**PENTESTER TOOLKIT**

**APP**

**Research Report I**

****

**Supervisor**

Ms. Sidra Shahid

**Submitted by**

M Ahtesham Sarwar

[2019-ag-6068]

Khawar Azeem

[2019-ag-6067]

**BS Computer Science**

**Department of Computer Science**

**University of Agriculture, Faisalabad.**

Table of Contents

[**1.** **INTRODUCTION** 1](#_Toc534026760)

[**1.1 Background** 1](#_Toc534026761)

[**1.2 Description** 1](#_Toc534026762)

[**1.3 Scope** 1](#_Toc534026763)

[**1.4 Objectives** 2](#_Toc534026764)

[**2.** **REQUIREMENTS** 2](#_Toc534026765)

[**2.1 Functional Requirements** 2](#_Toc534026766)

[**2.2 Non- Functional Requirements** 2](#_Toc534026769)

[**2.3 Hardware Requirements** 3](#_Toc534026770)

[**2.4 Software Requirements** 3](#_Toc534026771)

[**3.** **METHODOLOGY** 3](#_Toc534026772)

[**3.1 Tools & Technologies** 4](#_Toc534026774)

[**4. TIMELINE** 4](#_Toc534026775)

# **INTRODUCTION**

## 1.1 **Background:**

The **PENTESTER TOOLKIT** app is being developed in response to the need for a convenient, mobile solution for **Cyber Security Engineers** to access important tools for Web Application and Network Penetration Testing. Currently, there are various toolkits available for desktop devices, but there is a lack of similar solutions for mobile devices. The **PENTESTER TOOLKIT** app aims to fill this gap and provide mobility and convenience to Cyber Security Engineers by bringing these important tools to their mobile devices.

## **Description:**

The **PENTESTER TOOLKIT** app will be a mobile application designed to help Cyber Security Engineers in the field of Penetration Testing. The app aims to provide a single, convenient location for accessing a range of important tools for web application and network penetration testing, including:

* Http Monitor
* Site Mapper
* Intruder
* Repeater
* Ping Utility
* Port Scan
* Encoder
* Hasher
* Google Dorking
* Http Server
* Privacy
* Notes

The main goal of the app is to provide mobility and ease to Cyber Security Engineers, allowing them to access these tools from their mobile devices rather than relying on desktop computers. The app will be developed using Java and Android Studio, with a focus on user-friendliness and ease of use.

One of the key challenges in developing the **PENTESTER TOOLKIT** app will be balancing the need for a wide range of features and tools with the need to keep the app lightweight and easy to use. The team will address this challenge by carefully prioritizing and selecting the tools to be included in the app, and by implementing an **RAD (Rapid Application Development)** development process that allows for rapid prototyping and iterative improvements based on user feedback.

The app will be of interest to a wide range of people and organizations, including Cyber Security Engineers, IT professionals, and businesses with a need for web application and network penetration testing. The app will be released on the Google Play Store and will be available for download by anyone with an Android device running Android 6.0 or higher.

## 1.3 **Scope:**

The scope of the **PENTESTER TOOLKIT** app project includes the design, development, and testing of a mobile application for Android devices that provides a range of tools for web application and network penetration testing. The app will include the following features:

* **Http Monitor:**

Allows users to monitor Http requests and responses in real-time, including the ability to view request and response headers, bodies, and parameters.

* **Site Mapper:**

Allows users to create a map of a website's structure and links, either through automated crawling or user-directed exploration.

* **Intruder:**

Allows users to perform automated testing of web application vulnerabilities by performing various types of brute force attacks like sniper attack, pitch fork, cluster bomb etc.

* **Repeater:**

Allows users to manually send customized Http requests repeatedly to a server.

* **Ping Utility:**

Allows users to test the availability and response time of a server or website by sending a series of ping requests.

* **Port Scan:**

Allows users to scan a network for open ports, useful for identifying potential vulnerabilities.

* **Encoder:**

Allows users to encode and decode data using various algorithms like Base64, Hex, URL and HTML encoding.

* **Hasher:**

Allows users to calculate hash values for data using various hashing algorithms like MD5, SHA-1, and SHA-2 etc.

* **Google Dorking:**

Allows users to search the internet using advanced search operators and keywords, useful for finding hidden or sensitive information.

* **Http Server:**

Allows users to set up a local Http server, useful for testing and development purposes.

* **Privacy:**

Allows users to mask their identity and location while using the app, useful for preserving privacy and avoiding detection using proxies.

* **Notes**

Allows users to save the notes or payloads for various attacks in one place for later use.

The scope of the project does not include the development of any additional tools or features beyond those listed above. The app will be developed and tested on Android devices running Android 6.0 or higher and will be released on the Google Play Store. The app will be developed using an **RAD (Rapid Application Development)** approach, with a focus on rapid prototyping and iterative improvements based on user feedback.

## 1.4 **Objectives**

The goal of the **PENTESTER TOOLKIT** app project is to develop a mobile application for Android devices that provides a range of tools for web application and network penetration testing. The objectives of the project are as follows:

* Develop a user-friendly app interface that is easy to navigate and use.
* Implement a range of tools for web application and network penetration testing, including Http Request & Response Monitor, Site Mapper, Http Requests Intruder, Http Requests Repeater, Ping Utility, Port Scan, Encoder & Hasher, Google Dorking, and Http Server.
* Test the app on a range of Android devices running Android 6.0 or higher to ensure compatibility and performance.
* Gather user feedback and make iterative improvements to the app based on this feedback.
* Publish the app on the Google Play Store for download by users.

To achieve these objectives, the project will involve the design, development, and testing of the app, as well as the gathering and incorporation of user feedback. The project will follow an **RAD (Rapid Application Development)** development approach, with a focus on rapid prototyping and iterative improvements. The team will work collaboratively to design and develop the app, using RAD practices. The app will be tested on a range of Android devices to ensure compatibility and performance, and user feedback will be gathered and incorporated into future versions of the app. Once the app is complete, it will be published on the Google Play Store for download by users.

2. **REQUIREMENTS**

## 2.1 **Functional Requirements**

The functional requirements for the **PENTESTER TOOLKIT** app project are as follows:

**FR01: Http Monitor**

|  |  |
| --- | --- |
| FR01-01 | The app shall allow users to monitor Http requests and responses in real-time. |
| FR01-02 | The app shall display request and response headers, bodies, and parameters. |
| FR01-03 | The app shall allow users to filter and search Http requests and responses by various criteria. |
| FR01-04 | The app shall allow users to copy, save and export Http requests and responses for later analysis. |

**FR02: Site Mapper**

|  |  |
| --- | --- |
| FR02-01 | The app shall allow users to create a map of a website's structure and links. |
| FR02-02 | The app shall allow users to crawl a website automatically to discover links and content. |
| FR02-03 | The app shall allow users to manually explore a website and add links and content to the map. |
| FR02-04 | The app shall display the map in a graphical format, with links and content organized hierarchically. |

**FR03: Intruder**

|  |  |
| --- | --- |
| FR03-01 | The app shall allow users to perform automated testing of web application. |
| FR03-02 | The app shall support various types of brute force attacks like Sniper attack, Pitch Fork attack, Cluster Bomb attack etc. |
| FR03-03 | The app shall allow users to customize and save payloads for use in testing. |
| FR03-04 | The app shall allow users to customize the frequency and number of requests. |
| FR03-05 | The app shall display the results of the requests, including response times and status codes. |

**FR04: Repeater**

|  |  |
| --- | --- |
| FR04-01 | The app shall allow users to manually send customized Http request to a server. |
| FR04-02 | The app shall display the results of the request, including response time and status code. |
| FR04-03 | The app shall allow users to save and export the results of the requests. |
| FR04-04 | The app shall allow users to search for specific queries in the response of request. |

**FR05: Ping Utility**

|  |  |
| --- | --- |
| FR05-01 | The app shall allow users to test the availability and response time of a server or website. |
| FR05-02 | The app shall send a series of ping requests to the target server or website. |
| FR05-03 | The app shall display the results of the ping requests including response time. |
| FR05-04 | The app shall allow users to save and export the results of the ping requests. |

**FR06: Port Scan**

|  |  |
| --- | --- |
| FR06-01 | The app shall allow users to scan a range of ports on a server or device. |
| FR06-02 | The app shall allow users to customize the range of ports to scan and the timeout for responses. |
| FR06-03 | The app shall display the results of the port scan, including open and closed ports. |
| FR06-04 | The app shall allow users to save and export the results of the port scan. |

**FR07: Encoder**

|  |  |
| --- | --- |
| FR07-01 | The app shall allow users to encode and decode data using various encoding algorithms. |
| FR07-02 | The app shall support a range of encoding algorithms, including Base64, Hex, URL and HTML. |
| FR07-03 | The app shall allow users to input and output data in various formats, including text, hexadecimal, and binary. |
| FR07-04 | The app shall allow users to compare the results of different encoding algorithms. |

**FR07: Hasher**

|  |  |
| --- | --- |
| FR07-01 | The app shall allow users to calculate hash using various hashing algorithms. |
| FR07-02 | The app shall support a range of hashing algorithms, including SHA-1, SHA-2, and MD5. |
| FR07-03 | The app shall allow users to input and output data in various formats, including text, hexadecimal, and binary. |
| FR07-04 | The app shall allow users to compare the results of different hashing algorithms. |

**FR08: Google Dorking**

|  |  |
| --- | --- |
| FR08-01 | The app shall allow users to search Google using advanced search operators. |
| FR08-02 | The app shall provide a list of common Google search operators and their usage. |
| FR08-03 | The app shall allow users to customize and save search queries for reuse. |
| FR08-04 | The app shall display the results of Google searches, including links and snippets of text. |

**FR09: Http Server**

|  |  |
| --- | --- |
| FR09-01 | The app shall allow users to create a local Http server on their device. |
| FR09-02 | The app shall allow users to browse and edit the files and directories on the local server. |
| FR09-03 | The app shall allow users to access the local server using a web browser. |
| FR09-04 | The app shall allow users to set permissions for accessing and modifying the local server. |

**FR10: Privacy**

|  |  |
| --- | --- |
| FR10-01 | The app shall allow users to route their Http traffic through the Tor network or a proxy server. |
| FR10-02 | The app shall show users a list of available Proxies / Tor relays. |

**FR11: Notes**

|  |  |
| --- | --- |
| FR11-01 | The app shall allow users to create, edit, and delete payloads for use in testing web application vulnerabilities. |
| FR11-02 | The app shall allow users to categorize payloads and add notes for each payload. |
| FR11-03 | The app shall allow users to import and export payloads from and to external files. |
| FR11-04 | The app shall allow users to search and filter payloads by various criteria. |

## 2.2 **Non- Functional Requirements**

Some non-functional requirements for the **PENTESTER TOOLKIT** app:

|  |  |
| --- | --- |
| **NFR01** | The app shall have a response time of less than 3 seconds for all tools and features. |
| **NFR02** | The app shall have a user-friendly interface with clear navigation and intuitive controls. |
| **NFR03** | The app shall support offline usage, with all necessary tools and resources available without an internet connection. |
| **NFR04** | The app shall be compatible with Android 6.0 or higher devices. |
| **NFR05** | The app shall have minimal battery and resource usage. |
| **NFR06** | The app shall have a small footprint, taking up minimal storage space on the user's device. |
| **NFR07** | The app shall be updated regularly with new tools and features. |
| **NFR08** | The app shall be compliant with relevant regulations and industry standards for cybersecurity. |
| **NFR09** | The app shall be easy to maintain, with clear documentation and a simple update process. |
| **NFR10** | The app shall be user-friendly, with intuitive controls and a clean, easy-to-navigate interface. |
| **NFR11** | The app shall be compatible with a variety of devices and screen sizes. |

## 2.3 **Hardware Requirements**

Some hardware requirements for the **PENTESTER TOOLKIT** app:

|  |  |
| --- | --- |
| **HR01** | The app shall require a minimum of 1GB of RAM to run smoothly. |
| **HR02** | The app shall require a minimum of 1GB of storage space to install and run. |
| **HR03** | The app shall support devices with a screen size of at least 480x800 pixels. |
| **HR04** | The app shall support devices with a processor speed of at least 1.2GHz. |
| **HR05** | The app shall support devices with Wi-Fi, if any tool or feature requires it. |

2.4 **Software Requirements**

Some software requirements for the **PENTESTER TOOLKIT** app:

|  |  |
| --- | --- |
| **SR01** | The app shall support Android 6.0 or higher. |
| **SR02** | The app shall support a variety of file formats for certain tools and features, such as TXT and JSON. |

# 3. **METHODOLOGY**

For the **PENTESTER TOOLKIT** app, the development team has chosen to follow the **Rapid Application Development (RAD)** methodology. This methodology is particularly well-suited for the project because of its focus on rapid prototyping and iterative development.

The team will start by identifying the **key features** and **tools** that should be included in the app, and will then break these down into smaller, more manageable chunks that can be developed in parallel. Each of these chunks, or "**mini-projects**," will be time-boxed to ensure that they are delivered on schedule.

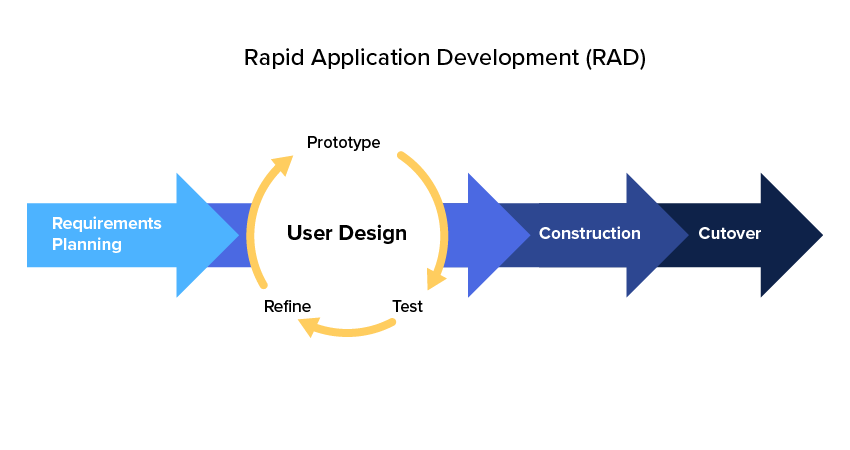
Once these mini-projects are completed, they will be **integrated** into a working prototype that can be tested and refined. This iterative process will continue until the final product is ready for release.

In addition to its focus on rapid prototyping and iterative development, the **RAD** methodology also emphasizes the importance of **feedback** and **collaboration**. The development team will work closely with potential audience to gather feedback and refine the app based on their needs and requirements.

Overall, the **RAD** methodology is a flexible and effective approach that will help the team deliver a high-quality, user-friendly app that meets the needs of cyber security professionals.

Phases of **PENTEST TOOLKIT** RAD (Rapid Application Development) Process**:**

* Requirements Gathering
* User Design
* Construction
* Testing & Integration
* Deployment



## 3.1**Tools & Technologies**

Here are some tools and technologies that may be used in the development of the **PENTESTER TOOLKIT** app:

* **Android Studio:**

This is the primary development environment for Android apps, and will be used to build and test the app.

* **Java:**

This is the primary programming language used for Android app development, and will be used to write the app's code.

* **XML:**

This markup language will be used to define the app's user interface and layout.

* **Git:**

This version control system will be used to manage the codebase and track changes made by the development team.

* **Gradle:**

This build tool will be used to automate tasks such as compiling the code, running tests, and generating the app's final package.

* **Third-party libraries and frameworks:**

The app may make use of various open-source libraries and frameworks to help with tasks such as networking, data storage, and testing.

* **Proxy servers:**

The app may allow users to connect to the internet through a proxy server to enhance their anonymity and protect their privacy.

In addition to these tools and technologies, the development team may also use various resources and documentation to learn about best practices for Android app development and security testing. This may include online tutorials, documentation, and forums.

# **4. Timeline**

Timeline of the **PENTESTER TOOLKIT** app development process:

