BSCS-S24-016

03-134202-008 Ali Hasan

03-134211-027 Muhammad Shahbaz Khan

SecureNet Analyzer

In partial fulfilment of the requirements for the degree of

Bachelor of Science in Computer Science

Supervisor: Tahir Iqbal

Department of Computer Sciences

Bahria University, Lahore Campus

January 2025

C e r t i f i c a t e

We accept the work contained in the report titled

"SecureNet Analyzer"

written by

Ali Hasan

Muhammad Shahbaz Khan

as a confirmation to the required standard for the partial fulfilment of the degree of Bachelor of Science in Computer Science.

Approved by:

Supervisor: Tahir Iqbal

_____ (Signature)

December 05, 2024

DECLARATION

We hereby declare that this project report is based on our original work except for citations and quotations which have been duly acknowledged. We also declare that it has not been previously and concurrently submitted for any other degree or award at Bahria University or other institutions.

Enrolment	Name	Signature
03-134202-008	Ali Hasan	
03-134211-027	Muhammad Shahbaz Khan	

Date : December 03, 2024

Specially dedicated to
my beloved grandmother, mother and father
(Ali Hasan)
my beloved grandmother, mother and father
(Muhammad Shahbaz Khan)

ACKNOWLEDGEMENTS

We would like to thank everyone who had contributed to the successful completion of this project. We would like to express our gratitude to my supervisor, Mr TAHIR IQBAL for his invaluable advice, guidance and his enormous patience throughout the development of the research.

In addition, we would also like to express my gratitude to our loving parent and friends who had helped and given me encouragement.

Ali Hasan
Muhammad Shahbaz Khan

SecureNet Analyzer

ABSTRACT

SecureNet Analyzer stands as a beacon of defense in the face of escalating cyber threats, offering a comprehensive suite of tools and services to fortify organizations against potential vulnerabilities. With its advanced web-based platform, SecureNet Analyzer swiftly detects and identifies website vulnerabilities, providing detailed reports in PDF or Word format, complete with solutions tailored to each issue. Its intuitive interface caters to organizations of all sizes, adeptly scanning IPs, URLs, and documents to ensure thorough coverage of potential security risks.

It delivers expert penetration testing services, conducted by seasoned security professionals. Through rigorous testing methodologies, our team simulates real-world cyber-attacks to uncover potential exploits and weaknesses within your infrastructure. These penetration tests not only identify vulnerabilities but also offer actionable recommendations and solutions, empowering organizations to prioritize remediation efforts effectively. By partnering with SecureNet Analyzer, organizations can bolster their security defences, mitigate risks, and maintain regulatory compliance in an ever-evolving cybersecurity landscape.

TABLE OF CONTENTS

DECLARATION **ii**
ACKNOWLEDGEMENTS **iv**
ABSTRACT **v**
TABLE OF CONTENTS **vi**
LIST OF TABLES **ix**
LIST OF FIGURES **x**
LIST OF SYMBOLS / ABBREVIATIONS **xi**
LIST OF APPENDICES **xii**

CHAPTERS

1 INTRODUCTION **1**
1.1 Background 1
1.2 Problem Statements 1
1.3 Aims and Objectives 1
1.4 Scope of Project 2

2 SOFTWARE REQUIREMENTS SPECIFICATIONS **3**
2.1 User Classes and Characteristic 3
2.1.1 User 3
2.2 Functional Requirements 4
2.2.1 User Authentication 4
2.2.2 Scanning Modules 4
2.2.3 Comprehensive Reporting 4
2.2.4 History of Services 4
2.3 Non-Functional Requirements 4
2.4 Final Deliverable of the Project and Beneficiaries 5

2.4.1 Final Deliverable project 5
2.4.2 Beneficiaries 5
2.5 Design and Implementation Constraints 6
2.5.1 External Interface Requirements 6
2.6 API Interfaces 7
2.7 Related Work 7
2.8 Use Case Diagrams 8
2.8.1 Class Diagram 8
2.8.2 Flowchart 9
2.8.3 Entities Relationship Diagram 11
2.9 Use Case Scenarios 12
2.9.1 Signup as User 12
2.9.2 Login as User 12
2.9.3 Initiate Website Scanning 13
2.9.4 Initiate IP Scanning 14
2.9.5 Initiate Document Scanning 14
2.9.6 View Scan History 15

DESIGN AND METHODOLOGY 16
3.1 Methodology 16
3.2 Feasibility Plans 16
3.2.1 Resource Requirements 16
3.3 Risks Involved 17

4 DATA AND EXPERIMENTS (and/or IMPLMENTATION) 18
4.1 Technologies used 18
4.1.1 Frontend Development 18
4.1.2 Backend Development 18
4.1.3 Database 18
4.2 Experiments Conducted 18
4.2.1 Performance Testing 18
4.2.2 Accuracy Testing 19
4.2.3 Scalability Testing 19

4.2.4 Usability Testing 19

5 USER MANUAL 20

5.1 Welcome Screen of SecureNet Analyzer (Dashboard) 20

5.2 Sign-up screen 20

5.3 Sign in Screen 21

5.4 Scanning Services 21

5.5 Ip Scanning 22

5.6 URL Scanning 22

5.7 Document Scanning 23

5.8 Website scanning 23

6 CONCLUSION AND RECOMMENDATIONS 24

6.1 Conclusion 24

6.2 Recommendations 24

REFERENCES 25

APPENDICES 26

LIST OF TABLES

TABLE	TITLE	PAGE
Table 2.9.1	Sign Up as User	12
Table 2.9.2	Login as a User	12
Table 2.9.3	Initiate Website Scanning	13
Table 2.9.4	Initiate IP Scanning	14
Table 2.9.5	Initiate Document Scanning	14
Table 2.9.6	View Scan History	15

LIST OF FIGURES

FIGURE	TITLE	PAGE
Figure 2.8.1: Class Diagram		9
Figure 2.8.2 Flowchart		10
Figure 2.8.3 ER Diagram		11
Figure 5.2.1 User Sign-up screen		20
Figure 5.3.1 User Sign-in screen		21
Figure 5.4.1 Scanning Services		21
Figure 5.5.1 IP Scanning Screen		22
Figure 5.6.1 URL Scanning Screen		22
Figure 5.7.1 Document Scanning Screen		23
Figure 5.8.1 Website scanning screen		23

LIST OF SYMBOLS / ABBREVIATIONS

Symbol / Abbreviation	Meaning
API	Application Programming Interface
URL	Uniform Resource Locator
IP	Internet Protocol
JWT	JSON Web Token
DB	Database
PDF	Portable Document Format
DOC	Document (typically Word document format)
HTTPS	Hypertext Transfer Protocol Secure
SQL	Structured Query Language
GUI	Graphical User Interface
DNS	Domain Name System
IPV4	Internet Protocol version 4
IPV6	Internet Protocol version 6
OWASP	Open Web Application Security Project

LIST OF APPENDICES

APPENDIX	TITLE	PAGE
	APPENDIX A: Technology Stack	54
	APPENDIX B: Use Cases	54
	APPENDIX C; Sequence Diagrams	54
	APPENDIX C; User Authentication	54

CHAPTER 1

INTRODUCTION

1.1 Background

Here we present SecureNet Analyzer, innovative and robust cybersecurity solution aimed at scanning and detection of all types of vulnerabilities. It gives the users a quick picture of the vulnerability of websites, IP address, URLs, and documents, and provides the detailed report in PDF or Word format. Some of these reports comprise of the evaluation of the risks that were identified and best mechanisms of dealing with them. It transcends the basic of merely scanning by also providing penetration test services performed by severance security analysts. With sophisticated methods of analysis, our specialists perform exercises that resemble an actual cyber campaign to determine your network's vulnerabilities. This is more than a defensive approach of pinpointing the security threats, and also offers ways on how to enhance the protection of structures. Through its simple interface and awesome features, it makes sure that companies and firms no matter how small or large they are secure their electronic property, minimize risks and possess credible security measures despite the emergent threats from the ingenious criminals in the cyberspace.

1.2 Problem Statements

Ever evolving threat actors have made it difficult for organizations to effectively prevent new and emerging threats to their digital business environment. Current approaches are not complete in the protection afforded, and there are openings in security logistics.

1.3 Aims and Objectives

The main purpose of SecureNet Analyzer is to offer powerful protection for organizations against a rising tide of cyber threats. Its purpose is to identify areas of weakness in web structures and provide recommendations for solutions from reports with various analyses.

Specific aims include:

- Real-time IP, URL, and document scanning.
- Website Scanning to simulate real-world attack scenarios.
- User-friendly interface for accessibility.

1.4 Scope of Project

It is designed to enrich organizations of various sizes, especially given the possibility to adjust the services provided to the given needs. It is also possible to adjust all the characteristics of the platform to provide diverse protection for enterprises and SMBs, focusing on each digital asset. As it handles vulnerabilities systematically for websites, networks and documents it also ensures enhanced security against new threats as a result of the all round cover. This functionality of the tool provides organizations with real-time scans together with information that gives them a competitive edge on potential risks. In addition, with the continual update of SecureNet Analyzer, it out-compares today's emerging cyber threats, helping businesses achieve secure operation in a dynamic business world. Thanks to its rich list of features, it offers constant business support to maximize protection and abate new cyber threats effectively.

CHAPTER 2

SOFTWARE REQUIREMENTS SPECIFICATIONS

2.1 User Classes and Characteristic

2.1.1 User

It is available to users to login securely into the platform. After this, some available services may include vulnerability scans involving IPs, URLs and documents, web site scans. The services that are consumed are stored in a record format on the website whereby the user may access data regarding previous activities as well as reports generated. Users are able to view full reports on specific types of vulnerabilities, monitor past scans, and analyze past actions in order to maintain an understanding of security threats as they alter over time. Through the provision of history, it makes it possible for the users to be in a position to navigate through records and secure their security needs as well.

- Register
- Login
- View Dashboard
- Services
    - Ip Scanning
    - URL Scanning
    - Document Scanning
    - Website Scanning

2.2 Functional Requirements

2.2.1 User Authentication

Users should be able to register and log in securely using username and password.

2.2.2 Scanning Modules

The system offers dedicated scanning modules to analyze different aspects of an organization's digital infrastructure

- **IP Scanning**: Identifies vulnerabilities within networked devices and servers.
- **URL Scanning**: Analyzes website vulnerabilities, detecting common exploits like SQL injection, cross-site scripting (XSS), and other web-based security issues.
- **Document Scanning**: Scans documents, such as PDFs files, for malware, embedded exploits, and sensitive data leaks.

2.2.3 Comprehensive Reporting

After scanning, the program creates vulnerability reports in PDF that consist of a list of vulnerabilities found together with their severity level and recommendations. These reports are flexible to allow for the requirements of the different groups that may include the technical department and the business management.

2.2.4 History of Services

It can be viewed by users that the platform provides an opportunity to find a history of all used services. These include past scan results, web site scanning reports and any other activity that they may have embarked upon. Use of the history offers those who use the application the opportunity to keep record of their progress in the past.

2.3 Non-Functional Requirements

Performance

- The platform should be responsive and performant, even under heavy load.
- User interactions such as posting, following and querying should have minimal latency.

- The platform is capable of processing large amounts of data in real-time, providing actionable insights without unnecessary delays.

Security

- User authentication and data transmission should be encrypted to ensure security.
- Only authorized users should have access to their respective functionalities.

Scalability

- The platform should be designed to handle a growing user base and increasing data volume
- The platform is designed to scale seamlessly with the growth of an organization.

Usability

- The interface should be easy to use for all users
- Clear instructions and well-organized features make it simple to perform scans, view reports, and act on findings.

Reliability

- The platform should work smoothly with little downtime and handle errors well.
- Data should be kept safe to prevent any loss or damage.
- It should be operational at all times, with minimal downtime or disruption.

2.4 Final Deliverable of the Project and Beneficiaries

2.4.1 Final Deliverable project

- Web Application

2.4.2 Beneficiaries

- Organizations
- Ethical Hackers
- Pen-Testers
- Bug Bounty Hunters
- Freelancers
- Students

2.5 Design and Implementation Constraints

SecureNet Analyzer faces several design and implementation constraints, including:

- **Compatibility with Multiple Platforms**: To ensure that the platform fits well in as many configurations as possible (OS: Windows, Linux, macOS; Browsers: Chrome, Firefox, Edge) requires a lot of considerations in terms of interface and proposed solutions to avoid performance issues.
- **Real-time Scanning**: Real-time scanning of large-scale networks and websites is a complex process that incurs significantly longer times than other forms of analysis; this is due to the need to develop efficient algorithms and infrastructure to ensure timely analysis for eventual use by the end user.
- **Security and Data Privacy**: While strong security features like storage are important and necessary since privacy is a major concern in todays world, implementing them increases the degree of difficulty and must be regularly modified to counter increasingly complex threats.
- **Scalability**: The longer organizations become and more twisted their digital environment is, the application must also expand and be able to perform stably, which requires high backend capabilities and integration with the cloud.
- **User-Friendly Interface**: The main difficulty here is in achieving the optimal level of the product's functional and mystical richness while maintaining a clear, even interface for a normal user who needs to analyze security reports and vulnerabilities data.

2.5.1 External Interface Requirements

User Interface

Web Interface: The platform should offer a GUI embedded into a web application that can operate through multiple web browsers such as Chrome, Firefox, Safari, Edge. It should be user friendly for users to be able to log in, pick services: IP, URL, document scanning, and past records of scans.

Mobile Access: While the main access to the platform is through a web browser, the platform may be available as a mobile application, or a version optimized for phone and tablets for accessing reports and services.

Hardware Requirements

Users will access the platform using their personal devices, including desktop computers, laptops, smartphones and tablets.

Software Requirements

Web Browsers

2.6 API Interfaces

RESTful API: SecureNet Analyzer will have API interfaces in an HTTP based REST model to enable interoperability with other systems such as Virus total, Zap Proxy and vulnerability management software among others. These can also be employed to initiate a scan, for discovering reports, or to store scan data.

Data Import/Export: APIs have to read scan results or security logs and have to write generated vulnerability reports in PDF, MS Word, CSV or other formats.

2.7 Related Work

In previous research, many cybersecurity measures have been developed to offer vulnerability assessment and penetration testing services to protect companies against threats. For instance, Nessus or Qualys have their vulnerability scanning services While SecureNet Analyzer has its vulnerability assessments, it goes further than just notifying the client about a vulnerability; it provides a detailed report on that hole in the system and recommendations on how the problem can be solved. Further, there are penetration testing services delivered by the companies such as Rapid7 and Synopsys, in which the professional security specialists stage the actual hacks. However, SecureNet Analyzer goes further by providing a more refined UI, which allows organizations of all kinds and of different technical competence to address the pending security risks effectively. Prior research and solutions have also revealed that adherence to industry standards is critical, which SecureNet Analyzer provides suggestions to enhance the security posture, and therefore fulfills changing standard norms adherence necessities.

OWASP ZAP (Zed Attack Proxy)

OWASP ZAP is an open source tool that can be used to Pen test web applications during development and Testing. Contained in the suite, you should purchase automated scanners together with a set of manual testing tools that can be used for sophisticated penetration testing.

Web Application Scanning (WAS)

Qualys WAS is a web application security assessment tool which is a cloud-based tool mainly used for scanning organization's web application. It assists organizations in diagnosing and treating security problems; providing a continuous solution to vulnerability and compliance.

Acunetix

Acunetix is a commercial web application security scanner that detects vulnerabilities such as SQL injections.

Nessus:

Nessus performs vulnerability analysis and is intended for detection of vulnerabilities in operating systems and network devices, web applications, etc.

2.8 Use Case Diagrams

2.8.1 Class Diagram

Primarily it was designed scan related features for the files, URLs, IPs or the web applications. The User class manages and authenticates the data concerning the users, connecting to numerous kinds of scans: FileScan, IpScan, WebScan, UrlScan, which contain information about the result's status for each scan. Every scan type has its FileScan Objects, IpScanObjects, WebScanObject, UrlScanObjects objects for a particular user to keep record details of the scan conducted. The JwtBearer class is also the one that handles the user authentication token as a middleware. These relationships reveal that a user is capable of handling several scans, and each of the scans can accommodate several scan objects arranged in a coherent manner for ease of user-tailored scan tracking and reporting.

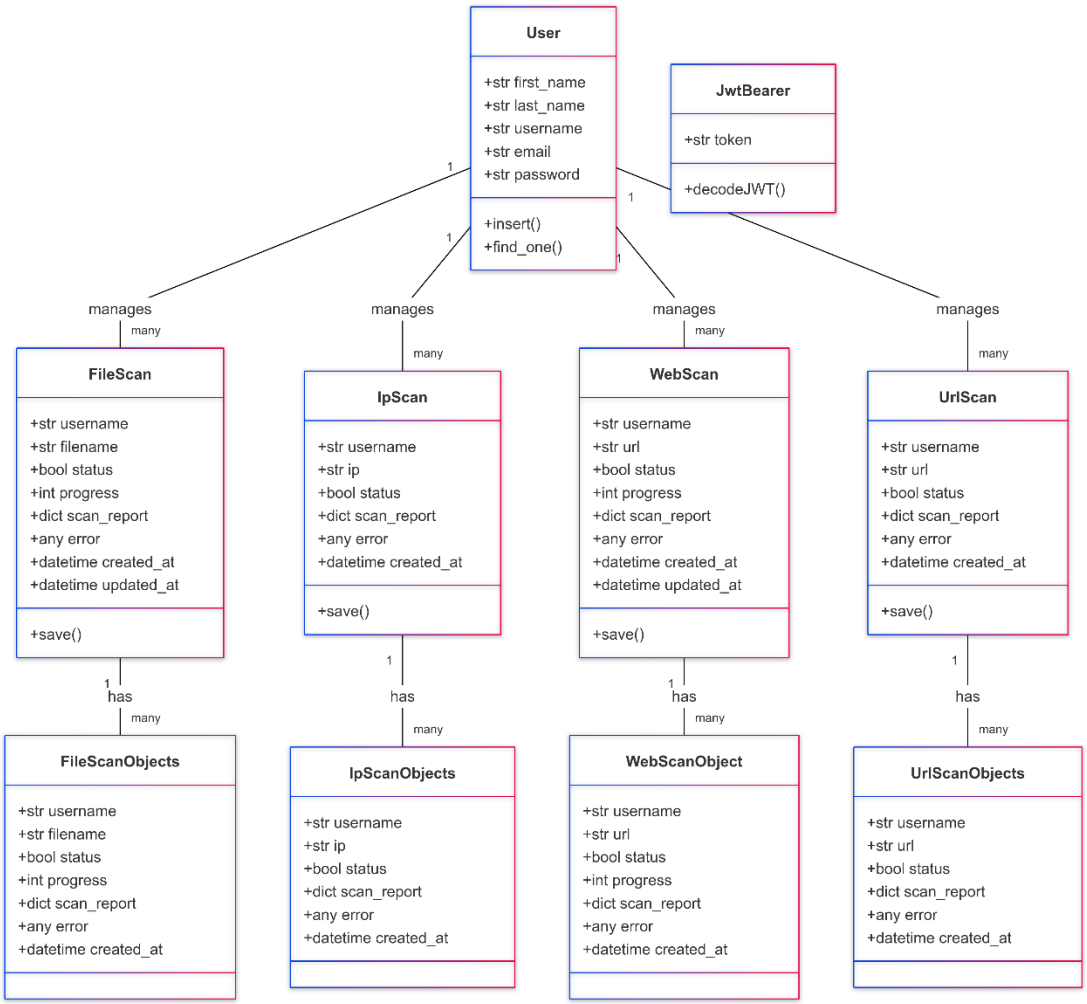

Figure 2.8.1: Class Diagram

2.8.2 Flowchart

The flowchart demonstrate the process flow of a web application with users logging in with their credentials which are confirmed by the artifact JWT token. Once a user logs in successfully, he/she can perform File, IP, Web or URL scans. Every scan activity initiates its specific service that categorizes and saves the scan result into the database. After the invention of the scan, the user can get the scan by entering the correct scan ID that is issued to the scan. If the scan ID is valid, then the report picked from the database is displayed to the user; in case of invalid scan ID the error message is displayed to the user. The user is then able to come to a stop of the session after reviewing the outcomengthening successful intervention.

Figure 2.8.2 Flowchart

2.8.3 Entities Relationship Diagram

The ER diagram represents the data structure of a system where a User can perform various types of scans, including file, IP, URL, and web scans. Each User has a unique username, email, and personal details, and is associated with multiple scan records. The FILE_SCAN**,** IP_SCAN**,** URL_SCAN**,** and WEB_SCAN entities store the results of the corresponding scans, including attributes like the scan status, report, error messages, and creation timestamps. The relationships between entities indicate that a **User** can have many scan records, creating a one-to-many relationship between the **User** and each type of scan. The diagram reflects how the user's actions (performing scans) are captured in the database, with each scan type being linked to the user who initiated it.

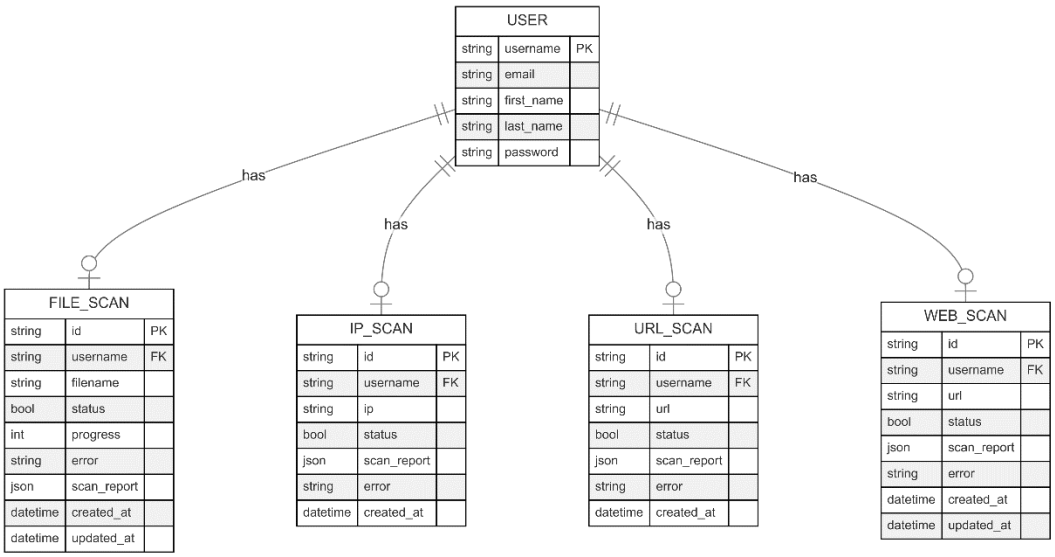

Figure 2.8.3 ER Diagram

2.9 Use Case Scenarios

2.9.1 Signup as User

Table 2.9.1 Sign Up as User

Use Case Name	Signup as User
Unique Identifier	UC-1
Objective	Allow a new user to register for an account.
Priority	High
Flow of Events	The user goes to the Sign Up page of the website. The user enters the required fields (First name, Last name, Username, Email, Password and Confirm password). In case of discrepancy the system also confirms certain input of information. If validation is successful, a new account is created into the system. Another email confirming the actions which the user did is sent to the user's email with an activation link. The user has to click the activation link to prove that he possesses the e-mail. The user is taken to the borrowing page that redirects the user to the Login page.
Precondition	The user is not yet registered in the system. The user has access to the **Sign-Up** page.
Post Conditions	The user account is successfully created and activated. The user is redirected to the **Login** page

2.9.2 Login as User

Table 2.9.2 Login as a User

Use Case Name	Login as User
Unique Identifier	UC-2
Objective	Allow an existing user to log into their account.
Priority	High
Flow of Events	When using any of the systems, the user accesses the Login page. User types his username/email and password. It can check their credentials The system can authenticate the individuals. When the user enters authentic credentials, then

	he is transferred to the dashboard page. If, however, all the credentials inputted by the user are invalid, the system returns an error message and waits for more input from the user.
Precondition	The user has a valid account on the platform. The user has internet access.
Post Condition	The user is successfully logged into their account and directed to the **Dashboard**.

2.9.3 Initiate Website Scanning

Table 2.9.3 Initiate Website Scanning

Use Case Name	Initiate Website Scanning
Unique Identifier	UC-3
Objective	Allow the user to initiate a website scanning process to identify vulnerabilities.
Priority	High
Flow of Events	The user logs in his/her account and goes to Website Scanning tab. The user inputs the www of the site they wish to scan. The system checks for the URL format that is entered into the system. The user types the URL and the scan of the specified website starts in order to find vulnerabilities. The system then presents a scan report once the scan is done together with the vulnerability identified, the severity level of the vulnerability and the course of action to rectify it. Finally, the user of the website has the opportunity to download the report in PDF or Word.
Precondition	The user is logged into their account. The user has a valid website URL to scan.
Post Condition	A website scan report is generated, and the user can view and download it.

2.9.4 Initiate IP Scanning

Table 2.9.4 Initiate IP Scanning

Use Case Name	Initiate IP Scanning
Unique Identifier	UC-4
Objective	Allow the user to scan a network or IP address to identify security vulnerabilities.
Priority	Medium
Flow of Events	The user goes to his/her account, and then moves to the IP Scanning feature of the website. The user inputs the field containing IP address or range of IP to be given a scan. The system checks if the given format of the IP address conforms to reality. This IP is then presented to the scanning process whereby the system scans for open port, vulnerability and probable security threat. This makes the system produce a thorough scan report that includes the outlined risks to a system and how they can be addressed. The user can either download the scan report or view it from the same page.
Precondition	The user is logged into their account. The user has the IP address or range to scan.
Post Condition	A scan report for the provided IP address is generated, and the user can download it.

2.9.5 Initiate Document Scanning

Table 2.9.5 Initiate Document Scanning

Use Case Name	Initiate Document Scanning
Unique Identifier	UC-5
Objective	Allow the user to scan documents for malware or sensitive information leakage.
Priority	Medium
Flow of Events	This is because when the user is logged in the account to their user control panel and go to the Document Scanning. In this

	scanning technique the user submits a document that they would like to have scanned for instance a PDF or word document. It examines the document for the presence of viruses such as macro; as well as scanning the document for any data that is sensitive. The system provides a report as to whether problems where detected and suggest courses of action to take if so. These format of the report made available in the website for the user to download as a PDF or Word format.
Precondition	The user is logged into their account. The user has a document to upload for scanning.
Post Condition	A document scan report is generated, and the user can download it.

2.9.6 View Scan History

Table 2.9.6 View Scan History

Use Case Name	View Scan History
Unique Identifier	UC-6
Objective	Allow the user to view and access past scan reports.
Priority	Low
Flow of Events	The user also opens their account and go to the History tab. The symbol initializes a list of all prior scans operated by the user (website, IP, and document scans). Conveniently, the user can filter the list of saves either by the type of scan or by the date it was performed. The user chooses one of the previous scans and analyses report with more details. The user can download the additional information in the form of a report to study it further on his own.
Precondition	The user is logged into their account. The user has previously performed scans using the platform.
Post Condition	The user can access and download previous scan reports.

CHAPTER 3

DESIGN AND METHODOLOGY

3.1 Methodology

It offers virtually complete and secure security analysis that can scan and detect threats in the websites, URLs, IPs, and files that users uploaded. The proposed methodology focuses on utilizing a highly effective back-end system coordinated with an optimal front end for users; it enables users to obtain elaborate reports on vulnerabilities in cybersecurity evaluation. Important design criteria are stability, extensibility and modularity. That way the platform will be able to accommodate the individual customers right up to the big multinationals. It also has a modular design within it means that additional security options could be added to it and/or it can cope with growing volumes of data in the future.

3.2 Feasibility Plans

3.2.1 Resource Requirements

Human Resources

- **Backend Developers**: Responsible for writing the core logic for scanning, report generation, and data processing.
- **Frontend Developers**: Ensure the user interface is intuitive and responsive, providing a smooth user experience.
- **Cybersecurity Experts**: Assist in the design of scanning algorithms and ensure the platform remains up-to-date with current cybersecurity threats.
- **Database Administrators**: Handle the management of databases, ensuring data security and optimization for large datasets.

Infrastructure

- **Servers**: A robust cloud-based server environment is used to host the application, ensuring high availability and scalability.
- **Database**: MongoDB is utilized as a NoSQL database for storing user information, scan results, and logs. Its flexibility allows efficient management of varied data types.
- **IT Equipment**: Workstations for development and testing, with appropriate software tools installed.

3.3 Risks Involved

Server Downtime: Should the physical host that supports the online platform develop network or hardware issues, the software must be shut down, and users cannot start scans or view reports. To address this, the system runs in a cloud environment allowing for failures to be handled by a failover system and back up systems.

Internet Connectivity Issues: Another vulnerability arises from the fact that the platform is based on constant scanning, so the offline or low speed connection may negatively affect the usage of the system or the process of scanning. In the event that the system identifies connectivity issues, users will be informed and a retry approach instituted.

Browser Compatibility: The platform is to be functional in and be compatible with such browsers as Chrome, Firefox, and Edge. To counter this, cross browser testing is done, while the frontend is developed using the responsive design to make sure it works on different screen sizes and resolutions.

Scan Time and Service Unavailability: Large web sites or documents may take considerable time to scan and some scans may take considerable time to complete. If scans are made long they may lead to the slow unlocking of any other tool on the platform. To reduce this impact, the asynchronous background processing is done, and a progress dialog is shown for long running scan.

CHAPTER 4

DATA AND EXPERIMENTS (and/or IMPLMENTATION)

4.1 Technologies used

4.1.1 Frontend Development

- **Technologies**: HTML5, CSS3, JavaScript for a responsive and intuitive user interface.

4.1.2 Backend Development

Python: The backend logic for the functions is in python due to flexibility, availability of secure libraries and the language ability to handle simultaneous requests.

FastAPI: The APIs used for creating high performance REST APIs which allows users to interact with the system with ease. The asynchronous functionalities of FastAPI allow the system to execute multiple scans at once.

Background Task Processing: Most of the background work such as web scanning or large file scanning is done using Celery or FastAPI's Background Tasks to allow the system to maintain responsiveness.

Docker: For containerization of the application, ensuring consistent environments across different stages of development and deployment.

4.1.3 Database

MongoDB: It is used for storing user details, scan records, and generated reports.

4.2 Experiments Conducted

The following experiments were conducted to evaluate the system's performance.

4.2.1 Performance Testing

- **Objective**: Measure the system's response time for different scan types.

- **Method**: Scanned datasets of varying sizes and complexity while recording processing times.

4.2.2 Accuracy Testing

- **Objective**: Validate the correctness of detected vulnerabilities or threats.
- **Method**: Cross-referenced scan results with existing vulnerability databases to calculate detection accuracy.

4.2.3 Scalability Testing

- Objective: Evaluate the system's performance under high user loads.
- Method: Simulated concurrent users accessing the platform and performing scans.

4.2.4 Usability Testing

- **Objective**: Assess the system's user interface and user experience.
- **Method**: Collected feedback from beta testers to improve usabili

CHAPTER 5

USER MANUAL

5.1 Welcome Screen of SecureNet Analyzer (Dashboard)

User can access dashboard after registering (Sign-up or Sign-in)

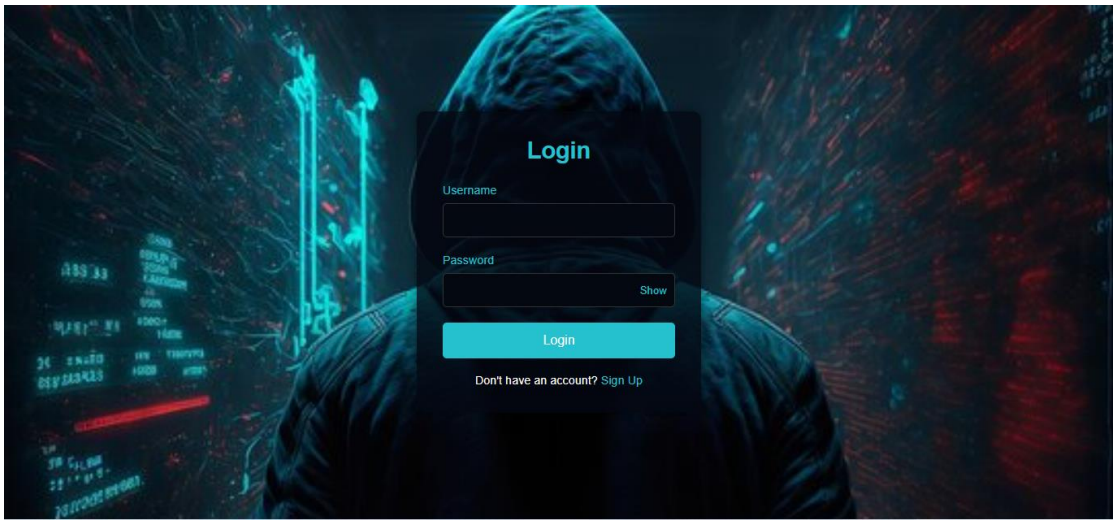

Figure 5.1.1 Welcome Screen

5.2 Sign-up screen

User can register by just signing up

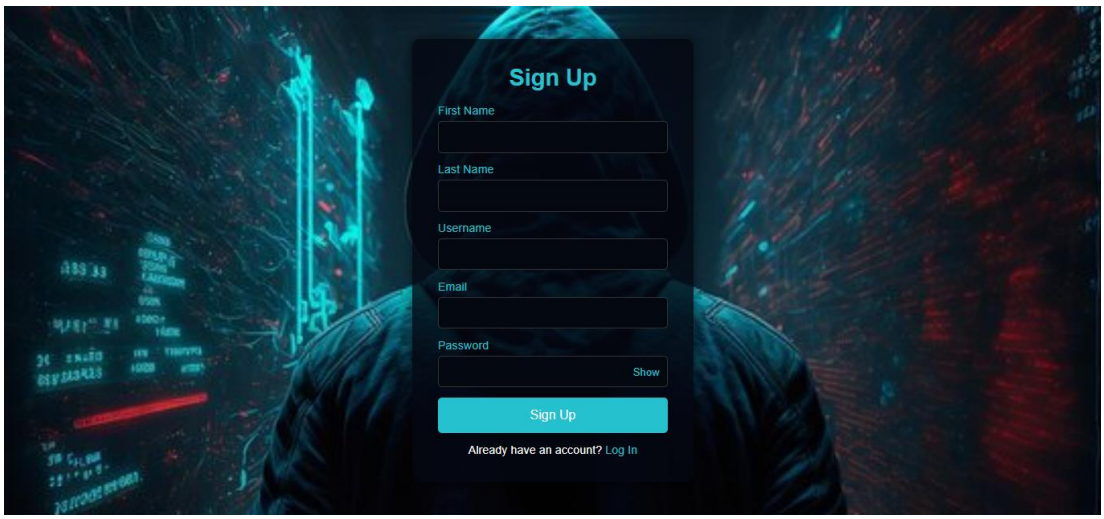

Figure 5.2.1 User Sign-up screen

5.3 Sign in Screen

If the user is already registered, then sign-in is required to verify credentials.

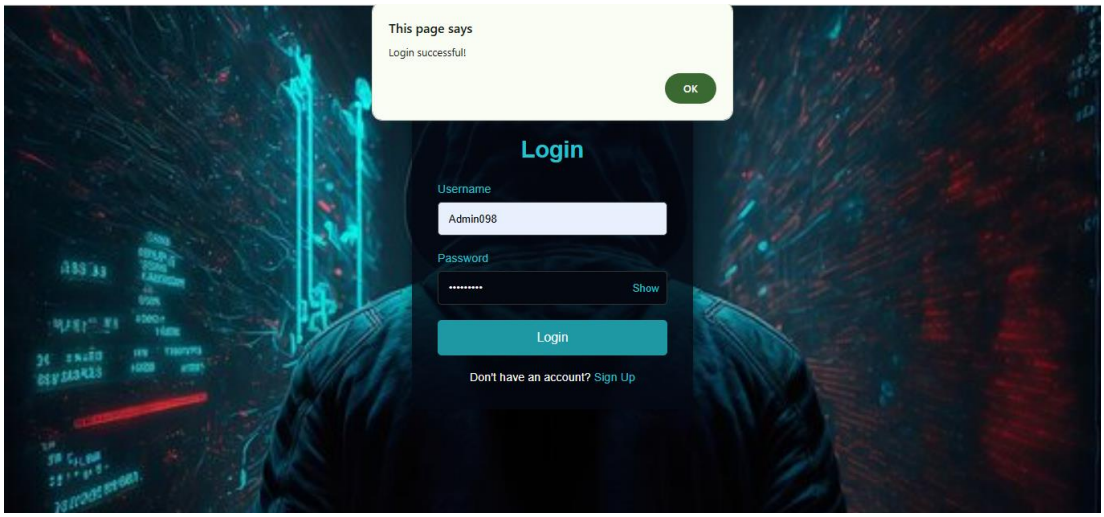

Figure 5.3.1 User Sign-in screen

5.4 Scanning Services

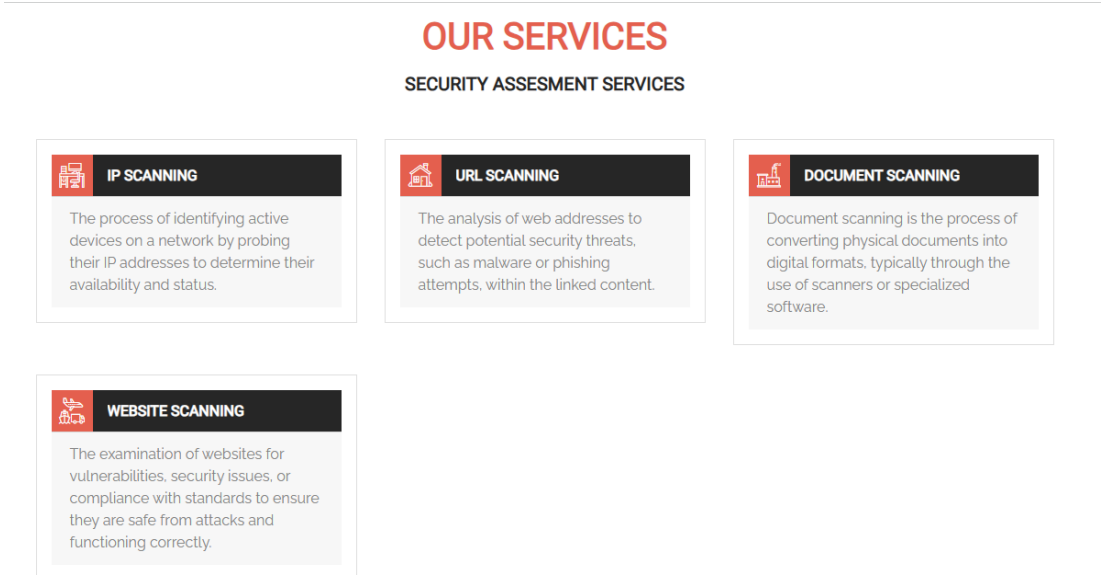

Figure 5.4.1 Scanning Services

5.5 Ip Scanning

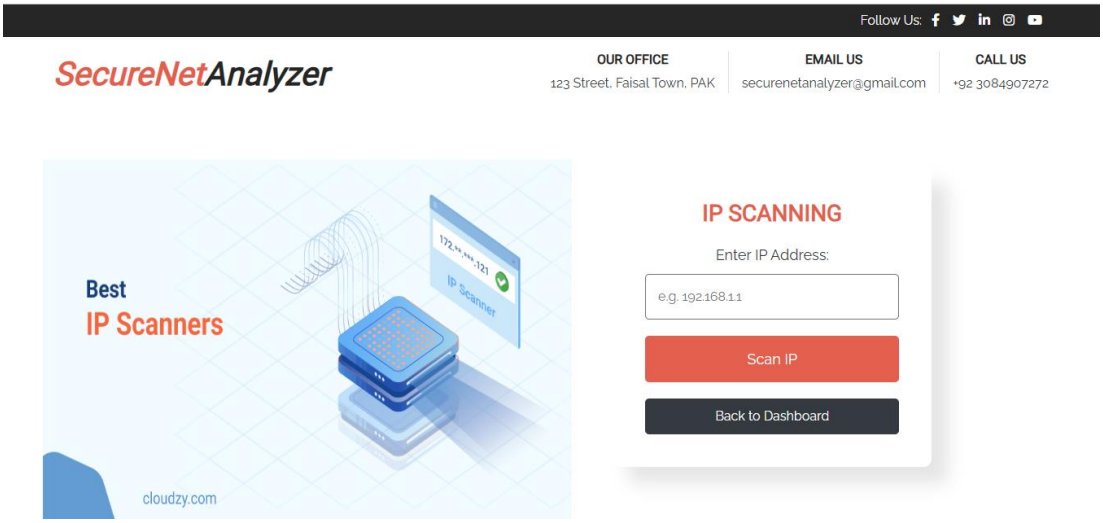

Figure 5.5.1 IP Scanning Screen

5.6 URL Scanning

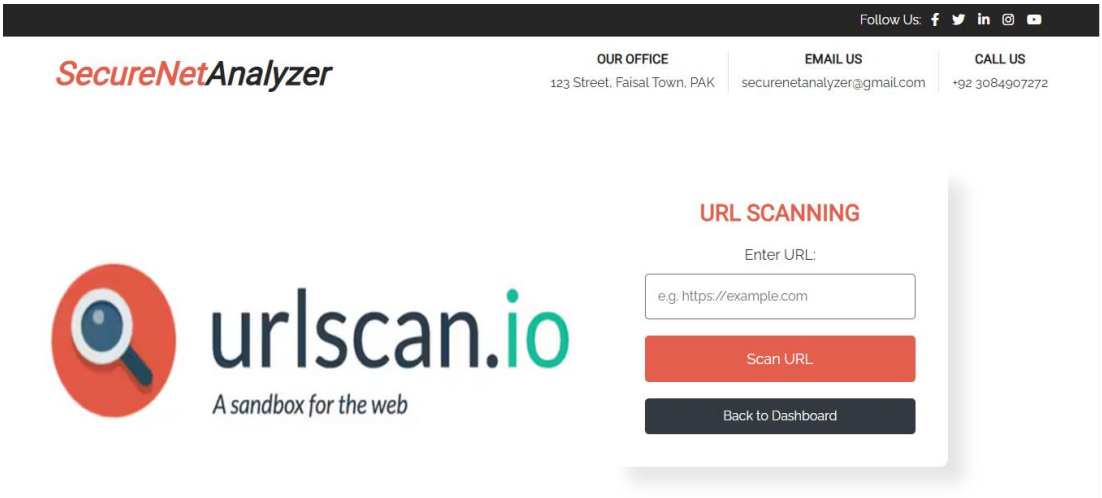

Figure 5.6.1 URL Scanning Screen

5.7 Document Scanning

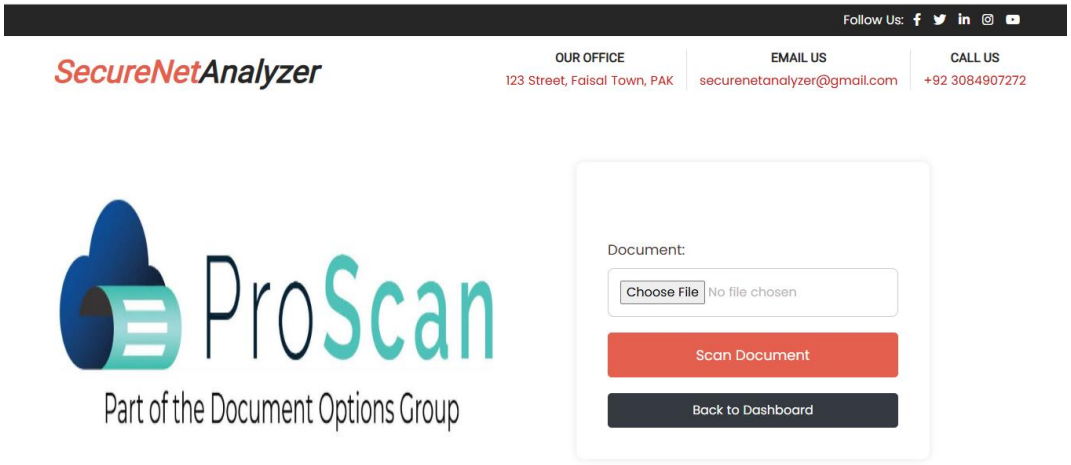

Figure 5.7.1 Document Scanning Screen

5.8 Website scanning

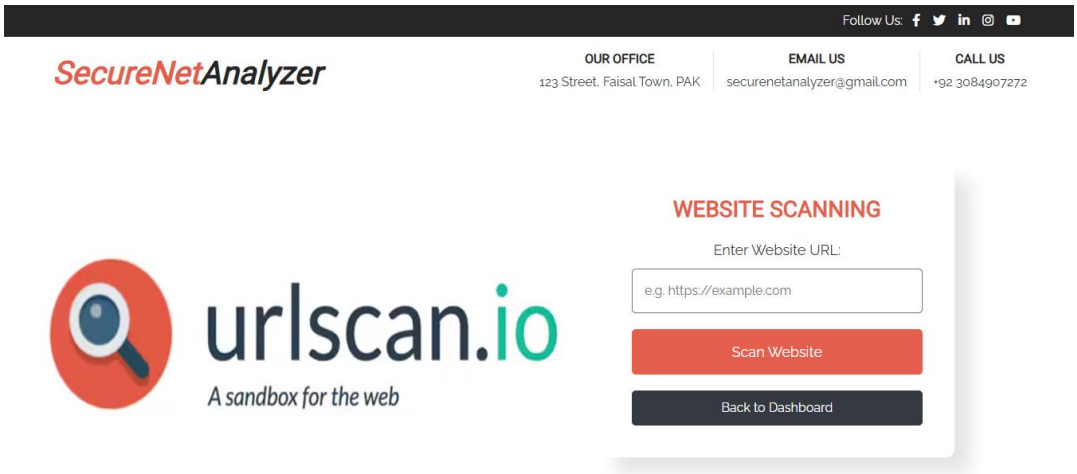

Figure 5.8.1 Website scanning screen

CHAPTER 6

CONCLUSION AND RECOMMENDATIONS

6.1 Conclusion

By using SecureNet Analyzer, it is possible to prove that it is the universal tool for cyber security allowing identifying and eliminating risks in IPs, URLs, documents, and websites. By incorporating features as Advanced Technology Scanning, Critical Realism Simulation of Cyber-attacks, and Detailed Report Generation, the project aims to satisfy the existing and emerging need to be ahead of other parties and take precautionary measures for an increasingly common problem of cyber threats..

Key achievements include

- Very simple graphical user interface to meet the requirements of both experienced computer users and novices.
- Adoption of specific scanner engines, which in the most efficient manner, identify threats and risks.
- Real-time generation of detailed reports in PDF and Word formats.
- Promotion of compliance with policies and procedures laid down under cyber security functional standards and requirements of the sector.

6.2 Recommendations

Despite the achievements of the SecureNet Analyzer in fulfilling its aims and purposes, there are some opportunities for further improvement and expansion of the possibilities to use it in modern conditions. They include enhancing the use of AI machine learning to detect threats in real-time, dually and shifting the service delivery model to cloud based, and creating smart phone application for scanning and reporting while on the move.

REFERENCES

1. https://ieeexplore.ieee.org/abstract/document/7945697
2. https://www.emerald.com/insight/content/doi/10.1108/IJWIS-01-2018-0001/full/html
3. https://www.proquest.com/openview/21b5d4f6c394cadc79a40e91f2d7ef4e/1?pq-origsite=gscholar&cbl=18750

APPENDICES

Appendix A: Glossary of Terms

- **IP Address**: A numerical label assigned to each device connected to a computer network that uses the Internet Protocol for communication.
- **URL**: Uniform Resource Locator, the address of a webpage or resource on the internet.
- **Vulnerability**: A weakness in a system that can be exploited by threats to gain unauthorized access or cause damage.
- **Penetration Testing**: Simulated cyberattacks performed to test a system's defenses.
- **OWASP**: Open Web Application Security Project, a foundation that works to improve software security.

Appendix B: System Requirements

1. **Hardware Requirements**
    - Processor: Intel Core i5 or higher
    - RAM: 8 GB or more
    - Storage: 20 GB available disk space
    - Network: Stable internet connection
2. **Software Requirements**
    - Operating System: Windows 10, macOS, or Linux
    - Browser: Latest versions of Chrome, Firefox, or Edge
    - Frameworks: Python 3.9+
    - Database: MongoDB

Appendix C: Test Cases

1. **Login Functionality**
    - **Test Case**: Verify login with valid credentials.
    - **Expected Result**: User is redirected to the dashboard.
2. **IP Scanning**
    - **Test Case**: Provide an invalid IP address.
    - **Expected Result**: System displays an error message indicating invalid input.

Appendix D: Project Timeline

Phase	Duration	Tasks Completed
Planning & Analysis	2 Weeks	Requirement gathering and tool selection
Design	3 Weeks	UI/UX design, database schema design
Development	6 Weeks	Front-end and back-end implementation
Testing	3 Weeks	Functional, security and usability testing
Documentation	2 Weeks	Preparation of documentation