

## ДОДАТОК А

### Графічний матеріал кваліфікаційної роботи

Харківський національний університет радіоелектроніки  
Кафедра ЕОМ  
Кваліфікаційна робота  
Перший(бакалаврський) рівень

## **Розробка веб застосунку для візуалізації даних з використанням баз даних часових рядів**

Здобувач  
Костянтин УТОЧКІН  
КІУКІ-21-5

Керівник:  
Яна НІ  
ст. викл. кафедри ЕОМ

1

## **Мета роботи**

---

- Розробка рішення для демонстрації переваг використання баз даних часових рядів при роботі з логуванням та даними різного обсягу
- Інтеграція системи візуалізації та моніторингу Grafana і Prometheus
- Реалізація візуалізації логів з Event Viewer

2

## База даних часових рядів

Розглядалась на прикладі Influx DB

Та інструменту Influxdata для управління базою даних

Швидкість запису за читання у баз даних часових рядів вищі завдяки структурованому та оптимізованому дизайну, розробленому спеціально для цієї мети.

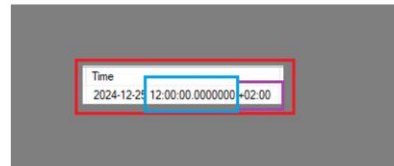
3

The image shows a screenshot of the InfluxDB web interface. On the left, there's a sidebar with navigation options. The main area displays a table with columns for 'Time', 'Value', and 'Error'. Below the table, there are two JSON snippets. The first is a log entry: `{ "message": "Prices section to InfluxDB v3", "count": 1877, "durationMs": 3886 }`. The second is a more detailed log entry: `{ "deleteTime": 5325, "insertTime": 4634, "resultMsg": "3386", "total": 3211, "count": 1878 }`. On the right, a REST client shows a POST request to `http://localhost:5278/api/v2/logsData/read-windows-logs` with a 'Key' parameter. The response body is a JSON object: `{ "message": "Event logs read and stored", "logCount": 32783, "deleteTimeSeconds": 6.2763, "totalWriteTimeSeconds": 3.5298358 }`.

## Основні поняття про часові ряди

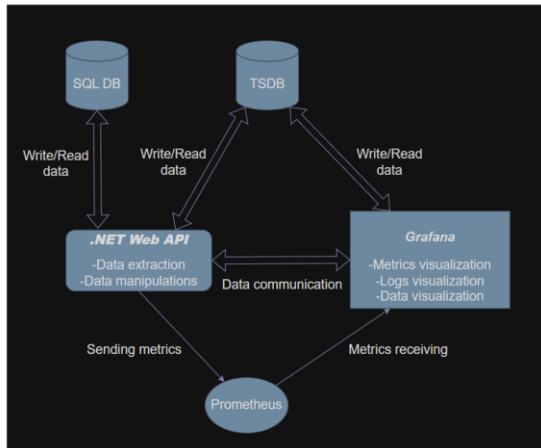
- Структура
- Зсув
- Точність
- Дельта часу

Time  
2024-12-25 12:00:00.000000000 +02:00



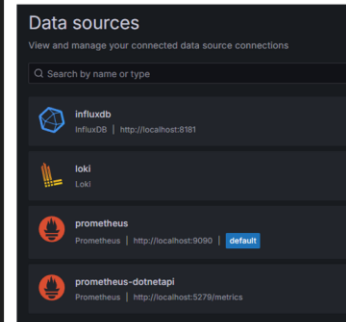
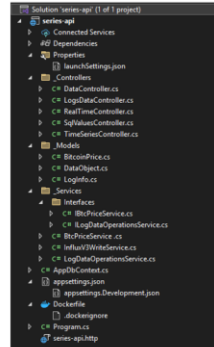
4

## Архітурне рішення



5

- Перевагою є наявність існуючої підтримки для інтеграції систем моніторингу у .NET
- Недоліком є загальне ускладнення системи але це дає можливість отримувати більше даних про застосунок і його роботу



## Приклади використання

ІоТ – Зміна параметрів пристроїв з часом

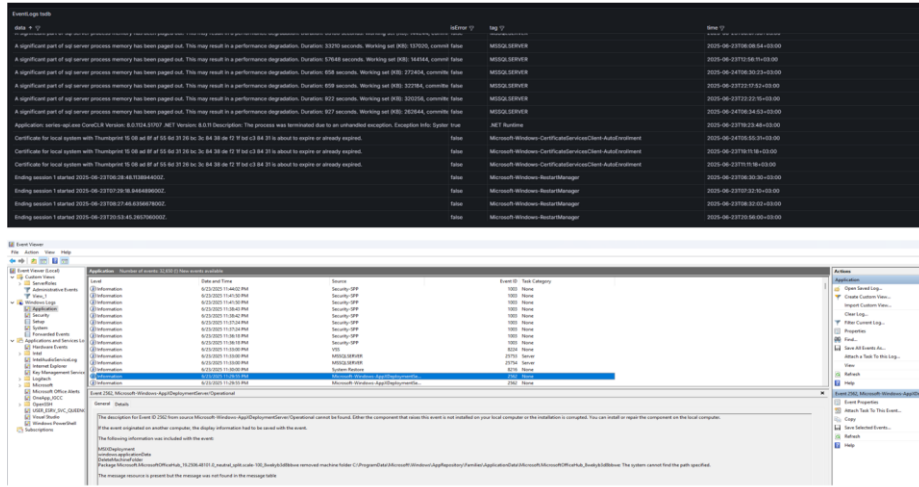
Фінанси – Історичні дані про ціни для майбутніх прогнозувань

Метрики – Інформація про роботу систем протягом часу

Логування – Збереження виняткових ситуацій та час їх виникнення для полегшення розробки і підтримки застосунків

6

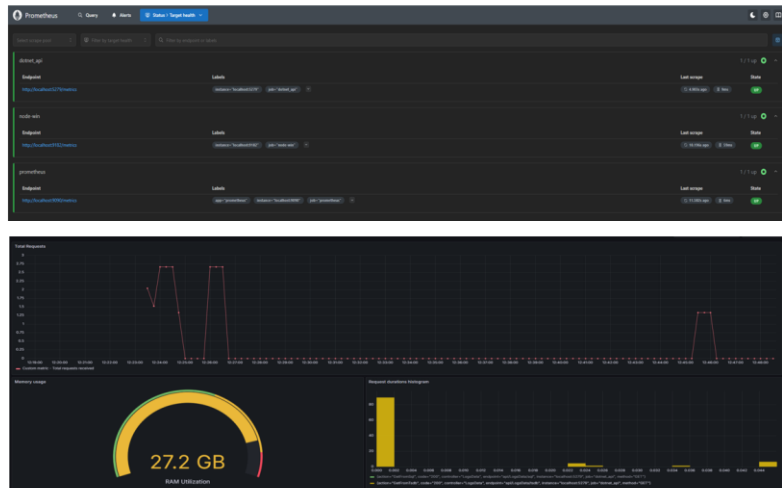
# Імпорт логів з Windows Events



Реалізовано імпортування логів з реалізацією фільтрування із зручним інтерфейсом у порівнянні з вбудованим

7

- Зручний REST-API спосіб взаємодії
- Широкий набір метрик
- Інтеграція з більшістю серверних фреймворків



# Система моніторингу Prometheus

8

## Висновки

---

Досягнуто мети і розроблено веб застосунок з інтеграцією сервісів для моніторингу і візуалізації даних що використовує одночасно класичну реляційну базу даних і базу даних часових рядів

Досліджено переваги та недоліки використання часових рядів при роботі з різними даними

Реалізовано візуалізацію логів з Event Viewer