

ДВНЗ «Донецький національний технічний університет»
Факультет комп'ютерно-інформаційних технологій та автоматизації
(повне найменування інституту, назва факультету)

Кафедра автоматики та телекомунікацій
(повна назва кафедри)

«До захисту допущено»
Завідувач кафедри автоматики та телекомунікацій
_____ Валерій ПОЦЕПАЄВ
(підпис) (ім'я та прізвище)
«__» _____ 2024 р.

Випускна кваліфікаційна робота

_____ магістра
(освітньо-кваліфікаційний рівень)

на тему «Розробка веб-додатка для керування віддаленими об'єктами з використанням технологій машинного навчання»

Виконав студент 2 курсу, групи СУАм-23
(шифр групи)

спеціальності 174 Автоматизація, комп'ютерно-інтегровані технології та робототехніка
(шифр і назва спеціальності)

_____ Канарський Юрій Володимирович
(Прізвище, Ім'я та По-батькові) (підпис)

Керівник зав. каф., к.т.н., доц. _____ Валерій ПОЦЕПАЄВ
(посада, науковий ступінь, вчене звання, прізвище та ім'я) (підпис)

Рецензент доц., к.т.н., доц. _____ Олександр ШТЕПА
(посада, науковий ступінь, вчене звання, прізвище та ім'я) (підпис)

Рецензент зав. каф., к.т.н., доц. _____ Олександр КОЛЛАРОВ
(посада, науковий ступінь, вчене звання, прізвище та ім'я) (підпис)

Нормоконтроль:

Дар'я ЖУКОВСЬКА

19.12.2024 р.

Засвідчую, що у цій випускній кваліфікаційній роботі немає запозичень з праць інших авторів без відповідних посилань.

Студент _____
(підпис)

Дрогобич – 2024 р.

ДВНЗ «Донецький національний технічний університет»
Факультет комп'ютерно-інформаційних технологій та автоматизації
(повне найменування інституту, назва факультету)

Кафедра автоматики та телекомунікацій
(повна назва кафедри)

Захист відбувся _____
(дата)

з оцінкою _____

Секретар ЕК _____
(підпис)

Випускна кваліфікаційна робота магістра

Тема: «Розробка веб-додатка для керування віддаленими об'єктами з використанням технологій машинного навчання»

Спецчастина: _____

Виконавець, студент

гр. СУАм-23

Юрій КАНАРСЬКИЙ

(підпис, дата, Ім'я ПРІЗВИЩЕ)

Керівник

Валерій ПОЦЕПАЄВ

(підпис, дата, Ім'я ПРІЗВИЩЕ)

Консультанти:

Вікторія ВОРОПАЄВА

(підпис, дата, Ім'я ПРІЗВИЩЕ)

Ганна ТЕЛИЧКО

(підпис, дата, Ім'я ПРІЗВИЩЕ)

Ярослав ЛЯШОК

(підпис, дата, Ім'я ПРІЗВИЩЕ)

Нормоконтроль

Дар'я ЖУКОВСЬКА

(підпис, дата, Ім'я ПРІЗВИЩЕ)

Дрогобич – 2024 р.

**Державний вищий навчальний заклад
"Донецький національний технічний університет"**

Інститут, факультет Комп'ютерно-інформаційних технологій та автоматизації

Кафедра Автоматика та телекомунікації

Освітньо-кваліфікаційний рівень магістр

Галузі знань 17 Електроніка, автоматизація та електронні комунікації

(шифр і назва)

Спеціальність 174 Автоматизація, комп'ютерно-інтегровані технології та
робототехніка

(шифр і назва)

ЗАТВЕРДЖУЮ
Завідувач кафедри АТ
Валерій ПОЦЕПАЄВ

« _____ » _____ 2024 року

З А В Д А Н Н Я
НА ВИПУСКНУ КВАЛІФІКАЦІЙНУ РОБОТУ МАГІСТРА

Канарський Юрій Володимирович

(прізвище, ім'я, по батькові)

1. Тема проекту (роботи) Розробка веб-додатка для керування віддаленими об'єктами з використанням технологій машинного навчання

керівник проекту (роботи) Поцєпасєв Валєрїй Валєрїйович

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу від 20.09.2024 року № 449

2. Строк подання студентом проекту (роботи) 12.12.2024 р.

3. Вихідні дані до проекту (роботи)

Технічна документація та матеріали з переддипломної практики

4.Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) вступ, аналіз та характеристика об'єкту, проектування веб-додатка для керування віддаленими об'єктами, моделювання системи та тестування системи, висновки

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)
Структурна схема, код веб-додатку

6. Консультанти розділів проекту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1	Воропаєва В.Я., проректор		
2	Теличко Г.О., доцент		
3	Поцепаєв В.В, зав. каф, доцент		
4	Ляшок Я.О., професор		
Нормо-контроль	Жуковська Д.О., асистент	-	19.12.2023 р.

7. Дата видачі завдання _____

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту (роботи)	Строк виконання етапів проекту (роботи)	При-мітка
1	Вивчення літератури за тематикою проекту	09.09.2024	
2	Розробка та узгодження технічного завдання	09.09.2024	
3	Підготовка матеріалів першого розділу дипломного проекту	17.10.2024	
4	Підготовка матеріалів другого розділу дипломного проекту	18.10.2024	
	Написання коду web-додатку	19.10.2024	
	Підготовка третього розділу дипломного проекту	21.10.2024	
	Підготовка четвертого розділу дипломного проекту	27.10.2024	
	Оформлення документації дипломного проекту	01.11.2024	
	Перевірка якості виконання магістерської кваліфікаційної роботи та усунення недоліків	12.12.2024	
	Підписи супроводжувальних документів у керівника, нормоконтролера		
	Перевірка «антиплагіат»		
	Попередній захист		

Студент

(підпис)

Юрій КАНАРСЬКИЙ

(прізвище та ініціали)

Керівник роботи

(підпис)

Валерій ПОЦЕПАЄВ

(прізвище та ініціали)

ЛИСТ ЗАУВАЖЕНЬ

Посада, Ім'я та ПРІЗВИЩЕ	Суть зауваження, оцінка та підпис

АНОТАЦІЯ

Канарський Ю.В. Розробка веб-додатка для керування віддаленими об'єктами з використанням технологій машинного навчання / Випускна кваліфікаційна робота на здобуття освітнього ступеня «магістр» зі спеціальності 174 Автоматизація, комп'ютерно-інтегровані технології та робототехніка. – ДВНЗ «ДонНТУ», Дрогобич, 2024.

Пояснювальна записка: 92 стор., 5 рис., 5 табл., 16 посилань.

У випускній кваліфікаційній роботі розроблено веб-додаток для керування віддаленими об'єктами із застосуванням технологій машинного навчання. Проведено аналіз існуючих систем дистанційного керування. Визначено вимоги до веб-додатка та сформульовано основні задачі розробки. Реалізовано інструменти для моніторингу та керування віддаленими об'єктами, а також інтегровано алгоритми машинного навчання для покращення ефективності управління. Виконано вибір технологій, розроблено архітектуру системи та протестовано її працездатність.

Ключові слова: ВЕБ-ДОДАТОК, ДИСТАНЦІЙНЕ КЕРУВАННЯ, МАШИННЕ НАВЧАННЯ, АЛГОРИТМ, ІНТЕРАКТИВНІСТЬ, МОНІТОРИНГ, ОБ'ЄКТИ.

ABSTRACT

Kanarskiy Y.V. Development of a web application for remote object management using machine learning technologies / Graduation qualifying work for obtaining an educational degree "Master" in specialty 174 Automation, Computer-Integrated Technologies, and Robotics - SHEE DonNTU Drohobych, 2024.

Explanatory note: 92 pages, 5 figures, 5 tables, 16 references.

The bachelor's thesis focuses on the development of a web application for managing remote objects using machine learning technologies. The existing remote management systems were analyzed. Requirements for the web application were defined, and the main development tasks were formulated. Tools for monitoring and managing remote objects were implemented, and machine learning algorithms were integrated to enhance management efficiency. The selection of technologies was performed, the system architecture was developed, and its functionality was tested.

Keywords: WEB APPLICATION, REMOTE MANAGEMENT, MACHINE LEARNING, ALGORITHM, INTERACTIVITY, MONITORING, OBJECTS.

ЗМІСТ

ВСТУП.....	10
1 АНАЛІЗ ТА ХАРАКТЕРИСТИКА ОБ'ЄКТУ.....	12
1.1 Аналіз існуючих систем для керування віддаленими об'єктами.....	12
1.2 Технологічні підходи та інструменти для створення веб-додатків.....	15
1.3 Огляд технологій машинного навчання для автоматизації управління.....	18
1.4 Аналіз сучасних рішень та систем-аналогів.....	20
Висновки до розділу 1.....	25
2 ПРОЕКТУВАННЯ ВЕБ-ДОДАТКА ДЛЯ КЕРУВАННЯ ВІДДАЛЕНИМИ ОБ'ЄКТАМИ.....	28
2.1 Архітектура системи.....	28
2.2 Постановка задачі: функціональні та нефункціональні вимоги.....	32
2.3 Розробка моделі даних та бази даних.....	35
2.4 Використання технологій машинного навчання для автоматизованого керування.....	38
2.5 Проектування інтерфейсу користувача.....	40
Висновки до розділу 2.....	46
3 ВИКОРИСТАННЯ ОБ'ЄКТУ МОДЕЛЮВАННЯ ДЛЯ ПРОЦЕСІВ ОХОРОНИ ПРАЦІ ПІД ЧАС НАДЗВИЧАЙНИХ СИТУАЦІЙ НА ПІДПРИЄМСТВІ.....	49
3.1 Роль об'єктів моделювання в оптимізації заходів охорони праці під час НС.....	49
3.2 Інтеграція об'єктів моделювання в системи оповіщення та евакуації під час надзвичайних ситуацій.....	51
3.3 Використання моделювання для прогнозування наслідків техногенних катастроф та розробки стратегій реагування.....	54
Висновки до розділу 3.....	59
4 МОДЕЛЮВАННЯ ТА ТЕСТУВАННЯ СИСТЕМИ.....	61

4.1 Реалізація та тестування веб-додатка.....	61
4.2 Оцінка ефективності та практичне впровадження.....	63
4.3. Важливість процесу моделювання та тестування перед повноцінним впровадженням у роботу веб-додатку.....	67
4.4. Веб-додаток для керування віддаленими об'єктами з використанням технологій машинного навчання	71
Висновки до розділу 4.....	74
ВИСНОВКИ.....	77
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	78
ДОДАТОК А Охорона праці.....	79
ДОДАТОК Б Приклад коду серверної частини.....	92
ДОДАТОК В Клієнтська частина: HTML, CSS та JavaScript	92

ВСТУП

Актуальність теми дослідження. Сучасний розвиток інформаційних технологій та машинного навчання дозволяє створювати інноваційні рішення для управління віддаленими об'єктами. Це особливо актуально для таких галузей, як промисловість – контроль обладнання, обслуговування підприємств; сільське господарство – моніторинг кліматичних умов, зрошення; транспортна логістика – керування транспортними засобами та вантажами; енергетика – контроль електростанцій та споживання енергії. В умовах глобалізації та цифровізації, необхідність у системах, які забезпечують моніторинг та управління віддаленими об'єктами в реальному часі, стає дедалі більшою. Впровадження веб-додатків для таких завдань забезпечує доступність, гнучкість і масштабованість систем. Водночас технології машинного навчання відкривають можливості для підвищення ефективності управління за рахунок автоматизації процесів, прогнозування та оптимізації ресурсів. Синергія веб-технологій та машинного навчання дозволяє створювати рішення, які не лише дозволяють контролювати віддалені об'єкти, але й роблять це більш інтелектуально та адаптивно. З огляду на це, розробка веб-додатка для керування віддаленими об'єктами із застосуванням технологій машинного навчання є актуальним науковим завданням, яке сприятиме підвищенню продуктивності систем, зменшенню витрат на їх обслуговування та поліпшенню управлінських процесів.

Метою роботи є розробка веб-додатка для керування віддаленими об'єктами з використанням технологій машинного навчання, що забезпечить ефективний моніторинг, контроль та автоматизацію управлінських рішень. Для досягнення мети роботи необхідно вирішити такі завдання:

1. Провести аналіз існуючих рішень для керування віддаленими об'єктами та технологій, які використовуються у цій сфері.
2. Визначити технологічний стек та інструменти для розробки веб-додатка.

3. Спроекувати архітектуру системи, базу даних і функціональний інтерфейс.

4. Розробити алгоритми машинного навчання для автоматизованого управління віддаленими об'єктами.

5. Реалізувати веб-додаток відповідно до поставлених вимог.

6. Провести тестування та оцінити ефективність роботи системи.

7. Сформулювати рекомендації щодо практичного впровадження додатка та визначити перспективи його розвитку.

Об'єкт і предмет дослідження. Об'єкт дослідження – процеси управління віддаленими об'єктами за допомогою веб-технологій. Предмет дослідження – методи та інструменти реалізації веб-додатка із застосуванням машинного навчання для автоматизації та оптимізації управлінських процесів.

Методи дослідження. У роботі використовуються наступні методи:

- системний аналіз для вивчення існуючих рішень та визначення вимог до системи;
- методи проектування програмного забезпечення для розробки архітектури веб-додатка;
- алгоритми машинного навчання (наприклад, регресійний аналіз, класифікація, нейронні мережі) для оптимізації та автоматизації процесів управління;
- експериментальні методи для тестування функціональності та продуктивності розробленої системи.

Наукова новизна і практична значимість. Наукова новизна роботи полягає у поєднанні веб-технологій та алгоритмів машинного навчання для керування віддаленими об'єктами, що дозволяє підвищити ефективність і точність процесів управління. Практична значимість полягає у можливості застосування результатів роботи для розробки реальних систем у промисловості, енергетиці, логістиці та інших сферах, де необхідно здійснювати контроль та управління об'єктами у віддаленому режимі.

Структура та обсяг кваліфікаційної роботи. Структура магістерської

кваліфікаційної роботи визначена її метою, завданнями та дослідженнями роботи. Робота складається з анотації на українській та англійській, вступу, чотирьох розділів, висновків з роботи, переліку посилань та додатків. Робота має 92 сторінок, 5 рисунків, 5 таблиць, 16 посилань.

1 АНАЛІЗ ІСНУЮЧИХ СИСТЕМ ДЛЯ КЕРУВАННЯ ВІДАЛЕНИМИ ОБ'ЄКТАМИ

1.1. Поняття та основні принципи віддаленого керування об'єктами

Віддалене керування об'єктами є одним із ключових процесів сучасного управління обладнанням, технологіями чи середовищами, особливо на значних відстанях, завдяки використанню новітніх інформаційних технологій. Воно базується на моніторингу, зборі даних та безпосередньому контролі об'єктів, забезпечуючи ефективність та автоматизацію багатьох процесів. Основним завданням віддаленого керування є забезпечення оперативності, точності й надійності управління, що стає можливим завдяки низці принципів.

Першим і основним принципом є доступ у реальному часі. Віддалені системи повинні забезпечувати збір, обробку та передачу даних без суттєвих затримок, що дозволяє контролювати стан об'єктів та здійснювати необхідні дії миттєво. Система моніторингу повинна оперативно реагувати на зміни параметрів об'єктів, передаючи дані до центрів обробки або користувачів. Це особливо важливо для систем, що працюють у критично важливих галузях, таких як медицина, промисловість чи транспорт.

Другим важливим принципом є надійність передачі даних. Віддалене керування неможливе без стабільної та безпечної передачі інформації між об'єктами і центральними системами управління. Для цього використовуються сучасні протоколи зв'язку, такі як HTTP, HTTPS, MQTT та WebSocket. Ці протоколи забезпечують безпечну та швидку передачу даних, що критично важливо для віддалених систем, особливо у випадках великого навантаження чи роботи в умовах нестабільного зв'язку.

Адаптивність є ще одним ключовим принципом віддаленого керування об'єктами. Сучасні системи повинні бути гнучкими та здатними підлаштовуватись під мінливі умови навколишнього середовища чи об'єктів. Це стосується як можливості інтегрувати нове обладнання, так і адаптації

алгоритмів обробки даних для оптимізації процесів управління. Наприклад, у сільському господарстві система має враховувати сезонні зміни чи непередбачувані погодні умови для автоматичного налаштування зрошувальних систем.

Інтеграція з сенсорами та пристроями IoT є основою сучасного віддаленого керування. Сенсори дозволяють зчитувати фізичні параметри об'єктів, такі як температура, тиск, вологість, освітленість тощо, передаючи ці дані до систем обробки. Віддалені системи керування часто працюють з великою кількістю пристроїв, об'єднаних у єдину мережу на базі технологій IoT. Це забезпечує можливість централізованого моніторингу та автоматизації управління.

Системи віддаленого керування складаються з кількох ключових компонентів. Сенсори та пристрої збору даних є базовими елементами, які забезпечують зчитування та первинну обробку інформації. Передані сенсорами дані надходять у системи передачі, які використовують технології LoRaWAN, ZigBee, GSM та різноманітні протоколи зв'язку для ефективного транспортування інформації на великі відстані. Завдяки цьому забезпечується безперебійне з'єднання навіть у важкодоступних місцях.

На центральному рівні функціонують системи управління. Вони складаються з програмних комплексів, що аналізують зібрані дані, генерують звіти та віддають команди для виконання певних дій. Завдяки алгоритмам обробки даних системи можуть оперативно реагувати на зміни та оптимізувати процеси управління об'єктами. Користувачам забезпечується доступ до функціоналу таких систем через інтерфейс, який зазвичай реалізований у вигляді веб-додатка чи мобільного застосунку. Це робить управління зручним, інтуїтивно зрозумілим та доступним у будь-якому місці з інтернет-з'єднанням.

Приклади віддалених систем керування демонструють їх ефективність у різних галузях. Одним із таких прикладів є SCADA-системи, які активно використовуються в промисловості. SCADA дозволяє контролювати обладнання, спостерігати за його станом та автоматизувати процеси

виробництва. Ще одним прикладом є IoT-платформи, такі як AWS IoT Core чи ThingSpeak, які дозволяють збирати дані з безлічі пристроїв та виконувати їх аналіз через хмарні сервіси. У сільському господарстві віддалені системи допомагають контролювати погодні умови, вологість ґрунту та забезпечують автоматичний запуск зрошувальних систем для оптимізації врожаїв.

Проте існують певні виклики, які обмежують ефективність традиційних систем віддаленого керування. Однією з основних проблем є низька швидкість обробки великих обсягів даних у режимі реального часу. Це особливо актуально для систем, які працюють із величезними масивами інформації, що надходить від сенсорів. Іншою проблемою є недостатня адаптивність традиційних систем, які часто не здатні ефективно налаштовуватись на нові умови без додаткових ручних коригувань. Недостатня інтеграція технологій машинного навчання також обмежує можливість автоматизації процесів та прогнозування потенційних загроз чи змін.

Таблиця 1.1 – Порівняльна характеристика традиційного та автоматизованого віддаленого керування

Критерії	Традиційне керування	Автоматизоване керування
Швидкість реагування	Низька	Висока
Роль людини	Вирішальна	Мінімальна
Прогнозування	Відсутнє	Наявне
Залежність від даних	Обмежена	Велика

Порівняльний аналіз традиційного та автоматизованого віддаленого керування демонструє переваги сучасних підходів. Традиційні системи мають низьку швидкість реагування, повністю залежать від рішень людини та не забезпечують прогнозування. Натомість автоматизовані системи характеризуються високою швидкістю обробки даних, мінімальною залежністю від участі людини та широким використанням аналітичних інструментів для прогнозування подій. Це робить сучасні системи ефективнішими, адаптивнішими та більш надійними.

Таким чином, віддалене керування об'єктами на основі сучасних інформаційних технологій є важливим етапом розвитку автоматизованих систем управління. Його основні принципи та компоненти забезпечують можливість ефективного контролю, збору даних і реагування в реальному часі. Завдяки інтеграції IoT-пристроїв та використанню сучасних протоколів зв'язку такі системи відкривають нові можливості для моніторингу й управління у різних сферах діяльності, одночасно вирішуючи виклики, які виникають у процесі впровадження автоматизації.

1.2 Технологічні підходи та інструменти для створення веб-додатків

Сучасні технологічні підходи та інструменти для створення веб-додатків відіграють ключову роль у розробці систем для віддаленого керування об'єктами. Вони охоплюють фронтенд- і бекенд-розробку, а також роботу з базами даних і архітектурними рішеннями, що дозволяють забезпечити ефективну, продуктивну та надійну роботу додатків. Веб-додатки є основним інтерфейсом взаємодії між користувачами та системами керування, а їхня якість та швидкодія безпосередньо залежать від вибраних технологічних інструментів.

На фронтенд-рівні створення веб-додатків зосереджується на розробці зручних та інтерактивних інтерфейсів користувача. У сучасному веб-розробленні провідну роль відіграють JavaScript-фреймворки, серед яких особливо популярними є React.js, Vue.js та Angular. React.js є бібліотекою для створення інтерфейсів користувача, яка отримала популярність завдяки своїй високій продуктивності та гнучкості. Основною особливістю React є використання віртуального DOM, що значно пришвидшує оновлення сторінки й забезпечує ефективну роботу інтерфейсу. React активно застосовується для створення SPA (Single Page Applications) – додатків, які оновлюються без перезавантаження сторінки.

Vue.js – ще один популярний фреймворк, який відзначається своєю

простотою і легкістю. Він надає можливість швидко створювати інтерактивні та адаптивні інтерфейси завдяки своїй доступній синтаксичній структурі. Vue ідеально підходить для невеликих та середніх проєктів, де важлива швидкість розробки. Angular, у свою чергу, є повноцінним фреймворком, що розроблений Google і орієнтований на створення масштабованих додатків. Angular забезпечує модульність, що дозволяє ділити додаток на окремі функціональні блоки, та підтримує інжекцію залежностей, що робить його дуже потужним для великих корпоративних проєктів.

На рівні бекенду, серверна частина веб-додатків відіграє критично важливу роль у забезпеченні обробки запитів, збереженні даних та виконанні бізнес-логіки. Популярними інструментами для бекенд-розробки є платформа Node.js та фреймворк Django. Node.js є платформою для виконання JavaScript-коду на сервері, що дозволяє реалізовувати асинхронну обробку запитів і забезпечує високу продуктивність веб-додатків. Завдяки своїй архітектурі Node.js дозволяє працювати з великим числом одночасних підключень, що робить його ідеальним для створення реального часу застосунків та сервісів.

Django є високорівневим фреймворком, написаним мовою Python, що орієнтований на швидку розробку веб-додатків із вбудованою підтримкою баз даних. Він пропонує багато інструментів для автоматизації процесів, таких як створення API та управління даними, що дозволяє розробникам зосередитись на бізнес-логіці застосунку. Django також підтримує масштабування та є хорошим вибором для складних, багатофункціональних веб-систем.

Бази даних є ще одним важливим елементом при створенні веб-додатків для керування віддаленими об'єктами. Вони забезпечують збереження, обробку та доступ до даних, які надходять від сенсорів, пристроїв або користувачів. Для роботи зі структурованими даними часто використовують реляційні бази даних, такі як PostgreSQL. PostgreSQL є потужною системою управління базами даних, що забезпечує високу продуктивність та надійність. Вона підтримує складні запити й дозволяє зберігати великі обсяги структурованої інформації.

Натомість для обробки неструктурованих або великих обсягів даних

перевага надається нереляційним базам, таким як MongoDB. MongoDB є NoSQL-базою даних, що відзначається гнучкістю та масштабованістю. Вона зберігає дані у вигляді документів JSON, що робить її ідеальним інструментом для роботи з динамічними структурами даних, які часто зустрічаються в системах віддаленого моніторингу.

Важливим аспектом сучасних веб-додатків є архітектурні підходи до побудови взаємодії між фронтендом і бекендом. RESTful API є стандартним підходом до розробки інтерфейсів взаємодії, що забезпечує ефективну комунікацію між клієнтською та серверною частиною. REST-архітектура базується на використанні HTTP-запитів для отримання, додавання, оновлення або видалення даних.

Вона є простою у реалізації та забезпечує високу швидкість обробки запитів.

GraphQL є альтернативою REST і пропонує більш гнучкий підхід до отримання даних. На відміну від REST, де сервер повертає фіксовану структуру даних, GraphQL дозволяє клієнту самому визначати, які саме дані йому потрібні. Це зменшує обсяги переданої інформації та підвищує ефективність роботи додатка, особливо в системах з великою кількістю запитів і даних.

Таким чином, створення веб-додатків для керування віддаленими об'єктами базується на використанні сучасних технологій, що охоплюють як фронтенд, так і бекенд-розробку. JavaScript-фреймворки React, Vue і Angular дозволяють створювати швидкі та зручні інтерфейси, тоді як платформи Node.js та Django забезпечують продуктивну обробку даних на сервері. Використання баз даних PostgreSQL та MongoDB гарантує ефективне збереження як структурованих, так і неструктурованих даних. Архітектурні підходи RESTful API та GraphQL забезпечують гнучку та надійну взаємодію між усіма компонентами системи, створюючи основи для побудови продуктивних та адаптивних веб-додатків.

1.3 Огляд технологій машинного навчання для автоматизації управління

Сучасні технології машинного навчання (ML) відіграють ключову роль в автоматизації процесів моніторингу та управління віддаленими об'єктами. Завдяки застосуванню алгоритмів ML можна підвищити ефективність роботи систем, забезпечити швидке реагування на зміни та мінімізувати вплив людського фактора на процеси управління. Технології машинного навчання дозволяють аналізувати великі обсяги даних, виявляти приховані закономірності та автоматизувати процес прийняття рішень у режимі реального часу.

Одним із найпоширеніших напрямів використання ML є прогнозування, де основну роль відіграють алгоритми регресії. Прогнозування дозволяє передбачати ключові показники функціонування об'єктів, такі як температурний режим обладнання, рівень споживання ресурсів чи технічний стан систем. У цьому контексті ефективно застосовується лінійна регресія, яка дозволяє на основі історичних даних будувати математичні моделі для прогнозування числових значень. Наприклад, за допомогою лінійної регресії можна прогнозувати зміну температури обладнання чи потребу в енергоресурсах на основі наявних трендів.

Таблиця 1.2 – Алгоритми ML для керування віддаленими об'єктами

Метод	Завдання	Приклад застосування
Лінійна регресія	Прогнозування числових значень	Температура обладнання
Дерева рішень	Класифікація станів	Виявлення аномалій у системі
Нейронні мережі	Обробка складних патернів	Прогнозування несправностей

Класифікація є ще одним важливим завданням машинного навчання, яке

дозволяє автоматично визначати стан об'єкта на основі отриманих даних. Алгоритми класифікації розподіляють об'єкти на різні категорії, такі як нормальний стан, критичний стан або аварійний стан. Для вирішення завдань класифікації широко застосовуються дерева рішень, оскільки вони надають зрозумілі моделі, які легко інтерпретувати. Випадковий ліс є розширенням дерев рішень і забезпечує високу точність завдяки об'єднанню результатів кількох дерев у єдину модель. Наприклад, у системах управління обладнанням дерева рішень можна використовувати для визначення, чи працює пристрій у нормальному режимі, чи існує ризик виходу з ладу.

Методи виявлення аномалій також відіграють значну роль у автоматизації процесів моніторингу та управління. Аномалії є відхиленнями від стандартної поведінки системи, і їх своєчасне виявлення дозволяє уникнути потенційних збоїв або аварійних ситуацій. Для цього застосовуються алгоритми машинного навчання, які навчаються на історичних даних і автоматично розпізнають нестандартні відхилення. Наприклад, алгоритми випадкового лісу можна використовувати для ідентифікації аномалій у роботі датчиків обладнання, що може свідчити про неполадки або перевищення критичних показників.

Нейронні мережі є потужним інструментом машинного навчання, який застосовується для обробки складних патернів даних та оптимізації динамічних процесів. Нейронні мережі особливо ефективні у завданнях прогнозування та виявлення прихованих залежностей у великих наборах даних. Завдяки багат шаровій архітектурі нейронні мережі можуть аналізувати нелінійні залежності між параметрами системи та здійснювати точні прогнози. Наприклад, нейронні мережі можна використовувати для прогнозування несправностей обладнання на основі даних, зібраних з датчиків та сенсорів, а також для оптимізації управління ресурсами в реальному часі.

Ключові алгоритми машинного навчання, які застосовуються для автоматизації управління, можна узагальнити наступним чином. Лінійна регресія використовується для прогнозування числових значень, таких як температурний режим або рівень споживання ресурсів. Дерев рішень та

випадковий ліс застосовуються для задач класифікації та виявлення аномалій, оскільки вони забезпечують високу точність та легку інтерпретованість результатів. Нейронні мережі використовуються для складних завдань, які потребують обробки великої кількості даних та побудови оптимізаційних моделей для динамічних процесів.

Сучасні системи моніторингу та управління віддаленими об'єктами активно інтегрують алгоритми машинного навчання для підвищення автоматизації та зменшення потреби у втручанні оператора. Наприклад, у системах керування енергоспоживанням ML-алгоритми прогнозують споживання електроенергії та оптимізують роботу обладнання з урахуванням поточного попиту.

У сільському господарстві алгоритми машинного навчання дозволяють автоматично аналізувати показники вологості ґрунту та погодні умови для оптимізації зрошувальних систем. У промислових системах нейронні мережі використовуються для прогнозування збоїв обладнання та попередження аварійних ситуацій.

Таким чином, технології машинного навчання є важливим елементом автоматизації управління віддаленими об'єктами. Вони дозволяють забезпечити точне прогнозування, автоматичну класифікацію станів об'єктів та виявлення аномалій у роботі систем. Лінійна регресія, дерева рішень, випадковий ліс та нейронні мережі є ключовими інструментами, що забезпечують ефективну обробку даних та автоматизацію управлінських рішень. Завдяки використанню ML-алгоритмів системи стають більш надійними, адаптивними та здатними до швидкого реагування на зміни в умовах експлуатації віддалених об'єктів.

1.4 Аналіз сучасних рішень та систем-аналогів

Штучний інтелект (ШІ) активно застосовується для автоматизації і оптимізації різних процесів на підприємствах. Програми для керування

процесами на основі ШІ використовують алгоритми машинного навчання, нейронні мережі, обробку природної мови та інші технології для аналізу даних, прийняття рішень та автоматизації складних завдань. Такі програми здатні підвищити ефективність, точність та швидкість управлінських процесів, знижуючи ймовірність помилок і покращуючи загальну продуктивність.

Ось кілька прикладів того, як ШІ може бути використаний для керування процесами:

Інтелектуальні системи управління виробництвом (MES). Системи управління виробництвом (Manufacturing Execution Systems, MES) за допомогою ШІ можуть автоматично контролювати та оптимізувати виробничі процеси. Ці системи включають:

Прогнозування потреб в ресурсах: На основі даних про попит, стан обладнання та матеріалів, ШІ може прогнозувати, коли потрібно поповнити запаси або обслуговувати машини.

Моніторинг якості: ШІ може автоматично перевіряти продукцію на відповідність стандартам якості, використовуючи технології комп'ютерного зору.

Планування виробництва: Алгоритми ШІ допомагають оптимізувати виробничі графіки, враховуючи змінні умови, затримки, та попит.

Системи для управління енергоспоживанням. Штучний інтелект може автоматизувати управління енергоспоживанням на підприємстві. Зокрема:

Аналіз енергетичних потоків: ШІ здатний аналізувати дані з різних джерел (енергоспоживання, температури, вологість) та оптимізувати використання енергії для зниження витрат.

Прогнозування потреб в енергії: За допомогою машинного навчання, програма може передбачити, коли споживання енергії збільшиться або зменшиться, і коригувати роботу обладнання.

Автоматизація управління ланцюгами постачання. ШІ може використовуватися для автоматизації та оптимізації ланцюгів постачання:

Прогнозування попиту: На основі великих обсягів даних, ШІ здатний

передбачити, коли і в яких обсягах потрібно замовити матеріали чи продукцію.

Оптимізація маршрутів постачання: Алгоритми ШІ допомагають визначати найефективніші маршрути для доставки товарів з урахуванням різних змінних (погодні умови, пробки, ціни на паливо).

Управління запасами: ШІ дозволяє точно прогнозувати потребу в матеріалах, оптимізуючи розмір запасів на складах.

Роботизовані системи управління персоналом. ШІ може допомогти в управлінні персоналом:

Підбір кандидатів: Алгоритми ШІ можуть аналізувати резюме та інші дані про кандидатів на основі заданих критеріїв і автоматично відбирати найкращих.

Прогнозування ефективності: На основі історичних даних про працівників, ШІ може передбачити їх ефективність та запропонувати індивідуальні рекомендації для підвищення продуктивності.

Інтелектуальні системи для моніторингу та безпеки. ШІ може використовуватися для моніторингу безпеки на підприємстві:

Охорона та моніторинг за допомогою комп'ютерного зору: Програми на основі ШІ можуть здійснювати відеоспостереження, виявляти підозрілі поведінкові патерни або вторгнення та автоматично повідомляти про інциденти.

Прогнозування ризиків та аварій: Машинне навчання може допомогти аналізувати дані з різних датчиків та виявляти аномалії, що вказують на потенційні загрози або аварії, наприклад, витіки газу або перегрівання обладнання.

Системи для автоматизованого обслуговування клієнтів. ШІ активно застосовується в автоматизації обслуговування клієнтів:

Чат-боти та голосові асистенти: Системи на основі ШІ можуть автоматично відповідати на запити клієнтів, вирішувати проблеми, обробляти замовлення та інші запити без втручання людини.

Аналіз відгуків клієнтів: Алгоритми ШІ можуть аналізувати відгуки

клієнтів, визначати задоволеність, виявляти проблеми та пропонувати рішення для їх вирішення.

Програми для аналізу та прийняття рішень. Штучний інтелект також може бути використаний для підтримки прийняття управлінських рішень:

Аналіз великих даних: ШІ здатен аналізувати великі обсяги даних (Big Data) і виділяти ключові інсайти для керівництва підприємства.

Прогнозування результатів: На основі історичних даних, ШІ може передбачати майбутні події або наслідки певних рішень, що дозволяє приймати обґрунтовані рішення в реальному часі.

Програмні платформи та інструменти для керування процесами на основі ШІ:

IBM Watson — набір інструментів для автоматизації бізнес-процесів та аналітики даних на основі ШІ.

Google Cloud AI — платформа для створення та впровадження штучного інтелекту в бізнес-процеси, включаючи машинне навчання і комп'ютерний зір.

Microsoft Azure AI — набір інструментів для розробки ШІ-систем для автоматизації процесів та аналізу даних.

SAP Leonardo — рішення для інтеграції ШІ в управлінські процеси на підприємстві, зокрема для автоматизації виробництва і управління ланцюгами постачання.

Automation Anywhere — система для автоматизації бізнес-процесів, яка використовує ШІ для полегшення рутинних завдань.

У цьому підрозділі здійснено аналіз існуючих платформ та систем, що забезпечують моніторинг і керування віддаленими об'єктами. Розглянуто такі рішення:

- AWS IoT Core – платформа від Amazon для підключення пристроїв, збору даних та автоматизації управління.
- Azure IoT Hub – сервіс від Microsoft, що забезпечує високу продуктивність і масштабованість.

Проведено детальне порівняння рішень за такими критеріями:

- Продуктивність і масштабованість.
- Інтеграція з алгоритмами машинного навчання.
- Надійність та безпека передачі даних. На основі аналізу сформульовано вимоги до розроблюваного додатка та визначено напрями його оптимізації.

Висновки до розділу 1

У результаті проведеного аналізу існуючих систем для віддаленого керування об'єктами та технологій, що застосовуються в цьому напрямку, можна зробити кілька важливих висновків, які підтверджують значну роль автоматизованих рішень для ефективного управління віддаленими об'єктами, а також надають рекомендації для подальшого розвитку даних систем.

Еволюція від традиційного до автоматизованого керування. Традиційне керування віддаленими об'єктами є обмеженим через низьку швидкість реагування, високу залежність від людини і відсутність можливості прогнозування. Водночас автоматизоване керування, яке ґрунтується на використанні новітніх технологій і методів, таких як алгоритми машинного навчання, нейронні мережі та IoT, забезпечує високу швидкість реагування, знижує роль людини в управлінні, а також дозволяє здійснювати прогнозування та адаптивне керування об'єктами.

Технології та інструменти для створення веб-додатків. Сучасні веб-додатки для керування віддаленими об'єктами використовують передові технології фронтенду та бекенду, які дозволяють створювати швидкі, зручні та ефективні інтерфейси. Використання таких фреймворків як React.js, Vue.js, Angular, а також серверних технологій на базі Node.js та Django дозволяє швидко і ефективно обробляти великі обсяги даних, забезпечуючи інтерактивний та зручний інтерфейс для кінцевих користувачів. Інтеграція з різноманітними базами даних, такими як PostgreSQL і MongoDB, дозволяє ефективно зберігати та обробляти інформацію, що надходить від віддалених

пристроїв.

Важливість машинного навчання в автоматизації керування. Машинне навчання є ключовим компонентом для автоматизації управлінських процесів. Алгоритми машинного навчання, зокрема лінійна регресія, дерева рішень та нейронні мережі, дозволяють здійснювати прогнозування стану об'єктів, виявляти аномалії в їх роботі та оптимізувати процеси керування. Такі підходи дозволяють знижувати кількість помилок, підвищувати точність і своєчасність реагування на зміну ситуації. Вони стають невід'ємною частиною автоматизованих систем для моніторингу та управління віддаленими об'єктами.

Аналіз систем на базі штучного інтелекту. Впровадження технологій штучного інтелекту (ШІ) у системи для керування віддаленими об'єктами надає безліч можливостей для оптимізації та автоматизації процесів. Використання ШІ в таких системах дозволяє значно підвищити ефективність керування, поліпшити прогнозування, автоматизувати рутинні завдання, а також зменшити людську помилку. Платформи, такі як AWS IoT Core, Azure IoT Hub і Google Cloud IoT, надають потужні інструменти для збору та обробки даних з пристроїв IoT, а також для інтеграції з алгоритмами ШІ для оптимізації управлінських рішень. Це дозволяє створювати системи з високою продуктивністю, масштабованістю і надійністю.

Визначення вимог до розробки додатка. Після детального аналізу існуючих рішень та систем, можна виділити кілька основних вимог, які повинні бути враховані під час розробки нових систем для керування віддаленими об'єктами. Перш за все, це повинна бути висока продуктивність і масштабованість системи, можливість інтеграції з різноманітними сенсорами і пристроями IoT, а також надійність та безпека передачі даних. Крім того, необхідно враховувати адаптивність системи, здатність її працювати в різних умовах та з різними типами об'єктів.

Напрями для подальшого розвитку. Розвиток технологій машинного навчання, штучного інтелекту, IoT та хмарних платформ відкриває нові можливості для покращення систем керування віддаленими об'єктами. В

майбутньому можна очікувати подальше вдосконалення автоматизації процесів, впровадження більш складних алгоритмів для прогнозування та оптимізації роботи об'єктів. Також важливим напрямом є підвищення рівня безпеки та захисту даних, що є критично важливим при роботі з чутливою інформацією.

Таким чином, технології віддаленого керування, у поєднанні з алгоритмами машинного навчання та платформами на базі штучного інтелекту, є основою для створення ефективних, надійних та масштабованих систем управління віддаленими об'єктами. Вони дозволяють не лише автоматизувати процеси, а й значно покращити точність, швидкість реагування та ефективність управління в різних галузях, таких як промисловість, сільське господарство, енергетика та інші.

2. ПРОЕКТУВАННЯ ВЕБ-ДОДАТКА ДЛЯ КЕРУВАННЯ ВІДДАЛЕНИМИ ОБ'ЄКТАМИ

2.1 Архітектура системи

Архітектура системи є основою для створення веб-додатка, що забезпечує моніторинг та управління віддаленими об'єктами. Вона визначає організацію всіх компонентів, їхню взаємодію та технології, які використовуються для реалізації основних функцій.

Ефективна архітектура повинна відповідати таким вимогам: бути масштабованою для підтримки великої кількості пристроїв, надійною для забезпечення стабільної роботи в реальному часі та безпечною для захисту даних і доступу до системи.

Основні протоколи:

- MQTT — легкий протокол для передачі даних у режимі реального часу.
- HTTP — стандартний протокол для обміну даними між клієнтом і сервером.
- CoAP — протокол для роботи з пристроями з низькою енергоємністю в IoT-середовищах. Також можуть застосовуватися технології зв'язку, такі як LoRaWAN або ZigBee, для передачі даних на великі відстані або в умовах слабого сигналу.

Усі компоненти системи взаємодіють між собою, утворюючи єдину архітектурну модель:

- Фронтенд відправляє запити на сервер для отримання або зміни даних.
- Сервер взаємодіє з базою даних для збереження або отримання інформації та викликає компоненти ML для виконання алгоритмів.
- Інтерфейс з пристроями забезпечує постійний обмін даними між фізичними об'єктами та сервером.

Таблиця 2.1 – Архітектурна система

Компонент	Опис	Технології
Клієнтська частина	Інтерфейс для взаємодії з користувачем, який відображає дані та надає можливість керування об'єктами.	React.js, Vue.js, Angular
Серверна частина	Відповідає за обробку запитів, взаємодію з базою даних та машинним навчанням.	Node.js, Django, Flask
База даних	Місце для зберігання даних про об'єкти, користувачів, історію показників тощо.	PostgreSQL, MongoDB
Компоненти машинного навчання	Алгоритми для автоматичного управління об'єктами на основі зібраних даних.	Scikit-learn, TensorFlow, Keras
Інтерфейс з пристроями	Комунікація з віддаленими об'єктами для зберігання даних і передачі команд на пристрої.	MQTT, HTTP, CoAP, LoRaWAN, ZigBee

Основні компоненти архітектури.

1. Клієнтська частина (фронтенд). Клієнтська частина є інтерфейсом взаємодії користувачів із системою. Основна її задача — візуалізувати дані, що надходять із сенсорів або пристроїв, та надавати користувачам інструменти для управління віддаленими об'єктами.

Для реалізації фронтенду використовуються сучасні JavaScript-бібліотеки та фреймворки, такі як:

- React.js — забезпечує високу швидкість оновлення інтерфейсу завдяки використанню віртуального DOM.
- Vue.js — проста у використанні технологія, що дозволяє швидко

створювати інтерактивні інтерфейси.

- Angular — потужний фреймворк для розробки складних веб-додатків з багатим функціоналом. Клієнтська частина підтримує інтерактивність, адаптивність і дозволяє працювати з великою кількістю даних у реальному часі.

2. Серверна частина (бекенд). Сервер відповідає за обробку запитів від клієнтської частини, управління бізнес-логікою системи, взаємодію з базою даних та виконання алгоритмів машинного навчання.

Популярні технології для реалізації бекенду:

- Node.js — забезпечує високу продуктивність і асинхронну обробку запитів.
- Django — фреймворк Python для створення безпечних та функціональних веб-додатків.
- Flask — легкий і гнучкий фреймворк для невеликих серверних рішень. Сервер також обробляє дані, що надходять від пристроїв, і забезпечує зворотний зв'язок, наприклад, для активації команд або відправлення повідомлень.

3. База даних. Для зберігання даних використовуються сучасні системи управління базами даних, які підтримують роботу зі структурованими та неструктурованими даними:

- PostgreSQL — ефективна реляційна база даних для обробки структурованих даних.
- MongoDB — нереляційна база даних, що зберігає дані у вигляді документів JSON, дозволяє працювати з великими обсягами динамічної інформації. База даних зберігає інформацію про користувачів, пристрої, параметри об'єктів, а також історію подій та аномалій. Використання індексів і зовнішніх ключів покращує швидкість доступу до даних і забезпечує їхню цілісність.

4. Компоненти машинного навчання (ML). Алгоритми машинного навчання є важливим елементом системи, оскільки забезпечують

автоматизацію управління віддаленими об'єктами.

Використовувані технології:

- Scikit-learn — набір інструментів для реалізації базових алгоритмів ML, таких як класифікація або регресія.
- TensorFlow — потужна бібліотека для роботи з нейронними мережами та глибоким навчанням.
- Keras — інструмент для швидкого створення та тренування моделей ML. Завдяки цим технологіям система може прогнозувати майбутній стан об'єктів, виявляти аномалії та виконувати автоматичні дії.

5. Інтерфейс з пристроями. Для комунікації з віддаленими об'єктами, отримання даних від сенсорів та передачі команд використовується спеціалізований інтерфейс.

Таблиця описує основні компоненти архітектури веб-додатка, їх функції та технології, які можуть бути використані для реалізації кожного з цих компонентів. Це допомагає зрозуміти структуру системи та її взаємодію між частинами.

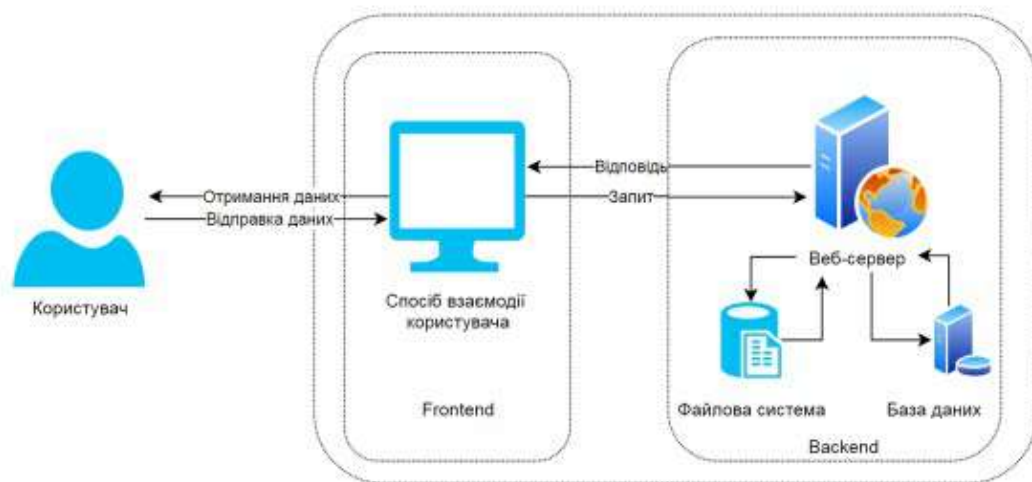


Рисунок 2.1 – Архітектура веб-додатку

Архітектура веб-додатка включає кілька основних компонентів:

Архітектура системи має бути модульною, щоб дозволяти додавання

нових функцій або компонентів без істотних змін у коді. Це забезпечує масштабованість і гнучкість рішення, необхідні для роботи в умовах постійного зростання кількості підключених пристроїв або обсягів даних.

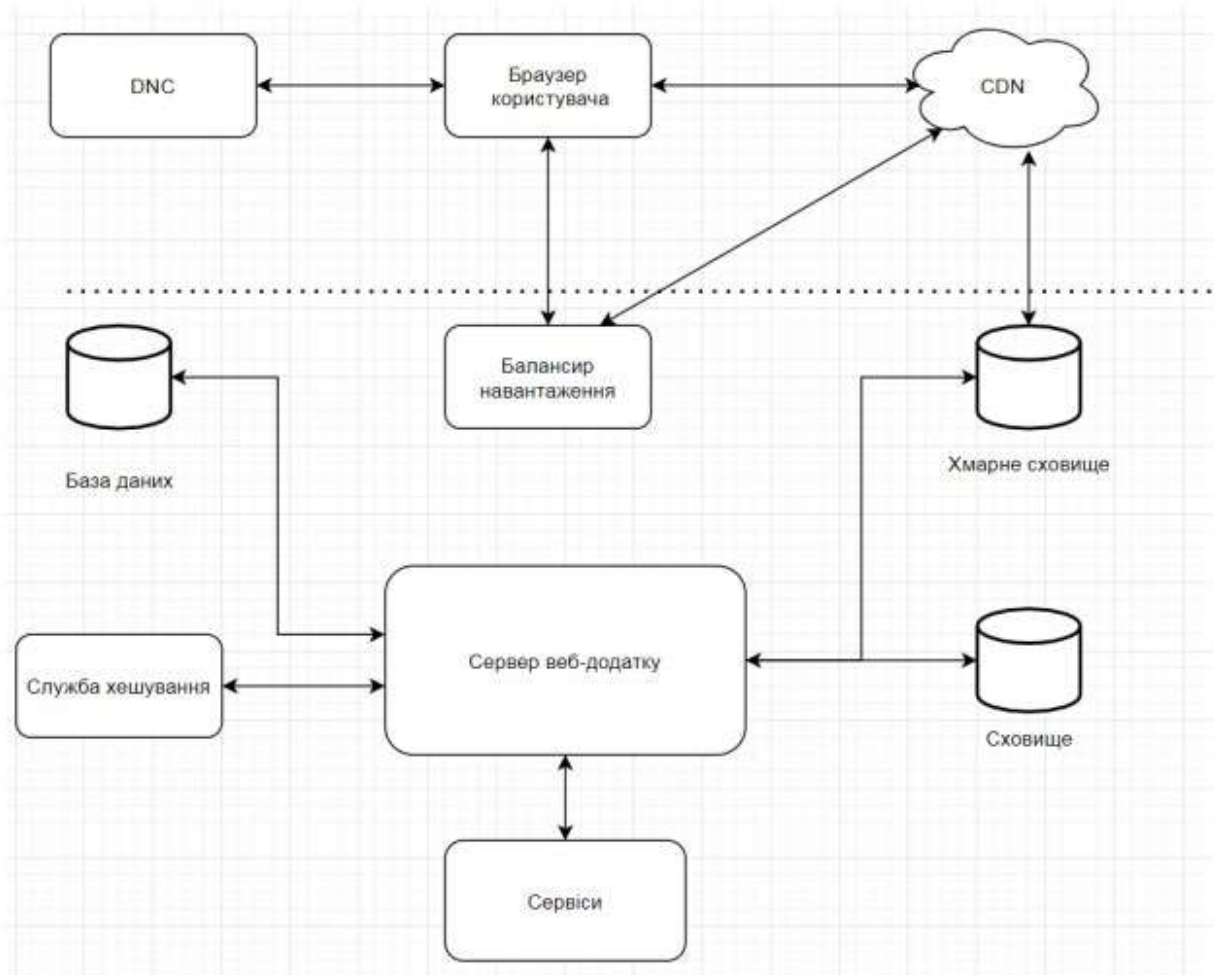


Рисунок 2.1–Схема архітектури веб-додатку

2.2 Постановка задачі: функціональні та нефункціональні вимоги

Постановка задачі є ключовим етапом проєктування веб-додатка для моніторингу та керування віддаленими об'єктами, оскільки вона визначає основні функціональні можливості системи та нефункціональні характеристики, які впливатимуть на продуктивність, безпеку та масштабованість рішення. Чітко визначені вимоги дозволяють сформувати основу для подальшої розробки архітектури додатка та його реалізації.

Функціональні вимоги описують конкретні завдання, які веб-додаток повинен виконувати для забезпечення ефективного управління віддаленими об'єктами. Одним із головних завдань є моніторинг стану об'єктів у режимі реального часу. Система повинна мати можливість отримувати дані з сенсорів і пристроїв, які контролюють параметри, такі як температура, вологість, тиск та інші показники. Ці дані повинні оброблятися та візуалізуватися на інтерфейсі користувача для швидкого аналізу та прийняття рішень.

Іншою важливою функціональною вимогою є автоматичне керування об'єктами. Система повинна мати механізми для автоматичного реагування на зміни параметрів за допомогою алгоритмів машинного навчання. Наприклад, якщо температура пристрою виходить за межі встановлених норм, система повинна автоматично виконати дії, що дозволять стабілізувати його роботу, зокрема ввімкнути охолодження або зупинити певні процеси. Застосування алгоритмів машинного навчання також передбачає можливість прогнозування потенційних проблем та оптимізації роботи обладнання на основі отриманих даних.

Для забезпечення безпеки роботи системи передбачено функціонал автоматичного повідомлення про аварії. У разі виявлення аномалій або критичних ситуацій система повинна автоматично генерувати сповіщення та надсилати їх користувачам через інтерфейс додатка або інші канали зв'язку, такі як електронна пошта чи SMS. Це дозволить своєчасно реагувати на можливі загрози та мінімізувати ризики простою або виходу з ладу обладнання.

Також система повинна надавати інтерфейс для ручного керування об'єктами. Користувачі повинні мати можливість змінювати параметри та стани об'єктів через зручний та інтуїтивно зрозумілий інтерфейс. Це дозволяє операторам не лише контролювати роботу системи, а й втручатися у процеси керування в разі необхідності, наприклад, для виконання налаштувань або екстрених дій.

Нефункціональні вимоги визначають якісні характеристики системи, що

забезпечують її надійну та ефективну роботу. Однією з ключових нефункціональних вимог є продуктивність. Система повинна забезпечувати високу швидкість обробки запитів навіть у разі одночасного надходження великої кількості даних з численних віддалених об'єктів. Збереження та обробка інформації повинні відбуватися без затримок, щоб користувачі мали доступ до актуальних даних у режимі реального часу.

Безпека є критично важливим аспектом роботи веб-додатка. Система повинна забезпечувати захист даних, що передаються та зберігаються, а також механізми автентифікації та авторизації для обмеження доступу до функціоналу додатка. Реалізація протоколів безпечного зв'язку, таких як HTTPS, а також шифрування даних допоможе запобігти несанкціонованому доступу та втраті інформації.

Масштабованість системи є ще однією важливою нефункціональною вимогою. Веб-додаток повинен бути готовим до розширення для обробки додаткових віддалених об'єктів та сенсорів. Збільшення кількості підключених пристроїв не повинно впливати на продуктивність та ефективність роботи системи. Для цього важливо застосовувати сучасні архітектурні підходи, такі як мікросервісна архітектура, що забезпечує легке розширення функціоналу.

Надійність системи передбачає її здатність працювати безперебійно у будь-яких умовах. Веб-додаток повинен мати високий рівень доступності, що означає мінімізацію часу простою та стійкість до можливих відмов. У разі виникнення неполадок система повинна автоматично відновлювати свою роботу або забезпечувати збереження даних для подальшого відновлення.

Таким чином, постановка задачі на основі функціональних та нефункціональних вимог є основою для створення ефективної системи моніторингу та управління віддаленими об'єктами.

Функціональні вимоги забезпечують виконання ключових завдань, таких як моніторинг, автоматичне керування, аварійні повідомлення та можливість ручного втручання. Нефункціональні вимоги, зокрема продуктивність, безпека, масштабованість та надійність, визначають якісні характеристики системи,

необхідні для її стабільної та ефективної роботи в реальних умовах експлуатації.

2.3. Розробка моделі даних та бази даних

Модель даних та структура бази даних є одним із ключових етапів проектування інформаційної системи. Вона визначає спосіб організації, зберігання та обробки даних, які використовуються для моніторингу та управління віддаленими об'єктами. Оптимальна модель даних забезпечує ефективне функціонування системи, її масштабованість і надійність у роботі з великими обсягами інформації.

База даних повинна включати кілька основних таблиць, які відповідають за збереження різних аспектів роботи системи:

1. Таблиця користувачів. Ця таблиця зберігає інформацію про всіх користувачів системи, включаючи їх облікові дані та роль у системі (наприклад, адміністратор або оператор).

Поля :

- user_id (PK): Унікальний ідентифікатор користувача.
- username: Логін користувача.
- password: Хешований пароль для забезпечення безпеки.
- role: Роль користувача (наприклад, “admin”, “operator”).

Ця таблиця використовується для аутентифікації та авторизації користувачів, а також для визначення їхніх прав доступу.

2. Таблиця пристроїв. Містить дані про всі підключені віддалені пристрої, включаючи їхні характеристики, статус і можливості.

Поля:

- device_id (PK): Унікальний ідентифікатор пристрою.
- device_name: Назва пристрою.
- status: Поточний статус пристрою (наприклад, “active”, “inactive”).

Таблиця дозволяє зберігати базову інформацію про пристрої та

забезпечує зв'язок із іншими таблицями для зберігання їхніх параметрів і подій.

3. Таблиця показників. Ця таблиця містить історію даних, які надходять від пристроїв у реальному часі. Вона дозволяє відстежувати такі параметри, як температура, вологість і тиск.

Поля:

- measurement_id (PK): Унікальний ідентифікатор запису.
- device_id (FK): Зовнішній ключ, який пов'язує показник із пристроєм.
- timestamp: Час запису показника.
- temperature: Значення температури.
- humidity: Значення вологості.
- pressure: Значення тиску.

Таблиця показників використовується для аналізу часового ряду даних і є основою для реалізації прогнозування та автоматичного реагування.

4. Таблиця подій. Використовується для логування важливих подій у системі, таких як аварії, зміни параметрів пристроїв або інші критичні ситуації.

Поля:

- event_id (PK): Унікальний ідентифікатор події.
- device_id (FK): Зовнішній ключ для зв'язку події з конкретним пристроєм.
- event_type: Тип події (наприклад, “error”, “warning”, “update”).
- timestamp: Час події.
- description: Опис події.

Завдяки цій таблиці система може вести журнал подій для подальшого аналізу, реагування або звітності. Аспекти розробки моделі даних

1. Типи даних. База даних повинна зберігати різні типи даних, включаючи числові (для показників), текстові (для опису подій) та дані часового ряду (для відстеження змін параметрів у часі). Використання правильних типів даних (наприклад, INTEGER, TEXT, TIMESTAMP) дозволяє зберігати інформацію ефективно й уникати надмірного споживання ресурсів.

2. Структура таблиць. Кожна таблиця має чітко визначені зв'язки та унікальні ключі. Наприклад, `user_id` у таблиці користувачів є первинним ключем, а `device_id` у таблиці показників є зовнішнім ключем, що забезпечує зв'язок із таблицею пристроїв.

3. Індеси. Індксація ключових полів (наприклад, `timestamp` у таблиці показників) дозволяє пришвидшити обробку запитів, особливо коли йдеться про великі обсяги даних, що зберігаються в базі.

4. Зв'язки між таблицями. Всі таблиці пов'язані за допомогою зовнішніх ключів. Наприклад, таблиця показників посилається на таблицю пристроїв через поле `device_id`. Це забезпечує цілісність даних і дозволяє виконувати складні запити для аналізу.

5. Оптимізація зберігання. Для зменшення навантаження на базу даних можна використовувати архівування застарілих даних або стиснення даних показників, які не потребують частого доступу. Також можна налаштувати резервне копіювання, щоб уникнути втрати інформації.

Реалізація моделі даних. Розроблена структура бази даних дозволяє системі:

- Зберігати всі необхідні дані про користувачів, пристрої, їхні показники та події.

Таблиця 2.2 – Модель даних та структура бази даних

Таблиця	Опис	Поля/Ключі
Таблиця користувачів	Зберігає інформацію про користувачів системи (логін,пароль,роль).	<code>user_id</code> (PK), <code>username</code> , <code>password</code> , <code>role</code>
Таблиця пристроїв	Містить дані про віддалені пристрої, їх характеристики і статус.	<code>Device_id</code> (PK), <code>device_name</code> , <code>status</code>
Таблиця показників	Зберігає історію вимірів з пристроїв: температура, вологість, тиск.	<code>measurement_id</code> (PK), <code>device_id</code> (FK), <code>timestamp</code> , <code>temperature</code> , <code>humidity</code> ,

		pressure
Таблиця подій	Логуювання важливих подій та аномалій в системі, таких як аварії або зміни стану пристроїв.	,event_id (PK), device_id (FK), event_type, timestamp, description

- Легко масштабуватись, додаючи нові пристрої або типи даних.
- Швидко обробляти запити завдяки індексації та оптимізованим зв'язкам між таблицями.
- Забезпечувати аналіз даних у реальному часі для прийняття рішень або прогнозування стану пристроїв.

Оптимальна модель даних є критично важливою для забезпечення надійної роботи веб-додатка. Вона дозволяє ефективно зберігати, обробляти та аналізувати великі обсяги даних, що надходять від віддалених об'єктів. Чітка структура таблиць, правильно обрані типи даних і зв'язки між ними створюють основу для реалізації функціональності системи, включаючи моніторинг, аналіз і автоматичне управління.

2.4. Використання технологій машинного навчання для автоматизованого керування

Використання технологій машинного навчання є ключовим компонентом для автоматизації процесів управління віддаленими об'єктами, оскільки ці технології дозволяють системам не лише аналізувати поточний стан об'єктів, а й прогнозувати їх поведінку, виявляти аномалії та реагувати на них без необхідності втручання людини. Інтеграція машинного навчання у систему забезпечує значне підвищення ефективності, надійності та автономності керування.

Одним із найважливіших завдань машинного навчання є прогнозування стану об'єкта. Для цього застосовуються методи регресії, що дозволяють передбачати майбутні значення ключових параметрів на основі історичних

даних. Наприклад, якщо система моніторить температуру обладнання протягом певного часу, алгоритм лінійної регресії може спрогнозувати зміну температурного режиму на основі трендів у даних. Це дозволяє заздалегідь виявляти потенційні відхилення та вживати необхідні заходи, щоб запобігти збоям чи аваріям. Прогнозування є особливо ефективним для процесів, які залежать від стабільних або періодичних показників, що дозволяє оптимізувати витрати ресурсів та підтримувати працездатність системи.

Ще однією важливою функцією є виявлення аномалій та відхилень у роботі об'єктів. Для цього використовуються алгоритми машинного навчання, такі як дерева рішень, методи кластеризації та нейронні мережі. Ці методи дозволяють аналізувати великі обсяги даних, щоб визначати нестандартні або критичні відхилення від нормальних параметрів роботи системи. Наприклад, якщо в процесі роботи датчик виявляє різке підвищення тиску або температури, система може автоматично ідентифікувати це як аномалію. Виявлення аномалій дозволяє швидко реагувати на можливі збої, зменшуючи ризики аварійного відключення обладнання та втрат ресурсів.

На основі отриманих прогнозів та виявлених аномалій можливе автоматизоване керування роботою об'єктів. У таких випадках система може самостійно змінювати параметри роботи обладнання або середовища відповідно до заданих алгоритмів. Наприклад, якщо система передбачає підвищення температури в приміщенні до критичного рівня, вона може автоматично активувати охолоджувальні пристрої. Також можливе налаштування оптимального режиму роботи обладнання для підвищення його енергоефективності чи продуктивності. Такий підхід дозволяє мінімізувати втручання оператора та забезпечити автономну роботу системи.

У складніших сценаріях управління використовується навчання з підкріпленням, що є особливо ефективним для систем, які потребують прийняття рішень у режимі взаємодії з навколишнім середовищем. У цьому випадку алгоритм навчається на основі постійного зворотного зв'язку, оцінюючи результат своїх дій і підлаштовуючи стратегію для досягнення

найкращого результату. Наприклад, у промислових системах навчання з підкріпленням може використовуватись для оптимізації процесів виробництва або налаштування обладнання в динамічному середовищі. Алгоритм отримує винагороду за досягнення потрібних результатів і штрафується за помилкові дії, що дозволяє йому поступово знаходити оптимальну стратегію керування.

Застосування технологій машинного навчання для автоматизованого управління віддаленими об'єктами відкриває нові можливості для підвищення продуктивності та надійності систем. Прогнозування дозволяє заздалегідь планувати необхідні заходи, а виявлення аномалій забезпечує своєчасне реагування на критичні ситуації. Автоматизоване керування знижує залежність від людини та підвищує ефективність використання ресурсів. Використання навчання з підкріпленням дозволяє оптимізувати дії системи навіть у нестандартних умовах, забезпечуючи гнучке та адаптивне управління. Такий підхід створює основу для інтелектуальних систем, які здатні самостійно аналізувати ситуацію, ухвалювати рішення та підтримувати безперебійну роботу обладнання у режимі реального часу.

2.5. Проектування інтерфейсу користувача

Проектування інтерфейсу користувача є одним із ключових етапів розроблення системи, оскільки від його якості залежить ефективність і зручність взаємодії кінцевих користувачів із додатком. Основна мета інтерфейсу полягає в тому, щоб забезпечити швидкий доступ до необхідної інформації, просте керування віддаленими об'єктами та оперативне реагування на критичні ситуації. У процесі проектування враховуються особливості цільової аудиторії, вимоги до функціональності та принципи адаптивного дизайну.

Одним із найважливіших елементів інтерфейсу є інтерфейс для моніторингу, який реалізується у вигляді інтерактивних дашбордів. На дашбордах відображається актуальна інформація про стан віддалених об'єктів,

зокрема ключові параметри, такі як температура, тиск, вологість, споживання енергії тощо. Для наочності дані подаються у вигляді графіків, таблиць і діаграм, що дозволяє користувачеві швидко оцінити поточну ситуацію. Використання технологій React.js для створення інтерфейсів у поєднанні з D3.js або Chart.js для візуалізації даних забезпечує високу продуктивність та динамічне оновлення інформації у реальному часі.

Другим важливим компонентом є інтерфейс для управління, що дозволяє користувачам змінювати параметри віддалених об'єктів, включати або вимикати пристрої та виконувати інші керуючі команди. Важливо, щоб елементи цього інтерфейсу були інтуїтивно зрозумілими та зручними для використання. Для реалізації форм керування використовуються сучасні технології фронтенд-розробки, зокрема React.js разом із бібліотеками Bootstrap та Material UI, які забезпечують сучасний вигляд інтерфейсу та зручність його використання. Кнопки, перемикачі, випадаючі списки та інші інтерактивні елементи дозволяють користувачеві легко налаштовувати роботу системи.

Таблиця 2.3 – Проектування інтерфейсу користувача

Елемент інтерфейсу	Опис	Технології
Дашборд	Інтерфейс для перегляду стану всіх віддалених об'єктів, показників і стану системи.	React.js, D3.js, Chart.js
Форма керування	Елементи для введення команд користувача для зміни налаштувань об'єктів (включення/вимкнення пристроїв).	React.js, Bootstrap, Material UI
Сповіщення	Система повідомлень для відображення критичних ситуацій, аварій та інших подій.	Socket.io, Push Notifications API
Адаптивний дизайн	Підтримка відображення на	CSS3, Bootstrap, Media

	різних пристроях, забезпечуючи зручність використання на мобільних телефонах та планшетах.	Queries
--	---	---------

Особливу роль у проектуванні відіграє система сповіщень, яка забезпечує інформування користувача про критичні ситуації, аварії або зміни у роботі об'єктів. Повідомлення можуть бути реалізовані у вигляді спливаючих вікон, push-сповіщень або звукових сигналів, що привертають увагу користувача. Це дозволяє оперативно реагувати на проблеми, запобігаючи можливим збоям у системі. Для реалізації системи сповіщень використовуються технології, такі як Socket.io для обміну даними у реальному часі та Push Notifications API для надсилання сповіщень на мобільні та веб-пристрої.

Ще одним важливим аспектом проектування є адаптивний дизайн, що забезпечує зручність використання інтерфейсу на різних пристроях, включаючи комп'ютери, планшети та смартфони. У сучасному середовищі користувачі все частіше використовують мобільні пристрої для керування віддаленими об'єктами, тому інтерфейс повинен автоматично підлаштовуватися під різні розміри екранів і роздільну здатність. Для досягнення адаптивності застосовуються CSS3, Bootstrap та Media Queries, що дозволяють оптимізувати відображення елементів інтерфейсу та забезпечити зручний доступ до функціоналу системи з будь-якого пристрою.

Таким чином, проектування інтерфейсу користувача базується на чітко визначених вимогах до моніторингу, керування, сповіщень та адаптивності. Інтерактивні дашборди забезпечують наочний моніторинг параметрів об'єктів, інтуїтивно зрозумілі форми керування дозволяють швидко змінювати налаштування, а система сповіщень гарантує оперативне реагування на критичні події. Адаптивний дизайн забезпечує зручність використання системи на різних пристроях, що є важливим фактором для сучасних веб-додатків. Поєднання передових технологій та ретельно спроектованого інтерфейсу

дозволяє створити ефективну, надійну та зручну систему для управління віддаленими об'єктами.

Проектування інтерфейсу користувача (UI) є критично важливим етапом у розробці будь-якої системи, особливо коли йдеться про складні інтерактивні платформи для моніторингу та управління віддаленими об'єктами. Інтерфейс повинен бути зрозумілим та інтуїтивно простим, щоб користувачі могли швидко освоїти його та ефективно використовувати для реагування на критичні ситуації або зміни в роботі системи.

Один з основних аспектів проектування — це забезпечення швидкого доступу до інформації в реальному часі. Наприклад, інтерфейси для моніторингу повинні надавати користувачеві можливість візуалізувати дані про віддалені об'єкти в динамічному вигляді, з інтуїтивно зрозумілими графіками та діаграмами. Це дозволяє оперативно оцінити поточну ситуацію, реагувати на можливі аномалії чи критичні показники та здійснювати необхідні коригування.

Підтримка реального часу. У системах, що працюють з віддаленими об'єктами, важливим елементом є моніторинг в реальному часі. Для цього застосовуються технології, які забезпечують миттєве оновлення даних, що допомагає користувачеві відразу реагувати на зміни в системі. Наприклад, використання бібліотек, таких як D3.js або Chart.js для візуалізації даних, дозволяє не тільки створювати динамічні графіки, а й інтегрувати їх із серверними рішеннями для миттєвого відображення оновленої інформації, зокрема, на моніторах диспетчерських або управлінських панелей.

Гнучкість та адаптивність. Сучасні користувачі потребують системи, яка може бути доступною не лише на великих екранах, але й на мобільних пристроях. Зважаючи на це, адаптивний дизайн стає необхідним елементом для забезпечення функціональності інтерфейсу на різних пристроях. Використання таких інструментів, як CSS3 і Media Queries дозволяє підлаштовувати інтерфейс під різні типи пристроїв, оптимізуючи елементи інтерфейсу для смартфонів, планшетів і ПК. Це дозволяє користувачам мати

безперервний доступ до інформації та можливість управління з будь-якого місця та за будь-яких обставин.

Пріоритет зручності та швидкості взаємодії. Ще однією важливою частиною проектування є надання можливості швидко та ефективно виконувати необхідні дії. Інтерфейс для керування має бути інтуїтивно зрозумілим, щоб користувачі могли без додаткових зусиль включати або вимикати пристрої, змінювати налаштування або виконувати інші операції. Потрібно враховувати, що в екстремальних ситуаціях важлива не тільки правильність дії, а й швидкість виконання команд. Тому у формі керування важливо продумати простоту використання кнопок, перемикачів і списків, щоб користувачі могли за кілька кліків здійснювати операції.

Система сповіщень. У контексті критичних ситуацій особливу роль відіграють сповіщення про зміни в стані об'єктів або аварійні ситуації. Система сповіщень повинна забезпечувати миттєве інформування користувачів про важливі події через різні канали — через спливаючі вікна на екрані, push-сповіщення на мобільних пристроях, або навіть звукові сигнали для привертання уваги в разі надзвичайних ситуацій. Це важливо для того, щоб користувачі могли швидко реагувати на загрози, не витрачаючи час на пошук необхідної інформації.

Для інтеграції такої системи сповіщень до інтерфейсу часто використовуються бібліотеки і технології, що підтримують обмін даними в реальному часі, такі як Socket.io. Це дозволяє отримувати дані від серверної частини додатка та відразу передавати їх користувачеві, забезпечуючи своєчасне інформування про критичні ситуації.

Підвищення безпеки та стійкості інтерфейсу. Оскільки інтерфейси для моніторингу й управління віддаленими об'єктами часто працюють у важливих системах, питання безпеки стають пріоритетними. Проектуючи інтерфейс, необхідно враховувати захист від несанкціонованого доступу, використовувати шифрування даних та надійні механізми автентифікації. Також важливо забезпечити стійкість системи до можливих технічних збоїв,

впроваджуючи резервне збереження даних, автоматичне відновлення роботи після збоїв і підтримку безперервного доступу до критичної інформації.

Завдяки використанню передових технологій, таких як React.js, D3.js, Bootstrap та інших бібліотек для створення інтерфейсів, можна забезпечити високий рівень інтерактивності та адаптивності, що робить управління віддаленими об'єктами більш зручним і ефективним. У кінцевому рахунку, проектування інтерфейсу користувача має на меті створення такої системи, що дозволить користувачам з максимальною швидкістю реагувати на будь-які зміни в умовах виробничих процесів або надзвичайних ситуацій.

Висновки до розділу 2

Розробка веб-додатку для управління віддаленими об'єктами передбачає впровадження комплексної архітектури, яка включає кілька ключових компонентів: клієнтську частину, серверну частину, базу даних, систему для автоматичного управління через алгоритми машинного навчання, а також інтерфейси для комунікації з пристроями. Враховуючи широкий спектр вимог до системи, архітектура повинна бути масштабованою, надійною та безпечною, що дозволяє забезпечити високий рівень ефективності та доступності веб-додатку.

Архітектура системи та технології. Веб-додаток для віддаленого керування об'єктами має складну багаторівневу архітектуру, що включає фронтенд (клієнтську частину), бекенд (серверну частину), базу даних, компоненти машинного навчання та інтерфейс з пристроями. Технології для фронтенду, такі як React.js, Vue.js або Angular, дозволяють створювати сучасні, інтерактивні інтерфейси, що підтримують реальний час. На серверній частині використовуються популярні фреймворки, такі як Node.js, Django або Flask, що забезпечують ефективну обробку запитів і взаємодію з базою даних. Важливою складовою є база даних, де зберігаються всі показники пристроїв, історія даних, а також інформація про користувачів. Для забезпечення

ефективної комунікації між віддаленими пристроями та сервером використовуються різні протоколи, такі як MQTT, HTTP, CoAP, що дозволяють передавати дані з високою швидкістю та надійністю.

Функціональні та нефункціональні вимоги. Визначення чітких функціональних вимог є необхідним для успішної реалізації системи. Веб-додаток повинен забезпечувати моніторинг стану об'єктів в режимі реального часу, підтримувати автоматичне управління через алгоритми машинного навчання, генерувати сповіщення про аварії та дозволяти користувачам вручну коригувати параметри пристроїв. Крім того, система повинна бути надійною, безпечною і масштабованою. Продуктивність системи є критично важливою, оскільки додаток має працювати без затримок, навіть при великих обсягах даних. Важливими нефункціональними вимогами є безпека, яка передбачає використання механізмів автентифікації та авторизації, а також масштабованість і надійність системи, що дозволяє обробляти додаткові пристрої без зниження ефективності.

Модель даних та структура бази даних. Для зберігання даних в системі необхідно розробити правильну модель даних та структуру бази даних. Модель включає таблиці для зберігання інформації про користувачів, пристрої, показники та події. Важливою частиною є забезпечення зв'язків між таблицями для ефективного обробки запитів та зберігання даних. Для забезпечення високої продуктивності потрібно створити індекси на ключові поля, що дозволить швидко здійснювати пошук та обробку великих обсягів даних. Крім того, важливою є оптимізація зберігання даних, зокрема через стиснення та архівацію даних, щоб зберегти історію параметрів, при цьому не перевантажуючи систему.

Використання машинного навчання для автоматизованого керування. Інтеграція машинного навчання є важливим аспектом для автоматизації процесу управління віддаленими об'єктами. Алгоритми машинного навчання, зокрема регресія, допомагають прогнозувати стан об'єктів, а виявлення аномалій дозволяє своєчасно виявляти та виправляти критичні відхилення в їх

роботі. Система може автоматично регулювати параметри роботи об'єктів, наприклад, включати або вимикати пристрої, змінювати температурний режим і так далі. Для складних сценаріїв, де необхідно оптимізувати дії на основі взаємодії з середовищем, використовується навчання з підкріпленням, що дозволяє системі самостійно адаптуватися та покращувати свої рішення.

Проектування інтерфейсу користувача. Інтерфейс користувача є важливою частиною додатка, адже він має забезпечити зручність і інтуїтивність у використанні. Розробка інтерфейсу передбачає створення зручного дашборду для моніторингу стану об'єктів, а також форм для керування, де користувачі можуть змінювати параметри пристроїв. Сповіщення про аварії та критичні ситуації є важливою функціональністю для оперативного реагування. Оскільки сучасні користувачі мають різні пристрої для доступу до системи, адаптивний дизайн дозволяє забезпечити комфортне використання як на комп'ютерах, так і на мобільних телефонах та планшетах.

Проектування системи для управління віддаленими об'єктами є складним, багатокомпонентним процесом, який вимагає ретельного підходу до вибору технологій і забезпечення високої продуктивності та безпеки. Успішна реалізація цього проекту дозволить створити ефективний інструмент для моніторингу і керування віддаленими об'єктами, що здатний автоматично реагувати на зміни, оптимізуючи роботу об'єктів і забезпечуючи високий рівень надійності та зручності для користувачів.

3 ВИКОРИСТАННЯ ОБ'ЄКТУ МОДЕЛЮВАННЯ ДЛЯ ПРОЦЕСІВ ОХОРОНИ ПРАЦІ ПІД ЧАС НАДЗВИЧАЙНИХ СИТУАЦІЙ НА ПІДПРИЄМСТВІ

3.1 Роль об'єктів моделювання в оптимізації заходів охорони праці під час НС

Моделювання надзвичайних ситуацій (НС) стало важливим і незамінним інструментом для підприємств у розробці, оптимізації та вдосконаленні заходів охорони праці. Це особливо актуально в умовах швидко змінюваного виробничого середовища, високих технологій і постійно зростаючих ризиків. Впровадження об'єктів моделювання дозволяє підприємствам створювати віртуальні моделі робочих процесів, виробничих ліній, умов праці та потенційних небезпек, що дозволяє ретельно аналізувати й прогнозувати різні сценарії надзвичайних ситуацій. Таким чином, це дає можливість заздалегідь підготуватися до можливих загроз, розробляти стратегії реагування на них та значно знижувати ймовірність виникнення нещасних випадків.

Основною перевагою використання об'єктів моделювання є можливість прогнозування та розробки адаптивних заходів для кожного конкретного підприємства, враховуючи його специфіку, масштаб виробництва, умови праці та інші фактори. Моделювання дозволяє змодельовати не лише самі техногенні катастрофи, але й людський фактор, поведінку персоналу в екстремальних ситуаціях, що допомагає покращити управління людськими ресурсами в умовах НС. Це в свою чергу дозволяє оптимізувати ресурси, які необхідні для швидкого і ефективного реагування на виникнення небезпеки.

Сучасні системи моделювання дають змогу створювати точні віртуальні копії реальних підприємств, відтворюючи їх у деталях, з урахуванням всіх виробничих процесів і потенційних загроз. Моделюючи надзвичайні ситуації в таких віртуальних умовах, можна вивчати різні сценарії розвитку подій, оцінювати можливі наслідки та виробляти стратегії реагування до того, як

ситуація насправді стане критичною. Це дозволяє не лише заздалегідь виявляти слабкі місця в існуючих системах безпеки, але й удосконалювати ці системи на основі досліджень.

Одним із важливих аспектів використання об'єктів моделювання є інтеграція їх у системи оповіщення та евакуації на підприємстві. Під час виникнення надзвичайної ситуації, швидке та ефективне оповіщення персоналу має вирішальне значення для запобігання масштабним втратам. За допомогою моделювання можна протестувати різні варіанти евакуаційних маршрутів і оцінити, як різні фактори впливають на швидкість і ефективність евакуації. Сюди можна включити розміщення обладнання, кількість людей на різних ділянках підприємства, а також наявність перешкод або складних умов. Моделювання таких сценаріїв дозволяє підприємствам створити точні та адаптивні плани евакуації, що враховують реальні умови і допомагають знизити ризики для здоров'я та життя працівників.

Завдяки застосуванню моделей можна також імітувати різні варіанти техногенних катастроф, таких як пожежі, витіки токсичних і вибухових речовин, затоплення чи технічні аварії. Такі моделі дозволяють прогнозувати наслідки цих подій для підприємства, навколишнього середовища та людей, що знаходяться на виробництві. Завдяки цим прогнозам можна розробляти докладні інструкції для персоналу, щоб мінімізувати шкоду та забезпечити швидке відновлення нормальних умов праці. У разі виникнення реальної ситуації, працівники будуть чітко обізнані з планом дій, що значно знижує ймовірність помилок і затримок у реагуванні.

Застосування об'єктів моделювання дає можливість створювати адаптивні та перевірені плани дій, що дозволяють підприємствам своєчасно реагувати на нові загрози та знижувати ризики. Важливою частиною цього процесу є постійне оновлення моделей на основі нових даних, які надходять під час реальних аварій або технічних обстежень. Таким чином, можна забезпечити своєчасну актуалізацію заходів безпеки, а також покращити систему прогнозування та запобігання аваріям. Це дозволяє підприємствам

залишатися на передовій лінії захисту здоров'я працівників і запобігання техногенним катастрофам.

Використання моделей для оцінки техногенних катастроф дозволяє не лише прогнозувати наслідки, але й активно впливати на стратегії відновлення після аварій. Після того, як моделювання дасть змогу оцінити можливі наслідки катастрофи, підприємство може розробити не лише інструкції з ліквідації наслідків аварії, а й створити більш ефективні системи для швидкого відновлення. Це можуть бути резервні системи живлення, автоматизовані лінії для обробки даних або засоби для швидкого надання медичної допомоги.

Важливо зазначити, що застосування об'єктів моделювання для оцінки надзвичайних ситуацій та їхніх наслідків має велике значення для не тільки для підприємства, але й для місцевої громади. Віртуальне моделювання дає можливість оцінити, як аварії можуть вплинути на навколишнє середовище та навколишні території, що дозволяє підприємству здійснювати превентивні заходи щодо мінімізації можливих збитків. Це підвищує рівень соціальної відповідальності підприємства і допомагає уникнути масштабних наслідків для навколишнього середовища та людей.

Таким чином, об'єкти моделювання є потужним інструментом для оптимізації заходів охорони праці під час надзвичайних ситуацій. Вони дозволяють підприємствам забезпечити надійний захист здоров'я працівників, оперативно реагувати на виникнення НС, а також мінімізувати наслідки катастроф для виробництва та навколишнього середовища. Завдяки використанню сучасних технологій моделювання, підприємства можуть значно підвищити рівень безпеки, знижуючи ймовірність аварій і покращуючи готовність до надзвичайних ситуацій.

3.2 Інтеграція об'єктів моделювання в системи оповіщення та евакуації під час надзвичайних ситуацій

Інтеграція об'єктів моделювання в системи оповіщення та евакуації є важливим елементом управління безпекою праці на підприємствах під час надзвичайних ситуацій (НС). Оскільки своєчасне оповіщення та організована евакуація є критично важливими для збереження життя та здоров'я працівників, інтеграція цих систем з моделями надзвичайних ситуацій дозволяє підприємствам ефективно реагувати на загрози і мінімізувати негативні наслідки аварій.

Основним завданням є розробка і тестування сценаріїв, які дозволяють оптимізувати процеси евакуації в разі виникнення НС. Для цього необхідно створювати віртуальні моделі будівель та виробничих приміщень, враховуючи розташування виходів, маршрутів евакуації, наявність перешкод, а також місця концентрації людей. Застосування таких моделей дає змогу аналізувати і прогнозувати різні варіанти розвитку подій, забезпечуючи ефективне планування евакуаційних заходів у реальному часі.

Одним з ключових аспектів використання моделей для оптимізації евакуаційних маршрутів є можливість тестування різних варіантів розвитку подій. У реальних умовах часто важко передбачити, як саме буде змінюватися ситуація під час надзвичайної події: чи будуть заблоковані виходи, чи виявляться обрані маршрути занадто довгими або небезпечними. Моделювання дає змогу вивчити різні сценарії таких подій, враховуючи всі можливі фактори, що можуть вплинути на хід евакуації. Наприклад, можна змодельовати ситуацію, коли одна частина будівлі закрита через пожежу, або коли на шляху евакуації виникають перешкоди, такі як обрушення конструкцій або витік токсичних газів. У такому випадку система моделювання може швидко запропонувати альтернативні варіанти евакуаційних маршрутів, що дозволяє зберегти час і забезпечити безпечне виведення всіх працівників.

Завдяки використанню об'єктів моделювання можна також проводити

оцінку ефективності систем оповіщення, що в разі НС повинні негайно інформувати працівників про загрозу. Моделюючи різні сценарії надзвичайних ситуацій, можна протестувати різні способи оповіщення: через сирени, звукові сигнали, телевізійні або радіо екрани, а також сучасні мобільні додатки, що використовуються для інформування персоналу. Важливо, щоб система оповіщення була максимально швидкою і чіткою, щоб не втрачати часу на непотрібні затримки, що можуть призвести до серйозних наслідків. Моделювання дозволяє перевірити, наскільки оперативно і ефективно працює система оповіщення, чи враховує вона всі можливі варіанти розвитку подій, а також як швидко кожен працівник отримує важливу інформацію про необхідність евакуації.

Для забезпечення високої ефективності систем оповіщення та евакуації важливо також враховувати фактори, які можуть вплинути на швидкість реагування, такі як кількість працівників на різних ділянках підприємства, рівень їхньої підготовленості до НС та наявність засобів індивідуального захисту. Моделювання дає змогу не лише проаналізувати ці фактори, а й передбачити, які дії слід вжити для забезпечення максимальної швидкості евакуації. Наприклад, в моделях можна врахувати, скільки часу займе евакуація людей із специфічних виробничих зон, чи є в наявності достатня кількість медичних працівників для надання допомоги, а також яку кількість транспорту потрібно для вивезення постраждалих.

Одним з важливих аспектів, який можна врахувати під час інтеграції об'єктів моделювання в систему евакуації, є оптимізація ресурсів, які використовуються під час НС. Наприклад, моделювання дозволяє прогнозувати необхідність у лікарських засобах, кількості аварійних машин, запасах води або медикаментів для надання першої допомоги. Такий підхід дозволяє підприємствам значно знизити витрати на організацію евакуації, розподіляючи ресурси найбільш ефективно, що у свою чергу підвищує рівень безпеки та ефективності заходів у надзвичайних ситуаціях.

Моделювання евакуації та оповіщення також допомагає в підготовці

персоналу до надзвичайних ситуацій. Віртуальні тренінги та симуляції на основі моделей дозволяють проводити навчання персоналу в умовах, наближених до реальних. Працівники можуть пройти через різні сценарії НС, отримуючи практичний досвід і навички для швидкої реакції. Це дозволяє підвищити рівень готовності всього колективу до критичних ситуацій і значно зменшити час на підготовку у разі реальної загрози.

Крім того, моделювання евакуаційних процесів дозволяє врахувати специфічні особливості різних категорій працівників: людей з обмеженими можливостями, дітей, похилих працівників чи тих, хто перебуває на лікуванні. Моделюючи такі сценарії, можна розробити додаткові маршрути та заходи для осіб, які потребують особливого догляду, що дозволяє забезпечити їхню безпеку в екстремальних умовах.

Таким чином, інтеграція об'єктів моделювання в системи оповіщення та евакуації є надзвичайно важливою для підприємств, які прагнуть забезпечити безпеку працівників під час надзвичайних ситуацій. Використання таких систем дає змогу оптимізувати евакуаційні маршрути, ефективно розподіляти ресурси, швидко реагувати на НС та покращувати загальну підготовленість підприємства до кризових ситуацій. Це дозволяє значно підвищити ефективність заходів охорони праці та забезпечити максимальний рівень безпеки працівників.

3.3 Використання моделювання для прогнозування наслідків техногенних катастроф та розробки стратегій реагування

Використання об'єктів моделювання для прогнозування наслідків техногенних катастроф є важливим елементом системи управління безпекою на підприємствах. У разі виникнення аварій чи катастроф таких як вибухи, пожежі, витоки небезпечних хімічних речовин чи інші техногенні загрози, моделювання дає змогу заздалегідь прогнозувати можливі наслідки цих подій, оцінювати масштаб збитків та розробляти ефективні стратегії для мінімізації

шкоди. Застосування об'єктів моделювання допомагає не лише виявляти потенційні небезпеки, а й створювати адаптивні плани дій, які враховують різні сценарії розвитку подій.

Моделювання техногенних катастроф дозволяє підприємствам оцінювати всі можливі ризики та небезпеки, пов'язані з виробничими процесами та обладнанням. Однією з основних переваг використання таких моделей є можливість моделювати різні типи катастроф, що можуть відбутися в реальних умовах. Наприклад, при виникненні пожежі на виробничому підприємстві, моделювання дозволяє проаналізувати, як швидко буде поширюватися вогонь, які зони підприємства будуть найбільше пошкоджені, і скільки часу буде потрібно для евакуації працівників. Завдяки такому аналізу можна знизити ймовірність виникнення серйозних наслідків і розробити чіткі інструкції для швидкого реагування.

Прогнозування наслідків надзвичайних ситуацій. Одним з головних завдань моделювання є прогнозування наслідків різних типів техногенних катастроф. Наприклад, у разі витоку токсичних чи вибухових речовин моделювання дозволяє визначити зону ураження, яка може бути обширною в залежності від кількості і характеру речовини, а також погодних умов. Створення віртуальної моделі підприємства дозволяє оцінити, як ці фактори будуть взаємодіяти між собою і впливати на хід подій. Моделюючи різні сценарії розвитку подій, можна передбачити, як швидко буде поширюватися небезпечна хімічна речовина, яку кількість працівників може опинитися під загрозою, і де виникне найбільше пошкоджень в інфраструктурі.

Моделювання дає змогу не лише прогнозувати матеріальні збитки, але й оцінювати ймовірність виникнення вторинних катастроф, таких як вибухи, пожежі чи задимлення, що можуть мати набагато серйозніші наслідки. Врахування всіх цих чинників дозволяє розробити стратегічні заходи, спрямовані на мінімізацію ймовірних втрат і швидке реагування на НС.

Розробка стратегій реагування. Для ефективного реагування на техногенні катастрофи, об'єкти моделювання дозволяють створювати адаптивні

стратегії. Зазвичай ці стратегії базуються на ретельному аналізі можливих сценаріїв розвитку подій. Моделювання дає змогу тестувати різні варіанти дій і оцінювати, які з них будуть найефективнішими в разі реальної загрози.

Одним з основних елементів стратегій реагування є визначення оптимальних ресурсів для боротьби з наслідками НС. За допомогою моделювання можна оцінити, яку кількість рятувальних команд і техніки потрібно залучити, скільки часу займе їх прибуття на місце події, і де слід зосередити зусилля. Моделюючи різні сценарії, можна також розрахувати, скільки часу знадобиться для надання медичної допомоги постраждалим, та чи достатньо буде наявних ресурсів, щоб справитись із надзвичайною ситуацією.

Окремо варто зазначити важливість розробки та тестування евакуаційних планів. Моделювання дає можливість оцінити, як працюють маршрути евакуації, чи враховуються всі можливі перешкоди, і чи зможуть працівники безпечно покинути небезпечну зону. Розроблені моделі евакуації можуть бути перевірені в умовах, наближених до реальних, що дозволяє максимально оптимізувати час евакуації і уникнути помилок у реальних умовах.

Зниження ризиків та вдосконалення систем безпеки. Однією з головних цілей моделювання є зниження ризиків для персоналу та навколишнього середовища. Під час проведення моделювань можна виявити слабкі місця в системах безпеки, що існують на підприємстві, і на основі отриманих результатів розробити заходи для їх усунення. Наприклад, якщо моделювання вказує на недостатньо ефективні засоби пожежогашіння чи систему вентиляції, підприємство може своєчасно вжити заходів для модернізації цих систем, що значно знижує ймовірність виникнення надзвичайної ситуації.

Моделювання також дозволяє постійно вдосконалювати стратегії управління ризиками. Оскільки техногенні катастрофи можуть бути спричинені різними факторами — від технічних поломок до людських помилок — моделювання дає змогу вивчати ці фактори та їх взаємодію. Завдяки цьому можна розробляти адаптивні заходи, які будуть ефективними навіть у нестандартних ситуаціях, що можуть виникнути під час аварії.

Роль моделювання в підготовці персоналу. Для ефективного реагування на техногенні катастрофи важлива також підготовка персоналу. За допомогою моделювання можна проводити навчання співробітників підприємства в умовах, максимально наближених до реальних. Віртуальні тренінги дозволяють працівникам здобути практичні навички, не піддаючи їх ризику в реальних умовах.

Моделювання дає змогу провести тренування, які охоплюють не тільки технічні аспекти — такі як дії під час пожежі чи аварії, — але й психологічні аспекти, наприклад, як реагувати в умовах стресу, як здійснювати евакуацію в екстремальних ситуаціях. Це дозволяє значно підвищити рівень підготовленості персоналу до реальних загроз.

Отже, моделювання є ключовим інструментом для підвищення ефективності системи охорони праці на підприємствах, особливо в умовах надзвичайних ситуацій. Використання різноманітних моделей дозволяє значно покращити готовність підприємств до можливих кризових ситуацій, забезпечити належну безпеку працівників та ефективно планувати стратегії реагування на техногенні катастрофи. Прогнозування, оцінка ризиків та розробка адаптивних планів дій є основними аспектами, що дозволяють знизити ймовірність виникнення нещасних випадків, а також оперативно реагувати на загрози.

Завдяки використанню статичних, динамічних та імітаційних моделей, підприємства можуть точніше прогнозувати можливі наслідки надзвичайних ситуацій та ефективно управляти ресурсами для забезпечення безпеки. Процес моделювання включає детальний аналіз ситуації, розробку сценаріїв та тестування різних варіантів реагування, що дозволяє максимально мінімізувати наслідки для працівників та виробничих процесів.

Незважаючи на численні переваги, використання моделювання в охороні праці має й певні недоліки, такі як потреба в кваліфікованих спеціалістах та високих фінансових витратах на необхідне обладнання та програмне забезпечення. Однак, з огляду на важливість зниження ризиків і забезпечення

безпеки на підприємствах, ці витрати є виправданими.

Враховуючи швидкий розвиток технологій і зростаючі вимоги до безпеки, моделювання стає важливим інструментом для адаптації підприємств до нових умов і викликів, зокрема в умовах надзвичайних ситуацій. Успішне впровадження систем моделювання забезпечує не тільки ефективну охорону праці, але й підвищує загальну стабільність і стійкість підприємства перед різноманітними кризами.

Висновки до розділу 3

Використання моделювання для прогнозування та аналізу охорони праці під час надзвичайних ситуацій є важливим інструментом для зниження ризиків та покращення безпеки на підприємствах. Це дозволяє не лише оцінити ймовірність виникнення небезпечних подій, але й розробити ефективні плани дій, що значно підвищує готовність підприємства до потенційних кризових ситуацій. Моделювання допомагає виявити слабкі місця у виробничих процесах, оцінити потенційні загрози та розробити стратегії для їх мінімізації.

Однією з головних переваг моделювання є можливість прогнозування та оцінки ризиків, що дозволяє підприємствам заздалегідь підготуватися до можливих загроз. Це дає змогу не лише мінімізувати наслідки аварій, але й зберегти здоров'я працівників, адже з допомогою моделювання можна визначити найбільш вразливі точки на виробництві, виявити потенційно небезпечні зони та розробити заходи для їх ліквідації. Тестування різних сценаріїв та перевірка ефективності існуючих планів дій є важливим етапом, що забезпечує більш точне реагування на реальні надзвичайні ситуації.

Типи моделей, які використовуються для охорони праці, охоплюють різноманітні аспекти, починаючи від простих статичних моделей для оцінки безпеки робочих місць до складних динамічних та імітаційних моделей, що дають змогу відтворювати реальні сценарії аварій і катастроф. Імітаційне моделювання, зокрема, дозволяє проводити тренування без реальних ризиків

для персоналу, що значно підвищує ефективність підготовки. Крім того, моделювання ризиків дозволяє підприємствам здійснювати грамотне управління ресурсами і правильно інвестувати в заходи безпеки, що забезпечує економічну ефективність підприємства.

Проте, як і будь-який інший метод, моделювання має свої недоліки. Перш за все, це високі витрати на створення та впровадження таких моделей, а також необхідність залучення висококваліфікованих спеціалістів для їх розробки і підтримки. Крім того, деякі моделі можуть не бути повністю точними, особливо коли мова йде про нові або маловивчені надзвичайні ситуації, що може знижувати їх ефективність. Однак ці недоліки можна мінімізувати за допомогою регулярного оновлення моделей і залучення фахівців, які мають досвід у роботі з кризовими ситуаціями.

В цілому, моделювання є потужним інструментом для підприємств, що прагнуть покращити свою готовність до надзвичайних ситуацій. Вони не лише дозволяють своєчасно реагувати на кризові ситуації, але й значно знижують ймовірність виникнення нещасних випадків. Використання таких моделей дозволяє підприємствам не тільки забезпечити безпеку працівників, але й оптимізувати свої ресурси та поліпшити взаємодію з підрозділами та службами екстреного реагування. Враховуючи швидкий розвиток технологій та збільшення складності виробничих процесів, моделювання стане ще більш важливим інструментом для підприємств у майбутньому, що дозволить не лише зберегти життя та здоров'я працівників, але й підвищити ефективність виробничих процесів в умовах кризових ситуацій..

4 МОДЕЛЮВАННЯ ТА ТЕСТУВАННЯ СИСТЕМИ

4.1 Реалізація та тестування веб-додатка

Реалізація веб-додатка Реалізація веб-додатка для керування віддаленими об'єктами починається з розробки основних функціональних блоків, таких як користувацький інтерфейс, серверна частина, база даних і алгоритми машинного навчання. Ключовими етапами реалізації є:

Створення фронтенду (інтерфейсу користувача): Клієнтська частина веб-додатка є важливою для забезпечення зручності використання та ефективної взаємодії з користувачем. Використовуються сучасні технології, такі як React.js або Vue.js, для розробки динамічних інтерфейсів. Графіки, таблиці та кнопки для керування пристроями повинні бути зручними і зрозумілими. Спеціальні елементи управління, такі як слайдери для налаштувань температури чи статусу пристрою, повинні бути інтерактивними та відображати актуальні зміни в реальному часі.

Розробка бекенду (серверної частини): Серверна частина відповідає за обробку запитів від клієнтської частини, зберігання і обробку даних, а також взаємодію з базою даних і компонентами машинного навчання. Для реалізації серверної частини можуть бути використані Node.js, Django або Flask, які дозволяють швидко обробляти запити і здійснювати необхідні операції над даними.

Інтеграція з базою даних: Розробка структури бази даних є важливим кроком. Вона повинна містити таблиці для користувачів, пристроїв, показників і подій. Структуровані дані повинні бути організовані таким чином, щоб забезпечити швидкий доступ до них. Використовуються такі бази даних, як PostgreSQL або MongoDB, для збереження даних у різних форматах (структурованих та неструктурованих).

Інтеграція з алгоритмами машинного навчання: Один з найбільш важливих аспектів цієї системи – це автоматизація управління віддаленими

об'єктами. Алгоритми машинного навчання використовуються для прогнозування і оптимізації роботи пристроїв. Наприклад, нейронні мережі можуть передбачити зміни в стані об'єкта, а алгоритми виявлення аномалій допомагають визначати відхилення від нормальних параметрів роботи. Використовуються такі бібліотеки, як TensorFlow або Scikit-learn. Тестування веб-додатка Тестування веб-додатка є важливим етапом для забезпечення високої якості і стабільності системи. Воно включає кілька основних етапів:

Функціональне тестування: Функціональне тестування забезпечує перевірку правильності виконання всіх функцій додатка згідно з вимогами. Перевіряється робота основних модулів: відображення даних на фронтенді, правильність збереження даних у базі, коректність реагування на введення користувача, а також автоматичне керування об'єктами через систему.

Навантажувальне тестування: Навантажувальне тестування проводиться для оцінки здатності системи працювати з великими обсягами даних і великою кількістю одночасних користувачів. Тестуються моменти, коли одночасно підключено кілька користувачів, а також коли система має обробляти великі обсяги вимірів та подій від віддалених об'єктів.

Тестування безпеки: Тестування безпеки необхідне для того, щоб переконатися в тому, що система захищена від можливих атак. Перевіряється наявність механізмів автентифікації та авторизації, а також шифрування даних, щоб запобігти несанкціонованому доступу.

Тестування на зручність користування (Usability testing): Оцінюється зручність інтерфейсу для користувача, перевіряється, наскільки інтуїтивно зрозумілим є управління додатком. Важливо, щоб користувач міг швидко налаштувати або керувати віддаленими об'єктами без необхідності в спеціалізованих знаннях.

Тестування на відповідність вимогам: Перевіряється, наскільки система відповідає вимогам, визначеним на етапі проектування. Тестуються всі функціональні та нефункціональні вимоги, а також чи виконуються вони згідно з документованими параметрами.

4.2 Оцінка ефективності та практичне впровадження

Оцінка ефективності та практичне впровадження системи є одним із найважливіших етапів розробки, адже саме на цьому етапі перевіряється відповідність функціональних і нефункціональних характеристик системи початковим вимогам. Водночас впровадження забезпечує реальне інтегрування розробленої системи у робоче середовище, підготовку кінцевих користувачів та підтримку стабільної роботи у процесі експлуатації.

Оцінка ефективності передбачає всебічний аналіз ключових характеристик системи, таких як продуктивність, надійність, масштабованість та точність роботи алгоритмів машинного навчання. Цей етап є критично важливим, оскільки він дозволяє виявити слабкі місця в архітектурі системи, алгоритмах чи процесах обробки даних, а також знайти способи їх оптимізації. Крім того, тестування проводиться в умовах, максимально наближених до реальних, щоб впевнитися у стійкості системи під навантаженням і її здатності виконувати всі заплановані функції.

Одним із перших аспектів, що оцінюються, є продуктивність системи. У випадку систем реального часу, де користувачі повинні отримувати оновлені дані без затримок, це має особливе значення. Продуктивність системи визначається швидкістю обробки запитів, часом відгуку серверів та оновленням даних у клієнтському інтерфейсі. Наприклад, під час тестування системи аналізуються такі параметри, як середній і максимальний час відгуку API, тривалість обробки запитів до бази даних та швидкість оновлення дашбордів. Якщо система не досягає бажаних показників, можливе впровадження додаткових оптимізацій, таких як кешування даних, розподіл запитів між кількома серверами або вдосконалення структури бази даних.

Другим важливим критерієм оцінки є надійність системи. Надійність визначає здатність системи працювати безперебійно, стійко реагувати на збої та відновлюватися після них без втрати даних чи зупинки основних функцій. Для цього проводяться тести, які імітують різні сценарії, зокрема відключення

мережі, збій роботи обладнання або перевантаження серверів. Система має автоматично зберігати дані перед аварією, а після відновлення підключення або усунення проблеми – повертатися до стабільної роботи. Важливим аспектом є також розподілена архітектура, яка дозволяє ізолювати збої в одній частині системи від інших компонентів.

Масштабованість системи – ще один критично важливий аспект. У сучасних умовах швидке зростання кількості підключених пристроїв або обсягів даних може значно вплинути на продуктивність системи, якщо вона не має можливості адаптуватися до таких змін. Масштабованість оцінюється через стрес-тести, що симулюють значне збільшення кількості запитів або підключень нових пристроїв. Під час таких тестів аналізується, як змінюється час відгуку та продуктивність системи. Якщо виявляються проблеми, можуть бути запропоновані рішення, такі як вертикальне масштабування (збільшення ресурсів серверів) або горизонтальне масштабування (додавання нових серверів до системи).

Особливої уваги вимагає точність алгоритмів машинного навчання, які є основою для автоматизації управління віддаленими об'єктами. Оцінка точності алгоритмів включає порівняння прогнозів із реальними даними, а також аналіз можливих відхилень. Наприклад, система, яка прогнозує температуру обладнання або вологість повітря, повинна забезпечувати мінімальні похибки у своїх передбаченнях. Для підвищення точності можуть бути проведені додаткові тренування моделей машинного навчання з використанням більшого обсягу даних або корекція алгоритмів на основі аналізу помилок.

Практичне впровадження системи починається з її розгортання та налаштування. Система встановлюється на сервери або хмарні платформи, які забезпечують її стабільну роботу та доступність для користувачів. На цьому етапі налагоджуються зв'язки між компонентами системи, такими як база даних, клієнтська частина та інтерфейс із пристроями. Забезпечується інтеграція із зовнішніми API, що можуть бути необхідними для роботи системи, а також впроваджуються механізми безпечної передачі даних,

наприклад шифрування інформації чи використання захищених протоколів зв'язку.

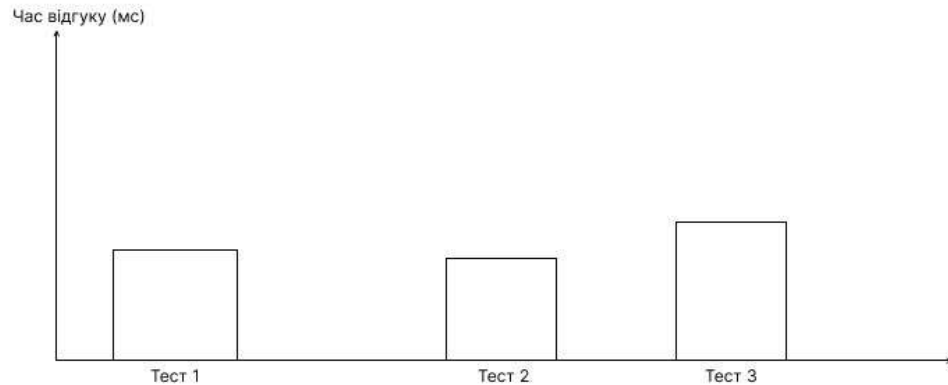


Рисунок 4.1 – Оцінка продуктивності системи

Наступним кроком є навчання користувачів системи. Це важливий процес, що охоплює різні категорії персоналу, які взаємодіятимуть із системою. Адміністратори отримують інструкції щодо налаштування, обслуговування та оновлення системи, включаючи детальний опис її архітектури та механізмів роботи. Кінцеві користувачі, такі як оператори чи технічний персонал, проходять навчання щодо використання інтерфейсу системи, налаштування параметрів управління віддаленими об'єктами, реагування на аварійні ситуації та аналізу отриманих даних. Цей етап також включає створення документації, що описує роботу системи, її функціональні можливості та інструкції для вирішення типових проблем.

Після завершення встановлення і навчання розпочинається етап моніторингу та підтримки системи. Постійний моніторинг дозволяє відстежувати продуктивність, виявляти можливі помилки та своєчасно реагувати на них. Це включає регулярне оновлення програмного забезпечення для забезпечення безпеки та додавання нових функцій, а також виконання

профілактичних робіт, таких як оптимізація бази даних або перевірка цілісності системних файлів. Крім того, на цьому етапі забезпечується технічна підтримка користувачів, що включає відповіді на їхні запити, допомогу в налаштуванні системи або вирішення технічних проблем.

Важливим аспектом впровадження є тестування системи в реальних умовах. Це дає змогу перевірити, як система функціонує у виробничому середовищі, де вона взаємодіє з реальними пристроями та користувачами. На основі результатів такого тестування можна вносити додаткові зміни або вдосконалення, щоб адаптувати систему до специфічних потреб організації.



Рисунок 4.2 – Схема впровадження системи

Таким чином, оцінка ефективності та практичне впровадження є завершальними та надзвичайно важливими етапами у розробці системи. Вони забезпечують не лише перевірку всіх характеристик системи, але й її підготовку до реального використання, включаючи налаштування, інтеграцію, навчання користувачів та подальшу підтримку. Ретельне виконання цих етапів гарантує, що система працюватиме стабільно, ефективно виконуватиме свої функції та відповідатиме потребам користувачів у реальних умовах експлуатації.

4.3. Важливість процесу моделювання та тестування перед повноцінним впровадженням у роботу веб-додатку

Сучасні веб-додатки є складними інформаційними системами, що виконують безліч функцій для кінцевих користувачів. Успіх веб-додатку багато в чому залежить від того, наскільки ефективно він був розроблений і наскільки якісно його протестували перед запуском. Моделювання та тестування є важливими етапами в розробці веб-додатку, оскільки вони дозволяють виявити помилки, покращити функціональність і забезпечити стабільну роботу системи.

В процесі роботи була розглянута важливість процесів моделювання та тестування перед повноцінним впровадженням веб-додатку в роботу, а також їх роль у забезпеченні безпеки, зручності та ефективності використання продукту.

Моделювання веб-додатку. Моделювання є процесом створення абстрактних представлень веб-додатку для кращого розуміння його структури та поведінки. Цей етап дозволяє розробникам уявити, як система буде працювати, навіть до того, як буде написано значну частину коду. Моделювання дає змогу передбачити різні ситуації, що можуть виникнути під час експлуатації, і допомагає виявити можливі проблеми на ранніх етапах.

Проектування архітектури. Однією з основних задач моделювання є створення архітектурної моделі веб-додатку. Це дозволяє спроектувати основні компоненти системи, їх взаємодії, а також зрозуміти, як буде здійснюватися обробка запитів, взаємодія з базою даних і відображення результатів на клієнтському інтерфейсі. Правильне проектування архітектури гарантує, що система буде масштабованою, ефективною та зручною для подальших змін.

Визначення вимог та функціональності. Моделювання дозволяє чітко визначити вимоги до функціональності веб-додатку. Це допомагає розробникам зрозуміти, які функції повинні бути реалізовані, як вони повинні працювати і як будуть взаємодіяти з іншими частинами системи. Наприклад, моделювання може допомогти визначити, як користувачі будуть реєструватися в системі, як вони будуть взаємодіяти з даними, або як додаток повинен реагувати на

специфічні дії користувача.

Створення прототипу інтерфейсу користувача. Моделювання дозволяє також створити прототипи інтерфейсу користувача (UI/UX), що є важливою частиною користувацького досвіду. На цьому етапі визначається, як саме буде виглядати додаток і як користувачі будуть взаємодіяти з інтерфейсами. Створення прототипу допомагає передбачити можливі труднощі та недоліки у дизайні, що дозволяє значно покращити досвід користувачів на пізніших етапах.

Оцінка безпеки. Моделювання також включає розробку плану безпеки веб-додатку. Це дозволяє виявити потенційні вразливості та спроектувати стратегії їх усунення. Безпека веб-додатків є критично важливою, оскільки злоумисники можуть використовувати різні техніки для проникнення в систему або викрадення даних. Задача моделювання полягає в тому, щоб визначити рівень захисту даних, а також гарантувати, що система буде здатна захиститися від різноманітних атак, таких як SQL-ін'єкції, XSS-атаки та інші.

Тестування веб-додатку. Тестування — це процес перевірки, чи працює веб-додаток відповідно до вимог і стандартів якості. Тестування дає можливість виявити дефекти та помилки в коді, а також оцінити різні аспекти роботи системи, такі як швидкість, стабільність і безпека. Цей етап необхідний для того, щоб гарантувати високу якість кінцевого продукту.

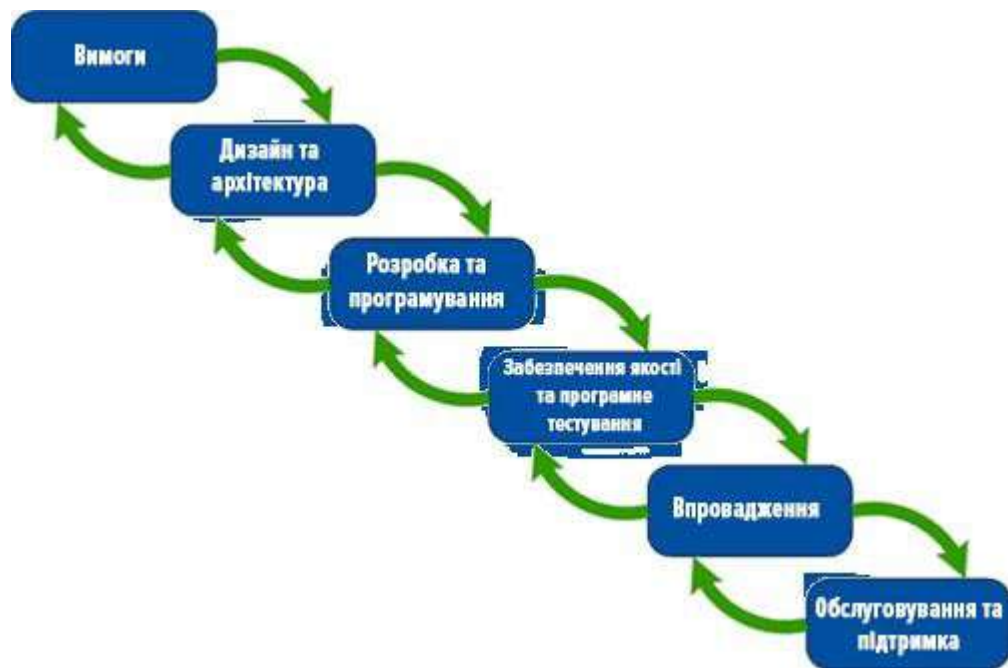


Рисунок 4.3 – Процес тестування веб-додатків

Функціональне тестування. Функціональне тестування спрямоване на перевірку того, чи виконуються всі функції веб-додатку відповідно до специфікацій. Це включає перевірку правильності обробки запитів, функціональність пошукових систем, авторизацію користувачів, а також правильність відображення результатів для користувачів. Тестування функціональності є важливим етапом для того, щоб переконатися, що веб-додаток виконує всі свої основні функції без помилок.

Тестування сумісності. Тестування сумісності дає змогу перевірити, як веб-додаток працює на різних браузерях, операційних системах та пристроях. Веб-додатки повинні забезпечувати стабільну роботу на всіх основних платформах, зокрема в різних браузерах (Chrome, Firefox, Safari, Edge) та на мобільних пристроях. Виявлення проблем із сумісністю дозволяє уникнути ситуацій, коли додаток працює тільки на певних пристроях або браузерах, що може призвести до втрати користувачів.

Тестування продуктивності. Перевірка продуктивності веб-додатку є важливою частиною тестування. Це дозволяє оцінити, як додаток поводить себе під навантаженням. Тестування на навантаження (наприклад, за допомогою стрес-тестування) дає змогу визначити, чи може система витримати велику

кількість одночасних користувачів і чи буде вона працювати швидко навіть у пікові моменти. Повільна робота веб-додатку або його збої в умовах високого навантаження можуть призвести до негативного досвіду для користувачів.

Безпека додатку. Веб-додатки часто є мішенню для атак, тому тестування безпеки є обов'язковим етапом. Тестування безпеки допомагає виявити вразливості додатку, такі як можливість проникнення через форми або інші елементи введення, неправильну обробку сесій або недостатньо захищені паролі. Забезпечення безпеки додатку є критично важливим для запобігання витоку конфіденційних даних користувачів і захисту від кібератак.

Юзабіліті тестування. Юзабіліті тестування спрямоване на перевірку зручності використання веб-додатку. Це дозволяє оцінити інтерфейс, навігацію та зручність взаємодії користувача з системою. Забезпечення простоти та зрозумілості інтерфейсу є важливим аспектом, оскільки користувачі не будуть використовувати додаток, якщо він буде складним або незручним. Юзабіліті тестування дозволяє виправити недоліки інтерфейсу і зробити його більш інтуїтивно зрозумілим.

Переваги моделювання та тестування. Моделювання та тестування веб-додатку забезпечують низку важливих переваг, таких як:

Запобігання серйозним помилкам. Ранні етапи моделювання і тестування дозволяють виявити дефекти до того, як вони стануть критичними і змусять команду вносити зміни вже в готовий продукт.

Покращення якості продукту. Підвищення стабільності, безпеки та продуктивності додатку, завдяки виявленню та усуненню помилок і проблем на етапах моделювання та тестування.

Економія часу та ресурсів. Виправлення помилок на етапах тестування значно дешевше і менш трудозатратно, ніж після впровадження додатку в роботу.

Покращення користувацького досвіду. Завдяки моделюванню та тестуванню можна забезпечити зручний інтерфейс, стабільну роботу і високий рівень безпеки, що підвищує задоволеність користувачів.

Моделювання та тестування є необхідними етапами в процесі розробки веб-додатку. Ці процеси дозволяють значно знизити ризики виникнення помилок і дефектів, покращити функціональність та безпеку, а також забезпечити високий рівень задоволеності користувачів. Підготовка веб-додатку до повноцінного впровадження через ретельне тестування і моделювання допомагає забезпечити ефективність і стабільність продукту, що має велике значення для успіху на ринку.

4.4. Веб-додаток для керування віддаленими об'єктами з використанням технологій машинного навчання

Розглядається процес розробки простого веб-додатку для віддаленого керування, який дозволяє користувачеві надсилати команди на сервер і отримувати відповідь у реальному часі через веб-інтерфейс. Основними технологіями для розробки такого додатку є Node.js для серверної частини, Socket.IO для реального часу і HTML/CSS для створення користувацького інтерфейсу.

Основи створення веб-додатку для віддаленого керування. Розробка веб-додатку для віддаленого керування передбачає необхідність забезпечення стабільної та безпечної взаємодії між клієнтом і сервером. Ключовими компонентами є:

Серверна частина — це компонент, що виконує обробку запитів, взаємодіє з базами даних або виконує команди на сервері.

Клієнтська частина — інтерфейс, з яким користувач взаємодіє, щоб надсилати команди та отримувати відгуки.

Реальний час — важливою частиною таких додатків є забезпечення взаємодії в реальному часі, що дозволяє миттєво передавати дані між клієнтом і сервером.

Веб-додатки для віддаленого керування використовують різні технології для створення інтерактивного інтерфейсу та надійного з'єднання між

користувачем і сервером. У цьому рефераті буде розглянута конкретна реалізація, де серверна частина написана на Node.js, а для взаємодії в реальному часі використано Socket.IO.

Серверна частина: Node.js та Socket.IO

Node.js — це середовище виконання JavaScript на сервері, яке активно використовується для створення високопродуктивних веб-додатків. Для створення сервера віддаленого керування використовуються наступні основні технології:

Express — популярний фреймворк для створення серверів на Node.js.

Socket.IO — бібліотека для створення двостороннього з'єднання між клієнтом і сервером в реальному часі.

Приклад коду серверної частини наведено в Додатку А.

Клієнтська частина: HTML, CSS та JavaScript

Клієнтська частина веб-додатку забезпечує інтерфейс для взаємодії з сервером. Користувач може вводити команду в текстове поле і натискати кнопку для її відправки на сервер. Відповідь сервера буде відображена на веб-сторінці.

Приклад коду клієнтської частини наведено в Додатку Б.

Переваги та обмеження веб-додатків для віддаленого керування. Переваги:

Зручність — користувач може керувати системою з будь-якого місця, де є доступ до Інтернету.

Реальний час — можливість миттєвої взаємодії з сервером через WebSocket.

Масштабованість — додатки можна масштабувати для роботи з кількома користувачами або великою кількістю серверів.

Обмеження:

Безпека — необхідно забезпечити захист від несанкціонованого доступу, адже зловмисники можуть використовувати такі системи для атак.

Налаштування та адміністрування — для використання таких систем може знадобитися спеціальна підготовка або адмінські права.

Залежність від Інтернету — система потребує стабільного з'єднання з Інтернетом для функціонування.

Веб-додатки для віддаленого керування дозволяють здійснювати моніторинг і контроль за різними системами через Інтернет. Використання технологій, таких як Node.js, Socket.IO і WebSockets, робить ці додатки потужними інструментами для віддаленого адміністрування. Однак важливо пам'ятати про безпеку і захист від несанкціонованого доступу, оскільки віддалене керування може стати мішенню для різноманітних атак.

Висновки до розділу 4

Реалізація веб-додатка для керування віддаленими об'єктами є складним та багатоетапним процесом, що вимагає інтеграції різних технологій та компонентів. Це включає розробку фронтенду, бекенду, бази даних і алгоритмів машинного навчання, кожен з яких має свою роль у забезпеченні функціональності та ефективності системи. Ключовими етапами розробки є створення зручного та інтуїтивно зрозумілого інтерфейсу користувача, налаштування серверної частини для обробки запитів та взаємодії з базою даних, а також інтеграція з алгоритмами машинного навчання для автоматизації управління пристроями.

Фронтенд веб-додатка забезпечує ефективну взаємодію з користувачем. Використання таких технологій, як React.js або Vue.js, дозволяє створювати динамічні та інтерактивні інтерфейси, що робить управління пристроями зручним та швидким. Створення графіків, таблиць та спеціальних елементів управління, таких як слайдери для налаштування температури чи інших параметрів, допомагає користувачам швидко налаштовувати пристрої, спостерігаючи за реальними змінами.

Розробка серверної частини, яка відповідає за обробку запитів, зберігання та обробку даних, а також взаємодію з базою даних, є важливим етапом для забезпечення швидкості і ефективності роботи системи. Використання таких

фреймворків, як Node.js, Django або Flask, дозволяє створювати гнучкі та масштабовані сервери, здатні обробляти велику кількість одночасних запитів без зниження продуктивності.

Інтеграція з базою даних є необхідною для зберігання та організації даних користувачів, пристроїв та їхніх параметрів. Важливо правильно спроектувати структуру бази даних для забезпечення швидкого доступу до інформації. Використання таких баз даних, як PostgreSQL або MongoDB, дозволяє ефективно зберігати як структуровані, так і неструктуровані дані, що є необхідним для роботи з різними типами пристроїв та подій.

Інтеграція алгоритмів машинного навчання відіграє ключову роль в автоматизації управління віддаленими об'єктами. За допомогою нейронних мереж та алгоритмів виявлення аномалій можна прогнозувати зміни в стані об'єктів, що дозволяє оптимізувати роботу системи та запобігати можливим аваріям. Використання бібліотек, таких як TensorFlow або Scikit-learn, дає змогу ефективно реалізувати ці алгоритми, підвищуючи точність прогнозів та швидкість реагування на критичні зміни.

Тестування веб-додатка є важливим етапом для забезпечення його стабільності та безпеки. Функціональне тестування перевіряє, чи виконуються всі функції додатка згідно з вимогами, включаючи коректне відображення даних на інтерфейсі та автоматичне керування пристроями. Навантажувальне тестування дозволяє оцінити здатність системи обробляти великі обсяги даних та одночасні запити користувачів, що є критично важливим для систем, які працюють з віддаленими об'єктами в реальному часі. Тестування безпеки забезпечує захист системи від можливих атак, перевіряючи наявність механізмів автентифікації та шифрування даних. Тестування на зручність користування гарантує, що інтерфейс є інтуїтивно зрозумілим і зручним для користувачів з різним рівнем технічної підготовки. Тестування на відповідність вимогам дозволяє перевірити, чи реалізовані всі функціональні та нефункціональні вимоги.

Оцінка ефективності системи після тестування включає в себе кілька

важливих аспектів. Продуктивність системи повинна бути достатньою для забезпечення швидкої обробки запитів і безперебійного оновлення даних в реальному часі. Надійність системи має бути перевірена на стійкість до збоїв і здатність до автоматичного відновлення після них. Масштабованість дозволяє оцінити, наскільки система може працювати з більшими обсягами даних і додатковими пристроями без суттєвих втрат продуктивності. Точність прогнозів є важливим показником ефективності машинного навчання, і для цього потрібно порівняти прогнозовані значення з реальними показниками.

Практичне впровадження системи включає кілька етапів. Перш за все, необхідно встановити систему на сервери та налаштувати коректне підключення до віддалених об'єктів для забезпечення безпечної передачі даних. Навчання користувачів є важливим етапом для забезпечення ефективного використання системи. Це може включати як технічне навчання для адміністрації системи, так і інструкції для кінцевих користувачів, щоб вони могли ефективно керувати віддаленими об'єктами. Моніторинг і підтримка системи після впровадження є необхідними для забезпечення її стабільної роботи в майбутньому, включаючи оновлення програмного забезпечення та своєчасне виправлення помилок.

Отже, розробка та впровадження веб-додатка для керування віддаленими об'єктами є складним, але важливим процесом, що включає в себе етапи проектування, розробки, тестування та практичного впровадження. Важливими аспектами є забезпечення зручності для користувачів, безпеки системи, ефективного управління даними та інтеграція сучасних технологій машинного навчання для автоматизації процесів. Успішне впровадження такої системи здатне значно покращити ефективність управління віддаленими об'єктами, зменшити ризики та покращити продуктивність.

ВИСНОВКИ

Розробка веб-додатка для керування віддаленими об'єктами з використанням технологій машинного навчання стала важливим кроком у напрямку автоматизації управлінських процесів. Рішення, що передбачає інтеграцію сучасних технологій, таких як машинне навчання, у сферу віддаленого моніторингу та керування, має величезний потенціал для оптимізації роботи різноманітних систем. Впровадження таких технологій дозволяє значно знизити залежність від людського фактору, зменшити кількість помилок та підвищити точність прийняття рішень.

У рамках розробки системи було здійснено детальний аналіз існуючих підходів та рішень для керування віддаленими об'єктами, що дозволило визначити ключові переваги та недоліки традиційних методів. Це стало основою для створення більш ефективного, автоматизованого інструменту, який здатний не лише виконувати управлінські функції, а й прогнозувати можливі відхилення від нормального функціонування системи. Вибір технологій для створення веб-додатка, таких як React.js для фронтенду, Node.js для бекенду, а також застосування баз даних, таких як PostgreSQL чи MongoDB, дозволив побудувати потужну та гнучку архітектуру, яка забезпечує високу швидкість роботи та зручність в обслуговуванні. Завдяки інтеграції алгоритмів машинного навчання в систему, зокрема використанню методів аналізу великих даних і виявлення аномалій, система здатна виявляти проблеми ще до їх виникнення, прогножуючи зміни в параметрах об'єктів. Це дозволяє знижувати ризики від непередбачених ситуацій і забезпечувати більш ефективне управління.

Автоматизоване управління в реальному часі стало основним перевагою системи, оскільки дає можливість здійснювати контроль без необхідності постійного втручання з боку оператора, що також знижує витрати на персонал і підвищує ефективність використання ресурсів. Особливу увагу було приділено проектуванню інтерфейсу користувача, що стало важливим кроком для

забезпечення доступності системи для широкого кола користувачів.

Інтуїтивно зрозумілий інтерфейс дозволяє навіть не експертам працювати з додатком без додаткового навчання або підтримки, що робить систему зручною у використанні та легко адаптованою до різних умов експлуатації.

Проектування інтерфейсу було орієнтоване на максимальну простоту в управлінні, щоб користувач міг швидко реагувати на зміни стану об'єктів і здійснювати необхідні корективи. Одним з важливих етапів роботи стало тестування системи. Тестування включало перевірку на різних етапах розробки, починаючи від перевірки окремих компонентів та закінчуючи комплексним тестуванням готової системи. Основними критеріями, за якими проводилась оцінка, були продуктивність, надійність, безпека та відповідність системи заявленим вимогам.

Ретельне тестування дозволило виявити і усунути потенційні вразливості та помилки, що є критичними для стабільної роботи в реальних умовах. Зокрема, проводилось навантажувальне тестування, яке допомогло визначити межі працездатності системи при максимальних навантаженнях, а також тестування безпеки, яке підтвердило надійність системи щодо можливих атак. Завдяки ефективності реалізації та тестування, система показала хороші результати у вирішенні завдань автоматизації управління. Вона здатна обробляти великі обсяги даних у реальному часі, здійснювати прогнозування параметрів об'єктів та вчасно реагувати на відхилення від нормального функціонування.

Система може бути застосована в багатьох сферах діяльності, де необхідно здійснювати контроль та управління віддаленими об'єктами, зокрема у промисловості, енергетиці, сільському господарстві, а також в сфері розумних будинків і IoT. Ключовою перевагою є автоматизація процесів, що дозволяє знизити витрати на обслуговування та підвищити ефективність роботи.

Прогнозування змін параметрів об'єктів за допомогою машинного навчання дає можливість значно покращити процеси управління. Оскільки система здатна навчатися на основі історичних даних, вона може адаптуватися

до нових умов і вчасно реагувати на зміни, що забезпечує високий рівень надійності. Прогнозування також дозволяє приймати обґрунтовані рішення без втручання оператора, що є важливим у випадках, коли від швидкості реагування залежить ефективність або безпека. Оцінка ефективності системи показала, що розроблений веб-додаток відповідає сучасним вимогам і може стати основою для подальшого розвитку та вдосконалення. Впровадження такої системи дозволить значно оптимізувати процеси керування, забезпечити високий рівень безпеки та знизити витрати на технічне обслуговування. Зі збільшенням кількості даних та удосконаленням технологій машинного навчання система може стати ще більш потужною та масштабованою, що дозволяє зберігати конкурентні переваги.

Практичне впровадження розробленого веб-додатка дозволяє здійснити повноцінне управління віддаленими об'єктами без постійного втручання людини, що значно підвищує ефективність роботи та знижує ризик помилок. Водночас система забезпечує високий рівень безпеки, зокрема через механізми шифрування даних і контролю доступу, що є важливим аспектом для забезпечення надійної роботи в критичних умовах.

Загалом, розроблений веб-додаток для керування віддаленими об'єктами з використанням технологій машинного навчання є інноваційним і потужним інструментом для автоматизації управління та моніторингу. Впровадження такої системи дозволяє значно підвищити ефективність і надійність управління, що має велике значення для розвитку сучасних технологій в різних сферах діяльності.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Балабан І. В. “Системи моніторингу та керування віддаленими об’єктами: методи, технології та інструменти” // Вісник Національного університету “Львівська політехніка”. – 2020. – №3. – С. 45-50.
2. Cohn, M. “Agile Estimating and Planning”. – Pearson Education, 2019. – 330 p.
3. Freeman, E., Robson, E. “Head First Design Patterns: A Brain-Friendly Guide”. – O’Reilly Media, 2020. – 680 p.
4. Goodfellow, I., Bengio, Y., Courville, A. “Deep Learning”. – MIT Press, 2019. – 800 p.
5. Горобець П. І. “Використання RESTful API для веб-додатків: огляд технологій” // Інформаційні системи та технології. – 2020. – №3. – С. 25-30.
6. Kurniawan, B. “Full-Stack Development: Building Modern Web Applications”. – Apress, 2021. – 340 c.
7. Martin, R. C. “Clean Architecture: A Craftsman’s Guide to Software Structure and Design”. – Prentice Hall, 2017. – 432 p.
8. Назаренко І. І. “Розробка адаптивного інтерфейсу для веб-додатків” // Журнал комп’ютерних наук. – 2020. – №1. – С. 15-20.
9. Романюк В. С. “Масштабованість систем для моніторингу даних” // Технічні науки і технології. – 2021. – №7. – С. 34-40.
10. Сідоренко А. А., Колесник О. П. “Застосування технологій машинного навчання для прогнозування стану систем” // Технології інтелектуальних систем. – 2021. – №4. – С. 21-28.
11. Shukla, A. “Practical Machine Learning with Python”. – Packt Publishing, 2019. – 400 p.
12. Sedgewick, R., Wayne, K. “Algorithms”. – Addison-Wesley, 2019. – 960 p.
13. Ткаченко О. П., Романенко Д. В. “Тестування продуктивності та надійності веб-систем” // Технології розподілених систем. – 2018. – №6. – С. 60-

65.

14. Войтенко С. В. “Архітектура веб-застосунків: принципи побудови та оптимізації” // Комп’ютерні технології та програмування. – 2019. – №2. – С. 12-18.

15. Witten, I. H., Frank, E., Hall, M. A., Pal, C. J. “Data Mining: Practical Machine Learning Tools and Techniques”. – Morgan Kaufmann, 2020. – 654 p.

16. Халімов В. Г., Петрова Н. Л. “Проектування баз даних для веб-додатків” // Вісник інформаційних технологій. – 2018. – №5. – С. 30-35.

ДОДАТОК А. Охорона праці

1. Основні шкідливі виробничі фактори, що впливають на умови праці на персональному комп'ютері

До основних шкідливих та небезпечних факторів виробничого середовища, пов'язаних з роботою на персональному комп'ютері належать: напруга зорових органів та перевтомлення очей; навантаження на кисті рук та пальці; тривале знаходження в статичному стані, що викликає застійні явища в організмі; випромінювання різного виду (рентгенівське, електромагнітне, інфрачервоне, статистичні поля); механічні шуми, пов'язані з роботою кулера, дискового приводу, оргтехніки; іонізація повітря; виділення в повітря робочого приміщення різних хімічних речовин (озон, триметілфосфат, біфеніли); небезпека статичної електрики.

До психологічно шкідливих факторів можна віднести розумову напругу та нервово-емоційне перевантаження, які виникають внаслідок підвищеної концентрації уваги. Усі ці фактори негативно впливають на здоров'я працівників відділу АСУ ТП та сприяють виникненню професійних захворювань: комп'ютерний зоровий синдром; радіохвильова хвороба; синдром висихання рогівки ока; кистьовий тунельний синдром; захворювання шкіри; захворювання кишкового тракту; серцево-судинні захворювання; комп'ютерна алергія.

3. Заходи щодо поліпшення умов праці робітників відділу АСУ ТП

Для зменшення шкідливого впливу негативних факторів виробничого середовища на працівників відділу АСУ ТП необхідно вжити заходів щодо поліпшення їх умов праці: при облаштуванні робочих місць з ПК не допускати розміщення комп'ютеру у підвальних приміщеннях; приміщення має бути обладнане системою опалення, кондиціонування повітря, вентиляцією; для боротьби зі статичним полем підтримувати відносну вологість повітря на рівні 50-60% за допомогою кондиціонування, для підлоги використовувати матеріали з антистатичного матеріалу; дотримуватися параметрів: площа на одного

працюючого – 6 м^2 , об'єм – 20 м^3 ; відстань від робочого місця з ПК до стіни з вікном повинна становити не менше ніж 1,5 м, від інших стін – на відстані 1 м, а відстань між столами – 1,5 м; не застосовувати для обробки приміщення полімерні матеріали, що виділяють шкідливі хімічні речовини; відстань від екрану комп'ютера до очей повинна складати 50-70 см; час праці за ПК не повинен перевищувати 20 годин на тиждень, при цьому робити перерву кожні 45 хвилин на 5 хвилин.

Для зменшення шкідливого впливу негативних факторів виробничого середовища робоче місце з ПК повинно відповідати наступним гігієнічним вимогам: екран та клавіатура повинні розташовуватись на оптимальній відстані від очей користувача, що становить 600...700 мм; висота робочої поверхні робочого столу має регулюватися в межах 680...800 мм; робочий стіл повинен мати простір для ніг заввишки не менше ніж 600 мм, завширшки не менше ніж 500 мм, завглибшки (на рівні колін) не менше ніж 450 мм, на рівні простягнутої ноги - ніж 650 мм.

Для зменшення негативного впливу мобільного апарату необхідно: скоротити до мінімуму час розмови по телефону; обирати мобільний телефон з мінімальним значенням SAR (Specific Absorbtion Rate). SAR – одиниця виміру питомої величини поглинання випромінювання організмом людини; максимальна потужність випромінювання телефону. Максимальне значення SAR у Європі складає 2 Вт/кг ;[4] підносити термінал до вуха після з'єднання з абонентом. Під час здійснення з'єднання збільшується потужність випромінювання телефону.

Згідно з приведеними вимогами можемо розробити схему необхідного розташування робочих місць, яка приведена на рис. 1:

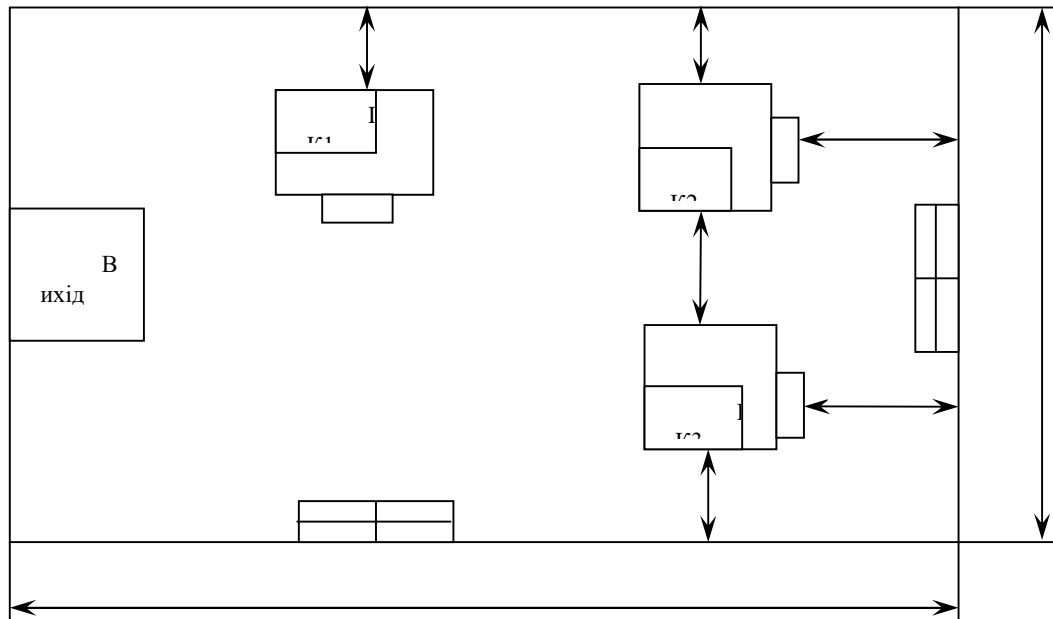


Рисунок А1 – Облаштованість робочих місць із ПК відділу

Таким чином, дотримано всіх вимог. При цьому працюючі не звернені обличчям до вікна при роботі за комп'ютером.

4. Розрахунок кондиціонування та вентиляції для відділу АСУ ТП

У відділі АСУ ТП є джерела екстремальних температур, тому необхідно визначити необхідні умови їх вентилявання. Витрату повітря у відділі з додатковим тепловиділенням визначаємо по формулі:

$$L = \frac{Q_{\text{над}}}{c \cdot p (t_{\text{в}} - t_{\text{н}})}, \quad (\text{A.1})$$

де $Q_{\text{над}}$ - надлишкове виділення тепла в робочому приміщенні, ккал/год.;

c - теплоємність повітря (0,237 ккал/кг);

p - обсягова вага повітря (1,226 кг/ м³);

$t_{\text{в}}$ - температура витяжного повітря (26°С);

$t_{\text{н}}$ - температура повітря, що притікає (18°С).

Розрахуємо надлишкове надходження тепла по формулі:

$$Q_{над} = Q_{уст} + Q_{пер} + Q_{осв} + Q_{ср}, \quad (A.2)$$

де $Q_{уст}$ - виділення тепла від устаткування;

$Q_{пер}$ - виділення тепла робітниками;

$Q_{осв}$ - надходження тепла від електричного освітлення;

$Q_{ср}$ - надходження тепла від сонячної радіації через вікна.

Визначимо виділення тепла від устаткування по формулі:

$$Q_{уст} = P \cdot K_a \cdot K_b \cdot 860 \text{ ккал/год}, \quad (A.3)$$

де P - сумарна потужність устаткування, кВт/год;

K_a - коефіцієнт установленної потужності (0,95);

K_b - коефіцієнт одночасної роботи (1,0).

$$Q_{уст} = [(x_1 \cdot k_1) + (x_2 \cdot k_2) + (x_3 \cdot k_3) + (x_4 \cdot k_4)] \cdot K_a \cdot K_b \cdot 860 \text{ ккал/год}, \quad (A.4)$$

де x - кількість системних блоків, моніторів, принтерів, кондиціонерів відповідно; k - потужність системних блоків, моніторів, принтерів, кондиціонерів відповідно.

Розрахуємо:

$$Q_{уст} = [(3 \cdot 0,5) + (3 \cdot 0,1) + (1 \cdot 0,4) + (1 \cdot 6,3)] \cdot 0,95 \cdot 1 \cdot 860 = 490,2 \text{ ккал/год}. \quad (A.5)$$

Визначимо виділення тепла від обслуговуючого персоналу за допомогою формули:

$$Q_{пер} = n \times g, \quad (A.6)$$

де n - кількість працюючих;

g - кількість тепла, що виділяє один працівник за годину (100 ккал/год.)

Розрахуємо:

$$Q_{\text{пер}} = 6 \times 100 = 600 \text{ ккал/год.} \quad (\text{A.7})$$

Визначимо надходження тепла від електричного освітлення по формулі:

$$Q_{\text{осв}} = E_{\text{м}} \cdot g_1 \cdot S, \quad (\text{A.8})$$

де $E_{\text{м}}$ – нормована освітленість для цієї зорової роботи, приймаємо рівним 400 лк;

g_1 – питома тепловиділення на 1 м² підлоги при 1 лк освітленості (для люмінесцентних ламп – 0,05 ккал/год.);

S – площа приміщення, м².

Розрахуємо:

$$Q_{\text{осв}} = 400 \cdot 0,05 \cdot 45,44 = 908,8 \text{ ккал/год.} \quad (\text{A.9})$$

Визначимо надходження тепла від сонячної радіації через вікна по формулі:

$$Q_{\text{ср}} = F \cdot g_2 \cdot K_{\text{осл}}, \quad (\text{A.10})$$

де F - площа віконних прорізів (1,55 м²);

g_2 - кількість тепла, що надходить через 1 м² віконного прорізу (65 ккал/год.);

$K_{\text{осл}}$ - коефіцієнт ослаблення, приймаємо - 0,4.

Розрахуємо:

$$Q_{\text{ср}} = 1,55 \cdot 65 \cdot 0,4 = 40,3 \text{ ккал/год.} \quad (\text{A.11})$$

Визначимо кількість надлишкового тепла:

$$Q_{\text{над}} = 490,2 + 600 + 908,8 + 40,3 = 2039,3 \text{ ккал/год.} \quad (\text{A.12})$$

Визначимо витрати повітря в приміщенні:

$$L = \frac{2039,3}{0,237 \cdot 1,226 \cdot (26-18)} = 877,31 \text{ м}^3 / \text{год} \quad (\text{A.13})$$

Існуюча в наявності система вентилявання має продуктивність 600 куб. м./годину, але це не задовольняє необхідним нормативам. Потрібно замінити функціонуючу систему вентилявання на більш потужнішу. Параметри мікроклімату на робочих місцях регламентуються ДСН 3.3.6.042-99 «Санітарні норми мікроклімату виробничих приміщень». Відповідно доданих санітарних норм температура повітря, швидкість руху повітря і відносна вологість у холодні періоди року повинна складати 22-24 градуса по Цельсію, 0,1 метра в секунду і 40-60 % відповідно. У теплі періоди року температура повітря повинна складати 23-25 градусів Цельсія, рухливість повітря 0,1-0,2 метрів секунду, вологість 40-60 %. Температура може коливатися від 22 до 26 градусів Цельсія при збереженні всіх інших параметрів мікроклімату. Вище зазначені норми цілком відповідають фактичним відділу АСУ ТП.

5. Розрахунок системи загального рівномірного освітлення з лампами розжарювання

Розміри відділу: довжина ($a=7,1$ м), ширина ($b=6,4$ м), висота ($h=3$ м). Визначимо норми освітлення та розрахункову висоту для виділених зон. Мінімальна освітленість за нормами $E=400$ лк. Приміщення має світлу побілку: коефіцієнт відбиття - $P_{\text{стелі}} = 70\%$, $P_{\text{стін}} = 50\%$. Висота робочих поверхонь (столів) $h_p = 0,7$ м. Для освітлення прийнято світильники типу УПМ-15, які підвищуються до стелі, відстань від світильника до стелі $h_c = 0,5$ м. Мінімальна освітленість за нормами $E=400$ лк.

1) Визначимо висоту підвісу світильників над підлогою:

$$h_0 = H - h_C = 3 - 0,5 = 2,5 \text{ м.} \quad (\text{A.14})$$

Для світильників загального освітлення з лампами розжарювання потужністю до 200 Вт мінімальна висота підвісу над підлогою відповідно до СНіП П-4-79 повинна бути у межах 2,5 - 4,0 м, залежно від характеристики світильника. В нашому випадку вона відповідає цій вимозі.

- 2) Визначимо висоту підвісу світильника над робочою поверхнею:

$$h = h_0 - h_p = 2,5 - 0,7 = 1,8 \text{ м.} \quad (\text{A.15})$$

Рівномірність освітлення досягається при відповідному співвідношенні відстані між світильниками (L) і висоти їх підвісу (h).

- 3) Визначимо рекомендовану відстань між світильниками:

$$L = 0,7h = 0,7 \cdot 1,8 = 1,26 \text{ м.} \quad (\text{A.16})$$

- 4) Розрахуємо необхідну кількість світильників:

$$N = \frac{ab}{L^2} = \frac{7,1 \cdot 6,4}{1,16^2} = 33,77 \approx 35 \quad (\text{A.17})$$

Приймаємо 35 світильників, враховуючи розміри приміщення розміщуємо їх у 3 ряди по 7 штук.

- 5) Світловий потік лампи світильника визначається за формулою:

$$\Phi_{\text{л}} = \frac{E \cdot K_3 \cdot S \cdot Z}{N \cdot n \cdot \eta} \text{ ккал / год}, \quad (\text{A.18})$$

де E - нормативна освітленість, лк;

K₃ - коефіцієнт запасу, що враховує зниження освітленості в результаті забруднення та старіння ламп (1,5);

S - площа приміщення, що освітлюється, м²;

Z - коефіцієнт нерівномірності освітлення для ламп розжарювання (=1,15);

N - кількість світильників;

n - кількість ламп у світильнику;

η - коефіцієнт використання світового потоку, який визначається за світлотехнічними таблицями залежно від показника приміщення (i) та коефіцієнтів відбиття стін та стелі.

6) Визначимо показник приміщення для кожної зони:

$$i = \frac{ab}{h(a+b)} = \frac{7,1 \cdot 6,4}{1,8 \cdot (7,1 + 6,4)} = 1,8 \quad (\text{A.19})$$

7) З таблиці 3.25 [2] знаходимо коефіцієнт використання ($\eta = 0,52$) для світильників УПМ-15 (при $P_{\text{стелі}} = 70\%$, $P_{\text{стін}} = 50\%$).

8) Світловий потік одного світильника, а значить і лампи, оскільки за конструктивним виконанням у світильнику цього типу встановлена лише одна лампа, дорівнює:

$$\Phi = \frac{E \cdot S \cdot K_3 \cdot Z}{N \cdot \eta} = \frac{400 \cdot 45,44 \cdot 1,5 \cdot 1,15}{35 \cdot 0,52} = 1722,73 \text{ лм} \quad (\text{A.20})$$

9) З таблиці 3.27 [2] обираємо лампу БК (біспіральна криптонова) потужністю 100 Вт, світловий потік якої становить 1450 лм. Хоча це значення менше розрахованого на 14%, однак не перевищує встановлену норму ($-0\% < \Delta\Phi_{\text{л}} < +20\%$). Сумарна електрична потужність усіх світильників, встановлених у приміщенні становить:

$$\Sigma P_{\text{св}} = P \cdot N = 100 \cdot 35 = 3,5 \text{ кВт}. \quad (\text{A.21})$$

6. Пожежна безпека

У приміщенні відділу АСУ ТП основні міри для забезпечення пожежної безпеки визначає Інструкція про заходи пожежної безпеки для службових приміщень. Вона є обов'язковою для виконання всіма співробітниками. В інструкції про засоби пожежної безпеки для службових приміщень забороняється: улаштовувати тимчасові електромережі, застосовувати саморобні плавкі вставки в запобіжниках, прокладати електричні проводи безпосередньо по пальній основі, експлуатувати світильники зі знятими ковпаками (розсіювачами), використовувати саморобні подовжувачі, що не відповідають вимогам Правил пристрою електроустановок; пристосовувати вимикачі, штепсельні розетки для підвішування одягу й інших предметів, обгортати електролампи і світильники, заклеювати ділянки електромережі пальною тканиною, папером; використовувати побутові електрокип'ятильники, чайники тощо без непалених підставок, залишати без нагляду включеними в електромережу кондиціонери, комп'ютери, рахункові і друкарські машинки і т.п.; захарашувати підступи до засобів пожежогасіння, використовувати пожежні крани, рукави і пожежний інвентар не за призначенням, зберігати документи, різні матеріали, предмети й інвентар у шафах (нішах) інженерних комунікацій; курити (крім спеціально відведених для цього адміністрацією місць, позначених написом «Місце для паління» і забезпечених урною чи попільницею з непаленого матеріалу), проводити зварювальні й інші вогневі роботи без оформлення відповідного дозволу, застосовувати легкозаймисті рідини.

7. Безпека при надзвичайних ситуаціях на підприємстві

Меблі й устаткування повинні розміщатися таким чином, щоб забезпечувався вільний евакуаційний прохід до дверей виходу з приміщення (шириною не менше 1 м). Евакуаційні шляхи і виходи необхідно постійно держати вільними, нічим не захарашувати. Засоби протипожежного захисту в приміщеннях потрібні триматися у справному стані.

У випадку виявлення пожежі слід: негайно повідомити державну пожежну охорону за телефоном «101», вказати при цьому адресу, кількість поверхів, місце виникнення пожежі, наявність людей, своє прізвище; повідомити про пожежу керівництву, а в нічний час черговому охоронцю;

У разі можливості почати гасіння пожежі наявними засобами, організувати зустріч пожежних підрозділів.

При виникненні пожежі у початковій стадії його розвитку випромінюється тепло, токсичні продукти згоряння, імовірні руйнування будівельних споруд. Тому слід як можна швидше провести евакуацію людей із палаючої будівлі. Показником ефективності евакуації є час, протягом якого працівники можуть при необхідності залишити окремі приміщення і будівлю в цілому. Безпека евакуації досягається тоді, коли час евакуації не перевищує час настання критичної фази розвитку пожежі, тобто часу від початку пожежі до досягнення граничних для людини впливів факторі пожежі (критичних температур, ступені задимлення, зниження концентрації кисню і т.п.). Число евакуаційних виходів повинно бути не менш двох. Вони повинні розташовуватися розосереджено.

Двері на шляхах евакуації повинні відкриватися у напрямку виходу із будівлі. У кожному приміщенні на видному місці повинен бути вивішений план евакуації при пожежі. При пожежі обов'язково необхідно враховувати небезпечні чинники і механізм їх дії на людину. Користуватися ліфтом категорично забороняється. Шахта ліфта є шляхом для поширення диму і отруйних продуктів горіння, до того ж при пожежі ліфт часто відключають і можна опинитися в пастці при пожежі.

Якщо ви знаходитесь в приміщенні де немає пожежі, але відрізани вогнем, димом, високою температурою від головних шляхів евакуації, то в першу чергу необхідно заважити доступу диму і продуктів горіння в це приміщення. Для чого необхідно негайно закрити усі щілини у дверях та під ними змоченими водою ганчірками, рушниками, робочими халатами та іншим.

Якщо приміщення все ж заповнено димом, необхідно підповзти до вікна, закрити при цьому рот та ніс змоченою тканиною, яка грає роль фільтру та в певної мірі захищає від продуктів горіння.

Рухатись у задимленій зоні поповзом або максимально пригнувшись, необхідно тому що більшість нагрітих газоподібних отруйних речовин та дим збираються у верхній зоні приміщення, окрім цього, в приміщенні при горінні температура на рівні очей людини у 6 разів вище за температуру на рівні полу, до того ж внизу завжди зберігається більша концентрація кисню. Коли ви опинились біля вікна трохи відкрийте його та дихайте через щілину, очікуючи прибуття пожежників. При їх прибутті негайно зверніть на себе увагу. Ніколи не стрибайте через вікно без відомої на це необхідності (кожний другий стрибок з 4-го поверху при пожежі - смертельний).

ДОДАТОК Б. Приклад коду серверної частини

```
const express = require('express');
const http = require('http');
const socketIo = require('socket.io');
const app = express();
const server = http.createServer(app);
const io = socketIo(server);
// Обробка підключення користувачів
io.on('connection', (socket) => {
  console.log('Користувач підключився');
  // Обробка команд
  socket.on('sendCommand', (command) => {
    console.log(`Отримана команда: ${command}`);
    // Для цього прикладу просто відправимо команду назад
    socket.emit('commandResponse', `Команда виконана: ${command}`);
  });
  // Відключення користувача
  socket.on('disconnect', () => {
    console.log('Користувач відключився');
  });
});
// Запуск сервера
server.listen(3000, () => {
  console.log('Сервер працює на порту 3000');
});
```

ДОДАТОК В. Клієнтська частина: HTML, CSS та JavaScript

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Веб-додаток для віддаленого керування</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      margin: 20px;
    }
    input, button {
      padding: 10px;
      font-size: 16px;
      margin: 10px 0;
    }
    #response {
      margin-top: 20px;
      padding: 10px;
      border: 1px solid #ccc;
      background-color: #f9f9f9;
    }
  </style>
</head>
<body>
  <h1>Віддалене керування сервером</h1>
  <input type="text" id="commandInput" placeholder="Введіть команду..." />
  <button id="sendCommandBtn">Відправити команду</button>
```

```

<div id="response"></div>
<script src="/socket.io/socket.io.js"></script>
<script>
  const socket = io(); // Підключення до серверу через WebSocket
  const sendCommandBtn = document.getElementById('sendCommandBtn');
  const commandInput = document.getElementById('commandInput');
  const responseDiv = document.getElementById('response');
  sendCommandBtn.addEventListener('click', () => {
    const command = commandInput.value;
    if (command) {
      socket.emit('sendCommand', command); // Відправка команди на сервер
      commandInput.value = ""; // Очищення поля введення
    }
  });
  // Отримання відповіді від сервера
  socket.on('commandResponse', (response) => {
    responseDiv.textContent = response;
  });
</script>
</body>
</html>

```