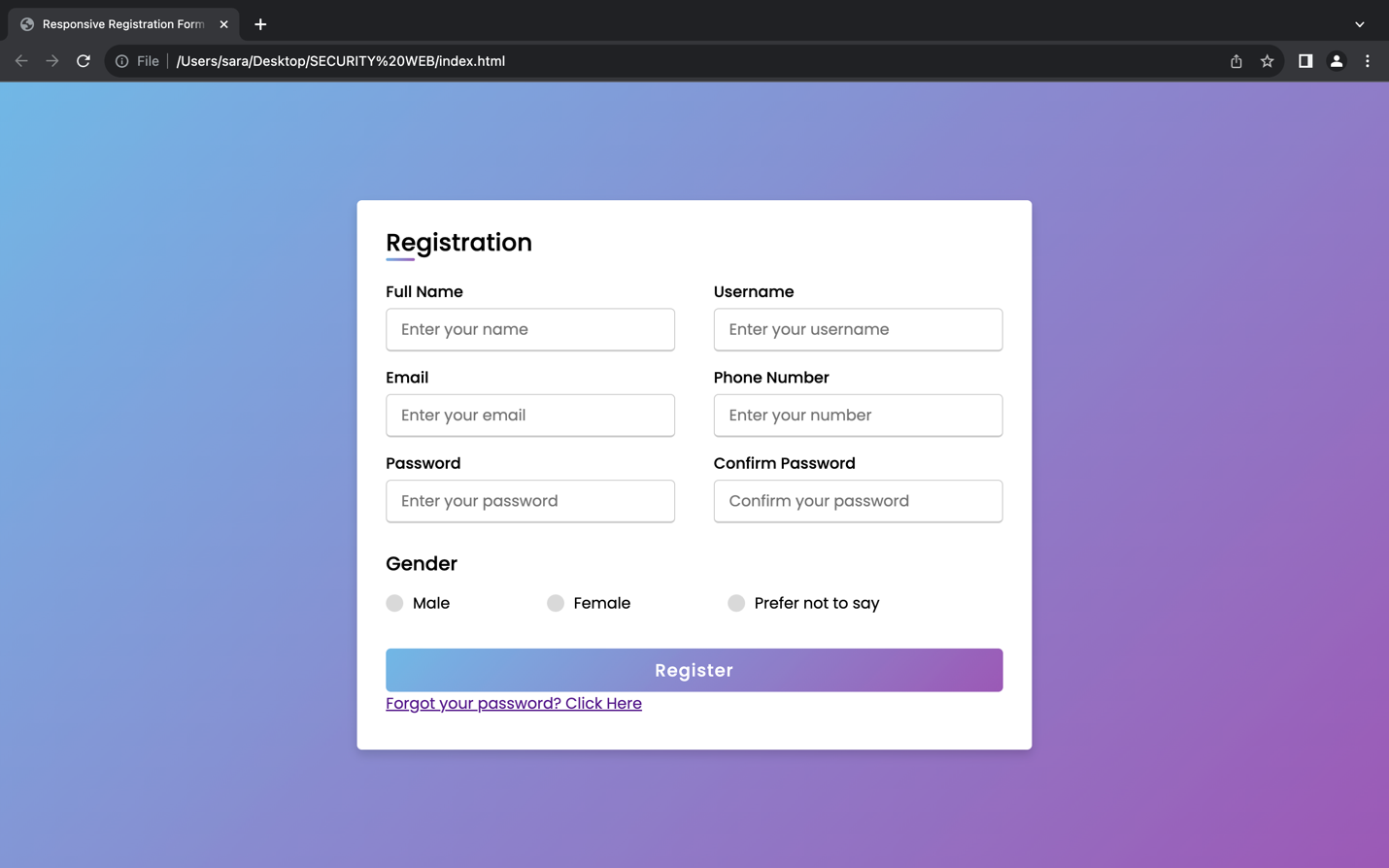
Finding security flaws in the system requires routinely auditing the website's security through penetration testing and security assessments.

Authentication

Authentication is the process of verifying the identity of a user, system, or entity attempting to access a resource. It is the first line of defense against unauthorized access. In the provided code snippets, the registration and login forms illustrate the basic principles of authentication.

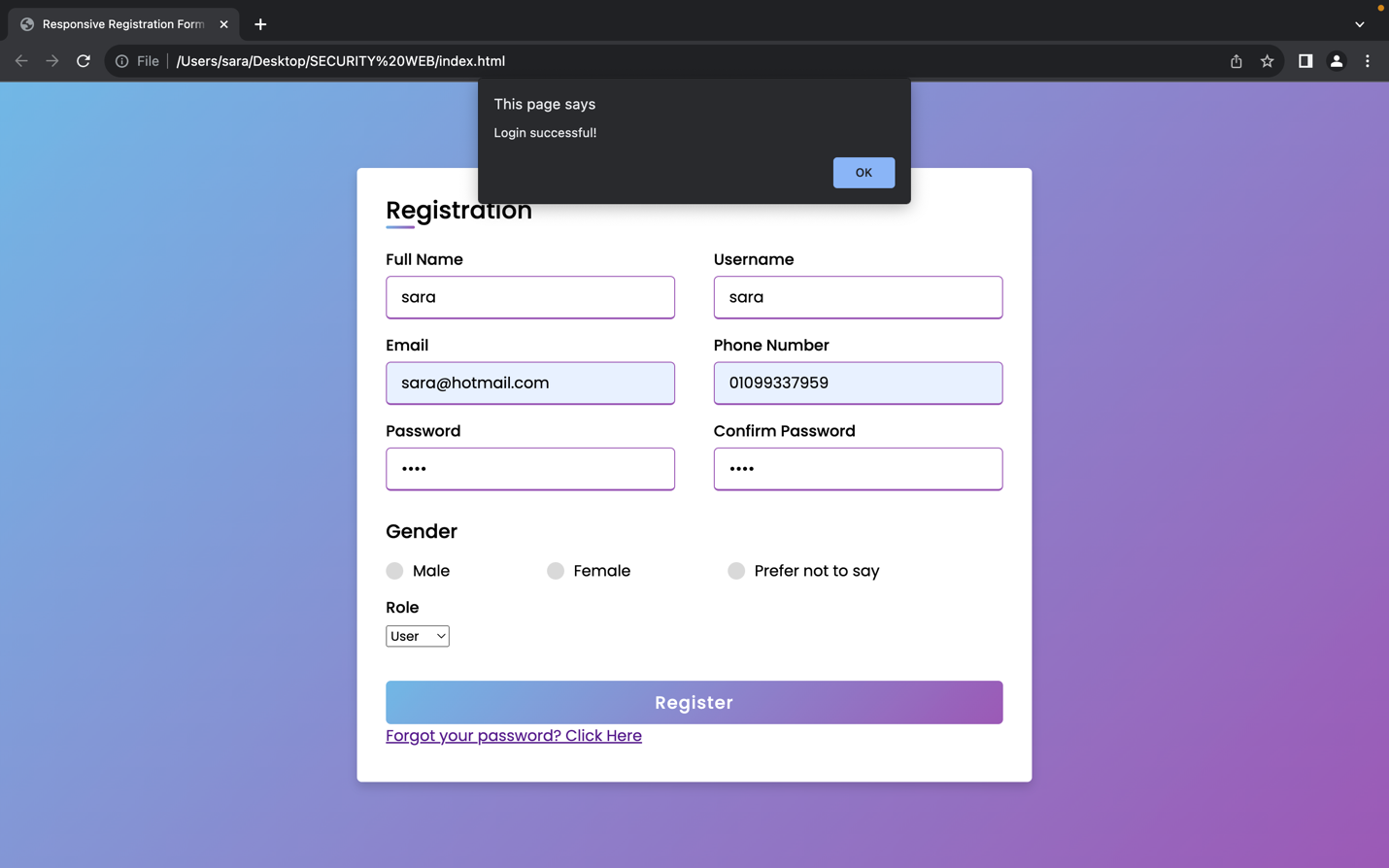
Registration Form: Users are required to provide essential information such as full name, email, phone number, and password to create an account, and a confirm your password when registering to make sure the password is the same as an extra type of verification . The username, a common authentication identifier, is included as well.

Login Form: Users must input their username and password for authentication. If the provided credentials match those stored during registration, access is granted. Here is a screenshot of the working mechanism for both the registration and the login pages.



A screenshot of a computer

Description automatically generated



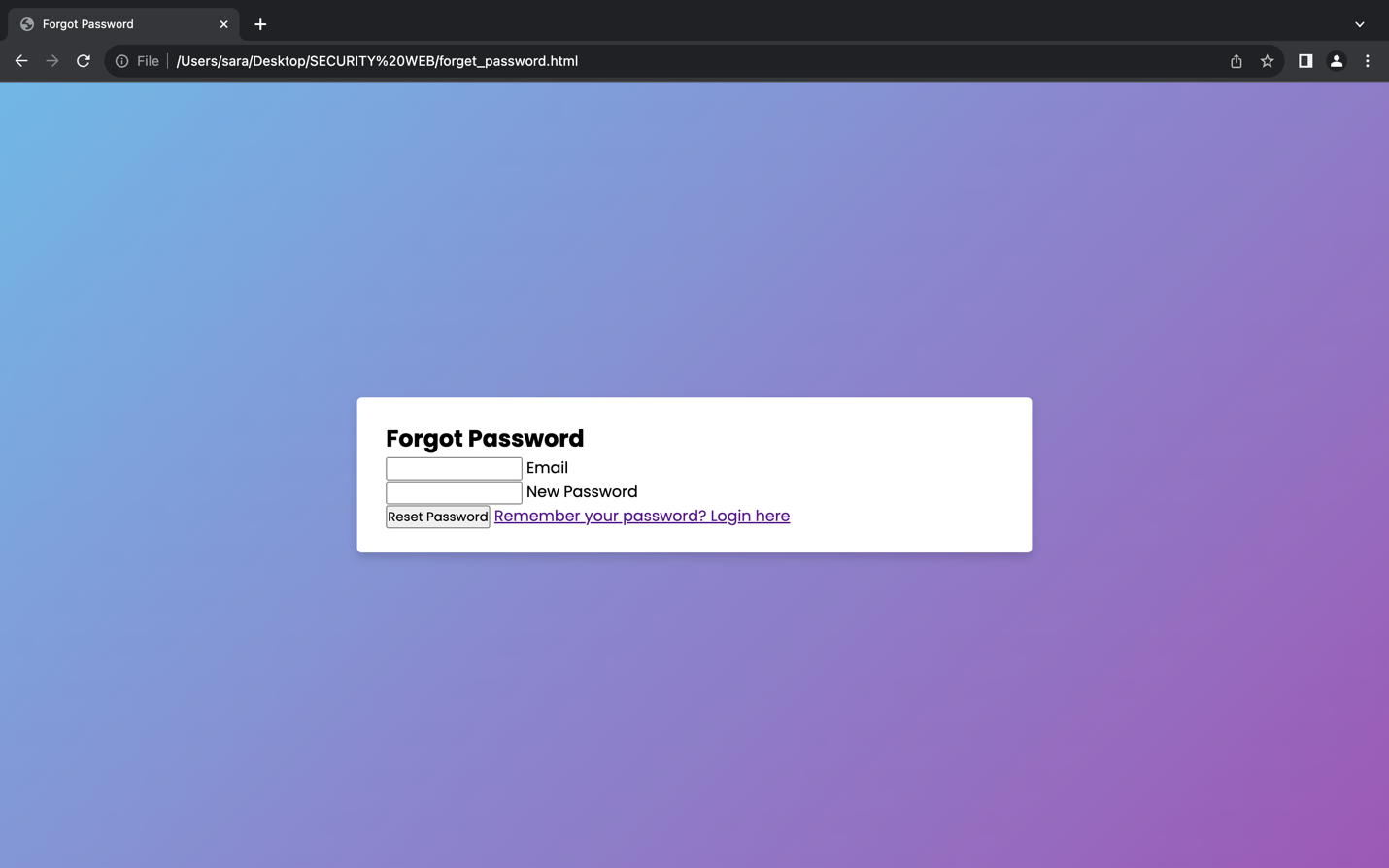
A screenshot of a computer

Description automatically generated

Password protection

The primary purpose of this feature is to allow users to reset their password in the event they forget it. The form prompts users to input their email address and a new password. To enhance security, the use of the "email" input type helps ensure a valid email address is entered. Additionally, the "password" input type ensures that the new password is obscured for privacy.

It's important to note that this implementation focuses on the user interface aspect of the security measure. The actual security extends beyond this HTML code to include server-side validation, secure transmission of data, and robust back-end processes for resetting and storing passwords securely. The reliance on HTTPS (not explicitly mentioned in the code) is assumed for encrypting data in transit. This feature allows the user to get sent an email to reset his own password.



Multi-Factor Authentication (MFA)

MFA enhances security by requiring users to provide two or more verification factors to access an account. The provided code includes an MFA setup section for users who choose to enable this additional layer of security after registration. It checks for the username and password and send a message where invalid or valid credentials as a way to check for this user in the database.

1. MFA Setup

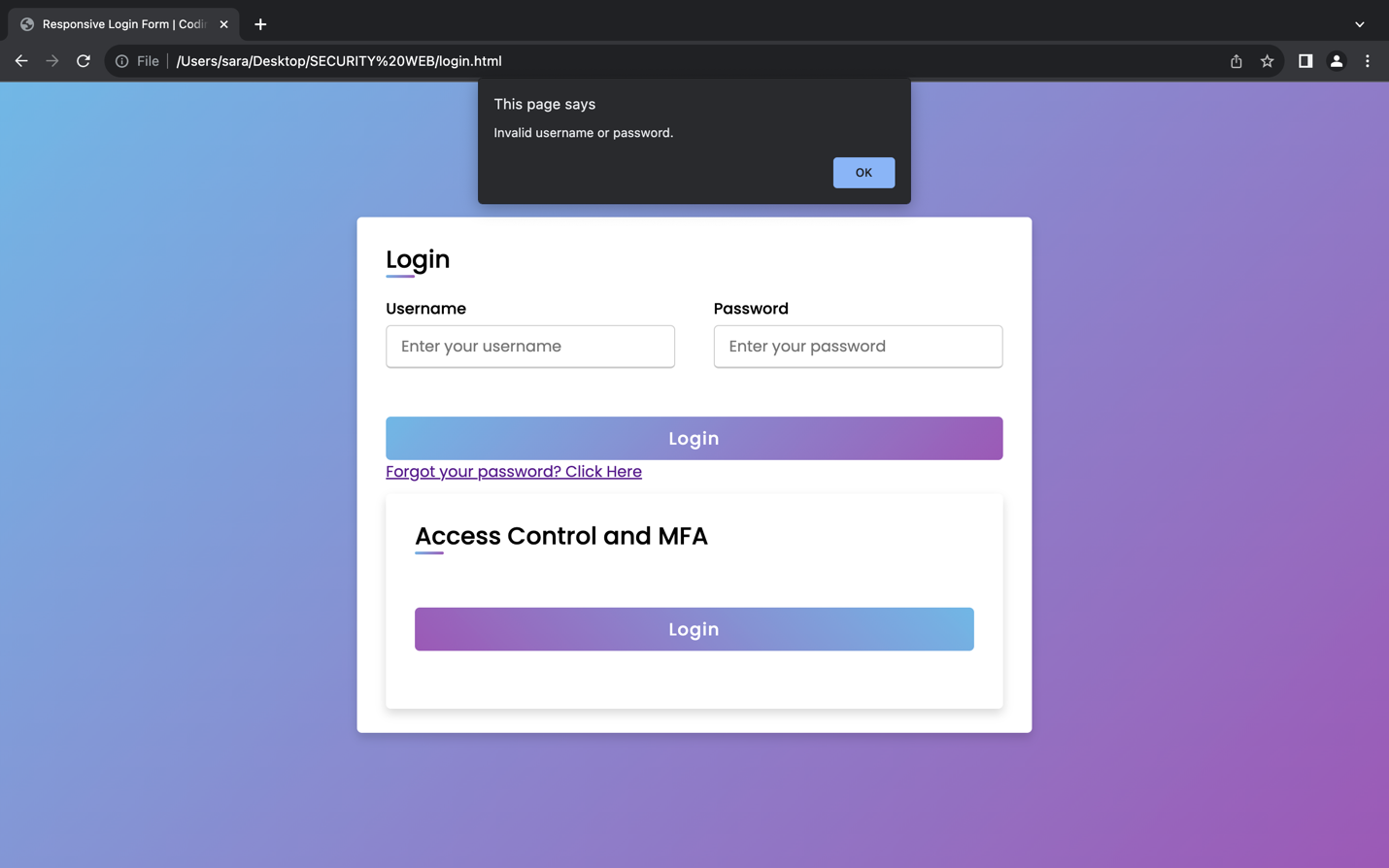
Secret Key Generation: The setupMFA function generates a secret key for MFA. This key is displayed to the user and is crucial for setting up authenticator apps.

1. Enabling MFA

MFA Verification: Users can enable MFA by entering a verification code generated by an authenticator app. The enableMFA function verifies the code against the stored secret key.

A screenshot of a computer

Description automatically generated



Cryptography for Username Encryption

Cryptography, an essential aspect of authentication, is employed to secure sensitive information such as usernames and passwords. In the HTML file, the <script> tag includes the CryptoJS library, enabling the encryption of the username before submitting the registration form.Data Integrityis a cryptographic techniques like hashing and digital signatures verify the integrity of data, ensuring that it has not been tampered with during transmission or storage.

Password Protection is storing hashed passwords instead of plaintext enhances security. Even if a database is compromised, attackers cannot easily retrieve actual user passwords.

The encryptUsername and decryptUsername functions use the AES encryption algorithm to secure the username. This helps protect sensitive user information from unauthorized access.

Here is a small demo for the encryption for the username. When the user registers an encrypted name is created for the user so that if data got leaked it will not be easy to steal the data and understand actually the data without having the secret key that is given when registering.

A screenshot of a computer

Description automatically generated

User Access Control

User Access Control involves managing and restricting user permissions to certain resources. The registration form includes a "Role" dropdown, allowing users to be categorized as either "User" or "Admin." This categorization determines the level of access and privileges a user possesses within the system. User access control is essential for maintaining the principle of least privilege, ensuring users only have the necessary permissions for their roles. The website sends a message if the user is a regular user and chose the role of an admin that they don’t have access to view data as much as an admin. This here is a drop down menu list that allows the customer to choose whether user or admin when creating their credentials.

A grey button with white text

Description automatically generated

Conclusion

In conclusion, the implementation of authentication, MFA, user access control, and cryptography collectively forms a robust security infrastructure for websites. The interconnectedness of these components creates a layered defense against a variety of cyber threats, enhancing the overall security posture. As the digital landscape continues to evolve, websites must stay vigilant and proactive in adopting and adapting security measures. The comprehensive integration of these security features not only protects user data and privacy but also strengthens the reputation and trustworthiness of the website. Investing in robust security measures is not just a necessity but a strategic imperative for any website operating in today's dynamic and interconnected online environment. In addition, the provided code snippets, and demo demonstrate fundamental concepts in authentication, MFA, user access control, and cryptography. These elements collectively contribute to building secure and resilient systems, safeguarding user information and system integrity.

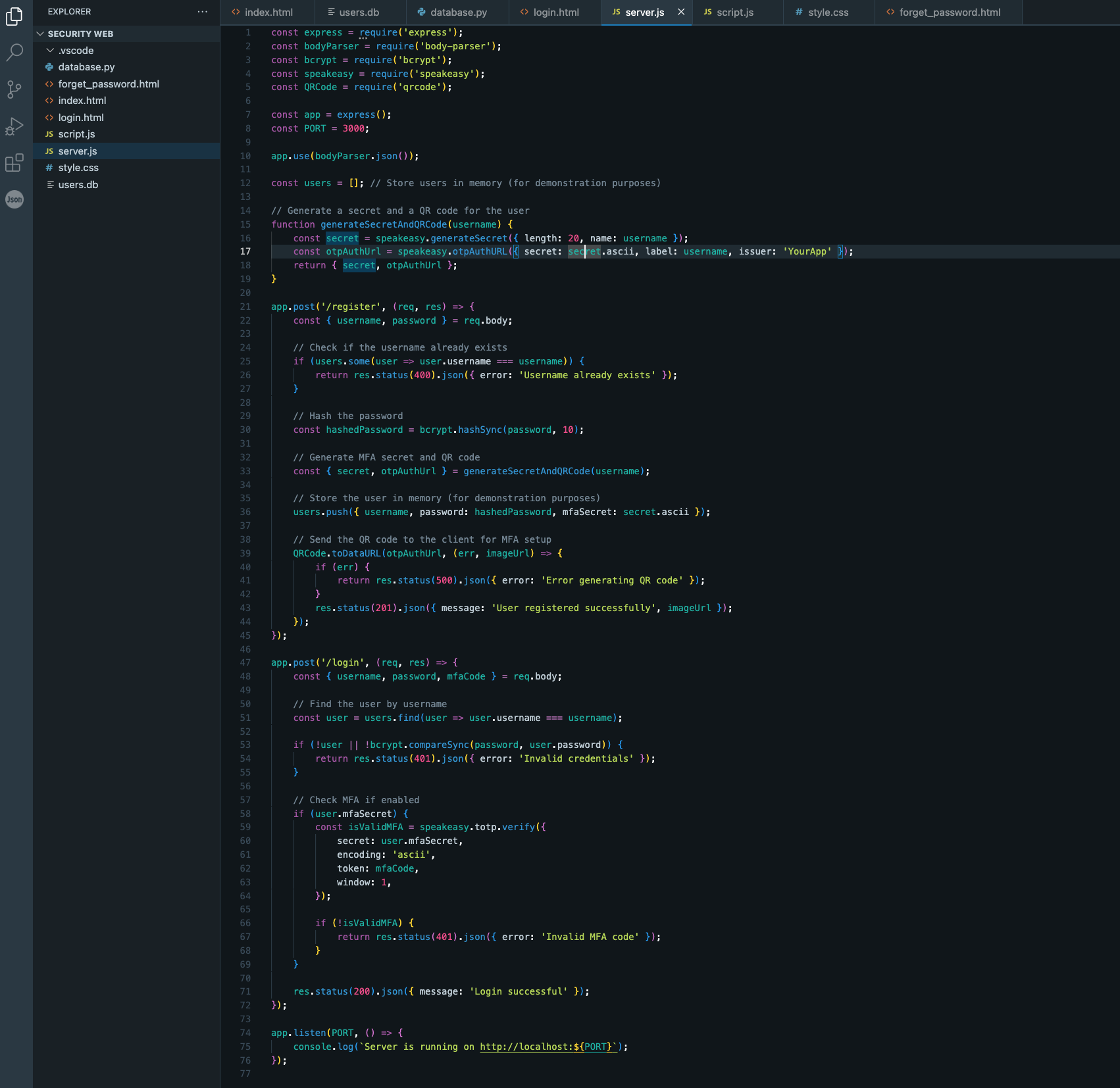
APPENDIX

This is the database connection code with SQLite:

A screenshot of a computer

Description automatically generated

Here are some java script code snippets added for more documentation and understanding.



A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generatedA screenshot of a computer program

Description automatically generatedTop of Form

Bottom of Form