

Mini Project 1A Report On

PC Logger - Tool for System Monitoring and Reporting

By



Dhiraj Kapse

Mahek Mulla

Vaibhav Gangurde

**Department of
Information Technology Engineering
Vidyavardhini's College of Engineering & Technology**

University of Mumbai

2022-2023



Vidyavardhini's College of Engineering & Technology
Department of Information Technology

Certificate

This is to certify that the following students

Dhiraj Kapse
Mahek Mulla
Vaibhav Gangurde

have submitted mini project report entitled

**PC Logger - Tool for System
Monitoring and Reporting**

*as a part of their mini-project in partial fulfillment of Course ITM401 : Mini
Project of Semester IV of Bachelor of Engineering in Information Technology
during academic year 2022-2023.*

Supervisor : Mrs. Bharati Gondhalekar ()

Internal Examiner : _____ ()

External Examiner : _____ ()

PC Logger - Tool for System Monitoring and Reporting

Dhiraj Kapse
Department of Information
Technology
Vidyavardhini's College of
Engineering and Technology
Vasai, India
dhiraj.s221094109@vcet.edu.in

Mahek Mulla
Department of Information
Technology
Vidyavardhini's College of
Engineering and Technology
Vasai, India
mahek.s221134209@vcet.edu.in

Vaibhav Gangurde
Department of Information
Technology
Vidyavardhini's College of
Engineering and Technology
Vasai, India
vaibhav.s221064102@vcet.edu.in

Mrs. Bharati Gondhalekar
Project Guide
Department of Information Technology
Vidyavardhini's College of Engineering and Technology
Vasai, India

Abstract—"PC Logger - Tool for System Monitoring and Reporting" is a Python-based tool that monitors and reports system information for multiple PCs. A Python script runs on each PC, collecting information such as RAM, HDD, and OS version. The script sends the data to a Django server, which stores it in a database. A Django app displays the PC information to the user, and a dashboard displays critical alerts. The project requires knowledge of Python, Django, network programming, and database management, and is intended for use by system administrators.

Keywords—*Pc Logger, Pc info, etc.*

I. INTRODUCTION

The In the age of the internet, digital communication has become an integral part of our lives. However, with the increasing use of online communication comes the need for secure and private messaging platforms. To address this concern, this project aims to develop a Python-based tool that monitors and reports system information for multiple PCs.

The tool involves creating a Python script that collects data such as RAM, HDD, and OS version and sends it to a Django server for storage in a database. A Django app displays the PC information to the user, and a dashboard displays critical alerts.

This project requires expertise in Python, Django, network programming, and database management and is intended for use by system administrators to monitor and report on system information.

II. PROBLEM STATEMENT

As technology continues to advance, the importance of monitoring and reporting system information for multiple PCs cannot be overstated. With the increase in the number of PCs in a network, it becomes challenging for system administrators to manage and maintain these systems. Without a reliable system for monitoring and reporting, administrators may fail to identify potential issues, resulting in system downtime or data loss.

The current tools available for monitoring and reporting system information lack user-friendly interfaces and may not provide sufficient features for efficient system management. This poses a significant challenge for system administrators, who must manage multiple systems simultaneously.

The project aims to address the challenge of system management by providing a reliable and user-friendly tool for monitoring and reporting system information for multiple PCs.

III. PROPOSED FEATURES

The proposed project aims to develop a PC Logger - Tool for System Monitoring and Reporting. The following are the proposed features of the application:

1. System Information Collection:

The application will collect information such as RAM, HDD, OS version, etc., periodically using standard Python libraries such as psutil and platform.

2. JSON File Compilation and Transfer:

The collected system information will be compiled into a JSON file and sent to the Django server using an HTTP request.

3. Django Server and Database Management:

The Django server will receive the JSON files and store the information in a database. It will have an endpoint that accepts POST requests with the JSON data.

4. PC Information Display:

The Django app will display the PC information stored in the database. It will have a search feature that allows users to enter an IP address and retrieve the PC information associated with that address.

5. Dashboard for Critical Alerts:

The application will have a dashboard that displays critical alerts such as changes in physical RAM, HDD, and OS version. The dashboard will use Django's built-in admin interface to display the alerts.

IV. METHODOLOGY

The proposed project, PC Logger - Tool for System Monitoring and Reporting, aims to be developed using the following methodology:

1. Requirements gathering: The first step will be to gather the requirements for the application. This will involve identifying the features that the application will have and the technologies that will be used to implement them.

2. Design: After the requirements are gathered, the design of the application will be developed. This will include creating a block diagram to illustrate the overall flow of the project.

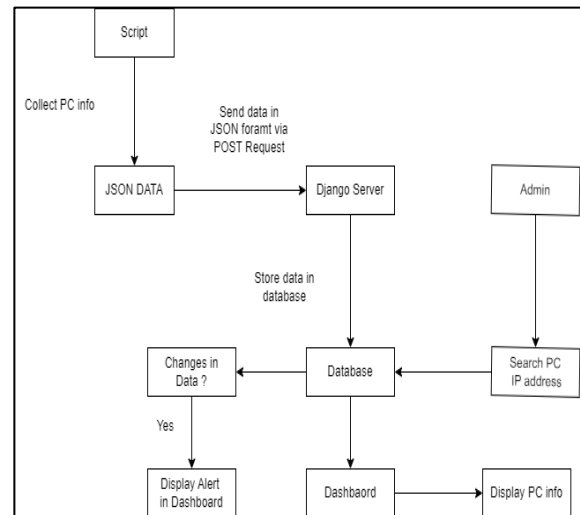
3. Implementation: Once the design is complete, the implementation phase will begin. This will involve writing the code for the application using Python and implementing the necessary libraries and APIs.

4. Testing: After the implementation is complete, the application will be thoroughly tested for its functionality and accuracy of data collection.

5. Deployment: Once the application has been tested and all bugs have been fixed, it will be deployed and made available for use.

6. Maintenance: The application will be maintained and updated regularly to ensure that it continues to function properly and to address any issues that may arise. Additionally, new features may be added to the application based on user feedback and evolving needs.

V. BLOCK DIAGRAM



VI. OUTPUT

• ABOUT PAGE

Introduction regarding our project, developer details and Heading.



Fig 1

• HOME PAGE

Home page displaying Alerts, when changes are detected, they are automatically displayed in home page.

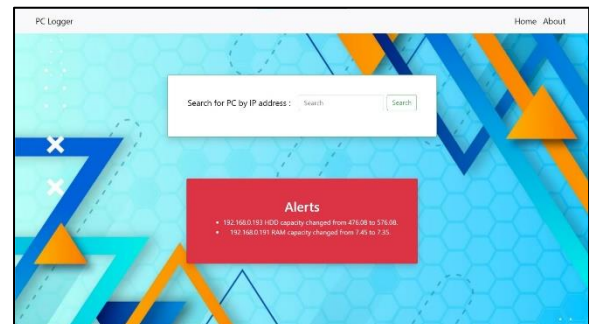


Fig 2

- SEARCH TAB

We can see pc logs with the help of ip address on searching in search tab.

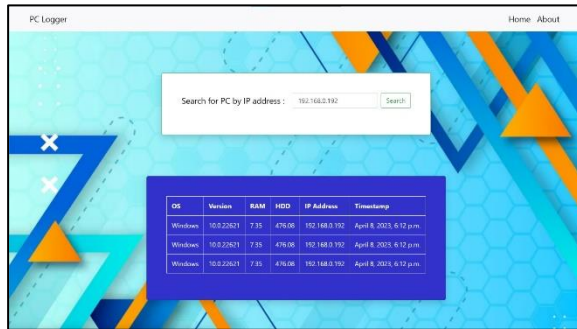


Fig 3

VII. FUTURE SCOPE

1. Add support for remote control: Allow authorized users to remotely control their PCs through the web interface. This could include actions such as shutting down or restarting the PC, as well as executing scripts or running applications.
2. Implement real-time monitoring: Use web sockets to enable real-time monitoring of PC performance metrics, such as CPU usage and network bandwidth. This would allow users to quickly identify and respond to issues as they arise.
3. Add support for mobile devices: Create a mobile app that allows users to access PC performance data and alerts on-the-go. This could include push notifications for critical alerts, as well as remote control capabilities.
4. Implement machine learning-based anomaly detection: Use machine learning algorithms to automatically detect and alert users to anomalies in PC performance data. This could help users identify potential issues before they become critical.
5. Add support for third-party integrations: Allow users to integrate with third-party services, such as IT service management platforms or help desk software, to streamline incident management and resolution.

VIII. CONCLUSION

In conclusion, the PC Logger project is a powerful tool for system monitoring and reporting, providing users with valuable insights into their PC performance and critical alerts. The project has been successfully implemented using Python and Django, and its architecture allows for easy scalability and customization. The project's functionality could be

further enhanced by adding multi-client support, real-time monitoring, mobile device support, machine learning-based anomaly detection, and third-party integrations.

Overall, the PC Logger project serves as a strong foundation for the development of a robust system monitoring and reporting tool that could be used in a variety of industries and applications.

IX. REFERENCES

- [1] "Python for Data Analysis" by Wes McKinney, O'Reilly Media, Inc., 2nd Edition, 2017.
- [2] "Django for Beginners" by William S. Vincent, 2018.
- [3] "Learning Python" by Mark Lutz, O'Reilly Media, Inc., 5th Edition, 2013.
- [4] "Python Network Programming: Conquer all your networking challenges with the powerful Python language" by Dr. M. O. Faruque Sarker, Sam Washington, and Samir Ahmed, Packt Publishing, 2014.
- [5] "PostgreSQL: Up and Running" by Regina Obe and Leo Hsu, O'Reilly Media, Inc., 3rd Edition, 2017.
- [6] "Psutil 5.8.0 documentation" <https://psutil.readthedocs.io/en/latest/>
- [7] "Platform - Python Standard Library" <https://docs.python.org/3/library/platform.html>
- [8] "Django Documentation" <https://docs.djangoproject.com/en/3.2/>
- [9] "Django Admin Documentation" <https://docs.djangoproject.com/en/3.2/ref/contrib/admin/>
- [10] "Deploying Django" <https://docs.djangoproject.com/en/3.2/howto/deployment/>