

FIREWALL RULE MANAGEMENT SIMULATOR

SYSTEM DOCUMENTATION

SUBMITTED BY,

FESTUS ONYIEGO MOGOI

DCF-03-0077/2024

A SYSTEM DOCUMENTATION SUBMITTED IN PARTIAL FULFILMENT FOR THE AWARD OF DIPLOMA IN CYBERSECURITY AND FORENSICS BY ZETECH UNIVERSITY

APRIL 2025

DECLARATION

I hereby acknowledge that this system project on Firewall Rule Management simulator is my primary work researched and no plagiarism in the document only for the external sources of information used in this proposal document have been properly cited and listed in the references, I believe that no any other author that have submitted the same proposal to the Zetech University

Student Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Sign: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date: \_\_\_\_\_\_\_\_\_\_

Supervisor Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Sign: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date: \_\_\_\_\_\_\_\_\_\_

DEDICATION

This project is dedicated to all the cybersecurity experts and other researchers that are working in improving network security, their focus on innovation and network defense against the emerging threats has really influenced me to develop this project. I also send my gratitude to my mentors who assisted in guidance and insights into this development journey, the contribution you provided in the project shaping has an impact on the secured systems.

ABSTRACT

The firewall rule management simulator is projected to enhance network security and give interactive interfaces for firewall configurations, it has been developed in an advanced networking procedures in rule conflict detection techniques and simulations to curb in redundancies, shadowing, firewall rules contradictions and generalizations. This project it has taken 8 months in development and design, using the following technologies and tools like python and java, firewall network simulation tools like, Cisco packet tracer and net filter. In database management I have used MySQL in storing the firewall rule sets and conflict analysis, scapy frameworks for analysis and manipulation in python. For cybersecurity and OSINT, wireshack and Metasploit has been used in development with visualization and reporting tools like Grafana. This project has solved the following problems lack of interactive learning tools, rule optimization challenges, firewall rule conflicts and performance bottlenecks in rule processing thus ensuring faster evaluations in processing rule efficiency as the best achievement.

DEFINITION OF KEY TERMS

Cisco Packet tracer: practical tool for simulations in network security policies.

Curb: to mitigate security issues in firewall simulations

Database: a central repository for data

Grafana: a visualization tool used as graph based for monitoring and reporting on the firewall performance.

Metasploit: software used to test rule effectiveness against any simulated cyber attacks

Redundancy: to ensure network availability to enhance efficiency

Scapy: tool used in packet analysis and manipulation in python

Shadowing: this is when previous rules set matches traffic conditions

Wireshark: packet analysis for testing firewall behavior

ABBREVIATIONS AND ACRONYMS

SQL – Structured Query Language

OSINT: open-source intelligence tool

LIST OF FIGURES

*Leave this section blank until you’ve done the rest of the document*

*Here, list down all the image captions you have in your document, and in the correct order in which they appear in the document, from start to end e.g.*

Fig 3.1-1 Login Page

Fig 3.1-2 Sign Up Form

Fig 3.2 Cart Management Code

LIST OF TABLES

*Leave this section blank until you’ve done the rest of the document*

*Here, list down all the table captions you have in your document, and in the correct order in which they appear in the document, from start to end e.g.*

Table 1.2 Functional Requirements

Table 1.3 Budget Breakdown

TABLE OF CONTENTS

*Leave this section blank until you’ve done the rest of the document*

Generate a table of contents for your document. Here is a quick guide on how to do that: <https://www.youtube.com/watch?v=0cN-JX6HP7c&t=87s>

CHAPTER ONE: PROJECT PLANNING AND ANALYSIS

(WORKPLAN)

* 1. Statement of Problem

Most of the organizations deeply depend on the firewall for security enhancement in policies strategy but redundancies and misconfigurations of policies attracts security breaches and bring performance inefficiencies. Using the manual way of in rule management in policy setup has many errors that make systems vulnerable to cyber-attacks and other infringements in the network hence causing the conflicts in the rule set and lack simulations. These problems with the manual way of policy in firewall rule management results to increased compliant risks causes delay in incidence responses and lastly cause overheads in IT personnel administration

* 1. Study Justification

This project focuses to minimize the gaps existed in the firewall rule management by initiating a simulator which will help the IT experts and other cybersecurity personnel to simulate traffic flow in the internet to validate in its effeteness automatically detect rule conflicts in redundancy and shadowing to effectively reduce human errors with the zero trust mechanism.

* 1. System Objectives
     1. General Objective

1. To develop a firewall rule management simulator which will integrate AI for report, threat detection training Realtime monitoring and adaptive learning.
   * 1. Specific Objectives
2. To automate rule conflict detection-in reducing the shadowing, redundancy, generalization and lastly, reduce rule contradiction
3. To increase environmental simulation
4. To maximize rule optimization techniques
5. To increase performance analysis
6. To design interactive GUI for rule creation and editing.
7. To generate compliance reports.

* 1. Functional Requirements

|  |  |  |
| --- | --- | --- |
| User | User Activities | Features |
| Network admin | * Create and edit read, update and delete firewall rules * Run simulations * View AI reports * Configure system settings | * Drag and drop with the rule builder syntax validation * Packet flow visualizer with real-time logging and AI simulation aid. * Automated conflict detection in identifying redundant and shadowed rules |
| Security analyst | * View, edit rules (limited to their scope) * Run simulations * Access AI vulnerability reports * Cannot change system settings * Rule set optimization | * Send compliance reports to {NIST, CIS} with scoring criteria * AI-driven * Recommendations for the rule minimization |
| Training user | * + Read-only access to rules (embeded sensitive data)   + Run simulations in a sandbox environment   + View AI-generated learning materials | To use tutorials and demos |

Table 1.4 Functional Requirements Table

* 1. Breakdown of Tools & Resources to Be Used

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Category | Tool/resource | Purpose | Accessibility/notes | budget | alternatives |
| Frontend | React.js+Typescript | Interactive GUI for rule management | WCAG-compliant components | free |  |
|  | D3.js | network topology and packet visualization | Supports high contract modes | Subscriptions =4000 | free |
|  | Electron | Desktop application deployment | Cross-platform (windows/Linux/macOS | free |  |
| Backend | Python (Django/Flask | rule processing logic and API endpoints | RESTful architecture | free |  |
|  | **PostgreSQL** | Storage for firewall rules, logs and audit trails | ACID compliance | Subsriptions=3000 |  |
| Simulation engine | Scapy | Synthetic traffic generation and simulation | Customizable traffic profiles | free |  |
|  | Custom algorithm(python) | Rule conflict detection like shadowing and redundancy | Tree-based precedence checks | 3500 | free |
| Security and osint tools | OWASP ZAP Metasploit OpenVAS | Vulnerability scanning for web/API features | Automated penetration testing | free | free |
|  | Hashcorp vault | Secure storage of API key/credential | Role-based access control RBAC |  |  |
| Testing and dubbuging | Pytest,wireshack,burp suit | Unit/integration testing for backend logic and debugging rule conflicts and system errors | CI/CD pipeline integration | Free |  |
|  | Selenium | Frontend ui/ux testing | Cross browser support | free | 6000 |
| DevOps | Docker | Containerization for consistent deployment | Kubernetes-compatible |  |  |
|  | AWS EC2 | Cloud-hosted demo environment | Auto-scaling capabilities |  |  |
| Accessibility | Axe accessibility toolkit | WCAG 2.1 compliance validation | Integrated into CI/CD |  |  |
|  | Response design frameworks | Adapt UI for mobiles/desktop/4k displays | Bootstrap 5 or material-UI |  |  |
| Network simulation | GNS3, cisco packet tracer | Testing and analyzing firewall rule behavior |  |  |  |



1.6 Project Schedule Breakdown



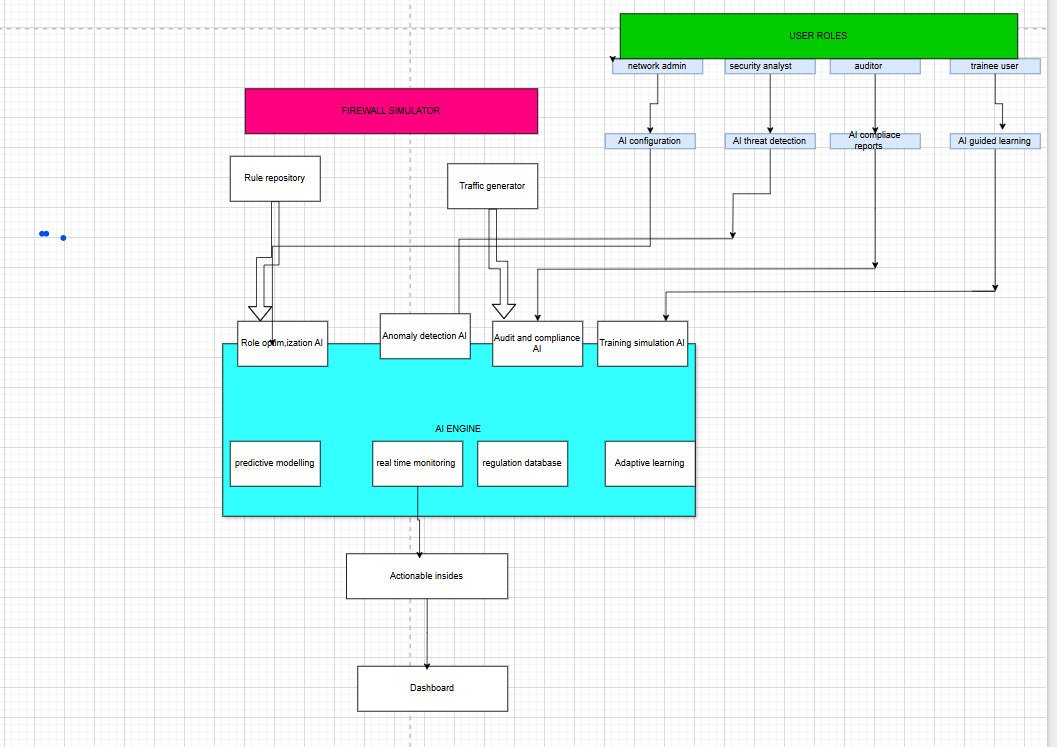
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | PROJECT MILESTONES | | | | | |
| WEEKS | Project Planning & Analysis  (System Documentation: Coverpage & Chapter One) | Project Design & Modeling  (System Documentation Chapter Two) | Project Development & Testing  (System Documentation Chapter Three) | Project Deployment  (System Documentation Chapter Three) | Final Touches of System Documentation  (Preliminary Pages, Chapter Four & References) | Project Presentation |
| 5-9 May |  |  |  |  |  |  |
| 12-16 May |  |  |  |  |  |  |
| 19-23 May |  |  |  |  |  |  |
| 26-30 May |  |  |  |  |  |  |
| 2-6 June |  |  |  |  |  |  |
| 9-13 June |  |  |  |  |  |  |
| 16-20 June |  |  |  |  |  |  |
| 23-27 June |  |  |  |  |  |  |
| 30-4 July |  |  |  |  |  |  |
| 7-11 July |  |  |  |  |  |  |
| 14-18 July |  |  |  |  |  |  |
| 21-25 July |  |  |  |  |  |  |
| 11 Aug |  |  |  |  |  |  |

*Table 1.6 Project Schedule Breakdown*

*Blue initially means that milestone has not achieved. Once you achieve that milestone, fill it with any other color of your choice to mean you’re done with that part. That helps you to monitor your own progress versus the time scheduled.*

*When printing, this page must be printed in LANDSCAPE as the rest is printed in portrait*

CHAPTER TWO: DESIGN AND MODELING





**1. Rule Creation/Editing**

* + AI real-time analysis of rule syntax and security implications
  + propose improvements (e.g., "focused on restricting source IP range")

**2. Simulation Engine:**

* + Anomaly detection during traffic simulation
  + Predict attack patterns (e.g., port scanning) and flag rules that allow them

**3. Reporting:**

* + Generate natural language summaries of firewall health
  + Compliance checks (e.g., PCI-DSS, HIPAA)

**4. Training Module (for Trainees)**

* + AI-generated questions based on rule sets
  + Interactive vulnerability fixing challenges

2.1 Introduction to Modelling

In this chapter is focused on the designs and models that have drawn to visualize the system before the development and this a benefit for the easy understanding of the development process of this project.

2.2 User Interface Models

1. Overall Process Flowchart (High-Level):

- This will show the main processes and how they connect.

2. UML Sequence Diagrams for:

a. Rule Creation and AI Analysis

b. Threat Detection and Response

c. Compliance Audit



2.3 Logic Models

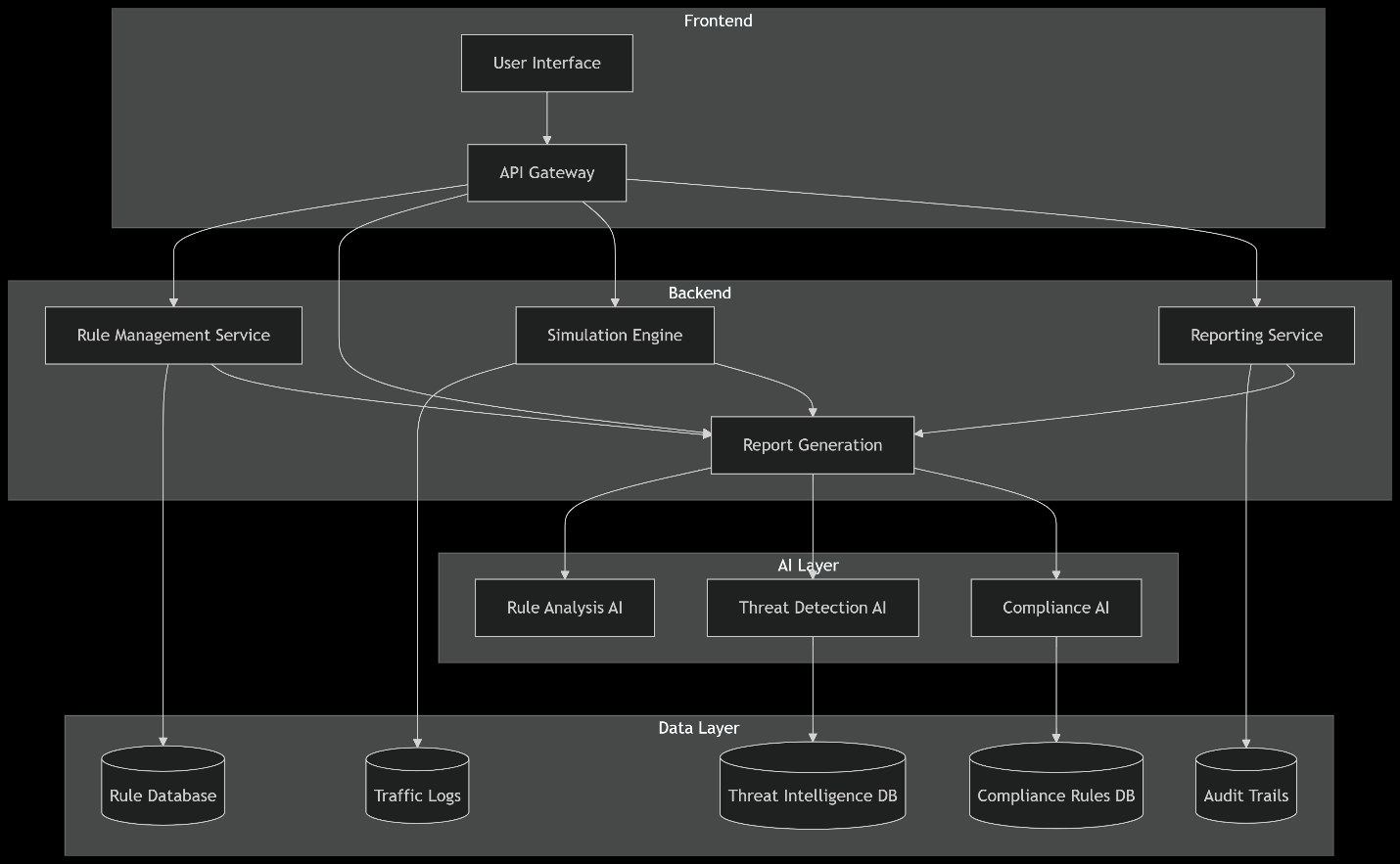
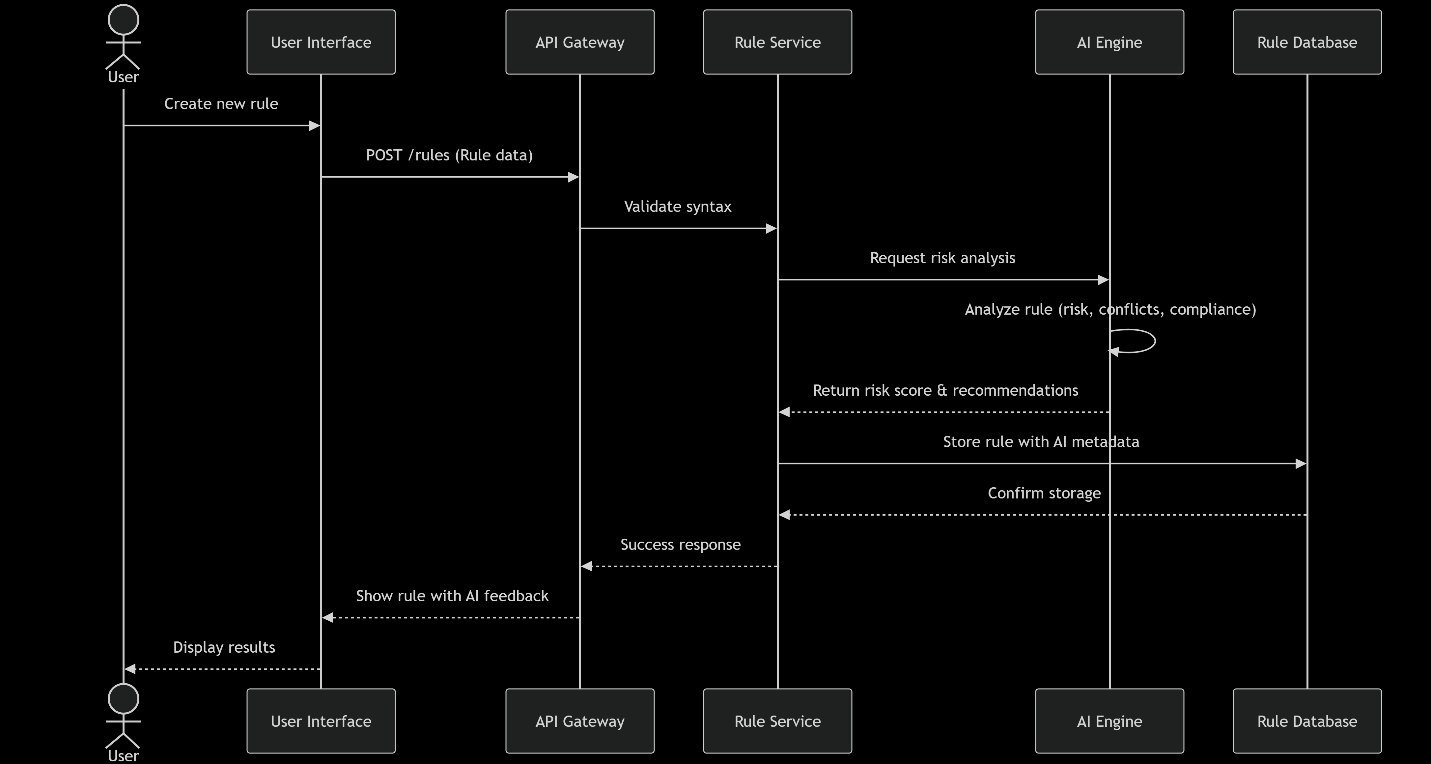


Figure 1overall system architecture

Figure 2 uml sequence diagram(**Rule Creation with AI Analysis**)

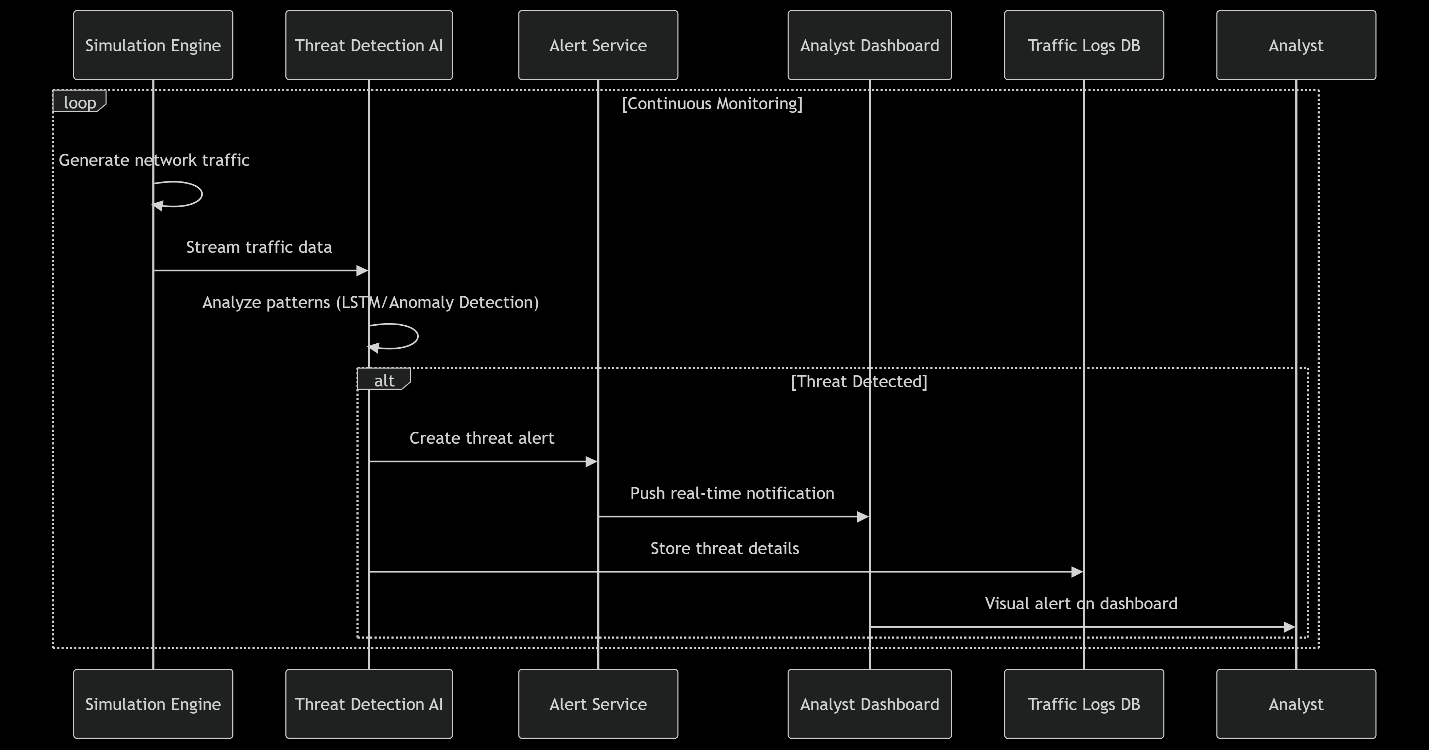


Figure 3 real-time threat detection

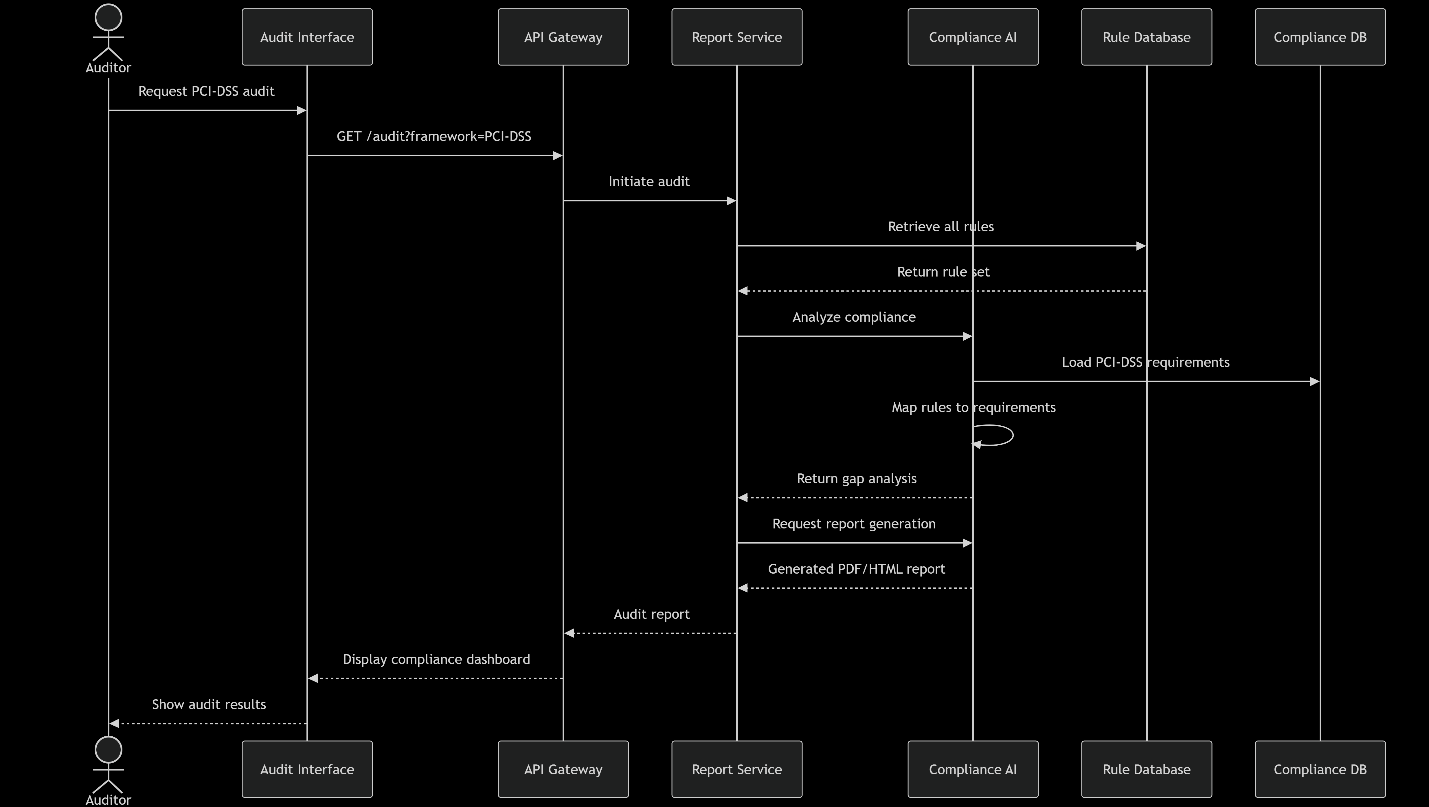


Figure 4 compliance audit process

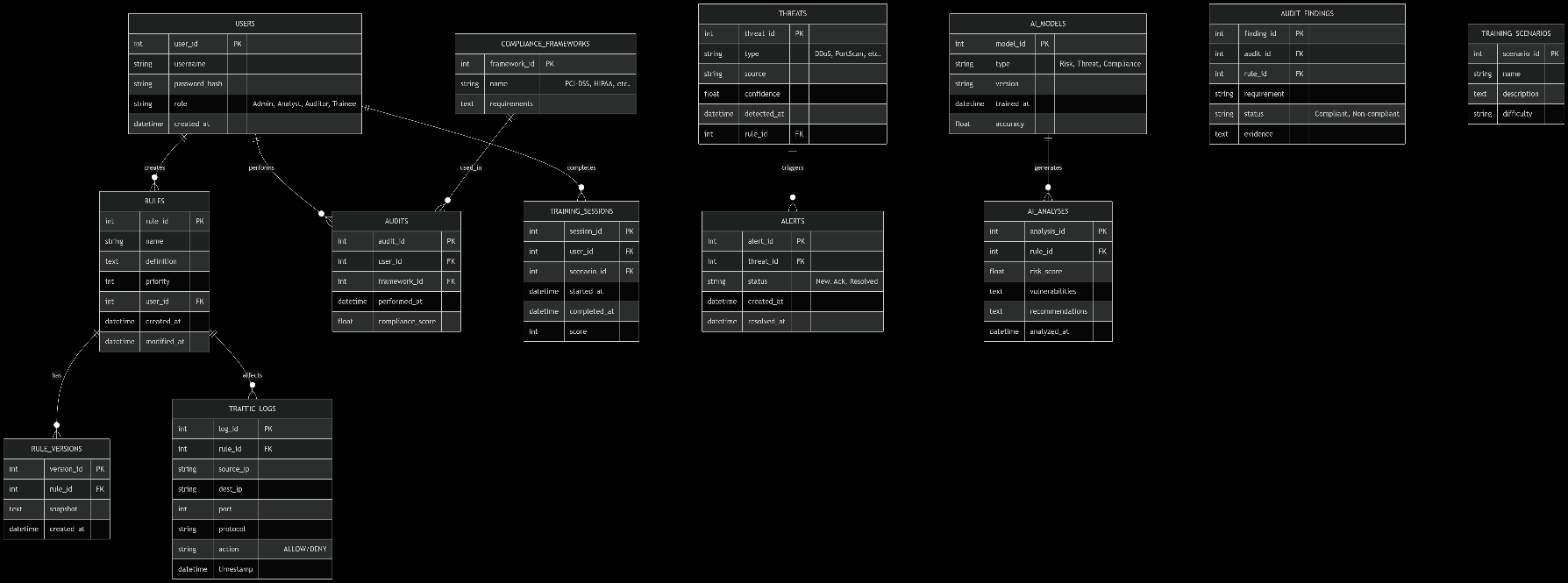


Figure 5 ERD FOR DATABASE DIAGRAM

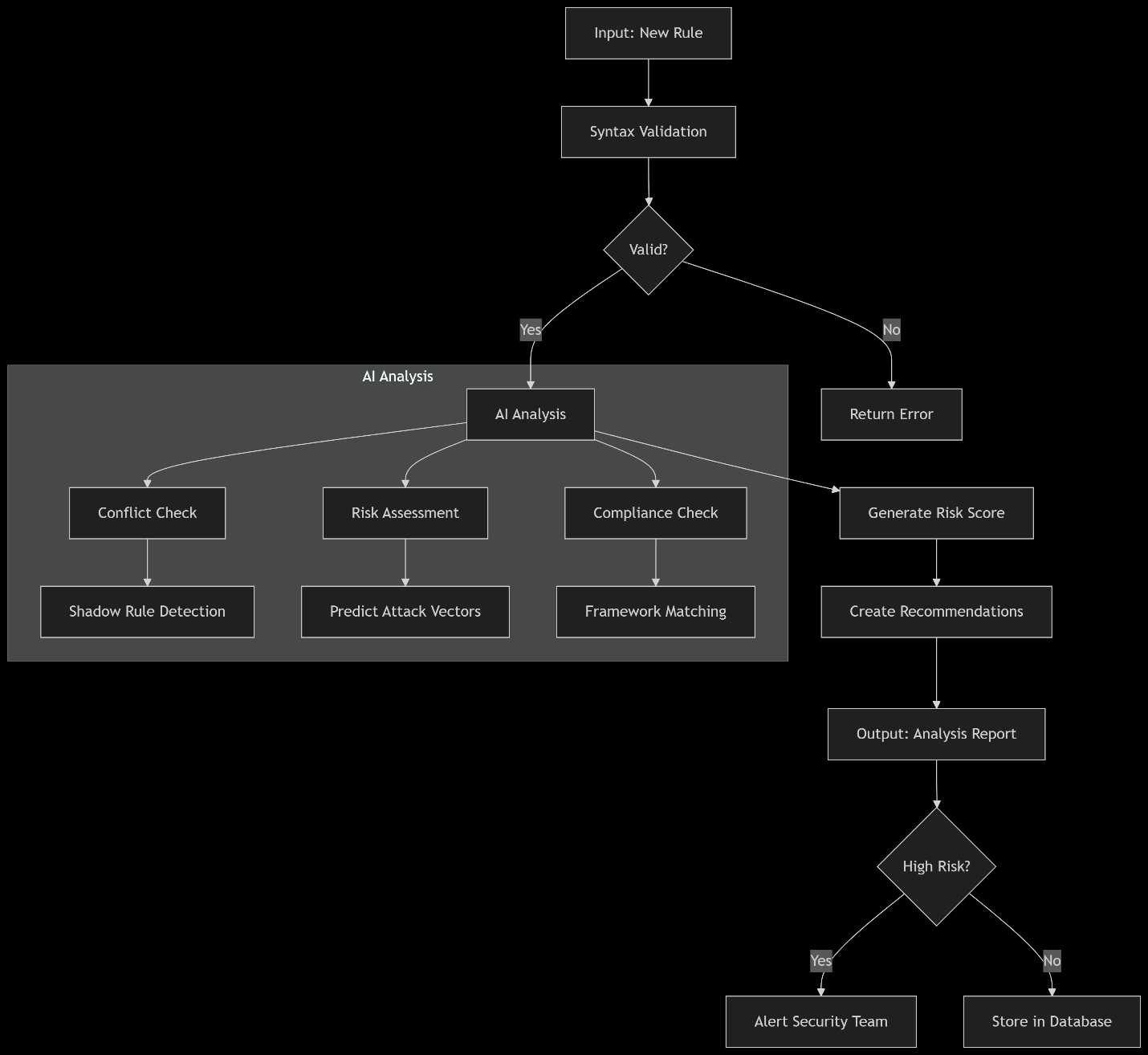


Figure 6 AI DECISION LOGIC FLOW

*Diagrams of all the process architecture diagrams that you sketched to visualize how the system will work e.g. flowcharts, class diagrams, sequence diagrams, data flow diagrams, decision trees, etc..*

*Organize them into subheadings e.g. 2.3.1 Login Process Flowchart, 2.3.2 Purchase/Checkout Process Flowchart, 2.3.3 Class Diagram for the System, 2.3.4 From Enquiry Making to Enquiry Reply Flowchart*

*In each subheading, have the hand-sketched (or if you’ve used a drawing tool) design of that diagram, and a caption underneath the diagram*

CHAPTER THREE: SYSTEM IMPLEMENTATION

(DEVELOPMENT, TESTING AND DEPLOYMENT)

*This is where you report on your development journey e.g. from when you downloaded your IDEs, and coding, to implementing your database to deploying your system*

3.1 Introduction

In this part you will be able to understand the development milestone from ;logging in, signing up , trainee user and the admin dashboards

3.2 User Interface Development

**Trainee visual login page**

START -> Login Page

|

v

[Trainee Dashboard]

|

|---> View Firewall Rules -> (Displays Read-Only Rules)

|

|---> Simulate Traffic -> (Run Network Traffic Tests)

|

|---> Access Logs -> (View Read-Only Security Logs)

|

|---> Tutorials & Learning -> (Interactive Lessons & Videos)

|

|---> Logout -> (Returns to Login Page)

|

END

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Trainee Dashboard</title>

<link rel="stylesheet" href="styles.css">

</head>

<body>

<div class="dashboard-container">

<h2>Firewall Rule Management - Trainee Dashboard</h2>

<div class="dashboard-options">

<button onclick="viewRules()">View Firewall Rules</button>

<button onclick="simulateTraffic()">Run Traffic Simulation</button>

<button onclick="viewLogs()">Access Network Logs</button>

<button onclick="openTutorials()">Tutorials & Learning</button>

</div>

</div>

<script src="trainee-functions.js"></script>

</body>

</html>

Figure 9 trainee dashboard feature and layout

*START -> Login Page (Authenticate User)*

*|*

*v*

*[Admin Dashboard] (Access Restricted to Admin Role)*

*|*

*|---> Manage Firewall Rules -> (Add, Edit, Delete Rules)*

*|*

*|---> Monitor Network Traffic -> (Live Traffic Analysis)*

*|*

*|---> Manage User Roles -> (Assign Roles & Permissions)*

*|*

*|---> Analyze Security Logs -> (Event & Incident Review)*

*|*

*|---> Configure System Settings -> (Adjust Policies & Preferences)*

*|*

*|---> Logout -> (Returns to Login Page)*

*|*

*END*

*<!DOCTYPE html>*

*<html lang="en">*

*<head>*

*<meta charset="UTF-8">*

*<meta name="viewport" content="width=device-width, initial-scale=1.0">*

*<title>Admin Dashboard</title>*

*<link rel="stylesheet" href="styles.css">*

*</head>*

*<body>*

*<div class="dashboard-container">*

*<h2>Firewall Rule Management - Admin Dashboard</h2>*

*<div class="dashboard-options">*

*<button id="manageRules">Manage Firewall Rules</button>*

*<button id="monitorTraffic">Monitor Network Traffic</button>*

*<button id="manageUsers">Manage User Roles</button>*

*<button id="analyzeLogs">Analyze Security Logs</button>*

*<button id="configureSettings">Configure System</button>*

*<button onclick="logout()">Logout</button>*

*</div>*

*</div>*

*<script src="admin-functions.js"></script>*

*</body>*

*</html>*

*body {*

*font-family: Arial, sans-serif;*

*background-color: #f4f4f4;*

*text-align: center;*

*}*

*.dashboard-container {*

*margin: 50px auto;*

*background: white;*

*padding: 20px;*

*border-radius: 8px;*

*box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);*

*}*

*.dashboard-options button {*

*display: block;*

*margin: 10px auto;*

*padding: 12px;*

*font-size: 16px;*

*background-color: #d9534f;*

*color: white;*

*border: none;*

*cursor: pointer;*

*}*

*.dashboard-options button: hover {*

*background-color: #c9302c;*

*}*

Figure 10 admin dashboard ui (html and css)

*START -> Login Page (User Authentication)*

*|*

*v*

*[Check User Role]*

*|*

*|---> Trainee Role? -> Redirect to [Trainee Dashboard]*

*| |*

*| |---> View Firewall Rules (Read-Only Access)*

*| |---> Simulate Traffic (Sandbox Mode)*

*| |---> View Network Logs (Read-Only)*

*| |---> Tutorials & Learning Resources*

*|*

*|---> Admin Role? -> Redirect to [Admin Dashboard]*

*|*

*|---> Manage Firewall Rules (Add/Edit/Delete)*

*|---> Monitor Network Traffic (Live Data)*

*|---> Manage User Roles & Permissions*

*|---> Analyze Security Logs*

*|---> Configure System Settings*

*|*

*|---> Invalid Role? -> Access Denied -> Redirect to Login*

*|*

*END*

*document.addEventListener("DOMContentLoaded", function () {*

*const userRole = localStorage.getItem("role");*

*if (!userRole || userRole !== "admin") {*

*alert("Access Denied. Redirecting...");*

*window.location.href = "login.html";*

*}*

*});*

*function logout() {*

*localStorage.removeItem("role");*

*alert("Logged out successfully.");*

*window.location.href = "login.html";*

*}*

Figure 11 role-based access control flowchart and code

*NB { To iintergrate AI that can learn to detect risky firewall rules that can help admins to optimize security settings*

*Secure backend backend with RBAC authentication*

*Visualize AI generated reports in dashboard UI}*

3.3 Logic Development

*Your system has two sides – the user side (where different users can interact with the system) and the logic side. The logic side is how the system is performing activities using data e.g. login validation, processing checkout and generating a receipt, storing data into database, retrieving products from DB and showing them on a certain page. This is called logic.*

*Under this section you will screenshot all the Codes or Simulations that handle different processes/activities. Organize them into subheadings e.g. 3.3.1 Login Validation Code (herein, explain in brief how the system handles login, take screenshots of the login validation code, and caption it)*

3.4 Testing

*Detail out what actions you took to test whether different features of your system are working. And corrections you made to different parts to ensure that they are working as planned.*

3.5 Deployment

On-premise, website and playstore platforms are ideal for deployment of this project.

*Mention on what platform have you deployed your system (e.g. if it’s a website, what web hosting platform did you use – and copy paste a link to your website, if it’s an android application – the process of putting up your app on Google Playstore.). Essentially, detail out the process of how you transformed your project into a form that can be run/installed by anybody.*

CHAPTER FOUR: CONCLUSION AND RECOMMENDATION

4.1 Conclusion

*A paragraph summarizing chapter one, two and three. And finish by telling the reader what new things you learnt during the entire time developing your project, and challenges faced.*

4.2 Recommendation

*Here, speak as if addressing another developer of the same project you’ve been working on. Suggest to him/her things that they can do to improve the system if they so choose to pick it up and enhance it. E.g. if you created an e-receipt for customers, suggest to him/her to consider integrating a physical receipt printer since some customers would prefer that and also you couldn’t implement that because you didn’t have the budget for a printer. If your system is a staff clock-in-and-out system and in your system a staff has to type in their staff number into the system, suggest to the developer what better technologies he/she can use to enhance that specific part e.g. biometric or swipe-card, maybe since you did not have the budget to buy those devices and/or didn’t have the technical skills for that. Or if for payment your system is using Pay-After-Delivery, suggest to him/her to consider integrating with MPESA or Bank Card for pre-delivery payments.*

*NB: Recommendation is not an excuse for a NOT WORKING system, e.g. it is not saying that your system can’t store any data because you didn’t have the time to implement a database, No. It is a suggestion for how your ALREADY WORKING SYSTEM (with the lower technologies you’ve used) can be MADE BETTER, with HIGHER/BETTER TECHNOLOGIES. E.g. you used MySQL to store structured data, suggest for them to use a NO-SQL approach to store non-structured data like customer feedback and product reviews.*

REFERENCES

*A list of all the websites, articles, books, YouTube channels, AI Conversations (copy link address) that helped you during your project.*

*Format them in APA7 format, in alphabetical order.*

*A guide on referencing from web articles/tutorials:* [*https://libraryguides.vu.edu.au/apa-referencing/7Webpages*](https://libraryguides.vu.edu.au/apa-referencing/7Webpages) *(see in-black-box)*

*A guide on referencing from social media articles:* [*https://libraryguides.vu.edu.au/apa-referencing/7SocialMedia*](https://libraryguides.vu.edu.au/apa-referencing/7SocialMedia) *(see in-black-box)*

*A guide on referencing from conversations you had with AI e.g. ChatGPT:* [*https://libraryguides.vu.edu.au/apa-referencing/generativeAI*](https://libraryguides.vu.edu.au/apa-referencing/generativeAI) *(scroll to end of page)*

*Done with references, now go back to PRELIMINARY PAGES (pages before chapter one), and fill those pages*