Work Instruction

PoC/Working Demo

|  |  |
| --- | --- |
| **Author:** | **Praveen Kumar Madhava Rao** |
| **Owner:** | **Praveen Kumar Madhava Rao** |
| **Contributors:** |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

|  |  |
| --- | --- |
| **Version:** | **1.0** |
| **Date:** | **17/02/2024** |

**Table of Contents**

[**Preface** 3](#_Toc159164896)

[**Introduction:** 4](#_Toc159164897)

[**Create a New Python Module for CPU Monitoring:** 4](#_Toc159164898)

[**Add Sample Code Snippet for CPU Consumption:** 5](#_Toc159164899)

[**Testing and Outcome:** 5](#_Toc159164900)

[**Conclusion:** 5](#_Toc159164901)

# **Preface**

The purpose of this document is to provide clear and concise instructions for performing the following:

* Integrating CPU monitoring metrics into the EcoVision plugin.
* Setting up the EcoVision project in Python.
* Creating sample code for CPU consumption.
* Testing and evaluating the outcome of the EcoVision plugin with integrated CPU monitoring.

# **Introduction:**

This work instruction provides step-by-step guidance on integrating CPU monitoring functionality into the EcoVision plugin project using Python.   
The objective is to enable real-time monitoring of CPU utilization within the EcoVision environment and provide actionable insights based on the observed CPU load.

**Prerequisites:**

1. Python installed on the system.
2. Access to the EcoVision plugin project in the Python environment.

**Open EcoVision Plugin Project:**

* Navigate to the directory containing the EcoVision plugin project in the Python environment.

# **Create a New Python Module for CPU Monitoring:**

* Create a new Python file named **cpu\_utilization\_monitor.py** within the EcoVision plugin project directory.
* Copy and paste the provided code into **cpu\_utilization\_monitor.py**. This module encapsulates the functionality to monitor CPU utilization using the **psutil** library.

import psutil

class CPUUtilizationMonitor:

def get\_cpu\_utilization(self):

return psutil.cpu\_percent(interval=1)

**Modify the EcoVision Plugin Code:**

* Open the main Python file (**ecovision\_plugin.py** or similar) in the EcoVision plugin project.
* Import the **CPUUtilizationMonitor** class from **cpu\_utilization\_monitor.py**.
* Instantiate the **CPUUtilizationMonitor** class and use it to monitor CPU utilization within the EcoVision plugin.

from cpu\_utilization\_monitor import CPUUtilizationMonitor

import time

def main():

cpu\_monitor = CPUUtilizationMonitor()

while True:

cpu\_utilization = cpu\_monitor.get\_cpu\_utilization()

print(f"CPU Utilization: {cpu\_utilization}%")

time.sleep(1) # Wait for 1 second

**Integrate CPU Monitoring into EcoVision Plugin:**

* Update the EcoVision plugin code to include functionality that interacts with CPUUtilizationMonitor to monitor CPU consumption.

/ Wait for

# **Add Sample Code Snippet for CPU Consumption:**

Within the EcoVision plugin code, add the provided sample code snippet that contributes to CPU consumption.

def heavy\_cpu\_task():

# Perform a heavy CPU task

total\_sum = 0

for i in range(1000000):

total\_sum += i

# Call the heavy CPU task function

heavy\_cpu\_task()

# **Testing and Outcome:**

* Execute the EcoVision plugin Python script.
* Monitor the CPU utilization displayed in the console.
* The EcoVision plugin should also provide a suggestion message to the user based on the observed CPU load.

# Example suggestion message

print("Suggestion: Consider optimizing the heavy\_cpu\_task function to reduce CPU load.")  
  
 **Additional Notes:**

* Ensure that the EcoVision plugin project is properly configured to interact with the CPUUtilizationMonitor class and provide actionable insights based on CPU utilization data.
* Test the integrated CPU monitoring functionality thoroughly to validate its effectiveness in providing meaningful suggestions to users.

# **Conclusion:**

This work instruction provides a detailed guide for integrating CPU monitoring functionality into the EcoVision plugin project using Visual Studio, ensuring that the EcoVision plugin interacts seamlessly with the **CPUUtilizationMonitor** class and provides actionable insights based on CPU utilization data. While this demonstration serves as a foundational example, there is room for further enhancements and optimizations. Please feel free to explore additional features and improvements to advance the solution. Remember, this is only a sample solution design demonstrating the capabilities of EcoVision, and customization is encouraged to meet our project needs.Top of Form

Bottom of Form