🚀 Advanced Exploratory Data Analysis EDA — High-Integrity Systems

This notebook focuses on developing deeper insights into high-integrity system data through EDA tasks. You'll investigate patterns, anomalies, and correlations across system metrics using Python.

This version includes:

- Conceptual understanding
- Thought-provoking analysis
- Hands-on statistical interpretation

In [3]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import skew, kurtosis

import seaborn as sns
sns.set(style='whitegrid')
```

Load the data

In [4]:
```
df = pd.read_csv("generated_his_system_metrics.csv", index_col=0)
df.head()
```

Out[4]:

system_id	sensor_latency_ms	cpu_usage_percent	memory_usage_mb	error_count	uptir
1	22.483571	58.892663	420.457194	0	13
2	19.308678	73.641250	316.802495	0	4
3	23.238443	24.021486	174.484049	1	9
4	27.615149	53.444539	96.321615	0	6
5	18.829233	35.240361	311.006058	2	4

In [189…
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 500 entries, 1 to 500
Data columns (total 10 columns):
 #   Column                       Non-Null Count  Dtype  
---  ------                       --------------  -----  
 0   sensor_latency_ms            500 non-null    float64
 1   cpu_usage_percent            500 non-null    float64
 2   memory_usage_mb              500 non-null    float64
 3   error_count                  500 non-null    int64  
 4   uptime_hours                 500 non-null    float64
 5   mode                         500 non-null    object 
 6   temperature_sensor           500 non-null    float64
 7   CPU_temperature              500 non-null    float64
 8   CPU_thermostat_stability     500 non-null    float64
 9   sensor_thermostat_stability  500 non-null    float64
dtypes: float64(8), int64(1), object(1)
memory usage: 43.0+ KB
```

Section 1, exploring distributions with histograms

✅ Task for students: Classify each variable as:

- Categorical (Nominal/Ordinal)
- Numerical (Discrete/Continuous)

By completing and running this cell, the categorical columns names will be saved as a list of strings in "categorical_cols". Likewise, the numerical_cols should be saved in "numerical_cols".

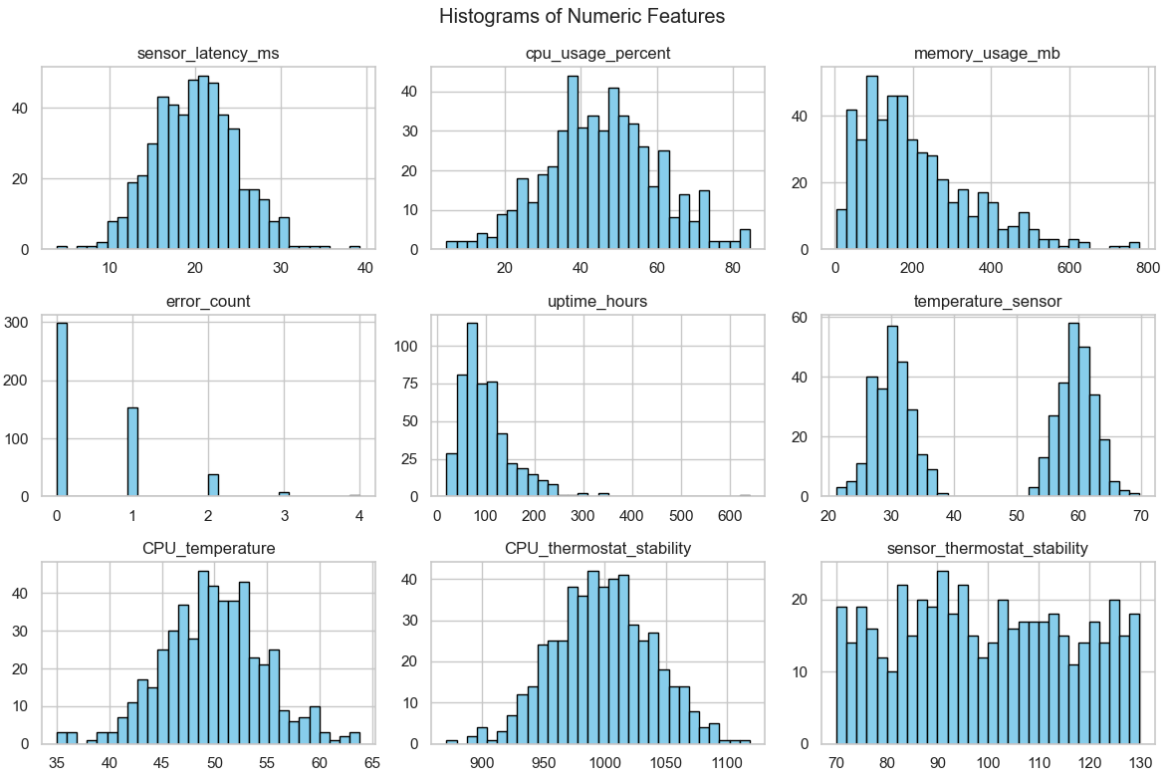
In [190…
```
categorical_cols = df.select_dtypes(include=['object', 'category']).columns.toli
numerical_cols = df.select_dtypes(include=['float64', 'int64']).columns.tolist()
```

✅ Task for students: You want now to check the distribution of the variables:

- When is a histogram preferable to a bar plot?
- Plot the histogram for the different variables

In [235…
```
Answer_Q11 = "when the variables are numerical"
```

In [191…
```
# Your code here: use matplotlib or pandas plotting to plot histograms
```



✅ Task for students:

- Describe the shape of the distribution of uptime.
- What maintenance strategies might be influenced by this shape?

In [236…
```
Answer_Q12 = "..."

Answer_Q13 = "..."
```

✅ Task for students:

- maintenance engineer presented the following plot for System Modes. Could this bar chart leing or hidding something? what should we change to fix it?
- do you think the maintenance engineer are responding well to the emergency?

In [ ]:
```
sns.countplot(x='mode', data=df)
plt.ylim(40, None)  # Start y-axis at 50
plt.title("System Mode Frequency")
plt.show()
```

In [ ]:
```
Answer_Q14 = "..."
```

In [1]:
```
# your code to fix it here
```

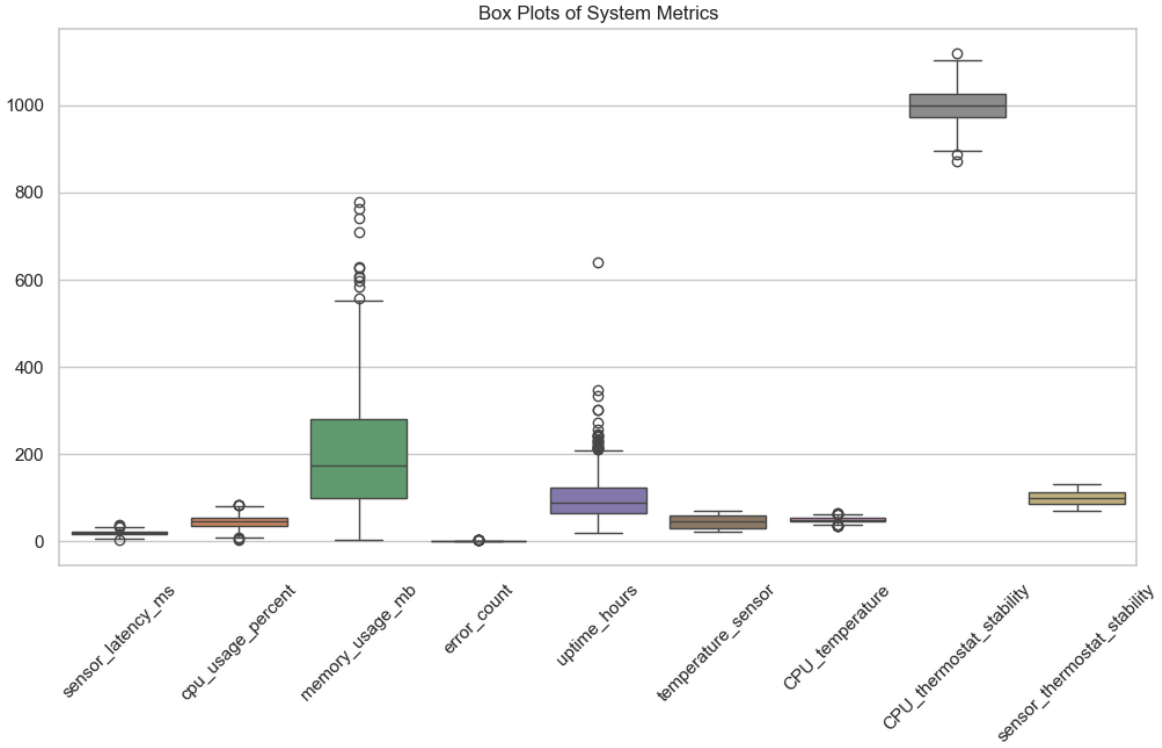
In [237…
```
Answer_Q15 = "..."
```

Section 2: exploring with box plots

✅ Task for students: Look at the following box plot and answer:

- what variables can a box plot be misleading? explain why?

In [239…
```
plt.figure(figsize=(12, 6))
sns.boxplot(data=df[numerical_cols])
plt.title('Box Plots of System Metrics')
plt.xticks(rotation=45)
plt.show()
```



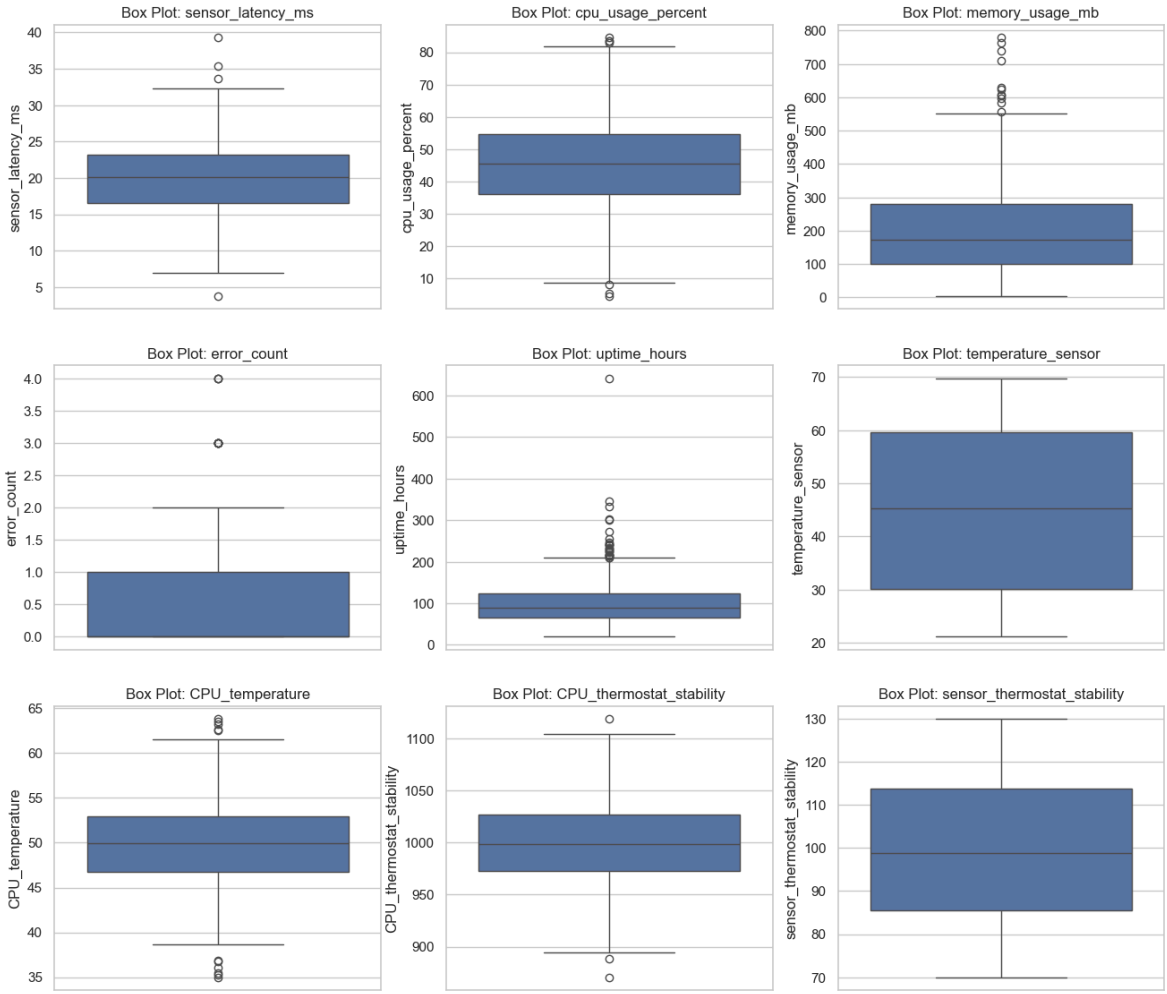
In [238…
```
# your answer here
Answer_Q21 = "..."
```

✅ Task for students: Look at the box plot above and answer:

- Is there anything else wrong with the plot and what can you do to improve it?

In [240…
```
# your answer here
Answer_Q22 = "..."
```

In [196…
```
# Your code to improve the plot here
```



Section 3: Measures of Center and Variability

✅ Task for students: print the descriptive statistics to summarize the central tendency, dispersion and shape of all columns in the data

In [197…

```
# your code here
```

Out[197…

	sensor_latency_ms	cpu_usage_percent	memory_usage_mb	error_count	uptime_h
count	500.000000	500.000000	500.000000	500.000000	500.00
mean	20.034190	45.477392	207.990699	0.530000	101.86
std	4.906266	14.669958	141.869183	0.755143	57.08
min	3.793663	4.546700	4.591898	0.000000	20.17
25%	16.498463	36.070624	99.706662	0.000000	65.49
50%	20.063986	45.427974	173.237281	0.000000	88.39
75%	23.183916	54.768634	280.858874	1.000000	123.13
max	39.263657	84.485731	778.689043	4.000000	641.05

✅ Task for students: Using the plots and the descriptive statistics above try to solve the following questions. Please complement with further measures if needed.

1. Which metric shows the most variability? Justify using both range and standard deviation.
2. Compare sensor latency and uptime. Which one is more consistent across systems
3. Can we compare variability across metrics with different units (e.g., ms vs MB)?
4. For which variable would the median be a better measure of "typical" than the mean?

In [241…
```
Answer_Q31 = "..."
Answer_Q32 = "..."
Answer_Q33 = "..."
Answer_Q34 = "..."
```

✅ Task for students:

1. Identify one metric where mean > median > mode. What does this indicate about skewness?
2. Can you find a metric there is not big differences between mean ≈ median ≈ mode? What does thus tell us about the distribution shape?

In [199…
```
# Your code here
```

```
sensor_latency_ms: Mean=20.03, Median=20.06, Mode=3.79
cpu_usage_percent: Mean=45.48, Median=45.43, Mode=4.55
memory_usage_mb: Mean=207.99, Median=173.24, Mode=4.59
error_count: Mean=0.53, Median=0.00, Mode=0.00
uptime_hours: Mean=101.86, Median=88.39, Mode=20.17
temperature_sensor: Mean=44.88, Median=45.29, Mode=21.21
CPU_temperature: Mean=49.86, Median=49.90, Mode=52.70
CPU_thermostat_stability: Mean=998.45, Median=998.54, Mode=870.76
sensor_thermostat_stability: Mean=99.67, Median=98.85, Mode=70.00
```

In [242…
```
Answer_Q35 = " ... "
```

In [201…
```
Answer_Q36 = " ... "
```

Section 4: Quantiles, percentiles, and Outliers

✅ Task for students:

- calculate quantiles [0.01, 0.25, 0.5, 0.75, 0.99] for uptime_hours and sensor_latency_ms

In [202…
```
# Your code here
```

Out[202…

	sensor_latency_ms	uptime_hours
0.01	9.802429	25.936007
0.25	16.498463	65.496874
0.50	20.063986	88.391079
0.75	23.183916	123.137689
0.99	30.958488	273.254100

✅ Task for students:

- calculate IQR for sensor_latency_ms

In [209…

```
# Your code here
# For sensor_latency_ms
```

```
IQR of sensor_latency_ms: 6.685453292992861
```

✅ Task for students:

- calculate IQR for uptime_hours

In [210…

```
# Your code here
# For uptime_hours
```

```
IQR of uptime_hours: 57.64081552793367
```

✅ Task for students: Can we use the IQR to flag extreme values (outliers) using percentiles, check how and apply this to detect and count the outliers in .

The IQR outlier rule:

- Lower bound = Q1 − 1.5 × IQR
- Upper bound = Q3 + 1.5 × IQR

In [204…

```
# Your code here
```

```
sensor_latency_ms Outliers detected: 4
```

In [205…

```
# Your code here
```

```
uptime_hours outliers detected: 486
```

✅ Task for students: Why might some outliers be legitimate and not errors?

Values outside this range can be flagged as potential outliers, under some assumptions — such as the data being approximately symmetric and without heavy tails. In highly skewed or long-tailed distributions, this rule may flag many legitimate values as "outliers.

In [206…

```
Answer_Q41 = "..."
```

✅ Task for students:

- How can you use the percentiles/ IQR to conclude about the shape of the distribution of the values of uptime_hours?
- Check box plot / histograms do they confirm the distribution shape that you can conclude using percentiles?

Section 5: compare variables

✅ Task for students: Which thermostat is more stable and consistent, print the mean values and standard deviations and explain why it can be misleading to use only these measures to compare

In [2]:
```
# Variables to compare
var1 = "CPU_thermostat_stability"
var2 = "sensor_thermostat_stability"

# Your code here
```

In [214…
```
# Your code here
Answer_Q51 = "..."
```

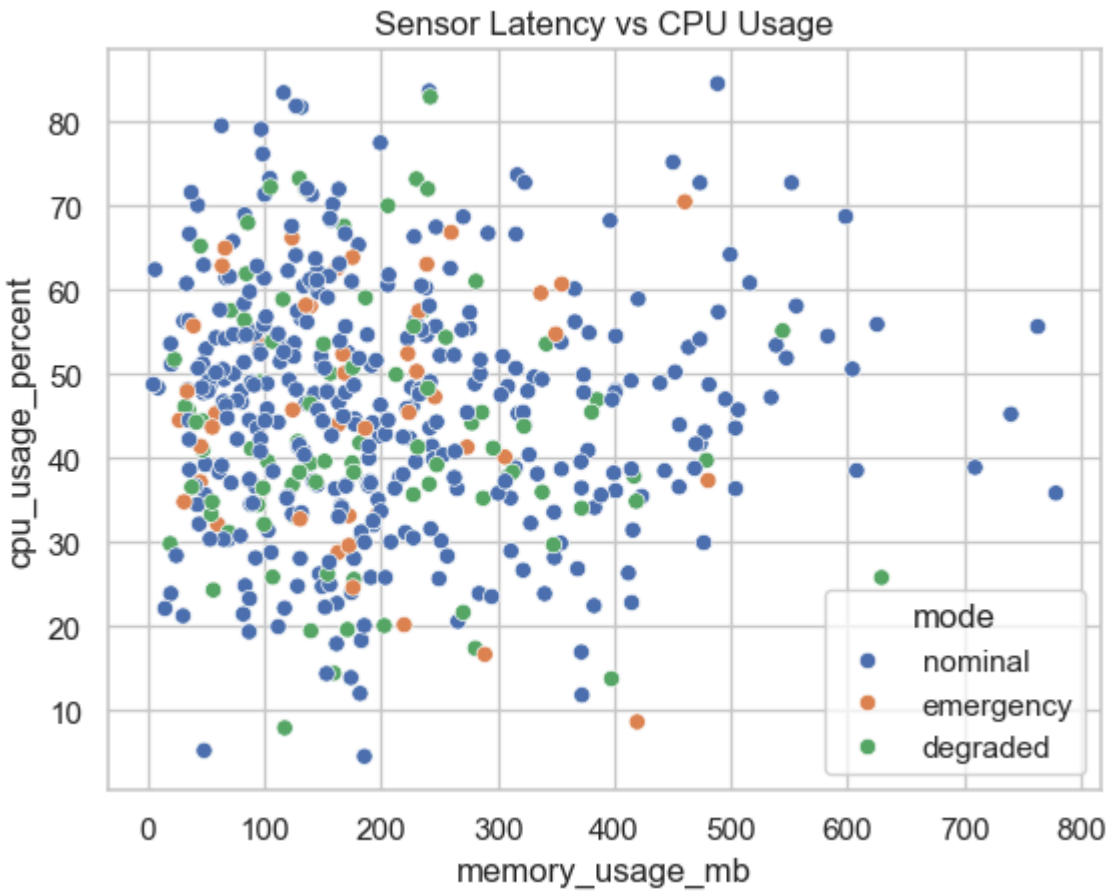
Section6: Scatter Plot Analysis

✅ Task for students:

1. can you identify any clusters or linear relationships between the variables: sensor_latency_ms and cpu_usage_percent. Which plot can we use to plot this?
2. Which mode is associated with higher CPU load?

In [243…
```
Answer_Q61 = "..."
```

In [229…
```
# Your code here
```



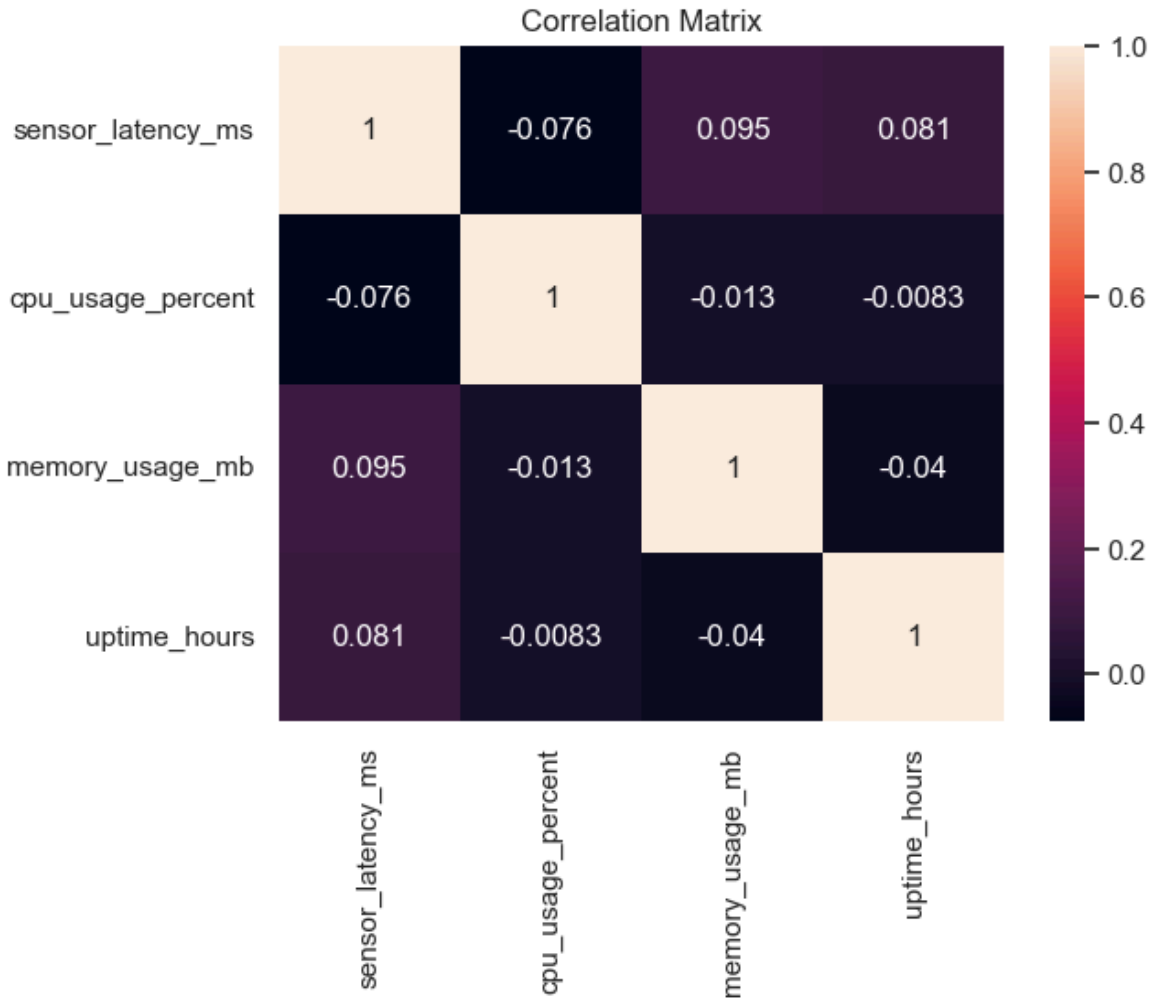
In [ ]:
```
Answer_Q62 = "..."
```

Section7: Correlation Matrix

✅ Task for students:

- Code to identify the strongest positive and negative relationships between the variables

In [9]:
```
# Your code here
```


Correlation Matrix
sensor_latency_ms
cpu_usage_percent
memory_usage_mb
uptime_hours
1
-0.076
0.095
0.081
-0.076
1
-0.013
-0.0083
0.095
-0.013
1
-0.04
0.081
-0.0083
-0.04
1
sensor_latency_ms
cpu_usage_percent
memory_usage_mb
uptime_hours
1.0
0.8
0.6
0.4
0.2
0.0

In [ ]: