Nama : Latief Reswandana

Kelas : III RPLK

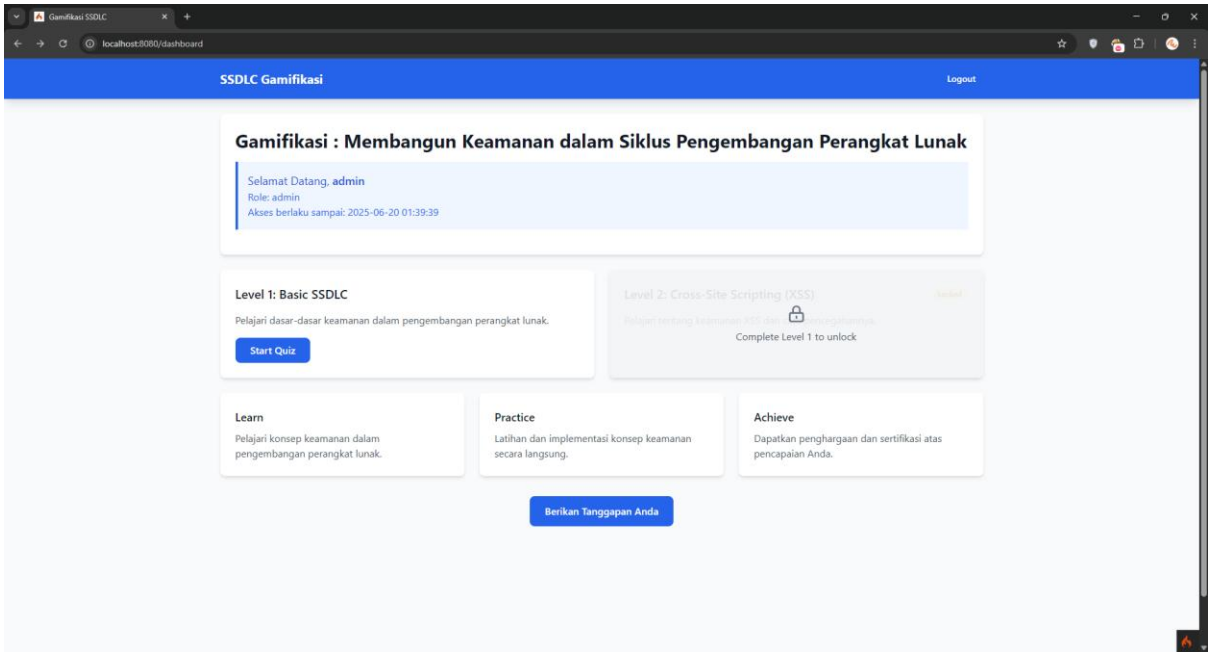
Soal : 3

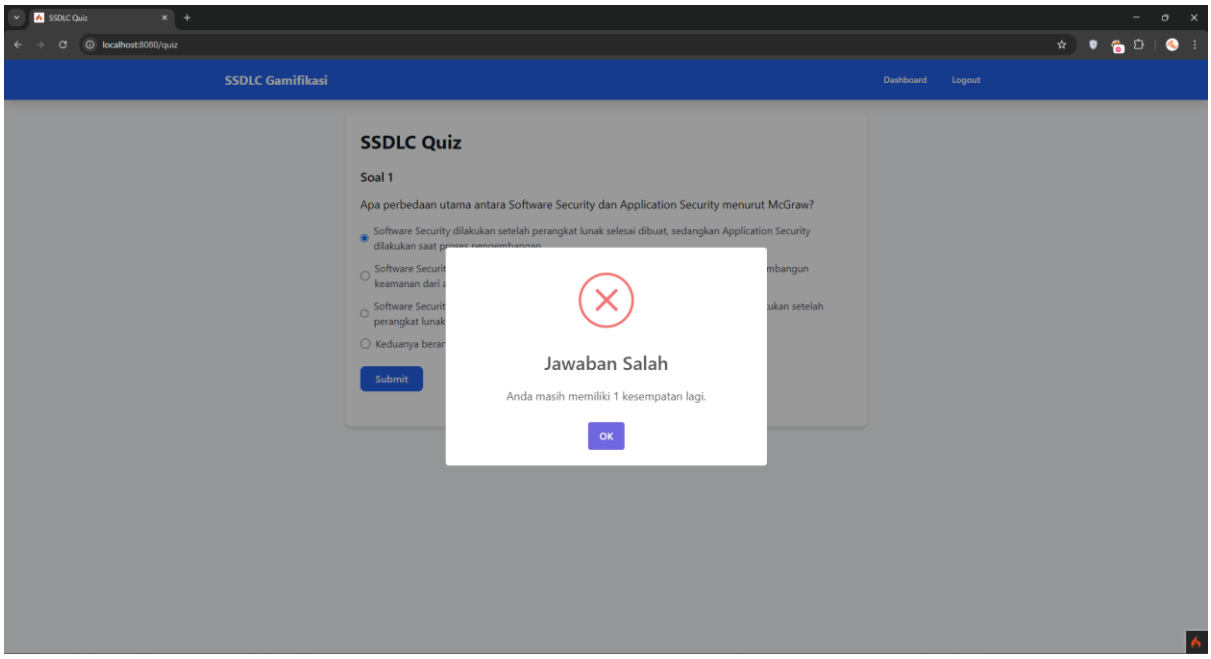
Aplikasi yang saya kembangkan berfokus pada edukasi interaktif mengenai keamanan dalam siklus hidup pengembangan perangkat lunak (Secure SDLC). Melalui pendekatan *gamifikasi*, pengguna diajak menjelajahi 2 level utama dan 1 level rahasia yang penuh tantangan.

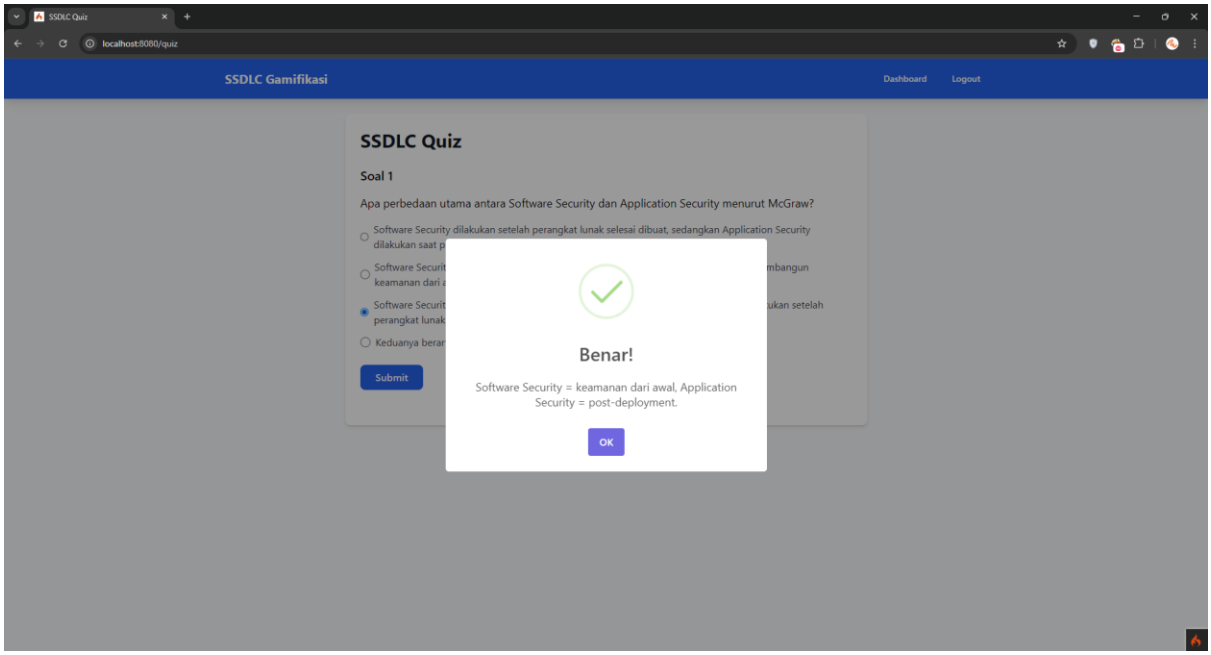
- Level 1 memperkenalkan konsep dasar keamanan dalam SDLC, memberikan pemahaman tentang pentingnya membangun keamanan sejak tahap awal pengembangan.
- Level 2 mengajak pengguna mengenal lebih dalam salah satu ancaman nyata dalam dunia aplikasi, yaitu Cross-Site Scripting (XSS), dengan skenario interaktif dan studi kasus sederhana.
- Level Tersembunyi menjadi tantangan spesial: pengguna diminta melakukan *pentest* sederhana untuk menemukan dan mengekstrak sebuah *flag* rahasia—uji nyali bagi mereka yang merasa cukup tangguh!

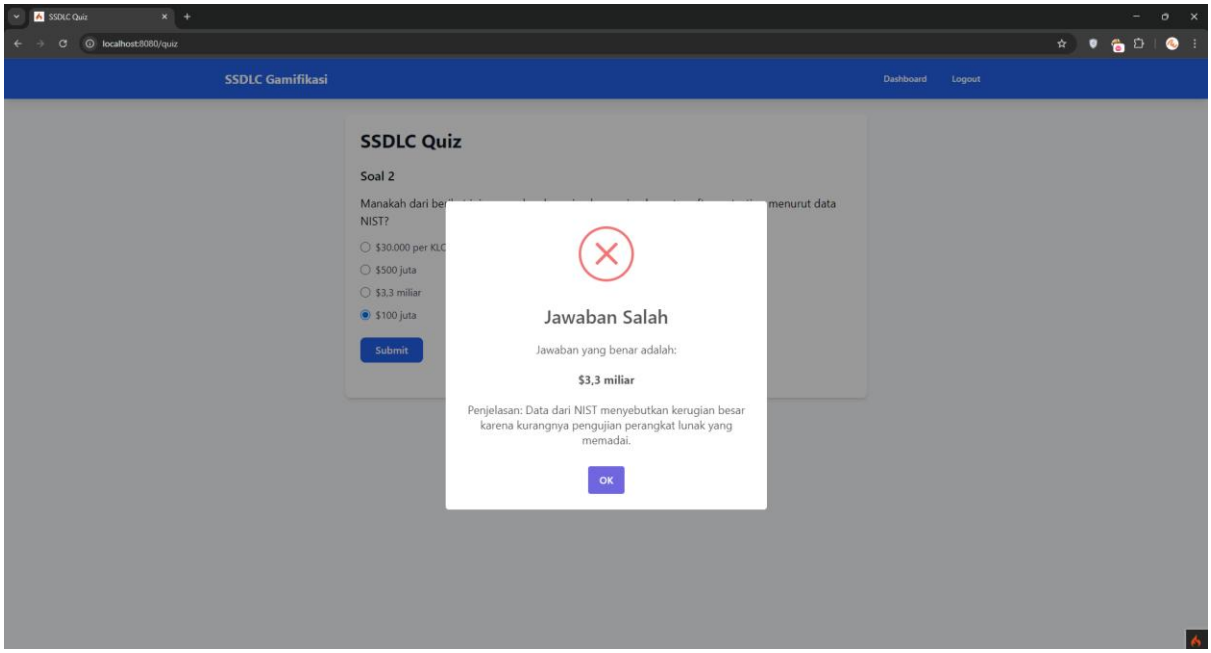
Dengan konsep ini, saya berharap pengguna tidak hanya memahami teori, tetapi juga dapat merasakan langsung pentingnya keamanan dalam setiap fase pengembangan perangkat lunak.

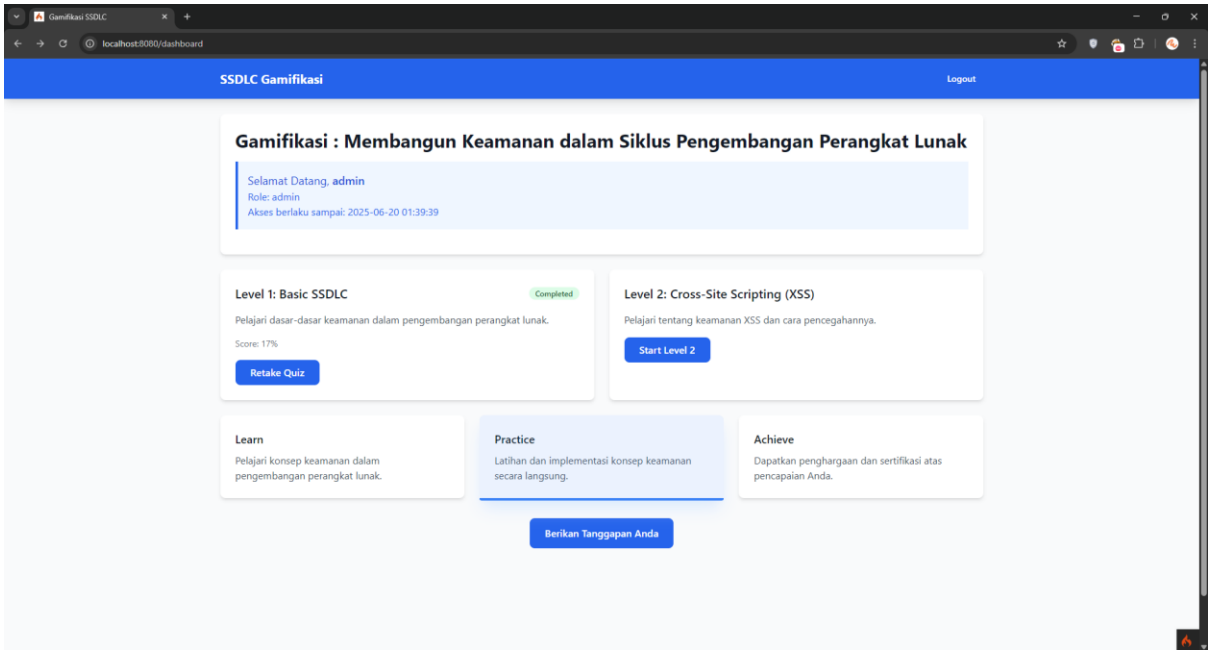
Dashboard :

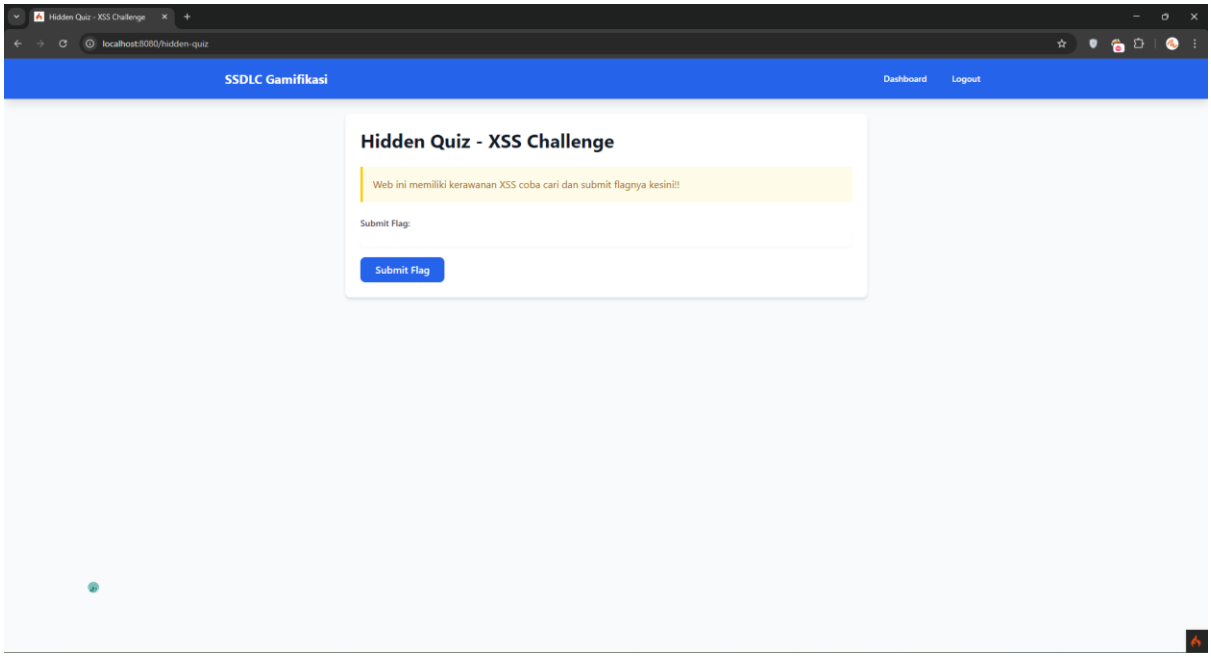

Gamifikasi SSDLC
localhost:8080/dashboard
SSDLC Gamifikasi
Logout
Gamifikasi : Membangun Keamanan dalam Siklus Pengembangan Perangkat Lunak
Selamat Datang, admin
Role: admin
Akses berlaku sampai: 2025-06-20 01:39:39
Level 1: Basic SSDLC
Pelajari dasar-dasar keamanan dalam pengembangan perangkat lunak.
Start Quiz
Level 2: Cross-Site Scripting (XSS)
Complete Level 1 to unlock
Learn
Pelajari konsep keamanan dalam pengembangan perangkat lunak.
Practice
Latihan dan implementasi konsep keamanan secara langsung.
Achieve
Dapatkan penghargaan dan sertifikasi atas pencapaian Anda.
Berikan Tanggapan Anda

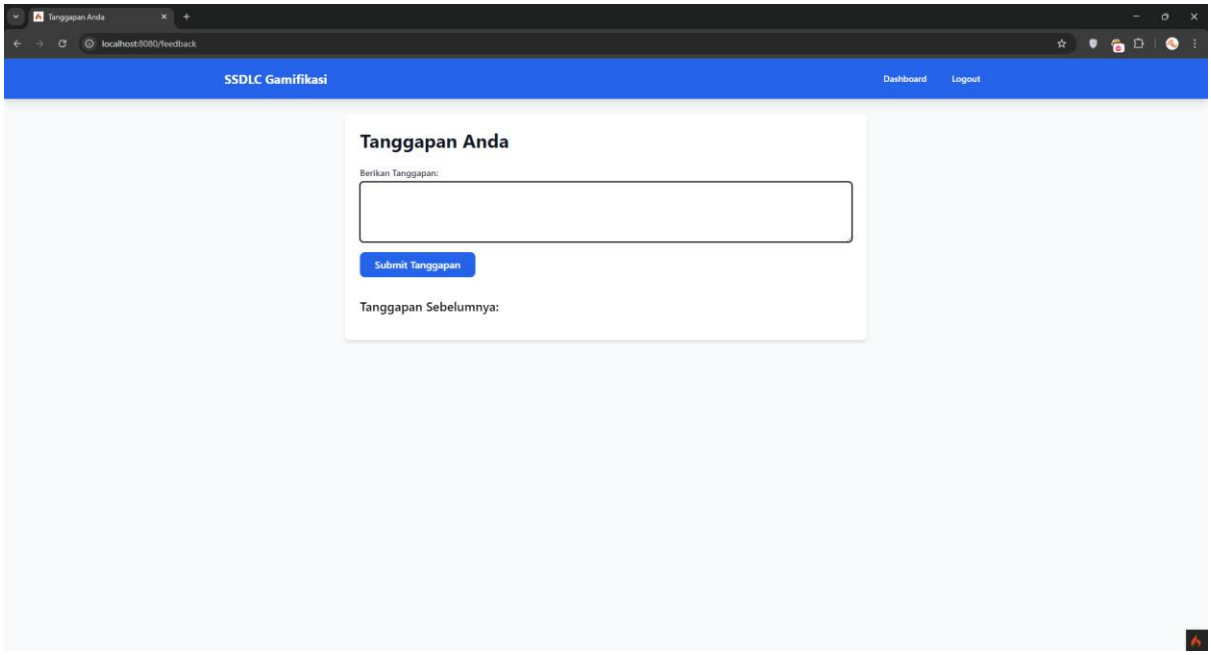

SSDLC Quiz
localhost:8080/quiz
SSDLC Gamifikasi
Dashboard
Logout
SSDLC Quiz
Soal 1
Apa perbedaan utama antara Software Security dan Application Security menurut McGraw?
Submit
Jawaban Salah
Anda masih memiliki 1 kesempatan lagi.
OK


SSDLC Quiz
localhost:8080/quiz
SSDLC Gamifikasi
Dashboard
Logout
SSDLC Quiz
Soal 1
Apa perbedaan utama antara Software Security dan Application Security menurut McGraw?
Submit
Benar!
Software Security = keamanan dari awal, Application Security = post-deployment.
OK

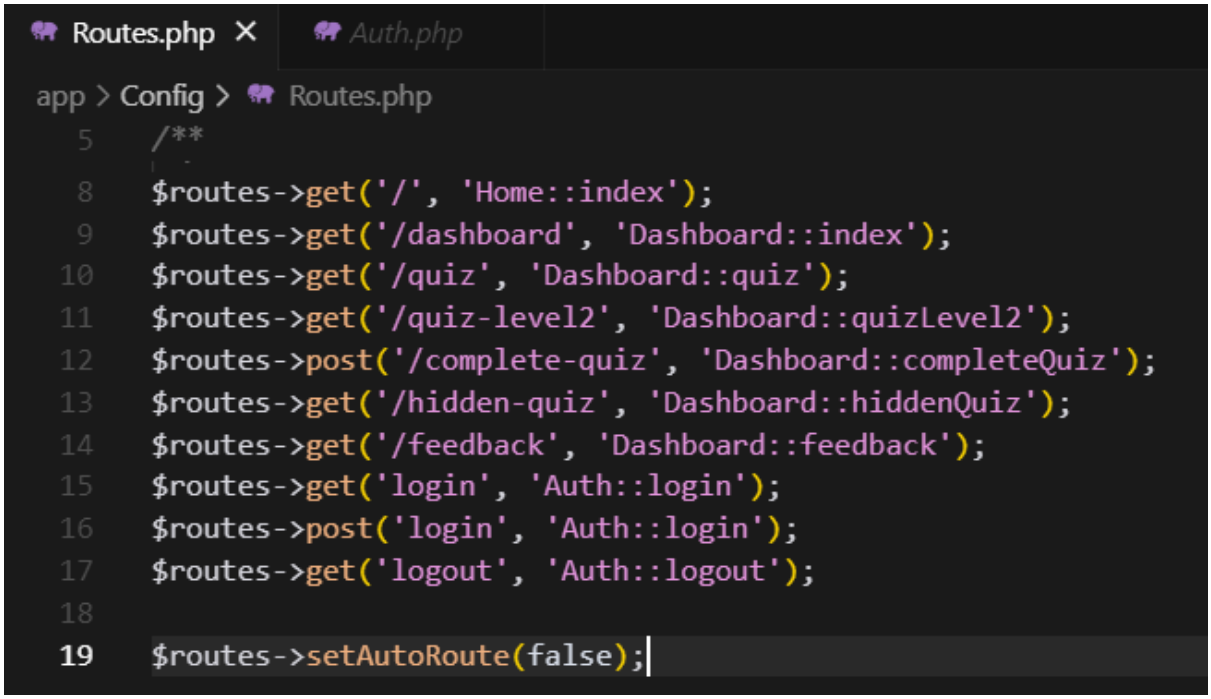

SSDLC Quiz
localhost:8080/quiz
SSDLC Gamifikasi
Dashboard
Logout
SSDLC Quiz
Soal 2
$500 juta
$3,3 miliar
$100 juta
Submit
Jawaban Salah
Jawaban yang benar adalah:
$3,3 miliar
Penjelasan: Data dari NIST menyebutkan kerugian besar karena kurangnya pengujian perangkat lunak yang memadai.
OK

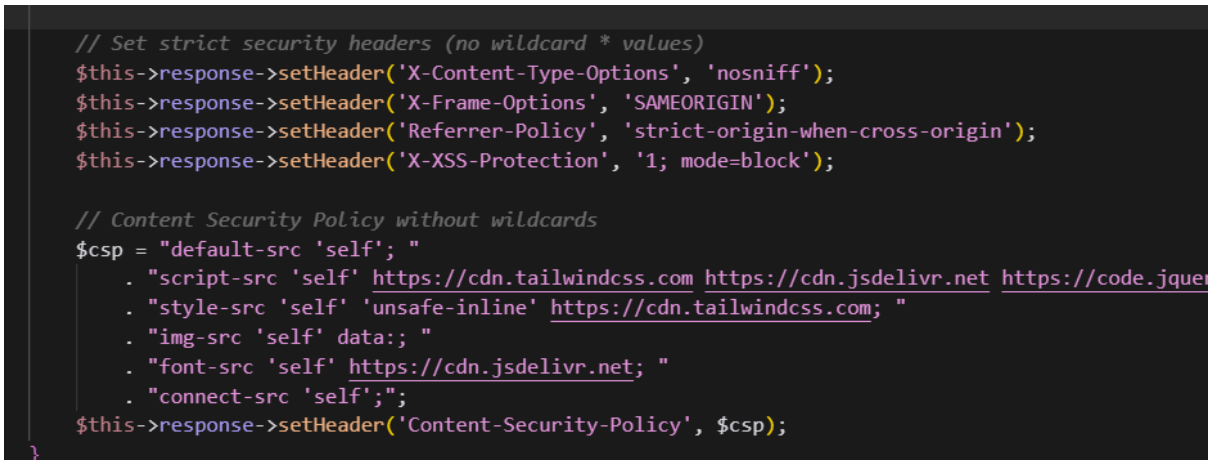

Gamifikasi SSDLC
localhost:8080/dashboard
SSDLC Gamifikasi
Logout
Gamifikasi : Membangun Keamanan dalam Siklus Pengembangan Perangkat Lunak
Selamat Datang, admin
Role: admin
Akses berlaku sampai: 2025-06-20 01:39:39
Level 1: Basic SSDLC
Completed
Pelajari dasar-dasar keamanan dalam pengembangan perangkat lunak.
Score: 17%
Retake Quiz
Level 2: Cross-Site Scripting (XSS)
Pelajari tentang keamanan XSS dan cara pencegahannya.
Start Level 2
Learn
Pelajari konsep keamanan dalam pengembangan perangkat lunak.
Practice
Latihan dan implementasi konsep keamanan secara langsung.
Achieve
Dapatkan penghargaan dan sertifikasi atas pencapaian Anda.
Berikan Tanggapan Anda


Hidden Quiz - XSS Challenge
localhost:8080/hidden-quiz
SSDLC Gamifikasi
Dashboard
Logout
Hidden Quiz - XSS Challenge
Web ini memiliki kerawanan XSS coba cari dan submit flagnya kesini!!
Submit Flag:
Submit Flag

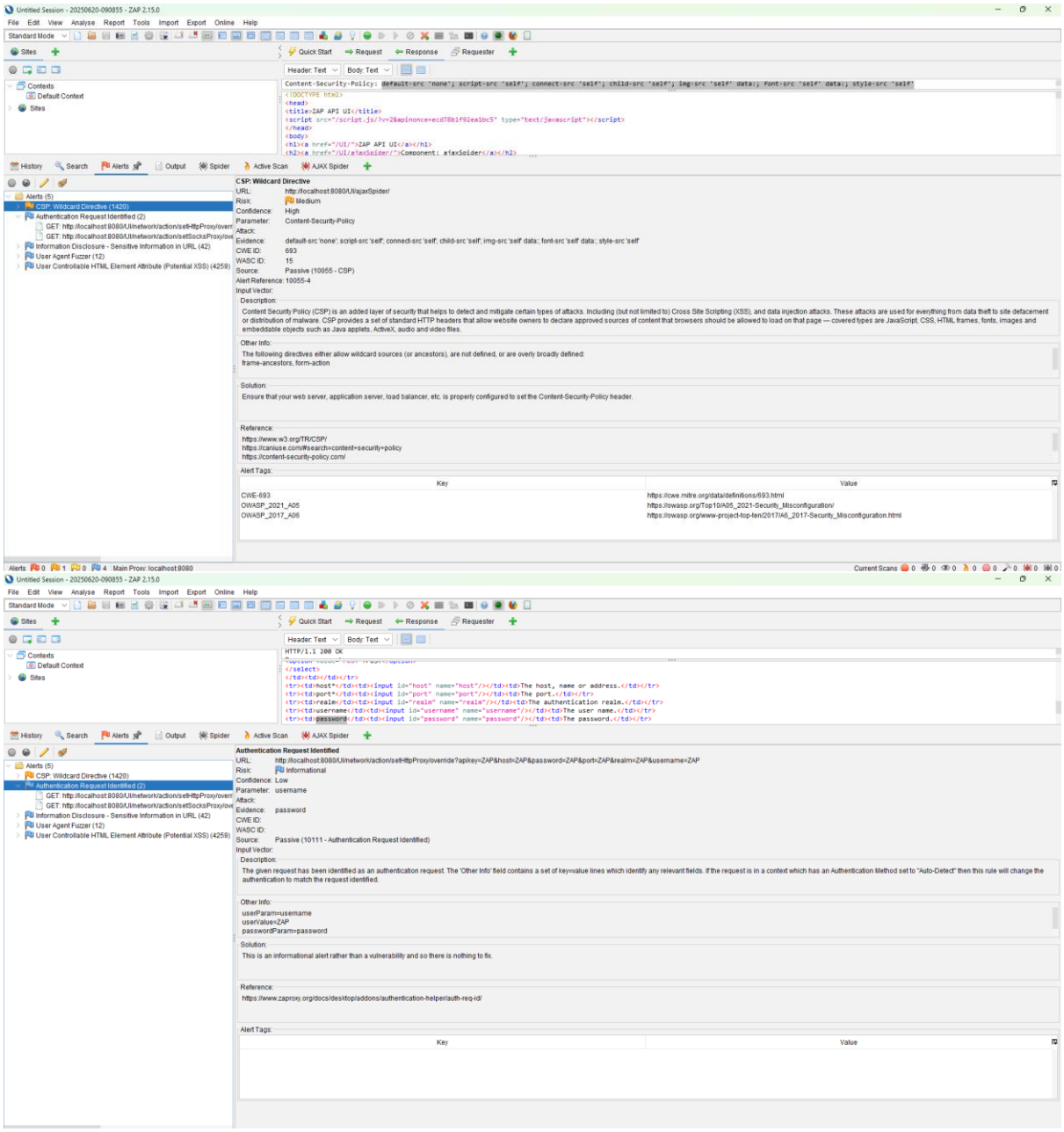

Tanggapan Anda
localhost:8080/feedback
SSDLC Gamifikasi
Dashboard
Logout
Tanggapan Anda
Berikan Tanggapan:
Submit Tanggapan
Tanggapan Sebelumnya:

Dari hasil dynamic analysis menggunakan ZAP ditemukan 1 kategori Medium warning dengan 1420 percobaan, dan 3 warning informal. Sebelumnya terdapat 1 kategori hard yaitu cloud metadata leak potential pada http://localhost:8080/latest/meta-data/. Tapi saya tangani dengan membatasi dengan setAutoRoute(false);.





Setelah di patch hasil scan jadi seperti ini



1. CSP: Wildcard Directive (1420)

- URL: http://localhost:8080/UI/ajaxSpider/
- Risiko: Medium
- Confidence: High
- Masalah:
    - Header Content-Security-Policy (CSP) terlalu umum atau menggunakan wildcard
- Solusi:
    - Konfigurasikan CSP secara lebih spesifik.
    - Hindari wildcard (*) dan gunakan domain yang terpercaya.

2. Authentication Request Identified (2)

- URL: http://localhost:8080/UI/network/action/setHttpProxy/override?apikey=ZAP&username=ZAP&password=ZAP
- Risiko: Informational
- Confidence: Low
- Masalah:
    - ZAP mengidentifikasi request yang mengandung parameter username dan password.
    - Ini hanya peringatan untuk menunjukkan request otentikasi — bukan kerentanan langsung.
- Solusi:
    - Tidak perlu tindakan, kecuali jika dikombinasikan dengan kelemahan lain seperti credential leak.
    - Gunakan metode otentikasi yang aman dan hindari menyimpan kredensial di URL.

3. User Agent Fuzzer (12)

- URL: http://localhost:8080/
- Risiko: Informational
- Masalah:
    - ZAP melakukan pengujian dengan user-agent berbeda (misalnya, bot, crawler, atau browser lama).
    - Tidak ditemukan perbedaan perilaku yang signifikan dari server.
- Solusi:

    - Tidak ada masalah besar yang perlu diperbaiki.
    - Namun, pastikan tidak ada akses istimewa berdasarkan user-agent tanpa validasi tambahan.

4. User Controllable HTML Element Attribute (Potential XSS) (4259)

- URL: http://localhost:8080/UI/alertView/alertCountsByRisk/...&url=https://zap.example.com
- Risiko: Informational
- Confidence: Low
- Masalah:
    - Parameter user-supplied (apikey, url, dll) bisa ditampilkan dalam atribut HTML.
    - Bisa berpotensi XSS jika tidak disanitasi dengan benar.
- Solusi:
    - Sanitasi dan validasi seluruh input dari user sebelum dimasukkan ke dalam HTML.
    - Gunakan library escaping seperti htmlspecialchars() di PHP atau encoding pada framework modern.