**Hack Me**



Session: 2023 – 2027

**Submitted by:**

Muhammad Talha 2023-CS-12

Anas Faiz 2023-CS-26

Muhammad Salman Anas 2023-CS-06

**Supervised by:**

Dr Faiza Iqbal

Department of Computer Science

**University of Engineering and Technology**

**Lahore Pakistan**

**Hostel-Era**

Contents

[**1.** **Introduction:** 3](#_Toc196684157)

[**1.1.** **Project Title:** 3](#_Toc196684158)

[**1.2.** **Background:** 3](#_Toc196684159)

[**1.3.** **Objective** 3](#_Toc196684160)

[**2.** **Project Overview:** 4](#_Toc196684161)

[**2.1.** **Description** 4](#_Toc196684162)

[**2.2.** **Features** 4](#_Toc196684163)

[**2.3.** **Technologies Used** 5](#_Toc196684164)

[**3.** **Implementation Details:** 5](#_Toc196684165)

[**3.1.** **Tools and Environment** 5](#_Toc196684166)

[**3.2.** **Code Snippets** 6](#_Toc196684167)

[**4.** **User Interface (UI)** 10](#_Toc196684168)

[**5.** **Future Implementations** 14](#_Toc196684169)

# **Introduction:**

## **Project Title:**

**Hack Me – A Hands-on Cybersecurity Training Platform**s.

## **Background:**

We are students of the Computer Science department at the prestigious **University of Engineering and Technology (UET) Lahore**. Currently enrolled in the **4th semester**, we are studying the course **Information Security**. As part of our course project, we were assigned to develop a website that promotes real-world cybersecurity skills. To fulfill this objective, we created an interactive learning platform focused on ethical hacking and web security. The application offers a secure environment where users can explore and defend against common vulnerabilities such as SQL Injection, Cross-Site Scripting (XSS), Cross-Site Request Forgery (CSRF), Open Redirects, Insecure File Uploads, and Server-Side Request Forgery (SSRF).

## **Objective**

The primary objective of our application is to provide a practical platform for users to develop and enhance their cybersecurity skills. The app allows users to practice exploiting and defending against a variety of real-world web vulnerabilities in a safe, controlled environment. It covers common attack vectors such as SQL Injection, Cross-Site Scripting (XSS), Cross-Site Request Forgery (CSRF), Open Redirects, Insecure File Uploads, and Server-Side Request Forgery (SSRF). From a technical perspective, the application is designed to simulate both vulnerable and secure states, enabling users to experience the complete cycle of attack and defense. This practical approach aims to bridge the gap between theoretical knowledge and real-world application in the field of information security.

# **Project Overview:**

## **Description**

Hack Me is an interactive cybersecurity training platform crafted with PHP, HTML, JavaScript, and MySQL, designed to immerse users in the world of web vulnerabilities and defensive techniques. The core objective of this project was to provide a hands-on, offline learning environment where users can experiment with and mitigate common security flaws such as SQL Injection, Cross-Site Scripting (XSS), Cross-Site Request Forgery (CSRF), Open Redirects, Insecure File Uploads, and Server-Side Request Forgery (SSRF). By simulating these vulnerabilities within a controlled system, Hack Me offers an invaluable experience in both offensive and defensive cybersecurity. The project provided us with practical knowledge in building secure web applications, managing databases effectively, and understanding the dynamics of web security, all while utilizing fundamental web development technologies.

## **Features**

* + - **User Authentication:** Separate login systems for users to access different areas of the platform, ensuring customized access for learners.
    - **Vulnerability Simulation:** A variety of real-world web vulnerabilities such as SQL Injection, Cross-Site Scripting (XSS), Cross-Site Request Forgery (CSRF), Open Redirects, Insecure File Uploads, and Server-Side Request Forgery (SSRF) are simulated for practical exploitation.
    - **Defensive Mechanisms:** After exploiting vulnerabilities, users can apply security measures and defend against attacks, helping reinforce the theory behind secure coding practices.
    - **Database Management:** MySQL is used for storing vulnerability scenarios, user progress, ensuring persistent data storage and easy access to practice modules.
    - **Interactive Dashboard:** Users can track their progress, review previously exploited vulnerabilities, and revisit challenges.
    - **Responsive Design:** The platform is designed to adapt smoothly across different screen sizes, providing an optimal experience for both desktop and mobile users.
    - **Efficient State Management:** Through XAMPP and PHP, the system ensures real-time updates and seamless performance for dynamic learning sessions.

## **Technologies Used**

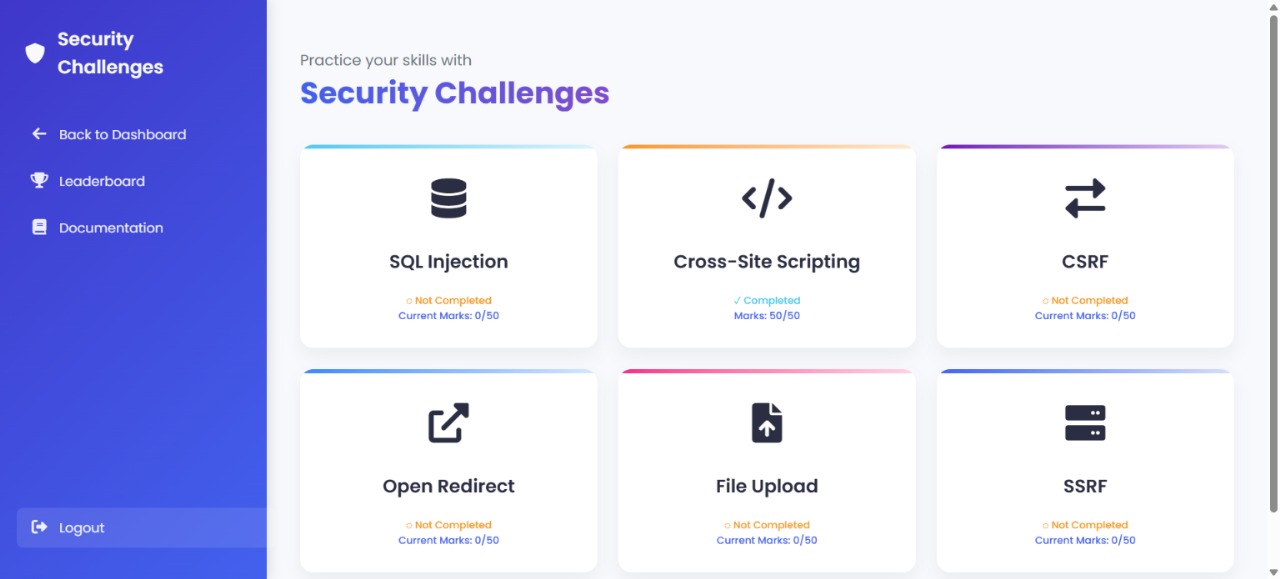
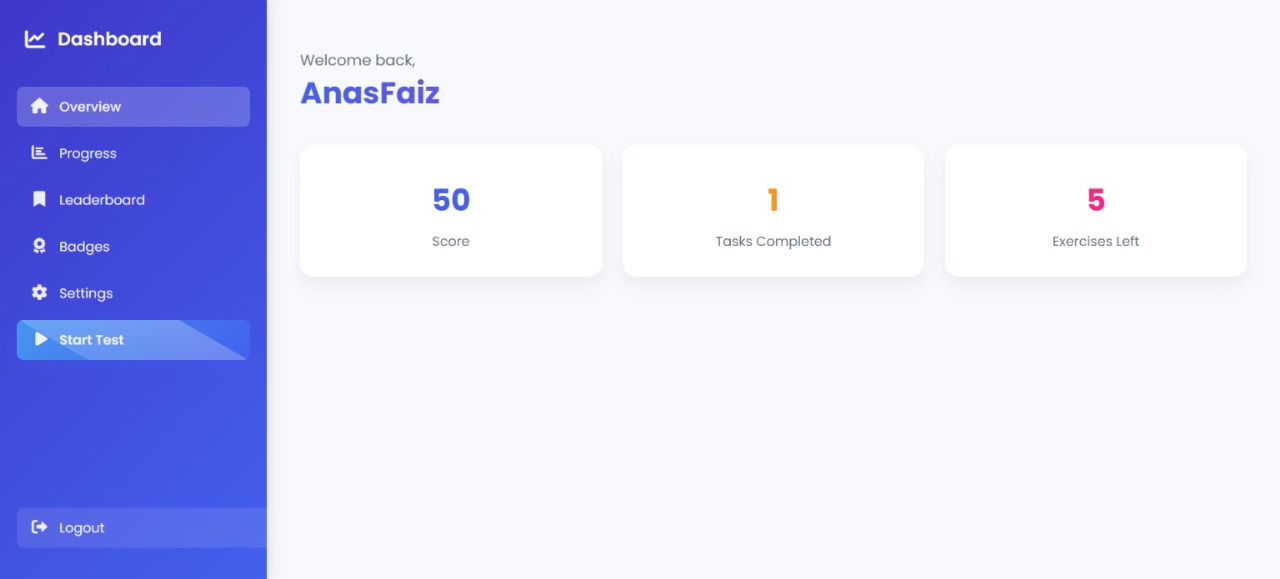
This project is developed using PHP, HTML, JavaScript, and MySQL, with XAMPP serving as the local server environment. It employs MySQL for storing all data, including user credentials, vulnerabilities, and progress. The platform is designed with a focus on modularity, allowing users to practice both exploiting and defending web vulnerabilities in a structured and secure manner. The application utilizes clean code practices for dynamic content updates and smooth user interactions. The system architecture ensures scalability, with easy integration of additional vulnerability modules and enhanced functionality as needed.

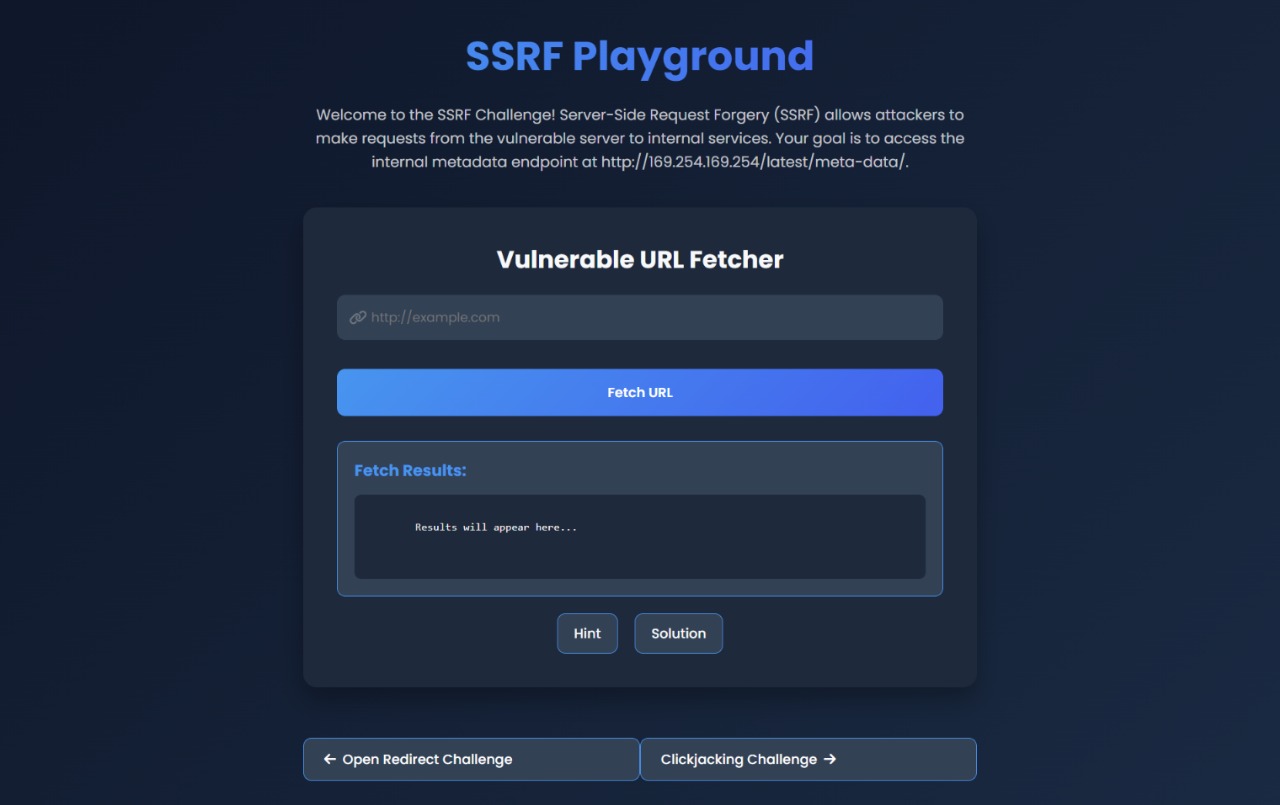
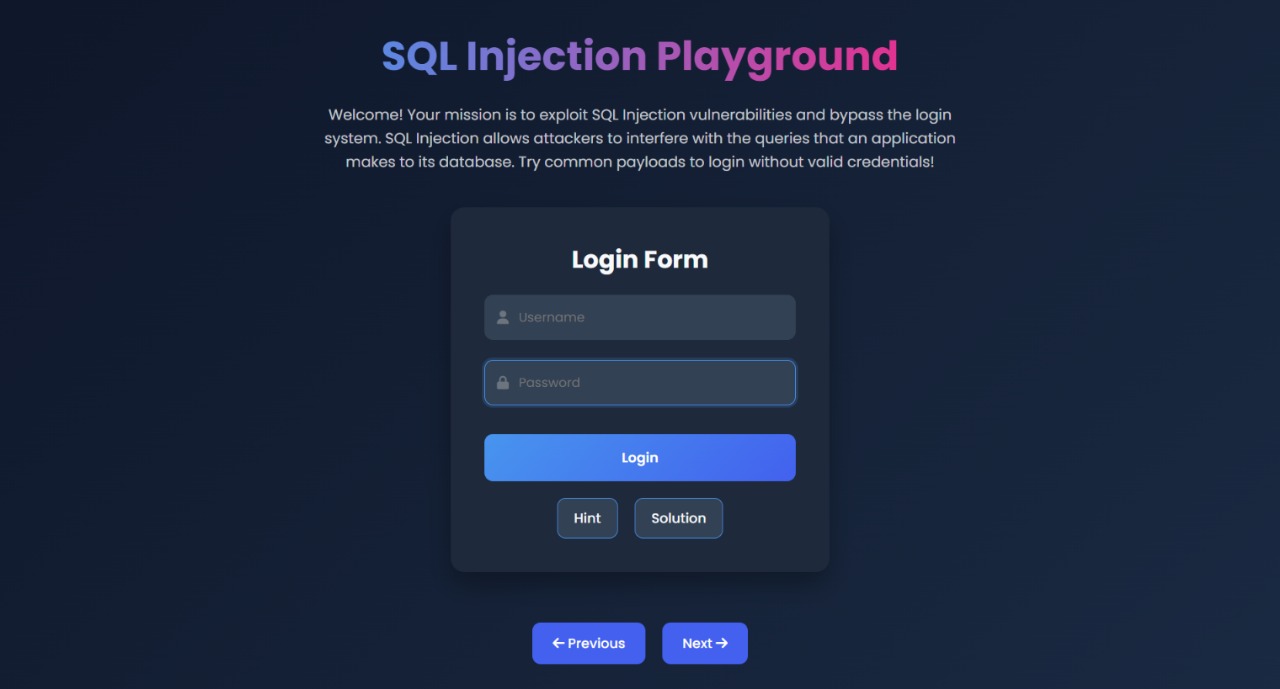
# **Implementation Details:**

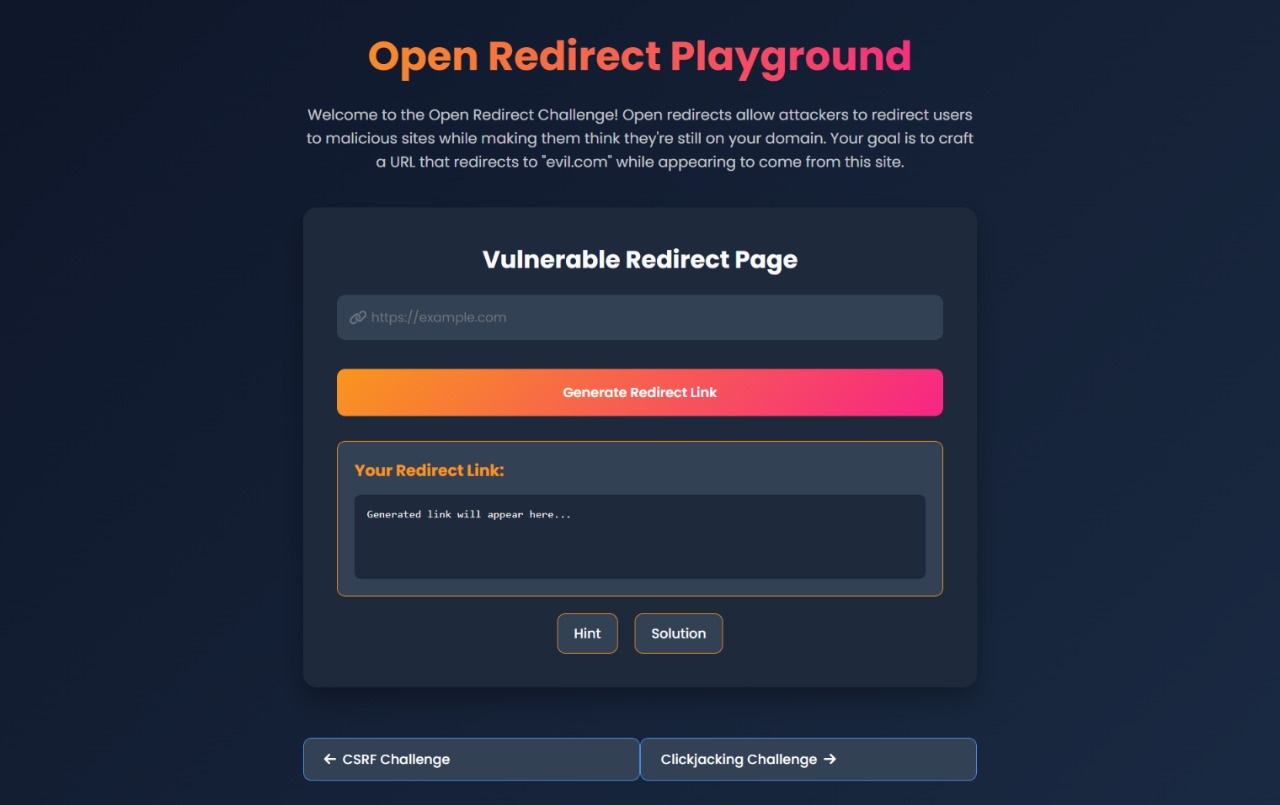
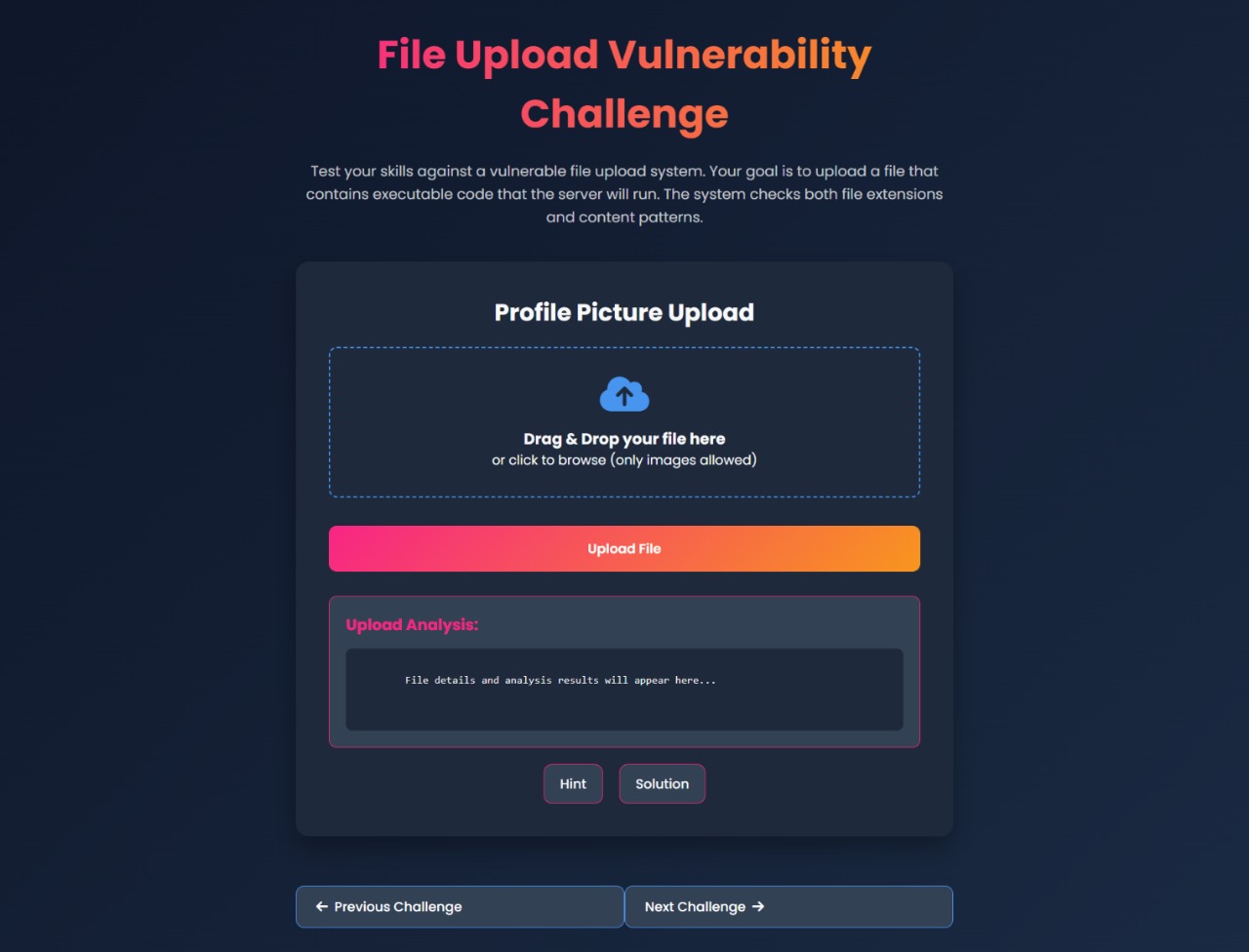
## **Tools and Environment**

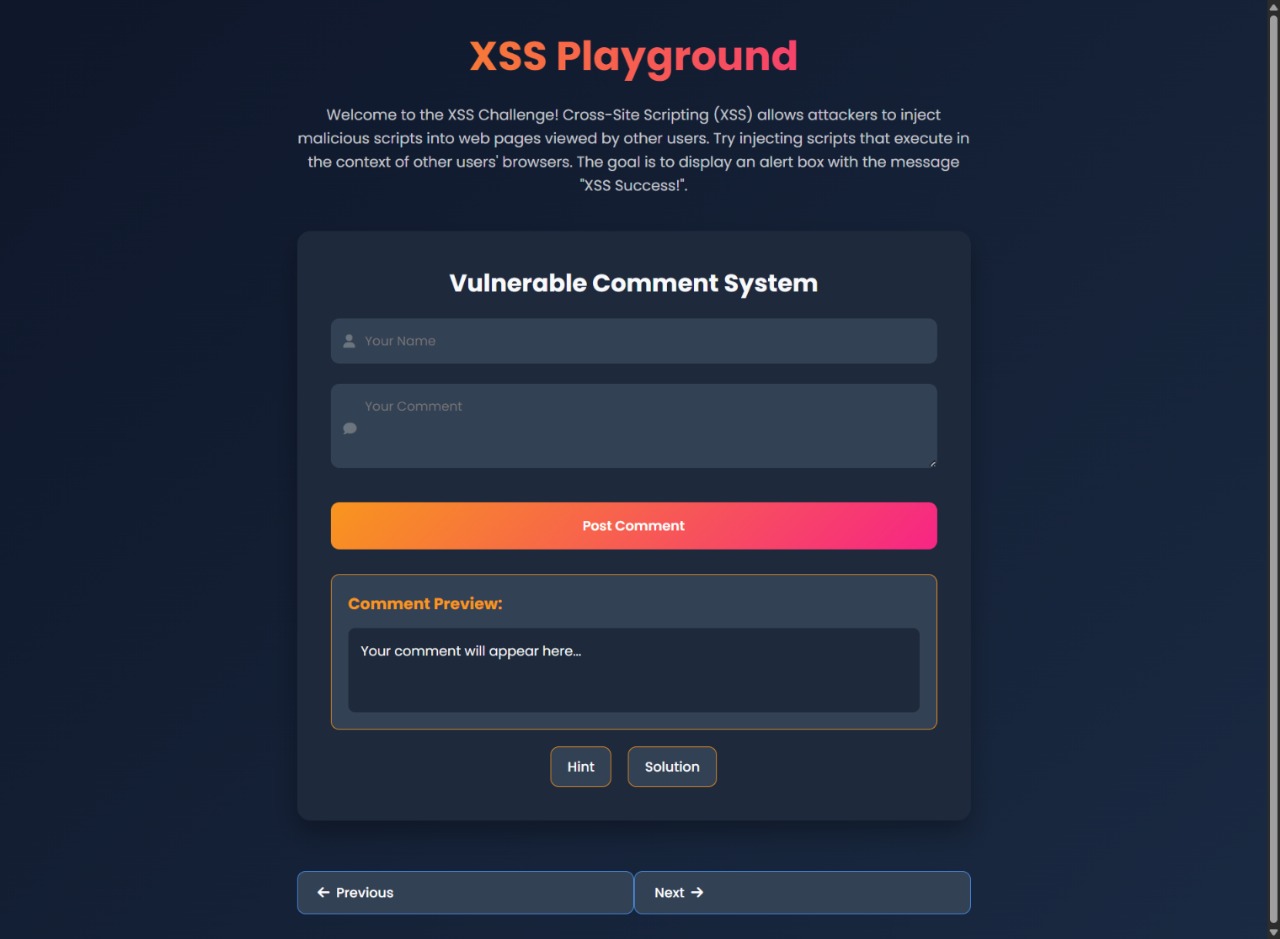
The development of this project was carried out using **PHP, HTML, JavaScript**, and **MySQL**, with **XAMPP** serving as the local server environment. The development environment was set up using a local server on **XAMPP** for seamless testing and deployment. **Visual Studio Code** was used as the primary code editor, with essential extensions such as **PHP Intelephense**, **MySQL**, and **Live Server** for efficient development. **Git** was employed for version control, ensuring smooth collaboration and code management throughout the project lifecycle. The app was primarily tested on a local server, with real-time updates and interactions simulated for the learners.

# **User Interface (UI)**











# **Future Implementations**

* + - * **Cloud Sync with Firebase**: Integrate Firebase for online data storage, enabling real-time synchronization of user progress, and vulnerability modules across multiple devices.
      * **Push Notifications**: Implement real-time notifications to alert users about their progress, new vulnerabilities to practice, and upcoming challenges or updates from administrators.
      * **Advanced Reporting and Analytics**: Provide detailed reports and analytics on user activity, successful exploit attempts, defense strategies, and areas needing improvement, helping users track their learning journey.
      * **Multilingual Support**: Offer multi-language support to make the platform accessible to a wider audience and enhance global usability.