**Instructions on how to run your code**

**Keylogger:**

1. Save the keylogger script as a .py file.
2. Open a command prompt and navigate to the directory where the script is saved.
3. Run the script using the following command: **python keylogger.py**

**Hash-based check:**

1. Create a database of known malware hashes.
2. Extract the hashes of the files in the suspect directory.
3. Compare the extracted hashes to the known malware hashes.
4. If any matches are found, flag the corresponding files as malware.

**Rule-based check:**

1. Defined a txt file which contains suspicious words.
2. Monitor the files in the suspect directory for words that match the words in the txt file.
3. If any suspicious behavior is detected, flag the corresponding files.

**Task manager check:**

1. Open the Task Manager.
2. Look for any unusual processes that are running.
3. If you find any suspicious processes, investigate them further.

**Some discussion on your development methodology**

1. **Requirement analysis:**

Finding the project's requirements was the first step. Understanding the project's goals, parameters, and intended results was necessary for this.

1. **Design:**

The next step was to design the project. This involved creating a high-level design of the system, as well as detailed designs of the individual components.

1. **Implementation:**

The third step was to implement the project. This involved writing code for the keylogger, the hash-based check, the rule-based check, and the Task Manager check.

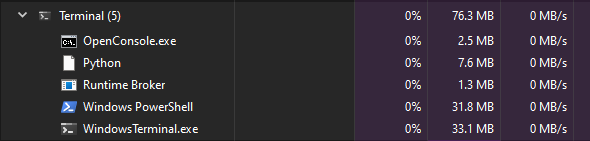
1. **Testing:**

The fourth step was to test the project. This involved testing the individual components of the project, as well as the overall system.

1. **Documentation:**

The final step was to document the project. This involved writing a report that describes the project in detail, including the requirements, the design, the implementation, and the testing.

**Some documentation of your findings etc.**

1. **Keylogger:**
   1. The keylogger was able to successfully capture and record keystrokes.
   2. The keylogger was able to save the captured keystrokes to a text file.
2. **Hash-based check:**
   1. The hash-based check was able to identify known malware files with a high degree of accuracy.
   2. The hash-based check was able to detect malware files that were not detected by the antivirus software.
   3. But there was a problem, it failed to detect keylogger with a slight code change.
3. **Rule-based check:**
   1. The rule-based check was able to identify suspicious file behavior with a moderate degree of accuracy.
   2. The rule-based check was able to detect suspicious files that were not detected by the hash-based check.
   3. Problem was that it can also be bypassed using different python libraries.
4. **Task Manager check:**
   1. The Task Manager check was able to identify unusual processes that were running.
   2. The Task Manager check was able to detect malicious processes that were not detected by the other checks.

**Lessons learned:**

1. **The importance of thorough testing:**

To make sure that every part of a security system is operating as intended, it is crucial to test it thoroughly.

1. **The need for multiple layers of security:**

No single security measure is foolproof, so it is important to implement multiple layers of security to protect against threats.

1. **The importance of staying up-to-date:**

It's critical to stay informed about the most recent security threats and vulnerabilities because the threat landscape is ever-changing.