2025

Activity Tracker with Azure Integration

CODECRAFT @APPTWARE

SUJHAL S GURAV

Title: Activity Tracker with Azure Integration

Abstract The Activity Tracker project is a robust and scalable solution designed to monitor user activity across devices while integrating with Azure services for data storage, analysis, and visualization. By leveraging Azure Monitor and Log Analytics, the system ensures real-time tracking, secure storage, and meaningful insights into user activity. This project allows customization and personalization of dashboards, enabling clients to tailor data visualization to their specific needs.

Introduction In the digital age, understanding user behaviour and system performance is critical for optimizing productivity and ensuring security. This project was developed to:

1. Track real-time user activities, such as mouse movements and application usage.
2. Integrate with Azure services for seamless data logging, storage, and analysis.
3. Provide clients with customizable dashboards to visualize and derive actionable insights.

This project addresses the need for transparency, productivity analysis, and enhanced decision-making using cloud technology.

Methodology The Activity Tracker employs a combination of Python scripting and Bash automation to:

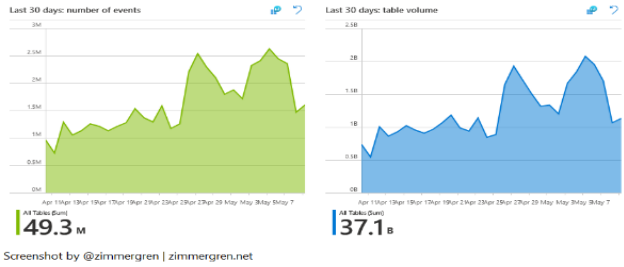
1. **Track User Activity:**
    - Monitor mouse movements using Python's pynput library.
    - Capture application usage and categorize activities as productive or non-productive.
2. **Data Logging:**
    - Use Azure Log Analytics Workspace to store data in custom tables.
    - Send logs to Azure every 15 seconds via API calls.
3. **Cloud Integration:**
    - Authenticate and interact with Azure services using manually generated tokens.
    - Purge outdated data periodically to ensure efficient storage.
4. **Dashboard Customization:**
    - Provide clients with personalized dashboards on Azure Monitor to analyze data tailored to their requirements.

Implementation

1. **Python Script for Activity Tracking:**
    - Tracks mouse movement, application usage, and system metrics.
    - Sends data to Azure Log Analytics every 15 seconds.
2. **Bash Script for Automation:**
    - Runs the Python script and ensures continuous operation.
    - Periodically purges old data from Azure Log Analytics.
3. **Azure Integration:**
    - Data is logged into a Log Analytics Workspace.
    - Insights are generated using KQL (Kusto Query Language) queries.

Results The project successfully captures and logs user activity data to Azure. Insights include:

- **Real-time User Behaviour:** Mouse movement counts and distinct positions.
- **Categorized Application Usage:** Productive vs. non-productive applications.
- **Dashboard Insights:** Visual representation of data, including bar charts and time-series graphs.



Future Scope

1. **Enhanced Tracking:**
    - Add keyboard activity monitoring.
    - Capture network activity and application response times.
2. **AI Integration:**
    - Use machine learning to detect anomalies in user behaviour.
    - Predict productivity trends based on historical data.
3. **Cross-Platform Compatibility:**
    - Extend functionality to macOS and Android devices.
4. **Client-Specific Customization:**
    - Allow users to define custom activity categories.
    - Automate report generation and delivery.

Conclusion The Activity Tracker project demonstrates how cloud technology can transform user activity tracking into actionable insights. By integrating with Azure, the project achieves scalability, security, and personalization. Future iterations aim to enhance functionality and broaden applicability, making it a comprehensive solution for productivity analysis and user behaviour monitoring.