

**Gurugram Police Summer Internship 2020**

**Tool Based Project**

**On**

**Virus Scanner**

**Submitted by:**

**Riyanshi Agrawal**

**Sunny**

**Ansh Bhide**

**Sushil Kathait**

**Supported by: Submitted To:**

***Society for Safe Gurgaon***

**SISL Infotech Pvt Ltd Mr. Rakshit Tandon**

**ACKNOWLEDGEMENT**

We express our sincere gratitude to Gurugram police for providing us an opportunity to be part of this Gurugram Police Summer Internship 2020 to complete this Tool based Project on ‘‘**Virus Scanner**’’.

We sincerely thank

1. Sh. K.K. Rao (Commissioner of Police, Gurugram)
2. Ms Nikita Gehlawat (DCP)
3. Ms Pankhuri Kumar (ACP SO1)
4. Mr. Karan Goyal (ACP Cyber Crime)
5. Mr. Rakshit Tandon (Cyber Safety Advisor Cyber Crime, Director Council of Information Security, Consultant-Internet & Mobile Association of India)
6. Ms Akshita Jain (Systems Engineer)

for their guidance and encouragement in carrying out this project work. We also express our gratitude. We also thank to all the experts invited to GPCSSI 2020.

Lastly, we thank almighty, our parents, our siblings and friends and whole Gurugram Police Team for their constant support without which this project would not be possible.

**TABLE OF CONTENTS**

1. Introduction

1.1 Abstract

1.2 Objective

1. System Requirements  
   2.1 Hardware Requirements  
   2.2 Software Requirements
2. Overall Description  
   3.1 Planning/Methodology  
   3.2 Working  
   3.3 Functions
3. Implementation of Project  
   4.1 Module 1: Windows Based  
   4.2 Module 2: CLI Based
4. References

**ABSTRACT**

The “Virus Scanner” will scan the system files and detect the virus and alerts the user whenever necessary. Viruses and other malicious programs are an ever increasing threat to current computer systems. They can cause damage and consume countless hours and system administration time to combat.

The project checks the virus in a file and avoids the systems from any problems that are occurring by virus.

Detection of a virus would be difficult at some points but due to tools and scanners it becomes an easy task. The virus scanner created by us can easily scan files which are infected with viruses and also detect the Md5 algorithm present in the file. Further this scanner has been programmed for both windows based and Linux based machines.

**MAIN OBJECTIVE OF VIRUS SCANNER**

Anti-virus is a program code which is used to capture or notify the malicious code and performs certain functions according to the description written by the programmers. The main objective behind the viral protection programs is to secure the system using these 3 tasks;

1. Scan the files.

2. Detection of the malicious code (if present).

3. Eradication/Deletion of malicious file.

**SYSTEM REQUIREMENTS**

HARDWARE REQUIREMENTS

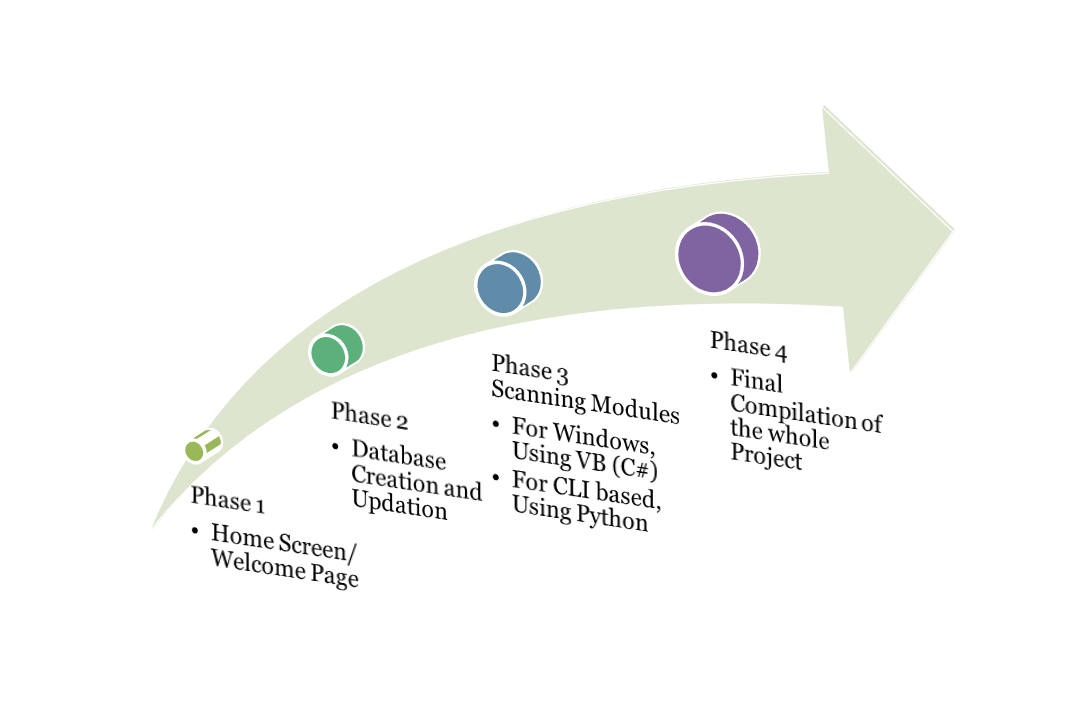
* i3 Processor Based Computer or higher
* Memory: 4 GB RAM or higher
* Hard Drive: 50 GB or higher

SOFTWARE TOOL REQUIREMENTS

* Windows 7 or higher
* Any Linux or Unix OS
* Python3
* C#
* Vi or Vim or any text editor
* Dotnet framework 4.5 or higher
* Visual Studio 2016 or 2019

**VIRUS SCANNER PLANNING & DESIGN**

The following are the different phases in which we have divided our complete project.



PHASE 1:

**HOME SCREEN/WELCOME PAGE:**

The home page of the “Virus Tracking System” contains the options the user can select the file from the system to be scanned. This is the first step of the project.

PHASE 2:

**DATABASE CREATION AND UPDATION:**

This module is the part of the project where the user can update the database of the antivirus so that we can increase the antivirus efficiency and accuracy.

The database is the text file where the admin can add the new Md5 Hash codes of the virus and can also delete the existing code from the virus database.

PHASE 3:

**SCANNING MODULES:**

This module is the core part which corresponds to the scanning part of the antivirus. Here the file to scan is matched with the database which contains the Md5 Hashes of the virus.

The scanning module is of two types:

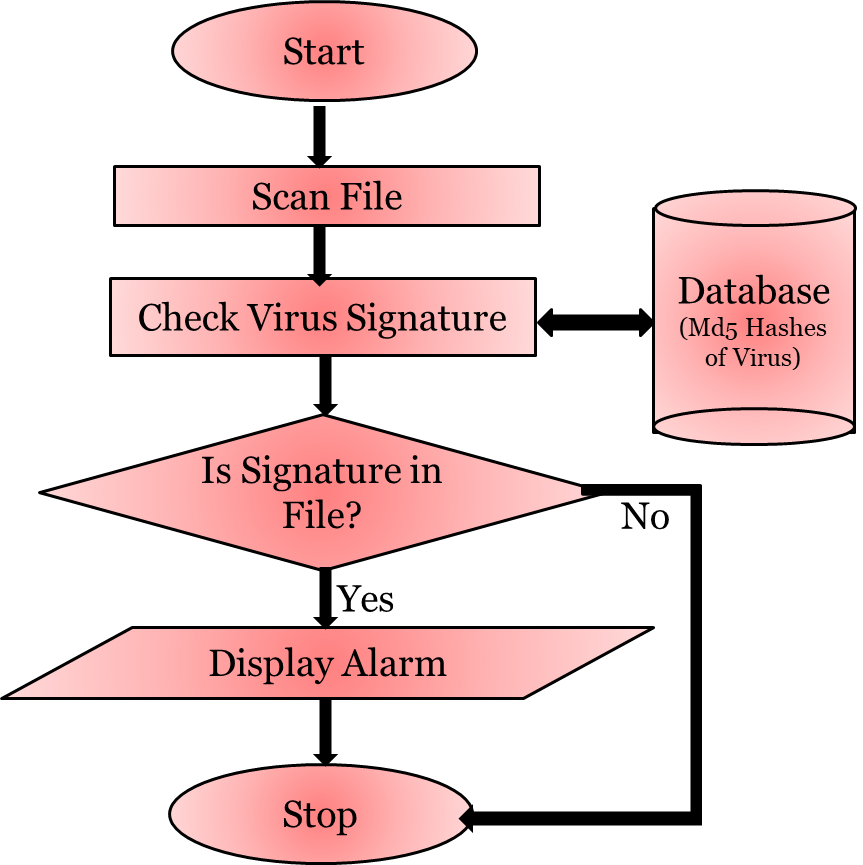
1. For Windows:   
   Using VB(C#)
2. For CLI based:  
   Using Python

PHASE 4:

**FINAL COMPILATION THE PROJECT:**

After the code was completed and checked the proper working of the project, it was assembled and the final compilation of the code was done.

**WORKING OF THE VIRUS SCANNER**



A robust virus scanner is more capable of tracking down Viruses, Worms, Trojans, Spyware and Malware software often circulated by Cyber criminals for their personal gains. So the working of the project is as follows:

1. **Start:**   
   This is the first step where the welcome page is displayed and the user selects the file to be scanned.
2. **Scan File:**The next step is to scan the selected file. When anti-virus scans a file, it will generate the hash of the selected file.
3. **Virus Database (Checkfile):**  
   Anti-virus scanner works in a very traditional way, it uses a database called a virus dictionary which has lots of Md5 hash codes from different known viruses.
4. **Check Virus Signature:**After generating the hash of the selected file, the generated hash is compared with the hashes in the virus database.
5. **Display Alarm:**If the hash of the selected file matches with the hash present in the virus database, then the file is infected and an alert is created to warn the user.
6. **Stop:**  
   This is the final step where the result is displayed whether the file is infected or not and the working is completed.

In simple words, the entire scan and detection process relies on the repository (virus dictionary/signature) of known viruses.

**FUNCTIONS OF THE VIRUS SCANNER**

The Md5 Virus scanner and hash calculator designed by us has some of the following functions :

(The functions can be further developed based on changes and modifications in the source code.)

* **Drag & Drop**

To check one or more files, the files can be selected effortlessly by dragging & dropping them into the program. The calculated MD5 checksums are displayed immediately.

* **Clipboard**

Already calculated MD5 hash values can be copied effortlessly to the clipboard and can then be inserted.

* **Create/Update Checkfiles**

MD5 hash values for several viruses can be added into a text file or saved as a Checkfile in the database. The Checkfile is generated once and can be updated for future checks. Checkfile can be updated immediately.

* **Validate Selected File**

MD5 hash values of the file to be scanned are compared to the latest hash values that are stored as checkfiles in the Database.

* **Create Alert**

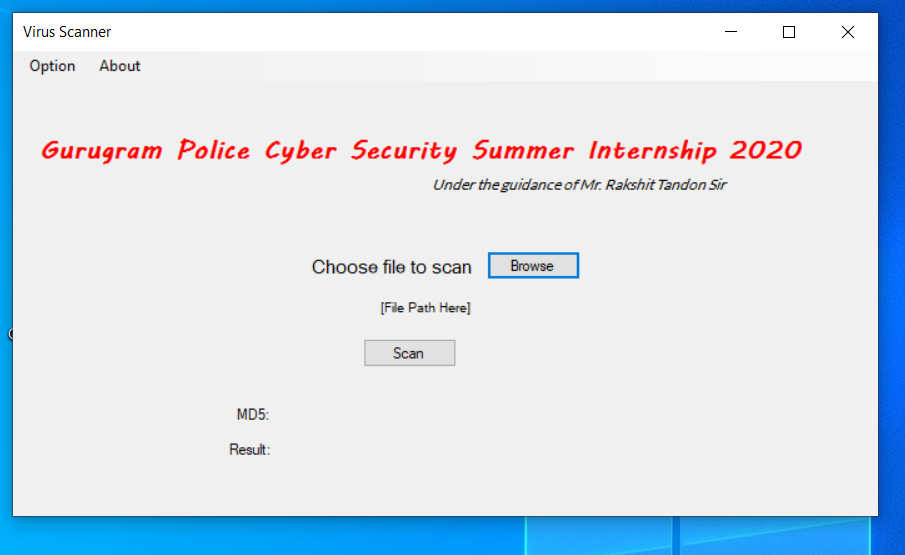
After completion of the checks, an alert will be generated if it is found to be malicious.

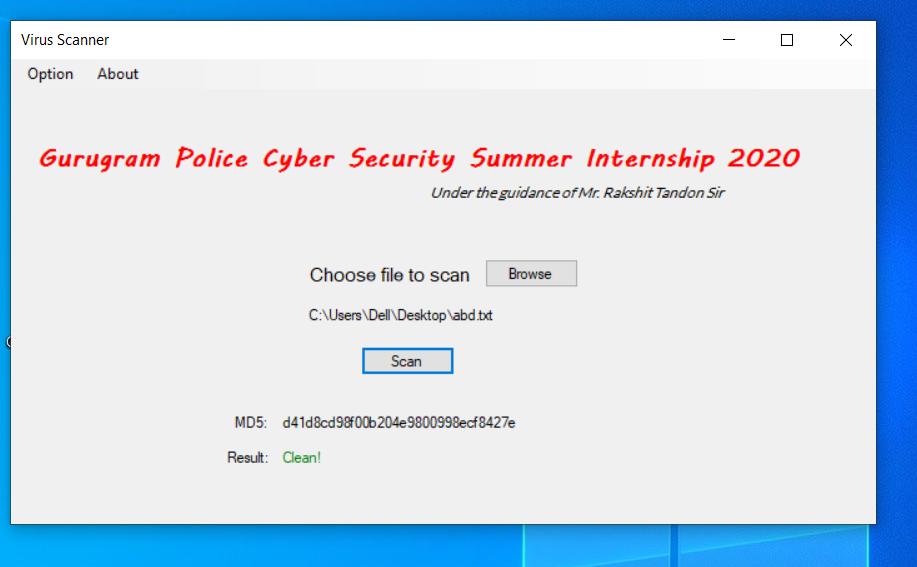
* **Speed Efficient**

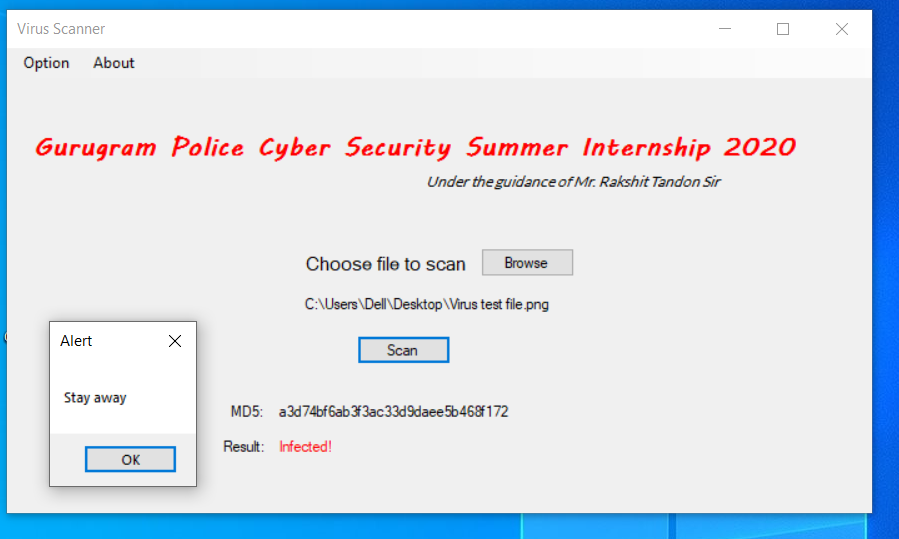
MD5 File Hasher calculates and verifies hash values uniquely fast. The size of files don’t matter. Optionally, the progress bar can be shown during the calculation which results in an additional feature when checking large amounts of data.

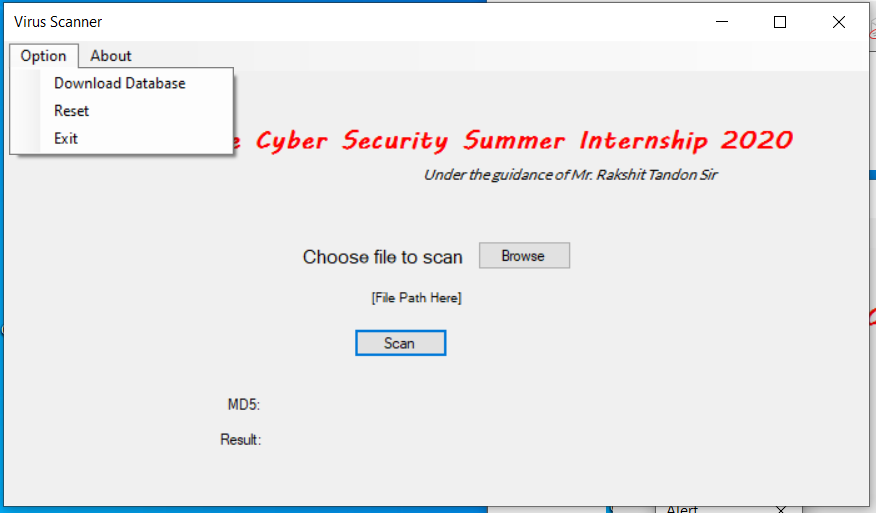
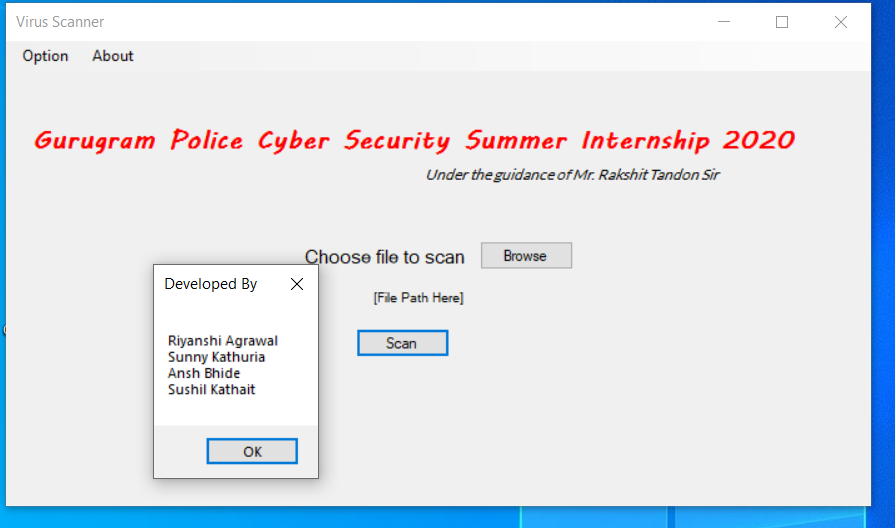
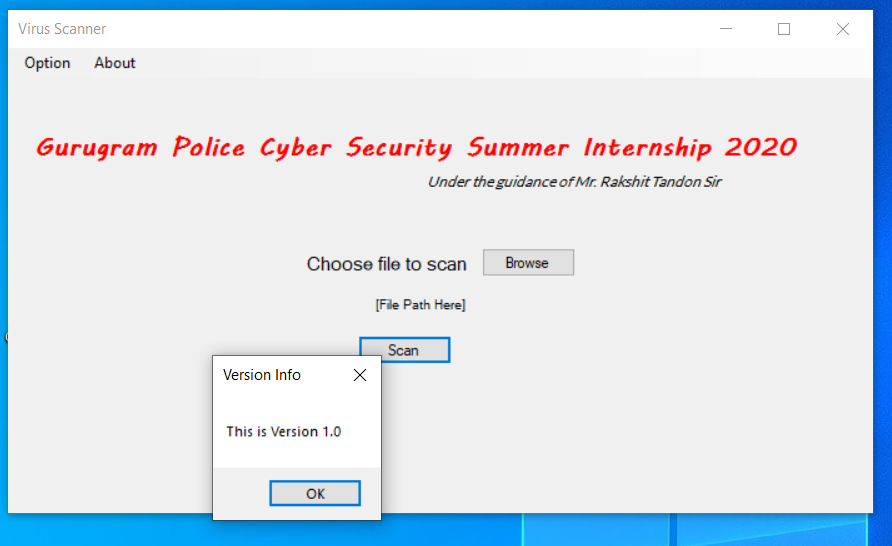
**IMPLEMENTATION**

**Module 1: Windows Based(GUI version)**

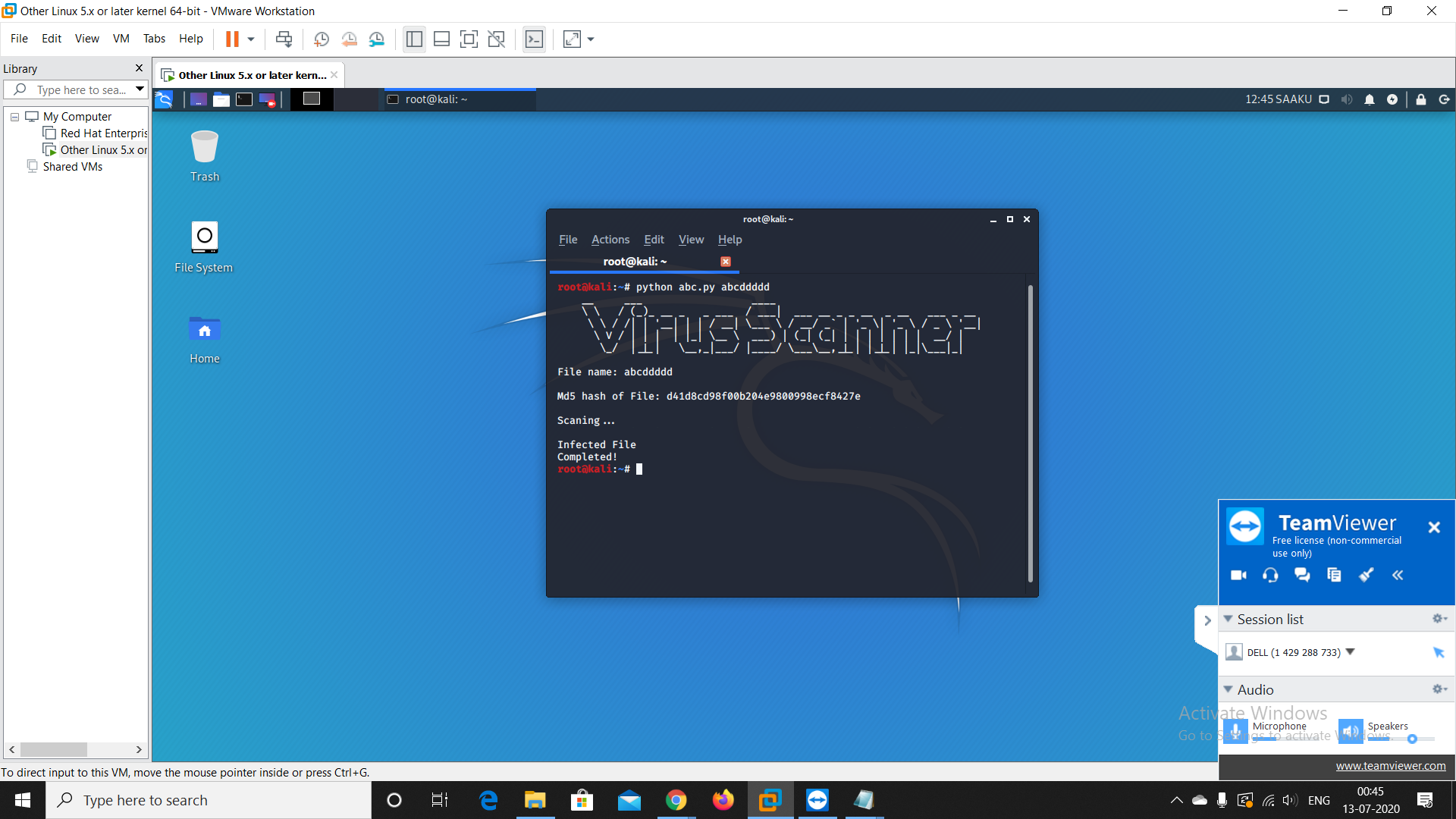
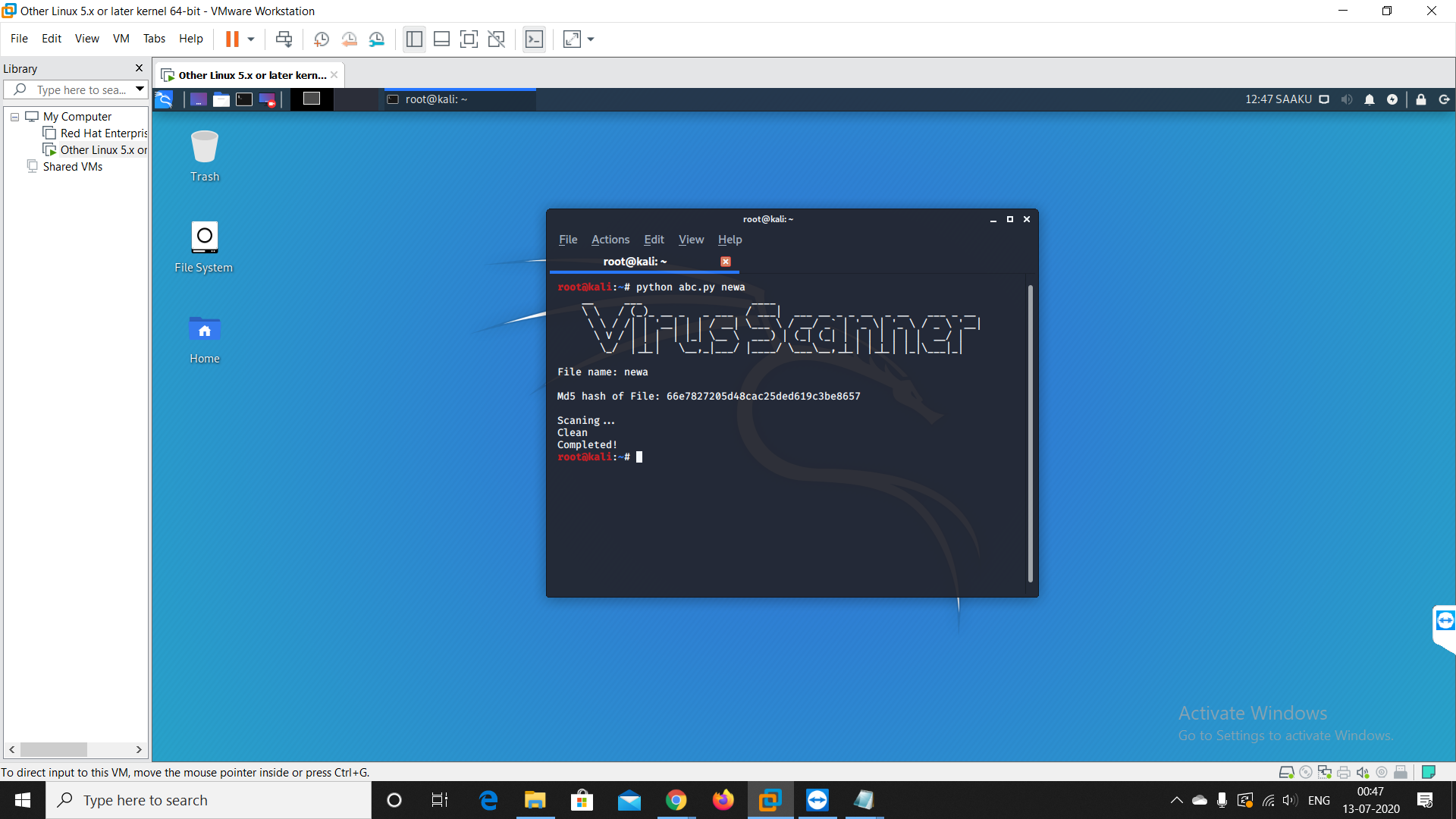
1. Home Page  
   
2. Scanning Module





1. Options Menu  
   
2. About Menu  
     
   

**Module 2: CLI Based**

1. Scanning Midule  
     
   

**CONCLUSION**

Our **Virus Scanner** project’s main objective was to provide the functionality to all users and to make for both popular Operating Systems i.e. Windows and Linux. To make the project more user friendly, we also created a module which is a GUI based module for Windows users. And for Linux users, we also created a CLI based module.

Windows based version is available in Executable(.exe) program and CLI based module is available in a Python(.py) program.

CLI version can be run on both windows and linux users using Command prompt or terminal. For simple usage of CLI based module, we can run it using simple commands.

We can change both modules with the requirement of the user and also to add more functionality to them.

Both Modules are scanning the file on the basis of MD5 signatures, we can also add different types of file signatures to scan the file.

**REFERENCES**

1. <https://support.microsoft.com/en-in/help/307966/how-to-provide-file-drag-and-drop-functionality-in-a-visual-c-applicat>
2. <https://docs.microsoft.com/en-us/dotnet/api/system.security.cryptography.md5>
3. <https://docs.microsoft.com/en-us/visualstudio/ide/create-csharp-winform-visual-studio?view=vs-2019>
4. <https://www.c-sharpcorner.com/csharp-tutorials>
5. <https://www.tutorialspoint.com/python/index.htm>
6. <https://docs.python.org/2/library/md5.html>
7. <https://www.tutorialspoint.com/md5-hash-encoding-using-python>
8. <https://stackoverflow.com/questions/16874598/how-do-i-calculate-the-md5-checksum-of-a-file-in-python/16876405>