

National College of Ireland BSHC Computing Cybersecurity 2023/2024 Sinead Doherty X20424574

X20424574@student.ncirl.ie

CyberSafe Technical Report

Contents

Execut	tive Summary2
1.0	Introduction
1.1.	Background2
1.2.	Aims3
1.3.	Technology3
1.4.	Structure4
2.0	System5
2.1.	Requirements5
2.1.	1. Functional Requirements5
2.1.	1.1. Use Case Diagram5
2.1.	1.2. Requirement 1 <name a="" few="" in="" of="" requirement="" words="">5</name>
2.1.	1.3. Description & Priority5
2.1.	1.4. Use Case
2.1.2	2. Data Requirements
2.1.3	3. User Requirements
2.1.4	4. Environmental Requirements11
2.1.	5. Usability Requirements11
2.2.	Design & Architecture
2.3.	Implementation12
2.4.	Graphical User Interface (GUI)18
2.5.	Testing25
2.6.	Evaluation32
3.0	Conclusions
4.0	Further Development or Research
5.0	References
6.0	Appendices
6.1.	Project Proposal
6.2.	Reflective Journals
6.3.	Other materials used

Executive Summary .

In this report I will provide an overview of what my final year project (CyberSafe) is about and how I went about creating it . The purpose of this report is to show all the many different elements it took to create this application .The introduction section of this report will include the background of my idea and why I choose to do it and how I went about turning that idea into a working web application and the goals for this project overall. I will also provide in my system section an overview of the key features of my application, the different pages, all the technologies that I used for the websites system architecture (frontend and backend) as well as the main functionality and how each section of my web application works and the outcomes with use case diagrams. I will also show in this report some of the code highlights as well as the different types of testing I did on my web application and evaluation. At the end of this report I will include features that could be implemented to further the development of the web application in the future are detailed further into the report.

1.0 Introduction

1.1. Background

When I was looking for some inspiration for an cybersecurity project I wanted to do something that helps users stay more aware when using the internet. That's when I came up with the idea of doing a website scanner. I choose to undertake this project because I think it would be a useful website that can cater to anyone that uses the internet. I hope I can create this to help online users to become aware of what websites are safe and be able to trust these websites along with showing them what websites are not so safe to use/ what websites are legitimate. I feel that many of the other scanners that I would use on the internet and that I have researched never give a straightforward answer when it comes to using safe websites this is why I wanted to create a website that was straightforward with the answers. I also felt doing my research that I thought some of the major scanner websites looked over complicated and hard to follow, so with this application I wanted to create a webpage that was easy to follow for anyone using it and user friendly.

1.2. Aims

The overall aim of my web application project is the create a user-friendly website that anyone can use that will help to look after users safety when it comes to using websites. This project will be a website that can inspect different characteristics of websites. This website will help look for security characteristics of a website in terms of firewalls, certificates, date of creation, and files on a website to verify the that websites are trustworthy . I plan for my website to have different scans that a user can click on the use. All a user would have to do sign up to the website, pick the scan they want to use, copy their URL of their desired web page into my site and my website will scan the link and show the information in relation to the scan they have picked. This is to help users become aware of legitimate websites and which ones are safe and which sites are actually verified. I want my website to look aesthetically pleasing on the eye as well as it being easy to use and followed by users.

1.3. Technology

Languages: , HTML, CSS, bootstrap for the frontend.

java script, Flask, python, SQLite3 for the backend

API'S: IPWHOIS, WAF

- Requests, subprocess, socket
- Frontend:

HTML – used to create my different web pages

CSS and bootstrap – used for the styling of my HTML pages

Backend:

JavaScript and python – used to help with the key functionalities of my webpages which are the scanners and log in pages as well as helping to incorporate the API that are used for the different scanners.

Flask: used the help with the APIs

SQLite3 : used to store the users and there login in credentials when submitted.

1. Visual Studio Code, - used to make all my frontend and backend applications for my project.

2. GitHub – used to store those different parts of the project.

- 3. API'S for helping with the scanner part of my website
 - IPWHOIS = helps with getting the location information about typed in URL
 - Request = helps getting the fuzzy information about the URLS
 - Request = helps firewall scanner to detect the firewalls of the URL looking for firewall 'waf', 'cloudflare', 'akamaigHost', 'naxsi.
 - WAF (web application firewall) : used for firewall detector
 - subprocess, socket: used to help in getting the requested URL information need for my website.

1.4. Structure

This document will give an overview of what my project is about with a look at the system section, testing and conclusions.

The system section will look at what a user is to expect when using this website and the main flow of what the site should be and it's key functions. I will discuss the functional requirements whereby I describe what each part of the web application does also with some use case diagrams about the flow of my website. Then I will talk about some requirements and design and implication and the different systems in my web application. This section will also include some screen shots of the application and what each webpage does.

Then I will show the different tests I have done the application and the results of those different tests.

Structure of System section

- Functional Requitements what each page does and the order they do it in and how they do it.
- Use Case different scenarios that could happen in my page
- Requirements Data , Environmental, user and usability
- Design and Architecture
- Implementation
- GUIs of each webpage

Testing section

2.0 System

2.1. Requirements

All requirements should be verifiable. For example, experienced controllers shall be able to use all the system functions after a total of two hours training. After this training, the average number of errors made by experienced users shall not exceed two per day.

2.1.1. Functional Requirements

2.1.1.1. Use Case Diagram

2.1.1.2. Requirement 1 user registering / sign in

2.1.1.3. Description & Priority

This is how the application should run when user opens the site and goes to make an account for the website. The website first page opens the index page gives a brief overview of what my site is about and On this page the user can press the log in button to gain further access. They will then be taken to the sign in page and then they will click the resister to sign up. They will then make their account by filling in a username, email and password and these details will then be saved in a sqlite database once they click the sign up button. They will then be redirected to the page that will say that registering worked and they can continue to use the site. When the users gets to the main page they can log out which will return then to the index page. Then to re access the main page the user can just go straight to the log in and use the email and password that they used for registering and then they will return to the main page.

2.1.1.4. Use Case

Registering / signing in

Scope

The scope of this use case is to let users create and account for this website and signing in.

Description

This use case describes the way that user can access the website by making an account and signing in

Use Case Diagram



Flow Description

Precondition

The system is in initialisation mode when the website is open and on the first page.(index page)

Activation

This use case starts when an users open the website.

Main flow

- 1. The system identifies the application, and the index pages shows up
- 2. The user clicks the log in button and then the register button (See A1)
- 3. The user clicks the log in button and enters details before even registering (see E1)
- 4. The user clicks the log in button and then the register button but registering with pre-exciting email (See E2)
- 5. The user is redirected to the registering worked page and can move on

Alternate flow

- A1 : < Registering >
 - 1. The user creates an account by filling in their details.
 - 2. Details are submitted into the database.
 - 3. The use case continues at position 5 of the main flow

Exceptional flow

- E1 : <fail log in >
 - 1. redirects to log in fail page
 - 2. returns to step 1
- E1 :

- 1. redirects to email already in use page
- 2. returns to step 1

Termination

The system presents the next registering success page.

Post condition

The system goes into a wait state

2.1.1.5. Use Case Diagram

2.1.1.6. Requirement 2 Picking a scanner

2.1.1.7. Description & Priority

Once they user has log in / signed up to the website they are present with the main page of my website. The user will be presented with a drop down menu with three scanner to choose form. The user can also choose to log out of the webpage here and threat brings them back to the index page.

2.1.1.8. Use Case

Picking a scanner

Scope

The scope of this use case is to get the user to pick one of the scanners.

Description

This use case describes the process in which the user can choose which one of three scanners to be displayed on the next page

Use Case Diagram



Flow Description

Precondition

The user has signed in/up their account.

Activation

This use case starts when an users opens up the drop down menu to pick a scanner.

Main flow

- 1. The webpage loads
- 2. User is presented with main page of application
- 3. User views the scanners in drop down menu <A1>
- 4. User presses log out button. <E1>
- 5. Next page loads

Alternate flow

A1 : < Picking scanner>

- 1. User views the scanners in drop down menu.
- 2. User chooses scanner.

3. Step 5 of main flow takes place

Exceptional flow

Termination

The system loads the next webpage.

Post condition

The system goes into a wait state

2.1.1.9. Requirement 3 working scanner

2.1.1.10. Description & Priority

This use case describes how the scanner should work when a user puts there desired URL on to the page of which scanner they choose. After the user has picked there scanner to scan they should be redirected to the decried scan page when they can paste their URL, press enter and the results will show up

2.1.1.11. Use Case

Working scanner

Scope

The scope of this use case is how the website works when the users chooses a scanner to scan their URL

Description

This use case describes the process in which the user can choose which one of three scanners to be displayed on the next page, paste their URL, press enter and the results will show up

Use Case Diagram



Flow Description

Precondition

The system is in wait state.

Activation

This use case starts when a user has selected their scanner.

Main flow

- 1. Scan picked.
- 2. Scanner pages loads
- 3. User enters a Valid URL and clicks search <A1>
- 4. User views results of scan
- 5. User clicks return to home page.

Alternate flow

A1 : Scanner

- 1. User presses search
- 2. System scans URL with API

- 3. Results are shown.
- 4. Position 5 can take place of main flow.

E1 : Error

- 1. The user doesn't enter a valid URL.
- 2. Scanner do not work.
- 3. Return to position 2 of main flow.

Termination

The system presents the next webpage.

Post condition

The system goes into a wait state

List further functional requirements here, using the same structure as for Requirement1.

2.1.2. Data Requirements

The website will use APIS the get the information required from the URL that the users submit. With the information my website will display it onto a html page.

2.1.3. User Requirements

The user should find navigating my website with no issues as I have plan to make it as user friendly as possible. The user needs a web browser in order to access my website and all the user needs to do is sign up or sign in with and previous made account submit a valid URL for scanning.

2.1.4. Environmental Requirements

No environmental requirements need as on based on online system.

2.1.5. Usability Requirements

Can be access anywhere by anyone on any device. Just need an online browser to access the site

2.2. Design & Architecture



2.3. Implementation



This is the code made from python that takes the users inputs of the URL they want to scan for the location and fuzzy scans.

es Reset Filters	Records: 25					Sea	arch 25 records.	•••
	name	abc -	email	abc -	password		1	
	Search column		Search column		Search column			
1	4		4@email.com		1 B	Sytes 🕻	↓	
2	p		p@email.com		1 B	Sytes 🕻	J	
3	r		r@email.com		1 B	Sytes 🕻	J	
4	t		t@gmail.com		1 B	Sytes 🕻	1	
5	rt		rt@email.com		2 B	Sytes 🕻	1	
6	а		a@email.com		1 B	Sytes 🕻	↓	
7	е		em@email.cok		1 B	Sytes 🕻	↓	
8	new		new@email.co	m	3 B	Sytes 🕻	1	
9	pic		pic@email.com	ı	3 B	Sytes 🕻	Ţ	
10	8		8@email.com		1 B	Sytes 🕻	↓	
11	q		qq@email.com		1 B	Sytes 🕻	↓	
12	r		rr@email.com		2 B	Sytes 🕻	Ţ	
13	s		ss@gmail.com		2 B	Sytes 🕻	Ţ	
14	f		fff@email.com		1 B	Sytes 🕻	Ţ	
≪ ⊲ 1	▷ ▷	Page 1/	1			Try S	– SQLite Viewer W	/eb ↗
							Go Live	L.

SQlite database use to store users log in credentials.



This is the. Python cod of how the database is create and the table and how the username, email and password will be stored.

```
#-- the registery page
@app.route("/register_success", methods = ["POST", "GET"])
def register_success():
    if request.method == "POST":
        email = request.form["email"]
        if check_data(email):
            email = request.form["email"]
            name = request.form["name"]
            password = request.form["password"]
            print(name)
            print(email)
            print(password)
            insert(name, email, password)
            return render_template("register_success.html")
        else:
            return render_template("register_fail.html")
#-- the registery fail page
@app.route("/register_fail")
def register_fail():
    return render_template("register_fail.html")
#-- the log in pages
@app.route("/login_success", methods = ["POST", "GET"])
def login_success():
    if request.method == "POST":
        email = request.form["email"]
        print(email)
        password = request.form["password"]
        print(password)
        if check_login_data(email, password):
            return render_template("login_success.html")
        else:
            return render_template("login_fail.html")
if __name__ == "__main__":
    app.run(debug=False)
```

This is the. Python cod of how when the user log into the registration page or the log in . for registration if details are right they will be stored and the user will be able to go to the next page. If the registration is wrong they will be relocated to the registration fail page. Same with the log in python code.

Firewall:



This is the code made from python that takes the users inputs of the URL they want to scan for the firewall scan.



The firewall.py page where the APIs subprocess uses STDONT to request the firewall information form the URL. The code here also prints out one of the firewall found if detected.

Location scanner:



The lookup.py page where the APIs IPWhois and sockets requests the information form the URL.



PYTHON to gets the results back form the lookup.py page where the API got the results. This code also helps to display the results of the scan

Fuzzer scanner:



This is java script code for the scanner to get the information when the fuzzer scanner is selected. This code also show up on the HTML webpage showing the loading



This is the python code that gets the results and displays them coinciding with java script code.



This creates random key for the fuzzer scan giving each scanner its own unique map .



Connect to the list.txt to get what different files are to be searched on the URL using the requests input

2.4. Graphical User Interface (GUI)



1. Index page: this is the first page the users see when opening up the site at first. The page gives a brief overview of what my site is about. On this page the user can press the log in button to gain further access.

CyberSafe Start Page			
imail:	Please log	jin	
Password:			
ont have account? Register: Here			

2A. Log in page: this is where the user can log in with a previously made account.



2B. correct log in : details entered on log in were correct



3C. wrong log in: details entered on log in were wrong



3A. register



3B: correct registering



3C: wrong registering



4: Main page: this is the main page of my application. The is what the user sees after they have log in successfully. Here that can access the about page, view the scanners in a drop down menu and log out.



2A. LOCATION SCANNER: This page this where the user can submit a URL link to be scanned for location and set up information about the URL



2b. fuzzer SCANNER: This page this where the user can submit a URL link to be scanned for file information about the URL



2b. Scanner the results of the fuzzer url

	CyberSafe Back to homepage
1.	Location Scan Results
	Ecoution oculi Roounto
cidr	
name	AT-88-7
handle	NET-52-192-0-0-1
range	52.192.0.0 - 52.223.191.255
description	Amazon Technologies Inc.
country	US
state	WA
city	Seattle
address	410 Terry Ave N.
postal_code	98109
emails	['aws-routing-poc@amazon.com', 'amzn-noc-contact@amazon.com', 'abuse@amazonaws.com', 'aws-rpki-routing-poc@amazon.com']
created	2015-09-02
updated	2020-09-24

3A.Results page (location) : This page will give out the results of the location scan that was conducted of the URL.

3B.Results page (fuzzer) : This page will give out the results of the fuzzer scan that was conducted of the URL.

CyberSafe Back to home page			
		De la Cartera de	
F	irewall detec	ctor	
https://www.rte	9.le		
	Detect Firewall ('[-] Firewall detected : ', 'cloudflare', 'WAF')		
1111111111			

2c. Firewall SCANNER: This page this where the user can submit a url link to be scanned for firewall information about the URL, this page will also display the firewall that is linked to the URL

2.5. Testing

Test Plan

This test plan shows what was tested in my application. This incules the testing of the log in and registering where users can make accounts and are stores in an SQLite database and also test out the three different scanners.

What will be tested/Scope:

- Overall performance of the webpage
- Signup/ sign in and if it saves to database
- Scanners (fuzzer, firewall and location scanners)

Test design techniques.

The overall performance of the webpage

• Black box testing - decision table testing technique

Signup/ sign in

- Black box testing equivalence partition, boundary analysis,
- White box testing code coverage, branch coverage

Scanners

- Black box testing equivalence partition, boundary analysis, decision table testing technique
 - White box testing code coverage, branch coverage

Test cases designed using black-box testing techniques

The overall performance of the website

Decision table testing technique

User signs up to site correctly	Т	Т	Т	F
User is allowed access	Т	Т	Т	F
User picks scanner from filter	Т	Т	F	F
User submits valid URL to scan	Т	F	F	F
Scan results are shown	Y	N	N	N

Test case 1:

Inputs: User signs in/up = T, User is allowed access = T, User picks scanner from filter = T, User submits valid URL to scan = T Scan results are shown = YES

Expected outcome: User gets their scan results = Y

Test case 2:

Inputs: User signs in/up = T, User is allowed access = T, User picks scanner from filter = T, User submits valid URL to scan = F Scan results are shown = NO

Expected outcome: User gets their scan results = N

Test case 3:

Inputs: User signs in/up = T, User is allowed access = T, User picks scanner from filter = F User submits valid URL to scan = F Scan results are shown = NO

Expected outcome: User gets their scan results = N

Test case 4:

Inputs: User signs in/up = F, User is allowed access = F, User picks scanner from filter = T, User submits valid URL to scan = F Scan results are shown = NO

Expected outcome: User gets their scan results = NO

Signing up to website

Equivalence partition

Condition	Valid information	invalid information
Email	 Email must new and not used already. 	 Email already used Email typed in is wrong
	 Email typed in is right. 	
Password		No password typed in

	Password is valid	
Username	Username is vaild	Username is vaild

Boundary analysis

Condition	Invalid	Valid	Invalid
	Equivalence	Equivalence	Equivalence
	Class	Class	Class
Username	///////////////////////////////////////	User types in	user leaves
		right	field blank
		username	
Email	User types in	User types in	user leaves
	wrong Email	right Email	field blank
Password	//////////	User types in right password	user leaves field blank

Signing in into website

Equivalence partition

Condition	Valid information	invalid information
Email	 Email must be premade in sign up page and stored in SQLite database 	 Username never made
		 Email typed in is wrong
	 Email typed in is right 	
Password	 Password must be premade in sign up page and stored in SQLite database 	 Password never made
		 Password typed in is wrong

 Password is right 	

Boundary analysis

Condition	Invalid	Valid	Invalid
	Equivalence	Equivalence	Equivalence
	Class	Class	Class
Email	User types in	User types in	user leaves
	wrong Email	right Email	field blank
Password	User types in wrong password	User types in right password	user leaves field blank

Test cases designed using white-box testing techniques

Sign in page

Code **Coverage** / branch Test cases: Sign in 1 log in = blank Email and password 2 log in =Invalid password 3. log in = Invalid Email 4 log in = Enter correct username and password

Test cases executed (IDs)	Number of executed branches	Total number of branches	Branch coverage
1	1	4	25%
1,2	2	4	50%
1,2,3	3	4	75%
1 ,2 ,3 ,4	4	4	100%

Sign up page Code **Coverage /** branch Test cases: Sign up 1.Sign up = blank Email and password and user name

2.Sign up =Invalid Email correct username and password

3.Sign up = Invalid Username correct email and password

4. Sign up = invalid Password correct Username and email

5.Sign up =Invalid Email invalid username and correct password

6.Sign up = Invalid Username invalid password and correct email

7. Sign up = Invalid Password invalid Email and correct Username

8 sign up = Enter correct username and password

Test cases executed (IDs)	Number of executed branches	Total number of branches	Branch coverage
1	1	8	12.5%
1 and 2	2	8	25%
1 and 2 and 3	3	8	37.5%
1 and 2 and 3 and 4	4	8	50%
1 and 2 and 3 and 4 and 5	5	8	62.5%
1 and 2 and 3 and 4 and 5 and 6	6	8	75%
1 and 2 and 3 and 4 and 5 and 6 and 7	7	8	87.5%
1 and 2 and 3 and 4 and 5 and 6 and 7 and 8	8	8	100%

Scanners (applys to each scanner)

Decision table testing technique.

User picks	Т	Т	F
scanner			
User submits valid URL to scan	Т	F	F
Scan takes place	Т	F	F
Scan results are shown	Y	N	N

Equivalence partition

Condition	Valid information	invalid information
Picking a scanner	 Fuzzzer Firewall Location 	No scanner is picked
url/ submitting and results	URL is validResults displayed	Invalid URLNo results

Boundary analysis

Condition	Invalid	Valid	Invalid
	Equivalence	Equivalence	Equivalence
	Class	Class	Class
Picking a	No scanner	User picks	user logs out
scanner	picked	scanner	
Submitting	No url	User types in	User types in
URL	submitted	valid URL	wrong URL/
			invalid URL

Test cases designed using white-box testing techniques

Scanner(applies to all 3 scanners)

Code Coverage / branch

Test cases: Scanner

- 1 Scanner = no scanner picked
- 2.Scanner = pick scanner , invalid url , no results
- 3 Scanner = pick scanner , valid url , results

Test cases executed	Number of executed	Total number of	Branch coverage
(IDs)	branches	branches	
1	1	3	33.33%

1,2	2	3	66.66%
1,2,3	3	3	100%

Testing my code with **CodeQL in github**:

When I submitted my code to github I ran a CodeQL scan on to test for security vulnerabilities it and here are the results .



Evaluation

es Reset Filters	Records: 25		Search 25 records
	name 🔤 - 🏹	email 🔤 - 🏳	password
	Search column	Search column	Search column
1	4	4@email.com	1 Bytes 🖵
2	р	p@email.com	1 Bytes 🖵
3	r	r@email.com	1 Bytes 🖵
4	t	t@gmail.com	1 Bytes 🖵
5	rt	rt@email.com	2 Bytes 🖵
6	а	a@email.com	1 Bytes 🖵
7	е	em@email.cok	1 Bytes 🖵
8	new	new@email.com	3 Bytes 🖵
9	pic	pic@email.com	3 Bytes 🖵
10	8	8@email.com	1 Bytes 🖵
11	q	qq@email.com	1 Bytes 🖵
12	r	rr@email.com	2 Bytes 🖵
13	s	ss@gmail.com	2 Bytes 🖵
14	f	fff@email.com	1 Bytes 🖵
≪ ⊲ 1	▷ 🖾 Page 1 /	1	Try SQLite Viewer Web A

Database test done by myself with password covered for protection.

Testing with rte.ie for firewall

Firewall SCANNER:



Testing with https://www.netflix.com/ie/ for fuzzer

Fuzzer scanner:

CyberSafe Back to homepage	
a desta	A DE TRANSPORT
Fuzzer Resu	Its
and the second	
File Name	Status
	ОК
/img/	OK
/image/	OK
/images/	OK
/file/	OK
/files/	OK
/upload/	OK
/uploads/	OK
/download/	OK
/downloads/	OK
/lib/	OK
/sys/	UK

Testing with ncirl.ie for location

location SCANNER:



3.0 Conclusions

Advantages: Got my 3 scanners working All aspects of the webpage work Database works for log in

Disadvantages

Databases does not hold the websites

4.0 Further Development or Research

I would like to add more features to the web application including letting users save there scans into there own profile and storing them into the database. I would also like to add my scanners into my web application including a functional nmap port scanner.

5.0 References

Whois scanner : <u>https://dev.to/grahammorby/lets-build-a-who-is-service-with-flask-5ab6</u>

Fade in image : <u>https://blog.hubspot.com/website/css-fade-in</u>

<u>SQLite database : https://docs.python.org/3/library/sqlite3.html</u>

https://www.sqlite.org/src/doc/trunk/ext/userauth/user-auth.txt

https://owlbuddy.com/login-and-signup-using-sqlite/

https://medium.com/analytics-vidhya/how-to-use-flask-login-with-sqlite3-9891b3248324

REQUEST FOR FUZZY : https://www.w3schools.com/xml/xml http.asp

6.0 Appendices

This section should contain information that is supplementary to the main body of the report.

6.1. Project Proposal

Contents

1.0	Objectives	. 39
2.0	Background	. 39
3.0	State of the Art	. 39
4.0	Technical Approach	. 40
5.0	Technical Details	. 41
6.0	Special Resources Required	.41
7.0	Project Plan	41
8.0	Testing	43

7.0 Objectives

(Max half Page)

What does this project set out to achieve?

This project will be a website that can inspect the security characteristics of any website. This website will help look for security characteristics of a website in terms of code, headers set, certificates, date of creation, reviews to determine a kind of score for how reliable or trustable the site is. Also plan to show if there was a recent data breach on website. This is to help users become aware of legitimate website and which ones are safe. All a user would have to do is copy the URL of their desired web page into my site and my website will scan the link a show the information stated above about the page.

8.0 Background

(Max half Page)

Why did you choose to undertake this project? How will you meet the objectives set out in Section 1.0?

I choose to undertake this project because I think it would be a useful website that can cater to anyone that uses the internet. I hope I can create this to help online users to become aware of what websites are safe and be able to trust these websites along with showing them what websites are not so safe to use.

9.0 State of the Art

(Max half page)

What similar applications exist already? What makes your project stand out? How does it differ from similar work of others?

Site Lock is a website similar to what I want to create. But the difference with mine is that it specialises in looking at the security characteristics of a website as well as any recent cyber breaches. My website will just give straightforward answers on any the security characteristics along with any recent cyber-attacks.

10.0 Technical Approach

<mark>(Max 1 page)</mark>

What approach will you take to development? How will you identify requirements? How will you break down requirements into project tasks, activities and milestones?

Requirements

- Find out what different code methods I need to use and which ones are the best
- To create my desired webpage in the way I want it to work, fast functionality and a website that is easy to navigate.
- To have my webpage be able to do its main functions to show the desired information on a webpage.

Project tasks

Pre

- Figure out what the exact information I want to collect in my project
- Decide what coding methods that I want to do
- Figure out if I need to develop this project with any other software.
- Create wireframes, UML diagrams to help with my process of creating my project

During

- Starting and completing with the code for the main functions of my webpage
- Once I have my desired functions for my website, create my css
- Test my project throughout this process. Create fake web URL with different information to test out my website and to demonstrate its functionality

Finish

• Finalising the look and function of my website so it can work at its very best

Milestones and activities

- Have a test website with minimum functionality by Christmas
- Have the bones of the project done by mid-March with the majority of the code done
- Hoping to have my project up and running by end of April
- Keep up with my monthly reports and contact with my supervisor

11.0 Technical Details

<mark>(Max 1 page)</mark>

Implementation language and principal libraries. What are the important algorithms or approaches under consideration for this work?

Languages: java, html, css, java script HTTP TCP, API'S

- 1. Visual Studio Code, Notepad++ used to make html, css, and javascript files.
- 2. GitHub used to store those different parts of the project.

Java, html, Java script : using these languages to crate the basics of my project along the with java script for functionality.

CSS: Using CSS to create the look of my website.

HTTP, TCP and API'S: using these to help connect with the URL's users submit in.

The overview of the website is that user submits the URL in a search bar on the first page of my website, then they will be transferred to the second webpage where they will get the information they are looking for. I want to add a filter to the first page so users can choose what certain information they want, if it's security information or cyber attacked information.

12.0 Special Resources Required

<mark>(Max half page)</mark>

What special resources if any will be required for this work?

N/A

13.0 Project Plan

(Max 2 pages)

Project plan with details on implementation steps and timelines. This project plan should provide as much detail as possible for now and will be revised with more detail with the mid point documentation.

November 2023:

- Create my wire frames and flow charts
- Decide what coding platforms I need and what additional software I made need
- Start off the test model for the website and have it running

December 2023

- Have my test model up and running highlighting the key functions I want my website to have
- Finalising what I want the css and look of the site to be
- Finalising the additional software I need to create my project and starting to see how I can imitate it for my website
- Having my midpoint project ready along with video and documentation

January 2024:

- Starting the process of coding the real application by expanding on my model
- Start to work on any additional software that may be needed in my project

February 2024:

- Starting on polishing up on the css
- Continuing to work on the code for the project
- Doing system test thought-out working on the website

March 2024:

• Have the bones of the project done by mid-March with the majority of the code done

April 2024:

- Hoping to have my project up and running by end of April
- Fixing any issues or bugs with my code
- Complete all my testing on the project

May 2024:

• Final project finish with no bugs and runs smoothly with no errors

14.0 Testing

<mark>(Max 1 page)</mark>

Describe how you will evaluate the system with real technical data using system tests, 3 etc. If applicable describe how you will evaluate the system with an **end user. (be careful here re Ethics etc)**

Unit testing: use to for the first page to search with information the user wants

Decision table testing: used to test the overall performance of the website.

What needs to be tested

- I need to test overall performance of the website.
 - To make sure the filter works on the first page
 - To make sure to right information is being presented on the second page
- To test that the link up to the webpages runs smoothly
- To test multiple URLs to see if I'm getting different results.

6.2. Reflective Journals

Month: October 2023

What?

Reflect on what has happened in your project this month?

Did my project pitch with got accepted and did my Project Proposal along with it

So What?

Consider what that meant for your project progress. What were your successes? What challenges still remain?

Successes: getting accepted on my project pitch and submitting my project proposal

Whets next: getting in contact with my supervisor and discussing the next steps of my project.

Now What?		
What can you do to address outstanding challenges?		
Plan for next month		
November 2023:		
 Create my wire frames and flow charts Decide what coding platforms I need and what additional software I made need Start off the test model for the website and have it running 		
Student Signature	Sinend Doherty	

Month: November

What?

Reflect on what has happened in your project this month?

This month I started my weekly meetings with my supervisor. Each week we discuss my project and how to shape it. I have be researching different ideas like mine and coming up with ideas on how to my project unique.

So What?

Consider what that meant for your project progress. What were your successes? What challenges still remain?

Successes:

Starting on the report for the midpoint presentation

Finalizing with technologies I need in order to work on my project.

Attending every meeting with my supervisor and taking his advice

Challenges:

This Month I need to start my work on the code for my mid-point and to add some functionality to it.

Finish my midpoint report and do the video

Now What?

What can you do to address outstanding challenges?

Challenges:

This Month I need to start my work on the code for my mid-point a to add some functionality to it.

Starting on the code asap and getting some of the main functionality's working.

Finish my midpoint report and do the video

- Finishing the report I have started on.
- Make the mid-point video showing of what I have in my project already.

Student Signature	Sincend Doherty

Month: December

What?

Reflect on what has happened in your project this month?

This month I started my weekly meetings with my supervisor. Each week we discuss my project and how to shape it. I have be researching different ideas like mine and coming up with ideas on how to my project unique.

So What?

Consider what that meant for your project progress. What were your successes? What challenges still remain?

Successes:

Starting on the report for the midpoint presentation

Finalizing with technologies I need in order to work on my project.

Attending every meeting with my supervisor and taking his advice

Challenges:

This Month I need to start my work on the code for my mid-point a to add some functionality to it.

Finish my midpoint report and do the video

Now What?

What can you do to address outstanding challenges?

Challenges:

This Month I need to start my work on the code for my mid-point a to add some functionality to it.

Starting on the code asap and getting some of the main functionality's working.

Finish my midpoint report and do the video

Finishing the report I have started on.	
Make the mid-point video showing of what I have in my project already.	
Student Signature	Sirrend Doherty

Month: January 2024

What?

Reflect on what has happened in your project this month?

I continued to do more research on what I want to add to my project.

So What?

Consider what that meant for your project progress. What were your successes? What challenges still remain?

Successes:

Continuing my research for my project and starting the code on my real project

Whets next:

Now that the midpoint is over I can start working on the actual main application and getting started on the main functionality on the main project.

Now What?	
What can you do to address outstanding challenges?	
Finalizing what additional software needed for my project.	
Plan for February	
 Starting on plan out what the final loc Continuing to work on the code for the functionality's Start doing basic system tests of the vertex of	k of my completed site with the CSS. e project by implement some of the main website
Student Signature	Sincend Doherty

Month: February 2024

What?

Reflect on what has happened in your project this month?

I continued to do more research and started working on the main code for the project. I focused on starting with one of the scans for my project which would be a port scan. My supervisor has been great with showing me different ways I can add this scan into my project. I also looked at bootstrap for the use of making my website look more pleasing to the eye

So What?

Consider what that meant for your project progress. What were your successes? What challenges still remain?

Successes:

Getting started on one of the scans for my website and the use of bootstrap

Whets next:	
Continue working on my project and once I get one sca	an working focus a bit more on the back end.
Now What?	
What can you do to address outstanding challenges?	
Finishing on getting the first scan done with would be a	a port scan
Plan for March	
 Continuing to work on the code for the and working on getting some of the ot the scan on my website.) Continuing to work on my report and the scan on my make and the scan on my make and the scan on my report and the scan on my make and the scan on the	e project and working on my port scanner ther main functions finish (showing results of try to expand on the parts where needed
Student Signature	Sinceral Doherty

Month: March 2024

What?

Reflect on what has happened in your project this month?

Continued to work on both the code and the report of the project. Got a port scanner working but need to add a couple more parts to it

So What?

Consider what that meant for your project progress. What were your successes? What challenges still remain?

Still to do

Finish the code and report and trying to do the testing part of the project.

Now What?

Continue with the code and try and get that finish

What can you do to address outstanding challenges?

- Hoping to have my project up and running by end of April
- Fixing any issues or bugs with my code
- Complete all my testing on the project

Student Signature	Sincerd Doherty

Month: April 2024

What?

Reflect on what has happened in your project this month?

Like last month I continued to work on both the code and the report of the project. Got both an Geo location and fuzzer scanner working along with adding bootstrap. With the report I have a good chunk of it finish and have all sections nearly done.

So What?

Consider what that meant for your project progress.

What were your successes?

Geo location scanner and the fuzzer scanner.

Most of the report is nearly finished

What challenges still remain?

Still to do

Finish the code and report and trying to do the testing part of the project along with the poster and the demo video.

Try to integrate another scanner into my webpage and clear my code up a bit .

Now What?

Continue with the code and try and get that finish

What can you do to address outstanding challenges?

- Having the testing and demo video done
- Fixing any issues or bugs with my code

Strand Doherty

6.3. Other materials used

Any other reference material used in the project for example evaluation surveys etc.