DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING
NORTH SOUTH UNIVERSITY

Operating Systems Design - CSE 323.6

Development of Monitoring and Activity Logging Application

Submitted by,
Rafid Ahmmad 2011237042

Submitted to,
Rashed Mazumder
Assistant Professor

Date of submission - 15/06/2023

Table Of Content

Abstract

The "Development of Monitoring and Activity Logging Application" project aims to create a user-friendly graphical user interface (GUI) application that monitors and logs system activity. By utilizing the psutil library, the application retrieves real-time data on CPU usage, memory usage, network activity, and running processes. This review provides an overview of the project, including its abstract, introduction, background study, methodology, results and discussion, conclusion, and references.

Introduction

The "Development of Monitoring and Activity Logging Application" project focuses on creating a GUI application to monitor and log system activity in real-time. The application leverages the psutil library to retrieve vital system information such as CPU usage, memory usage, network activity, and running processes. By displaying this data on the GUI, users can effectively monitor system performance, identify resource-consuming processes, and analyze overall system activity. The following review provides an in-depth analysis of the project's components and outcomes.

Background Study

Similar Applications: In the field of system monitoring and activity logging, various applications have been developed to provide insights into system performance and resource utilization. Some notable examples include:

1. Task Manager (Windows):
   - Task Manager is a built-in system monitoring tool in Windows operating systems.
   - It displays real-time information on CPU usage, memory utilization, disk activity, and network activity.
   - Users can view running processes, their resource consumption, and terminate or prioritize them if needed.
2. Activity Monitor (macOS):
   - Activity Monitor is the macOS equivalent of Task Manager.
   - It provides information on CPU usage, memory usage, disk activity, and network activity.
   - Users can monitor running processes, analyze resource consumption, and manage system performance.
3. htop (Linux):
   - htop is a command-line system monitoring tool available for Linux-based systems.
   - It offers a comprehensive view of CPU usage, memory usage, and process management.
   - htop displays running processes in a hierarchical structure, allowing users to monitor resource consumption effectively.

These existing applications have proven to be valuable tools for system administrators, developers, and users interested in monitoring and optimizing system performance. They provide insights into key system metrics and enable users to manage processes efficiently.

Methodology

The "Development of Monitoring and Activity Logging Application" project follows a well-defined methodology to achieve its objectives. The key steps involved in the methodology are as follows:

1. Importing Libraries:
    - The project begins by importing the necessary libraries, including psutil for system information retrieval and tkinter for GUI development.
2. Retrieving System Information:
    - The application utilizes the psutil library to collect real-time data on CPU usage, memory utilization, network activity, and running processes.
    - The get_system_info() function is implemented to retrieve and organize the system information.
3. Creating the GUI:
    - The project uses the tkinter library to create a graphical user interface for the application.
    - Labels are used to display system information such as CPU usage, memory usage, and network activity.
    - The process_list table, implemented using the Treeview widget, presents the list of running processes.
4. Updating System Information:
    - The update_system_info() function is designed to refresh the system information in the GUI at regular intervals.
    - It retrieves the latest system data using get_system_info() and updates the relevant labels and process_list table.
5. Running the Application:
    - The project runs the main event loop (root.mainloop()) to ensure the application remains responsive and handles user interactions.

The chosen methodology allows for efficient retrieval of system information and real-time updates of the GUI elements. By leveraging the capabilities of the psutil library and tkinter GUI framework, the project creates a functional and user-friendly monitoring and activity logging application.

Please note that the methodology discussed here is based on the provided code and may be subject to specific modifications or enhancements in the actual implementation.

Results and Discussion

The "Development of Monitoring and Activity Logging Application" project successfully creates a functional GUI application for monitoring and logging system activity. By leveraging the psutil library and tkinter GUI framework, the application retrieves real-time data on CPU usage, memory utilization, network activity, and running processes.

The GUI elements, including labels for CPU usage, memory usage, and network activity, are dynamically updated with the latest system information. This enables users to monitor the performance of their system in real-time. The process_list table provides a comprehensive view of the running processes, displaying their process ID (PID), name, username, CPU usage, and memory usage. This allows users to identify resource-intensive processes and analyze their impact on system performance.

The application's design ensures a responsive user experience, as the system information is updated at regular intervals. By scheduling the update_system_info() function to run periodically, the application continuously refreshes the displayed data. This provides users with an up-to-date view of the system's performance metrics and running processes.

The "Development of Monitoring and Activity Logging Application" project's GUI interface and data visualization features enhance the usability and effectiveness of system monitoring. Users can easily interpret the collected data and make informed decisions regarding resource allocation and process management. The application serves as a valuable tool for system administrators, developers, and users interested in optimizing system performance and identifying potential bottlenecks.

Conclusion

In conclusion, the "Development of Monitoring and Activity Logging Application" project achieves its objective of creating a user-friendly GUI application for monitoring and logging system activity. By leveraging the psutil library, the application retrieves real-time data on CPU usage, memory utilization, network activity, and running processes. The GUI interface, including labels and the process_list table, is dynamically updated to provide users with a comprehensive view of their system's performance metrics.

The project's methodology ensures efficient retrieval and display of system information, while the chosen libraries (psutil and tkinter) offer robust functionality and flexibility. The application successfully fulfills its purpose by allowing users to monitor system performance, identify resource-consuming processes, and make informed decisions to optimize system efficiency.

Overall, the "Development of Monitoring and Activity Logging Application" project serves as a valuable tool for monitoring and analyzing system activity. It contributes to improving system performance and facilitating effective resource management for a wide range of users.

References

- Psutil library documentation. Retrieved from: <https://psutil.readthedocs.io/en/latest/>
- Python Software Foundation. (n.d.). Tkinter — Python interface to Tcl/Tk. Retrieved from: <https://docs.python.org/3/library/tkinter.html>