

Я, Безсонова Дар`я Юріївна, як здобувачка вищої освіти ХНУРЕ, розумію і підтримую політику закладу із академічної доброчесності. Я не надавала і не одержувала недозволену допомогу під час підготовки кваліфікаційної роботи. Я не використовувала штучний інтелект для підготовки кваліфікаційної роботи. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело.

"07" червня 2025 р



Дар`я БЕЗСОНОВА

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

Факультет _____ АКТ _____
Кафедра _____ КІТАР _____
Рівень вищої освіти _____ перший (бакалаврський) _____
Спеціальність 151 Автоматизація та комп'ютерно-інтегровані технології
(код і повна назва)
Тип програми _____ Освітньо-професійна _____
Освітня програма _____ Системна інженерія _____
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____ (підпис)

« 28 » квітня 2025 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

Здобувачеві _____ Безсоновій Дар'ї Юріївні _____
(прізвище, ім'я, по батькові)

1. Тема роботи Розроблення web-додатку для моніторингу роботизованої виробничої ділянки

затверджена наказом по університету від 19.05.2025 р. № 391 Ст

2. Термін подання здобувачем роботи до екзаменаційної комісії 15.06.2025 р.

3. Вихідні дані до роботи Intel Core i5-10400 або AMD Ryzen 5 3600; тактова частота від 3.6 GHz, 6 ядер / 12 потоків; оперативна пам'ять: 16 GB RAM, DDR4 - 3200 MHz; відеокарта: NVIDIA GTX 1660 або AMD RX 580, 4 GB відеопам'яті, апаратне декодування 4K відео.

4. Перелік питань, що потрібно опрацювати в роботі _____
Аналіз предметної області; Постановка задачі; Створення моделей та візуальних схем для системи моніторингу роботизованої виробничої ділянки; Створення програмного забезпечення для системи моніторингу роботизованої виробничої ділянки; Засоби розробки та мова програмування; Розробка бази даних для системи; Розробка SQL-запитів для сервісної частини системи; Висновки

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій

Демонстраційний матеріал, представлений у форматі презентації
PowerPoint (*.ppt) 12 с. формату А4

6. Консультанти розділів роботи (п.6 включається до завдання за наявності консультантів згідно з наказом, зазначеним у п.1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Аналіз предметної області	28.04-04.05.2025	виконано
2	Постановка задачі	05.05-10.05.2025	виконано
3	Створення моделей та візуальних схем для системи моніторингу роботизованої виробничої ділянки	11.05-17.05.2025	виконано
4	Створення програмного забезпечення для системи моніторингу роботизованої виробничої ділянки	18.05-25.05.2025	виконано
5	Розробка бази даних для системи. Розробка SQL-запитів для сервісної частини системи	26.05-31.05.2025	виконано
6	Оформлення пояснювальної записки	04.06-11.06.2025	виконано

Дата видачі завдання 28 квітня 2025 р.

Здобувач _____ Дар'я БЕЗСОНОВА
(підпис) (власне ім'я, прізвище)

Керівник роботи _____ доц. Наталія ДЕМСЬКА
(підпис) (посада, власне ім'я, прізвище)

РЕФЕРАТ

Пояснювальна записка: 67 с., 16 рис., 2 дод., 21 джерело.

РОБОТИЗОВАНА ВИРОБНИЧА ДІЛЯНКА, ВЕБДОДАТОК, ХМАРНА СИСТЕМА ВІДЕОСПОСТЕРЕЖЕННЯ, МОНІТОРИНГ, СЕРВЕР, БАЗА ДАНИХ, КОРИСТУВАЧ, RUBY ON RAILS, POSTGRESQL.

Об'єкт розробки – процес моніторингу роботизованої виробничої ділянки.

Предмет розробки – інформаційні технології для розробки компонентів інформаційної системи моніторингу роботизованої виробничої ділянки у складі бази даних та вебінтерфейсу доступу до неї з використанням хмарної системи відеоспостереження.

Метою даної роботи є розробка концепції веб-додатку для моніторингу роботизованої виробничої ділянки, інтегрованого з хмарною системою відеоспостереження.

Методи дослідження – методи логічного та системного аналізу, методи структурного моделювання реляційних баз даних, об'єктно-орієнтованого програмування на мові Ruby та використанні фреймворку Ruby on Rails, методи проектування інтерфейсу користувача.

Результат роботи – розроблена концепція веб-застосунку, що складається з моделі бази даних, опису серверної частини та принципів взаємодії з хмарною системою відеоспостереження.

Проведені дослідження відповідають цілям сталого розвитку (ЦСР) 9, так як наукові дослідження та інновації є важливими факторами, що сприяють сталому розвитку.

ABSTRACT

Explanatory note: 67 pages, 16 pictures, 2 application, 21 sources.

ROBOTIC PRODUCTION AREA, WEB APPLICATION, CLOUD VIDEO SURVEILLANCE SYSTEM, MONITORING, SERVER, DATABASE, USER, RUBY ON RAILS, POSTGRESQL.

The object of the development is the process of monitoring a robotic production area.

The subject of the development is information technologies for the development of components of the information system for monitoring a robotic production area as part of a database and a web interface for accessing it using a cloud video surveillance system.

The purpose of the work is to develop a functional, effective and convenient monitoring system that will provide operational control over the operation of a robotic production area, contribute to timely response to emergency situations and improve the analysis of production processes.

Research methods – methods of logical and system analysis, methods of structural modeling of relational databases, object-oriented programming in the Ruby language and using the Ruby on Rails framework, methods of user interface design.

The result of the work – a developed concept of a web application consisting of a database model, a description of the server part and principles of interaction with a cloud video surveillance system.

The research conducted is in line with Sustainable Development Goals (SDGs) 9, as scientific research and innovation are important factors contributing to sustainable development.

ЗМІСТ

Перелік скорочень	6
Вступ	7
1 Аналіз предметної області	9
1.1 Хмарне відеоспостереження	10
1.1.1 Переваги та недоліки хмарного відеоспостереження	13
1.2 Апаратне (локальне) відеоспостереження	16
1.3 Архітектура системи відеоспостереження	19
1.4 Архітектура системи хмарного відеоспостереження	22
1.5 Постановка задачі	25
2 Створення моделей та візуальних схем для системи моніторингу роботизованої виробничої ділянки	27
2.1 Діаграма прецедентів для моделювання взаємодії користувача з системою	27
2.2 Модель предметної області	29
2.3 Діаграма класів	32
2.4 Діаграма послідовностей	36
3 Створення програмного забезпечення для системи моніторингу роботизованої виробничої ділянки	38
3.1 Засоби розробки та мова програмування	38
3.2 Розробка бази даних для системи	40
3.3 Розробка SQL-запитів для сервісної частини системи	43
3.4 Розробка SQL-запитів для сервісної частини системи	45
3.5 Забезпечення безпечних умов праці при розробленні web-додатку	49
Висновки	52
Перелік джерел посилання	54
Додаток А Код програми	57
Додаток Б Демонстраційний матеріал	64

ПЕРЕЛІК СКОРОЧЕНЬ

БД – бази даних;

РАВД – роботизовані виробничі ділянки;

РВД – роботизовані виробничі ділянки;

API – Application Programming Interface;

ERP – Enterprise Resource Planning;

IDE – Integrated Development Environment;

MES – Manufacturing Execution System;

NVR – Network Video Recorder;

PoE – Power over Ethernet;

RMS – Remote Manipulator System;

RoR – Ruby on Rails;

RPA – Robotic Process Automation;

SCADA – Supervisory Control and Data Acquisition;

VMS – Video Management System;

VS – Code Visual Studio Code.

ВСТУП

В умовах стрімкого розвитку промисловості та зростання рівня автоматизації виробничих процесів, особливого значення набуває ефективний моніторинг та контроль за роботою складних технологічних систем, таких як роботизовані виробничі ділянки (РВД). Впровадження робототехніки дозволяє підвищити продуктивність, знизити витрати та покращити якість продукції, проте ефективне управління такими ділянками вимагає сучасних інструментів для візуалізації, аналізу та оперативного реагування на події, що відбуваються [1].

Традиційні підходи до моніторингу, що базуються на локальних системах відеоспостереження та ручному зборі даних, часто є недостатньо гнучкими, інформативними та оперативними. Існує нагальна потреба у створенні інтегрованих систем, які б поєднували можливості візуального контролю з даними про стан обладнання та виробничі процеси, забезпечуючи централізований доступ до інформації для різних категорій користувачів.

У цьому контексті перспективним напрямком є використання хмарних технологій відеоспостереження, які пропонують значні переваги щодо масштабованості, доступності та зниження витрат на інфраструктуру. Інтеграція таких систем з веб-додатками відкриває нові можливості для створення зручних та функціональних інструментів моніторингу, доступних з будь-якого підключеного до мережі пристрою [1].

Актуальність теми дослідження визначається зростаючою потребою підприємств у ефективних засобах контролю та управління роботизованими виробничими ділянками, а також потенціалом використання хмарних технологій для створення гнучких та масштабованих систем моніторингу.

Метою даної роботи є розробка концепції веб-додатку для моніторингу роботизованої виробничої ділянки, інтегрованого з хмарною системою відеоспостереження.

Для досягнення цієї мети були поставлені наступні завдання:

- провести аналіз предметної області моніторингу роботизованих виробничих ділянок та існуючих рішень;
- визначити функціональні та нефункціональні вимоги до веб-додатку;
- створити моделі та візуальні схеми системи моніторингу, включаючи діаграму прецедентів, модель предметної області та діаграму класів;
- розробити структуру бази даних для зберігання інформації, необхідної для функціонування веб-додатку;
- визначити засоби розробки та мову програмування для створення сервісної частини системи;
- розробити основні SQL-запити для забезпечення взаємодії сервісної частини з базою даних.

Об'єктом розробки є процес моніторингу роботизованої виробничої ділянки.

Предметом розробки є інформаційні технології для розробки компонентів інформаційної системи моніторингу роботизованої виробничої ділянки у складі бази даних та вебінтерфейсу доступу до неї з використанням хмарної системи відеоспостереження.

Практичне значення роботи полягає у розробці концепції веб-додатку, який може слугувати основою для створення реальної системи моніторингу, що дозволить підвищити ефективність управління роботизованими виробничими процесами на підприємствах.

Кваліфікаційну роботу виконано згідно вимог ДСТУ 3008 – 15 [2] та керуючись положенням про протидію плагіату [3] і методичними вказівками з написання кваліфікаційної роботи бакалавра [4].

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

Промисловий розвиток характеризується поступовим підвищенням ефективності, продуктивності та результативності виробничих процесів. Одним із ключових напрямків досягнення цих цілей є розвиток роботизованих систем. Роботизовані виробничі ділянки (РАВД) автоматизують рутинні, трудомісткі та небезпечні завдання, зменшуючи невідповідність продукції та мінімізуючи шкоду, спричинену людиною [1], [5].

Однак ефективне функціонування РВД залежить не лише від належного впровадження, а й від добре налагодженого моніторингу та контролю. Зі зростанням складності промислових процесів та збільшенням кількості роботизованих пристроїв, традиційні підходи до управління можуть здаватися неефективними та неадекватними з точки зору забезпечення інформацією. Отже, існує потреба в розробці рутинних інструментів, які можуть відображати ключові показники роботи РВД у режимі реального часу та дозволяти своєчасне виявлення та прийняття обґрунтованих управлінських рішень.

Веб-додаток – це перспективний спосіб створення таких інструментів моніторингу. Веб-додатки можуть забезпечити пасивний та ефективний інтерфейс для візуалізації інформації про обладнання роботизованих виробничих ділянок, продуктивність володіння, стан закупівель та інші важливі параметри завдяки своїй кросплатформності, доступності до Інтернету та можливості інтеграції з різними джерелами даних (рис. 1.1).

У цьому контексті розробка веб-додатків для моніторингу роботизованих ділянок з використанням камер, має життєво важливе та практичне значення для компаній, що спеціалізуються на робототехнічних технологіях або планують їх впроваджувати. Ці додаткові можливості дозволяють покращити прозорість ваших виробничих процесів, швидко реагувати на надзвичайні ситуації, оптимізувати доступні ресурси та зрештою

підвищити загальну ефективність виробництва.

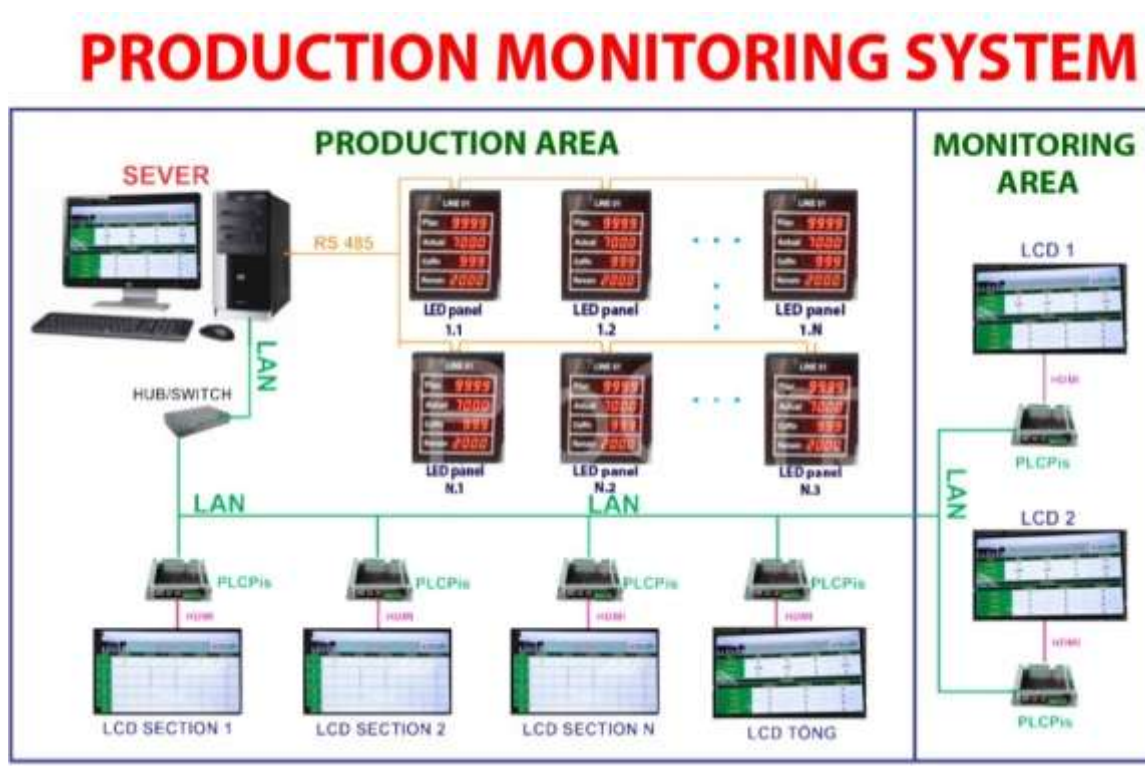


Рисунок 1.1 – Будова системи моніторингу виробничої ділянки [6]

1.1 Хмарне відеоспостереження

В епоху цифрової трансформації в промисловості ефективний моніторинг виробничих процесів стає ключовим фактором успіху компанії. РВД, як високотехнологічні комплекси, потребують особливої уваги, коли йдеться про управління та моніторинг. У цьому контексті хмарне відеоспостереження служить не лише заходом безпеки, а й потужним інструментом для комплексного моніторингу та оптимізації роботи RPS [7].

Традиційні системи відеоспостереження, що базуються на локальній інфраструктурі, часто стикаються з низкою обмежень. Необхідність придбання та обслуговування дорогого обладнання для зберігання та обробки відео, складність масштабування при розширенні виробництва, а також обмежений доступ до відеоархівів – все це може ускладнити ефективний моніторинг.

Хмарне відеоспостереження пропонує якісно новий спосіб організації системи відеоконтролю. Переносючи основні функції зберігання, обробки та аналізу відеоданих на віддалені сервери у хмарного провайдера, відкриваються великі можливості для оптимізації моніторингу RPS.

Одна з найважливіших переваг хмарного відеоспостереження – його масштабованість. Компанія може легко збільшити або зменшити кількість підключених камер та обсяг виділеного сховища залежно від своїх поточних потреб. Це особливо важливо для динамічно розвиваючихся галузей, яким може знадобитися розширення робототехнічних зон.

Доступність – ще одна важлива перевага. Завдяки хмарній інфраструктурі, авторизовані користувачі можуть отримувати доступ до відео в режимі реального часу та переглядати архівні записи з будь-якого пристрою, підключеного до Інтернету. Це забезпечує оперативний контроль над виробничими процесами незалежно від того, де знаходяться відповідальні особи, що особливо цінно у зв'язку з глобалізацією та необхідністю дистанційного управління [7].

Надійність хмарних сервісів зазвичай на високому рівні. Постачальники хмарних послуг значно інвестують у резервне копіювання даних, забезпечуючи безперервну роботу серверів та захист від кібератак. Це забезпечує збереження важливих відеоархівів та постійну доступність системи спостереження.

Можливість інтеграції з іншими хмарними сервісами відкриває нові перспективи для аналізу виробничих даних. Наприклад, інтеграція із системами штучного інтелекту може забезпечити автоматичне виявлення аномалій у роботі обладнання, порушень правил безпеки або дефектів продукції, що значно підвищить ефективність моніторингу та швидкість реагування на надзвичайні ситуації.

З точки зору економічної ефективності [8], хмарне відеоспостереження може бути більш прибутковим, ніж традиційні системи. Відсутність значних капітальних витрат на придбання та встановлення локального обладнання, а

також зниження витрат на обслуговування та споживання енергії роблять хмарні рішення привабливими для компаній будь-якого розміру.

Застосування хмарного відеоспостереження (рис. 1.2) у зв'язку з робототехнічними виробничими зонами є різноманітним:

- моніторинг продуктивності роботів: візуальний огляд роботів, що виконують запрограмовані операції, виявлення порушень, помилок у роботі маніпуляторів або інших відхилень від норми;

- моніторинг дотримання вимог безпеки: моніторинг відсутності перебування персоналу в небезпечних зонах під час роботи робота, контроль використання засобів індивідуального захисту;

- аналіз ефективності виробничого процесу: запис та аналіз відеопотоків для виявлення вузьких місць у виробничому циклі, оптимізація шляхів переміщення деталей та визначення часу простою непродуктивного обладнання;

- контроль якості продукції: візуальний огляд якості виконуваних операцій, виявлення будь-яких дефектів на ранніх стадіях виробництва;

- розслідування інцидентів: надання відеодоказів у разі надзвичайних ситуацій, аварій або порушень процесу.

Traditional CCTV



Cloud CCTV



Рисунок 1.2 – Будова системи Cloud CCTV [9]

Вибираючи постачальника хмарних послуг відеоспостереження для РВД, необхідно враховувати ряд важливих критеріїв [9]:

- якість відеозображення: камери високої роздільної здатності та потокове відео високої якості забезпечують чітке уявлення про деталі виробничого процесу;

- пропускну здатність мережі: стабільне та швидке інтернет-з'єднання для передачі великих обсягів відеоданих без затримок;

- місткість хмарного сховища: достатня ємність для зберігання відеоархіву необхідної глибини, враховуючи інтенсивність виробничого процесу;

- функціональність платформи: наявність інтелектуальних функцій, таких як розпізнавання об'єктів, виявлення руху, аналіз поведінки, створення теплових карт активності;

- безпека даних: забезпечення конфіденційності та цілісності відеоданих за допомогою сучасних методів шифрування та контролю доступу;

- інтеграція з іншими системами: можливість інтеграції з існуючими системами управління виробництвом (MES), системами планування ресурсів підприємства (ERP) та іншими інструментами моніторингу;

- вартість обслуговування: оцінка загальної вартості володіння системою з урахуванням тарифних планів зберігання даних, кількості.

1.1.1 Переваги та недоліки хмарного відеоспостереження

Ця технологія пропонує низку суттєвих переваг, які можуть значно оптимізувати контроль над виробничими процесами, підвищити безпеку та сприяти загальній ефективності компанії. Водночас, як і всі технологічні рішення, хмарне відеоспостереження має певні недоліки, які необхідно враховувати під час його впровадження [10].

Серед найважливіших переваг хмарного відеоспостереження варто виділити, перш за все, його масштабованість та гнучкість. У мінливому виробничому середовищі, де може виникнути потреба в розширенні або

оптимізації роботизованих ліній, можливість легко додавати нові камери або збільшувати ємність сховища без значних капіталовкладень у локальну інфраструктуру є надзвичайно цінною. Хмарні рішення дозволяють компанії адаптувати систему відеоспостереження до своїх поточних потреб та платити лише за фактично спожиті ресурси.

Ще однією важливою перевагою є доступність та мобільність. Завдяки хмарній інфраструктурі, авторизовані користувачі можуть контролювати роботу системи відеоспостереження в режимі реального часу та переглядати архівні записи з будь-якого пристрою, підключеного до Інтернету. Це забезпечує оперативний контроль з будь-якої точки світу, що особливо важливо для керівників виробництва, інженерів та технічного персоналу, яким може знадобитися віддалений доступ до інформації про стан виробничого об'єкта [10].

Зменшення початкових витрат також є значною перевагою хмарного відеоспостереження. Відсутність необхідності купувати та встановлювати дороге локальне обладнання, таке як сервери та системи зберігання даних, дозволяє значно зменшити початкові інвестиції. Модель підписки, яка часто використовується для хмарних сервісів, означає, що ви платите лише за фактичне використання ресурсів, що робить цю технологію економічно привабливою, особливо для малого та середнього бізнесу, який впроваджує робототехніку.

Не менш важливою є надійність та безпека даних, що надаються постачальниками хмарних послуг. Вони значні кошти інвестують у створення надійної інфраструктури з резервуванням даних, захистом від збоїв обладнання та сучасними інструментами кібербезпеки. Зберігання відеоархівів у географічно розподілених центрах обробки даних мінімізує ризик втрати важливої інформації [11].

Легке обслуговування та оновлення – ще одна перевага хмарних рішень. Обслуговування системи, оновлення програмного забезпечення та управління інфраструктурою повністю лягають на плечі постачальника хмарних послуг.

Це звільняє ІТ-персонал компанії від рутинних завдань та дозволяє їм зосередитися на більш стратегічних питаннях, пов'язаних з автоматизацією виробництва.

Зрештою, можливість інтеграції з іншими хмарними сервісами відкриває нові перспективи для аналізу та оптимізації виробничих процесів. Інтегруючи відеодані зі штучним інтелектом, платформами Інтернету речей або іншими аналітичними інструментами, ви можете отримати глибоке розуміння операцій досліджень та розробок, виявити закономірності, передбачити потенційні проблеми та приймати обґрунтовані управлінські рішення.

Але окрім багатьох переваг, хмарне відеоспостереження має й деякі недоліки, які необхідно враховувати. Одна з найважливіших – це залежність від стабільного інтернет-з'єднання. Для передачі великих обсягів відеоданих у хмару та забезпечення безперебійного доступу до них потрібне надійне та швидке інтернет-з'єднання. Збої в мережі можуть призвести до втрати моніторингу в режимі реального часу та доступу до архівних записів, що може бути критично важливим для безперервності виробничого процесу.

Також існують потенційні витрати на передачу та зберігання даних. Залежно від обраного тарифного плану хмарного провайдера, велика кількість камер високої роздільної здатності, що генерують значний обсяг трафіку, може призвести до значних щомісячних витрат. Тому необхідно ретельно аналізувати тарифні плани та прогнозувати обсяг споживаних ресурсів при виборі хмарного рішення [12].

Питання безпеки даних та конфіденційності також потребують особливої уваги. Хоча хмарні провайдери вживають серйозних заходів для захисту інформації, завжди існує певний теоретичний ризик несанкціонованого доступу або витоку даних. Компанія повинна ретельно вибрати надійного провайдера та перевірити, чи відповідає його політика безпеки внутрішнім вимогам та стандартам.

У деяких випадках може виникати затримка відеопотоку (латентність)

через Інтернет. Для більшості завдань відеоспостереження RVD невелика затримка може не бути критичною, але в ситуаціях, що вимагають негайного реагування на події, це може бути проблемою.

Крім того, існує певна залежність від обраного постачальника послуг. Перехід до іншого хмарного провайдера може бути складним і передбачати передачу великих обсягів даних. Тому важливо ретельно оцінити надійність та довгострокові перспективи потенційних провайдерів, перш ніж приймати рішення [13].

Нарешті, варто зазначити, що обмежена функціональність базових тарифних планів може бути недоліком для компаній, яким потрібні розширені функції, такі як інтелектуальна відеоаналітика. Такі функції часто доступні лише в дорожчих підписках, що може збільшити загальну вартість володіння системою.

На завершення, хмарне відеоспостереження – це потужний інструмент для моніторингу роботизованих виробничих зон, що має значні переваги з точки зору масштабованості, доступності, вартості та надійності. Однак, під час його впровадження необхідно враховувати потенційні недоліки, такі як залежність від підключення до Інтернету, витрати на передачу та зберігання даних, проблеми безпеки та можливі функціональні обмеження. Ретельний аналіз потреб компанії та збалансований підхід до вибору постачальника послуг дозволять вам скористатися перевагами хмарного відеоспостереження для підвищення ефективності та безпеки роботизованого обладнання.

1.2 Апаратне (локальне) відеоспостереження

У контексті забезпечення надійного та ефективного моніторингу роботизованих виробничих зон (RPA), поряд із хмарними рішеннями, існують також більш традиційні методи, засновані на локальній (апаратній) інфраструктурі відеоспостереження. Ці системи передбачають розгортання власного набору обладнання для відеозапису, зберігання та управління

відеоданими без залучення зовнішніх хмарних сервісів.

Основними компонентами типової локальної системи відеоспостереження для RPA є IP-камери, які відповідають за безпосередній збір візуальної інформації з виробничих зон. Ці камери зазвичай підключені до центрального пристрою обробки та зберігання даних – мережевого відеореєстратора (NVR). NVR – це спеціалізований апаратний комплекс, оснащений потужним процесором та значним обсягом дискового простору, призначений для одночасного запису відеопотоків з багатьох IP-камер, їх зберігання протягом певного періоду часу та забезпечення можливості відтворення архівних записів.

Для фізичного підключення IP-камер та відеореєстраторів до єдиної локальної мережі використовується мережеве обладнання, таке як комутатори. У випадках, коли необхідно подавати живлення до камер через мережеві кабелі, можна використовувати комутатори PoE (Power over Ethernet), що спрощує встановлення та зменшує кількість необхідних кабелів. До системи підключаються один або декілька моніторів для безпосереднього перегляду відео в реальному часі та роботи з архівними записами [14].

Ключову роль в управлінні локальною системою відеоспостереження відіграє програмне забезпечення для управління відео (VMS). Це програмне забезпечення можна інтегрувати безпосередньо в NVR або встановити на окремому комп'ютері, підключеному до локальної мережі. VMS надає користувачам інструменти для налаштування параметрів запису, перегляду відео в реальному часі, пошуку та відтворення архівованих записів, керування правами доступу користувачів і використання функцій відеоаналітики, якщо вони підтримуються камерами та VMS.

Однією з головних запропонованих локальних (апаратних) систем відеоспостереження є повний контроль над даними. Оскільки всі відеозаписи зберігаються на вашому власному пристрої, компанія має повну автономію з точки зору безпеки та конфіденційності.

Оскільки вся обробка та зберігання відео відбувається в локальній

мережі, надійність системи не залежить від якості та стабільності зовнішнього інтернет-каналу. Це особливо важливо для промислових об'єктів, які можуть мати проблеми з інтернет-з'єднанням. Крім того, немає щомісячних платежів за підписку або зберігання даних, що може призвести до зниження загальної вартості володіння системою в довгостроковій перспективі, незважаючи на значні початкові інвестиції.

Локальні системи також забезпечують високу пропускну здатність для локального доступу. Перегляд відео в локальній мережі зазвичай відбувається швидко та без затримок, оскільки він не обмежений швидкістю інтернет-з'єднання. Крім того, такі системи часто мають кращі можливості для інтеграції зі складними промисловими системами безпеки та автоматизації, використовуючи стандартні промислові протоколи та інтерфейси.

Однак локальні (апаратні) системи відеоспостереження мають свої недоліки. Одним з одного є дуже велика відповідальність за технічне обслуговування всього обладнання та програмного забезпечення, включаючи камери, відеореєстратори, жорсткі диски, мережеве обладнання та монітори. Крім того, власник системи несе повну відповідальність за обслуговування та підтримку всього обладнання та програмного забезпечення, включаючи усунення несправностей та оновлення [12].

Складність масштабу також може бути недоліком локальних систем. Додавання нових камер або збільшення обсягу сховища може вимагати придбання додаткового обладнання та налаштувань сховища. Існує також ризик втрати даних у разі фізичного пошкодження обладнання, такого як вихід з ладу відеореєстратора або жорстких дисків, а також у разі пожежі чи крадіжки. Заблоковані резервні копії даних є постачанням системи. Нарешті, віддалений доступ до відео може бути складнішим у налаштуванні та залежить від надійності інтернет-з'єднання об'єкта, а також може вимагати додаткових налаштувань безпеки мережі.

Таким чином, вибір між локальною (апаратною) та хмарною системою відеоспостереження для моніторингу роботизованого виробничого

майданчика залежить від багатьох факторів, включаючи бюджет, вимоги до безпеки даних, надійність інтернет-з'єднання, необхідність віддаленого доступу та наявність кваліфікованого персоналу для обслуговування системи. Локальні системи забезпечують більший контроль та незалежність, але вимагають значних початкових інвестицій та незалежного обслуговування.

1.3 Архітектура системи відеоспостереження

В основі локальної системи відеоспостереження, призначеної для моніторингу встановлених виробничих зон (RPA), лежать IP-камери, стратегічно розміщені в ключових частинах виробничого процесу. Ці цифрові камери, кожна з яких має унікальну IP-адресу, підключаються до локальної мережі компанії за допомогою надійних кабелів Ethernet, часто з використанням технології Power over Ethernet (PoE) для одночасної передачі живлення та даних, або через бездротове Wi-Fi з'єднання. Завдання IP-камери полягає в безперервному захопленні високоякісних відеозображень, фіксуючи кожен крок роботизованих маніпуляторів, переміщення деталей на конвеєрних стрічках, взаємодію обладнання та дотримання стандартів безпеки персоналом. Кількість та розташування цих камер ретельно плануються, виходячи з розміру виробничої зони, складності технологічних процесів та необхідності загального візуального контролю [15].

Всі відеопотоки, що генеруються IP-камерами, надсилаються до центрального вузла системи – мережевого відеореєстратора (NVR). Це спеціалізоване обладнання є мозковою системою, що відповідає за безперервний запис відеоданих на вбудовані жорсткі диски значної ємності, забезпечуючи збереження відеоархіву протягом певного періоду. NVR також надає базовий інтерфейс для локального перегляду відео в реальному часі, часто у вигляді мозаїки з усіх підключених камер, а також інструменти для пошуку та відтворення архівних записів через підключений монітор. Для забезпечення надійної передачі даних між кількома IP-камерами та NVR

використовується локальна мережа, ключовими елементами якої є мережеві комутатори (свічі), що забезпечують ефективну маршрутизацію трафіку. У більш складних системах керовані комутатори можуть використовуватися для оптимізації продуктивності мережі та налаштування параметрів відеопотоків.

У деяких випадках, особливо у великих виробничих компаніях з розгалуженою системою відеоспостереження, може використовуватися окремий сервер або потужна робоча станція з встановленим програмним забезпеченням для керування відео (VMS). VMS розширює функціональність системи, забезпечуючи централізоване керування великою кількістю камер та NVR, розширену відеоаналітику (наприклад, розпізнавання об'єктів, виявлення аномалій), інтеграцію з іншими системами безпеки та автоматизації, а також додаткові функції доступу клієнтів через спеціалізовані клієнтські інструменти або веб-інтерфейси, проте часто обмежуються локальною мережею [16].

Хоча локальні системи відеоспостереження [16] можуть мати низьку надійність та функціональність, вони можуть мати кілька проблем, особливо у зв'язку з сучасними вимогами до моніторингу виробничих процесів. Одна з основних проблем – обмежений локальний доступ до відеоданих. Для перегляду відео в реальному часі або архівних записів зазвичай потрібно знаходитися поруч із відеореєстратором та використовувати підключений монітор або локальний клієнтський додаток VMS. Це створює незручності для менеджерів, інженерів та технічного персоналу, які можуть отримати миттєвий доступ до видимої інформації про стан відеореєстратора з різних місць у компанії або навіть віддалено.

Ще однією суттєвою проблемою є надійність інтеграції локальних систем з іншими критично пошкодженими виробничими системами, такими як системи управління виробництвом (MES), системи управління технологічними процесами (SCADA) або системи моніторингу стану обладнання. Така ізоляція ускладнює отримання цілісної та всебічної картини роботи VMS, що обмежує можливості поглибленого аналізу та прийняття

обґрунтованих управлінських рішень на основі складних даних.

Інтерфейси користувача, що надаються відеореєстраторами або локальними VMS, часто незручні та важкі для освоєння, особливо для користувачів, які не мають спеціалізованих знань у системах відеоспостереження. Це може ускладнити швидке отримання необхідної інформації та знизити ефективність використання системи.

У випадку компаній, що працюють у кількох виробничих зонах, кожна з яких оснащена власною локальною системою відеоспостереження, виникає проблема створення централізованого моніторингу. Операторам та менеджерам доводиться працювати з кількома різними системами, що ускладнює загальний контроль та координацію виробничих процесів.

Крім того, аналітичні функції, вбудовані в локальні відеореєстратори або системи управління відеоспостереженням, можуть бути обмеженими або вимагати складного налаштування, яке не завжди доступне звичайним користувачам. Це поєднує потенціал системи для введення закономірностей, прогнозування проблем або автоматичного реагування на наступну ситуацію.

Нарешті, налаштування та генерування оперативних сповіщень про важливі події, що відбуваються в РВД (наприклад, виявлення руху в забороненій зоні, вимкнення обладнання) з локальної системи може бути недостатнім і вимагати складного налаштування, часто обмеженого лише локальними каналами сповіщень.

Розробка веб-додатку для моніторингу роботизованого виробничого об'єкта відкриває значні можливості для вирішення цих проблем шляхом інтеграції з існуючими локальними системами відеоспостереження [17]. Веб-додаток може забезпечити зручний та інтуїтивно зрозумілий віддалений доступ до потоків відео в реальному часі та архівних записів через стандартний браузер з будь-якого пристрою, підключеного до мережі. Завдяки розробленому API, веб-додаток може стати місцем для інтеграції даних відеоспостереження з іншими виробничими системами, надаючи цілісну картину роботи РВД. Сучасний та ергономічний веб-інтерфейс значно

спростить роботу з відеоданими для широкого кола користувачів. Веб-додаток може бути розроблений для централізованого моніторингу кількох незалежних локальних систем, забезпечуючи єдину точку керування всіма системами опалення, вентиляції та кондиціонування повітря компанії. Використовуючи можливості серверної частини веб-додатку, можна реалізувати розширені функції відеоаналітики та відображати результати аналізу у зручному форматі. Нарешті, веб-додаток може забезпечити гнучку та добре досліджену систему сповіщень про важливі події, з можливістю їх доставки через різні канали [17].

Таким чином, інтеграція веб-застосунку з існуючою локальною інфраструктурою відеоспостереження на роботизованому виробництві не лише використовує багато типів обмежень таких систем, але й значно розширює їх функціональність, роблячи спостереження більш ефективним, доступним та інтегрованим у загальний виробничий процес.

1.4 Архітектура системи хмарного відеоспостереження

На відміну від традиційних локальних систем, хмарне відеоспостереження для моніторингу роботизованих виробничих зон базується на принципово іншій, розподіленій архітектурі [18]. Найважливішою особливістю є перенесення основних функцій обробки, зберігання та управління відеоданими на потужні віддалені сервери, що належать хмарному провайдеру. У цій моделі IP-камери, встановлені безпосередньо на виробничому майданчику для захоплення відеозображень, підключені до локальної мережі компанії та використовують стабільне та швидке інтернет-з'єднання для передачі потоків відеоданих безпосередньо на ці хмарні сервери.

Таким чином, центральним елементом системи є не фізичний пристрій на місці, як у випадку локального мережевого відеореєстратора (NVR), а інфраструктура постачальника хмарних послуг. Саме на його серверах

відеоархіви отримуються, обробляються (включаючи кодування за необхідності) та безпечно зберігаються протягом певного періоду з обов'язковим резервним копіюванням для забезпечення високої надійності. Крім того, програмне забезпечення, що працює на цих серверах, відповідає за керування всіма підключеними камерами, налаштування параметрів відеозапису, адміністрування прав доступу для різних користувачів та обробку різних подій, записаних камерами. Деякі хмарні платформи інтегрують або пропонують можливість підключення додаткових сервісів відеоаналітики, які обробляють відеодані безпосередньо в хмарі, забезпечуючи розширені можливості для інтелектуального спостереження [18].

Користувачі можуть отримати доступ до всіх функцій хмарної системи відеоспостереження через зручні веб-інтерфейси або спеціалізовані мобільні додатки. Ці клієнтські програми взаємодіють із серверами хмарного провайдера через Інтернет, дозволяючи їм переглядати відео в режимі реального часу з будь-якої авторизованої точки, відтворювати архівні записи за певний період, дистанційно керувати налаштуваннями підключених камер та отримувати швидкі сповіщення про важливі події, що відбуваються на виробничому майданчику. Щоб забезпечити гнучкість та можливість інтеграції з іншими інформаційними системами, включаючи веб-застосунок для моніторингу РВД, що розробляється, хмарні платформи часто надають інтерфейси прикладного програмування (API) [19].

Ключові відмінності між хмарною архітектурою та локальними системами досить суттєві. У локальних системах центральним елементом є фізичний відеореєстратор, розташований безпосередньо на об'єкті, який відповідає за запис і зберігання відео на локальних жорстких дисках. Обробка відео також переважно виконується на тому ж обладнанні або на локальному сервері з встановленим програмним забезпеченням для керування відео (VMS). Доступ до відео часто обмежується прямим підключенням монітора до відеореєстратора або використанням клієнтської програми в локальній мережі. На відміну від цього, хмарна система повністю залежить від

стабільного інтернет-з'єднання, оскільки всі дані передаються та зберігаються на віддалених серверах, а доступ до них здійснюється через інтернет [19].

Коли йдеться про масштабованість, хмарні рішення мають значну перевагу, оскільки дуже легко розширити систему, додавши нові камери або збільшивши обсяг сховища, часто шляхом зміни тарифного плану та простого підключення нових камер до мережі. Локальні системи в цьому відношенні більш обмежені фізичною потужністю встановленого обладнання. Початкові витрати на розгортання хмарної системи зазвичай значно нижчі, оскільки немає потреби купувати дороге обладнання – користувач сплачує лише щомісячну абонентську плату за послугу та обсяг використаного сховища. З іншого боку, локальні системи потребують значних капіталовкладень на початковому етапі. У довгостроковій перспективі порівняння витрат може відрізнитися та залежить від розміру системи та тривалості використання.

Обслуговування хмарної інфраструктури повністю лягає на плечі постачальника хмарних послуг, що звільняє ІТ-персонал компанії від цих завдань. Коли йдеться про локальні системи, відповідальність за обслуговування, оновлення та усунення несправностей лежить на власнику системи. Питання безпеки даних у хмарних системах є відповідальністю постачальника, а користувач повинен ретельно оцінити його надійність та вжиті заходи безпеки. У локальних системах безпека даних повністю залежить від заходів, вжитих самою компанією. Віддалений доступ є стандартною функцією в хмарних системах, що надається через веб-інтерфейс або мобільні додатки, тоді як налаштування в локальних системах може бути складнішим і вимагати додаткових зусиль [20]. Зрештою, інтеграція з іншими системами в хмарних рішеннях часто забезпечується за допомогою добре документованих АРІ, що спрощує розробку користувацьких інтеграцій, включаючи веб-застосунок для моніторингу RVD.

Розуміння цих фундаментальних відмінностей в архітектурі між локальними та хмарними системами відеоспостереження є критично важливим для прийняття обґрунтованого рішення щодо вибору оптимального

підходу для моніторингу роботизованої виробничої ділянки та для проектування ефективного веб-додатку, здатного інтегруватися з обраною інфраструктурою.

1.5 Постановка задачі

Аналіз теми робототехнічного моніторингу виробничих майданчиків (RMS) виявив низку суттєвих слабких місць та неоптимізованих аспектів традиційних та існуючих підходів. Зокрема, локальні системи відеоспостереження, хоча й забезпечують базовий рівень візуального контролю, часто страждають від обмеженого локального доступу до відеоданих, труднощів в інтеграції з іншими критичними виробничими системами, незручних інтерфейсів користувача, відсутності централізованого моніторингу для кількох виробничих ділянок, обмежених можливостей аналізу та труднощів у налаштуванні гнучких систем сповіщення. Ці недоліки значно ускладнюють оперативний та комплексний контроль складних робототехнічних процесів, обмежуючи можливості швидкого реагування на надзвичайні ситуації, поглибленого аналізу ефективності виробництва та прийняття обґрунтованих управлінських рішень.

З іншого боку, потенціал хмарних систем відеоспостереження все ще значною мірою не використовується в контексті комплексного моніторингу. Хоча хмарні рішення пропонують важливі переваги, такі як масштабованість, доступність та нижчі початкові витрати, їх використання часто обмежується базовим переглядом відео без глибокої інтеграції з даними від робототехнічного обладнання, датчиків та інших виробничих систем. Це призводить до фрагментарного сприйняття виробничого процесу та ускладнює отримання повної картини функціонування відділу досліджень та розробок.

Усвідомлюючи ці існуючі обмеження та можливості, виникла нагальна потреба в розробці інтегрованого рішення, яке б поєднувало переваги хмарної

технології відеоспостереження з потребами комплексного моніторингу роботизованих виробничих зон. Тому було прийнято рішення створити спеціалізований веб-додаток, призначений для моніторингу RVD шляхом глибокої інтеграції з хмарною системою відеоспостереження.

Цей веб-додаток має на меті усунути виявлені вразливості, надаючи користувачам зручний та інтуїтивно зрозумілий віддалений доступ до потоків відео в реальному часі та архівних записів з будь-якого мережевого пристрою. Важливим аспектом розробки буде забезпечення безперешкодної інтеграції з різними виробничими системами через API, що дозволить поєднувати візуальні дані з іншою важливою інформацією про роботу R&D-центру, забезпечуючи цілісну картину виробничого процесу. Веб-додаток буде розроблено з урахуванням вимог зручності використання, що зробить його доступним для широкого кола користувачів, незалежно від рівня їхньої технічної освіти. У випадку кількох роботизованих зон, додаток забезпечить централізований моніторинг усіх систем з однієї точки доступу. Крім того, планується впровадження розширених аналітичних можливостей, використовуючи можливості як хмарної платформи, так і власних алгоритмів обробки даних, а також гнучку та налаштовану систему сповіщень про критичні події.

Таким чином, розробка цього веб-застосунку є своєчасним та актуальним рішенням, спрямованим на підвищення ефективності моніторингу роботизованих виробничих зон шляхом інтеграції передових хмарних технологій відеоспостереження з потребами комплексного управління сучасним виробництвом.

2 СТВОРЕННЯ МОДЕЛЕЙ ТА ВІЗУАЛЬНИХ СХЕМ ДЛЯ СИСТЕМИ МОНІТОРИНГУ РОБОТИЗОВАНОЇ ВИРОБНИЧОЇ ДІЛЯНКИ

2.1 Діаграма прецедентів для моделювання взаємодії користувача з системою

Діаграма варіантів використання є важливим інструментом для візуалізації функціональних вимог системи з точки зору кінцевих користувачів, які взаємодіють з нею для досягнення певних цілей. У контексті нашого веб-застосунку для моніторингу роботизованої виробничої лінії, інтегрованої з хмарною системою відеоспостереження, визначено чотири основні категорії користувачів або суб'єктів, кожна з яких має свій власний набір потреб та дозволів під час роботи з системою.

Перший актор – адміністратор. Ця роль передбачає високий рівень доступу та відповідальності за управління системою в цілому. Адміністратор запускає варіант використання автентифікації, щоб отримати доступ до адміністративних функцій. Після успішної автентифікації він може виконувати такі варіанти використання, як керування користувачами, що включає додавання нових облікових записів, редагування існуючих, видалення користувачів та призначення відповідних ролей і прав доступу. Ще одним важливим варіантом використання для адміністратора є налаштування системи, яке дозволяє налаштовувати загальні параметри системи, можливо, включаючи керування підключеними камерами системи в хмарі (якщо така функціональність доступна через API), встановлення правил для створення сповіщень про події та інші параметри системи. Окрім адміністративних функцій, адміністратор також має можливість виконувати всі попередні дії, доступні іншим користувачам, такі як перегляд відео в реальному часі, перегляд архівних записів, керування відеодисплеєм, отримання сповіщень,

перегляд стану робота (перегляд поточної інформації про роботу робота та даних датчиків) та перегляд статистики та звітів (доступ до індикатора).

Другим учасником є оператор, який є основним користувачем для щоденного моніторингу роботи робота. Оператор також починає свою роботу з прецедентом автентифікації. Після входу в систему, він активно використовує попередній перегляд відео в режимі реального часу для візуального контролю виробничих процесів. У разі інцидентів або для аналізу минулих подій оператор використовує прецедент «Перегляд архівного відеоматеріалу» з можливістю фільтрації відеозаписів за часом та камерою. Для зручнішого моніторингу оператор може використовувати прецедент для керування відображенням відео, вибору потрібних камер для одночасного перегляду, зміни розміру вікна трансляції та використання повноекранного режиму. Важливим аспектом роботи оператора є своєчасне отримання інформації про нештатні ситуації, що забезпечується прецедентом «Отримувати сповіщення», який дозволяє переглядати та, можливо, налаштовувати отримання сповіщень про події. Щоб отримати загальне уявлення про стан виробничої зони, оператор використовує прецедент статусу «Огляд RVD», який включає відображення поточної активності робота та даних від підключених датчиків.

Третій діяч – це інженер, головною метою якого є аналіз виробничих процесів та діагностика можливих проблем. Інженер також проходить автентифікацію для доступу до системи. Його ключовим прецедентом є огляд архівних записів, які він використовує для детального вивчення послідовності подій у минулому. Щоб отримати контекстну інформацію про поточний стан обладнання, інженер також використовує прецедент статусу «Огляд RVD», аналізуючи дані датчиків та інформацію про роботу робота.

Четвертим діячем є менеджер, зацікавлений у загальному огляді роботи RWD та отриманні аналітичної інформації для прийняття управлінських рішень. Після автентифікації менеджер може використовувати попередній перегляд відео в реальному часі та прецеденти перегляду архівних

відеоматеріалів для візуального контролю ключових виробничих процесів. Для оцінки загальної ефективності роботизованої виробничої системи менеджер використовує прецедентний статус роботизованої виробничої системи. Ключовим прецедентом для менеджера є перегляд статистики та звітів, який надає йому доступ до згенерованих звітів про продуктивність, час простою обладнання, кількість виробленої та бракованої продукції, а також до інших важливих аналітичних даних.

Таким чином, діаграма прецедентів чітко визначає роль користувачів та їх основні способи взаємодії з розробленим веб-застосунком, виділяючи функціональність системи, спрямовану на забезпечення ефективного моніторингу роботизованого виробничого майданчика за допомогою хмарної системи відеоспостереження.

2.2 Модель предметної області

Модель предметної області (рис. 2.1) є фундаментальним елементом у проектуванні всіх інформаційних систем, оскільки вона забезпечує чітке та структуроване розуміння важливих концепцій, сутностей та їх зв'язків у межах досліджуваної області. У контексті нашої системи моніторингу роботизованого виробничого майданчика (RPS) з використанням хмарної системи відеоспостереження ми визначили кілька ключових сутностей, кожна з яких відіграє важливу роль у функціонуванні системи та відображає певний аспект реального виробничого процесу.

Першим і одним з найважливіших підрозділів є роботизована виробнича дільниця. Він являє собою певний фізичний виробничий майданчик, оснащений роботизованими системами, який контролюється. Кожна система захисту прав людини (СРП) може мати унікальні характеристики, такі як назва для ідентифікації, географічне розташування в межах підприємства та детальний опис її функціонального призначення та складу обладнання.

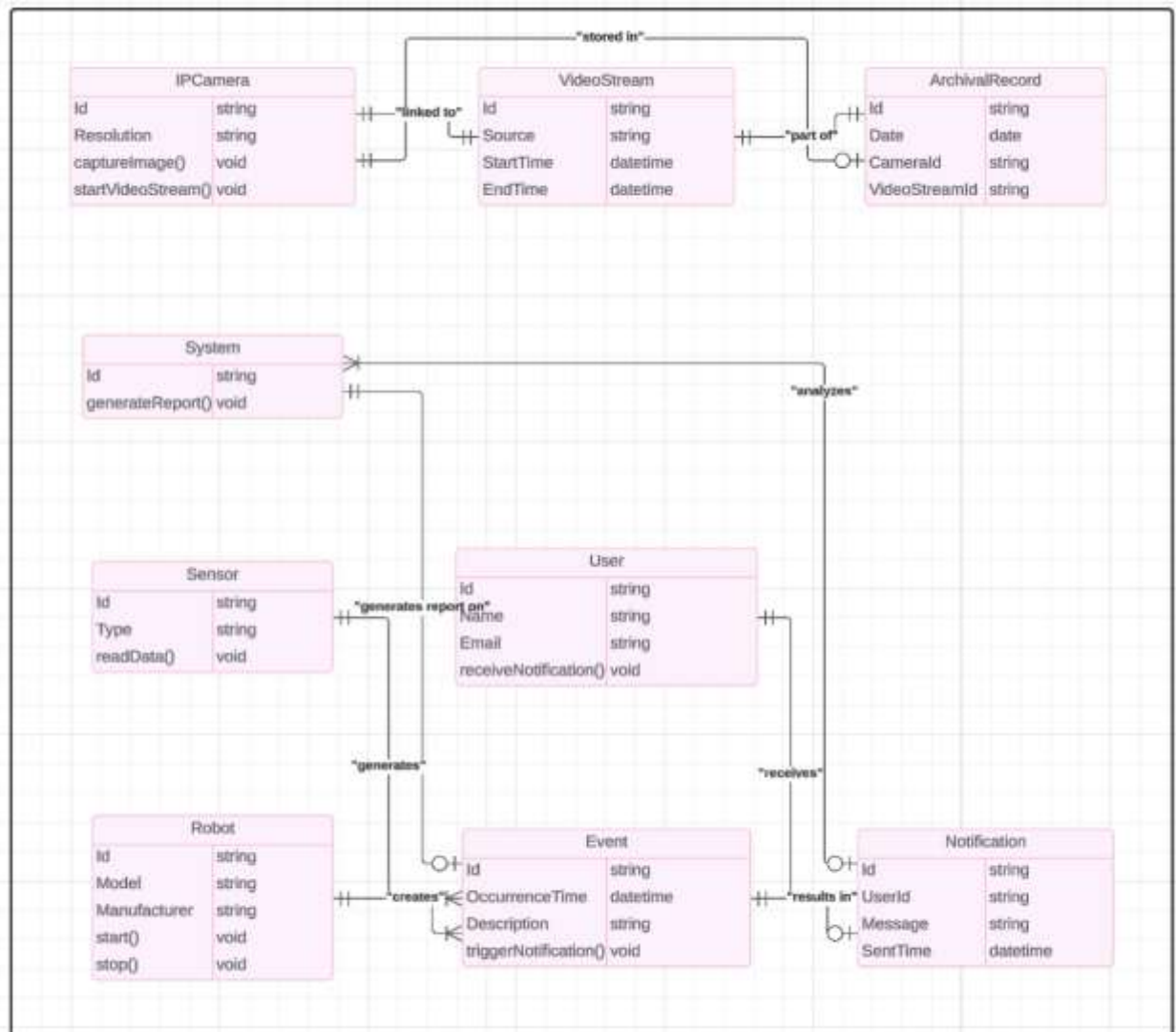


Рисунок 2.1 – Модель предметної області

У кожній RPS можуть працювати один або декілька роботів. Цей агрегат являє собою окремий роботизований блок, що бере участь у виконанні виробничих завдань. Кожен робот ідентифікується унікальним ідентифікатором і може мати такі атрибути, як тип (наприклад, маніпулятор, зварювальний робот), модель виробника та поточний стан (наприклад, працює, очікує, зупинений через помилку).

На РВД встановлені датчики для збору різних даних про стан виробничого середовища та обладнання. Кожен датчик має свій унікальний ідентифікатор, визначається його тип (наприклад, датчик температури, тиску, вологості, вібрації), вказуються одиниці вимірювання для отриманих даних та

реєструється поточне значення вимірюваного параметра.

Візуальний огляд РДВ забезпечується за допомогою IP-камер. Кожна камера має свій власний ідентифікатор, описову назву, вказується її фізичне розташування на виробничому майданчику та визначається роздільна здатність відео, яке вона записує. Кожна IP-камера генерує один або декілька відеопотоків, які характеризуються унікальною URL-адресою доступу, форматом передачі даних та частотою кадрів. Відеопотоки передаються в режимі реального часу на хмарну систему відеоспостереження. Окрім потокового відео, кожна IP-камера може створювати безліч архівних записів, кожен з яких має свій власний ідентифікатор, описову назву, записує точний час початку та закінчення запису, містить URL-адресу для доступу до збереженого відео та пов'язаний з певною IP-камерою, яка зробила запис.

Під час роботи РВД можуть відбуватися різні події, які реєструються системою. Кожна подія має унікальний ідентифікатор, чітко визначений тип (наприклад, виявлення руху в певній зоні, неочікуваний зупин робота, значення датчика, що перевищує встановлені межі), точний час і дату події, детальний опис і може бути пов'язана з одним або кількома конкретними системними об'єктами, такими як робот, датчик або IP-камера.

Щоб швидко інформувати користувачів про важливі події, система генерує сповіщення. Кожне сповіщення має унікальний ідентифікатор, містить текст повідомлення, записує час і дату створення, має статус (наприклад, прочитане або непрочитане) та пов'язане з певним користувачем, для якого воно призначене.

Кожен, хто взаємодіє із системою, представлений сутністю користувача. Кожен користувач має унікальний ідентифікатор, ім'я для ідентифікації, логін та пароль для автентифікації, а також визначену роль, яка визначає його права доступу та функціональність у системі (наприклад, адміністратор, оператор, інженер, менеджер).

Для аналізу функціонування РВД та надання зведеної інформації система може генерувати різні звіти. Кожен звіт має унікальний

ідентифікатор, назву, дату та час створення, визначений тип (наприклад, звіт про продуктивність, звіт про прості обладнання) та містить структурований контент з аналітичними даними.

Взаємозв'язки між цими одиницями є ключовими для розуміння функціонування системи. RVD може містити багато роботів, датчиків та IP-камер. Кожна IP-камера генерує кілька відеопотоків і може мати велику кількість архівних записів. Різні події можуть бути пов'язані з роботами, датчиками та IP-камерами. Кожна подія може ініціювати одне або кілька сповіщень для відповідного користувача. Кожен користувач може отримувати кілька сповіщень. Система може створювати кілька звітів на основі зібраних даних.

2.3 Діаграма класів

Діаграма класів (рис. 2.2) представляє статичну структуру розроблюваної системи, визначаючи основні класи, їхні атрибути та взаємозв'язки між ними. Вона слугує blueprint-ом для подальшої розробки програмного забезпечення та бази даних.

У нашій системі визначено наступні ключові класи.

Клас `RoboticWorkcell` (Роботизована Виробнича Ділянка) є центральним агрегатом, що представляє фізичну виробничу ділянку, за якою здійснюється моніторинг. Кожен екземпляр цього класу має унікальний ідентифікатор (`workcellId`), назву (`name`), інформацію про розташування (`location`) та детальне текстове `description` ділянки.

Клас `Robot` (Робот) відображає окремий роботизований пристрій, що є частиною певної `RoboticWorkcell`. Кожен робот має свій унікальний `robotId`, описову `name`, вказує на свій `type` (наприклад, зварювальний, маніпулятор), `model` виробника, поточний `status` (який може набувати значень з переліку `ACTIVE`, `INACTIVE`, `ERROR`, `IDLE`) та ідентифікатор `workcellId`.

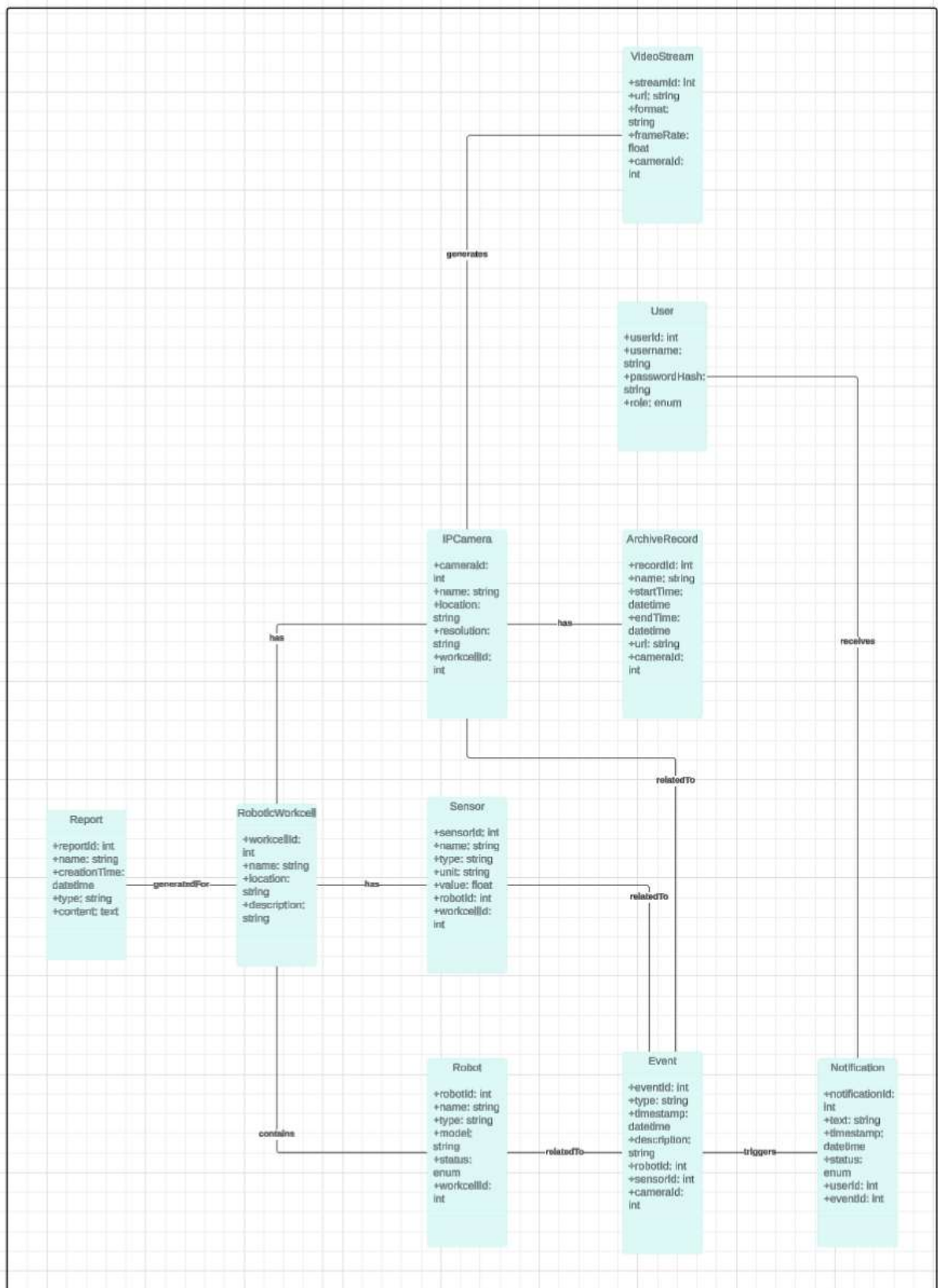


Рисунок 2.2 – Діаграма класів

Клас Sensor (Датчик) представляє сенсорний пристрій, встановлений на роботі або безпосередньо на виробничій ділянці для збору різноманітних

даних. Кожен датчик має унікальний `sensorId`, `name`, вказує на свій `type` (наприклад, температура, тиск), одиниці виміру (`unit`), поточне вимірне `value`, а також ідентифікатори `robotId` (якщо датчик встановлений на роботі) та `workcellId`, що вказують на його розташування.

Клас `IPCamera` (IP-Камера) відображає камеру відеоспостереження, встановлену на певній `RoboticWorkcell`. Кожна камера має унікальний `cameraId`, `name`, вказує на своє `location` на ділянці, `resolution` відео та ідентифікатор `workcellId`.

Клас `VideoStream` (Відеопотік) представляє потік відеоданих, що генерується конкретною `IPCamera`. Кожен відеопотік має унікальний `streamId`, `url` для доступу до потоку, `format` відео та `frameRate`, а також ідентифікатор `cameraId`, що вказує на джерело потоку.

Клас `ArchiveRecord` (Архівний Запис) відображає збережений відеозапис з певної `IPCamera`. Кожен запис має унікальний `recordId`, описову `name`, час початку (`startTime`) та закінчення (`endTime`) запису, `url` для доступу до архіву та ідентифікатор `cameraId`.

Клас `Event` (Подія) фіксує будь-яку значущу подію, що відбувається в системі (наприклад, виявлення руху, помилка обладнання). Кожна подія має унікальний `eventId`, вказує на свій `type`, час виникнення (`timestamp`), детальний `description`, а також може мати посилання на пов'язані `robotId`, `sensorId` або `cameraId` (ці ідентифікатори можуть бути нульовими, якщо подія не пов'язана з конкретним об'єктом).

Клас `Notification` (Сповіднення) представляє повідомлення для користувача про певну `Event`. Кожне сповіщення має унікальний `notificationId`, `text` повідомлення, час створення (`timestamp`), `status` (наприклад, прочитано чи ні), ідентифікатор користувача (`userId`), якому воно адресоване, та ідентифікатор пов'язаної `eventId`.

Клас `User` (Користувач) відображає користувача системи з його унікальним `userId`, `username`, хешом `passwordHash` для безпечного зберігання пароля та `role` (яка може набувати значень з переліку `ADMIN`, `OPERATOR`,

ENGINEER, MANAGER), що визначає його права доступу.

Клас Report (Звіт) представляє згенерований системою звіт про роботу RoboticWorkcell. Кожен звіт має унікальний reportId, name, час створення (creationTime), type звіту та його content у текстовому форматі. Звіт також пов'язаний з конкретною RoboticWorkcell, для якої він був створений.

Взаємозв'язки між класами відображають структурні зв'язки між сутностями: RoboticWorkcell агрегує (1 до *) Robot, Sensor та IPCamera. IPCamera асоціюється (1 до *) з VideoStream та агрегує (1 до *) ArchiveRecord. Класи Robot, Sensor та IPCamera можуть бути асоційовані (0..1 до *) з класом Event. Клас Event асоціюється (1 до *) з класом Notification, а клас User асоціюється (1 до *) з класом Notification. Клас Report асоціюється (* до 1) з класом RoboticWorkcell, вказуючи, для якої ділянки звіт був згенерований.

Ця детальна діаграма класів (рис. 2.2) є основою для проектування бази даних та розробки серверної частини веб-додатку, визначаючи структуру даних та основні сутності, з якими буде оперувати система.

2.4 Діаграма послідовностей

Діаграми послідовностей (рис. 2.3) є важливим інструментом для моделювання поведінки системи, оскільки вони показують взаємодії між об'єктами в часовій послідовності для виконання певного варіанту використання (випадку використання). Вони візуалізують обмін повідомленнями між різними об'єктами в системі та показують порядок цих повідомлень у часі. Кожен об'єкт представлений вертикальною лінією життя, а повідомлення між об'єктами представлені горизонтальними стрілками, позначеними назвою повідомлення та, за необхідності, надісланими параметрами. Час на діаграмі тече зверху вниз.

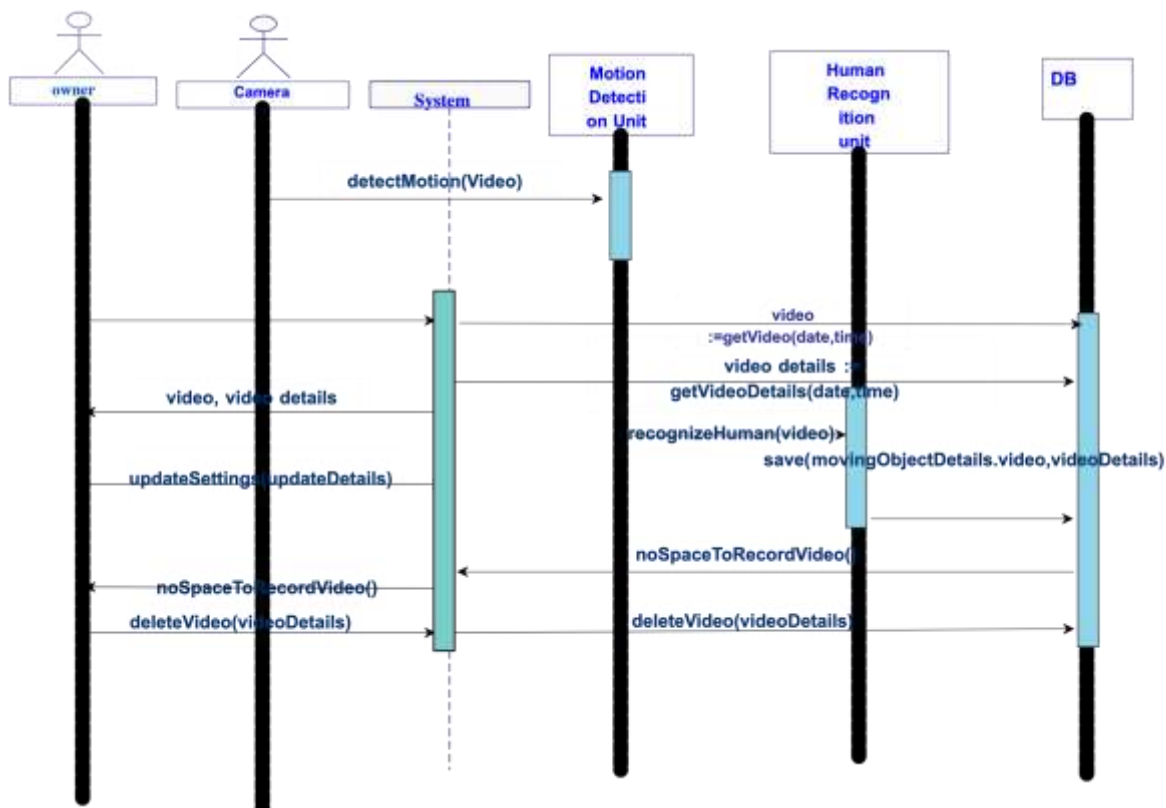


Рисунок 2.3 – Діаграма послідовностей

У контексті нашої системи відеоспостереження RVD, діаграми послідовностей допоможуть нам зрозуміти, як різні компоненти системи (наприклад, користувачі через веб-інтерфейс, серверну частину веб-застосунку, хмарну систему відеоспостереження, базу даних) взаємодіють один з одним для реалізації певних функцій, визначених на діаграмі варіантів використання. Наприклад, ми можемо створити діаграму послідовності для сценарію «Перегляд відео в реальному часі», яка показує, як користувач ініціює запит через веб-інтерфейс, як цей запит обробляється сервером веб-застосунків, як сервер взаємодіє з API хмарної системи для отримання відеопотоку та як цей потік надсилається назад користувачеві для перегляду.

Аналогічно, ми можемо розробити діаграми послідовностей для інших важливих сценаріїв, таких як «Перегляд архівних записів», «Отримання сповіщень», «Автентифікація користувача» та інші. Ці діаграми допомагають визначити потенційні точки взаємодії, залежності між компонентами, можливі затримки або проблеми з продуктивністю, а також уточнити деталі реалізації

кожного функціонального процесу.

Коли ми будуємо діаграми послідовностей, ми ідентифікуємо сутності, що беруть участь у сценарії, визначаємо послідовність повідомлень, якими вони обмінюються, та вказуємо будь-які умови або цикли. Це дозволяє нам отримати чітке розуміння динамічної поведінки системи та розробити ефективну взаємодію між її компонентами, включаючи наш веб-застосунок та хмарну систему відеоспостереження.

3 СТВОРЕННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ СИСТЕМИ МОНІТОРИНГУ РОБОТИЗОВАНОЇ ВИРОБНИЧОЇ ДІЛЯНКИ

3.1 Засоби розробки та мова програмування

На етапі створення програмного забезпечення для моніторингу нашої системи роботизованого виробництва, найважливішим рішенням є вибір технологічного стеку, який забезпечить ефективну, надійну та масштабовану роботу майбутнього веб-застосунку. Враховуючи вимоги до швидкої розробки, підтримки складної бізнес-логіки та інтеграції із зовнішніми сервісами, такими як хмарні системи відеоспостереження, вибір мови програмування Ruby є надзвичайно обґрунтованим. Ruby, відомий своєю елегантністю та орієнтованим на людину синтаксисом, дозволяє розробникам писати чистий, зрозумілий та легко підтримуваний код, що є вирішальним для довгострокового успіху проекту. Завдяки своїй об'єктно-орієнтованій природі, Ruby сприяє створенню модульної архітектури, що полегшує розширення та модифікацію функціональності веб-застосунку в майбутньому.

Ruby on Rails (RoR) – це не просто набір бібліотек, це цілісна екосистема, яка значно спрощує та пришвидшує процес веб-розробки завдяки своїй філософії «умовність переваги над конфігурацією». Це означає, що фреймворк забезпечує розумні налаштування за замовчуванням для більшості типових завдань, звільняючи розробників від необхідності приймати багато низькорівневих рішень та дозволяючи їм зосередитися на додатках бізнес-логіки. RoR пропонує потужну систему маршрутизації, інтегровану підтримку роботи з базами даних через ORM Active Record, зручну систему шаблонів для генерації HTML та багато інших вбудованих інструментів і генераторів коду, які значно підвищують продуктивність розробки.

Щоб забезпечити зручний та ефективний процес кодування, розробники

можуть перейти на одне з провідних інтегрованих середовищ розробки (IDE). Інтегроване середовище розробки RubyMine для коментарів доступне для Ruby on Rails. Автокомпліт коду, навігація, рефакторинг, налагодження, профілювання та інтеграції з системою контролю. Як безкоштовна, надзвичайно популярна альтернатива, Visual Studio Code (VS Code), який завдяки своїй гнучкості та великій кількості розширень для Ruby та Rails, включаючи лінтери, форматери, налагоджувачі та інструменти для роботи з Git, стає потужним інструментом в руках розробників Ruby. Також варто відзначити простий та швидкий текстовий редактор Sublime Text з широкими можливостями налаштування та підтримкою Ruby за допомогою плагінів.

Вибір системи керування базами даних має вирішальне значення для зберігання та управління даними в нашому вебзастосунку. PostgreSQL – це потужна, надійна реляційна база даних промислового класу, яка бездоганно інтегрується з Ruby on Rails через платформу, забезпечуючи високий рівень узгодженості та масштабованості. MySQL також широко використовується як сервер реального часу з великою спільнотою та гарною підтримкою в Rails через mysql2, що пропонує баланс між продуктивністю та простотою використання. Для невеликих проектів або локальної розробки можна використовувати просту базу даних SQLite.

Керування залежностями в проектах Ruby стандартизоване за допомогою Bundler. Цей інструмент дозволяє точно визначити та встановити необхідні зовнішні бібліотеки для проекту, а також виправити їх версії, запобігаючи конфліктам та забезпечуючи узгодженість у середовищі розробки та розгортання.

Для розгортання застосунків Ruby on Rails на продакшн-серверах часто використовуються сервери застосунків, такі як Puma (багатопотоковий сервер, що підходить для більшості вебзастосунків) або Unicorn (сервер на основі попередньо потокових процесів).

Обираючи Ruby та екосистему Ruby on Rails, ми отримуємо потужний, гнучкий та продуктивний інструмент для розробки серверної частини нашого

веб-застосунку для моніторингу роботизованого виробництва, який дозволяє швидко створювати складні веб-рішення з високоякісним кодом та ефективною інтеграцією із зовнішніми сервісами.

3.2 Аналіз часових затримок у контурі управління людина-машина

На етапі створення програмного забезпечення для моніторингу нашої системи роботизованого виробництва, найважливішим рішенням є вибір технологічного стеку. Для обґрунтування цього вибору необхідно вийти за межі простої оцінки продуктивності та розглянути систему з точки зору теорії автоматичного управління.

Систему моніторингу, де оператор спостерігає за даними та приймає рішення, можна розглядати як контур управління людина-машина. У цьому контурі оператор виступає в ролі керуючого елемента (регулятора), а веб-додаток є його інтерфейсом до об'єкта управління (роботизованої ділянки). Ефективність такого контуру напряду залежить від загальної часової затримки (latency) – часу, що проходить від моменту виникнення події на виробництві до моменту її усвідомлення оператором.

Загальна затримка T_{total} є ключовим параметром якості управління. Велика затримка може призвести до несвоєчасної реакції на критичні ситуації, помилкових дій оператора та, як наслідок, до збоїв у виробництві або аварій. Тому завданням є вибір такої архітектури та технологій, які мінімізують T_{total}

Загальна затримка складається з суми затримок на кожному етапі передачі та обробки інформації. Розглянемо повний шлях сигналу:

$$T_{total} = T_{event} + T_{net1} + T_{proc} + T_{delivery} + T_{net2} + T_{render}.$$

Ця загальна затримка складається з кількох ключових компонентів. По-перше, це час на фіксацію самої події датчиком та відправку даних промисловим контролером робота (T_{event}), який ми можемо умовно прийняти

за 50 мс. Далі, дані повинні пройти через локальну мережу від виробничої ділянки до сервера, що вносить затримку T_{net1} , яку оцінимо у 20 мс. Після отримання даних сервер витрачає час на їх обробку, наприклад, запис у базу даних, що складає затримку T_{proc} (прийmemo 80 мс). Ключовим елементом, який залежить від нашої архітектури, є час доставки даних від сервера до клієнта ($T_{delivery}$). Потім дані проходять через зовнішню мережу до браузера оператора, що додає затримку T_{net2} (наприклад, 40 мс). І нарешті, браузеру потрібен час, щоб відобразити (відрендерити) отримані зміни в інтерфейсі, що складає затримку T_{render} (прийmemo 30 мс).

Для вибору технології порівнюємо два фундаментально різних підходи до реалізації доставки даних.

1. Періодичне опитування (HTTP Long-Polling).

Дана архітектура передбачає, що клієнтська частина застосунку з визначеною періодичністю I_{poll} ініціює HTTP-запит до сервера з метою отримання оновлених даних. Це є стандартною реалізацією для багатьох систем, що функціонують на базі REST API. Затримка доставки даних $T_{delivery}$ в такій моделі є випадковою величиною, розподіленою в діапазоні від 0 до I_{poll} .

Для аналізу приймається її середнє математичнє очікування, що дорівнює половині інтервалу опитування:

$$T_{delivery} = \frac{I_{poll}}{2}.$$

Для забезпечення балансу між актуальністю даних та навантаженням на системні ресурси, інтервал опитування I_{poll} приймається рівним 2 секунди (2000 мс). Більш часті запити можуть спричинити надлишкове навантаження, тоді як більш рідкісні призводять до втрати актуальності інформації. Таким чином, середня затримка доставки становить:

$$T_{delivery} = \frac{2000}{2} = 1000 \text{ мс.}$$

Враховуючи це значення, розрахуємо повну затримку для архітектури на основі HTTP Polling:

$$T_{delivery} = 50 + 20 + 80 + 1000 + 40 + 30 = 1220 \text{ мс} = 1,22 \text{ секунди.}$$

2. Архітектура на основі постійного з'єднання (WebSockets).

Технологія WebSockets дозволяє встановити стійкий, двонаправлений комунікаційний канал між клієнтом та сервером. На відміну від моделі «запит-відповідь», дана архітектура реалізує модель «видавець-підписник» (Publisher/Subscriber), де сервер ініціює передачу даних клієнту одразу після їх появи.

Внаслідок цього, системна затримка доставки даних $T_{delivery}$ не залежить від інтервалу очікування та є нехтовно малою величиною, оскільки обмежується лише часом проходження пакетів мережею, який вже враховано в інших компонентах загальної затримки. Тому приймаємо $T_{delivery\ ws}$ за 0.

Розрахуємо повну затримку для архітектури на основі WebSockets:

$$T_{total\ ws} = 50 + 20 + 80 + 0 + 40 + 30 = 220 \text{ мс} = 0,22 \text{ секунди.}$$

Аналіз часових затримок у контурі управління «людина-машина» наочно демонструє перевагу архітектури, що базується на технології WebSockets. Різниця у загальній затримці (1,22 с проти 0,22 с) є критичною. Затримка більше однієї секунди є неприйнятною для систем моніторингу, де потрібна швидка реакція на події, такі як зупинка конвеєра, помилка робота або спрацювання датчика безпеки.

Затримка у 220 мс, яку забезпечує підхід з WebSockets, дозволяє створити відчуття роботи в реальному часі, що значно підвищує ефективність

та безпеку управління роботизованою ділянкою.

Таким чином, на основі проведеного аналізу, обґрунтованим є вибір технологічного стеку, який нативно підтримує технологію WebSockets. До таких стеків належать Node.js (з бібліотеками Socket.IO або ws), Ruby (з фреймворками Ruby on Rails або Grape), чи Java (з Spring WebFlux). Цей вибір забезпечить мінімально можливий час реакції системи моніторингу, що є фундаментальною вимогою з точки зору теорії автоматичного управління.

3.3 Розробка бази даних для системи

Для ефективного зберігання та управління даними, необхідними для функціонування нашого веб-додатку, розробляється реляційна база даних (БД) (рис. 3.1), структура якої базується на визначеній раніше моделі предметної області та діаграмі класів.

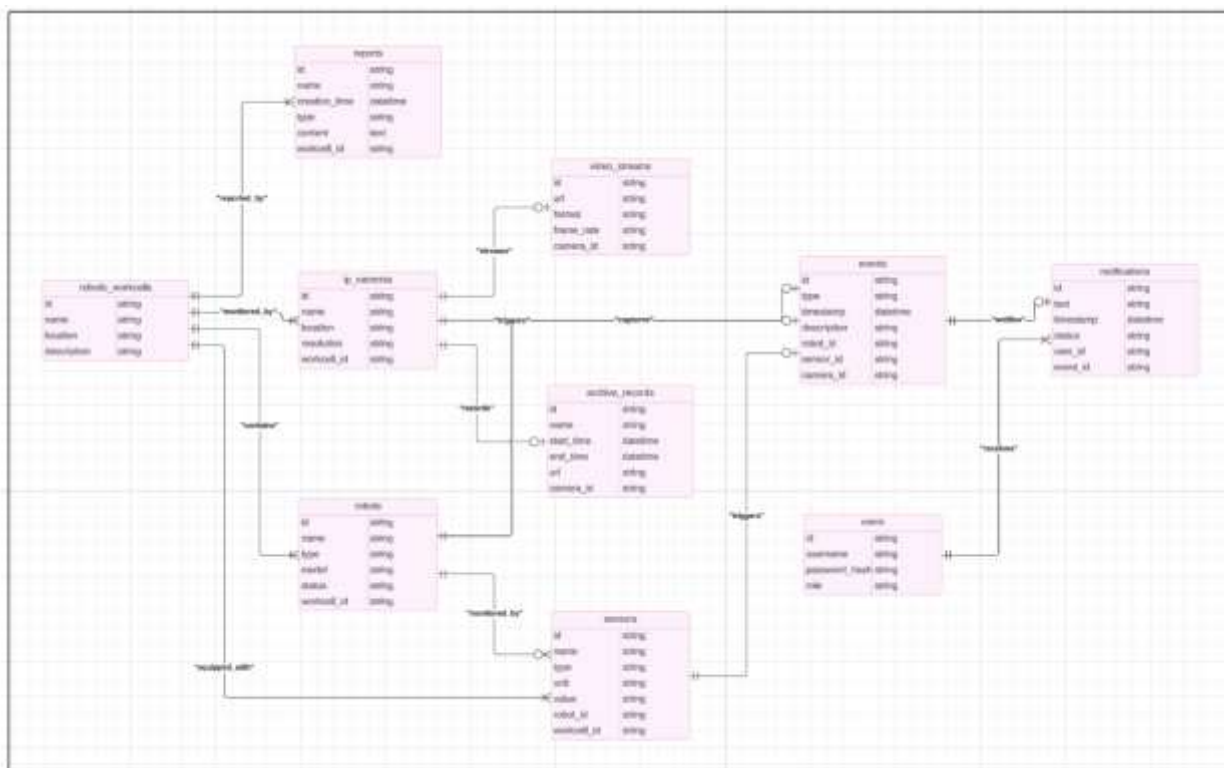


Рисунок 3.1 – Схема бази даних

Основу схеми складають десять взаємопов'язаних таблиць, кожна з яких відповідає за зберігання інформації про конкретну сутність системи. Центральною таблицею є `robotic_workcells`, яка містить інформацію про кожну роботизовану виробничу ділянку, включаючи її унікальний ідентифікатор, назву, місцезнаходження та детальний опис. З кожною виробничою ділянкою пов'язані таблиці `robots`, що зберігає дані про окремі роботи, включаючи їхні ідентифікатори, типи, моделі, поточний статус та зовнішній ключ, що вказує на їхню приналежність до конкретної РВД, та `sensors`, яка містить інформацію про різноманітні датчики, встановлені на роботах або безпосередньо на виробничих ділянках, фіксуючи їхні типи, одиниці виміру, поточні значення та зв'язки з відповідними роботами та РВД.

Для забезпечення візуального моніторингу використовуються таблиці `ip_cameras`, що зберігає дані про встановлені IP-камери, включаючи їхні ідентифікатори, назви, розташування та роздільну здатність, та `video_streams`, що містить інформацію про потокові відеодані з кожної камери, включаючи URL-адреси, формати та частоту кадрів, з зовнішнім ключем, що вказує на відповідну камеру. Архівні відеозаписи зберігаються в таблиці `archive_records`, де фіксуються їхні ідентифікатори, назви, час початку та закінчення, URL-адреси та зв'язок з камерою, що здійснила запис.

Для відстеження подій, що відбуваються в системі, призначена таблиця `events`, яка зберігає інформацію про тип події, час виникнення, опис та зовнішні ключі, що пов'язують подію з відповідними роботами, датчиками або IP-камерами. Для інформування користувачів про ці події використовується таблиця `notifications`, яка містить текст сповіщення, час створення, статус (прочитано/не прочитано), зовнішній ключ, що вказує на користувача, якому адресовано сповіщення, та зовнішній ключ, що пов'язує сповіщення з конкретною подією. Інформація про користувачів системи, включаючи їхні унікальні ідентифікатори, логіни, хеші паролів та ролі, зберігається в таблиці `users`. Для формування аналітичних звітів використовується таблиця `reports`, яка містить ідентифікатори звітів, їхні назви, час створення, типи, вміст та

зовнішній ключ, що вказує на РВД, для якої звіт був створений.

Взаємозв'язки між цими таблицями реалізуються за допомогою зовнішніх ключів, що забезпечує цілісність даних та відображає асоціації, визначені на рівні моделі предметної області. Наприклад, зв'язок "одна РВД має багато роботів" реалізується через зовнішній ключ `workcell_id` у таблиці `robots`, що посилається на первинний ключ таблиці `robotic_workcells`. Аналогічно встановлюються зв'язки між камерами та відеопотоками, архівними записами, подіями та сповіщеннями, користувачами та сповіщеннями, а також між звітами та відповідними РВД.

3.4 Розробка SQL-запитів для сервісної частини системи

Для забезпечення ефективної взаємодії веб-додатку з базою даних та отримання необхідної інформації для відображення користувачам та обробки бізнес-логіки сервісної частини системи, розробляється набір SQL-запитів, оптимізованих для нашої спроектованої схеми реляційної бази даних. Ці запити охоплюють основні потреби у вибірці, оновленні та, в перспективі, додаванні даних, забезпечуючи гнучкий та безпечний доступ до інформації про роботизовані виробничі ділянки, роботи, датчики, камери, відеопотоки, архівні записи, події, сповіщення та звіти. Важливою умовою при розробці цих запитів (рис. 3.2) є використання параметризованих запитів або підготовлених виразів для мінімізації ризиків SQL-ін'єкцій та підвищення загальної безпеки системи.

```
1 SELECT id, name, location, description
2 FROM robotic_workcells;
```

Рисунок 3.2 – Запит отримання списку всіх роботизованих виробничих ділянок

Запит на отримання списку всіх РВД (рис. 3.3) (SELECT id, name, location, description FROM robotic_workcells;) використовується для відображення загального переліку доступних виробничих ділянок у веб-додатку, надаючи користувачам можливість вибору конкретної ділянки для детальнішого моніторингу.

```
1 SELECT id, name, location, description
2 FROM robotic_workcells
3 WHERE id = :workcell_id;
```

Рисунок 3.3 – Запит отримання інформації про конкретну РВД за її ID

Запит на отримання інформації (рис. 3.4) про конкретну РВД за її ID (SELECT id, name, location, description FROM robotic_workcells WHERE id = :workcell_id;) дозволяє отримати повну інформацію про обрану користувачем виробничу ділянку, використовуючи її унікальний ідентифікатор, що передається як параметр.

```
1 SELECT id, url, format, frame_rate
2 FROM video_streams
3 WHERE camera_id = :camera_id;
```

Рисунок 3.4 – Запит отримання списку всіх IP-камер на конкретній РВД

Запит на отримання списку всіх IP-камер на конкретній РВД (рис. 3.5) (SELECT id, name, location, resolution FROM ip_cameras WHERE workcell_id = :workcell_id;) надає перелік усіх камер відеоспостереження, встановлених на обраній виробничій ділянці, з їхніми основними характеристиками.


```

1 SELECT id, name, start_time, end_time, url
2 FROM archive_records
3 WHERE camera_id = :camera_id
4 AND start_time >= :start_time AND end_time <= :end_time
5 ORDER BY start_time DESC;

```

Рисунок 3.5 – Запит отримання списку архівних записів для конкретної IP-камери за певний період часу

Запит на отримання списку архівних записів для конкретної IP-камери (рис. 3.6) за певний період часу (SELECT id, name, start_time, end_time, url FROM archive_records WHERE camera_id = :camera_id AND start_time >= :start_time AND end_time <= :end_time ORDER BY start_time DESC;) дозволяє користувачам переглядати збережені відеозаписи з обраної камери за вказаний часовий проміжок, з можливістю сортування за часом початку запису.

```

1 SELECT e.id, e.type, e.timestamp, e.description,
2       r.name AS robot_name, s.name AS sensor_name, c.name AS camera_name
3 FROM events e
4 LEFT JOIN robots r ON e.robot_id = r.id
5 LEFT JOIN sensors s ON e.sensor_id = s.id
6 LEFT JOIN ip_cameras c ON e.camera_id = c.id
7 WHERE r.workcell_id = :workcell_id OR s.workcell_id = :workcell_id OR c.workcell_id = :workcell_id
8 ORDER BY e.timestamp DESC
9 LIMIT 10;

```

Рисунок 3.6 – Запит отримання списку останніх подій на конкретній РВД

Запит на отримання списку останніх подій (рис. 3.7) на конкретній РВД (SELECT e.id, e.type, e.timestamp, e.description, r.name AS robot_name, s.name AS sensor_name, c.name AS camera_name FROM events e LEFT JOIN robots r ON e.robot_id = r.id LEFT JOIN sensors s ON e.sensor_id = s.id LEFT JOIN ip_cameras c ON e.camera_id = c.id WHERE r.workcell_id = :workcell_id OR s.workcell_id = :workcell_id OR c.workcell_id = :workcell_id ORDER BY e.timestamp DESC LIMIT 10;) використовується для відображення останніх подій, що відбулися на обраній виробничій ділянці, з можливістю отримання інформації про пов'язані роботи, датчики або камери.

```

1 SELECT n.id, n.text, n.timestamp, e.type AS event_type,
2       r.name AS robot_name, s.name AS sensor_name, c.name AS camera_name
3 FROM notifications n
4 JOIN events e ON n.event_id = e.id
5 LEFT JOIN robots r ON e.robot_id = r.id
6 LEFT JOIN sensors s ON e.sensor_id = s.id
7 LEFT JOIN ip_cameras c ON e.camera_id = c.id
8 WHERE n.user_id = :user_id AND n.status = 'unread'
9 ORDER BY n.timestamp DESC;

```

Рисунок 3.7 – Запит на отримання списку нових сповіщень для конкретного користувача

Запит на отримання списку нових сповіщень (рис. 3.8) для конкретного користувача (SELECT n.id, n.text, n.timestamp, e.type AS event_type, r.name AS robot_name, s.name AS sensor_name, c.name AS camera_name FROM notifications n JOIN events e ON n.event_id = e.id LEFT JOIN robots r ON e.robot_id = r.id LEFT JOIN sensors s ON e.sensor_id = s.id LEFT JOIN ip_cameras c ON e.camera_id = c.id WHERE n.user_id = :user_id AND n.status = 'unread' ORDER BY n.timestamp DESC;) дозволяє користувачам отримувати перелік нових, ще не прочитаних сповіщень, пов'язаних з їхнім обліковим записом, з детальною інформацією про подію, яка спричинила сповіщення.

```

1 SELECT value, unit, timestamp AS last_update
2 FROM sensors
3 WHERE id = :sensor_id;

```

Рисунок 3.8 – Запит отримання поточного значення конкретного датчика

Запит на отримання поточного значення (рис. 3.9) конкретного датчика (SELECT value, unit, timestamp AS last_update FROM sensors WHERE id = :sensor_id;) використовується для відображення актуальних показників обраного датчика, включаючи останнє зафіксоване значення, одиниці виміру

та час останнього оновлення даних.

```
1 UPDATE notifications
2 SET status = 'read'
3 WHERE id = :notification_id AND user_id = :user_id;
```

Рисунок 3.9 – Запит оновлення статусу сповіщення як "прочитано"

Запит на оновлення статусу сповіщення як "прочитано" (UPDATE notifications SET status = 'read' WHERE id = :notification_id AND user_id = :user_id;) використовується для фіксації факту перегляду сповіщення користувачем, змінюючи його статус на "прочитано".

Запит на отримання списку звітів (рис. 3.10) для конкретної РВД за певний період (SELECT id, name, creation_time, type FROM reports WHERE workcell_id = :workcell_id AND creation_time >= :start_time AND creation_time <= :end_time ORDER BY creation_time DESC;) надає користувачам можливість переглядати згенеровані звіти для обраної виробничої ділянки за вказаний часовий проміжок.

```
1 SELECT id, name, creation_time, type
2 FROM reports
3 WHERE workcell_id = :workcell_id
4 AND creation_time >= :start_time AND creation_time <= :end_time
5 ORDER BY creation_time DESC;
```

Рисунок 3.10 – Запит на отримання списку звітів для конкретної РВД за певний період

3.5 Огляд розробленого web-додатку

Розроблений веб-застосунок є централізованою інформаційною системою, призначеною для моніторингу та візуального контролю за станом роботизованих виробничих ділянок. Метою його створення є надання

операторам, інженерам та керівному персоналу зручного та ефективного інструменту для спостереження за виробничими процесами в реальному часі, а також для доступу до архівних даних.

Інтерфейс застосунку спроектовано з урахуванням принципів мінімалізму та функціональності, щоб забезпечити швидкий доступ до ключових елементів керування та мінімізувати час реакції користувача на позаштатні ситуації.

Після успішної автентифікації користувач потрапляє на головну сторінку системи, яка слугує дашбордом. На цій сторінці у вигляді інформативних карток або списку представлені всі виробничі сектори (ділянки), додані до системи моніторингу. Кожен елемент списку містить назву сектора та його поточний статус (наприклад, "Активний", "Потребує уваги", "Офлайн"), що дозволяє миттєво оцінити загальну ситуацію на виробництві.



Рисунок 3.11 – Головна сторінка застосунку зі списком виробничих секторів

При натисканні на картку конкретного сектора користувач переходить на сторінку детального огляду цієї ділянки. Ця сторінка містить більш

детальну інформацію про сектор та, що найголовніше, відображає список усіх камер відеоспостереження, встановлених у межах даної ділянки. Для кожної камери вказується її назва або ідентифікатор та її поточний статус (наприклад, "Передає сигнал", "Недоступна").



Рисунок 3.12 – Сторінка обраного сектора зі списком доступних камер

Вибір конкретної камери зі списку перенаправляє користувача на сторінку її перегляду. Центральним елементом цієї сторінки є відеоплеєр, який транслює відеопотік з камери в реальному часі. Це ключова функція системи, що забезпечує безпосередній візуальний контроль за обладнанням, персоналом та виробничими операціями. Під відеоплеєром можуть розташовуватися елементи керування для роботи з архівом записів (вибір дати та часу).



Рисунок 3.13 – Інтерфейс перегляду відеотрансляції з камери

Система також розроблена для обробки можливих збоїв. У випадку, якщо камера не в мережі, пошкоджена або з будь-якої іншої причини не може передавати відеосигнал, відеоплеєр відобразить відповідне повідомлення про помилку або спеціальну заглушку. Це дозволяє користувачеві чітко ідентифікувати проблемні точки в системі спостереження.



Рисунок 3.14 – Відображення стану камери за відсутності сигналу

Система має адміністративну панель, доступну для користувачів з відповідними правами. Однією з базових адміністративних функцій є можливість додавання нових виробничих секторів до системи моніторингу. Для цього передбачена спеціальна форма, де адміністратор повинен вказати назву нового сектора та, за потреби, надати його короткий опис. Після збереження новий сектор автоматично з'явиться на головному дашборді.



The image shows a web form titled "Add new sector". It contains three main elements: a text input field for "Sector Name", a dropdown menu for "Status" currently set to "Active", and a green button labeled "Create Sector".

Рисунок 3.15 – Форма для додавання нового виробничого сектора

3.6 Забезпечення безпечних умов праці при розробленні web-додатку

Належна організація робочого простору є одним із ключових факторів, що впливають на продуктивність, безпеку та загальне самопочуття працівника. У цьому розділі проводиться детальний аналіз умов праці на досліджуваному робочому місці з метою встановлення їх відповідності актуальним державним санітарно-гігієнічним нормам України [21].

Досліджуване робоче місце розташоване в офісному приміщенні, яке має площу 18 м² та висоту стель 3,2 метри. У кабінеті облаштовано два робочих місця, кожне з яких оснащено персональним комп'ютером.

Нормативною базою для оцінки є «Вимоги щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями» (наказ № 207 від 14.02.2018). Згідно з цим документом, на кожне робоче місце має припадати не менше 6,0 м² площі та 20,0 м³ об'єму повітря.

Шляхом простого розрахунку встановлено, що на кожне з двох робочих місць у даному приміщенні припадає по 9 м² площі, що значно перевищує нормативний мінімум. Загальний об'єм приміщення складає 57,6 м³, відповідно, на одного працівника припадає 28,8 м³ повітряного простору.

Таким чином, можна констатувати, що вимоги до площі та об'єму приміщення виконані із суттєвим запасом. Це створює комфортні умови, забезпечуючи достатню свободу пересування, психологічний комфорт та сприяючи кращій циркуляції повітря в приміщенні [21].

Якість освітлення безпосередньо впливає на зорову працездатність та є вирішальним фактором у профілактиці професійних захворювань органів зору. Вимоги до освітлення регламентуються Державними будівельними нормами ДБН В.2.5-28:2018 «Природне і штучне освітлення».

Природне освітлення в приміщенні забезпечується через віконні прорізи, що є обов'язковою вимогою для постійних робочих місць. Важливим є не лише його наявність, а й якість. Орієнтація вікон на північний схід є оптимальною, оскільки такий напрямок унеможливує потрапляння прямих сонячних променів, які створюють відблиски на екранах моніторів та сліпучий ефект. Нормований коефіцієнт природної освітленості (КПО) для робіт, пов'язаних із високою зоровою напругою, становить не менше 1,5%. Умови в приміщенні відповідають цій вимозі, що підтверджує достатній рівень денного світла.

Штучне освітлення організоване за допомогою системи загального рівномірного освітлення. В приміщенні встановлено 4 сучасних світлодіодних (LED) світильника. Потужність кожного світильника становить 36 Вт, що дає сумарну потужність освітлювальної установки в 144 Вт. Щоб оцінити ефективність системи, було розраховано питому потужність, яка показала

значення 8 Вт/м².

Згідно з довідковими даними для світлодіодних систем, така питома потужність створює рівень освітленості на робочих поверхнях в межах 400-450 люкс (лк). Норматив ДБН В.2.5-28:2018 для офісних робіт встановлює мінімально допустимий рівень у 300 лк, а рекомендований – 500 лк.

Отже, фактичний рівень освітленості в приміщенні не просто відповідає мінімальним вимогам, а наближається до рекомендованих значень. Це створює сприятливе світлове середовище, яке знижує зорову втому, підвищує концентрацію уваги та позитивно впливає на загальну працездатність протягом усього робочого дня [21].

ВИСНОВКИ

У даній кваліфікаційній роботі було успішно розглянуто та розроблено концепцію веб-додатку для моніторингу роботизованої виробничої ділянки з використанням хмарної системи відеоспостереження. Проведений аналіз предметної області виявив актуальність та необхідність створення такого інструменту для підвищення ефективності контролю та управління сучасними виробничими процесами.

В рамках роботи було визначено функціональні та нефункціональні вимоги до майбутнього веб-додатку, розроблено модель предметної області, яка чітко окреслила ключові сутності системи та їхні взаємозв'язки. На основі цієї моделі було спроектовано структуру реляційної бази даних, що забезпечує надійне зберігання та ефективне управління даними.

Особливу увагу було приділено вибору технологій розробки серверної частини, де перевагу було віддано мові програмування Ruby та веб-фреймворку Ruby on Rails за їхню продуктивність, гнучкість та розвинену екосистему. Було також розглянуто основні інструменти розробки, необхідні для створення якісного програмного забезпечення.

Для забезпечення взаємодії веб-додатку з базою даних було розроблено набір ключових SQL-запитів, які охоплюють основні потреби у вибірці, оновленні та фільтрації даних про роботизовані виробничі ділянки, роботи, датчики, камери, відеопотоки, події, сповіщення та звіти. Ці запити є основою для реалізації функціональності сервісної частини системи.

Розроблена концепція та запропоновані рішення є важливим кроком у напрямку створення ефективної системи моніторингу роботизованих виробничих ділянок. Подальша робота може бути спрямована на розробку клієнтської частини веб-додатку, інтеграцію з конкретною хмарною системою відеоспостереження через її API, реалізацію розширених функцій аналітики та тестування розробленої системи в реальних виробничих умовах.

Впровадження розробленого веб-додатку дозволить підприємствам

підвищити рівень контролю за роботизованими виробничими процесами, оперативно реагувати на нештатні ситуації, проводити глибокий аналіз виробничих даних та приймати обґрунтовані управлінські рішення, що в кінцевому підсумку сприятиме підвищенню ефективності та конкурентоздатності виробництва.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Невлюдов І. Ш. Комп'ютерно-інтегровані технології виробництва технічних засобів автоматизації. Частина 1: Підручник Харків: ФОП Панов А. М., 2021., 604 с.
2. ДСТУ 3008-15. Документація. Звіти у сфері науки та техніки. структура та правила оформлення. Введ. 2015-06-22. К. Держстандарт України, 2017. 29 с.
3. Положення про академічну доброчесність [Електронний ресурс]: Наказ ХНУРЕ від 02 лютого 2021 р. № 50. – Режим доступу: https://nure.ua/wpcontent/uploads/Main_Docs_NURE/polozhennja-pro-akademichnu-dobrochesnist.pdf.
4. Методичні вказівки з підготовки кваліфікаційної роботи бакалавра для здобувачів першого (бакалаврського) рівня вищої освіти спеціальності 151 Автоматизація та комп'ютерно-інтегровані технології освітньої програми А.І. Бронніков. Харків: ХНУРЕ, 2022. 66 с.
5. Lowenberg-DeBoer, J., Huang, I. Y., Grigoriadis, V., & Blackmore, S. (2020). Economics of robots and automation in field crop production. *Precision Agriculture*, 21(2), 278-299.
6. Chen, W. (2020). Intelligent manufacturing production line data monitoring system for industrial internet of things. *Computer communications*, 151, 31-41.
7. Alsayfi, M. S., Dahab, M. Y., Eassa, F. E., Salama, R., Haridi, S., & Al-Ghamdi, A. S. (2022). Securing real-time video surveillance data in vehicular cloud computing: A survey. *IEEE Access*, 10, 51525-51547.
8. Техніко-економічне обґрунтування інженерних рішень в інтелектуальному виробництві: підручник / І. Ш. Невлюдов. Кривий Ріг: Чернявський Д. О., 2024. 388 с.
9. Song, J., & Nang, J. (2024). Pedestrian Abnormal Behavior Detection System Using Edge–Server Architecture for Large–Scale CCTV

Environments. *Applied Sciences*, 14(11), 4615.

10. Alsayfi, M. S., Dahab, M. Y., Eassa, F. E., Salama, R., Haridi, S., & Al-Ghamdi, A. S. (2022). Securing real-time video surveillance data in vehicular cloud computing: A survey. *IEEE Access*, 10, 51525-51547.

11. Diogenes, Y., & Ozkaya, E. (2019). *Cybersecurity—Attack and Defense Strategies: Counter modern threats and employ state-of-the-art tools and techniques to protect your organization against cybercriminals*. Packt Publishing Ltd.

12. Ramesh, A., Pradhan, V., & Lamkuche, H. (2021, July). Understanding and analysing resource utilization, costing strategies and pricing models in cloud computing. In *Journal of Physics: Conference Series* (Vol. 1964, No. 4, p. 042049). IOP Publishing.

13. Dash, D., Moser, A., Feldman, S., Saliba, D., Bakaev, I., Smalbrugge, M., Costa, A. P. (2024). Focusing on provider quality measurement: continued consensus and feasibility testing of practice-based quality measures for primary care providers in long-term care. *Journal of the American Medical Directors Association*, 25(2), 189-194.

14. Pawałowski, P., Mazurek, C., Leszczuk, M., Moureaux, J. M., & Chaabouni, A. (2020). Video cloud services for hospitals: designing an end-to-end cloud service platform for medical video storage and secure access. *JMIR Biomedical Engineering*, 5(1), e18139.

15. Atencio, E., Komarizadehasl, S., Lozano-Galant, J. A., & Aguilera, M. (2022). Using RPA for performance monitoring of dynamic SHM applications. *Buildings*, 12(8), 1140.

16. Elharrouss, O., Almaadeed, N., & Al-Maadeed, S. (2021). A review of video surveillance systems. *Journal of Visual Communication and Image Representation*, 77, 103116.

17. Stan, L., Nicolescu, A. F., Pupăză, C., & Jiga, G. (2023). Digital Twin and web services for robotic deburring in intelligent manufacturing. *Journal of Intelligent Manufacturing*, 34(6), 2765-2781.

18. McCann, J., Quinn, L., McGrath, S., & Flanagan, C. (2022). Video Surveillance Architecture from the Cloud to the Edge. *International Journal for Computers & Their Applications*, 29(3).

19. Ofoeda, J., Boateng, R., & Effah, J. (2025). An institutional perspective on application programming interface development and integration. *Information Technology & People*, 38(2), 984-1016.

20. Hu, V. C., Iorga, M., Bao, W., Li, A., Li, Q., & Gouglidis, A. (2020). General access control guidance for cloud systems. *NIST Special Publication*, 800(210), 50-2ex.

21. Охорона праці та цивільний захист [Електронний ресурс] : навчальний посібник / КПІ ім. Ігоря Сікорського ; уклад.: О. Г. Левченко. Електронні текстові данні (1 файл: 7,22 Мбайт). Київ : КПІ ім. Ігоря Сікорського, 2024. 362 с.