A PROJECT REPORT ON

“**Website Scanner”**

SUBMITTED TO

SHIVAJI UNIVERSITY, KOLHAPUR

**IN THE PARTIAL FULFILLMENT OF REQUIREMENT**

FOR THE AWARD OF DEGREE

BACHELOR OF TECHNOLOGY IN INFORMATION TECHNOLOGY AND ENGINEERING

SUBMITTED BY

**Ms. Shital S Gaikwad 17UIT11011XX**

**Ms. Mayuri A Koli 17UIT11022XX**

**Ms. Monika N Masal 17UIT11035XX**

**Ms. Manali S Kulkarni 17UIT11025XX**

**Mr. Rishikesh R Patil 16UIT12071XX**

UNDER THE GUIDANCE OF

## Prof. D. M.Kulkarni



DEPARTMENT OF INFORMATION TECHNOLOGY AND ENGINEERING

D.K.T.E. SOCIETY’S TEXTILE AND ENGINEERING INSTITUTE, ICHALKARANJI

(An Autonomous Institute, Affiliated to Shivaji University, Kolhapur)

Accredited with 'A+' Grade by NAAC, An ISO 9001: 2015 Certified

## YEAR 2020-2021

**D.K.T.E. SOCIETY’S**

**TEXTILE AND ENGINEERING INSTITUTE, ICHALKARANJI**

(An Autonomous Institute, Affiliated to Shivaji University, Kolhapur)

Accredited with 'A+' Grade by NAAC, An ISO 9001: 2015 Certified

## **DEPARTMENT OF INFORMATION TECHNOLOGY AND ENGINEERING**

**CERTIFICATE**

This is to certify that, project work entitled

“**Website Scanner”**

is a bonafide record of project work carried out by

**Ms. Shital S Gaikwad 17UIT11011XX**

**Ms. Mayuri A Koli 17UIT11022XX**

**Ms. Monika N Masal 17UIT11035XX**

**Ms. Manali S Kulkarni 17UIT11025XX**

**Mr. Rishikesh R Patil 16UIT12071XX**

In the partial fulfillment of award of degree, Bachelor of Technology in Computer Science and Engineering prescribed by Shivaji University, Kolhapur for the academic year 2020-2021.

Prof. D. M.Kulkarni (PROJECT GUIDE)

Prof. ( Dr.) D.V.KODAVADE Prof.(Dr.) P.V.KADOLE (H.O.D. I.T) (DIRECTOR)

EXAMINER

# DECLARATION

We Students From B.Tech Of Branch Information Technology Are Here Declaring that, the Mega Project report With Title **Website Scanner**written and also submitted under the guidance of Respected Prof. D. M. Kulkarni is original work From Our Mega Project Group. The Information And Foundation Of this report are Totally based on the data collected by us with full sincerity. The matter combined in this report is not result of reproduction of any readymade report.

**PRN Name Signature**

17UIT11011XX Miss. Shital S Gaikwad

17UIT11022XX Miss. Mayuri A Koli

17UIT11025XX Miss. Manali S Kulkarni

17UIT11035XX Miss. Monika N Masal

16UIT12071XX Mr. Rishikesh R Patil

# ACKNOWLEDGEMENT

With great pleasure we wish to express our deep sense of gratitude to prof. D. M. Kulkarni for her valuable guidelines, support and encouragement in completion of project report.Also, we would like to take opportunity to thank our head of department Dr. D. V. Kodavde for his co-operation in preparing this project report.

We feel gratified to record our cordial thank to other staff members of Information Technology department for their support, help and assistance which they extended as and when required.

Thank you,

Miss. Shital S Gaikwad 17UIT11011XX

Miss. Mayuri A Koli 17UIT11022XX

Miss. Manali S Kulkarni 17UIT11025XX

Miss. Monika N Masal 17UIT11035XX

Mr. Rishikesh R Patil 16UIT12071XX

# ABSTRACT

A Web Scanner is developed for scanning full web page of websites. On the whole the objective of the project is to detect security flaws. This is used for check security of website. A Security Vulnerability is a flaw, error, or weakness found in a security system that has the potential to be supported by a threat agent in order to compromise a secure network .

In short security is very important for each and every website so all the constraints like database, web server version etc. of website must be very secure. With the increasing concern for security in the network, many approaches are laid out that try to protect the network from unauthorized access. Web applications are typically developed with often deployed with security vulnerabilities and are hard time constraints.

web scanners can help to locate these vulnerabilities and are popular tools among developers of web applications. The purpose of the application is to protect the website from hackers .The security is important because for not loss data, not leak data. SQL injection and Cross Site Scripting (XSS) these are two types of the vulnerabilities in web applications. In our project for purpose checking security of website ZAP, Nikto , Nmap these tools are used.

**INDEX**

1. **Introduction**
2. **Problem Description**

a. Problem Definition

b. Need of the project with motivating example

c. Objectives of project

d. Scope & limitations of project

e. Timeline for project

**3. Background Study and Literature Overview**

a. Technology review

b. Literature review (Critical appraisal of earlier work in same area)

**4. Requirement Analysis**

**a. Functional Requirements**

1) External interface Requirements

a) User Interface requirements

b) Hardware interface requirements (if applicable)

c) Communication protocol requirements (if applicable)

2) Modular/ Component Requirements

**b. System Requirements**

1) Hardware Requirements

2) Operating System Requirements

3) Application or Web server Requirements (if any)

4) Database server Requirements (if any)

5) Tools and Technologies Requirements

**5 System Design**

a) Architectural Design

(Interaction between components and or modules)

b) User Interface Design

c) Algorithmic description

d) System Modeling

1) Data flow Diagram

2) Sequence Diagram

3) Use case Diagram

**6. Implementation**

a) Environmental settings for running the module

**7. Integration and Testing**

**8. Performance Analysis**

**9. Applications**

**10. Installation Guide and User Manual**

**11. Cost Estimation**

**12. Ethics**

**13. References**

**Introduction**

A penetration test, also known as a pen test, is a simulated cyber attack against your computer [system to check for exploitable](https://www.imperva.com/learn/application-security/cyber-security/) vulnerabilities. In the case of web application security, penetration testing is commonly used to augment a [web application firewall](https://www.imperva.com/products/web-application-firewall-waf/). Penetration testing can involve the attempted to find vulnerabilities, such as invalid inputs that are susceptible to code injection attacks. The first step involves Defining the scope and goals of a test, including the systems to be addressed and the testing methods to be used. Collecting intelligence to better understand how a target works and its potential vulnerabilities. The next process is to understand how the target application will respond to various intrusion attempts. Static analysis is typically done using Static analysis and dynamic analysis. Inspecting an application’s code to estimate the way it behaves while running. This is a more practical way of scanning the website, as it provides a real time view of an application’s performance. This stage uses web application attacks, such as [cross-site scripting](https://www.imperva.com/learn/application-security/cross-site-scripting-xss-attacks/), [SQL injection](https://www.imperva.com/learn/application-security/sql-injection-sqli/) to uncover a target’s vulnerabilities. Testers then try and find these vulnerabilities, typically by escalating privileges, stealing data, intercepting traffic, etc. to learn the damage they can do. The goal is to see if vulnerability can be used to achieve a persistent presence in the exploited system, which often remain in a system for months in order to steal an organization’s most sensitive data. This data is analyzed by personnel of security to help configure an enterprise’s [web application firewall](https://www.imperva.com/products/web-application-firewall-waf/) settings and also other application security solutions to patch [vulnerabilities](https://www.imperva.com/learn/application-security/vulnerability-management/) and protect against future attacks which can happen.

#### Problem definition:

Objective of the project is to detect the security flaws.

#### Aim and objectives of the Project:

* + To check website is secure or not
  + Provide the solution.

#### Scope and limitation of the Project:

###### Scope

* + It provides the Analyst with all necessary security issues and its solution to prevent by hackers.
  + To access and modify the data intended for them it provides users all necessary privileges.
  + It mostly automize the scanning process and all the data used but that doesn’t entirely replace the existing system
  + It eventually recognize the value and necessity of this system and understand the problems involved in the manual process.

###### Limitations

* + Vulnerabilities are also missed by A vulnerability scanning tool, there is no guarantee that systems are not vulnerable. And It is one of the biggest limitations which all scanning tools can have, because Still hackers can exploit vulnerabilities Which still present there. Two reasons possible for this are:-  
    -for example it has only just been found the scanner is not knowing of the vulnerability.  
    - The attack is not trivial to automate so the vulnerability is very much complex to be discovered by an automated tool.
* In order to ensure that the most recent vulnerabilities are found, you need to make sure the tool is continually updated.
* It can be hard to understand the impact of the findings/vulnerabilities of the scanning tool if you Particularly have a large IT infrastructure, lots of servers and services.

#### TimeLine of Project

|  |  |  |
| --- | --- | --- |
| **TOPIC** | **START DATE** | **END DATE** |
| Domain Selection | 30/07/2020 | 06/08/2020 |
| Domain Finalization | 06/08/2020 | 13/08/2020 |
| Selection of Problem Statement | 13/08/2020 | 20/08/2020 |
| Finalization of Problem Statement | 20/08/2020 | 03/09/2020 |
| Study on Research Paper | 03/09/2020 | 07/09/2020 |
| Documentation of Synopsis | 07/09/2020 | 14/09/2020 |
| Requirement Analysis | 14/09/2020 | 17/09/2020 |
| System Requirement | 17/09/2020 | 21/09/2020 |
| Module Identification | 21/09/2020 | 24/09/2020 |
| System Architecture | 24/09/2020 | 29/07/2020 |
| Implementation 25% | 29/07/2020 | 12/10/2020 |
| Testing 25% | 12/10/20120 | 07/01/2021 |
| Implementation 50% | 07/01/2021 | 28/01/2021 |
| Testing 50% | 28/01/2021 | 31/01/2021 |
| Implementation 75% | 31/01/2021 | 14/02/2021 |
| Testing 75% | 14/02/2021 | 22/02/2021 |
| Implementation 100% | 01/03/2021 | 15/04/2021 |
| Testing 100% | 16/04/2021 | 05/05/2021 |
| Report Making | 15/09/2020 | 10/05/2021 |

# Background study

**and Literature overview**

## 2. Literature Overview

##### Literature Overview:

**1. Technology review :-**

Penetration tool web vulnerability scanner is an automated web application security testing tool. penetration audits our web applications by checking the vulnerabilities. Cross Site Scripting (XSS), SQL injection these RE Some examples of vulnerabilities. Through penetration we can scan the vulnerability of any website or any web application that is accessible via a web browser and use the HTTP or HTTPS protocols. penetration Web Vulnerability Scanner is one of the most useful tools in discovering [security](https://www.bartleby.com/topics/security) holes or vulnerabilities.

**2. Literature review (Critical appraisal of earlier work in same area):-**

A literature survey by Li and Xue [36] broadly systematized the techniques within the area of web application security. They proposed a framework with three main categories for securing applications: security by construction, verification or protection. Security by verification holds the vulnerability analyzing techniques of program analysis and testing. Security by protection contains run-time techniques like instrumentation of components and proxy safeguards .Li and Xue [38] later completed another literature survey, focusing more into on how the server side of web applications get secured in the some categories proposed in their previous work means they secured in 3 categories [36]. input validation problems, session management vulnerabilities and application logic vulnerabilities. Deepa and Thilagam [17] did encompasses access control, XSS and workflow violation after comprehensive survey looking into injection vulnerabilities, session management vulnerabilities, and what they classify as logic vulnerabilities.

They had a paper count to give an overview of their broadness of their overview of other literature reviews. For example for review of other literature studies, they are stating that for[38], 115 papers (which is the total number of references) were considered for individual survey. The papers that go into their taxonomy (their main table), is only 75, which would be a more correct number. A few of the papers that go into the primary body of their survey appear to be slightly imprecisely classified. They categorize MACE [42] under dynamic and black-box analysis, while what is found in this project is that they perform static, white-box analysis.

##### Investigation of current Project and Related work:

Web vulnerability scanners are the best way to protect your web application from malicious hackers. Because of the increase in attacks, manual testing can’t keep up. So we need automatic scanner to find vulnerabilities in website. web application scanner is an automated vulnerability assessment solution that crawls a website (either automatically or has been trained) looking for vulnerabilities within web apps. The solution analyzes all web pages and files that it finds, and builds a structure of the entire website

Cybercriminals do not rest in their attempt to resize valuable data and information to take advantages of it. According to the Annual Threat Report from SonicWall, a firm specializing in data protection and advanced network security, attacks against web application grew by 56% in 2018.

The company also revealed that last year it stopped 10.52 bilion malware attacks. According to their data this is the highest number of cyber attacks ever recorded. It is precisely for this reason that more and more companies are investing in services that allow them to predict and prevent incidents.

# Requirement Analysis

## System Requirement

##### Software and Hardware requirements:

* Hardware requirement:
* System memory: 512 mb and above
* Hard disk-space: 20gb and above
* Software requirements:
* Windows 10
* Linux
* Web server: xampp
* Database : MySQL

## Functional Requirements:

1) External interface Requirements

1. Output design

1.Output Defination

2.Output type:-

External output

Internal output

Interface output

Operational output

3.Output media

1. Input design

1.Input Stages

2.Input type

3.Input media

2) Modular/ Component Requirements:-

1)New User : Register to login first time time

2)Existing User : Login page and use service

3) Admin : Manage database and user profile

# System Design

## System Design

a. Architectural Design of System :-

(Interaction between components and or modules)

Scanning Results

Website vulnerability Monitor

Result collector

Result Analyzer

Website 1

Website 3

Website 2

Evaluation Report

* 1. **Data Design**
     1. **Data Flow Diagram**

Website

URL

Vulnerability

(Result)

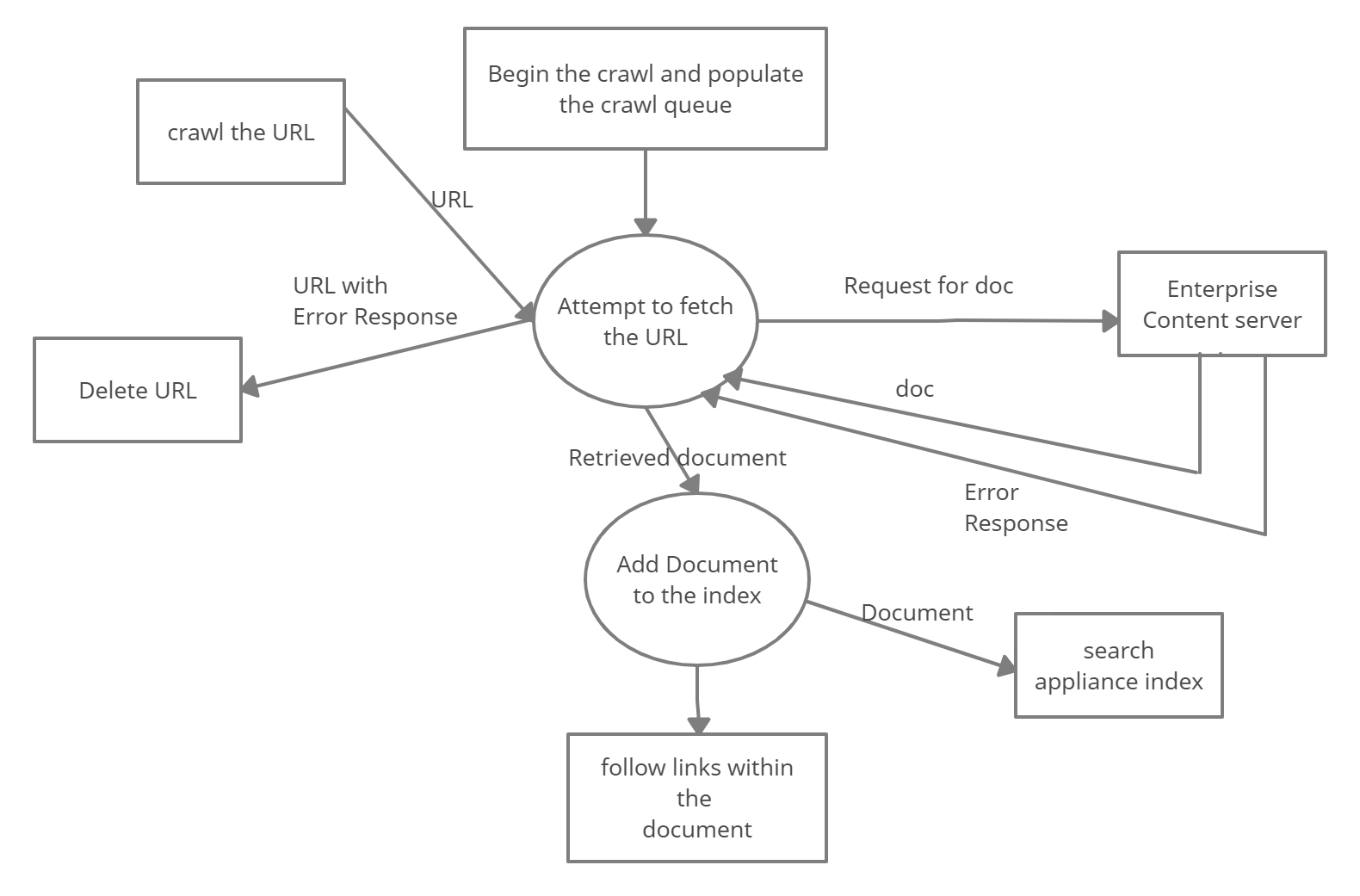
HTTP Request rkhiekjjjjejejejejejejejffbfbbfrREquestResponse

rRRRRREREREreRErResponse

HTTP Response

client

Level-0 Data Flow Diagram



Level**-**1(a) Data Flow Diagram

* + 1. **Sequence Diagram**

user

website

Website scanner system

Enter URL to be scanned

Pass URL to web scanner

Return result of web scanner

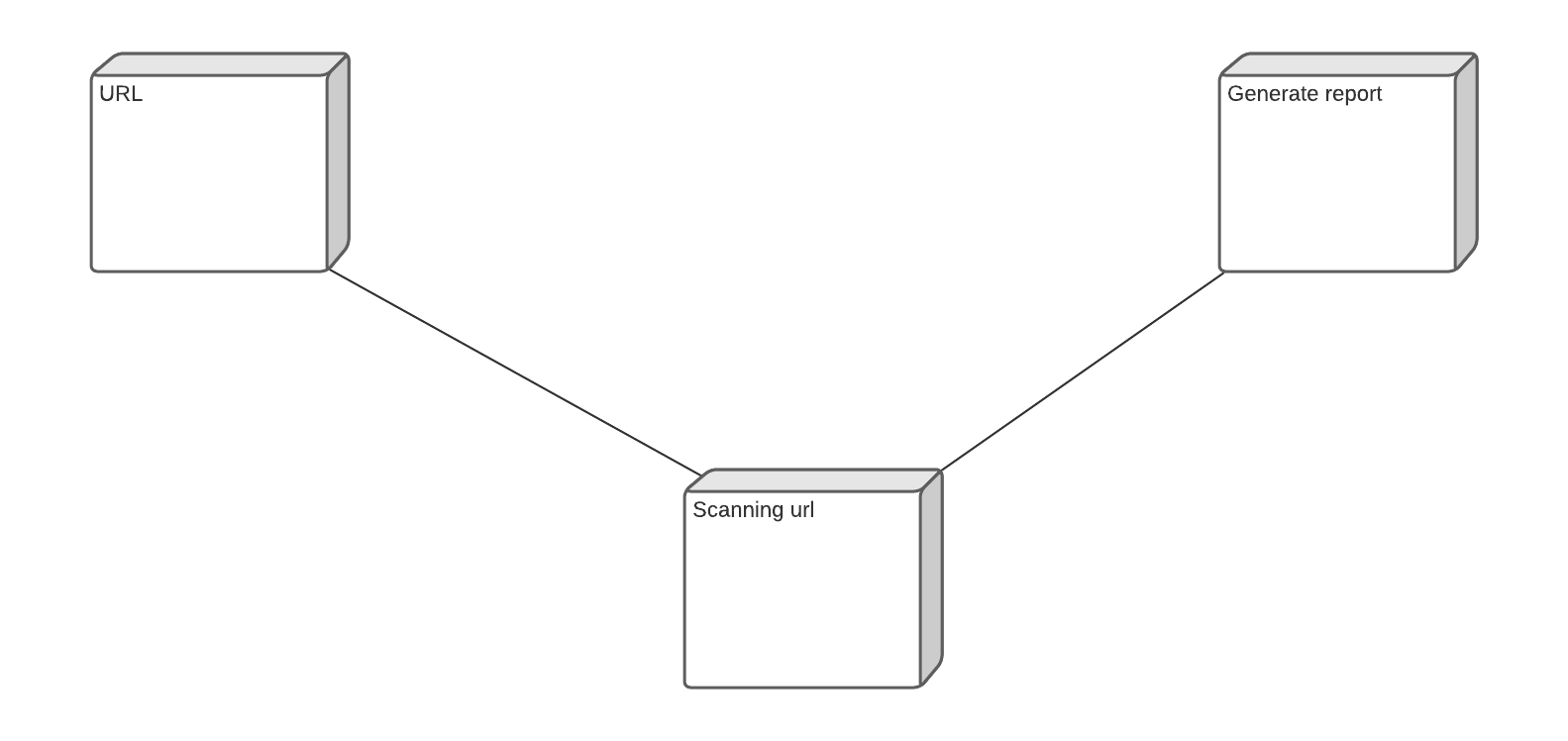
Show result of web scanner

Select link from the result

Give result of each link found by scanner

Show result of link one by one

* + 1. **Deployment Diagram**



**Implementation**

1. Detailed Description of Methods

* Implementation-

1] user login

2] application will take the URL of the website as input.

3] find vulnerabilities

4] if issue created:

i]notification send

ii] developer commits a fix

iii] continuous penetration testing

5] check website is secure or not

6] Get detailed technical reports to understand and address identified vulnerabilities

# Integration and Testing

## Integration and Testing

**Integration test cases generation and its testing reports**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test case No | Test case | Input | Expected  Output | Actual Output | Status |
| 01 | Vulnerability detection | Valid URL | Report Generated | Report Generated | Pass |
| 03 | Vulnerability detection | Invalid URL | Pop up message(please Enter valid URL) | Pop up message(please Enter valid URL) | Pass |

**System test cases generation and its testing report**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Test  case No |  | Test case |  | Input | Expected  Output |  | Actual Output |  | Status |
| 01 |  | Registration |  | User Data(Name, Password  Phone Number) | Successfully Registered and Navigate to Login page |  | Successfully Registered and Navigate to Login page |  | Pass |
| 02 |  | Login Using Registered Username and Password |  | Correct Username and Password | Navigate to Website scanner page |  | Navigate to Website scanner page |  | Pass |
| 03 |  | Login Using Username and Password |  | Wrong Username or Password | Navigate to Login page |  | Navigate to Login page |  | Pass |

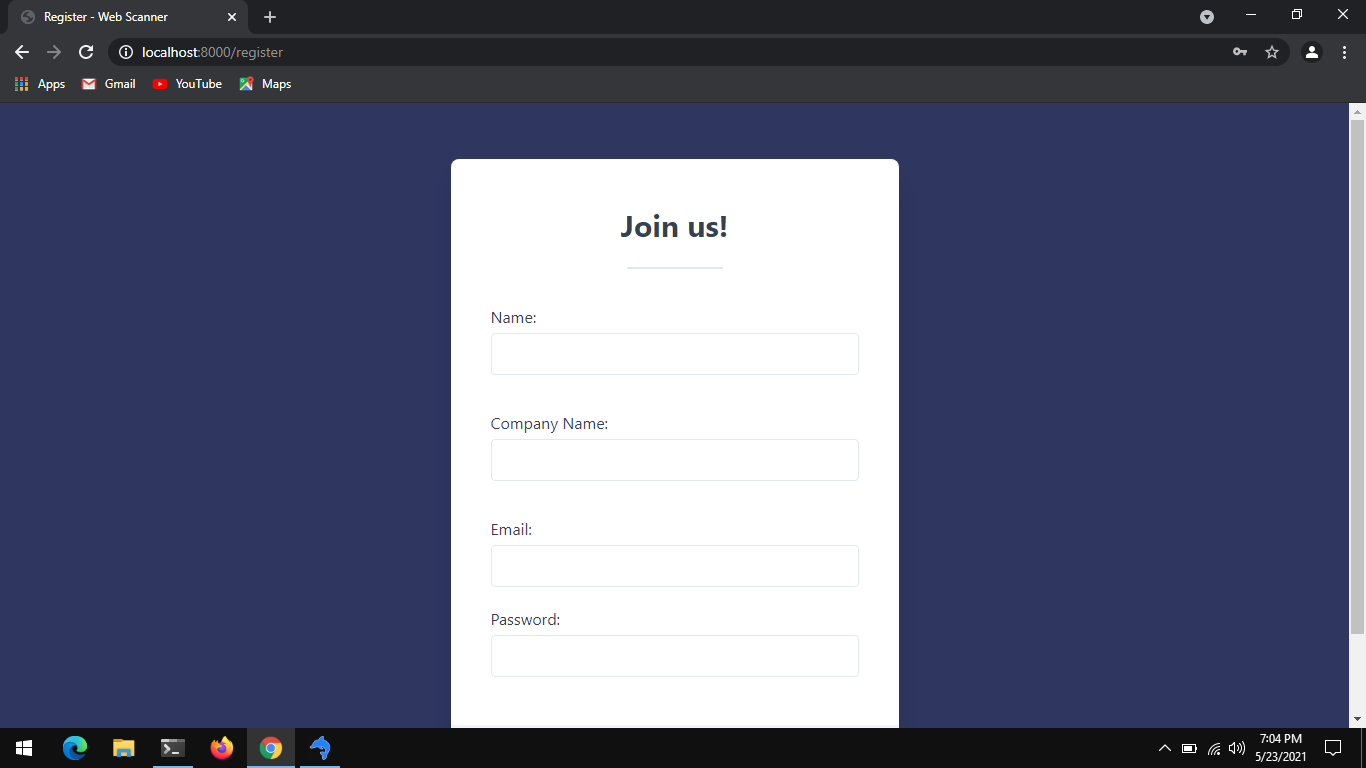
|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Test case No | Test case |  | Input |  | Expected  Output |  | Actual Output |  | Status |
| 04 |  | If User Enter  Username and password and click cancel Button |  | Username and Password | Navigate to Home page |  | Navigate to Home page |  | Pass |
| 05 |  | If User click cancel Button After successfully Login |  | Username and Password | Navigate to Home page |  | Navigate to Home page |  | Pass |
| 06 |  | User Enter Website URL |  | Valid URL | Generate  Report |  | Generate  Report |  | Pass |
| 08 |  | User Enter Website URL |  | Invalid URL | Pop up message will appear(please  Enter valid URL) |  | Pop up message will appear(please  Enter valid URL) |  | Pass |

**Performance Analysis**

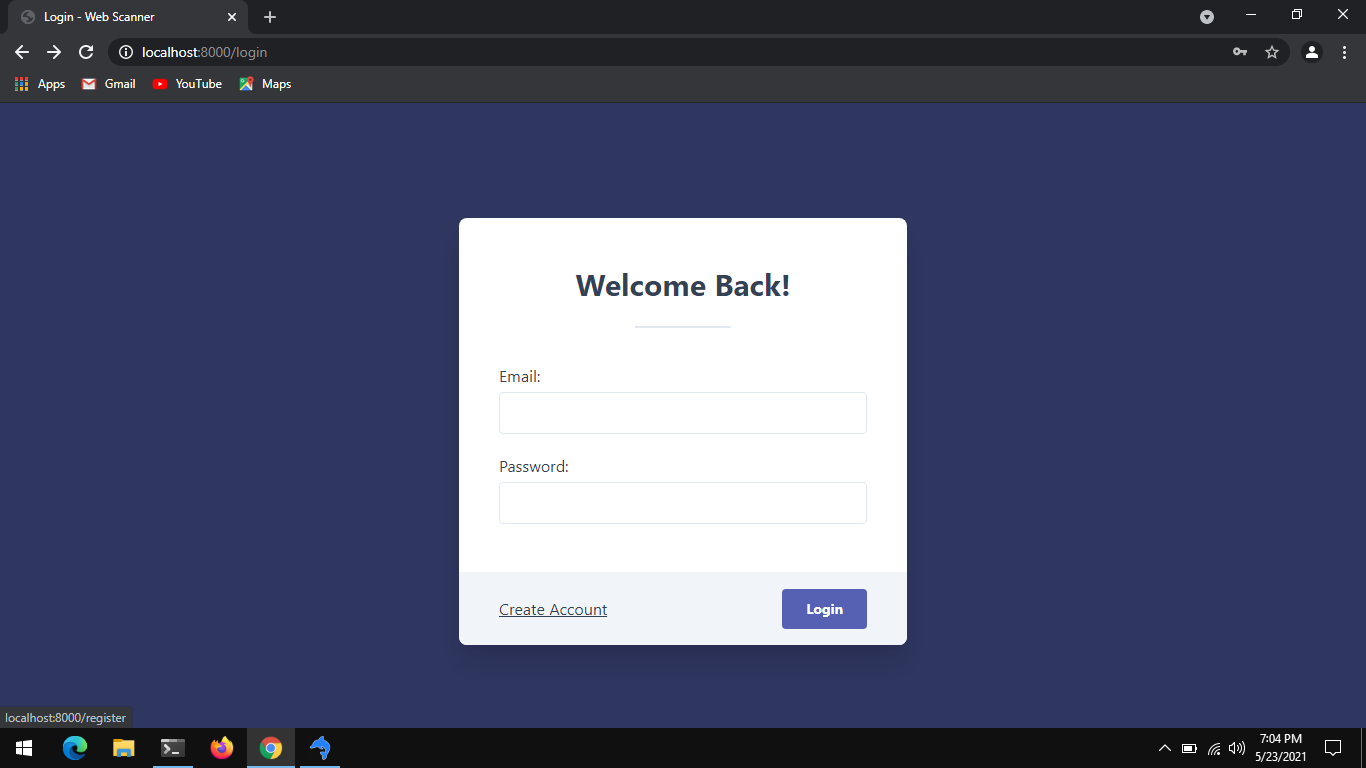
To develop a secure web application, one must know how they will be attacked. Penetration testing helps in finding vulnerabilities before an attacker does. The main goal of penetration testing is to allow easy penetration testing to find the vulnerabilities in web applications. Penetration testing tools create a proxy server and make the website traffic pass through the server. The use of auto scanners in penetration testing tools helps to intercept the vulnerabilities on the website.

**Output Screen**:

Screenshot 1: Registration form



Screenshot 2:  login page of website



# 

# Screenshot 3: homepage of website

# 

# 

# Screenshot 4:

# 

# Screenshot 5:

# 

# Screenshot 6:

# 

# Screenshot 7:

# 

# Applications

**Applications**

* User friendly registration system
* Faster web crawler
* Easy to control session
* Free registration
* Wide range of test
* Faster scanner

# Installation Guide and User Manual

### Installation :-

* 1. FirstInstall Apache and Updating the Firewall
  2. Then install MySQL.
  3. Then install php.

Steps to install -

## **1) Installing Apache and Updating the Firewall**

1)Install Apache using Ubuntu’s package manager, apt:

* $ sudo apt update
* $ sudo apt install apache2

2) To confirm Apache’s installation by pressing Y, then ENTER

3) Need to adjust firewall settings to allow HTTP traffic.

4) To list all currently available UFW application profiles, you can run:

* $ sudo ufw app list

5) To only allow traffic on port 80, use the Apache profile:

* $ sudo ufw allow in "Apache"

## **2)Installing MySQL**

1)used for install

* $ sudo apt install mysql-server

2) confirm installation by typing Y, and then ENTER.

3)Start the interactive script by running:

* $ sudo mysql\_secure\_installation

4) test if you’re able to log in to the MySQL console by typing:

* $ sudo mysql

5) To exit the MySQL console, type:

* exit

## **3) Installing PHP**

1)To install these packages, run:

$ sudo apt install php libapache2-mod-php php-mysql

# Cost Estimation

Project Cost

* 1. Hardware Cost:

|  |  |
| --- | --- |
| **Hardware** | **Cost** |
| Computer System | Rs. 45000/- |
| Internet | Rs.800/- |
| Light Source | Rs. 500/- |
| Total | Rs.46300/- |

In this project the Cost Estimation based on COCOMO (Constructive Cost Model) the formula for the this Model is follows

Effort = Constant × (Size) scale factor× Effort Multiplier Effort in terms of person-months

Constant: 2.45 in 1998 based on Organic Mode – Size: Estimated Size in KLOC –

Scale Factor: combined process factors

Effort Multiplier (EM): combined effort factors The basic COCOMO equations take the form Effort Applied (E) = ab(KLOC)b b [ man-months ]

Development Time (D) = cb(Effort Applied)d b [months]

People required (P) = Effort Applied / Development Time [count]

# ETHICS

#### Declaration of Ethics

As A Computer Science & Engineering Student, I believe it is Unethical To,

1. Surf the internet for personal interest and non-class related purposes during classes
2. Make a copy of software for personal or commercial use
3. Make a copy of software for a friend
4. Loan CDs of software to friends
5. Download pirated software from the internet
6. Distribute pirated software from the internet
7. Buy software with a single user license and then install it on multiple Computers
8. Share a pirated copy of software
9. Install a pirated copy of software

# References

* <https://www.researchgate.net/publication/4322871_Testing_and_Comparing_Web_Vulnerability_Scanning_Tools_for_SQL_Injection_and_XSS_Attacks>
* https://www.csoonline.com/article/3569609/4-best-practices-to-avoid-vulnerabilities-in-open-source-code.htmlhttps://bookauthority.org/author/Gen.-James-L-Jones
* [The **CARVER Target Analysis and Vulnerability Assessment Methodology**](https://www.amazon.com/dp/1732429707?tag=uuid10-20)

[A Practical Guide for Evaluating Security Vulnerabilities(](https://www.amazon.com/dp/1732429707?tag=uuid10-20)[Leo Labaj](https://bookauthority.org/author/Leo-Labaj)[,](https://www.amazon.com/dp/1732429707?tag=uuid10-20)[Luke Bencie](https://bookauthority.org/author/Luke-Bencie)[,](https://www.amazon.com/dp/1732429707?tag=uuid10-20)[Gen. James L Jones](https://bookauthority.org/author/Gen.-James-L-Jones)[)](https://www.amazon.com/dp/1732429707?tag=uuid10-20)

## [A three component hazards of place model](https://www.amazon.com/dp/3659193097?tag=uuid10-20)

### [**A First Complete Vulnerability Analysis (**](https://www.amazon.com/dp/3659193097?tag=uuid10-20)[**Charles Yorke**](https://bookauthority.org/author/Charles-Yorke) **)**

* [https://www.slideshare.net/kupiliarchana1/model-based-vulnerability-testing-report](https://www.amazon.com/dp/3659193097?tag=uuid10-20)