**A PROJECT REPORT ON**

**KEYLOGGER**

SUBMITTED TO

MIT SCHOOL OF COMPUTING, LONI, PUNE IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF THE DEGREE

**BACHELOR OF TECHNOLOGY**

**(Computer Science & Engineering)**

**BY**

Hemanshu Moundekar MITU22BTCS0330

Durgesh Borse MITU22BTCS0281

Sourabh Karmakar MITU22BTCS0851

**Under the guidance of**

**Prof. UDAY MANDE**



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**MIT School OF COMPUTING**

**MIT Art, Design and Technology University**

**Rajbaug Campus, Loni-Kalbhor, Pune 412201**

**2023- 2024**

****

**MIT SCHOOL OF COMPUTING**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

MIT ART, DESIGN AND TECHNOLOGY UNIVERSITY,

RAJBAUG CAMPUS, LONI-KALBHOR, PUNE 412201

**CERTIFICATE**

This is to certify that the project report entitled

**“KEYLOGGER”**

Submitted by

Hemanshu Moundekar MITU22BTCS0330

Durgesh Borse MITU22BTCS0281

Sourabh Karmakar MITU22BTCS0851

is a bonafide work carried out by them under the supervision of **Prof. Uday Mande** and it is submitted towards the partial fulfillment of the requirement of MIT ADT university, Pune for the award of the degree of Bachelor of Technology (Computer Science and Engineering)

**Prof. Uday Mande Prof. Dr. Ganesh Pathak**

Guide Head of Department

**Prof. Dr. Rajneeshkaur Sachdeo Prof. Dr. Ramchandra Pujeri**

Director Dean

Seal/Stamp of the College

Place: PUNE

Date:

**DECLARATION**

We, the team members

|  |  |
| --- | --- |
| **Name**  HEMANSHU MOUNDEKAR | **Enrollment No**  (MITU22BTCS0330) |
| DURGESH BORSE | (MITU22BTCS0281) |
| SOURABH KARMAKAR | (MITU22BTCS0851) |

Hereby declare that the project work incorporated in the present project entitled **“Keylogger”** is original work. This work (in part or in full) has not been submitted to any University for the award or a Degree or a Diploma. We have properly acknowledged the material collected from secondary sources wherever required. We solely own the responsibility for the originality of the entire content.

Date: 30/04/2024

Name & Signature of the Team Members

HEMANSHU MOUNDEKAR: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

DURGESH BORSE: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

SOURABH KARMAKAR: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Name and Signature of Guide**

Seal/Stamp of the College

Place: Pune

Date: 30/04/2024



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

MIT SCHOOL OF COMPUTING,

RAJBAUG, LONI KALBHOR,

PUNE – 412201

**EXAMINER’S APPROVAL CERTIFICATE**

The project report entitled “KEYLOGGER” submitted by HEMANSHU MODEKAR (MITUBTCS0330), DURGESH BORSE (MITU22BTCS0281), SOURABH KARMAKAR (MITU22BTCS0851) in partial fulfillment for the award of the degree of Bachelor of Technology (Computer Science & Engineering) during the academic year 2023-24, of MIT-ADT University, MIT School OF COMPUTING, Pune, is hereby approved.

**Examiners:**

**1.**

**2.**

**ACKNOWLEDGEMENT**

I express my profound thanks to my Guide **Prof. Uday Mande** for his expert guidance, encouragement and inspiration during this project work.

I would like to thank **Dr. Reena Pagare**, Project Coordinator, Department Computer Science & Engineering for extending all support during the execution of the project work.

I sincerely thank to **Prof. Dr. Ganesh Pathak**, Head, Department of Computer Science & Engineering, MIT School of Engineering, MIT-ADT University, Pune, for providing necessary facilities in completing the project.

I am grateful to **Prof. Dr. Rajneeshkaur Sachdeo**, Director, MIT School of Engineering, MIT-ADT University, Pune, for providing the facilities to carry out my project work.

I sincerely thank to **Prof. Dr. Ramchandra Pujeri**, Dean, Department of Computer Science & Engineering, MIT School of Engineering, MIT-ADT University, Pune, for providing necessary facilities in completing the project.

We express our humble gratitude to Dr. C. Muthamizhchelvan, Vice Chancellor (I/C), SRM

Institute of Science and Technology, for the facilities extended for the project work and his

continued support.

We extend our sincere thanks to Dr. Revathi Venkataraman, Professor & Chairperson,

School of Computing, SRM Institute of Science and Technology, for her invaluable support.

We wish to thank Dr. M. Lakshmi, Professor & Head, Department of Data Science and

Business Systems, SRM Institute of Science and Technology, for her valuable suggestions and

encouragement throughout the period of the project work.

We are extremely grateful to our Academic Advisor Dr. E. Sasikala, Assistant Professor,

Department of Data Science and Business Systems, SRM Institute of Science and Technology,

for her great support at all the stages of project work.

We register our immeasurable thanks to our Faculty Advisor, Dr. J. Shobana, Assistant

Professor, Department of Data Science and Business Systems, SRM Institute of Science and

Technology, for leading and helping us to complete our course.

Our inexpressible respect and thanks to our guide, Dr. Jeba Sonia, Assistant Professor,

Department of Data Science and Business Systems, SRM Institute of Science and Technology,

for providing me an opportunity to pursue my project under her mentorship. She provided me the

freedom and support to explore the research topics of my interest. Her passion for solving real

problems and making a difference in the world has always been inspiring.

We sincerely thank the staff and students of the Data Science and Business Systems, SRM

Institute of Science and Technology, for their help during my research.

Finally, we would like to thank my parents, our family members and our friends for their

unconditional love, constant support and encouragement.

We express our humble gratitude to Dr. C. Muthamizhchelvan, Vice Chancellor (I/C), SRM

Institute of Science and Technology, for the facilities extended for the project work and his

continued support.

We extend our sincere thanks to Dr. Revathi Venkataraman, Professor & Chairperson,

School of Computing, SRM Institute of Science and Technology, for her invaluable support.

We wish to thank Dr. M. Lakshmi, Professor & Head, Department of Data Science and

Business Systems, SRM Institute of Science and Technology, for her valuable suggestions and

encouragement throughout the period of the project work.

We are extremely grateful to our Academic Advisor Dr. E. Sasikala, Assistant Professor,

Department of Data Science and Business Systems, SRM Institute of Science and Technology,

for her great support at all the stages of project work.

We register our immeasurable thanks to our Faculty Advisor, Dr. J. Shobana, Assistant

Professor, Department of Data Science and Business Systems, SRM Institute of Science and

Technology, for leading and helping us to complete our course.

Our inexpressible respect and thanks to our guide, Dr. Jeba Sonia, Assistant Professor,

Department of Data Science and Business Systems, SRM Institute of Science and Technology,

for providing me an opportunity to pursue my project under her mentorship. She provided me the

freedom and support to explore the research topics of my interest. Her passion for solving real

problems and making a difference in the world has always been inspiring.

We sincerely thank the staff and students of the Data Science and Business Systems, SRM

Institute of Science and Technology, for their help during my research.

Finally, we would like to thank my parents, our family members and our friends for their

unconditional love, constant support and encouragement.

We express our humble gratitude to Dr. C. Muthamizhchelvan, Vice Chancellor (I/C), SRM

Institute of Science and Technology, for the facilities extended for the project work and his

continued support.

We extend our sincere thanks to Dr. Revathi Venkataraman, Professor & Chairperson,

School of Computing, SRM Institute of Science and Technology, for her invaluable support.

We wish to thank Dr. M. Lakshmi, Professor & Head, Department of Data Science and

Business Systems, SRM Institute of Science and Technology, for her valuable suggestions and

encouragement throughout the period of the project work.

We are extremely grateful to our Academic Advisor Dr. E. Sasikala, Assistant Professor,

Department of Data Science and Business Systems, SRM Institute of Science and Technology,

for her great support at all the stages of project work.

We register our immeasurable thanks to our Faculty Advisor, Dr. J. Shobana, Assistant

Professor, Department of Data Science and Business Systems, SRM Institute of Science and

Technology, for leading and helping us to complete our course.

Our inexpressible respect and thanks to our guide, Dr. Jeba Sonia, Assistant Professor,

Department of Data Science and Business Systems, SRM Institute of Science and Technology,

for providing me an opportunity to pursue my project under her mentorship. She provided me the

freedom and support to explore the research topics of my interest. Her passion for solving real

problems and making a difference in the world has always been inspiring.

We sincerely thank the staff and students of the Data Science and Business Systems, SRM

Institute of Science and Technology, for their help during my research.

Finally, we would like to thank my parents, our family members and our friends for their

unconditional love, constant support and encouragement.

We express our humble gratitude to Dr. C. Muthamizhchelvan, Vice Chancellor (I/C), SRM

Institute of Science and Technology, for the facilities extended for the project work and his

continued support.

We extend our sincere thanks to Dr. Revathi Venkataraman, Professor & Chairperson,

School of Computing, SRM Institute of Science and Technology, for her invaluable support.

We wish to thank Dr. M. Lakshmi, Professor & Head, Department of Data Science and

Business Systems, SRM Institute of Science and Technology, for her valuable suggestions and

encouragement throughout the period of the project work.

We are extremely grateful to our Academic Advisor Dr. E. Sasikala, Assistant Professor,

Department of Data Science and Business Systems, SRM Institute of Science and Technology,

for her great support at all the stages of project work.

We register our immeasurable thanks to our Faculty Advisor, Dr. J. Shobana, Assistant

Professor, Department of Data Science and Business Systems, SRM Institute of Science and

Technology, for leading and helping us to complete our course.

Our inexpressible respect and thanks to our guide, Dr. Jeba Sonia, Assistant Professor,

Department of Data Science and Business Systems, SRM Institute of Science and Technology,

for providing me an opportunity to pursue my project under her mentorship. She provided me the

freedom and support to explore the research topics of my interest. Her passion for solving real

problems and making a difference in the world has always been inspiring.

We sincerely thank the staff and students of the Data Science and Business Systems, SRM

Institute of Science and Technology, for their help during my research.

Finally, we would like to thank my parents, our family members and our friends for their

unconditional love, constant support and encourage

**HEMANSHU MOUNDEKAR (MITU22BTCS0330)**

**DURGESH BORSE (MITU22BTCS0281)**

**SOURABH KARMAKAR (MITU22BTCS0851)**

**ABSTRACT**

Keylogging is the action of recording the key stroke on a keyboard, typically in a covert manner. Software Keyloggers are detected based on behavioral characteristics. They don't provide root privileges; detection is based on permission from kernel and prone to many attacks. Software Keyloggers is a software program that can be installed onto a computer, which monitors all the user activities on the computer. Keyloggers steal confidential information and they completely run in stealth mode. When Keyloggers is installed in a computer, it is not shown either in start-up icons or anywhere else on the computer that is being monitored. Software Keyloggers have posed a great threat to user privacy and security. Detection of Keyloggers is difficult because they run in hidden mode. Detection of Software Keyloggers is done using various techniques namely Anti-Hook techniques, HoneyID: Spyware detection, bot detection, safe access to password protected accounts and dendritic cell algorithm. These algorithms are used to detect the existence of Keyloggers in computers, which strengthens user privacy and security

**CONTENTS**

|  |  |
| --- | --- |
| Certificate ………………....………………………….…………. | i |
| Certificate (From Company If Any) …………………...………. | ii |
| Declaration ……………….…………….………………....………. | iii |
| Examiner’s Approval Certificate…….…………………………………. | iv |
| Acknowledgement ……………….………………..……………. | v |
| Abstract …………………………………………..…...……………. | vi |
| List of Figures ………………………..………….…………………. | viii |
| List of Tables ……………………….…………….…………………. | ix |

[Chapter 1 INTRODUCTION 1](#_heading=h.3znysh7)

[1.1.](#_heading=h.2et92p0) Purpose 1

[1.2. Scope of Developing the Project 2](#_heading=h.tyjcwt)

[Chapter 2 PROBLEM IDENTIFICATION 5](#_heading=h.3rdcrjn)

[2.1 Project Function 5](#_heading=h.26in1rg)

[2.2. Operating Environment 7](#_heading=h.lnxbz9)

[2.3. Features 10](#_heading=h.44sinio)

[Chapter 3 Code Implementation and Testing 11](#_heading=h.z337ya)

[5.1 Project scope 11](#_heading=h.3j2qqm3)

[5.2 User Classes & Characteristics Coder 11](#_heading=h.1y810tw)

[Chapter 4 RESULT](#_heading=h.4i7ojhp)

[Chapter 5 SOFTWARE TESTING 13](#_heading=h.2xcytpi)

[Chapter 6 CONCLUSION AND FUTURE WORK 14](#_heading=h.1ci93xb)

[BIBLIOGRAPHY 15](#_heading=h.3whwml4)

[ANNEXURE A: List of Publications and Research Paper (In its Original formats) 16](#_heading=h.2bn6wsx)

[ANNEXURE B: Plagiarism Report 17](#_heading=h.qsh70q)

ANNEXURE C: Empathy Chart and Solution Document

Annexure D: Project Tracker

# CHAPTER 1

### INTRODUCTION:

A keylogger is programming or equipment that catches and tracks what people type on their console. You might have utilized a PC with keystroke logging programming introduced for observing and guaranteeing protected or approved use. Nonetheless, normal clients' impression of keylogging varies essentially from cybercriminals' insights. Vindictive entertainers can utilize them to take your own and monetary data, as well as PIN codes and record numbers, Mastercard numbers, usernames, passwords, and other delicate data, which can all be utilized to direct misrepresentation or fraud.

Keyloggers are often used for various purposes, ranging from legitimate ones like monitoring children's internet activity or employee computer usage to malicious ones such as stealing personal information, login credentials, or financial data for illicit purposes.

There are two main types of keyloggers: software-based and hardware-based. Software keyloggers are programs installed on a computer or device, often disguised as legitimate software, and run in the background without the user's awareness. Hardware keyloggers are physical devices that are inserted between the keyboard and the computer, intercepting keystrokes before they reach the computer's operating system.

While keyloggers can serve legitimate purposes in certain contexts, they also pose significant privacy and security risks when used maliciously. As such, their development, distribution, and use are subject to legal restrictions in many jurisdictions.

1. A keylogger is programming or equipment that catches and tracks what people type on their console. You might
2. have utilized a PC with keystroke logging programming introduced for observing and guaranteeing protected or
3. approved use.
4. Nonetheless, normal clients' impression of keylogging varies essentially from cybercriminals' insights. Vindictive
5. entertainers can utilize them to take your own and monetary data, as well as PIN codes and record numbers,
6. Mastercard numbers, usernames, passwords, and other delicate data, which can all be utilized to direct
7. misrepresentation or fra.

### 1.1 PURPOSE:

1. The individual or association utilizing the keylogger should agree with the accompanying lawful prerequisites:
2. • There is no unlawful utilization of information included.
3. • Be the item's proprietor, producer, or lawful watchman assuming the item is being utilized by a young
4. person.
5. • Use it as per the laws of their separate ward.
6. • This rundown perceptibly needs assent. Clients of keyloggers are not expected to get assent except if
7. neighborhood regulations urge it. The fact that they are being watched makes this is clearly dishonest in
8. circumstances where individuals uninformed.
9. You might acknowledge keystroke logging under clear language regarding administration or an agreement in
10. assented cases. This incorporates clicking "acknowledge" to utilize public Wi-Fi or marking an agreement with a
11. business

The individual or association utilizing the keylogger should agree with the accompanying lawful prerequisites:

* There is no unlawful utilization of information included.
* Be the item's proprietor, producer, or lawful watchman assuming the item is being utilized by a young person.
* Use it as per the laws of their separate ward.
* This rundown perceptibly needs assent. Clients of keyloggers are not expected to get assent except if neighborhood regulations urge it. The fact that they are being watched makes this is clearly dishonest in circumstances where individuals uniformed.

You might acknowledge keystroke logging under clear language regarding administration or an agreement in assented cases. This incorporates clicking "acknowledge" to utilize public Wi-Fi or marking an agreement with a business.

Keyloggers are ordinarily utilized for the accompanying legitimate purposes:

• IT investigating is the most common way of social occasion data about client issues and precisely settling

them.

• Client input is accumulated and merchandise are worked on in PC item improvement.

• Unapproved client conduct on web servers is observed by business servers.

• Representative observing — to guarantee that organization property is utilized securely consistently

• IT investigating is the most common way of social occasion data about client issues and precisely settling

them.

• Client input is accumulated and merchandise are worked on in PC item improvement.

• Unapproved client conduct on web servers is observed by business servers.

• Representative observing — to guarantee that organization property is utilized securely consistently

• IT investigating is the most common way of social occasion data about client issues and precisely settling

them.

• Client input is accumulated and merchandise are worked on in PC item improvement.

• Unapproved client conduct on web servers is observed by business servers.

• Representative observing — to guarantee that organization property is utilized securely consistently

• IT investigating is the most common way of social occasion data about client issues and precisely settling

them.

• Client input is accumulated and merchandise are worked on in PC item improvement.

• Unapproved client conduct on web servers is observed by business servers.

• Representative observing — to guarantee that organization property is utilized securely consistently

• IT investigating is the most common way of social occasion data about client issues and precisely settling

them.

• Client input is accumulated and merchandise are worked on in PC item improvement.

• Unapproved client conduct on web servers is observed by business servers.

• Representative observing — to guarantee that organization property is utilized securely consistently

• IT investigating is the most common way of social occasion data about client issues and precisely settling

them.

• Client input is accumulated and merchandise are worked on in PC item improvement.

• Unapproved client conduct on web servers is observed by business servers.

• Representative observing — to guarantee that organization property is utilized securely consistently

• IT investigating is the most common way of social occasion data about client issues and precisely settling

them.

• Client input is accumulated and merchandise are worked on in PC item improvement.

• Unapproved client conduct on web servers is observed by business servers.

• Representative observing — to guarantee that organization property is utilized securely consistently

• IT investigating is the most common way of social occasion data about client issues and precisely settling

them.

• Client input is accumulated and merchandise are worked on in PC item improvement.

• Unapproved client conduct on web servers is observed by business servers.

• Representative observing — to guarantee that organization property is utilized securely consistently

• IT investigating is the most common way of social occasion data about client issues and precisely settling

them.

• Client input is accumulated and merchandise are worked on in PC item improvement.

• Unapproved client conduct on web servers is observed by business servers.

• Representative observing — to guarantee that organization property is utilized securely consistently

• IT investigating is the most common way of social occasion data about client issues and precisely settling

them.

• Client input is accumulated and merchandise are worked on in PC item improvement.

• Unapproved client conduct on web servers is observed by business servers.

• Representative observing — to guarantee that organization property is utilized securely consistently

• IT investigating is the most common way of social occasion data about client issues and precisely settling

them.

• Client input is accumulated and merchandise are worked on in PC item improvement.

• Unapproved client conduct on web servers is observed by business servers.

• Representative observing — to guarantee that organization property is utilized securely consistently

• IT investigating is the most common way of social occasion data about client issues and precisely settling

them.

• Client input is accumulated and merchandise are worked on in PC item improvement.

• Unapproved client conduct on web servers is observed by business servers.

• Representative observing — to guarantee that organization property is utilized securely consistently

• IT investigating is the most common way of social occasion data about client issues and precisely settling

them.

• Client input is accumulated and merchandise are worked on in PC item improvement.

• Unapproved client conduct on web servers is observed by business servers.

• Representative observing — to guarantee that organization property is utilized securely consistently

• IT investigating is the most common way of social occasion data about client issues and precisely settling

them.

• Client input is accumulated and merchandise are worked on in PC item improvement.

• Unapproved client conduct on web servers is observed by business servers.

• Representative observing — to guarantee that organization property is utilized securely consistently

• IT investigating is the most common way of social occasion data about client issues and precisely settling

them.

• Client input is accumulated and merchandise are worked on in PC item improvement.

• Unapproved client conduct on web servers is observed by business servers.

• Representative observing — to guarantee that organization property is utilized securely consistently

IT investigating is the most common way of social occasion data about client issues and precisely settling

them.

• Client input is accumulated and merchandise are worked on in PC item improvement.

• Unapproved client conduct on web servers is observed by business servers.

• Representative observing — to guarantee that organization property is utilized securely consistently.

* IT investigating is the most common way of social occasion data about client issues and precisely settling them.
* Client input is accumulated and merchandise are worked on in PC item improvement.
* Unapproved client conduct on web servers is observed by business servers.
* Representative observing to guarantee that organization property is utilized securely consistently.

Legitimate keyloggers might be more pervasive in your day-to-day existence than you understand. Assuming the observing party has requested admittance, you as a rule have the ability to control your information. If you would rather not utilize the keyloggers beyond work, you can just deny their approval.

**• Programming Environment:**

1. Java JDK 22

2. Eclipse 2024-03

**• Program Files Used:**

1. JNativeHook.jar

## 1.2 Scope of developing the project

## With other malware, there are usually clear signs of infection. These indications can be as small as excessive notifications and pop-ups on the user’s screen, or as large as full denial of access to the device.

## Keyloggers can be significantly more dangerous since they are usually difficult for the user to detect. Keyloggers can capture screenshots, log personal data, and collect any information the user types in complete silence. This means users are often unaware of the problem and take no action to solve it. For businesses that don’t have large and dedicated cybersecurity teams to constantly monitor device security, such compromises can have an outsized effect over time, even more so if the devices used by employees are unmanaged.

## 1.3 Objectives

The main objective of keyloggers is to interfere in the chain of events that happen

when a key is pressed and when the data is displayed on the monitor as a result of a

keystroke.

## 1.4 Scope

**1.Parental Control:** Keyloggers can be used by parents to monitor their children's online activities and ensure they are not engaging in inappropriate behaviour or interacting with potentially dangerous individuals.

**2.Employee Monitoring:** In corporate settings, keyloggers may be deployed by employers to monitor employee productivity, ensure compliance with company policies, and protect sensitive information from being leaked.

**3.Security Testing:** Ethical hackers and security professionals may use keyloggers as part of penetration testing or security audits to identify vulnerabilities in computer systems and networks.

**4.Law Enforcement:** Keyloggers can be used by law enforcement agencies as part of criminal investigations to gather evidence related to cybercrimes, such as hacking, identity theft, or fraud.

**5.Personal Use:** Some individuals may use keyloggers to keep track of their own keystrokes for personal productivity or record-keeping purposes.

# CHAPTER 2

**2.0 Problem identification**

The need for keyloggers arises primarily in security-related scenarios such as monitoring employee activity, parental control, or investigating unauthorized access. The need for keyloggers arises primarily in security-related scenarios such as monitoring employee activity, parental control, or investigating unauthorized access.

**2.1 Problem Function**

Keyloggers, software or hardware tools that record keystrokes on a computer, can serve legitimate purposes like parental control or employee monitoring. However, they are frequently associated with malicious intent, such as stealing sensitive information like passwords or financial data. Ethical considerations include ensuring user consent and lawful use.

1. **2 Operating Environment**

A keylogger can work in any environment such as a PC, Laptop, Desktop, Smart Phones, Tablets, etc. A Keylogger is installed in a device like a payload and it starts capturing any keystroke or any movements that is happening in the device.

* 1. **Features**

## The solution to the above existing problem is that we can build a software keyloggers instead of hardware keyloggers. The proposed model provides the solution that reduces the difficulties while installing the keylogger in the target system. Since, software keylogger can be installed remotely and does not need any physical access of the target system. Proposed software is efficient enough to get installed in targeted system by itself when the user for example clicks the malicious link sent to him through mail or any social media and finally captures all the keystrokes of the user while he is logged into the system, saves the logs in a folder or sends the log directly to the mail address of the third party.

**Figure STYLEREF 1 \s 2. SEQ Figure \\* ARABIC \s 1 1: This is my First Figure**

* 1. **Module Used:**

A Java package called JNativeHook makes it possible to establish global keyboard and mouse listeners. When you need to keep an eye on or react to system-wide input events—even when your application is not focused—it is especially helpful. JNativeHook accomplishes this by interacting with the operating system's input event systems via native code.

This is how it operates: JNativeHook builds a system-wide hook that records operating system-level keyboard and mouse events. When an event happens, it is converted by the native code into a Java event and sent to the Java application. This makes it feasible for the program to react to events such as mouse movements and global shortcuts, which would be hard to record with pure Java.   
  
Additionally, JNativeHook has the capability of posting native events back to the original operating system, which enables the program to mimic mouse and keyboard interaction.   
  
All things considered, JNativeHook gives Java apps a low-level interface with the OS, opening up functionalities that would not be achievable with pure Java.

Implemented as a library, JNativeHook offers Java global keyboard and mouse listeners. It creates low-level system-wide hooks using Java's native interface, which is platform-dependent, and then leverages those events to send them to the application. This makes it feasible for the application to detect mouse movements or global shortcuts, which is not possible with pure Java.   
  
There are many different types of events that JNativeHook can handle, including mouse click, mouse move, mouse drag, mouse wheel, key press, key release, and key typed events.   
  
The application can simulate keyboard and mouse input thanks to the library's ability to transmit native events back to the native operating system.

1. The product will be operating in windows, Linux environment. The hardware
2. configuration include Hard Disk: 40 GB, Monitor: 15” Color monitor, Keyboard: 122
3. keys. The basic input devices required are keyboard, mouse and output devices are

monitor, mobile devices etc**2.4 Modules used**The pynput library in Python enables the programmers to control and monitor

**CHAPTER 3**

**3.0 Code Implementation**

In this Project we are using Java as our programming language so we will be using Eclipse as our IDE. Here we have created a package named keylogger and in that package we had created 3 class files in java named as “Keylogger.java”, “KeyloggerDriver.java” and at last “Logger.java”.

1. **Keylogger.java:**



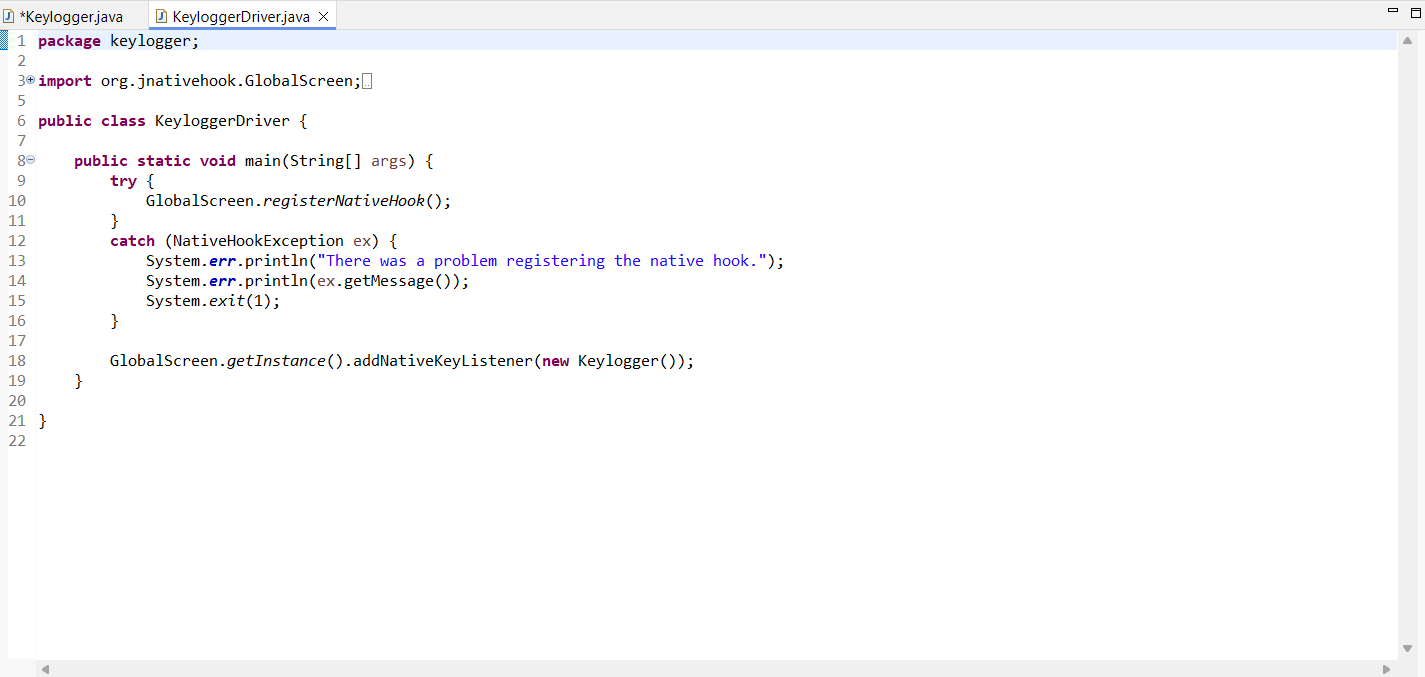
This Java code makes use of the JNativeHook package to construct a keylogger. Here's a summary of what is taking place:   
**Keylogger for Classes**:

1. The JNativeHook library's NativeKeyListener interface is implemented by this class. There are ways to listen to native keyboard events through this interface.
2. There are two fields in the class:
3. The object called logger is an instance of a personalized Logger class that records keystroke events in a file.
4. path: a static string that indicates the location of the file (C:\\users\\public\\log.txt) where the log is to be written.

**Keylogger Constructor() :**  
By giving the path string as an argument, the constructor initializes the logger field with a new instance of the Logger class.

**NativeKeyEvent arg0: NativeKeyPressed method:**  
i) Every time a native keyboard key press event takes place, this method is triggered.   
ii) It accepts as an argument a NativeKeyEvent object, which stands for the keyboard event.   
iii) The NativeKeyEvent object is passed as a parameter when the method invokes the logger object's log method.   
iv) The log method prints "Success" to the console if it returns true. "Fail" is printed if not.

1. **KeyloggerDriver.java**

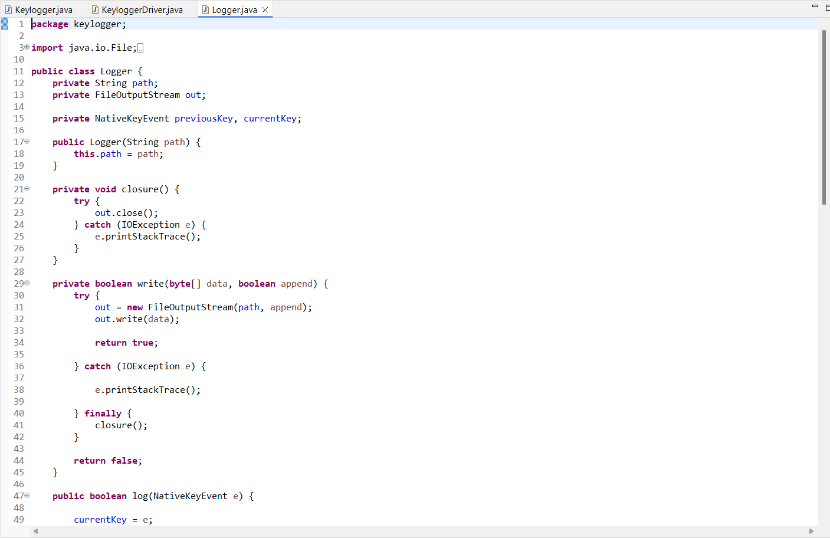


This Java code serves as the keylogger application's driver. Here's a summary of what is taking place:   
  
**Keylogger Driver Class:**The main method of the class serves as the Java application's entry point.

**Main method (String[] args):**  
i) The technique uses the GlobalScreen.registerNativeHook() to attempt to register a native hook. The JNativeHook library's registerNativeHook() function.   
ii) After a successful registration, the code moves on to the following line. When there's an exception, the code captures the NativeHookException and sends a message to the standard error stream stating that the native hook registration failed.   
iv) Publishes the message about the exception to the default error stream.   
v) System. Exit(1) is used to exit the application with a status code of 1.

**Registering the Keylogger:**

1. The code uses GlobalScreen.getInstance() to obtain an instance of the GlobalScreen class if the native hook registration is successful.
2. Next, it uses addNativeKeyListener(new Keylogger()) to add a new instance of the Keylogger class as a native key listener.
3. As a result, alerts for native keyboard events—like key presses, releases, and types—will be sent to the Keylogger class.
4. In conclusion, this code adds a key listener to receive notifications for native keyboard events and registers a native hook to start a keylogger application.
5. An error message appears and the application terminates if there is a problem registering the native hook.
6. **Logger.java**







The Logger class in this Java code is in charge of recording keyboard events to a file. Here's a summary of what is taking place:   
  
**Logger Constructor (String path)**:  
i) The file path where the log will be stored is represented by the path string that is passed to the constructor.   
ii) The supplied value is initialized in the path field.

**Closure() method**:  
i) To stop resource leaks, the FileOutputStream object is closed using this function.   
ii) Any IOException that might arise during the closing procedure is caught by it.

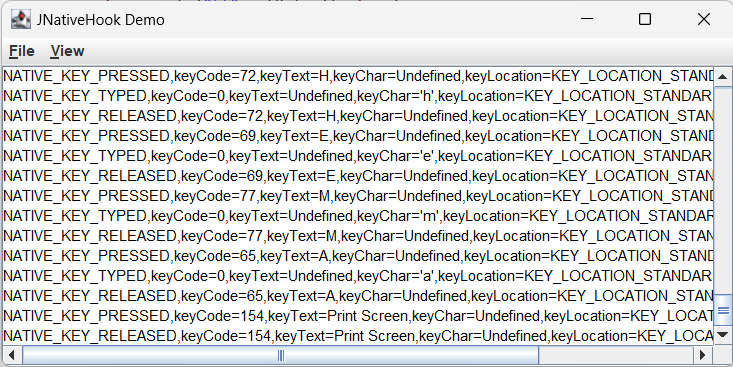
**Method write(byte[] data, boolean append)**:  
  
i) The file indicated by the path field is where this procedure saves the data byte array that was supplied.   
ii) To determine whether to add to the file or overwrite it, it accepts a boolean input called append.   
iii) The file is opened in append mode if append is true; otherwise, it is opened in overwrite mode.   
iv) It uses the FileOutputStream object out to write the data byte array to the file.   
v) It captures the exception and publishes the stack trace if an IOException happens during writing.   
vi) Finally, it closes the FileOutputStream object by invoking the closure() method.

**Method log(NativeKeyEvent e)**:  
i) Every time there is a keyboard event, this method is invoked.  
ii) It accepts as an argument a NativeKeyEvent object e, which stands for the keyboard event.  
iii) It verifies the kind of significant event and carries out the subsequent tasks:  
 a) When a modifier key (such as Caps Lock, Alt, Control, Windows, Escape, or Dead Acute) is used as the key event, the previousKey field is updated to reflect the new key event and false is returned.  
 b) It sends a newline character (\n\r) to the file and returns true if the key event is an Enter key.  
 c) It writes a space character() to the file and returns true if the key event is a Space key.

d) The backspace() method is called to erase the last character from the file and returns true if the key event is the Backspace key.   
 e) It inserts a single quotation character (') to the file and returns true if the key event is a quote key.   
 f) When a Comma or Stop key is pressed, the function writes the character (,) to the file and returns true.   
 g) It writes a period character (.) to the file and returns true if the key event is a Period key.   
 h) It sends a tab character (\t) to the file and returns true if the key event is a Tab key.

i) In case the key event consists of a Shift key accompanied by a Slash or Backslash key, the file is written with a question mark character (?) and you get true.   
 j) If not, it uses NativeKeyEvent.getKeyText(e.getKeyCode()) to retrieve the key text from the NativeKeyEvent object and writes it to the file as a byte array.

**3.1 Testing**



**CHAPTER 4**

# 4.0 SOFTWARE REQUIREMENT SPECIFICATION

1. Windows 7 or higher
2. • PHP
3. • Google Chrome Browser
4. • MYSQL
5. • XAMPP Server
6. • Sublime Text / Jupyter Notebook

• Windows 7 or higher

• JAVA

• ECLIPSE

• JDK 17 or higher

• JNativeHook.jar

# CHAPTER 5

# 5.0 CONCLUSION AND FUTURE WORK

The product can play out the proposed work like a fundamental keylogger does to get all secret data from client of the framework by getting their keystrokes occasions and mouse clicks without the information on the client. So client of the framework is ignorant of things occurring in foundation. The software is able to monitor data and store the data in a specific file. The software is also able to hide itself from the owner if the system while it runs in background. Thus, I accept that my methodology extensively increases current standards for observing the information and gathering it for either lawful or unlawful reason.Developing a software to monitoring employee activity for security purposes, parental control to ensure child safety online, and forensic investigations into unauthorized access. In these contexts, keyloggers can provide valuable insights into user behaviour and help prevent security breaches or identify wrongdoing.

# CHAPTER 6

# 6.0 BIBLIOGRAPHY

# [1] Keystroke Logging Keylogging" Erudio Security, LLC T Olzak T. Olzak." Keystroke Logging Keylogging" Erudio Security, LLC. 2008.

# [2] Thorsten Holz, M. E. (2009). Learning More about the Underground Economy: A Case-Study of Keyloggers and Dropzones. Thorsten Holz, Markus Engelberth, Felix C. Freiling, 1-18.

[3] The Evolution of Keylogger Technologies: A Survey from Historical Origins to Emerging Opportunities Marco Salas-Nino, Grant Ritter, Daniel Hamdan, Tao Wang, Tao Hou

[4] Skeleton keys (2010): The purpose and applications of keyloggers, Oleg Zaitsev

[5]Analysis of Keyloggers in Cybersecurity, Viraj Prajapati, <https://www.researchgate.net/publication/346337625_Analysis_of_Keyloggers_in_Cybersecurity>

[1] Martin Vuagnoux, S. P. (2009). Compromising electromagnetic emanations of

wired and wireless keyboards. USENIX security symposium, 1–16.

[2] S. P. Goring, J. R. (2007). Anti-keylogging measures for secure internet login:

an example of the law of unintended consequences. Computers & Security, 1-9.

[3] Strahija, N. (2003, February 8). Student charged after college computers

hacked. Retrieved from Xtrix security: http://www.xatrix.org/article2641.html

[4] Thorsten Holz, M. E. (2009). Learning More about the Underground Economy:

A Case-Study of Keyloggers and Dropzones. Thorsten Holz, Markus Engelberth,

Felix C. Freiling, 1-18.

[1] Martin Vuagnoux, S. P. (2009). Compromising electromagnetic emanations of

wired and wireless keyboards. USENIX security symposium, 1–16.

[2] S. P. Goring, J. R. (2007). Anti-keylogging measures for secure internet login:

an example of the law of unintended consequences. Computers & Security, 1-9.

[3] Strahija, N. (2003, February 8). Student charged after college computers

hacked. Retrieved from Xtrix security: http://www.xatrix.org/article2641.html

[4] Thorsten Holz, M. E. (2009). Learning More about the Underground Economy:

A Case-Study of Keyloggers and Dropzones. Thorsten Holz, Markus Engelberth,

Felix C. Freiling, 1-18