



Insecure by Design: A Hands-On Learning Platform for Web Security Training

Progress Report 1

CSIS 4495-003 | Applied Research Project

Winter 2026

Instructor: Arasanipalai Kandhadai, Padmapriya

Work Log

Allwyn Mascarenhas

Date	Number of Hours	Description of Work
15 th Jan	2	Conducted initial research on OWASP Top 10 vulnerabilities and reviewed documentation for DVWA, WebGoat, and Juice Shop. Compared their learning approaches and identified opportunities for a simpler, framework-based platform.
20 th Jan	1	Participated in instructor-led project consultation. Discussed project direction, technology stack, and how to embed vulnerabilities safely within a Spring Boot MVC application.
21 st Jan	1.5	Co-drafted the pre-proposal for instructor approval. Summarized research findings and outlined the initial problem definition, research questions, and project scope.
22 nd Jan	1	Researched Spring Boot security misconfigurations and common vulnerability patterns (e.g., insecure authentication flows, improper input validation). Documented potential implementation strategies.
23 rd Jan	1	Attended scheduled Friday team meeting. Finalized task distribution and reviewed proposal template.
23 rd Jan	1.5	Created and organized the GitHub Project Kanban board. Added tasks for proposal writing, research, vulnerability implementation, and documentation. Assigned responsibilities and set up workflow columns.
24 th Jan	2	Collaborated with Thilan on drafting the full project proposal. Wrote sections on scope, methodology, and technology stack. Ensured the proposal met the rubric requirements for introduction and research design.
25 th Jan	2	Continued refining the proposal independently. Reviewed the draft for clarity, strengthened the methodology section, and added additional justification for the selected technology stack.

Date	Number of Hours	Description of Work
26 th Jan	1	Reviewed and refined the proposal draft. Added details to the methodology and expected results sections. Submitted the document to Blackboard and GitHub.
27 th Jan	1	Participated in in-class proposal check-in. Instructor requested adding industry data to justify the project. Reviewed Fortinet's 2024 Skills Gap Report and incorporated the statistic that 58% of IT decision-makers cite lack of hands-on skills as a top cause of breaches. Updated the introduction accordingly.
31 st Jan	1	Attended scheduled Friday team meeting. Discussed Week 5 implementation tasks, including setting up the Spring Boot skeleton and configuring the H2 database.
1 st Feb	1.5	Attended scheduled Saturday team meeting. Reviewed architecture outline and refined backend workflow diagrams. Finalized vulnerability placement strategy for SQL Injection, Broken Authentication, and IDOR.
3 rd Feb	1	Began setting up backend project structure. Created initial controller and service templates to prepare for SQL Injection implementation.
4 th Feb	1.5	Completed Spring Boot skeleton setup. Verified project builds successfully, configured H2 database, and tested basic backend routing.
5 th Feb	1	Conducted deeper research on SQL Injection exploitation in Java applications. Reviewed insecure query patterns and prepared notes for implementing the first vulnerability.
6 th Feb	1	Drafted backend-focused content for Progress Report 1, summarizing architecture planning, backend setup, and SQL Injection research.

Summary Description

During this reporting period, I contributed significantly to the research, proposal development, and backend planning for our project Insecure by Design. My work began with analyzing existing vulnerable applications such as DVWA, WebGoat, and Juice Shop to understand their strengths and limitations. This research helped shape our motivation for creating a modern, framework-aligned platform using Spring Boot MVC.

I co-authored the pre-proposal and full proposal, focusing on the scope, methodology, and technology justification. After receiving instructor feedback on January 27, I reviewed Fortinet's 2024 Cybersecurity Skills Gap Report and integrated industry-supported evidence showing that 58% of IT decision-makers attribute breaches to staff lacking hands-on cybersecurity skills. This strengthened the rationale for our project.

Following proposal submission, I participated in weekly Friday and Saturday meetings to plan the application architecture and vulnerability placement strategy. I set up the backend project structure, configured the H2 database, and prepared templates for implementing SQL Injection. I also conducted deeper research into insecure query patterns to support upcoming development tasks. The project is progressing smoothly and remains aligned with the original proposal.

Repo Check-In of Implementation Completed

Since the start of the term, I have checked in several updates to the GitHub repository, including:

- Creation and organization of the GitHub Project Kanban board
- Backend project skeleton (controllers, services, initial routing)
- H2 database configuration and application startup verification
- Architecture planning notes and vulnerability placement documentation
- Proposal contributions and updated introduction with Fortinet justification
- Progress Report 1 draft (this document)

All commits were made under my own account, following the requirement for individual contributions and consistent weekly check-ins.

AI Use Section

Table of AI Tools and Specific Use

AI Tool	Version	Feature Used	Value Add
Chatgpt	5.1 free	Grammar and writing help	Used only as a writing aid to flesh out original ideas
Claude	4.5 free	Prototyping and coding	Used to generate quick mocks before making final commitment to the design.
Chatgpt	5.1 free	Coding	Quick UI code generation

Appendix

References

Fortinet. (2024). 2024 Cybersecurity Skills Gap Report. Retrieved January 2026, from
<https://www.fortinet.com/content/dam/fortinet/assets/reports/2024-cybersecurity-skills-gap-report.pdf>

Mudge, R. (digininja). (n.d.). Damn Vulnerable Web Application (DVWA) [GitHub repository]. Retrieved January 2026, from
<https://github.com/digininja/DVWA>

OWASP Foundation. (n.d.). OWASP Juice Shop. Retrieved January 2026, from
<https://owasp.org/www-project-juice-shop/>

OWASP Foundation. (n.d.). OWASP WebGoat. Retrieved January 2026, from

<https://owasp.org/www-project-webgoat/>