

УДК 004.7+004.415.2

*Попко А. К., здобувач,
Ніколаєнко А. Ю., к.т.н., асистент
Київський національний університет імені Тараса Шевченка*

АРХІТЕКТУРА ПРОГРАМНОЇ СИСТЕМИ МОНІТОРИНГУ ПРОМИСЛОВОГО ОБЛАДНАННЯ

Вступ. Сучасні підприємства експлуатують обладнання від різних виробників із власними закритими системами моніторингу. Наслідком цього є фрагментація даних, що ускладнює комплексний аналіз виробничих процесів. Крім того, більшість наявних рішень функціонує реактивно, фіксуючи лише факт несправності. Такий підхід унеможливило запобігання аваріям і призводить до непередбачуваних простоїв та фінансових втрат.

Метою роботи є проєктування архітектури універсальної програмної системи моніторингу, яка забезпечить централізовану агрегацію телеметрії з різних виробничих вузлів і використання предиктивної аналітики для завчасного виявлення аномалій та ймовірних відмов [1].

Архітектура запропонованої ІІоТ-системи базується на розділенні потоків обробки даних (рис. 1). Високочастотна телеметрія від сенсорів надходить через брокер повідомлень (Mosquitto) за протоколом MQTT до бекенду (ASP.NET Core API). Оскільки синхронний запис посекундних даних з багатьох вузлів генерує критичне навантаження на базу даних (PostgreSQL), застосовано патерн пакетної буферизації. Дані тимчасово накопичуються в пам'яті сервера та записуються об'ємними транзакціями, що суттєво підвищує пропускну здатність системи.

Проте такий підхід створює затримку, неприпустиму для критичних інцидентів, коли параметри виходять за межі норми. Для нівелювання цього недоліку паралельно працює подієво-орієнтований маршрут (Fast Path). У разі фіксації аномалії система ігнорує буфер, миттєво записує події в базу та ініціює Push-сповіщення на фронтенд (Angular/React) через SignalR з нульовою затримкою реакції.

Накопичений масив даних використовується для предиктивної аналітики (Slow Path) [2]. Оскільки використання необробленої телеметрії містить багато шуму і призводить до перенавчання моделей, система попередньо агрегує дані за допомогою ковзних часових вікон. Замість абсолютних значень обчислюються статистичні ознаки (дисперсія, тренди, ковзні середні), які надають можливість модулю ML.NET розпізнавати поступову фізичну деградацію механізмів.

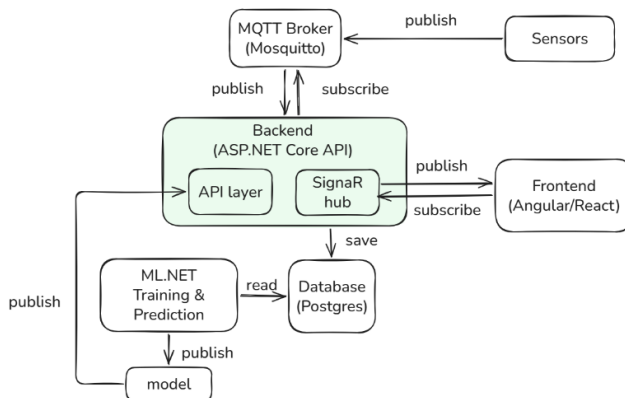


Рис. 1. Структурна схема маршрутизації та обробки даних

Для класифікації ймовірності відмов вибрано алгоритм градієнтного бустингу (LightGBM) [3]. Щоб інженери не лише бачили факт можливої поломки, а й розуміли її причину, впроваджено концепцію Explainable AI. Модель розраховує внесок кожної ознаки у фінальний прогноз, і ці метадані зберігаються в PostgreSQL у неструктурованому форматі JSONB. Така денормалізація усуває потребу в складних зв'язках (JOIN) на рівні бази даних та дає змогу гнучко відображати кореневі причини аномалій в інтерфейсі.

Висновки. Створений прототип IoT-системи вирішує проблему консолідації телеметрії з різноманітного обладнання. Гібридна архітектура з розділенням потоків (Fast/ Slow Path) забезпечує баланс між пропускну здатністю та оперативністю критичних сповіщень. Інтеграція машинного навчання з Explainable AI формує інтерпретовані прогнози відмов, що створює надійне підґрунтя для переходу підприємств до предиктивного обслуговування.

Список використаних джерел:

1. Zonta T., da Costa C. A., da Rosa Righi R., De Lima M. J., Da Trindade E. S., Li G. P. Predictive maintenance in the Industry 4.0: A systematic literature review. *Computers & industrial engineering*. 2020. Vol. 150:106889. <https://doi.org/10.1016/j.cie.2020.106889>.
2. Mihigo I. N., Zennaro M., Uwitonze A., Rwigema J., Rovai M. On-Device IoT-Based Predictive Maintenance Analytics Model: Comparing TinyLSTM and TinyModel from Edge Impulse. *Sensors*, 2022. Vol. 22, No 14:5174. <https://doi.org/10.3390/s22145174>.
3. LightGBM. URL: <https://github.com/microsoft/LightGBM> (дата звернення: 03.03.2026 р.)