

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ІНФОРМАЦІЙНО-
КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ
ТЕХНОЛОГІЙ
КАФЕДРА ІНФОРМАЦІЙНИХ СИСТЕМ ТА ТЕХНОЛОГІЙ

КВАЛІФІКАЦІЙНА РОБОТА

на тему:

«ПЛАТФОРМА ЦИФРОВОГО ДУБЛЮВАННЯ ЯК ІНСТРУМЕНТ
ОПТИМІЗАЦІЇ СКЛАДНИХ ПРОМИСЛОВИХ СИСТЕМ НА БАЗІ ПОТ»

на здобуття освітнього ступеня магістр

за спеціальності 126 Інформаційні системи та технології

(код, найменування спеціальності)

освітньо-професійної програми Інформаційні системи та технології

(назва)

*Кваліфікаційна робота містить результати власних досліджень.
Використання ідей, результатів і текстів інших авторів мають посилання на
відповідне джерело*

(підпис)

Даніїл ТИХОНОВ

(ім'я, ПРІЗВИЩЕ здобувача)

Виконав: здобувач вищої освіти гр.ІСДМ-61

Даніїл ТИХОНОВ

(ім'я, ПРІЗВИЩЕ)

Керівник:

к.т.н.

Олег СЕНЬКОВ

(ім'я, ПРІЗВИЩЕ)

Рецензент:

(ім'я, ПРІЗВИЩЕ)

Київ 2025

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ**

Навчально-науковий інститут Інформаційних технологій

Кафедра Кафедра Інженерії систем та технологій

Ступінь вищої освіти магістр

Спеціальність 126 Інформаційні системи та технології

Освітньо-професійна програма Інформаційні системи та технології

ЗАТВЕРДЖУЮ

Завідувач кафедру ІСТ

_____ Каміла СТОРЧАК

“ _____ ” _____ 2025 року

**ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

Тихонова Данііла Олеговича

(прізвище, ім'я, по батькові здобувача)

1. Тема кваліфікаційної роботи: Платформа цифрового дублювання як інструмент оптимізації складних промислових систем на базі IoT

керівник кваліфікаційної роботи Олег СЕНЬКОВ, к.т.н., доц.кафедри ІСТ

(ім'я, ПРІЗВИЩЕ, науковий ступінь, вчене звання)

затверджені наказом Державного університету інформаційно-комунікаційних технологій від 30 «жовтня» 2025р. №467

2. Строк подання кваліфікаційної роботи 26 «грудня» 2025 р.

3. Вихідні дані кваліфікаційної роботи:

1. Сучасні інформаційні технології.
2. Платформа цифрового двійника.
3. Експериментальні дослідження.
4. Науково-технічна література.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити):

1. Аналіз сучасного стану концепції цифрового двійника і платформи його реалізації.
2. Дослідження етапів впровадження системи цифрового двійника.
3. Експериментальні дослідження та оцінювання ефективності платформи цифрового двійника.

5. Перелік ілюстраційного матеріалу: *презентація*

6. Дата видачі завдання 30 «жовтня» 2025р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1.	Підбір технічної літератури	30.10.25-03.11.25	Виконано
2.	Аналіз сучасного стану концепції цифрового двійника	03.11.25-11.11.25	Виконано
3.	Огляд платформних рішень реалізації реалізації цифрового двійника	11.11.25-20.11.25	Виконано
4.	Дослідження етапів впровадження системи цифрового двійника.	20.11.25-26.11.25	Виконано
5.	Висновки по роботі	26.11.25-02.12.25	Виконано
6.	Експериментальні дослідження та оцінювання ефективності платформи цифрового двійника.	02.12.25-10.12.25	Виконано
7.	Розробка демонстраційних матеріалів, доповідь. Оформлення роботи	10.12.25-24.12.25	Виконано

Здобувач вищої освіти _____

(підпис)

Даніїл ТИХОНОВ

(ім'я, ПРІЗВИЩЕ)

Керівник кваліфікаційної роботи _____

(підпис)

Олег СЕНЬКОВ

(ім'я, ПРІЗВИЩЕ)

РЕФЕРАТ

Текстова частина кваліфікаційної роботи на здобуття ступня магістр: 70 стор., 25 рис., 6 табл., 23 джерела.

Мета роботи – дослідження процесу впровадження цифрових двійників в мережі IoT.

Об'єкт дослідження – процеси створення, впровадження та функціонування цифрових двійників у виробничих системах.

Предмет дослідження – методи та програмно-апаратні засоби побудови цифрового двійника

Короткий зміст роботи. Виконуючи поставлені завдання виконано дослідження концепції цифрового двійника, а також переваги його використання та застосування. Досліджено архітектуру системи, що досліджується, а саме обрано рішення для реалізації платформи ЦД. Коротко описано контекстуалізацію інших важливих складових архітектури.

Виконано детальний опис реалізацію, виконану в даній роботі. Представлені всі функціональні можливості платформи цифрового двійника.

У практичній частині роботи представлені експерименти, проведені для перевірки функціональних можливостей платформи цифрового двійника. Окрім експериментів, також описано роботу деяких виконаних процесів. За допомогою логіки BPMN описано метод впровадження цифрових двійників в промислові процеси.

КЛЮЧОВІ СЛОВА: ЦИФРОВИЙ ДВІЙНИК, ПЛАТФОРМА, ДОДАТОК, ТЕХНОЛОГІЯ, ІНТЕРНЕТ РЕЧЕЙ, ІДЕНТИФІКАТОР, ІНТЕРФЕЙС, ШТУЧНИЙ ІНТЕЛЕКТ, СЕРВЕР.

ABSTRACT

The text part of the qualifying work for obtaining a bachelor's degree: 70 pp., 25 fig., 6 tables, 23 sources.

The purpose of the work is to study the process of implementing digital twins in the IIoT network.

Object of research - processes of creation, implementation and functioning of digital twins in production systems.

Subject of research - methods and software and hardware for building a digital twin

Summary of the work. In carrying out the tasks set, a study of the concept of a digital twin was carried out, as well as the advantages of its use and application. The architecture of the system under study was studied, namely, a solution was selected for the implementation of the CD platform. The contextualization of other important components of the architecture is briefly described.

A detailed description of the implementation performed in this work was performed. All the functional capabilities of the digital twin platform are presented.

The practical part of the work presents experiments conducted to verify the functional capabilities of the digital twin platform. In addition to the experiments, the operation of some of the performed processes is also described.

KEYWORDS: DIGITAL TWIN, PLATFORM, APPLICATION, TECHNOLOGY, INTERNET OF THINGS, IDENTIFIER, INTERFACE, ARTIFICIAL INTELLIGENCE, SERVER.

ЗМІСТ

ВСТУП.....	9
1 АНАЛІЗ СУЧАСНОГО СТАНУ КОНЦЕПЦІЇ ЦИФРОВОГО ДВІЙНИКА І ПЛАТФОРМИ ЇЙОГО РЕАЛІЗАЦІЇ.....	11
1.1 Еволюція інформаційних технологій у промисловості.....	11
1.2 Архітектура та застосування цифрових двійників.....	14
1.3 Ключові технології реалізації цифрового двійника.....	20
2 ДОСЛІДЖЕННЯ ЕТАПІВ ВПРОВАДЖЕННЯ СИСТЕМИ ЦИФРОВОГО ДВІЙНИКА.....	29
2.1 Метод розробки та застосування цифрових двійників у виробництві....	29
2.2 Архітектура платформи цифрового двійника.....	33
2.3 Модель даних.....	35
2.4 Інтеграція платформи цифрового двійника.....	37
2.5 Цифрова платформа Twin Core.....	42
2.6 Валідація основних функцій.....	51
3 МЕТОД РОЗРОБКИ ТА ВПРОВАДЖЕННЯ ЦД У ВИРОБНИЦТВІ.....	55
3.1 Цифрові двійники на основі даних для виробництва.....	55
3.2 Метод впровадження цифрових двійників в промислові процеси.....	68
ВИСНОВКИ.....	73
ПЕРЕЛІК ПОСИЛАНЬ.....	75
ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ (Презентація)	78

ВСТУП

Актуальність теми. Сучасний розвиток промисловості визначається концепцією Індустрії 4.0, яка спрямована на створення інтелектуальних і взаємопов'язаних виробничих систем. Це зумовлює перехід від традиційних централізованих систем управління до розподілених кіберфізичних архітектур, де окремі компоненти є автономними, гнучкими та взаємодіють у реальному часі.

Ключовою рушійною силою цих змін став Промисловий Інтернет речей (ІоТ), що забезпечує інтеграцію фізичних пристроїв у єдину цифрову мережу, усуваючи обмеження в обміні та обробці даних. Разом із ним важливе значення мають кіберфізичні системи (КФС), які поєднують фізичні об'єкти з їхніми обчислювальними моделями, та цифрові двійники (ЦД) — віртуальні копії фізичних систем, що дозволяють здійснювати аналіз і оптимізацію процесів у реальному часі.

Поєднання концепцій ІоТ, КФС і ЦД формує основу для створення децентралізованих інтелектуальних систем управління, здатних до самоналаштування й адаптації до змін середовища. Однак практичне впровадження цих технологій у промисловості залишається обмеженим через відсутність уніфікованих платформ і недостатню гнучкість обладнання.

Таким чином, дослідження реалізації платформи цифрового дублювання на базі ІоТ є актуальним науково-практичним завданням, спрямованим на підвищення ефективності та оптимізацію складних промислових систем.

Мета роботи дослідження процесу впровадження цифрових двійників в мережі ІоТ.

Для досягнення мети, у магістерській роботі успішно виконано наступні завдання:

- аналіз сучасного стану концепції цифрового двійника і платформи його реалізації;
- дослідження етапів впровадження системи цифрового двійника;
- експериментальні дослідження та оцінювання ефективності платформи

цифрового двійника.

Об'єкт дослідження – процеси створення, впровадження та функціонування цифрових двійників у виробничих системах.

Предмет дослідження – методи та програмно-апаратні засоби побудови цифрового двійника

Методи дослідження. Для досягнення поставленої мети та реалізації основних завдань у роботі використано комплекс взаємодоповнюючих методів дослідження, а саме: аналітичні методи, методи системного аналізу, моделювання та структурного проєктування, експериментальні методи та методи порівняльного аналізу.

Наукова новизна одержаних результатів. Наукова новизна проведеного дослідження полягає у реалізації та обґрунтуванні архітектури платформи цифрового двійника, що забезпечує гнучку інкапсуляцію функцій моніторингу, візуалізації та дистанційного керування фізичними об'єктами в межах кіберфізичних виробничих систем.

На відміну від існуючих рішень, досліджувана модель передбачає узгоджену взаємодію між фізичним та віртуальним рівнями системи, що забезпечує реалістичне відтворення поведінки об'єкта у реальному часі та створює основу для подальшої автоматизації виробничих процесів.

Практична значущість одержаних результатів. Практична значущість результатів дослідження полягає у створенні діючої платформи цифрового двійника, яка може бути застосована для моніторингу та дистанційного керування технологічними процесами у виробничих системах.

Апробація результатів магістерської роботи. Основні результати та напрацювання магістерської роботи були представлені на двох наукових конференціях, проведених на базі Державного університету інформаційно-комунікаційних технологій.

1 АНАЛІЗ СУЧАСНОГО СТАНУ КОНЦЕПЦІЇ ЦИФРОВОГО ДВІЙНИКА І ПЛАТФОРМИ ЙОГО РЕАЛІЗАЦІЇ

1.1 Еволюція інформаційних технологій у промисловості

Концепція цифрового двійника (ЦД) розглядалася кілька разів, враховуючи її реальні переваги застосування в різних сценаріях. Кілька авторів намагалися та продовжують шукати способи кращого визначення ЦД, оскільки знання про нього зростають, а питання, що все ще існують, стають більш прозорими.

Один з перших підходів до концепції ЦД був зроблений у 2003 році передовим вченим-виробником Майклом Грівзом у презентації щодо управління життєвим циклом продукту в промисловості. З моменту свого використання концепція поширилася на інші можливі середовища застосування, а також отримала високий пріоритет у промисловій сфері. Основою запропонованої концепції є те, що віртуальна модель інформації, що відображає фізичну систему, може сама бути окремою сутністю.

Більш детальне визначення цифрового двійника можна знайти в [1]: «Цифровий двійник — це набір віртуальних інформаційних конструкцій, які повністю описують потенційний або фактичний фізичний вироблений продукт від мікроатомного до макрогеометричного рівня. У своєму оптимальному стані будь-яка інформація, яку можна отримати шляхом перевірки фізичного виробленого продукту, може бути отримана з його цифрового двійника». Це визначення представляє більш загальний погляд на концепцію цифрового двійника.

Аналогічно, NASA у 2012 році також запропонувало своє визначення цифрового двійника в [2]: «Цифровий двійник — це інтегроване мультифізичне, багатомасштабне, ймовірнісне моделювання побудованого транспортного засобу або системи, яке використовує найкращі доступні фізичні моделі, оновлення датчиків, щоб відобразити термін служби відповідного літаючого двійника». На відміну від визначення, запропонованого в [1], цей підхід NASA є дуже

специфічним, оскільки він оформлений у контексті її проектів, а саме розробки космічних апаратів.

Можливо зазначити, що хоча ЦД можна описати за допомогою визначення, яке охоплює його фундаментальні аспекти, найчастіше автори пов'язують та уявляють ЦД у контексті його застосування.

Поняття ЦД та CPS часто неправильно розуміють. Тому важливо розрізнити ці дві концепції, щоб зрозуміти їхній потенціал та те, як вони разом є вирішальними для досягнення розумного виробництва.

Перш ніж визначати вищезазначені концепції, також важливо зрозуміти, що таке розумне виробництво та Промисловість 4.0. Як представлено в [1], Промисловість 4.0 – це національна стратегія, спрямована на досягнення розумного виробництва. Як видно на рис.1.1, розвиток інформаційних технологій був вирішальним для входження виробництва в цифровий світ. Ці нові інформаційні технології, такі як Інтернет речей, хмарні обчислення, аналітика великих даних та інші, створили низку нових викликів, які призвели до того, що такі ініціативи, як Промисловість 4.0, стали головним пріоритетом у науковому порядку денному.

Отже, розумне виробництво використовує переваги цього поєднання нових інформаційних технологій для досягнення майбутнього стану виробництва. Таким чином, концепції ЦД та CPS можна розглядати як похідні від усіх цих ініціативних стратегій, оскільки лише завдяки їм можна реалізувати настільки бажане інтелектуальне виробництво.



Рис.1.1. Еволюція інформаційних технологій у промисловості

Щодо цих нових інформаційних технологій, одна з них потребує особливої уваги. Термін IoT відіграє важливу роль у виникненні стратегії Industry 4.0. Інтернет речей можна розглядати як глобальну мережу, в якій усі пристрої можуть взаємодіяти один з одним через Інтернет.

Таким чином, IoT має на меті створити зв'язок між кількома пристроями, щоб вони могли обмінюватися відповідною інформацією для задоволення потреб користувача. Спрямовуючи цю концепцію на сферу промисловості, термін IIoT постає як основа концепції IoT, застосованої до промислового середовища, що охоплює потреби розумного виробництва. Наразі IIoT має найбільший вплив на виробничі системи.

Враховуючи вплив впровадження IoT на промисловість, настала нова промислова ера, і концепції CPS та ЦД набули сили, оскільки вони стали важливими у розвитку виробництва. Як видно з рис. 1.1, концепції CPS та концепція ЦД знаходяться в різних галузях. Концепція CPS формально використовується в промисловості як CPDS та належить до наукової категорії, тоді як концепція ЦД належить до інженерної категорії. Як уже згадувалося, обидві концепції пов'язані з конвергенцією між фізичним та цифровим світами. Як визначено в [3], кіберфізична система визначається як система, яка за допомогою мережі Інтернету речей (IoT) збирає інформацію в режимі реального часу з фізичного ландшафту. Вся інформація, зібрана через мережу IoT для кіберпростору, аналізується та обробляється технологіями, підготовленими для оптимізації цих даних, а потім ця покращена інформація повертається до фізичного активу для вирішення проблем. ЦД розглядається як інструмент, інтегрований у CPS, який здійснює збір даних у режимі реального часу з попередньо встановленої мережі IoT та надсилає їх до кібер-сторони.

Крім того, ЦД створює синхронний канал, який пов'язує фізичний світ з кіберсвітом для досягнення мети інтеграції CPS.

1.2 Архітектура та застосування цифрових двійників

Цифровий двійник, по суті цифрове представлення фізичної сутності, процесу чи особи, контекстуалізованого у віртуальному середовищі, є ключовим інструментом для організацій для моделювання реальних сценаріїв та результатів, тим самим покращуючи можливості прийняття рішень. Як показано на рис.1.2 архітектура цифрових двійників зазвичай структурована на п'ять основних рівнів:

1. Фізичний рівень. Цей базовий рівень містить фактичні фізичні об'єкти або сутності. Він використовує сенсорну технологію для збору даних і може отримувати команди від віртуального рівня. Цей рівень забезпечує зворотний зв'язок у режимі реального часу з моделлю цифрового двійника.

2. Шар зондування даних. Цей рівень відповідає за збір різноманітної інформації та використовує різні датчики для детального моніторингу стану системи та робочого процесу. Різноманітність та різноманітність даних зумовлені різноманітними джерелами генерації даних, такими як датчики Інтернету речей, інформаційні системи та портативні пристрої.

3. Рівень передачі даних. Як ключова ланка, цей рівень забезпечує передачу даних між фізичним та віртуальним рівнями. Він використовує протоколи інтеграції зв'язку та інтерактивні технології безпеки для полегшення цієї передачі.

4. Віртуальний шар. На цьому рівні компоненти реального світу реконструюються цифровим способом. Він створює колекцію цифрових двійників, використовуючи дані, передані з фізичного рівня доповнений історичними або інтегрованими мережевими даними. Цей рівень динамічно налаштовується на основі даних реального часу з фізичного рівня та може залежати від змін, внесених на рівні додатку.

5. Рівень додатків. Цей рівень візуалізує дані та симуляції, отримані з віртуального рівня, представляючи графічну модель, яку персонал може легко інтерпретувати. Зміни параметрів фізичного або віртуального рівня можуть призвести до змін у симуляції. Потім їх можна переглянути та оптимізувати на основі спостережуваних або екстрапольованих результатів. Кожен шар у цій

п'ятирівневій архітектурі відіграє окрему, але взаємопов'язану роль, що разом дозволяє цифровому двійнику функціонувати як комплексна, динамічна система для аналізу, моделювання та вдосконалення реальних процесів і сутностей.

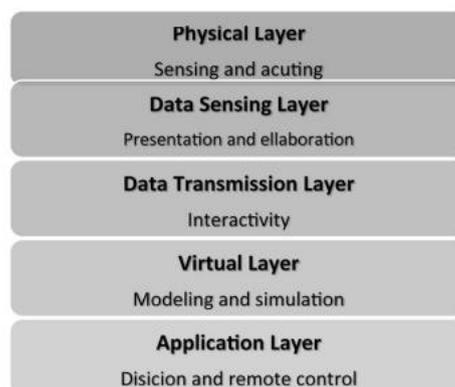


Рис.1.2 Архітектура цифрових двійників

Концепція DT не зустрічає однастайності, і існує широке розмаїття підходів через велику різноманітність варіантів використання ЦД, що призводить до пропонування численних інструментів для її впровадження відповідно до особливостей кожної системи. На даний момент, враховуючи, що концепція ЦД продовжує розвиватися та набувати корисності в кількох сферах, більшість існуючих застосувань з'являються в галузі розумних міст, енергетичного сектору, будівництва будівель, транспорту, виробництва та охорони здоров'я. У табл.1.1 представлено короткий виклад деяких застосувань ЦД, розділених на різні категорії, враховуючи огляд літератури досліджень [4, 5].

Таблиця 1.1 - Зведений огляд застосувань цифрового двійника

Категорія	Застосування
Розумні міста	-Міське планування -Розробка стратегії
Енергія	-Моніторинг та управління енергоспоживанням -Аналіз відмов -Експлуатація та обслуговування інтелектуальної мережі

Продовження таблиці 1.1 - Зведений огляд застосувань цифрового двійника

Категорія	Застосування
Будівля	-Моніторинг прогресу -Контроль та коригування бюджету -Оцінка якості будівництва -Моніторинг безпеки працівників -Розподіл ресурсів та відстеження відходів
Транспорт	-Моніторинг транспорту -Графік подорожей
Охорона здоров'я	-Моніторинг здоров'я -Персоналізована медицина -Розподіл медичних ресурсів
Виробництво	-Перевірка проекту/Планування компонування -Прогнозуюче обслуговування -Планування та контроль виробництва -Оптимізація процесів

Розумні міста. Як пояснювалося в [4], застосування ЦД у розумних містах набуло значного значення, оскільки все більше технологічних пристроїв у містах підключаються через мережі Інтернету речей. Оскільки все більше датчиків підключаються та починають збирати дані в межах міст, це дозволяє системам, що входять до їх складу, обмінюватися інформацією для боротьби з проблемами.

Головною перевагою використання ЦД у розумних містах є цінність даних, зібраних з пристроїв Інтернету речей. Ці дані можна використовувати для покращення планування та розвитку майбутніх розумних міст, а також для цілей енергозбереження. Використання технологій ЦД, таких як інструменти аналізу та моніторингу, завдяки даним, зібраним датчиками вздовж встановлених пристроїв, дозволяє створювати віртуальні моделі, що відображають поведінку міст, і таким чином тестувати сценарії та розуміти певні екологічні поведінки.

Будівництво будівель. Щодо будівельної галузі, застосування ЦД зосереджено на проблемах виявлення та діагностики несправностей у забудованому середовищі. Для кращого розуміння цілей цього підходу ЦД у будівельному секторі було запроваджено концепцію інформаційного моделювання будівель (BIM). Ці моделі відповідають вимогам проектування, продуктивності та

будівництва будівель. У [16] представлено дослідження впровадження ЦД для управління життєвим циклом будівель.

Транспортна галузь. У транспортній галузі застосування ЦД централізовано в управлінні та моніторингу проблем великомасштабних автомобільних вантажних перевезень. Ці рішення спрямовані на зменшення споживання палива завдяки інтегрованому плануванню маршрутизації та транспорту. Ці застосування для дорожніх транспортних засобів також застосовуються в деяких підходах ЦД до управління повітряним середовищем. Деякі дослідження також виявляють певні переваги в інтеграції хмарних рішень між транспортними засобами та мобільними хмарними обчисленнями.

Енергетичний сектор. У дослідженнях, проведених у [4] та [5], енергетичний сектор виявляє у впровадженні ЦД підвищення енергоефективності в різних секторах застосування. На розумних заводах впровадження ЦД може зменшити виробничі витрати та викиди парникових газів. Також, з додаванням розумних мереж у розумних містах, транспорті та інших сферах, можна виявляти та реагувати на локальні зміни у споживанні енергії.

Охорона здоров'я. Застосування ЦД у сфері охорони здоров'я все ще перебуває на початку, проте деякі переваги вже відомі. Як зазначено в [4], одна з найбільших переваг застосування ЦД для підтримки медичної допомоги пов'язана з їхнім впливом на економіку. Впровадження ЦД у цьому середовищі призводить до економічної ефективності. У більш реалістичному підході ми можемо знайти чудові способи використання ЦД у цьому середовищі, такі як створення моделі людського тіла для аналізу та моніторингу в режимі реального часу, для хірургічної підтримки або навіть для тестування ліків. Загалом, виникає багато інших застосувань як для підтримки різних проблем на даний момент, так і для прогнозування того, як діяти в майбутньому. Тому, подібно до застосувань у виробництві, прогностична функція ЦД також може бути використана в охороні здоров'я. Враховуючи таку прогностичну поведінку, поєднуючи технології штучного інтелекту та впровадження цифрового трансферу даних, також можна розширити корисність цієї програми для прогнозування обслуговування медичного

обладнання або в іншому баченні для безперервного лікування, яке не залучає безпосередньо пацієнта.

Виробництво. Застосування ЦД у виробництві має важливе значення в контексті цієї роботи, оскільки впровадження ЦД у виробничих системах є сферою, де пропонується це рішення.

Враховуючи огляд літератури, наведений у статті в журналі [7], впровадження ЦД у виробничому секторі забезпечує моделювання та реальне представлення моделей машин, вбудованих у виробничі лінії. Ці моделювання дозволяють керувати всіма процесами, що відбуваються на виробничій лінії, а також забезпечити їх постійну оптимізацію. Таким чином, шляхом моніторингу та контролю процесів, що відбуваються в системі, ЦД забезпечує більш ефективні, продуктивні та конкурентоспроможні виробничі лінії. Саме через необхідність досягнення цієї ефективності, конкурентоспроможності та продуктивності все більше заводів впроваджують такі ініціативи, як Індустрія 4.0, які надають методології та технології, здатні покращити продуктивність їхніх виробничих ліній.

У виробництві впровадження ЦД має застосування в плануванні виробництва та управлінні виробничими лініями, прогнозованому обслуговуванні обладнання та процесів виробничих ліній, вирішенні питань проектування та компонування виробничих ліній та оптимізації процесів у виробничих лініях. Схема виробничого середовища знаходить спосіб ефективного планування розподілу фізичного обладнання з метою покращення продуктивності виробничої системи в схемі обробки даних.

Як видно на рис.1.3, сферою застосування, що має основне значення, є планування та контроль виробництва. За допомогою схеми обробки даних ЦД у виробництві можна покращити процес планування замовлень на основі попередньої інформації, покращити ефективність планування за допомогою систем підтримки рішень та забезпечити автоматичний контроль та виконання замовлень.



Рис.1.3. Галузі застосування ЦД у виробництві

Сфера застосування технічного обслуговування дозволяє визначити вплив змін стану на процеси, які потрібно оцінити, визначити та оцінити профілактичні заходи з технічного обслуговування, перевірити стан фізичного обладнання на основі алгоритмів та методів, що описують фізичну сутність, а також відстежувати різні стани життєвого циклу обладнання, щоб забезпечити правильну передачу інформації та постійний ефективний моніторинг обладнання. ЦД дозволяє автоматично контролювати вплив схеми виробничої лінії на продуктивність заводу.

Концепція цифрових двійників допомагає вирішити типові проблеми, пов'язані з сучасним виробництвом, перелічені нижче:

1. Часто змінювані вимоги. Потреби клієнтів та ринків постійно змінюються, що призводить до частих змін у запитуваних кількостях та типах продукції. Щоб відобразити змінні потреби, змінюються і виробничі лінії. Отже, традиційна виробнича парадигма перейшла від жорсткої до настроюваних, гнучких та реконфігурованих систем, що призвело до того, що індивідуальні симуляційні моделі застаріли невдовзі після їх розробки.

2. Потреба в моніторингу в режимі реального часу. Виробничі системи, як зазначалося вище, зазнають частих змін для задоволення потреб, що збільшує необхідність їх постійного моніторингу в режимі реального часу. Моніторинг у

режимі реального часу дозволяє приймати кращі та більш обґрунтовані рішення, які можуть вплинути на досягнення пов'язаних з ними цілей продуктивності.

3. Складність виробничих систем. Через зміну парадигми від звичайних до реконфігурованих систем, схеми та операції повних виробничих систем виявляються надзвичайно складними. Автоматизований та базований на даних спосіб керування та оптимізації цих систем для задоволення потреб клієнтів, який можуть забезпечити цифрові двійники, допомагає вирішити цю проблему.

4. Потреба в більш економічно ефективних виробництвах. Через високу конкуренцію у виробничому секторі існує потреба в більш інноваційних ідеях для зниження витрат та підвищення економічної ефективності виробничих процесів. Підвищення економічної ефективності неможливе без точних та своєчасних імітаційних моделей різних аспектів виробничих процесів та інших пов'язаних з ними процесів. Наявність точних та своєчасних моделей дозволяє проводити більш ефективну та корисну описову, прогнозну та прескриптивну аналітику для досягнення економічної ефективності на конкретному ринку.

1.3 Ключові технології реалізації цифрового двійника

На даний момент все ще бракує об'єднуючих платформ для реалізації цифрових двійників (ЦД), оскільки існує велика різноманітність програм, які намагаються вирішити конкретні проблеми на основі конкретного випадку використання. У даному аналізі мається на меті визначити набір доступних технологій та ключових функцій, які допоможуть у реалізації ЦД з урахуванням деяких категорій. Ці категорії обрані відповідно до ключових функцій для структурування ЦД, що, у свою чергу, допоможе в його реалізації. Після огляду літератури, проведеного в [8], було визначено наступний набір функцій, які можуть бути присутніми в ЦД:

1. Канал передачі даних та з'єднання - Концепція ЦД має на меті встановити зв'язок між фізичним та цифровим каналом об'єкта. Тому функція каналу передачі даних, загалом, має на меті створити "материнську плату" між фізичним та

цифровим компонентами. Як зазначено в [8]: «Мета каналу передачі даних полягає в тому, щоб діяти як центр для всієї інформації, пов'язаної з фізичним двійником. Функція каналу передачі даних з'єднує цифрові речі один з одним і залишає цифрово-фізичне з'єднання для функції поєднання...». Для реалізації ЦД (терміну передачі даних) вкрай важливо, щоб функція каналу передачі даних була сумісною з існуючими інтернет-браузерами, щоб зробити ЦД простішим для використання людьми. З іншого боку, інша функція, функція поєднання, має на меті представити зв'язок між фізичною сутністю та її ЦД. Ця функція встановлює двосторонній інтерфейс між фізичним аналогом та цифровим світом. Інтерфейс дозволяє потік даних від фізичної частини до відповідного ЦД, або ЦД може керувати фізичною сутністю. Хоча сьогодні все ще існують сумніви щодо того, чи є фізична частина частиною ЦД, усі дослідники стверджують, що фізичний компонент є важливим. Більшість доступних публікацій припускають, що фізичний компонент можна назвати фізичним двійником, таким чином реальний аналог є паралельним DT.

2. Ідентифікатор – Функцію ідентифікатора можна розділити на цифровий ідентифікатор та фізичний ідентифікатор.

Фізичний ідентифікатор дозволяє з'єднання між фізичним об'єктом та його цифровим аналогом. Цифровий ідентифікатор є зв'язком між цифровим представленням та мережею.

3. Безпека – Порівняно з комп'ютерною безпекою, кібербезпека є новою галуззю, яка все ще намагається вирішити деякі проблеми. Кібербезпека має як свої основні проблеми численні пов'язані інформаційні системи в кіберсвіті. Функція безпеки в контексті ЦД все ще є новою проблемою. Різні дослідники стверджують, що CPPS піддаються численним загрозам.

Отже, важливо розглянути методи захисту реалізацій ЦД. Для досягнення безпеки в різних ЦД кожен складений модуль повинен бути розроблений з використанням безпечних та надійних методів.

4. Зберігання даних – Однією з фундаментальних характеристик рішення ЦД є зберігання даних. Наразі існує широкий спектр методів та місць для зберігання даних, які можна застосовувати в ЦД. Враховуючи складність ЦД, ці місця

зберігання даних відрізняються одне від одного відповідно до цієї складності. Величезні обсяги даних потребують швидких та легкодоступних баз даних. Важливо забезпечити легкий зв'язок між сховищем даних, що знаходиться в DT, та функцією каналу передачі даних.

5. Інтерфейс користувача – Інтерфейс користувача є дуже важливою особливістю ЦД. Саме через нього користувач може отримати візуальну складову відображеної системи. Графічний інтерфейс дозволяє користувачеві не лише бачити поведінку ЦД, але й, враховуючи особливості інтерфейсу, маніпулювати ЦД. Ця категорія ЦД є однією з тих, що мають найбільшу свободу реалізації, оскільки інтерфейс користувача, як правило, реалізується відповідно до потреб ЦД та змінюється залежно від різних випадків використання.

6. Моделювання – Функція моделювання стосується моделей, здатних описувати графічну, візуальну та/або числову частину фізичного об'єкта. Спочатку моделі моделювання використовувалися для надання штучно згенерованих даних цифровому аналогу фізичного активу. Ці моделі ефективно відтворювали поведінку реального світу. У певний момент моделювання плутали з самою концепцією ЦД. Однак деякі автори визначають моделювання як властивість, здатну пов'язати життєвий цикл фізичного аналога з ЦД.

7. Аналіз. Обробка даних є однією з можливостей, які має ЦД. Цю інформацію можна зібрати фізичним компонентом об'єкта або за допомогою симуляцій. Функція аналізу в рамках ЦД має на меті надати користувачеві або функції ШІ методів ЦД для прийняття рішень.

8. Штучний інтелект. Функція ШІ тісно пов'язана з функцією аналізу. Вона відповідає за прийняття рішень на основі даних, оброблених функцією аналізу. Іноді ШІ плутають з терміном машинне навчання. Хоча ці два поняття мають різні цілі. Машинне навчання – це набір алгоритмів та моделей, здатних обробляти частину даних для ШІ. З цієї причини ми можемо використовувати концепцію машинного навчання у функції аналізу. Функція ШІ в рамках реалізації ЦД є дуже корисною, оскільки вона дозволяє ЦД бути автономним у своїх рішеннях. Ця

інтелектуальна функція дозволяє ЦД виконувати завдання з власними рішеннями, отриманими в результаті глибокого навчання аналізу даних.

Цей набір функцій вводить ідею про те, що впровадження цифрового двійника є дуже складним завданням. Проектування цифрового двійника вимагає використання різних інструментів та платформ для досягнення інтеграції різних визначених функцій. Дослідження [9] пропонує розділити різні технології забезпечення та відповідні інструменти цифрового двійника за категоріями. Таким чином, дослідження розділило ключові технології забезпечення цифрового двійника, як показано на рис.1.4. Далі описано різні інструменти, розділені на категорії, до яких вони внесені, враховуючи дослідження [9].

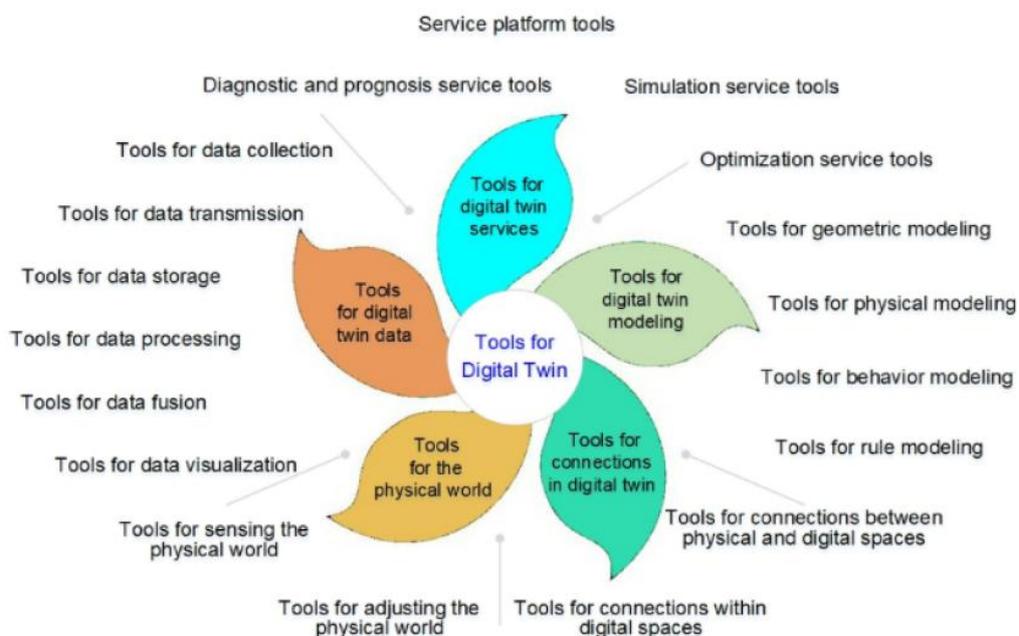


Рис.1.4.Інструменти для впровадження цифрового двійника

Інструменти для фізичного світу. Інструменти для фізичного аналога цифрового двійника можна розділити на інструменти для пізнання фізичного світу та інструменти для керування фізичним світом. Інструменти для пізнання фізичного світу призначені для сприйняття та збору даних з фізичного світу в цифровий світ з метою його оптимізації. Інструменти для керування фізичним світом надають ефективну та безпечну інформацію зворотного зв'язку фізичному

об'єкту. Ця інформація зворотного зв'язку аналізується та обробляється заздалегідь у цифровому світі. Однією з головних цілей впровадження цифрового транспорту є коригування фізичного активу, головним чином шляхом керування операціями зі зворотним зв'язком. У табл. 1.2 представлено набір різних інструментів, розділених на інструменти сенсорного керування та інструменти керування.

Таблиця 1.2 - Інструменти ЦД для фізичного світу

Інструменти для сприйняття фізичного світу	Інструменти для контролю фізичного світу
VisionPro, Visionscape, ROS, Matlab, Labview, Ali Cloud IoT, Evision, Dassault's 3D, Experience, інше програмне забезпечення	TwinCat, Codesys, Predix, MindSphere, Dassault's 3D, Experience, інше програмне забезпечення

Інструменти для моделювання цифрового двійника (ЦД). Інструменти для моделювання ЦД можна розділити на інструменти геометричного моделювання, інструменти фізичного моделювання, інструменти моделювання правил та інструменти моделювання поведінки. У табл. 1.3 ці інструменти представлені розділеними на визначені категорії. Щодо інструментів геометричного моделювання, вони призначені для опису форми, розміру, положення та зв'язку складання фізичних об'єктів з метою виконання аналізу завдань, що вимагаються від фізичного об'єкта. Інструмент фізичного моделювання використовується для побудови фізичної моделі, подібної до фізичної сутності, у геометричній моделі. Ці моделі дозволяють аналізувати стан фізичної сутності за їх допомогою. Інструменти моделювання поведінки дуже корисні для реалізації моделей, здатних відображати зовнішню поведінку фізичної сутності та, відповідно, реагувати на збурення. Крім того, ці інструменти покращують продуктивність сервісу моделювання ЦД. Це покращення також є характеристикою інструментів моделювання правил, оскільки ці інструменти моделюють закони та правила фізичної поведінки.

Таблиця 1.3 - Інструменти для моделювання ЦД

Інструменти для моделювання цифрового двійника			
Геометричні	Фізичний	Правило	Поведінкові
AutoCAD SolidWorks Ansys Twin Builder FreeCAD OpenSCAD інші	Simulink Конструктор подвійних Ansys Stella Algor ADINA інші	MindSphere Spider Matlab Toolbox TensorFlow SuerSense інші	Ansys Twin Builder 3D Max SimuWorks інші

Інструменти для даних цифрового двійника. У кожній реалізації цифрового двійника дані відповідають за транспортування інформації. Тому важливо визначити інструменти управління даними цифрового двійника. Як показано в табл.1.4, ці інструменти можна розділити на інструменти збору даних, інструменти передачі даних, інструменти зберігання даних, інструменти обробки даних, інструменти об'єднання даних та інструменти візуалізації даних.

Інструменти збору даних, як запропоновано, є інструментами управління даними, що відповідають за збір стабільних та ефективних даних із сенсорної мережі. Інструменти передачі даних мають на меті забезпечити передачу даних у режимі реального часу безпечними методами, щоб гарантувати автентичність переданих даних.

Однією з найважливіших вимог до рішення цифрового двійника є наявність модуля, здатного забезпечити зберігання даних, що проходять через нього. Таким чином, інструменти зберігання даних є важливими для реалізації цифрового двійника. Ці інструменти реалізують класифікацію та збереження даних і мають функціональність, здатну реагувати в режимі реального часу на різні події. Інструменти обробки даних відповідають за обробку отриманих даних, щоб усунути перешкоди та суперечливу інформацію.

Інструменти об'єднання даних об'єднують усі оброблені дані з інструментами обробки даних та використовують їх для майбутнього планування, перевірки та діагностики. Нарешті, інструменти візуалізації даних надають інтуїтивно зрозумілі

та чіткі дані для моніторингу в режимі реального часу та швидкого збору цільової інформації.

Таблиця 1.4 - Інструменти для управління даними DT

Інструменти управління даними цифрового двійника	
Категорія	Інструменти
Збір даних	Predix, Apache Flume, MindSphere та інші
Передача даних	MindSphere, Aspera, RaySync, інші
Зберігання даних	MySQL, MongoDB, Beacon, Cassandra, 3D Experience, Postgres, InfluxDB, інші
Обробка даних	Predix, 3D Experience, Beacon, Storm, Spark та інші
Злиття даних	Spyder, Matlab, 3D Experience, Opencv, Predix, Beacon та інші
Візуалізація даних	Excel, Діаграми, Google Chart, Highcharts, Grafana, Prometheus, інші

Інструменти для сервісів цифрових двійників. У табл. 1.5 наведено набір інструментів, доступних для сервісних застосувань. Ці інструменти поділяються на сервісні інструменти платформи, сервісні інструменти оптимізації, сервісні інструменти моделювання та сервісні інструменти діагностики та прогнозування.

Таблиця 1.5 - Інструменти для послуг цифрового трансферу даних

Інструменти застосування послуг цифрового двійника	
Категорія	Інструменти
Інструменти платформи	Beacon, 3D Experience, MindSphere, Predix, Sysware, HiaCloud та інші
Інструменти оптимізації	Beacon, Predix, Simulink, Flexsim, EnergyPlus та інші
Інструменти моделювання	Matlab, Ansys Twin Builder, Fluent/Simulink, Labview та інші
Інструменти діагностики та прогнозування	Predix, 3D Experience, Beacon, Azure IOT, Matlab, Ansys Twin Builder, інші

Як уже зазначалося, такі інструменти, як IoT, BDA, штучний інтелект та інші, надають низку послуг, що допомагають у розробці цифрових двійників. Ці сервіси мають набір інструментів, здатних реалізувати свої цілі, і їх можна назвати інструментами платформних сервісів. Інструменти сервісів оптимізації

відповідають за комплексну оцінку зібраних даних. Ці інструменти зчитують дані та запускають імітаційні моделі для оптимізації даних та внесення змін у фізичний світ. Таким чином, операції системи можна оптимізувати або контролювати під час роботи, щоб зменшити ризики, витрати та споживання енергії, а також підвищити ефективність системи. Інструменти сервісів моделювання відповідають за побудову імітаційних моделей, що використовуються для обробки даних та їх оптимізації. Нарешті, інструменти діагностики та прогнозування можуть забезпечувати інтелектуальні стратегії прогнозного обслуговування обладнання шляхом аналізу та обробки даних двійників.

Інструменти для з'єднань у цифрових двійниках. Інструменти для зв'язків у ЦД можна розділити на інструменти для зв'язку фізичного та цифрового світу або інструменти для зв'язку різних компонентів цифрового світу. У табл.1.6 ці інструменти представлені розділеними на відповідні типи: фізичний простір - цифровий простір або цифровий простір - цифровий простір.

Таблиця 1.6 - Інструменти для з'єднань у цифрових двійниках

Інструменти зв'язку в цифрових двійниках	
Фізичний простір - Цифровий простір	Цифровий простір - Цифровий простір
Predix, MindSphere від Siemens, 3D Experience від Dassault, Центр керування Jasper, ABB Ability, Lumada, HADA та інші	Siemens MindSphere, Predix, Azure IoT, 3D Experience від Dassault, Jasper Control Center, ABB Ability, Lumada, HADA та інші

Лише завдяки різним зв'язкам у ЦД можна досягти результатів для цілей, яким воно присвячене. Ці зв'язки дозволяють комунікацію, взаємодію та обмін інформацією між різними компонентами ЦД. Саме завдяки цим зв'язкам можна розробляти діагностику, вирішувати проблеми та виконувати прогнозне керування та моніторинг на основі фізичного об'єкта. Крім того, це дозволить оптимізувати продуктивність фізичного активу.

Стандарт ІЕС-61499. Як уже згадувалося, стандарт ІЕС-61499 є подальшим підходом до реалізації перспективної Індустрії 4.0. Цей стандарт описує, що фундаментальним є розробка механізму інкапсуляції програмного забезпечення,

щоб його можна було легко повторно використовувати та інтегрувати в систему. Для цього використовуються функціональні блоки (ФБ) для розподілу цих програмних модулів. Завдяки використанню фб можна розробляти різні функціональні можливості за допомогою певних алгоритмів, які потім встановлюються та розгортаються в розподіленій системі.

2 ДОСЛІДЖЕННЯ ЕТАПІВ ВПРОВАДЖЕННЯ СИСТЕМИ ЦИФРОВОГО ДВІЙНИКА

2.1 Метод розробки та застосування цифрових двійників у виробництві

Для впровадження в реальні застосування потрібен метод, який би керував процесом розробки цифрового двійника від початкової ідеї через впровадження до фази використання. Життєвий цикл цифрового двійника складається з семи фаз:

- 1 – Початок.
- 2 - Проектування та розробка.
- 3 - Верифікація та валідація.
4. Розгортання.
5. Експлуатація та моніторинг.
6. Переоцінка.
7. Завершення експлуатації.

Однак різні фази представленого ітеративного підходу не уточнюються та не дають практичних рекомендацій щодо процесу впровадження. Згідно з визначенням цифрового двійника, перші три кроки методу описують розробку та впровадження цифрового майстра як основи цифрового двійника. На четвертому кроці створюється початкова цифрова тінь шляхом створення цифрового майстра як тестового екземпляра. Зв'язок цифрового майстра з цифровою тінню за допомогою алгоритмів, моделей моделювання або кореляції формує цифрового двійника, який потім ітеративно перевіряється на відповідність початково визначеним вимогам та об'єкту.

Рисунок 2.1 показує процедуру розробки та застосування ЦД.

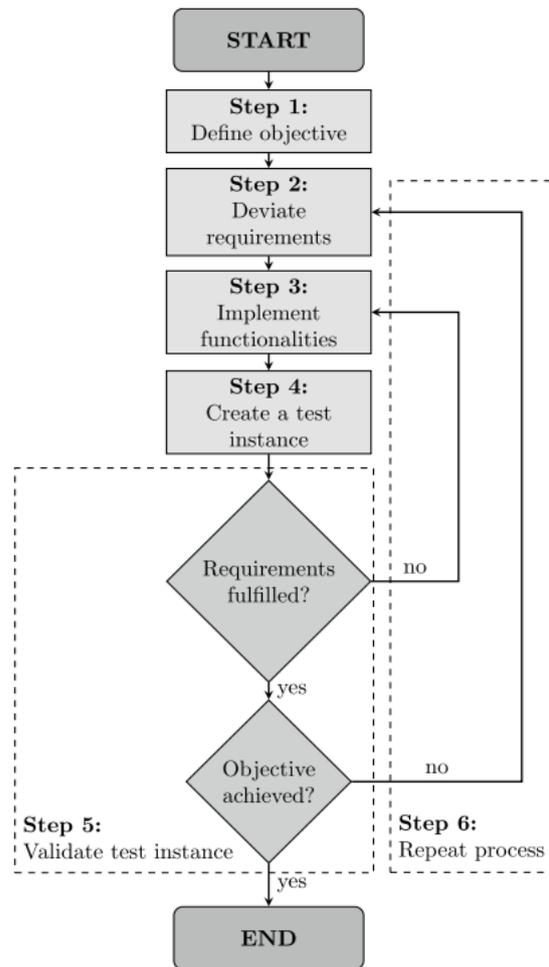


Рис.2.1. Процес розробки, впровадження, тестування та вдосконалення цифрових двійників у виробництві

Визначення мети. На першому етапі розробки цифрового двійника для виробництва основна увага приділяється точному визначенню мети або проблеми, яку має вирішити цифровий двійник. Цей початковий етап вимагає глибокого розуміння конкретних викликів і цілей у виробничому контексті. Він включає визначення основних проблем, які має вирішити цифровий двійник, а також визначення основних компонентів, систем або процесів фізичної сутності, які необхідно представити у віртуальному просторі. Це закладає основу для цілеспрямованого та ефективного цифрового представлення.

Відхилення від вимог. Переходячи до другого кроку, акцент зміщується на формулювання вимог до розробки цифрової основної версії, ключового елемента цифрового двійника. Тут завдання полягає у визначенні та документуванні основних функцій, характеристик та критеріїв продуктивності, якими цифрова основна версія повинна володіти для ефективного вирішення визначеної проблеми. Цей етап виступає керівною основою протягом усього процесу розробки, забезпечуючи відповідність результуючої цифрової версії практичним потребам виробничого сценарію. Після того, як проблему визначено та вимоги виведені, третім кроком є реалізація функціональності цифрового майстра.

Впровадження функціональних можливостей. Третій крок додатково сприяє розробці цифрового майстра. Цей етап перетворює концептуалізовану мету та вимоги на функціональну модель. Для реалізації цифрового майстра можуть бути використані різні технології. Наприклад, його можна реалізувати на ПЛК для безшовної інтеграції в системи промислової автоматизації або як код Python для розгортання на периферійному пристрої, який можна інтегрувати в різноманітні виробничі середовища. Цей етап служить ключовим мостом між концептуалізацією та застосуванням, закладаючи основу для подальших процесів оцінки та тестування під час розробки цифрового двійника.

Створення тестового екземпляра. На четвертому кроці генерується тестовий екземпляр цифрового майстра. Це може бути, наприклад, навчання нейронної мережі для моделювання поведінки машини або математичний опис системи. Це знаменує собою важливу віху в процесі впровадження цифрового двійника, створюючи цифрову тінь як перший екземпляр цифрового майстра. Завдяки зв'язку створеного таким чином екземпляра цифрової тіні з раніше визначеним цифровим майстром представлення цифрового двійника є завершеним. Цей крок включає створення тестового екземпляра, який служить репрезентативною цифровою моделлю для оцінки та валідації в реальному світі. Він діє як контрольоване середовище, де можна оцінити продуктивність та функціональність як цифрового майстра, так і цифрової тіні перед повномасштабним розгортанням.

Перевірка тестового екземпляра. П'ятий крок зосереджений на тестуванні та оцінюванні створеного тестового екземпляра. Це включає піддавання цифрового двійника різним сценаріям, умовам та вхідним даним для оцінки його чутливості та точності. Одночасно докладаються зусилля для виявлення потенційних помилок або слабких місць у реалізації, що надає цінну інформацію для ітеративного вдосконалення. Таке комплексне тестування не лише підтверджує стійкість цифрового двійника, але й відіграє вирішальну роль у підвищенні його надійності та ефективності в різних виробничих умовах.

Повторення процесу. Останній крок підкреслює ітеративний процес удосконалення, наголошуючи на вирішальній ролі коригування у відповідь на результати тестування, отримані на п'ятому кроку. Після того, як цифровий двійник піддасть різним сценаріям та оцінить його продуктивність, будь-які виявлені недоліки або відхилення від попередньо визначених вимог спонукають до систематичного перегляду рішення. Ітеративний цикл включає адаптацію реалізації на основі результатів тестування, постійне прагнення узгодити цифровий двійник із заданими критеріями. Цей процес повторюється, доки не будуть виконані всі вимоги, гарантуючи, що цифровий двійник відповідає поставленій меті. Роблячи акцент на цьому ітеративному вдосконаленні, метод сприяє постійному вдосконаленню, швидкому реагуванню та адаптивності, що зрештою призводить до створення цифрового двійника, точно налаштованого на динамічні потреби виробничих процесів.

Запропонована методологія розроблена універсальним способом, що дозволяє застосовувати її як до існуючих, так і до планованих установок. Для планованих установок методологія може бути безперешкодно інтегрована у фази проектування та розробки, що дозволяє проактивно узгоджувати цілі та вимоги цифрового двійника з архітектурою системи. Натомість, застосування методології до існуючих установок може вимагати додаткових кроків, таких як модернізація застарілих систем, інтеграція додаткових датчиків або вирішення проблем комунікації ІТ-ОТ для забезпечення сумісності з платформою цифрового двійника. Хоча методологія забезпечує структурований підхід для обох сценаріїв, користувачі

повинні враховувати конкретні обмеження своїх установок. Наприклад, існуючі системи можуть вимагати більш масштабних ітеративних налаштувань під час етапів впровадження та перевірки через існуючі обмеження інфраструктури. Враховуючи ці відмінності, методологія забезпечує адаптивність, водночас враховуючи потенційні обмеження залежно від типу установки.

2.2 Архітектура платформи цифрового двійника

Для дослідження реалізації платформи цифрового двійника була розглянута архітектура для структурування кількох цифрових двійників, які контролюються та контролюють. Ця архітектура представлена діаграмою об'єктів, показаною на рис. 2.2. Як видно, архітектура складається з трьох основних об'єктів, які описані наступним чином:

1. Цифровий двійник. Цей об'єкт є основним компонентом платформи. Як видно, цей об'єкт вважається вершиною архітектури, він має унікальну назву та відповідає за керування функціональністю фізичного обладнання.

2. Функціональність. Кілька вже розроблених рішень для обробки даних мають основне обмеження: вони відображають поведінку лише однієї фізичної сутності та не дуже гнучкі до адаптацій, необхідних у середовищі. Об'єкт "Функціональність" виникає як спосіб кращої організації потреб обробки даних. Після створення обробки даних користувачеві потрібно буде визначити, яку функціональність ця обробка даних має відображати. Функціональність дозволить моніторити системні змінні та запускати виконання певних дій. Це робить систему набагато ефективнішою, оскільки користувач зможе аналізувати лише ті сутності, які він хоче, відповідно до функціональності, яку він вирішив моніторити.

3. Пристрій. Для того, щоб платформа виконала мету відображення поведінки фізичної сутності в цифровому світі, одним із незамінних об'єктів є пристрій. Таким чином, об'єкт «Пристрій» – це фізична сутність, роботу якої потрібно контролювати.

Структурно архітектура визначає, що один ЦД може бути пов'язаний з кількома функціональними можливостями, проте одна Функціональність аналізує поведінку лише одного ЦД. Крім того, кожен ЦД також може бути підключений до кількох Пристроїв. Ця модель дозволяє одному ЦД підтримувати більше однієї можливої Функціональності, залежно від конкретного аналізу, що проводиться. Так само вона дозволяє ЦД не зосереджуватися лише на аналізі поведінки фізичної сутності, а на кількох, як задумано.

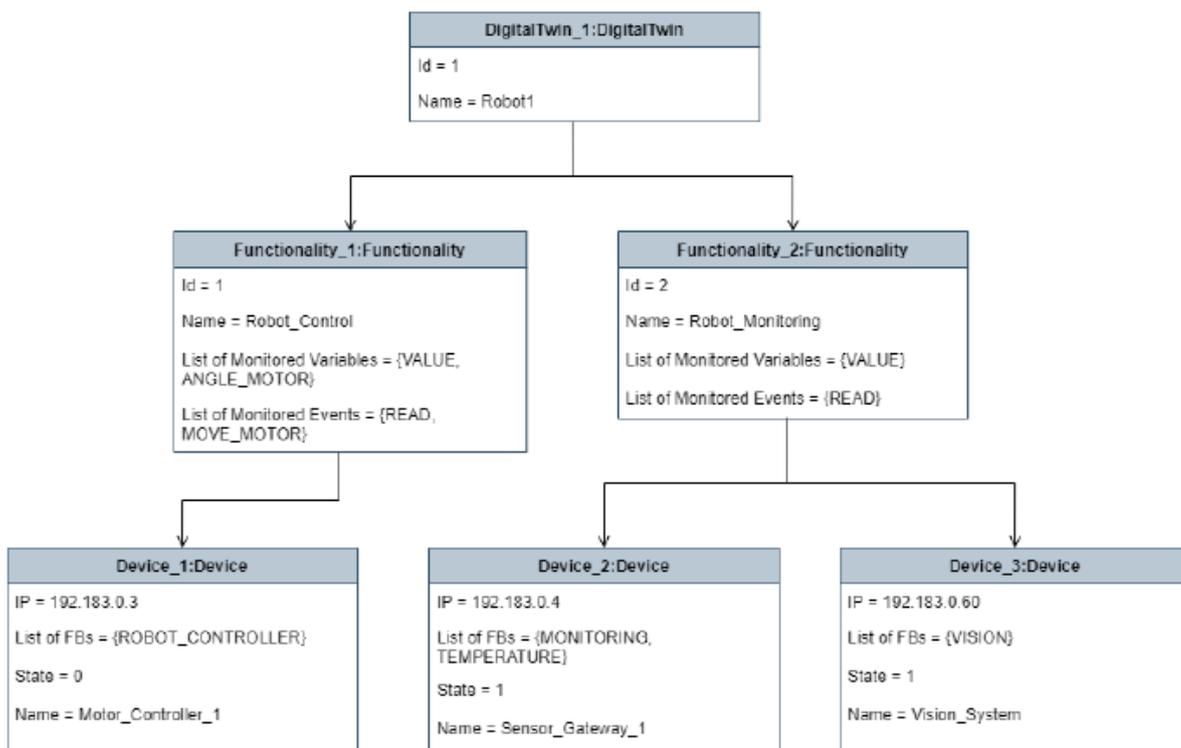


Рис.2.2. Архітектура моніторингу та керування цифровим двійником

Як видно з архітектури, представленої на рис. 2.2, вона представляє приклад системи, яку можна реалізувати на виробничій лінії. Система має лише один ЦД, проте він відповідає за складний моніторинг, як реальний випадок застосування в промисловості. У цьому прикладі DigitalTwin_1 відповідає роботу під назвою "Robot1", який має дві пов'язані функціональності: Functionality_1 відповідає керуванню роботом, а Functionality_2 відповідає за компонент моніторингу "Robot1". Кожна з функціональностей, як видно, містить список певних змінних та подій, які дозволяють виконувати різні вимоги її функцій.

До цього ЦД, "Robot1", підключено загалом три пристрої, які є інтелектуальними компонентами, відповідальними за роботу пов'язаних функціональних блоків. Однак ці пристрої не пов'язані з однаковими функціональностями. Функціональність "Robot_Control" аналізує лише один пристрій, яким у цьому випадку є "Motor_Controller_1". Таким чином, ЦД за допомогою цієї функціональності може керувати рухом "Robot1", надсилаючи позицію для переміщення. Робот розмістить двигуни у правильному положенні. Можливим прикладом такого керування може бути подія "Move_Motor", присутня у списку подій, пов'язаних з "Motor_Controller_1". З іншого боку, функціональність "Robot_Monitoring" представляє компонент, більше спрямований на моніторинг поведінки "Robot1". Ця функціональність аналізує два окремі інтелектуальні компоненти, які відповідають за різні процеси моніторингу. У випадку "Device_2" він відповідає за збір інформації з датчика, який може збирати дані про температуру системи. Натомість "Device_2", який називається "Vision_System", відповідає за компонент зору "Robot1".

Ця модель робить платформу ЦД дуже гнучкою та адаптивною в різних можливих застосуваннях.

Вона дозволяє застосовувати концепцію ЦД у більш складних системах, таких як CPS.

2.3 Модель даних

У будь-якій системі, яка вимагає високого рівня обміну та аналізу даних, спосіб збору та подальшого розподілу інформації має розглядатися як один з перших кроків впровадження, щоб зробити його максимально гнучким.

На рис.2.3 представлено модель даних платформи цифрового двійника.

Додані сутності описані наступним чином:

1. Цифровий двійник. Головною метою цієї сутності є зберігання інформації, пов'язаної з цифровими двійниками, що додаються через платформу. Таким чином, щоразу, коли новий ЦД створюється через платформу, його характеристики

зберігаються в базі даних. Серед цих характеристик є ідентифікатор ЦД та назва ЦД. Як ідентифікатор, так і назва кожного ЦД є унікальними, тому щоразу, коли користувач намагається додати новий ЦД з тим самим ідентифікатором або назвою, повертається помилка.

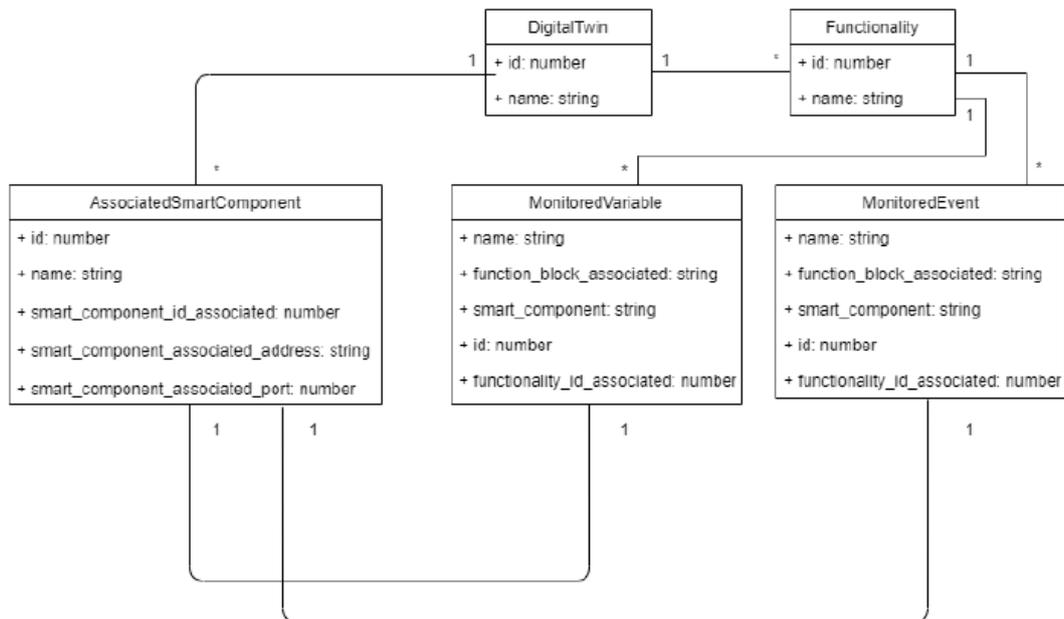


Рис.2.3. Модель даних платформи цифрового двійника

2. Функціональність. Сутність функціональності відповідає за розподіл інформації, що стосується функціональних можливостей, створених на платформі. Подібно до сутності цифрового двійника, вона також розподіляє інформацію про характеристики кожної функціональності, такі як ідентифікатор та назва функціональності. Також ідентифікатор та назва кожної функціональності є унікальними характеристиками.

3. Пов'язаний інтелектуальний компонент. Головною метою сутності пов'язаного інтелектуального компонента є розподіл інформації про інтелектуальні компоненти, пов'язані з кожним ЦД. Під час створення ЦД одним із основних полів, яке користувач повинен надати, є дані про пристрої, які будуть пов'язані з цим новим ЦД, тобто які інтелектуальні компоненти будуть контролюватися ЦД. Тому необхідно виділити інформацію про кожен інтелектуальний компонент, пов'язаний з ЦД. Отже, сутність складається з унікального ідентифікатора, назви, адреси та

порту інтелектуального компонента, який пов'язується. Крім того, також зберігається ідентифікатор ЦД, що відповідає йому.

4. Контрольована змінна. Однією з головних особливостей платформи ЦД є її компонент моніторингу в режимі реального часу. Користувач має можливість маніпулювати цим компонентом моніторингу, вибираючи змінні та/або події, які він хоче проаналізувати. Тому необхідно мати сутність, відповідальну за розподіл інформації щодо цих змінних та/або подій. Сутність "Контрольована змінна" складається лише з даних, що стосуються змінних, які додаються користувачем через платформу. Серед цих даних сутність виділяє унікальний ідентифікатор, назву, пов'язаний функціональний блок, ідентифікатор пов'язаної функціональності та пов'язаний інтелектуальний компонент для кожної змінної, що підлягає моніторингу.

5. Контрольована подія – ця сутність подібна до сутності «Контрольована змінна», проте вона відповідає за розподіл інформації, що стосується подій, які має моніторити платформа.

2.4 Інтеграція платформи цифрового двійника

Рішення, представлене раніше, представляє платформу ЦД, інтегровану в програму. Ідея полягає в повторному використанні структури програми, використовуючи деякі реалізовані компоненти, під час побудови платформи ЦД. Для цього було створено набір деяких інтеграцій, які описані в наступних розділах.

API. Серверна частина програми надає HTTP REST API, відповідальний за обробку всіх HTTP-запитів, що виконуються в програмі. Програма вже має у своєму складі набір окремих модулів API, які забезпечували контроль списку запитів. Окремі модулі API, що надаються програмою - це модуль FB API та модуль Smart Component API. Модуль FB API відповідає за всі операції управління FB у програмі, а кінцева точка модуля Smart Component API відповідає за з'єднання з DINA.

Подібно до програми, необхідно було надати набір кінцевих точок, відповідальних за підтримку нових функцій платформи цифрового двійника. Ці

кінцеві точки дозволяють виконувати всі операції з управління цифровим двійником. Таким чином, було створено новий модуль API під назвою «Модуль цифрового двійника API». Кінцеві точки цього нового модуля описані нижче.

1. Кінцеві точки модуля цифрового двійника API

- GET /digital-twin/: Ця кінцева точка відповідає за отримання списку всіх цифрових двійників, створених з веб-програми. Через цей список користувач може отримати доступ до всіх цифрових двійників, які все ще існують і які може проаналізувати платформа цифрового двійника.

- POST /digital-twin/: Через цю кінцеву точку API створює новий цифровий двійник на платформі.

Для його створення користувачеві потрібно визначити набір аргументів нового цифрового двійника, як зазначено в моделі даних, представлений на рисунку 4.3, стосовно сутності цифрового двійника.

- PUT /digital-twin/:id: Ще однією функцією, що надається API, є можливість редагування певного цифрового двійника. Через цю кінцеву точку можна редагувати DT, використовуючи ідентифікатор DT, що міститься в параметрах URL-адреси. Усі аргументи, присутні в сутності цифрового двійника, оновлюються за допомогою цієї кінцевої точки.

- GET /functionality/: Ця кінцева точка повертає список усіх існуючих функціональних можливостей у додатку. Список потім можна використовувати у веб-додатку.

- POST /functionality/: Використовуючи цю кінцеву точку, користувач може створити нову функціональність на платформі DT. Для успішного створення нової функціональності необхідно надати API необхідні аргументи сутності функціональності, як показано на рис.2.3.

- PUT /functionality/:id: Ця кінцева точка дозволяє користувачеві редагувати певну функціональність. Вона використовує ідентифікатор функції, присутній як параметр в URL-адресі.

- DELETE /functionality/:id: Окрім можливості редагування певної функціональності, через цю кінцеву точку DELETE можна видалити функціональність з веб-додатку.

Подібно до випадку редагування, функціональність, яку потрібно видалити, визначається параметром, присутнім в URL-адресі.

- GET /associated-smart-component/: Ця кінцева точка відповідає за повернення списку з інформацією про Smart Components, пов'язані з різними ЦД.

- POST /associated-smart-component/: Через цю кінцеву точку можна створити пов'язаний Smart Component у застосунку.

- GET /monitored-variable/: З цієї кінцевої точки можна отримати список усіх змінних, які будуть контролюватися платформою ЦД.

- POST /monitored-variable/: Ця кінцева точка відповідає за створення нової змінної, яку потрібно контролювати. Для успішного створення нової змінної на платформі поля сутності Monitored Variable, присутньої в моделі даних, показаній на рис.2.3, повинні бути належним чином заповнені.

- DELETE /monitored-variable/:id: За допомогою цієї кінцевої точки можна видалити певну змінну з платформи. Ідентифікатор змінної надається параметром URL.

- GET /monitored-event/: З цієї кінцевої точки можна отримати список усіх подій, які мають відстежуватися платформою DT.

- POST /monitored-event/: Ця кінцева точка відповідає за створення нової події для відстеження. Для успішного створення нової події на платформі поля сутності «Відстежувана подія», присутні в моделі даних, показаній на рисунку 4.3, повинні бути належним чином заповнені.

- DELETE /monitored-event/:id: За допомогою цієї кінцевої точки можна видалити певну подію з платформи. Ідентифікатор змінної надається параметром URL.

Контролери. Спосіб організації системи застосунком полягав у поділі її на кілька компонентів та реалізації контролерів, відповідальних за керування всіма

необхідними операціями. Як згадувалося, реалізованими контролерами були контролер FB та головний контролер інтелектуального компонента.

Враховуючи основні цілі платформи ЦД, можна зрозуміти, що це також не простий компонент, і тому була отримана логіка розробки контролера, здатного керувати його функціями. Таким чином, було реалізовано контролер ЦД. Загалом, цей контролер досить схожий на контролер FB, про який йшлося раніше в цій роботі. Контролер ЦД розроблений як одноелементний клас з набором різних методів, відповідальних за забезпечення різного керування ЦД у застосунку. Методи, які підтримує контролер наступні:

- методи створення, оновлення, видалення та отримання ЦД;
- методи створення, оновлення, видалення та отримання функціональних можливостей;
- методи створення та отримання пов'язаних інтелектуальних компонентів платформи ЦД;
- методи створення, видалення та отримання контрольованих змінних платформи ЦД;
- методи створення, видалення та отримання контрольованих подій платформи ЦД.

Цей контролер дуже важливий для підтримки платформи ЦД, оскільки він відповідає за організацію різних ЦД платформи та їхніх відповідних характеристик. Щоб зрозуміти структуру цього нового контролера, діаграми компонентів, представлені раніше, розширені новим контролером ЦД. Ця діаграма представлена на рис.2.4.

Моделі зв'язку. Під час побудови застосунку, для підтримки роботи HTTP REST API, було реалізовано набір моделей, які відповідають за те, що є репліками моделей бази даних. Це дозволяє, коли інформація щодо кількох сутностей надходить на сервер застосунку, легко маніпулювати нею та доставляти її користувачам через веб-застосунок. Загалом, саме через ці моделі API отримані дані належним чином групуються різними інтерфейсами, подібно до того, як вони

були розподілені в базі даних. Реалізація цих моделей здійснюється за допомогою типізованої мови програмування Typescript.

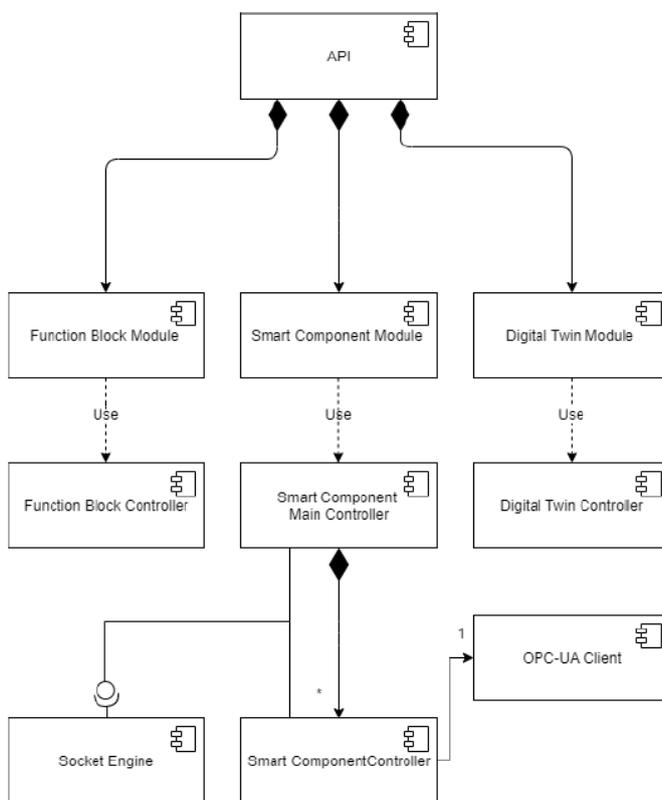


Рис.2.4. Розширена діаграма компонентів з контролером ЦД

Під час інтеграції платформи ЦД у застосунок одним із принципів була спроба відтворити методи реалізації різних компонентів застосунку, щоб зробити платформу ЦД послідовною та ефективною адаптацією. Тому, подібно до реалізації в застосунку, створено набір інтерфейсів, що відповідають потребам платформи ЦД. Ці інтерфейси містять набір полів, що стосуються характеристик кожної моделі та зв'язків між ними. Потім для платформи ЦД реалізуються наступні моделі:

- 1.Цифровий двійник.
- 2.Функціональність.
- 3.Контрольована змінна.
- 4.Контрольована подія.
- 5.Пов'язаний смарт-компонент

Важливо підкреслити, що не лише REST API використовуватиме ці моделі для обміну даними, але й дані, якими обмінюватимуться через зв'язок WebSockets (через Socket Engine), використовуватимуть ці моделі як підтримку. Таким чином, можна зрозуміти, що вся комунікація між різними компонентами системи використовуватиме ці моделі в процесах обміну інформацією.

2.5 Цифрова платформа Twin Core

Коли йдеться про ЦД, одразу ж асоціюється концепція комунікації в реальному часі. Головною метою впровадження ЦД є відображення поведінки фізичної сутності в реальному часі.

Той факт, що цю поведінку можна аналізувати в реальному часі, є основною причиною, чому впровадження ЦД у найрізноманітніших сферах є таким бажаним. У промисловому контексті, цей фактор є основним аспектом, який слід враховувати, оскільки моніторинг та керування виробничими лініями в реальному часі є ключем до підвищення ефективності систем.

З огляду на ці міркування, забезпечення підтримки платформою ЦД цього застосунку комунікації в реальному часі є важливим фактором. Як уже згадувалося, застосунок має систему комунікації в реальному часі, яка розділена на дві частини з урахуванням компонентів, що використовують цей комунікацію. Коли комунікація в реальному часі відбувається між сервером вузла застосунку та веб-застосунком, використовується протокол WebSockets. З іншого боку, коли учасниками є Node Server та Smart Components, підключені до програми, використовується протокол OPC-UA. Оцінюючи тип даних, які ЦД повинен отримувати в режимі реального часу, легко зрозуміти, що платформі знадобиться як зв'язок WebSockets, щоб зробити інформацію доступною для користувача через графічний інтерфейс, так і зв'язок OPC-UA для збору даних з пристроїв для компонентів моніторингу та керування. Починаючи з інтеграції, виконаної на рівні зв'язку в режимі реального часу між Node Server та веб-програмою, платформа ЦД скористалася тим фактом,

що вже існує системний компонент, здатний підтримувати такий зв'язок, - протокол WebSockets.

Як видно на діаграмі на рис.2.4, одним із серверних компонентів є Socket Engine, який забезпечує всі функції зв'язку на стороні сервера, що виконуються Sockets. На стороні клієнта інтерфейс служби Socket, є ключем до випромінювання та отримання певних подій. На більш конкретному рівні, до Socket Engine та Socket Service Interface потрібно було додати набір функцій, щоб забезпечити виконання наступних вимог платформи ЦД:

- запит до сервера моніторингу Smart Component, що аналізується платформою, та подальша відповідь із необхідною інформацією;
- дозвіл веб-застосунку легко запускати певну дію, тобто, за допомогою функції, присутньої в графічному інтерфейсі застосунку, подія запускається користувачем.

Компонент моніторингу цифрового двійника.

Одна з головних вимог архітектури передбачає наявність у платформі ЦД компонента моніторингу, здатного відображати роботу фізичних сутностей, якими в цьому випадку є інтелектуальні компоненти, та направляти цю інформацію до візуального компонента платформи.

Враховуючи цю вимогу, необхідно адаптувати різні вже існуючі системні компоненти в додатку, а також створити деякі функції, які б дозволили весь процес запиту моніторингу змінних аж до відповіді з цією інформацією користувачеві.

Ця інтеграція включала внесення змін та додавання деяких функцій в інтерфейс Socket Service Interface (на стороні клієнта), в Socket Engine (бекенд додатку), а також у контролери, відповідальні за моніторинг інтелектуальних компонентів.

Система ЦД організована набором сутностей: сутністю ЦД, сутністю функціональності та сутністю пристрою. Загалом, користувач у певний момент може аналізувати ЦД, який, у свою чергу, належить до категорії моніторингу, функціональності. Крім того, цей ЦД може аналізувати різноманітний набір пристроїв, іншими словами, інтелектуальні компоненти. Саме з цієї архітектури

виникає компонент моніторингу, оскільки ці інтелектуальні компоненти, інформація про які збирається, розміщують набір функціональних блоків, які мають змінні та події, здатні допомогти користувачеві зрозуміти реальний контекст того, що відбувається. Тому перша інтеграція, яку необхідно виконати, полягає в тому, щоб дозволити користувачеві легко створювати різні ЦД та функціональні можливості, а також визначати в цих ЦД, які пристрої будуть аналізуватися, і, нарешті, вибирати змінні та події, які будуть моніторитися. Однак, хоча значна частина управління ЦД здійснюється користувачем через різні компоненти веб-застосунку, багато процесів мають непряму дію з боку користувача.

Компонент керування цифровим двійником.

Компонент керування є одним із найважливіших компонентів платформи цифрового двійника. Окрім того, що це компонент, який надає велику цінність додатку з точки зору користувача, цей компонент має велику корисність у додатку на виробничих лініях.

Графічний інтерфейс цифрового двійника

Одним із найважливіших компонентів будь-якої платформи цифрового двійника є її графічний інтерфейс, що відповідає за взаємодію між користувачем та середовищем цифрового двійника. Нище представлено приклад сторінок графічного інтерфейсу у веб-додатку для керування цифровим двійником.

Загалом, головною метою компонента візуалізації платформи цифрового двійника є надання користувачеві можливості створювати та маніпулювати цифровими двійниками, які можуть виконувати набір основних функцій. Таким чином, платформа дозволяє користувачеві групувати пристрої, які він хоче контролювати та керувати, у категорію під назвою цифровий двійник, і згодом він може пов'язати цей цифровий двійник із загальною функціональністю. Оскільки платформа цифрового двійника побудована на основі додатку, і оскільки вона вже має веб-додаток, розроблений за допомогою фреймворку React, підхід, який використовувався для побудови компонентів інтерфейсу користувача, буде повторно використаний для розробки компонентів платформи цифрового двійника. Веб-додаток має компоненти інтерфейсу користувача, які складають додаток, і ці

компоненти організовані в папки відповідно до основних функцій, яким задовольняє набір елементів інтерфейсу користувача. Таким чином, для елементів інтерфейсу користувача платформи ЦД створено папку «Цифровий двійник», яка містить усі елементи інтерфейсу, відповідальні за функції управління ЦД. Слід зазначити, що всі інші папки, створені раніше для задоволення потреб управління функціональними функціями та інтелектуальними компонентами торгового майданчика додатку, також були повторно використані. З точки зору користувача, ці елементи інтерфейсу, згруповані в папці «Цифровий двійник», аналізуються як окремі веб-сторінки, де кожен користувач може інтерактивно користуватися ними.

Сторінка «Новий цифровий двійник». Основною метою сторінки «Новий цифровий двійник» є дозволити користувачеві додавати новий ЦД із певною назвою та пов'язувати інтелектуальні компоненти з цим новим ЦД. Макет сторінки показано на рис.2.5. Сторінка має меню для створення нового ЦД, і в цьому меню користувач має доступ до всіх пристроїв, які взаємодіють із застосунком в режимі реального часу, як видно на рис.2.5. Таким чином, щоб створити новий ЦД, користувачеві потрібно лише заповнити поле, щоб вставити назву нового ЦД та відкрити список доступних пристроїв і вибрати ті, які він хоче пов'язати. Слід зазначити, що користувач може вибрати більше одного смарт-компонента, як показано на рис. 2.7. Для цього користувачеві потрібно лише додати різні пристрої за допомогою кнопки «Додати».

Рис. 2.5. Сторінка «Новий цифровий двійник»

Натиснувши кнопку «підтвердити», користувач може створити новий цифровий двійник, лише якщо він вибрав хоча б один смарт-компонент. Важливо

зазначити, що якщо користувач не вкаже коректну назву для цифрового двійника або не вибере смарт-компонент, на сторінці з'явиться сповіщення про помилку. Після заповнення обов'язкових полів та натискання кнопки для підтвердження нового цифрового двійника на сторінці з'явиться сповіщення про успіх або помилку.

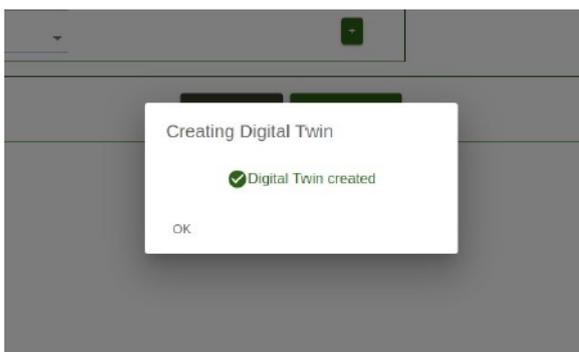


Рис. 2.6. Меню для створення нового заповненого ОУ

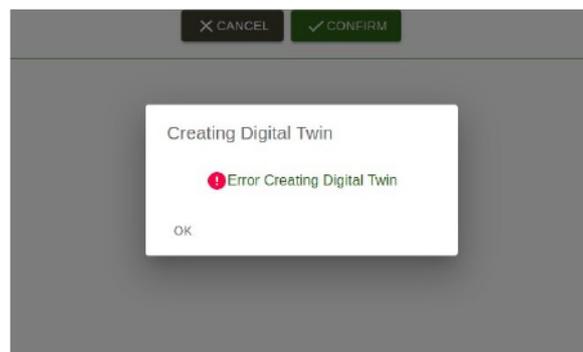


Рис.2.7. Поля для заповнення меню для створення ЦД

Сторінка моніторингу цифрового двійника є фундаментальною з точки зору користувача, оскільки саме з неї здійснюється все керування ЦД. Як показано на рис.2.9, на цій сторінці користувач може спостерігати за функціями, які наразі активні з відповідним ЦД, до якого вони підключені. Окрім можливості спостерігати за активними функціями, на цій сторінці користувач також може створювати інші функції та пов'язувати ЦД, здатний спостерігати за змінними та/або подіями, що цікавлять.



а). Поле успішного створення нового ЦД



б). Поле помилки створення нового ЦД

Рис 2.8. Вікно сповіщення, що повертається під час створення нового ЦД



Рис.2.9. Сторінка моніторингу цифрового двійника

Спочатку, розглянувши функцію сторінки, що стосується створення нової функціональності, користувач може вибрати назву функціональності, яку він хоче додати до платформи моніторингу. Ця назва має бути унікальною. Після вибору назви користувач також повинен вибрати одну ЦД з діапазону доступних ЦД, які були раніше створені на сторінці «Новий цифровий двійник», яку він хоче пов'язати з новою функціональністю. Таким чином, нова функціональність матиме лише одну пов'язану ЦД. Однак на цій сторінці можна спостерігати кілька функціональних можливостей, які можуть мати ті самі пов'язані ЦД, але з різними змінними та/або подіями, що використовуються для виконання різних цілей. Після заповнення необхідних полів для створення нової функціональності вам потрібно лише натиснути кнопку «Додати функціональність». Невдовзі після цього натискання з'явиться вікно підтвердження, щоб переконатися, що це дійсно бажання користувача. На рис. 2.10 показано вікно підтвердження, згадане вище.

Крім того, після перевірки створення з'явиться сповіщення про успіх або помилку, щоб зрозуміти, чи функціональність була створена належним чином. Ці сповіщення представлені на рис.2.11.

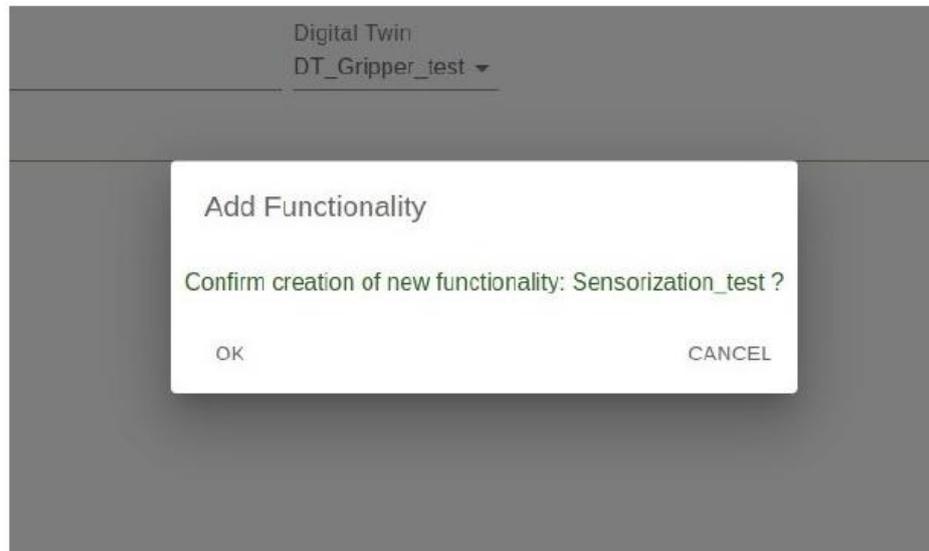
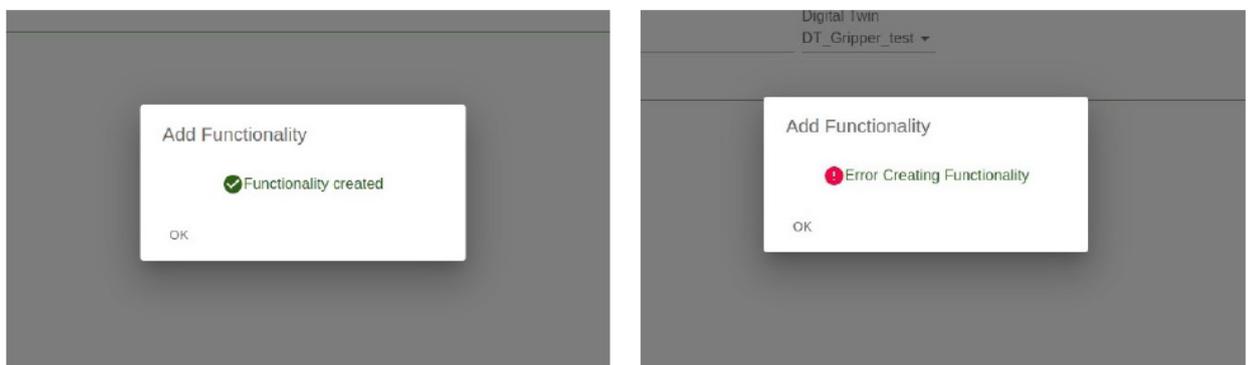


Рис. 2.10. Вікно підтвердження створення нової функціональності



а). Поле успішного створення нової функціональності

б). Поле помилки створення нової функціональності

Рис. 2.11. Повідомлення про помилку під час створення нової функціональності

На цій сторінці користувач матиме у своєму розпорядженні таблицю зі списком усіх активних на даний момент функцій та набором можливостей. Ці функції представлені на рис. 2.12 та описані в наступних темах:

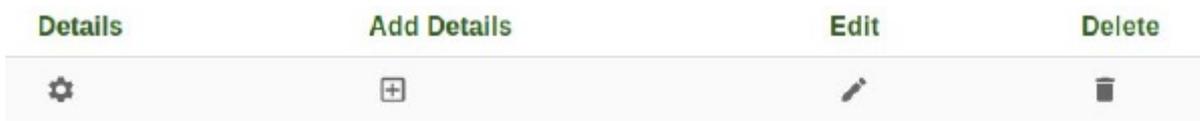


Рис. 2.12. Функції сторінки моніторингу цифрового двійника

1. Деталі. Ця функція дозволяє користувачеві детальніше переглядати інформацію, яка збирається зі змінних та подій, що наразі відстежуються. Натискаючи кнопку «деталі функції», користувач перенаправляється на нову сторінку, де відображається інформація.

2. Додати деталі. Одна з головних переваг веб-застосунку полягає в можливості вибору користувачем змінних та подій, які він хоче відстежувати на заданому FB. За допомогою кнопки «додати деталі» інтерфейс повертає поле даних, де користувач може ввести інформацію, яку він хоче переглянути. На рис.2.13 показано доступні поля.

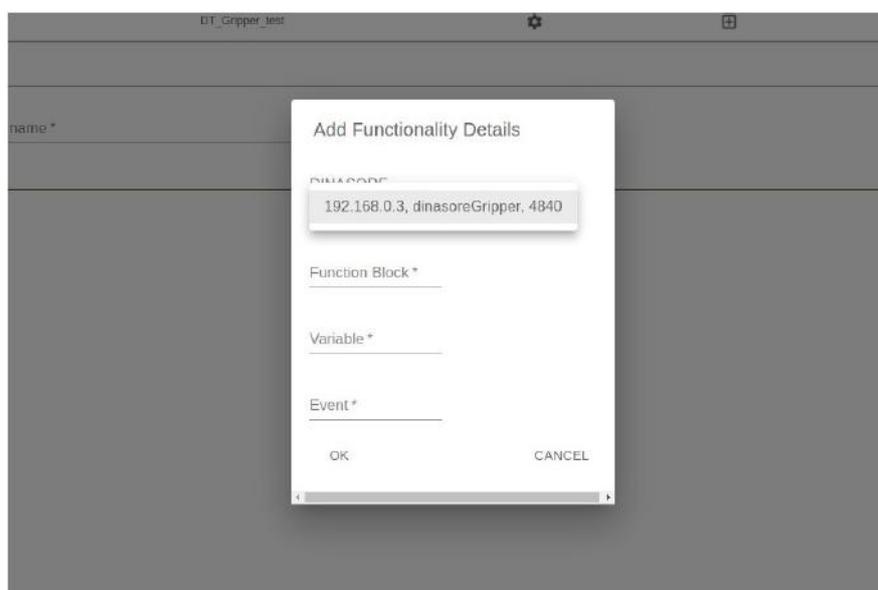


Рис.2.13 Поле додавання деталей

3. Редагувати. Подібно до функції «додати деталі», для кожної доступної функціональності в таблиці можна редагувати назву функціональності. Після натискання кнопки редагування функціональності на сторінці з'являється діалогове

вікно, як показано на рис. 2.14. У цьому полі користувач може редагувати назву функціональності.

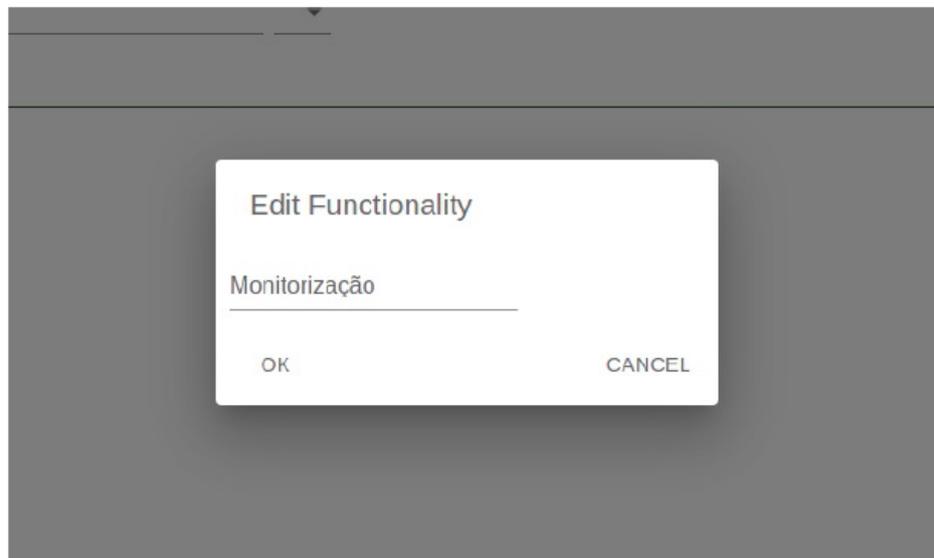


Рис. 2.14 Поле редагування функціональності

4. Видалити. Окрім можливості редагування та маніпулювання змінними та подіями, які відстежуватиме кожна функціональність, користувач також може видалити доступні функціональні можливості, які він хоче, відповідно до потреб на даний момент.

Сторінка з детальною інформацією про функціональність. Однією з головних функцій сторінки моніторингу цифрових двійників є можливість спостерігати за детальною інформацією про функціональність, а саме моніторинг змінних, які відображаються в інтелектуальних компонентах, підключених до програми, та керування подіями, також доступними в цих інтелектуальних компонентах. Тому, щоразу, коли користувач натискає кнопку з детальною інформацією про функціональність, інтерфейс перенаправляє на сторінку, подібну до тієї, що показана на рис.2.15.

Variable	Function Block	Smart Component	Current Value	Delete
COST	ENERGY_COSTS_1	dtwars1	1203.816007422025	🗑️
TEMPERATURE	OPTIMIZE_ENERGY	dtwars1	1	🗑️

Event	Function Block	Smart Component	Trigger Event	Delete
READ	ENERGY_COSTS	dtwars0	↓	🗑️

Рис. 2.15. Сторінка з детальною інформацією про функціональність

Сторінка містить дві різні таблиці: одну для моніторингу змінних, а іншу для ініціювання подій. Зверніть увагу, що на сторінці відображаються лише ті змінні та події, які були попередньо вибрані на сторінці моніторингу цифрового двійника. Таким чином, користувач має у своєму розпорядженні інформацію, організовану або за змінними, або за подіями.

2.6 Валідація основних функцій

Моніторинг змінних. Процес моніторингу змінної з платформи DT починається, коли користувач додає змінну до функціональності за потреби. На сторінці моніторингу цифрового двійника, користувач може додати потрібну змінну або подію з меню додавання деталей до функціональності. У цьому меню користувач вказує деякі поля, такі як назва змінної, назва події (якщо користувач хоче ініціювати певну дію), пов'язаний FB, а також смарт-компонент, де виконується моніторинг. Після того, як користувач додає змінні, які потрібно моніторити, вони надсилаються до бази даних разом з детальною інформацією. З моменту появи змінних для моніторингу починається процес моніторингу. Після того, як користувач натискає кнопку деталей функціональності, яку потрібно моніторити, його перенаправляють на сторінку "Деталі функціональності", яка містить інформацію про змінні та події, які потрібно моніторити. Щоразу, коли ця сторінка відображається, між веб-додатком та серверною частиною встановлюється зв'язок через WebSockets. Поряд з цим з'єднанням веб-додаток надсилає запит на відображення змінних, які розподілені в базі даних. Коли серверна частина отримує

запит на зчитування змінних, що підлягають моніторингу, вона отримує з бази даних інформацію про всі змінні, що підлягають моніторингу, та фільтрує їх. Під час цієї фільтрації змінні, що підлягають моніторингу, організовуються смарт-компонентом, з яким вони пов'язані. Після організації змінних інформація про них розподіляється в масиви. Після того, як інформація про змінні, відфільтрована та організована кожним смарт-компонентом, створюється запит на підписку через бібліотеку OPC-UA. Ця підписка дозволяє встановити надійний зв'язок між серверною частиною додатку та OPC-зв'язком, а також дозволяє серверній частині прослуховувати всі зміни її значень. Таким чином, щоразу, коли відбувається зміна значення підписаної змінної, серверна частина повідомляється про це, і вона пересилає цю ж інформацію через WebSockets до графічного інтерфейсу.

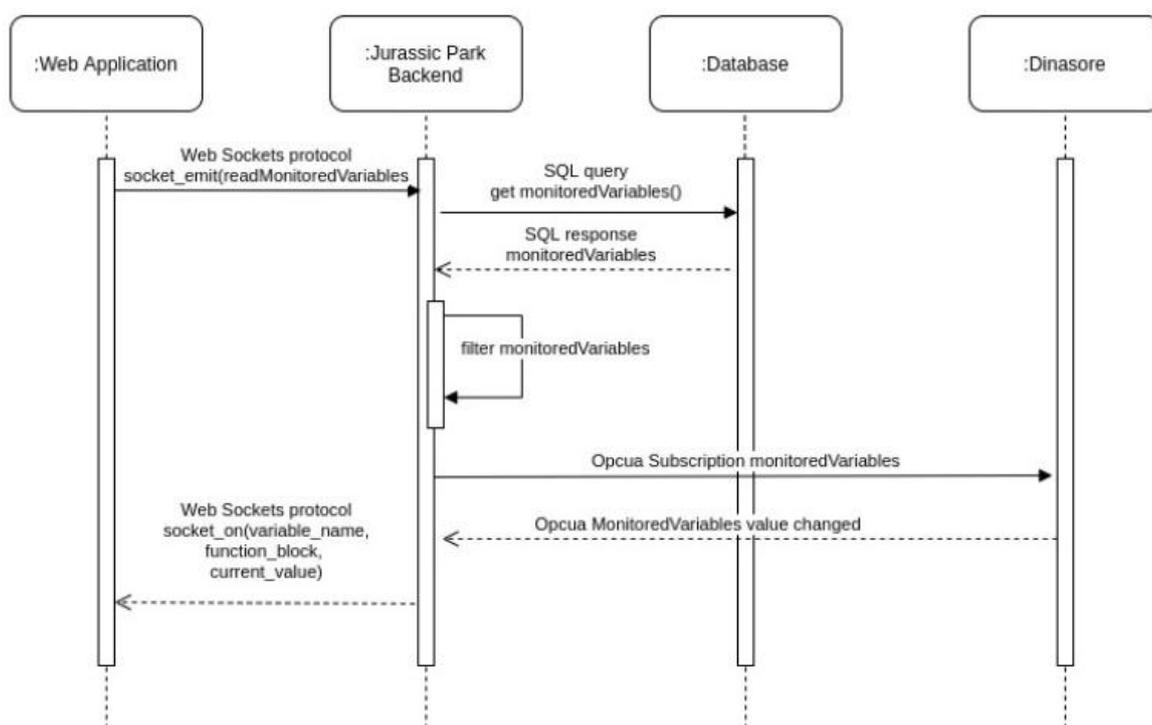


Рис.2.16. Моніторинг змінної послідовності

Важливим зауваженням щодо моніторингу змінних є те, що між зміною значення змінної та виявленням цієї зміни бібліотекою OPC-UA, що використовується додатком, була виявлена невелика затримка. Як раніше зазначалось, бібліотекою, що використовувалася на стороні сервера для збору

інформації з інтелектуальних компонентів, підключених до платформи, була Node-Орсуа. Ця бібліотека використана, оскільки вона єдина, яка підтримує ту саму мову, що й Node Server додатку, мову JavaScript. Інші можливі бібліотеки, що підтримують протокол OPC-UA, розроблені іншими мовами, можна знайти в [13]. Незважаючи на виявлену мінімальну затримку, зміна бібліотеки для цього зв'язку в режимі реального часу не вважалася корисною, оскільки це означало б дуже значні зміни в інших компонентах системи.

Запуск події. Процес запуску події ініціалізується з моменту натискання користувачем кнопки запуску події для певної події на сторінці «Деталі функціональності» для певної функціональності. Коли кнопка натискається, веб-застосунок зчитує інформацію про подію, тобто назву події, функціональний банк, до якого вона належить, та смарт-компонент, на якому вона розміщена.

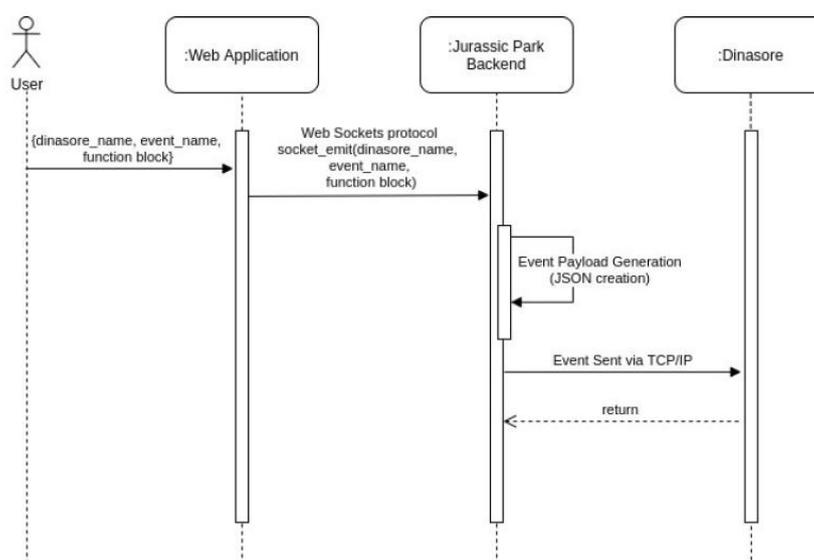


Рис.2.17. Запуск послідовності подій

Ця інформація надсилається через WebSockets до серверної частини застосунку. У серверній частині сокет-зв'язок готується до отримання інформації про подію, яку потрібно запустити. Після отримання наступним кроком є створення JSON-файлу з інформацією про подію, яка буде надіслана до пов'язаного смарт-компонента. JSON-файл складається з інформації, пов'язаної з подією та смарт-

компонентом, на якому подія запускається. Тому для створення файлу важлива інформація, що надходить з графічного інтерфейсу. IP-адреса та динамічний порт смарт-компонента також потрібні для JSON-файлу. Ця інформація береться з OPC-UA-зв'язку шляхом порівняння з також отриманими параметрами. Після створення JSON-файлу він надсилається через TCP/IP до відповідного смарт-компонента.

3 МЕТОД РОЗРОБКИ ТА ВПРОВАДЖЕННЯ ЦД У ВИРОБНИЦТВІ

3.1 Цифрові двійники на основі даних для виробництва

Ідея цифрового двійника, керованого даними, мотивована постійно зростаючими обсягами даних, що генеруються на виробництвах. Акцент робиться на даних, оскільки вони є ключовим елементом, який можна використовувати для визначення та моніторингу станів складних систем у заводському середовищі. Ефективне використання цих даних допоможе в розробці високоточних моделей для виробничих систем. Інструменти машинного навчання та аналізу процесів є ключовими методами, які можна використовувати для отримання знань з таких даних.

Їхня роль у моделюванні, а отже, і в структурі цифрового двійника, полягатиме в розширенні можливостей моделі за допомогою отриманих знань. Ці знання будуть оновлюватися онлайн, щоб відображати будь-які зміни, що відбулися в реальному виробничому середовищі. Це дозволить багатьом функціям постійно покращувати різні аспекти виробничих процесів навіть за наявності деяких змін. Новизна запропонованої структури полягає в тому, що пропонується використовувемо досягнення машинного навчання та інтелектуального аналізу процесів, щоб подолати недоліки попередніх підходів до автоматичної/напівавтоматичної розробки моделей моделювання та створювати моделі на основі даних, які постійно перевіряються та оновлюються.

Ключові елементи, що забезпечують створення цифрового двійника на основі даних для виробництва, наступні:

1. Збір даних. Середовище розумної фабрики пов'язане з великими обсягами даних, отриманих з багатьох джерел. Першим завданням, яке необхідно виконати перед початком збору даних, є ідентифікація сутностей. Цей крок включає пошук та збір даних про сутності, що стосуються мети цифрового двійника в реальній системі, що моделюється. Після завершення цього завдання дані, які

постійно генеруються ідентифікованими сутностями у виробничих системах фабрики, зберігаються в базі даних.

Ключовими джерелами даних є наступні:

- Датчики: до виробничих компонентів прикріплені різні види датчиків, які записують дані, цінні для виконання функцій моніторингу справності та виявлення несправностей.
- Вхідні дані кінцевого користувача: зацікавлені сторони виробничих систем, такі як клієнти, часто надають дані, що відображають їхні вподобання, обмеження та потреби.
- Внутрішня мережа: взаємозв'язки між пристроями та частинами заводу дозволяють збирати вичерпні дані про завод в цілому.

Це стало можливим завдяки розвитку хмарних і туманних обчислень, а також Інтернету речей.

Різні типи даних, що генеруються, включають наступні:

- Діагностичні дані: такі дані корисні для моніторингу стану виробничих систем. Вони часто збираються за допомогою датчиків, прикріплених до машин.
- Журнали та сліди: історія роботи компонентів або машин в цілому записується у вигляді журналів, що може виявитися безцінним для розуміння процесів, що існують у заводському середовищі.
- Дані ринку/кінцевих користувачів: зворотний зв'язок, обмеження та потреби кінцевих користувачів часто використовуються для коригування роботи машин для досягнення заздалегідь визначених цілей.

Ще однією важливою частиною процесу збору даних є інтеграція даних. Це стосується процесу перетворення мультимодальних даних у вигляді зображень, журналів, аудіо тощо у зручний для використання та уніфікований формат.

2. Перевірка даних. Оскільки дані є основною сутністю, яка визначає правильність моделі, вкрай важливо, щоб дані спочатку пройшли ретельну перевірку. Це може бути у формі перевірок достовірності та очищення. Деякі з проблем, пов'язаних з даними, включають наступне.

-Шум: дані можуть бути спотворені помилками, внесеними вимірювальним пристроєм, навколишнім середовищем та людськими помилками.

- Відсутні значення: через різні причини, такі як несправності пристрою, проблеми з мережею та інші фактори, дані можуть мати відсутні значення.

- Різні способи обробки: різні типи даних, такі як зображення, текст та аудіо, фіксуються датчиками, присутніми в рамках інтелектуальної фабрики.

- Великий розмір: часто в заводському середовищі генеруються величезні обсяги даних.

Потенційно висока розмірність, а також розмір, можуть ускладнити генерацію моделі та зробити її неточною. Багато алгоритмів машинного навчання погано справляються з даними величезної розмірності.

Достовірність даних має першорядне значення, і тому попередня обробка даних з цією метою є важливим завданням. Точність та якість витягнутих моделей може серйозно постраждати, якщо базові дані містять помилки, які не були виявлені. Деякі із завдань попередньої обробки включають імпутацію та очищення. Імпутація стосується завдання заповнення відсутніх значень за допомогою статистичних або машинного навчання підходів. Очищення даних стосується всіх інших завдань, які перетворюють дані у формат, який може бути використаний на етапі вилучення знань.

Для вирішення проблеми перевірки даних ми пропонуємо використовувати схему виявлення аномалій, в якій існує поняття «норми» та позначає все, що відхиляється від цієї норми, як аномальне. На сьогодні було запропоновано кілька методів виявлення аномалій з різними підходами до розрізнення нормальних та аномальних даних. Було запропоновано кілька підходів на основі класифікації, які використовують класифікатор, такий як метод опорних векторів (SVM), для визначення межі прийняття рішень між нормальними та аномальними даними. Проблема з використанням контрольованого виявлення аномалій на основі класифікатора полягає в тому, що дисбаланс кількості екземплярів у наборі даних зазвичай суттєво впливає на продуктивність виявлення. Неконтрольовані методи також успішно використовуються для виявлення аномалій. Серед цих методів є

методи кластеризації та методи на основі підпростору. Хоча більшість методів у сфері виявлення аномалій були перевірені на стандартних еталонних наборах даних, їхня продуктивність у реальних даних, як правило, залежить від застосування. У літературі обговорюється, що серед цих різних підходів немає чіткого переможця. Таким чином, у нашій системі ми будемо використовувати комбінацію цих підходів для забезпечення достовірності даних за допомогою виявлення аномалій. Конкретно, залежно від типу даних, які ми отримуємо від датчиків у межах розумної фабрики, буде обрано найбільш підходящий метод виявлення аномалій.

3. Видобування знань. Завдання вилучення знань з перевірених даних, отриманих на попередньому кроці. Це завдання передбачає такі процеси:

- Виявлення та маркування подій Існують певні релевантні події, які є вхідними даними для симуляційної моделі, і тому ідентифікація цих подій, релевантних для цілей симуляції, що відбуваються на розумній фабриці, є важливою частиною симуляційного моделювання на основі даних. Події є ключовими елементами, що стимулюють розробку дискретно-подійних симуляційних моделей, а здатність автоматично виявляти події, що відбуваються на розумній фабриці, визначає ефективність симуляційного моделювання. ML пропонує різні методи, такі як класифікація та кластеризація, підходи для виявлення подій з даних інтелектуальної фабрики. Виявлення особливих подій, таких як несправності, вважається важливою частиною аналізу надійності виробничих систем. На швидкість реагування та точність моделі інтелектуального заводу впливає здатність виявляти виникнення несправностей. З точки зору машинного навчання, це завдання виявлення несправностей можна класифікувати як завдання виявлення аномалій, для якого в літературі існує ряд алгоритмів.

- Виявлення процесу. Окрім подій, процеси, що відбуваються в межах розумного заводу, є ще одним джерелом знань для розуміння стану заводу, допомагаючи зрозуміти такі параметри, як швидкість, тривалість, надійність тощо. Виявлення таких процесів було предметом дослідження протягом останніх кількох років. Одним з ключових джерел даних для виявлення процесів є журнали, або,

точніше, журнали подій. Щоб виявити процеси з журналів подій, формальний опис потоку процесу в системі виводиться за допомогою певної мови моделювання або специфікації процесу.

4. Розробка моделі. Розробка єдиної моделі на основі даних є наступним ключовим компонентом запропонованої схеми. Виявлені події та процеси слугують вхідними даними для цього кроку. Модель моделювання розробляється з використанням цих вхідних даних, спочатку з певним рівнем втручання людини. Ця початкова модель потім буде автоматично оновлюватися, щоб відобразити зміни в розумній фабриці. Для полегшення цього оновлення необхідно чітко визначити зв'язки між моделлю та потоками даних розумної фабрики. Крім того, необхідно буде розробити алгоритми для вилучення та оновлення моделі для підтримки процесів автоматичного/ напівавтоматичного моделювання моделювання.

5. Валідація моделі. Ретельна валідація моделі є невід'ємною частиною підходу до моделювання на основі даних, метою якого є побудова високоточних моделей, що допомагають у правильному та своєчасному прийнятті рішень. Оскільки дані безперервно збираються в режимі реального часу, є очевидна можливість для постійної валідації моделей моделювання. Ця наявність даних надає можливість для розробки нових підходів до валідації моделей, які підходять до процесу валідації поступово, дозволяючи валідувати частини моделей моделювання, щойно про них є «достатньо» даних. Повна залежність від даних для валідації моделі вимагає, щоб самі дані були ретельно валідовані, наприклад, щоб гарантувати, що моделі отримані з високоякісних даних. Цю проблему необхідно буде вирішити для підтримки запропонованої основи для моделювання моделювання на основі даних.

Мета валідації моделі полягає в тому, щоб забезпечити достатню ефективність моделі на невидимих даних, а також досягти хорошої продуктивності на вже бачених даних.

У описаній в цьому розділі системі використовується метод бутстрепа поза вибіркою, який працює наступним чином. Вибірка екземплярів даних вибирається випадковим чином із заміною з вихідного набору даних. Навчання моделі

виконується на цій вибірці, а тестування проводиться на вихідному наборі даних. Цей процес повторюється. Кожні t секунд, і повідомляється середнє значення показників продуктивності, отриманих у кожному процесі навчання-тестування. Процедура валідації моделі буде виконуватися через заздалегідь визначені проміжки часу, щоб гарантувати, що модель відповідає даним, які були згенеровані на даний момент. У нашій системі питання застарівання моделей, коли надходять нові дані внаслідок змін у системі, вирішується циклом, що включає валідацію даних, вилучення знань, розробку моделі та валідацію. Точніше, використання даних, отриманих у певний момент часу, витягуються знання, за допомогою яких розробляється модель. Ця модель проходить етап перевірки, на якому перевіряється правильність моделі з точки зору отриманих даних та цільового показника моделі. Якщо модель відповідає цільовому показнику та даним, то параметри моделі зберігаються. В інший час, $t+k$, після чого система змінюється та генерує нові дані, та сама процедура, що й вище, повторюється, щоб забезпечити відповідність поточної моделі поточним даним.

Пропонується фреймворк для цифрових двійників на основі даних для розумних фабрик, як показано на рисунку 3.1. Розумна фабрика, яка є реальним об'єктом, що моделюється, безперервно виробляє дані за допомогою своїх пристроїв та датчиків Інтернету речей. Ці дані є відправною точкою підходу моделювання на основі даних. Процеси збору даних включати ідентифікацію відповідної сутності та зберігання даних у базах даних. Ідентифікація сутності включає чітке перерахування відповідних сутностей, таких як виробничі системи, технології управління, а також інші компоненти в рамках розумної фабрики. Одним із відповідних підходів до збору та інтерпретації виробничих даних є стандарт *Haystack*. Цей стандарт є ініціативою з відкритим кодом, яка пропонує стандартні семантичні моделі даних з метою спрощення вилучення значення з даних, що генеруються різними пристроями Інтернету речей, включаючи пристрої на виробничих об'єктах. Наступним логічним кроком є перевірка даних за допомогою загального очищення даних або попередньої обробки та інтеграції. Дані неявно фіксують відповідні події, що відбуваються на заводі. Щоб зробити ці події більш

чіткими та допомогти на етапі розробки моделі, маркування подій виконується напівавтоматично, тобто шляхом втручання людини.

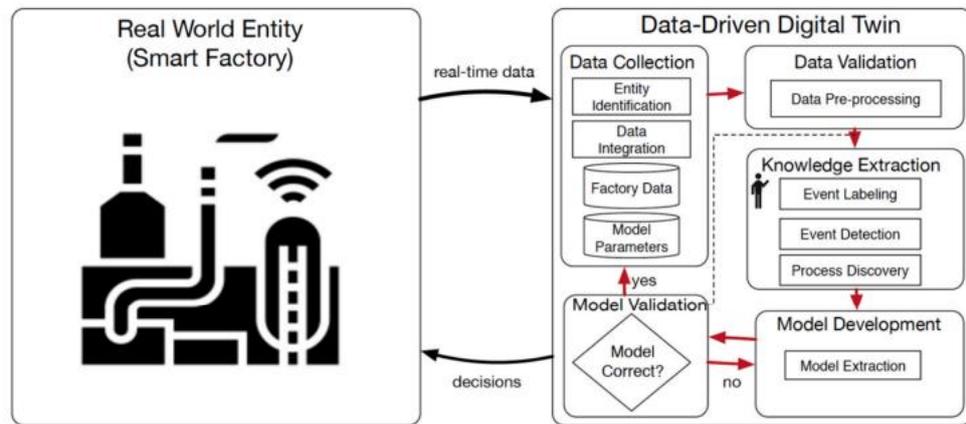


Рис.3.1. Узагальнена схема для інтелектуальної фабрики на основі даних

Процеси виявлення та маркування подій можуть підтримуватися додатковим процесом самостійного навчання (наприклад, кластеризації) подій. У цьому додатковому процесі всі виявлені події характеризуються та кластеризуються на основі їхньої подібності. Зрештою, експерт спостерігатиме за згенерованими кластерами та їхніми характеристиками, і за необхідності втручатиметься, а також надаватиме мітки для кластерів. В результаті цих процесів виявлення та маркування подій ідентифікується та маркується ряд відповідних подій. Моделі машинного навчання можуть використовуватися для безперервного та автоматичного виявлення подій з використанням попередньо позначених даних. В результаті створюються журнали подій, які далі використовуються для виявлення процесів за допомогою алгоритмів аналізу процесів.

Як процеси, так і події, які були ідентифіковані, потім використовуються для побудови імітаційної моделі для даної розумної фабрики. Фундаментальним етапом етапу розробки моделі є безперервна перевірка моделі, яка відбувається поступово, перевіряючи модель у міру її вилучення. Після перевірки вилученої моделі відповідні параметри моделі зберігаються для подальшого використання. Цю модель потім можна використовувати для виконання різних сценаріїв «що, якщо» в

рамках симуляційного дослідження, які потім оцінюються з урахуванням заздалегідь визначених показників ефективності. Результати цих експериментів можуть допомогти зацікавленим сторонам у прийнятті обґрунтованих рішень щодо оптимальної роботи певної розумної фабрики.

Для дослідження оберемо виробничу лінію, яка складається з 5 активів, а саме:

- 1) Склад з автоматизованим комплектуванням та прийманням замовлень,
- 2) Автономний мобільний робот з маніпулятором, який може завантажувати, транспортувати та розвантажувати товари,
- 3) Високошвидкісний складальний шлях з магнітно прикріпленим транспортом пристрої,
- 4) Дві складальні комірки, оснащені роботами-маніпуляторами, здатними виконувати певні завдання,
- 5) Людино-машинний інтерфейс (НМІ) для керування та моніторингу виробничого процесу.

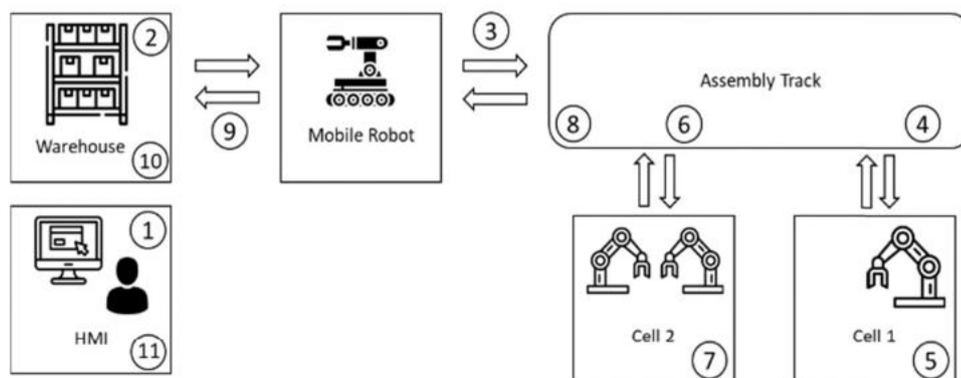


Рис.3.2. Ілюстрація виробничого процесу

Окремі етапи виробництва:

- 1) Користувач запускає виробничий процес, надсилаючи замовлення з НМІ.
- 2) Склад готує деталі для замовлення.
- 3) Мобільний робот транспортує деталі до складальної колії.
- 4) Колія транспортує деталі до першої складальної комірки.

5) Комірка один виконує перший крок складання, тобто з'єднує двигуни з шасі.

6) Коля транспортує деталі до другої складальної комірки.

7) Комірка два виконує другий крок складання, тобто кріпить ротори до двигунів.

8) Коля транспортує зібраний виріб до кінця колії.

9) Мобільний робот транспортує виріб на склад.

10) Склад розміщує виріб на складі.

11) Користувача інформують про завершення процесу.

Щоб продемонструвати модель, представлено мережу Петрі (Рис. 3.3). Мережа Петрі зображує суворо послідовний характер процесу. Оскільки в поточній конфігурації виробничої лінії немає надлишковості чи буфера, вихід з ладу одного з активів призводить до зупинки всього виробництва. Змоделивали можливі збої під час виконання активу як дуги інгібітора, що запобігають спрацюванню певних переходів. Після збою активу спрацьовує перехід «Збій» активу та створює токен у певному місці «Збій», який активує дуги інгібітора. Таким чином, жоден перехід, що залежить від активу, що вийшов з ладу, не може спрацювати. Щойно актив буде відремонтовано, токен у місці збою активу знищується, а дуги інгібітора деактивуються.

Настпні поширені несправності можуть виникати на цій виробничій лінії квадрокоптерів, що можуть спричинити збій системи, і перераховані нижче:

- Мобільний робот не розпізнає коробку із сировиною, наданою складом.
- Мобільний робот скидає сировину поруч із транспортними пристроями зі складальної колії.
- Через високу швидкість складальної колії деталі час від часу випадають з транспортних пристроїв.
- Деталь нахилена або повернута на опорних плитах - колаборативні роботизовані маніпулятори можуть не мати змоги підняти деталь.

- У комірці 2 невелика пружина кріпиться до двигуна до кріплення ротора – майже в половині випадків пружину неможливо підняти; ротор все ще кріпиться до двигуна.

У майбутньому до виробничої лінії будуть додані додаткові етапи виробництва та складальні комірки. Наприклад, це можуть бути комірки, що дозволяють роботам і людям працювати разом, а також комірки, що базуються виключно на взаємодії з людиною. Крім того, можна додати буфери та резервування, щоб зробити виробництво більш схильним до помилок. Регулярне технічне обслуговування активів також сприятиме загальній стабільності системи. Запропонована структура, разом із запропонованою методологією, дозволяє витягувати моделі, такі як модель мереж Петрі для надійності Рис.3.3 з даних, зібраних за допомогою датчиків, з обмеженою підтримкою експертних знань.

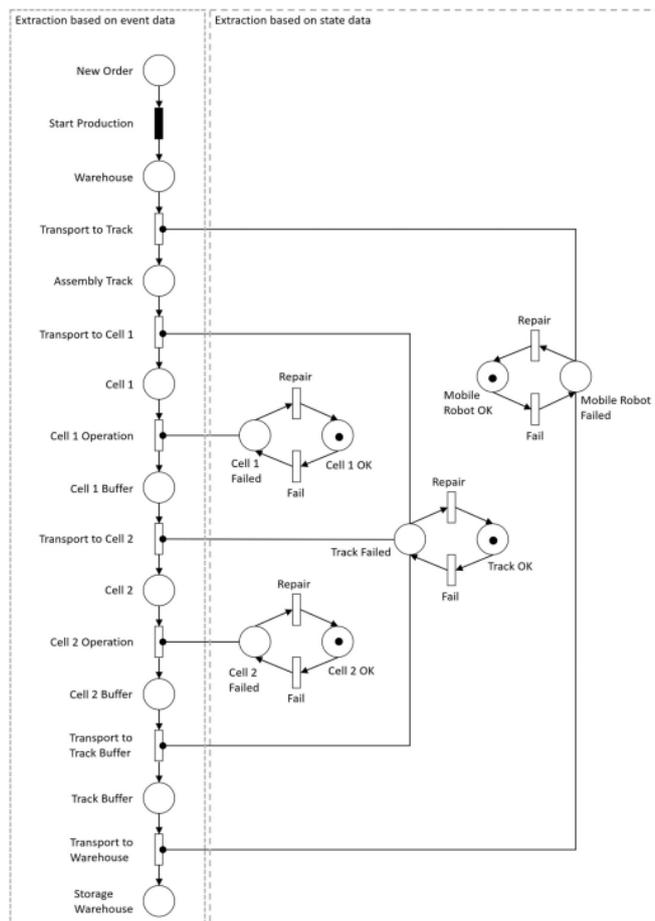


Рис.3.3. Орієнтована на надійність мережа Петрі виробничого заводу в її поточному стані.

На основі даних подій, модель процесу, що описує виробництво, може бути вилучена за допомогою методів виявлення процесів, таких як Alpha Miner. Дані про стан можна використовувати для вилучення та моделювання змін експлуатаційного стану активів. Обидва аспекти моделювання вказані в мережі Петрі. Після вилучення мережі Петрі, орієнтованої на надійність, її якість слід оцінити за допомогою відповідних критеріїв. Це можна зробити або на основі журналів даних, з яких було вилучено мережу Петрі (перевірка відповідності), або на основі базової мережі Петрі. Якщо модель процесу визначення реальних даних відома (як у нашому випадку), ми можемо перевірити та оцінити якість отриманих моделей, використовуючи точність поширених вимірювань P , згадати R та їхнє гармонійне середнє F_1 . Точність являє собою відношення кількості правильно призначених ребер до загальної кількості призначених ребер. Нагадаємо, що — це відношення кількості правильно призначених ребер до кількості всіх ребер, присутніх у вихідній моделі. Ми можемо математично сформулювати ці показники наступним чином:

$$P = \frac{TP}{TP+FP} \quad (3.1)$$

$$R = \frac{TP}{TP+FN} \quad (3.2)$$

$$F_1 = 2 * \frac{P*R}{P+R} \quad (3.3)$$

де TP – множина ребер, що існують як у витягнутій моделі, так і у вихідній моделі; FP – множина ребер, що існують у витягнутій моделі, але відсутні у вихідній моделі; та FN – множина ребер, що відсутні у витягнутій моделі, але існують у вихідній моделі.

Щоб використовувати отриману мережу Петрі для аналізу надійності, необхідно знати розподіли ймовірностей часу відмов та ремонту для кожного виробничого активу.

Дані про стан надають цю інформацію, обчислюючи час до кожної відмови та час для кожного ремонту на основі відповідних змін робочого стану. Для оцінки теоретичного розподілу, якому відповідають емпірично зібрані дані, використовуються такі методи, як оцінка максимальної правдоподібності (MLE)

можна використовувати. MLE оцінює параметри розподілу ймовірностей шляхом максимізації функції правдоподібності таким чином, щоб передбачуваний теоретичний розподіл найкраще описував спостережувані дані. Після оцінки правильного розподілу та відповідних параметрів розподілу можна побудувати графік функції надійності. Функція надійності (часто звана функцією виживання) є додатковою кумулятивною функцією розподілу оціненого розподілу та відображає ймовірність того, що виробничий актив проіснує протягом певного часу.

Окрім мереж Петрі, для оцінки надійності системи зазвичай використовуються дерева відмов. У виробничому сценарії для створення дерев відмов спочатку необхідно ідентифікувати події, що призводять до відмови системи. Якщо дані про події недостатньо деталізовані для охоплення подій, пов'язаних з несправностями, це можна зробити за допомогою методів виявлення та класифікації несправностей, на основі даних моніторингу стану виробничого підприємства.

Представлена структура для цифрових двійників на основі даних у розумних фабриках має низку пов'язаних з цим проблем, які перелічені нижче:

- Визначення та повне визначення відповідного ступеня втручання людини для сприяння застосуванню підходу, керованого даними: ця проблема стосується використання переважно підходу, керованого даними, який потребує та залежить від втручання людини для забезпечення автоматизації. Чітко визначені точки втручання людини забезпечать належне налаштування моделювання на основі даних, а також його інтерпретованість та ефективність. Інтерпретованість стосується аспекту підходу, керованого даними/інтелектуального підходу, який пропонує певний принциповий спосіб розуміння рішень, прийнятих цифровим двійником, з точки зору зацікавленої сторони. Ефективність стосується аспекту цифрового двійника, керованого даними, який включає точність та своєчасність.
- Прийняття рішень у режимі реального часу: цифрові двійники можуть допомогти зацікавленим сторонам приймати вигідні рішення з точки зору різних показників ефективності. Така підтримка рішень стала можливою завдяки тому, що цифрові двійники на основі даних витягують відповідні знання з усіх зібраних

даних, що може виявитися безцінним для зацікавлених сторін. Що стосується своєчасності та критичності рішень, потреби щодо своєчасності повинні відповідати критичності та вазі пов'язаних з ними рішень. Це має відобразитися в деталізації даних, що збираються. Рішення, пов'язані з безпекою, особливо в критично важливих для безпеки системах, вимагатимуть вищої своєчасності, ніж рішення, пов'язані з виробництвом.

- Еволюційні цілі: інтелектуальні виробничі системи спрямовані на забезпечення стабільних та економічно ефективних частот виробництва. Ця мета тісно пов'язана з іншими характеристиками, такими як надійність виробничі системи, зменшення матеріальних відходів та споживання енергії тощо. Окрім цього, екологічні цілі, такі як зменшення викидів парникових газів, є ще однією важливою метою, яку слід враховувати та враховувати. Створення імітаційних моделей для відображення, оновлення та обслуговування таких різноманітних та еволюціонуючих цілей є дуже складним і величезним викликом, який необхідно вирішити.

Однак представлена структура також відкриває нові можливості:

- Розробка високоточних цифрових двійників: завдяки тому, що запропонована структура базується на даних, моделі, як правило, є точним відображенням відповідних виробничих систем, що відображають актуальну поведінку процесів на «розумних» фабриках. Крім того, постійна перевірка як даних, так і моделей забезпечує постійну підтримку цієї точності.

- Покращене розуміння процесів та прийняття рішень: виявлення процесів є важливою частиною запропонованого нами підходу. Це досягається за допомогою аналізу зібраних журналів подій, на додаток до інших компонентів, керованих даними. Це допомагає покращити розуміння потоків процесів у системі та буде корисним для покращення прийняття рішень. Використовуючи потоки процесів у системі, зацікавлені сторони можуть приймати відповідні рішення щодо майбутнього системи, а також її цифрового двійника.

Підвищена надійність: надійність запропонованої структури пояснюється постійною перевіркою даних і моделі. Акцент на постійному перевірці, а не на

перевірці через фіксовані проміжки часу, гарантує, що система розвивається відповідно до швидкості надходження нових даних. Система перевірки гарантує, що дані, які є важливою частиною нашої структури, є правильними, а також що модель точно відображає дані, які вона моделює.

- Гнучке виробництво: цифровий двійник на основі даних може запропонувати надійну платформу для впровадження концепції гнучкого виробництва, яка дозволяє виробничим підприємствам швидко реагувати на зміни ринку, зберігаючи при цьому хорошу якість та контролюючи витрати.

Підсумовуючи, впровадження цифрових двійників пропонує кілька можливостей у рамках розумної фабрики, але водночас створює різні проблеми. Ці можливості можна розглядати як мотивацію для впровадження цифрових двійників на основі даних у виробничих системах.

3.2 Метод впровадження цифрових двійників в промислові процеси

Попри значний поступ у створенні та моделюванні цифрових двійників у сфері виробництва, практичні кроки щодо їхнього реального впровадження все ще недостатньо окреслені. Чимало існуючих моделей і фреймворків перевіряють лише в лабораторних умовах або у штучно створених сценаріях, не надаючи конкретних рекомендацій для застосування в реальних виробничих процесах. У цьому розділі запропоновано структурований підхід до інтеграції цифрового двійника у виробниче середовище. Методика поділена на три складові, що відповідають ключовим рівням архітектури цифрового двійника: фізичному, віртуальному та рівню взаємозв'язку. Кожен модуль забезпечує необхідні технічні механізми для реалізації свого рівня. Така структура дозволяє систематизувати принципи проектування та впорядкувати їх у послідовний і зрозумілий алгоритм дій.

Модуль I – Фізичний рівень конфігурації.

Перший модуль формує фундаментальні параметри впровадження. На цьому етапі визначають потік створення цінності, а також обирають конкретний сценарій застосування, релевантний управлінню цим потоком. У межах двох початкових

кроків уточнюються реальні фізичні межі процесу, після чого встановлюють вимоги до даних, необхідні для обраного варіанта використання.

На основі цих вимог формується перелік точок збору даних для кожної операції або етапу, що забезпечує безперервний і повний моніторинг процесу. Для впорядкування інформації створюється матриця вимог до даних, яка допомагає чітко визначити й призначити всі необхідні точки даних. Після цього здійснюється перевірка — які з потрібних точок уже існують у чинних ІТ-системах, а які необхідно додати.

Фінальним результатом стає повна матриця, що відображає всі наявні та відсутні точки збору даних. Далі виконують впровадження бракуючих точок, щоб наповнити базу даних та забезпечити цілісне уявлення про потік створення цінності. Цей етап є ключовим, оскільки саме він дозволяє організації приймати обґрунтовані рішення та підтримувати безперервне вдосконалення процесів.

Після завершення інтеграції всіх точок збору даних фізичний рівень вважається реалізованим, і стає можливим перехід до наступного - віртуального рівня.

Модуль II – Віртуальний рівень конфігурації.

Налаштування віртуального рівня стартує з виконання двох взаємопов'язаних напрямів роботи: створення інтерфейсу між фізичною та цифровою частинами системи та організації сховища даних. Під час розроблення інтерфейсу користувачам надаються рекомендації щодо вибору засобів комунікації й аналізу доступних протоколів передачі даних.

Формування бази історичних даних передбачає визначення рівня деталізації, з якою потрібно фіксувати інформацію. У цьому випадку оптимальним вважається зберігання даних для кожного елемента та кожного кроку процесу, що забезпечує точне відтворення подій і побудову достовірної цифрової моделі.

Після того як усі необхідні точки даних були зібрані, формується концептуальна модель даних. На цьому етапі визначають елементи потоку створення цінності, їх характеристики та зв'язки між ними. Далі ці складові

впорядковують і перетворюють у логічну модель даних, уточнюючи типи атрибутів і встановлюючи первинні та зовнішні ключі.

Наступним кроком є вибір системи керування базою даних — рішення, яке суттєво впливає на подальшу реалізацію. Щоб полегшити цей етап, проводиться огляд існуючих типів баз даних, які можуть підтримати моделювання потоку створення цінності. Після визначення оптимальної СУБД запускають два паралельні процеси: налаштування процедур очищення та попередньої обробки даних, а також побудову фізичної моделі даних. Обидві дії є критичними для стабільної роботи системи, тому їх виконують синхронно, щоб уникнути дублювання та помилок. Завершальним кроком є перевірка готовності фізичної моделі до експлуатації. Із цього моменту можна вважати, що віртуальний рівень повністю сформований.

Модуль III – Рівень з'єднання конфігурації

Фінальний модуль спрямований на інтеграцію віртуальної та фізичної складових. Спочатку проводять детальний аналіз наявних даних, досліджуючи як статичні, так і динамічні моделі, щоб отримати комплексне уявлення про роботу потоку створення цінності. Це розуміння є основою для формування вимог до програмного забезпечення, що визначає ключові функціональні можливості майбутньої системи. Подальше ранжування цих вимог забезпечує правильне розставлення пріоритетів і врахування всіх важливих аспектів у процесі вибору рішення.

Опісля здійснюється пошук та оцінювання можливих програмних продуктів. Цей етап передбачає ґрунтовне вивчення ринку, щоб знайти найефективніше та найвідповідніше рішення. Аналіз доступних інструментів допомагає сформуванню перелік кандидатів, серед яких обирають найкращий варіант.

На завершення обране програмне забезпечення інтегрується з фізичною моделлю даних, яка є основою подальшої аналітики та обробки інформації. Результатом цього етапу стає можливість повноцінного використання цифрового двійника та отримання цінності від його застосування.

ВИСНОВКИ

У ході виконання дослідження було досягнуто поставленої мети – реалізації платформи цифрового двійника (ЦД), здатної забезпечити гнучке керування, моніторинг та дистанційну взаємодію з фізичними об'єктами в межах кіберфізичних виробничих систем. Проведена робота дозволила сформулювати цілісне уявлення про концепцію цифрового двійника, її практичне застосування та технологічні можливості для підвищення ефективності сучасного виробництва.

У процесі дослідження визначено сутність та ключові характеристики концепції цифрового двійника. Проаналізовано наукові підходи до її трактування та встановлено, що універсального визначення поняття ЦД поки що не існує, що зумовлює необхідність формування адаптивної архітектури, здатної враховувати специфіку конкретних галузей. Визначено основні напрями застосування цифрових двійників, серед яких моніторинг, прогнозування, оптимізація та дистанційне керування технологічними процесами. Здійснений аналіз показав, що основними обмеженнями впровадження ЦД у промислових системах залишаються складність інтеграції, потреба в стандартизації даних та висока вартість апаратних і програмних засобів.

В межах роботи досліджено розробку архітектури платформи цифрового двійника, що забезпечує гнучку інкапсуляцію функцій управління, візуалізації та моніторингу. Досліджене рішення дозволяє об'єднати в єдиній структурі компоненти, відповідальні за відображення фізичного стану об'єкта, аналіз його поведінки та реалізацію керуючих впливів. Цифрові двійники на основі даних є необхідністю для успішного впровадження Індустрії 4.0. Ручне моделювання моделювання не є варіантом для сучасних виробничих систем, які зазнають численних та швидких реконфігурацій протягом свого життєвого циклу. З цієї причини ми дослідили вимоги, які дозволили б розробку імітаційних моделей на основі даних для інтелектуальних виробничих систем як основу для їхніх цифрових двійників. Цифрові двійники на основі даних у реальному часі, автоматично

розроблені на основі даних, зібраних з пристроїв Інтернету речей інтелектуальних фабрик, можуть задовольнити значні потреби, наприклад, завжди актуальні імітаційні моделі, що точно відображають поточні зміни в схемі фабрик. Крім того, цифрові двійники на основі даних дозволять проводити інтегровану постійну перевірку моделі, яку можна буде розпочати, як тільки частини моделі будуть вилучені. Наявність даних з реальної системи зробить це можливим.

ПЕРЕЛІК ПОСИЛАНЬ

1. Michael Grieves and John Vickers. Digital Twin: Mitigating Unpredictable, Undesirable Emergent Behavior in Complex Systems, pages 85–113. Springer International Publishing, 2022.
2. E. H. Glaessgen and D. S. Stargel. The digital twin paradigm for future NASA and U.S. Air force vehicles. Collection of Technical Papers - AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference, pages 1–14, 2022.
3. Yifei Tan, Wenhe Yang, Kohtaroh Yoshida, and Soemon Takakuwa. Application of IoT-aided simulation to manufacturing systems in cyber physical system. *Machines*, 7(1), mar 2019.
4. A. Fuller, Z. Fan, C. Day, and C. Barlow. Digital Twin: Enabling Technologies, Challenges and Open Research. *IEEE Access*, 8:108952–108971, 2020.
5. A. Rasheed, O. San, and T. Kvamsdal. Digital twin: Values, challenges and enablers from a modeling perspective. *IEEE Access*, 8:21980–22012, 2020.
6. Siavash H. Khajavi, Naser Hossein Motlagh, Alireza Jaribion, Liss C. Werner, and Jan Holmstrom. Digital Twin: Vision, benefits, boundaries, and creation for buildings. *IEEE Access*, 7:147406–147419, 2019.
7. W. Kritzinger, M. Karner, G. Traar, J. Henjes, and W. Sihn. Digital Twin in manufacturing: A categorical literature review and classification. *IFAC PapersOnLine*, 51(11):1016–1022, 2018.
8. Juuso Autiosalo, Jari Vepsalainen, Raine Viitala, and Kari Tammi. A Feature-Based Framework for Structuring Industrial Digital Twins. *IEEE Access*, 8:1193–1208, 2020.
9. Qinglin Qi, Fei Tao, Tianliang Hu, Nabil Anwer, Ang Liu, Yongli Wei, Lihui Wang, and A. Y.C. Nee. Enabling technologies and tools for digital twin. *Journal of Manufacturing Systems*, 2022.

10. Guolin Lyu and Robert William Brennan. Towards IEC 61499-Based Distributed Intelligent Automation: A Literature Review. *IEEE Transactions on Industrial Informatics*, pp.22–26, 2021.
11. Luis Neto, Joao Reis, Ricardo Silva, and Gil Goncalves. Sensor SelComp, a smart component for the industrial sensor cloud of the future. *Proceedings of the IEEE International Conference on Industrial Technology*, pp. 56–61, 2023.
12. Damien Arrachequesne. The socket instance (client-side), Jun 2021. Accessed: 2021-06-15. Дата звернення: 09.11.2025. URL: <https://socket.io/docs/v4/client-socket-instance/>.
13. Liliana Antão, Rui Pinto, João Reis, and Gil Gonçalves. Requirements for testing and validating the industrial internet of things. *Proceedings - 2020 IEEE 11th International Conference on Software Testing, Verification and Validation Workshops, ICSTW 2020*, pp. 110–115, 2020.
14. K. P. Jan Stentoft Kent Adboll Wickstrøm and A. Haug, “Drivers and barriers for Industry 4.0 readiness and practice: Empirical evidence from small and medium-sized manufacturers,” *Prod. Plan. Control*, vol. 32, no. 10, pp. 811– 828, 2021,
15. V. Vijayaraghavan and J. Rian Leevinson, “Internet of Things applications and use cases in the era of Industry 4.0,” In *The Internet of Things in the Industrial Sector: Security and Device Connectivity, Smart Environments, and Industry 4.0*, Z. Mahmood, Ed., Cham: Springer International Publishing, 2019, pp. 279–298,
16. Onaji, I., Tiwari, D., Soulatiantork, P. et al. (2 more authors) Digital twin in manufacturing : conceptual framework and case studies. *International Journal of Computer Integrated Manufacturing*, 35 (8). pp. 831-858, 2022.
17. Roberto Saracco «Digital Twins: Evolution in Manufacturing» *IEEE Digital Reality* 23 May 2022.
18. C. Kober, B. N. Algan, M. Fette, and J. P. Wulfsberg, “Relations of Digital Twin Fidelity and Benefits: A Design-to-Value Approach,” *Procedia CIRP*, 2023.
19. C.Kobera*, M. Fettea , J.P. Wulfsberga «A Method for Calculating Optimum Digital Twin Fidelity» 56th CIRP Conference on Manufacturing Systems, CIRP CMS ‘23, South Africa.

20. F. Tao, H. Zhang, A. Liu, and A. Y. C. Nee, “Digital Twin in Industry: State-of-the-Art,” *IEEE Transactions on Industrial Informatics*, vol. 15, no. 4, pp. 2405–2415, 2019.

21. C. Kober, M. Benfer, F. Gomez Medina, J. P. Wulfsberg, V. Martinez, and G. Lanza, “Digital Twin Stakeholder Communication Model - Facilitating Communication Across Stakeholders,” *International Journal of Production Economics*, 2023

22. E. Oztemel and S. Gursev, “Literature review of Industry 4.0 and related technologies,” *Journal of Intelligent Manufacturing*, vol. 31, no. 1, pp. 127– 182, 2020.

23. SmartM2M; Digital Twins: Functionalities and communication Reference Architecture. ETSI TS 103 846 V1.1.1,2024.

ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ