

Центральноукраїнський національний технічний університет
Центр заочної та дистанційної освіти
Кафедра кібербезпеки та програмного забезпечення

”Допущено до захисту”
Завідувач кафедри кібербезпеки
та програмного забезпечення
д.т.н., професор
_____ Олексій СМІРНОВ
“ ____ ” _____ 2024 р.

ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА
за другим (магістерським) рівнем вищої освіти
на тему
“Дослідження та програмна реалізація системи керування
інфраструктурою на основі Cisco Digital Ceiling”

Виконав здобувач вищої освіти
II курсу, групи КІ-23Мз
ОПП «Комп’ютерна інженерія»
спеціальності 123 «Комп’ютерна інженерія»
_____ Лівер В.С.
« ____ » _____ 2024 р.

Керівник проекту
кандидат фізико-математичних наук, доцент
_____ Якименко Н.М.
« ____ » _____ 2024 р.
Рецензент _____

Центральноукраїнський національний технічний університет

Центр *Заочної та дистанційної освіти*

Кафедра *Кібербезпеки та програмного забезпечення*

Рівень вищої освіти *магістр*

Галузь знань 12 *“Інформаційні технології”*

Спеціальність 123 *“Комп’ютерна інженерія”*

Освітньо-професійна (освітньо-наукова) програма *“Комп’ютерна інженерія”*

ЗАТВЕРДЖУЮ

Завідувач кафедри

д.т.н., проф.

Олексій СМІРНОВ

« 6 » вересня 2024 року

**ЗАВДАННЯ НА ВИПУСКНУ КВАЛІФІКАЦІЙНУ РОБОТУ ЗА
ДРУГИМ (МАГІСТЕРСЬКИМ) РІВНЕМ ВИЩОЇ ОСВІТИ
ЗДОБУВАЧА ВИЩОЇ ОСВІТИ**

Лівер Вікторії Сергіївни

(прізвище, ім'я, по батькові)

1. Тема роботи

Дослідження та програмна реалізація системи керування інфраструктурою на основі Cisco Digital Ceiling

2. Керівник роботи

Якименко Наталія Миколаївна, канд. фіз.-мат. наук, доцент

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

затверджені наказом вищого навчального закладу № 21-13 від 07.08.2024 року

3. Строк подання студентом роботи до захисту 2.12.2024 р.

4. Мета та завдання випускної кваліфікаційної роботи: *Метою розробки є дослідження та програмна реалізація системи керування інфраструктурою на основі Cisco Digital Ceiling*

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

1. Призначення та область використання.

6. Наукова новизна.

2. Перегляд аналогічних існуючих систем.

7. Маркетингове та економічне обґрунтування ІТ-проєкту.

3. Опис і обґрунтування проєктних рішень.

8. Заходи з охорони праці та техніки безпеки.

4. Етапи програмування системи.

9. Висновки.

5. Впровадження системи в промислову експлуатацію

6. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)

Наукова новизна

1 аркуш

Структурна схема системи

1 аркуш

Функціональна схема системи

1 аркуш

Діаграма процесів

1 аркуш

Блок-схема алгоритму роботи додатку

2 аркуша

Показники економічної ефективності

1 аркуш

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Економічний	Доренська А.О.	05.10.2024	14.11.2024
Охорона праці	Марченко К.М., к.т.н., доцент	06.10.2024	16.11.2024

7. Дата видачі завдання « 6 » вересня 2024 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти	Строк виконання етапів випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти	Примітка
1.	Аналіз існуючих систем	10.10.2024 р.	
2.	Постановка задачі, оформлення ТЗ	15.10.2024 р.	
3.	Розробка моделі компонента	20.10.2024 р.	
4.	Розробка структур даних	25.10.2024 р.	
5.	Розробка алгоритмів зв'язку та відображення	30.10.2024 р.	
6.	Програмування алгоритмів	10.11.2024 р.	
7.	Розрахунок економічної ефективності	13.11.2024 р.	
8.	Розрахунки з охорони праці та техніки безпеки	15.11.2024 р.	
9.	Оформлення ПЗ	17.11.2024 р.	
10.	Попередній захист роботи	2.12.2024 р.	

Дата видачі завдання
« 6 » вересня 2024 р.

Підпис керівника

_____ (прізвище та ініціали)

Завдання прийнято до виконання
« 6 » вересня 2024 р.

Підпис здобувача

_____ (прізвище та ініціали)

АНОТАЦІЯ

Лівер В.С. Дослідження та програмна реалізація системи керування інфраструктурою на основі Cisco Digital Ceiling. 123 Комп'ютерна інженерія. Центральноукраїнський національний технічний університет. Кропивницький. 2024.

В даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти розроблено програмне забезпечення, яке призначено для системи керування інфраструктурою на основі Cisco Digital Ceiling.

Метою розробки є дослідження та програмна реалізація системи керування інфраструктурою на основі Cisco Digital Ceiling.

Об'єктом дослідження є процес керування інфраструктурою на основі Cisco Digital Ceiling.

Предметом дослідження є методи керування інфраструктурою на основі Cisco Digital Ceiling.

Методи дослідження базуються на методах теорії комп'ютерних мереж, методах математичної статистики, методах розробки програмного забезпечення.

Результат роботи – програмна реалізація системи керування інфраструктурою на основі Cisco Digital Ceiling.

В процесі роботи над програмною моделлю виконано аналіз існуючих апаратних та програмних засобів. В повній мірі описані всі компоненти розробленого програмного забезпечення.

Розроблено зручний інтерфейс користувача. Наведені інструкції по роботі з програмними засобами.

Програма може використовуватися на ПЕОМ з ОС Windows 10/11.

Програму розроблено в середовищі Python.

Ключові слова: комп'ютерна інженерія, Cisco Digital Ceiling

ABSTRACT

Liver V.S. Research and software implementation of an infrastructure management system based on Cisco Digital Ceiling. 123 Computer engineering. Central Ukrainian National Technical University. Kropyvnytskyi. 2024.

In this graduation thesis for the second (master's) level of higher education, software is developed, which is intended for the infrastructure management system based on Cisco Digital Ceiling.

The purpose of the development is the research and software implementation of the infrastructure management system based on Cisco Digital Ceiling.

The object of research is the infrastructure management process based on Cisco Digital Ceiling.

The subject of research is infrastructure management methods based on the Cisco Digital Ceiling.

Research methods are based on computer network theory methods, mathematical statistics methods, and software development methods.

The result of the work is the software implementation of the infrastructure management system based on Cisco Digital Ceiling.

In the process of working on the software model, an analysis of existing hardware and software was performed. All components of the developed software are fully described.

A convenient user interface has been developed. Instructions for working with software tools are provided.

The program can be used on a PC with Windows 10/11 OS.

The program was developed in the Python environment.

Keywords: computer engineering, Cisco Digital Ceiling

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ	3
ВСТУП.....	4
1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ	7
1.1 Призначення системи.....	7
1.2 Область застосування.....	7
2 ПЕРЕГЛЯД АНАЛОГІЧНИХ ІСНУЮЧИХ СИСТЕМ	9
2.1 Огляд існуючих систем, технологій, архітектур та програмних рішень за профілем теми випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти.....	9
2.2 Обґрунтування вибору засобів для побудови системи та мови програмування.....	19
2.3 Розгорнута постановка завдання	24
3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ	25
3.1 Опис функціонування системи	25
3.2 Розробка структурної схеми.....	30
3.3 Розробка функціональної схеми	33
3.4 Розробка діаграми процесів.....	42
4 РЕАЛІЗАЦІЯ РОБОТИ. РОЗРАХУНКИ І ЕКСПЕРИМЕНТАЛЬНІ ДАНІ, ЩО ПІДТВЕРДЖУЮТЬ ВІРНІСТЬ ПРОЕКТНИХ ТА ПРОГРАМНИХ РІШЕНЬ.....	44
4.1 Розробка блок-схем та опис алгоритмів функціонування системи.....	44
4.2 Захист розробленого програмного забезпечення.....	52
5 ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ	54
6 НАУКОВА НОВИЗНА	60

						ВКРМ-123.24.0003.00.00.ПЗ		
Вим	Арк.	№ докум.	Підп.	Дата		Лім.	Аркуш	Аркушіє
Розроб.	Тівер В.С.				Дослідження та програмна реалізація системи керування інфраструктурою на основі Cisco Digital Ceiling	М	1	85
Перев.	Якименко Н.М.					ЦНТУ КІ-23Мз		
Н.контр.	Коваленко А.С.							
Затв.	Смірнов О.А.							

7	МАРКЕТИНГОВЕ ТА ЕКОНОМІЧНЕ ОБҐРУНТУВАННЯ ІТ-ПРОЄКТУ	61
7.1	Визначення цільової аудиторії кінцевого готового продукту	61
7.2	Оцінка привабливості шляхом застосування методів експертних оцінок ...	62
7.3	Вибір методу оцінки вартості ПЗ	64
7.4	Розрахунок економічної ефективності від впровадження реалізованого ПЗ як фактору його привабливості.....	64
7.5	Пропозиція алгоритму просування проєкту розробки ПЗ	66
7.6	Оптимізація каналів збуту та шляхів реалізації ПЗ	67
7.7	Визначення ключових факторів успіху конкретного проєкту.....	68
8	ЗАХОДИ З ОХОРОНИ ПРАЦІ ТА ТЕХНІКИ БЕЗПЕКИ	69
8.1	Вступ.....	69
8.2	Аналіз санітарно-гігієнічних умов праці на робочому місці програміста ..	70
8.3	Розробка заходів з умов поліпшення охорони праці.....	73
8.4	Розрахункова частина	74
8.5	Висновки до розділу.....	76
9	ОСНОВНІ ВИСНОВКИ.....	77
	СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	79

КБПЗ-2024

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		2

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ

EOM	–	електронно-обчислювальна машина
ATM	–	Asynchronous Transfer Mode – асинхронний спосіб передачі даних
BER	–	Bit Error Rate – імовірність ушкодження одного біта
CBWFQ	–	class based weighted fair queueing
DiffServ	–	Differentiated Services – механізм який залежно від вимог до якості обслуговування записує в IP заголовки пакетів спеціальні мітки
DSCP	–	DiffServ CoSde Point
ICMP	–	Internet CoSntrol Message Protocol – міжмережний протокол управляючих повідомлень
ISP	–	Internet Service Provider – провайдер
LLQ	–	low latency queueing
MPLS-TE	–	Multiprotocol Label Switching Traffic Engineering
PQ	–	priority queueing
RIO	–	RED with Input Output
RTT	–	Round Trip Time – час між відправкою запиту та отриманням відповіді
STM	–	Synchronous Transfer Mode – синхронний спосіб передачі даних
QoS	–	Quality of Service – якість обслуговування
WFQ	–	Weighted Fair Queuing – механізм планування пакетних потоків даних
WRED	–	Weighted random early detection – взвішане значення довжини черги, у якості фактора, визначаючого імовірність відкидання пакета

ВСТУП

Актуальність теми. Cisco Digital Ceiling, запущений компанією Cisco, об'єднує багато мереж у будівлі, зокрема освітлення, опалення, охолодження та відео, в одну мережеву платформу. Тобто за фактом Cisco Digital Ceiling реалізує Інтернет речей (IoT), а точніше у даній роботі проект розумного будинку. Цифрова трансформація спричинить серйозні зміни способів побудови ІТ-систем і споживання ІТ-сервісів. Так, Cisco готує нові стеки ІТ-рішень, оптимізовані під контейнери й мікросервіси.

Інтернет речей (IoT) і цифрова трансформація стали головними темами минулі в середині лютого в Берліні форуму CiscoLive 2024 – найбільшого заходу за межами США за всю історію компанії, що відвідали близько 12 тис. чоловік. По оцінці Cisco, за останні два роки число стартапів, що працюють над рішеннями IoT, виросло на порядок – з 127 компаній в 2022 році до 1502 в 2023-м, а венчурні інвестиції в цю область торік склали майже 2 млрд доларів. За той же період число встановлених датчиків і сенсорів збільшилося приблизно в п'ять разів – до 54,7 млрд, а для взаємодії з ними задіється 18,2 млрд мережних підключень.

Поняття «Інтернет речей» і «цифровізація», звичайно, не є синонімами, але часто використовуються у зв'язуванні: IoT формує технологічну платформу для нових додатків, які «оцифровують» бізнес, відкриваючи двері для нових можливостей і бізнесів-моделей. При цьому відношення до терміна IoT далеко не однозначне. Якщо технічним фахівцям поняття «цифровізація» знайома давно, то кардинальна зміна відносно компаній до цифровізації відбулося відносно недавно, коли про неї заговорили представники самих різних галузей: автовиробники, лікарі, учителі й т.д.

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		4

Мета й завдання дослідження. Метою роботи є дослідження та програмна реалізація системи керування інфраструктурою на основі Cisco Digital Ceiling.

Для досягнення поставленої мети визначена програма дослідження, що складається з наступних завдань:

– Огляд існуючих систем керування інфраструктурою на основі Cisco Digital Ceiling.

– Дослідження системи керування інфраструктурою на основі Cisco Digital Ceiling.

– Програмна реалізація системи керування інфраструктурою на основі Cisco Digital Ceiling.

Об'єктом дослідження є процес керування інфраструктурою на основі Cisco Digital Ceiling.

Предметом дослідження є методи керування інфраструктурою на основі Cisco Digital Ceiling.

Методи дослідження базуються на методах теорії комп'ютерних мереж, методах математичної статистики, методах розробки програмного забезпечення.

Наукова новизна отриманих результатів. У процесі рішення завдань, обумовлених цілями дослідження, отримані наступні результати:

– Удосконалено метод керування інфраструктурою на основі Cisco Digital Ceiling.

– Розроблено вітчизняний продукт керування інфраструктурою на основі Cisco Digital Ceiling, який має більш широкі можливості, на відміну від існуючих аналогів.

Практична цінність отриманих результатів полягає в тому, що розроблені алгоритми дозволяють успішно вирішувати задачі керування інфраструктурою на основі Cisco Digital Ceiling.

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		5

Достовірність наукових результатів підтверджена теоретичними викладеннями, даними комп'ютерного моделювання, коректними дослідженнями параметрів на функціонуючій обчислювальній мережі, а також відповідністю отриманих результатів окремим результатам, наведеним у науковій літературі.

Робота апробована на LVII Науково-технічній конференції здобувачів вищої освіти LV науково-технічної конференції «Наука в ЦНТУ: основні досягнення та перспективи розвитку» (2024 р.), основні положення випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти надруковані у статті збірника праць молодих науковців ЦНТУ, випуск №15.

Таким чином, виходячи з вищеперерахованого, дослідження та програмна реалізація системи керування інфраструктурою на основі Cisco Digital Ceiling, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти.

КБПЗ - 2024

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		6

1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ

1.1 Призначення системи

Представники Cisco приводять і демонструють велику кількість вражаючих прикладів використання IoT у різних сферах. Один з показових прикладів – проект у страховій компанії АОК. Це одна із самих великих фірм медичного страхування в Німеччині, її клієнтами є більше 24 млн чоловік, тобто приблизно третина населення країни застрахована в АОК. При участі Cisco і використанні її рішень впроваджена система по обміну медичними даними. Це можуть бути самі різні показники: тиск, пульс, рівень цукру в крові й т.д. Маючи ці дані, лікарі можуть попередити можливий розвиток захворювання в здорових людей або скорегувати лікування захворілих. Крім того, у рамках розгорнутого в АОК рішення реалізована можливість вилучених консультацій за допомогою відеозв'язку, що особливо важливо для якісного медичного обслуговування сільських районів. Для людей, що прагнуть вести здоровий спосіб життя, розроблений спеціальний додаток FitMit АОК для смартфона: зокрема, воно фіксує тривалість занять спортом у тренажерному залі й нараховує за це бонусні бали, які можна потім обміняти на призи, різні знижки та ін.

1.2 Область застосування

Чому Cisco проявляє такий великий інтерес до теми IoT і цифрової трансформації? Один із ключових елементів будь-якої цифровізації – сучасна мережа, а Cisco займається рішеннями для мережних інфраструктур уже близько 30 років, будучи визнаним лідером у цій області. Розвиток додатків IoT, безумовно, міняє вимоги до мереж – наприклад, посилює вимоги до затримки трафіку. Для зниження затримки Cisco розробила й реалізує концепцію «мрячних

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		7

обчислень», коли частина функцій по аналізу даних виконується максимально близько до їхніх джерел, наприклад у прикордонному маршрутизаторі. Реалізований у цього ПЗ інтелект дозволяє оперативно виробляти керуючі впливи. І тільки частина даних або результати їхньої обробки й аналізу відправляють «наверх», у хмари.

Таким чином, виходячи з вищеперерахованого, дослідження та програмна реалізація системи керування інфраструктурою на основі Cisco Digital Ceiling, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти.

КБПЗ_2024

					VKPM-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		8

2 ПЕРЕГЛЯД АНАЛОГІЧНИХ ІСНУЮЧИХ СИСТЕМ

2.1 Огляд існуючих систем, технологій, архітектур, програмних рішень за профілем теми випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти

Короткий огляд LPWAN технологій для IoT Sigfox

Компанією Sigfox (Франція) розроблена й впроваджується технологія зверхвужькополосного (UNB – Ultra Narrow Band) бездротового зв'язку для передачі даних у субгігагерцовому неліцензованому діапазоні 868.8 МГц. Мережа зв'язку компанії розгорнута в Європі: Франція, Італія, Великобританія, Ірландія, Іспанія, Португалія, Люксембург, Нідерланди, Бельгія, Данія.

Технології LTE-M (LTE-MTC), NB LTE-M, NB-IoT

Ці технології ще називають CIoT (від англ. Cellular Internet of Things – Інтернет речей у мережах стільникового зв'язку), маючи на увазі, що вони розробляються спеціально під операторів стільникового зв'язку. Технології покликані, насамперед, охопити ринок IoT існуючими стільниковими операторами зв'язку з найменшими витратами. Технологію LTE-M (LTE-MTC) розробляють Nokia, а NB LTE-M Huawei Technologies. NB-IoT використовує DSSS (від англ. Direct Sequence Spread Spectrum – метод прямої послідовності для розширення спектра), просувається компаніями Huawei, Ericsson, Qualcomm, і Vodafone. Розгортання цих мереж можливо установкою додаткових трансиверів з модернізацій програмного забезпечення на існуючому встаткуванні, не потрібно навіть заміна антенно-фідерного господарства. Ці стандарти працює в субгігагерцовому діапазоні, частоти вимагають одержання ліцензії.

Технологія WiFi HaLow

Крім зазначених стандартів WiFi Alliance у січні 2023 оголосив про початок розробки нового стандарту HaLow. Даний вид зв'язку буде функціонувати в

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		9

діапазоні менш 1 ГГц, на підставі незатвердженої (поки) специфікації IEEE 802.11ah

Технологія LoRa

Технологія LoRa розроблена компанією Semtech. У січні 2015 року створений LoRa Alliance для стандартизації протоколу LoRaWAN в LPWAN (Low Power Wide Area Networks) мережах. Організаторами LoRa Alliance з'явилися такі компанії як: Actility, Cisco, Eolane, IBM, Kerlink, IMST, MultiTech, Sagemcom, Semtech, і Microchip Technology.

Стан IoT 2024: 10 нових тенденцій IoT, що сприяють зростанню ринку

Тенденція 1: Напівпровідникові компанії інвестують у безпеку вбудованих чіпсетів

Напівпровідникові компанії все більше інвестують у захист вбудованих чіпсетів, щоб усунути зростаючі загрози безпеці, з якими стикаються пристрої IoT. Захист апаратного забезпечення на рівні чіпсета за допомогою захищених елементів і фізичних неклонованих функцій (PUF) може допомогти захистити дані, що надходять із периферійних пристроїв у хмару.

Приклад

Екосистема американської напівпровідникової компанії Intel зростає завдяки партнерам із безпеки, таким як американська компанія цифрової автентифікації Intrinsic ID. У лютому 2024 року Intel Foundry додала Intrinsic ID до своєї програми Accelerator IP Alliance, щоб забезпечити доступність апаратних рішень root-of-trust для членів Foundry. Визнаючи, що безпека та надійність є цінними для додатків, Foundry відкрила доступ до сімейства продуктів QuiddiKey X00 Intrinsic ID із рішеннями root-of-trust (RoT), які використовують стандартну SRAM як PUF для генерації апаратної RoT без потреби додаткового захисту кремній.

Компанії, що займаються виробництвом напівпровідників, знаходяться в авангарді боротьби з зростаючими загрозами безпеці, з якими стикаються пристрої IoT, зосереджуючись на інвестиціях у безпеку вбудованих чіпсетів, оскільки апаратне забезпечення формує фундаментальний рівень.

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		10

Тенденція 2: апаратне забезпечення промислової автоматизації стає розумнішим завдяки інтеграції чіпсетів ШІ

З розвитком технології штучного інтелекту компанії тепер прагнуть використовувати штучний інтелект на периферії, збільшуючи попит на аналіз даних у реальному часі також на межі. Чіпсети штучного інтелекту також стають меншими за розміром, а потужність зростає. Ця тенденція призвела до появи IPC і шлюзів із вбудованими чіпсетами штучного інтелекту, що призвело до появи периферійного обладнання штучного інтелекту, яке може виконувати паралельні обчислення та навчати алгоритми з дуже низькою затримкою обчислювальної відповіді.

Однією з переваг використання чіпсетів AI на краю є прискорення обробки даних безпосередньо на промисловому обладнанні. Це, у свою чергу, зменшує мережевий трафік і підвищує безпеку, оскільки зменшується кількість даних, які надходять у хмару для обробки.

Приклад

На виставці SPS у листопаді 2023 року німецький виробник систем автоматизації та цифровізації Siemens представив свій ПК SIMATIC IPC520A Box PC із вбудованим 6-ядерним процесором NVIDIA Carmel для периферійних можливостей і графічним процесором NVIDIA Jetson Xavier NX для AI, що робить його придатним для операцій, орієнтованих на штучний інтелект. IPC520A розроблений для бездоганної роботи з додатками на основі штучного інтелекту в різних галузях промисловості, включаючи автоматизацію виробництва та логістику.

Удосконалення чіпсетів штучного інтелекту, спеціально розроблених для периферії, заслуговують на увагу. Вбудовування набору мікросхем штучного інтелекту в периферійне обладнання, таке як IPC і шлюз, наближає процес прийняття рішень до периферії та відкриває двері для нових програм IoT, таких як машинне бачення.

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		11

Тенденція 4: генеративний штучний інтелект має позитивний (а не негативний!) вплив на робочу силу у виробництві

Змагання за інтеграцією рішень GenAI у виробництво – і те, чим вони відрізняються від інших технологій досі – обертаються навколо швидкості впровадження та рівня інвестицій у технологію. Протягом трьох місяців після публічного запуску ChatGPT охопив приблизно 123 мільйони користувачів, що є неймовірним досягненням для нового типу продукту. Крім того, невдовзі після запуску ChatGPT Microsoft інвестувала 10 мільярдів доларів у OpenAI, сприяючи розширенню профілю ChatGPT. Ця інвестиція також показала серйозність, яку великі технологічні компанії, такі як Microsoft, приділяють цій технології, що призвело до того, що компанії в різних галузях замислилися над тим, як вони можуть використовувати GenAI у своїх процесах.

Впровадження нових технологій у виробництво часто пов'язане з негативним впливом на робочу силу. Однак очікується, що впровадження GenAI у виробництво сприятиме підвищенню зайнятості та підвищенню кваліфікації, перемістивши фокус з автоматизації на стратегічне зростання. Згідно з McKinsey, оскільки GenAI щорічно вносить у світову економіку 2,6–4,4 трильйона доларів США, виробники, ймовірно, збільшать свої інвестиції в технології ШІ.

Однак вплив штучного інтелекту на робочу силу здається нелогічним для поширених наративів автоматизації. За даними Manufacturing Leadership Council, 32% виробників очікують збільшення чисельності персоналу завдяки штучному інтелекту, що свідчить про те, що штучний інтелект створюватиме нові ролі та потребуватиме підвищення кваліфікації, а не просто автоматизації існуючих. Оскільки 96% виробників планують збільшити інвестиції в штучний інтелект, існує чітка тенденція до використання штучного інтелекту для економії коштів, зростання та отримання прибутку.

Нові посади, ймовірно, включатимуть менеджерів зі стратегії штучного інтелекту та спеціалістів з обробки даних, що відображатиме перехід до вищої когнітивної роботи.

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		13

розвивається.

Ринки хмарних гіперскейлерів наразі лідирують у витратах на хмарне програмне забезпечення, оскільки багато компаній уже взяли на себе витрати на хмарні технології, які можна використовувати для закупівлі програмного забезпечення з цих платформ.

Тенденція 7. Структура даних стає передовою еволюцією озер даних

Незважаючи на те, що це відносно новий термін, структура даних описує комплексну структуру інтеграції та управління даними. Він охоплює архітектуру, інструменти керування та спільні набори даних і призначений для допомоги організаціям у обробці їхніх даних.

Структури даних відрізняються від озер даних тим, що вони виходять за рамки зберігання необроблених даних, і від сховищ даних тим, що вони обробляють лише оброблені або вдосконалені дані. Основна перевага мереж даних полягає в тому, що вони пропонують узгоджений, послідовний інтерфейс користувача та доступ до даних у режимі реального часу для всіх членів організації, незалежно від їх глобального розташування.

Приклади

У 2023 році кілька великих постачальників засобів керування даними оновили свої вже існуючі рішення для обробки даних або запустили нові рішення:

– У лютому американський постачальник платформи інтеграції даних Talend – один із перших користувачів терміну data fabric – оголосив про оновлення свого рішення Talend Data Fabric, яке було запущено в 2015 році.

– У травні 2023 року американська технологічна та програмна компанія Microsoft представила Microsoft Fabric і запустила її в листопаді.

Враховуючи збільшення складності даних через експоненціальне зростання великих даних, що стимулюється гібридною хмарою, штучним інтелектом, Інтернетом речей і периферійними обчисленнями, здається, що для постачальників [пропонуючи структуру даних] є гарна можливість.

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		15

Тенденція 8: Гіпермасштабувальники орієнтують свої крайні стратегії, щоб інновувати та закріпити свою позицію на ринку Інтернету речей

Хмарні провайдери стратегічно адаптуються до розвитку ринку ІоТ. У нещодавніх розробках у сфері Інтернету речей відбулися важливі зміни серед основних провайдерів хмарних послуг, приділяючи більше уваги стратегіям периферії та контейнеризації.

Приклади

– Microsoft орієнтує свою стратегію Azure ІоТ на Kubernetes. Корпорація Майкрософт, зокрема, переорієнтувала свою стратегію на Kubernetes, що вказує на поворот у її пропозиції Azure ІоТ Operations . Цей перехід, що супроводжується організаційною реструктуризацією та припиненням каталогу сертифікованих пристроїв Azure, підкреслює, як стратегія Microsoft ІоТ, здається, розвивається зі зміною технологій і ринкової динаміки.

– Google припиняє роботу з ІоТ Core, але зберігає Manufacturing Connect через Kubernetes. Подібним чином увагу привернуло рішення Google припинити пропозицію ІоТ Core у серпні 2023 року. Незважаючи на це закриття, рішення Google ІоТ, Manufacturing Connect, залишається життєздатним завдяки своїй сумісній з Kubernetes архітектурі, стратегічній орієнтації, що відображає технічну майстерність компанії в цій сфері.

– AWS збільшує інвестиції в Інтернет речей завдяки покращенням Sitewise і Monitron. Здається, AWS подвоює перевагу ІоТ, що особливо помітно в поживленні Sitewise та інших пропозиціях, як-от Monitron.

У міру того як ринок Інтернету речей розвивається, хмарні гіганти, такі як Microsoft, Google і AWS, просуваються далі, використовуючи Kubernetes і розширюючи можливості периферійних обчислень. Їхні стратегії зосереджені не на тому, щоб самі обслуговувати окремі випадки використання, а радше на участі в рівні крайової (програмної) платформи, яка служить основою цифровізації для їхніх партнерів і клієнтів.

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		16

Тенденція 9: промислові постачальники активно інвестують у рішення DataOps

Кілька постачальників інвестують у промислові рішення DataOps для вирішення проблем інтеграції та аналізу даних.

Industrial DataOps – це підхід до інтеграції даних, зосереджений на підвищенні якості даних за допомогою контекстуалізації та моделювання. Цей підхід привертає все більше уваги в промисловому просторі підключення.

Приклади

Деякі з ключових постачальників, які пропонують промислові DataOps для контекстуалізації та моделювання даних, включають:

– Нові постачальники: Cognite – CDF , HighByte – Intelligence Hub , Prosys – OPC UA Forge , Litmus – Litmus Edge , Element – Unify і Crosser – Flow Studio.

– Традиційні промислові компанії: ABB – Ability Genix , Aveva – PI System , AspenTech – Inmation , GE Digital – Asset Modeler і Halliburton – DecisionSpace 365.

У світі ІІТ дані ОТ є наріжним каменем. Саме якість і контекст цих даних справді дають змогу зрозуміти. Ми спостерігаємо, як постачальники агресивно вдосконалюють інструменти DataOps для моделювання та контекстуалізації, як нові учасники, так і відомі OEM-виробники вдосконалюють свої рішення. Ці спільні зусилля підкреслюють ключову роль підвищення якості даних із різноманітних активів/програмного забезпечення для розкриття повного потенціалу цифрової трансформації».

Тенденція 10: Роботи з погодинною оплатою починають замінювати ручну працю через брак робочої сили

Компанії-виробники можуть отримати вигоду від обладнання як послуги (EaaS), замінивши операційні витрати, пов'язані з оплатою праці, іншим типом операційних витрат: робототехнікою як послугою (RaaS). RaaS – це відносно нова бізнес-модель, у якій робот надається машинобудівником на основі результату

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		17

(оплата за деталі, виготовлені з обладнанням) або на основі тривалості роботи (оплата за години використання обладнання), а не як пряма покупка.

Приклад

Американський виробник шасі причепів для вантажівок Cheetah Chassis вирішив найняти зварювальних робітників на годину та пояснив, що не може знайти достатньо зварювальників, щоб задовольнити попит. Її генеральний директор Гаррі Хартман пояснив, що раніше вона пробувала робототехніку, але вона була невдалою, оскільки не мала можливості програмувати та обслуговувати робітників. Завдяки RaaS Cheetah Chassis тепер може користуватися перевагами робототехніки без необхідності робити це, оскільки це надається постачальником RaaS.

Компанії, швидше за все, будуть споживати обладнання погодинно, якщо вони намагатимуться заповнити прогалини в робочій силі (яка вже оплачується погодинно), а не купувати нове обладнання».

Висновок

Сектор IoT переживає трансформаційні зміни. Новий сторінковий звіт «Стан IoT – 2024» підкреслює безперервну еволюцію та стійкість ринку IoT, спричинені технологічним прогресом і стратегічними змінами. Зміщення гіпермасштабувальників у бік периферійних і контейнерних стратегій, інтеграція штучного інтелекту в промислову автоматизацію, поява генеративного штучного інтелекту у виробництві та зростання рішень для обробки даних представляють лише деякі з динамічних подій, які переосмислюють корпоративну екосистему IoT. Ці тенденції, поряд з іншими ринковими даними у звіті, не лише відображають поточний стан ринку, але й дають уявлення про майбутнє зростання та інновації. Орієнтуючись у цьому ландшафті, для компаній вкрай важливо залишатися поінформованими та адаптуватися. З прогнозованим CAGR у 17% до 2030 року потенціал для зростання та трансформації в секторі Інтернету речей є величезним.

						ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата			18

2.2 Обґрунтування вибору засобів для побудови системи та мови програмування

Python – динамічна інтерпретована об'єктно-орієнтована скриптова мова програмування із строгою динамічною типізацією. Офіційний сайт мови програмування Python <https://www.python.org/>. Python – багатоцільова мова програмування, яка дозволяє писати код, що добре читається. Відносний лаконізм мови Python дозволяє створити програму, яка буде набагато коротше свого аналога, написаного на іншій мові. Python – багатоплатформова мова програмування. Це означає, що програми на Python можна запускати в різних операційних системах без будь-яких змін. Ще однією перевагою Python є його стандартна бібліотека, яка встановлюється разом з Python і містить готові інструменти для роботи з операційною системою, веб-сторінками, базами даних, різними форматами даних, для побудови графічного інтерфейсу програм тощо. Програми, написані на мові програмування Python, можуть бути як невеликими скриптами, так і складними системами. Python абсолютно безкоштовний.

Швидкість виконання коду Python

Один з можливих недоліків Python – швидкість виконання коду. Python не є компільованою мовою. Код на Python спочатку компілюється у внутрішній байт-код, який потім виконується інтерпретатором Python. У більшості випадків при використанні Python виходять програми повільніші в порівнянні з такими мовами, як С. Втім, сучасні комп'ютери мають таку обчислювальну потужність, що для більшості застосунків швидкість розробки важливіша швидкості виконання, а програми на Python зазвичай пишуться набагато швидше. Окрім того, Python легко розширюється модулями, написаними на С або С++. Такі модулі можуть використовуватися для виконання частин програми, що створюють інтенсивне навантаження на процесор.

Використання Python

Python використовується для різних цілей: для створення ігор і веб-застосунків, розробки внутрішніх інструментів для різноманітних проектів. Мова

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		19

також широко застосовується в науковій області для досліджень і розв'язування прикладних завдань. Застосування мови програмування Python:

1. BitTorrent – протокол для обміну даними.
2. Ubuntu Software Center – вільне програмне забезпечення для пошуку, установки і видалення пакунків в системі Ubuntu Linux.
3. Blender – програма для створення тривимірної комп'ютерної графіки, що включає засоби моделювання, анімації, вимальовування, пост-обробки відео, а також створення відеоігор.
4. GIMP – растровий графічний редактор, із підтримкою векторної графіки.
5. World of Tanks.
6. Вільна енциклопедія Вікіпедія.
7. Пошукова система Google.
8. DropBox – файловий хостинг, що включає персональне хмарне сховище, синхронізацію файлів і програму-клієнт.
9. YouTube – популярне відеосховище.
10. ...

Версії Python

Мови програмування з часом змінюються – розробники додають в них нові можливості, а також виправляють помилки. Так з'являються різні версії мови. Наприклад, код написаний на Python 2 у більшості випадків не буде працювати у версії Python 3 без внесення додаткових змін.

Процесор є найважливішим компонентом в комп'ютері. Одна з основних функцій процесора – це обробка даних згідно комп'ютерної програми, яка є списком інструкцій, шляхом виконання арифметичних і логічних операцій над фрагментами даних.

Кожна інструкція в програмі – це команда, яка «повідомляє» процесору, яку операцію він повинен виконати. Процесор комп'ютера може розуміти лише ті інструкції, які написані на машинній мові. Машинна мова – це штучна мова, створена для передачі команд комп'ютеру. За допомогою машинної мови

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		20

створюються ефективні програми, оскільки розробник отримує доступ до всіх можливостей процесора. Машинна мова – мова низького рівня.

Інструкція машинної мови існує для кожної операції, яку процесор здатний виконати – є інструкція для додавання чисел, є інструкція для віднімання чисел і т.д. Увесь набір інструкцій, який центральний процесор може виконати, відомий як набір інструкцій процесора.

Наприклад, у вас є певна програма, яка зберігається на диску вашого комп'ютера. Для виконання програми, ви здійснюєте подвійний клік на значку програми. Це змушує програму копіюватися з диска в оперативну пам'ять, після чого процесор комп'ютера виконує копію програми, яка знаходиться в оперативній пам'яті.

Коли процесор виконує інструкції програми, він бере участь у процесі, який є відомим як цикл `fetch – decode – execute` (отримати – декодувати – виконати). Цей цикл виконується для кожної інструкції у програмі і складається з трьох кроків:

Отримати

Програма – це послідовність інструкцій на машинній мові. Першим кроком циклу є завантаження (отримання) наступної інструкції з пам'яті в процесор.

Декодувати

Інструкція машинної мови – це двійкове число, яке представляє команду, що повідомляє процесору виконати певну операцію. На цьому кроці процесор декодує інструкцію, яку було «витягнуто» з пам'яті, для визначення того, яка операція повинна виконуватись.

Виконати

Останній крок циклу – виконати операцію.

Хоча процесор комп'ютера розуміє тільки машинну мову, людині непрактично писати програми на машинній мові. Така програма може мати

тисячі або навіть мільйони бінарних інструкцій, і написання такої програми буде дуже обтяжливим процесом.

З цієї причини була створена мова асемблера як альтернатива машинній мові. Замість використання двійкових чисел для написання інструкцій, мова асемблера використовує короткі слова, відомі як мнемокоди.

Незважаючи на те, що мова асемблера не вимагає двійкових інструкцій, як у випадку машинної мови, проте вона вимагає високих знань про процесор. Використовуючи мову асемблера, навіть для найпростішої програми, необхідно написати велику кількість інструкцій.

Мова програмування високого рівня дозволяє створювати складні програми, не знаючи, як працює процесор, і не записуючи великої кількості інструкцій низького рівня. Крім того, більшість мов програмування високого рівня використовують слова, які легко зрозуміти.

Python – одна із популярних сучасних мов програмування високого рівня. Python – інтерпретована мова програмування. Python – це високорівнева інтерпретована мова програмування, на відміну від C++, яка є прикладом компільованої мови програмування. Назва Python відноситься як до мови програмування, так і до інтерпретатора – комп'ютерної програми, яка зчитує початковий код (написаний на Python) і виконує інструкції (команди).

Для перекладу мови високого рівня на машинну мову доступні два типи програм:

1. Компілятор.
2. Інтерпретатор.

Завантаження Python

Версії інтерпретатора Python для різних операційних систем доступні для безкоштовного завантаження за адресою <https://www.python.org/downloads>.

Середовище програмування для Python

Для написання програм використовують текстові редактори або інтегровані середовища розробки, які включають в себе різні інструменти для

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		22

роботи з кодом: засіб для написання коду (текстовий редактор), інтерактивний інтерпретатор, відлагоджувач тощо.

Текстові редактори та інтегровані середовища програмування для Python:

– IDLE – стандартний редактор Python. Встановлюється разом з Python для користувачів Windows, окремим пакунком для користувачів Linux.

– Notepad++ – безкоштовний текстовий редактор початкового коду, який підтримує велику кількість мов, в тому числі і Python. Лише для користувачів Windows.

– Visual Studio Code – це легкий, але потужний редактор початкового коду, який розповсюджується безкоштовно і доступний у версіях для платформ Linux, Windows і macOS.

– PyScripter – інтегроване середовище розробки для мови програмування Python. Для користувачів Windows. Поширюється безкоштовно.

– Wing IDE 101 – вільне інтегроване середовище для Python, розроблене для навчання програмістів-початківців. Для користувачів Linux, Windows і macOS. Поширюється безкоштовно.

– Geany – вільний текстовий редактор з базовими елементами інтегрованого середовища розробки, доступний для операційних систем Linux, Windows і macOS.

– PyCharm – інтегроване середовище розробки для мови програмування Python. PyCharm є власницьким програмним забезпеченням. Наявна безкоштовна версія Community з усіченим набором можливостей. Для користувачів Linux, Windows і macOS.

– Thonny – IDE для вивчення програмування мовою Python. Для користувачів Linux, Windows і macOS.

– Mu – редактор коду Python для програмістів-початківців. Для користувачів Linux, Windows і macOS.

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		23

2.3 Розгорнута постановка завдання

Згідно з технічним завданням на випускню кваліфікаційну роботу за другим (магістерським) рівнем вищої освіти, реалізації підлягає програмне забезпечення, яке призначено для системи керування інфраструктурою на основі Cisco Digital Ceiling.

В процесі розробки випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти необхідно виконати наступний обсяг роботи:

а) провести аналіз існуючих систем-аналогів для виявлення їх позитивних і негативних якостей. Результати аналізу врахувати в подальших розробках;

б) вибрати та обґрунтувати методику побудови системи контролю роботи технологічного обладнання на виробництві в автоматизованому режимі. Розробити функціональну та структурну схеми системи;

в) розробити програмне забезпечення системи, що дозволить реалізувати поставлену технічним завданням задачу. Побудувати блок-схеми алгоритмів програми та підпрограми;

г) організувати інтерфейс користувача з метою формування та виводу на екран ЕОМ повідомлень про некоректні дії користувача та нестандартні ситуації в роботі технологічного обладнання;

д) розробити рекомендації по організаційних та методичних заходах, які забезпечать впровадження системи в промислову експлуатацію та її подальшу успішну експлуатацію;

е) провести розрахунки по визначенню економічної ефективності розробленої системи;

ж) розробити заходи по охороні праці при впровадженні та експлуатації системи, а також розробити заходи з цивільного захисту;

з) сформулювати висновки про виконаний обсяг робіт та одержані результати.

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		24

3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ

3.1 Опис функціонування системи

Cisco Digital Ceiling, яка має на меті об'єднати багато мереж у будівлі, не працювала б без партнерів-постачальників роз'ємів і кабелів.

Оскільки як підключення, так і кабельні рішення відіграють таку важливу роль у мережах, Cisco співпрацює з кількома постачальниками рішень, включаючи Siemon і Molex, обидва з яких виробляють з'єднувальні та кабельні технології, які можна використовувати в інсталяціях Digital Ceiling.

Digital Ceiling спрямований на об'єднання раніше розрізнених будівельних систем і послуг в єдину мережу Інтернет-протоколу. За словами Cisco, завдяки об'єднанню систем, а також датчиків і виконавчих механізмів на одній платформі рішення покращує ефективність і стійкість будівлі та дозволяє власникам будинків ефективніше керувати ними та контролювати їх. Мешканці будівель можуть використовувати платформу для керування системами освітлення, опалення та охолодження, де вони живуть, незалежно від того, чи це кімната чи квартира в будівлі.

Розглянемо сценарій на робочому місці, де збирання денного світла, автоматичне керування затемненням і освітлення на основі присутності заощаджують енергію та зменшують витрати. Уявіть собі середовище роздрібною торгівлі, де аналітика трафіку Digital Ceiling використовується для покращення макетів магазинів, розміщення товарів, розподілу персоналу та безпеки.

За даними Cisco, у середовищі роздрібною торгівлі вдосконалення освітлення та вивісок може призвести до збільшення доходу від продажів приблизно на сотні доларів за квадратний метр.

Рішення також забезпечує економію енергії власникам будівель. Digital Ceiling може адаптувати освітлення до реальних потреб користувачів, що може

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		25

призвести до потенційної економії енергії в розмірі 1,08 доларів на квадратний метр на рік у комерційних приміщеннях.

Ключовою частиною Digital Ceiling є системи та пристрої, які живляться від мережевих комутаторів за допомогою Power-over-Ethernet (PoE), включаючи світлодіодні світильники з живленням PoE і датчиками, які можуть допомогти знизити витрати на електроенергію до 85%. (Power-over-Ethernet дозволяє мережевим кабелям передавати електроенергію.)

Cisco назвала Siemon партнером Digital Ceiling для свого рішення ConvergeIT Cabling Solution, яке підтримує системи з підтримкою Power-over-Ethernet. Структурована кабельна система об'єднує дані, голос, відео та системи низької напруги в будівлі в єдиній уніфікованій фізичній інфраструктурі, що забезпечує економію коштів і стабільність протягом усього терміну служби об'єкта. Siemon довгий час був експертом і прихильником інтегрованих систем через єдине уніфіковане кабельне рішення, яке поєднує живлення та контроль будівельних пристроїв і освітлення для скорочення витрат, економії енергії та покращення загального контролю, управління та безпеки будівлі.

Molex, ще один партнер Digital Ceiling, пропонує портфоліо кабелів, датчиків, шлюзів PoE та освітлювальних приладів, які можна використовувати в рамках реалізації Digital Ceiling. До них відноситься система світлодіодного освітлення Molex Transcend Network Connected, яка використовує структурований IP-кабель або бездротове підключення.

Система світлодіодного освітлення, підключена до мережі, пропонує набагато більше, ніж нові світильники. Інтегровані мережеві рішення, такі як система Transcend, – це повні системи управління світлодіодним освітленням, які можна відстежувати та контролювати для різних будівельних середовищ за допомогою даних датчиків».

У міру розробки нових світлодіодних рішень архітектори, дизайнери освітлення, забудовники та власники отримують доступ до технології, яка значно

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		26

зменшує складність і є менш дорогою для встановлення та розгортання, ніж традиційні світлодіодні продукти змінного струму.

Крім того, підключені світлодіодні системи пропонують таке ж або більше зниження енергії, ніж світлодіодні продукти змінного струму.

Водночас, Siemon рекомендує дизайн зонної кабельної розводки для Digital Ceiling і пропонує вказівки та рішення для реалізації цього підходу. Зональна кабельна розводка складається з горизонтальних кабелів, що проходять від комутаторів Cisco UPOE у телекомунікаційних приміщеннях до проміжних точок з'єднання, розташованих в одному з корпусів зональних блоків Siemon, розташованих у стелі. З корпусів зональних блоків світлодіодні світильники та пристрої підключаються за допомогою коротших з'єднань, які прості в управлінні.

Це підтримує швидшу реорганізацію пристроїв з меншими збоями, що особливо важливо, враховуючи величезну кількість світлодіодних ліхтарів і IP-пристроїв, які можуть проникати в стелю приміщення.

З зональним кабелем зміни обмежуються коротшим сегментом кабелю між пристроєм і корпусом зонного блоку, а не всією довжиною горизонтального кабелю від комутатора в телекомунікаційній кімнаті.

Метод кабелю також спрощує розгортання нових пристроїв у програмі Digital Ceiling, вимагаючи лише короткого підключення до корпусу зонного блоку.

Перед Інтернетом речей стоїть ряд проблем, які можуть перешкодити нам скористатися його потенційними перевагами. Постійні повідомлення про злом підключених до Інтернету пристроїв, проблеми ведення спостереження й побоювання відносно особистої конфіденційності вже привернули увагу громадськості. На справжній момент технічні питання продовжують залишатися невирішеними, а також виникають нові складності в області політики, законодавства й подальшого розвитку.

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		27

Разом зі стрімким збільшенням числа підключених пристроїв і проникненням IoT в усі нові області ростуть і вимоги до безпеки. Більше того, як вважають в Cisco, саме забезпечення безпеки – ключова умова цифровізації бізнесу. Відповідно до проведеного компанією дослідження, близько 60% організацій не готові до розгортання цифрових продуктів і сервісів саме через високий ризик, пов'язаного з несанкціонованим одержанням інформації або доступом до IT-систем. При цьому практично всі опитані (96%) зв'язують успіх впровадження підключених пристроїв і використання нових бізнес-моделей із забезпеченням безпеки. Cisco пропонує комплексне рішення завдань безпеки з уніфікованим керуванням на основі інтегрованої архітектури, що охоплює мережа, хмари й кінцеві крапки. Однією з новинок, представлених на CiscoLive 2024, став міжмережний екран нового покоління Firepower 4100 Series. Головною перевагою цього високопродуктивного пристрою (продуктивність до 80 Гбіт/с) з інтерфейсами 10G і 40G, виконаного в конструктиві висотою 1RU, є саме реалізація заявленого інтегрованого підходу з підтримкою великої кількості сервісів безпеки. Зокрема, у цьому міжмережному екрані реалізований захист від вторгнень нового покоління (NGIPS), контроль додатків, фільтрація трафіку по URL, захист від розподілених атак DDo (Radware DefensePro DDo) і шкідливих програм (Advanced Malware Protection, AMP). Керування всім цим багатим функціоналом здійснюється через єдиний уніфікований інтерфейс. Поява подібного рішення, безумовно, повинна порадувати замовників, які вже утомилися від необхідності накопичення великого числа «коробка», кожна зі своєю системою керування.

3.2 Розробка структурної схеми

Якщо IT-інфраструктура, включаючи рішення по безпеці, – традиційна для Cisco область, де компанія завжди була сильна, то зараз вона прагне зайняти лідируючі позиції й в області платформ, у першу чергу хмарних, для розробки й

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		30

запуску нових додатків. У цілому Cisco націлена на надання не окремих продуктів, а готових рішень, які просто впроваджувати й використовувати. Для цього розгорнута масштабна партнерська програма DevNet, що нараховує вже близько 370 тис. учасників. Прекрасно розуміючи, що, для того щоб встигнути за інноваціями, своїх сил явно недостатньо, Cisco активно залучає сторонні таланти. Із цією метою організуються центри інновацій, подібні тому, що організовано в Берліні – openBerlin. У ньому є все необхідне для технічної творчості – від гарного кава до потужної ІТ-інфраструктури (у розгорнутому поруч контейнерному ЦОДі). Причому обстановка в центрі неформальна. Сам будинок центра, хоча і є досить старим, завдяки зусиллям Cisco і її партнерів перетворене в інтелектуальне – зокрема, у ньому розгорнута «розумна» система освітлення, регульована із планшетного комп'ютера.



Рисунок 3.1 – Структурна схема системи

Але якщо в офісі openBerlin для подачі електроживлення до світлодіодних світильників використовується класична силова проводка, то для анонсованого на форумі CiscoLive рішення Digital Ceiling така проводка вже не потрібна:

електроживлення подається за допомогою технології PoS по тій же мережі Ethernet, по якій здійснюється збір даних. Як відзначають експерти Cisco, система Digital Ceiling відкриває можливість конвергенції інженерних систем будинку, включаючи освітлення, опалення, охолодження, безпеку й т.д. Зокрема, рівень потужності будь-якого світильника може автоматично регулюватися залежно від присутності або відсутності в кімнаті людей, рівня зовнішньої освітленості (з вулиці) і т.д. Дані, що збираються датчиками, можуть використовуватися для налаштування параметрів систем опалення, кондиціонування й вентиляції, у роботі систем безпеки й т.д. Результат – зниження витрат при одночасному підвищенні рівня комфорту й безпеки для людей.

Проект Digital Ceiling уже залучив 15 партнерів, і їхнє число продовжує рости. У нього входять, зокрема, такі найвідоміші у своїх областях компанії, як Philips (системи освітлення), Eaton (системи електроживлення), Johnson Controls (засобу автоматизації будинків). У цілому перехід на світлодіодне освітлення дозволяє скоротити витрати на 40%, а при використанні інтелектуальної системи керування – на 70–80%. Як приклад він привів проект, реалізований у штаб-квартирі компанії Alliander, оператора мереж подачі електрики й газу в Голландії. У цьому офісі на площі приблизно 4000 м² встановлено близько 2200 LED-світильників, підключених до мережі Ethernet на базі встаткування Cisco. Поруч із кожним світильником розташовані датчики, які дозволяють, зокрема, визначати число людей у даній кімнаті. Кожний користувач має можливість зі свого смартфона відслідковувати рівень освітленості й регулювати його. Експлуатаційні служби будинку одержують точну статистику про наявність людей у всіх приміщеннях будинку, що дозволяє більш ефективно використовувати наявні площі. Загальне скорочення витрат на електрику в даному проекті склало близько 70%.

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		32

3.3 Розробка функціональної схеми

Функціональна схема розробленої системи зображена на рисунку 3.2. Існує не занадто багато способів забезпечення системи керування інфраструктурою на основі Cisco Digital Ceiling. Найпростіший з них – збільшення смуги пропускання мережі за рахунок нарощування апаратних можливостей устаткування. Можна використовувати й такі прийоми, як завдання пріоритетів даних, організація черг, запобігання перевантажень і формування трафіку. Керування мережею за заданими правилами в перспективі повинне об'єднати всі ці способи в єдину автоматизовану систему, що буде гарантувати якість послуг абсолютно на всіх ділянках мережі.

Збільшення апаратної потужності, безсумнівно, є найбільш ефективним засобом реалізації системи керування інфраструктурою на основі Cisco Digital Ceiling у локальній мережі. Тиск із боку конкурентів, необхідність підвищення ефективності виробництва, поява нових технологій, що дозволяють оснащувати спеціалізовані мікросхеми (ASIC) найрізноманітнішими функціями, – все це змушує постачальників комутаційного встаткування для локальних мереж викидати на ринок усе більше швидкодійчі пристрої за цінами, порівняним з вартістю моделей колишнього покоління.

Малоймовірно, що в доступному для огляду майбутньому даний підхід до підтримки системи керування інфраструктурою на основі Cisco Digital Ceiling у локальних мережах перестане бути пріоритетним. Оскільки в локальних мережах вдається забезпечити гарантовану якість послуг, не прибігаючи до дорогої модернізації усього встаткування й серйозних змін у системі керування мережею, мережні адміністратори будуть звертати увагу й на програмні засоби, що дозволяють реалізувати системи керування інфраструктурою на основі Cisco Digital Ceiling.

Отже, найбільше поширення, швидше за все, одержить комбінований підхід. Деякі виробники висловлюються на його користь, затверджуючи, що найкраще збільшувати пропускну здатність мережі не прямо, а за рахунок

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		33

інтелектуальних можливостей устаткування, що має засоби забезпечення системи керування інфраструктурою на основі Cisco Digital Ceiling. Правда, виробники мережних пристроїв навряд чи можуть бути об'єктивними в цьому питанні, тому що вони зацікавлені в збуті тих самих продуктів, які підтримують гарантовану якість послуг. У глобальних мережах нарощування апаратних потужностей використовується рідше. Звичайно, зниження вартості смуги пропускання зробило б передачу даних по глобальних мережах доступною для більше широкого кола користувачів (і навіть трохи знизило б актуальність впровадження гарантованої якості послуг). Але в найближчому майбутньому вартість смуги пропускання в глобальних мережах буде залишатися досить високою, тому й нарощування апаратної потужності не стане настільки популярним, як у локальних мережах.

З рисунку видно, що розроблена система складається з наступних функціональних частин:

– Блок керування завантаженістю мережі з використанням системи керування інфраструктурою на основі Cisco Digital Ceiling.

– Блок моніторингу мережі.

– Блок дослідження можливостей механізмів WRED.

– Блок дослідження можливостей механізмів WFQ.

– Блок призначення пріоритетів.

– Блок організації та обслуговування черг.

– Блок управління навантаженням.

– Блок формування трафіку.

– Блок інтерфейсу користувача.

– Блок визначення параметрів системи керування інфраструктурою на основі Cisco Digital Ceiling.

– Блок примусового завдання параметрів системи керування інфраструктурою на основі Cisco Digital Ceiling.

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		34

трафіком та заданими параметрами якості обслуговування (системи керування інфраструктурою на основі Cisco Digital Ceiling).

Блок моніторингу мережі

Призначений для аналізу поточного стану мережі.

Блок реалізації механізмів WRED

Одним з методів системи керування інфраструктурою на основі Cisco Digital Ceiling, призначених для забезпечення необхідних вимог до різних потоків даних – запобігання перевантажень (congestion avoidance). Він заснований на обмеженні розмір черги, сигналізуючи джерелам даних про необхідність зменшити швидкість передачі інформації (WRED – Weighted random early detection).

Блок реалізації механізмів WFQ

Другим з методів системи керування інфраструктурою на основі Cisco Digital Ceiling, призначених для забезпечення необхідних вимог до різних потоків даних – керування перевантаженням (congestion management). Він заснований на присвоєнні квот і пріоритетів потокам, і у випадку перевантаження, потоки одержують якість, обмежену їхньою квотою й пріоритетом (WFQ – Weighted Fair Queuing).

Блок призначення пріоритетів

Нарівні з нарощуванням апаратного забезпечення мережі для реалізації системи керування інфраструктурою на основі Cisco Digital Ceiling застосовуються й засобу типу завдання пріоритетів даних і організації черг. Маршрутизатори підтримують ці механізми протягом багатьох років, як і деякі з нових комутаторів для каналів Gigabit Ethernet. Однак ПЗ для керування мережею за заданими правилами, яких необхідно для практичного втілення цієї технології, поки не розроблено. Серед нових комутаторів такого класу можна назвати CoreBuilder 3500, CoreBuilder 9000 і SuperStack II компанії 3Com, пристрою серії Accelar фірми Bay Networks, SmartSwitch Router компанії Cabletron Systems, а також Catalyst 5000 і Catalyst 8000 виробництва Cisco.

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		36

Способи пріоритезації даних можна умовно підрозділити на явні й неявні.

При неявному призначенні пріоритетів маршрутизатор або комутатор автоматично привласнює послугам відповідні рівні, виходячи із заданих адміністратором мережі критеріїв (наприклад, типу додатка для застосовуваного протоколу передачі або адреси джерела). Кожний вхідний пакет аналізується (фільтрується) на відповідність цим критеріям. Механізм неявної пріоритезації підтримують практично всі маршрутизатори.

Деякі комутатори теж здатні задавати пріоритети, але мають обмежений набір функцій. Так, комутатори можуть забезпечувати пріоритезацію даних по типу віртуальної локальної мережі, адресі джерела або адресата, але не використовують інформацію більш високого рівня (протокол передачі або тип додатка). Розроблювальні в цей час системи керування мережею за заданими правилами дозволяють реалізувати більше зроблені схеми пріоритезації даних при роботі з такими комутаторами.

При явній пріоритезації даних користувач або додаток запитує певний рівень служби, а комутатор або маршрутизатор намагається задовольнити запит. Імовірно, самим популярним механізмом явної пріоритезації стане протокол IP Precedence (протокол старшинства), що одержав другу назву IP TOS (IP Type Of Service), – один з розділів четвертої версії протоколу IP.

IP TOS резервує спеціальне поле в заголовку пакета, де можуть бути зазначені ознаки системи керування інфраструктурою на основі Cisco Digital Ceiling, що визначають час затримки, швидкість пропущення й рівень надійності передачі пакета. Однак знайдеться небагато популярних додатків – за винятком мультимедійного ПЗ, – у які реалізована підтримка протоколу IP TOS.

Зараз розробляється протокол резервування ресурсів RSVP, що передбачає більше складний, чим в IP TOS, механізм передачі від додатка до маршрутизатору запиту на гарантовану якість послуг. Як і IP TOS, протокол RSVP поки не одержав широкої підтримки розроблювачів – він реалізований лише в окремих типах маршрутизаторів. Поширення RSVP стримується через те, що не вирішені

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		37

Якщо в мережі виникає перевантаження, служба черг не гарантує своєчасного досягнення пункту призначення найбільш важливими даними. Гарантується лише те, що ці пакети будуть передані раніше, ніж ті, що мають більш низький пріоритет.

Сучасні служби системи керування інфраструктурою на основі Cisco Digital Ceiling вирішують таке завдання за рахунок резервування смуги пропускання. Кожній із черг (або їхніх груп) виділяється заздалегідь задана величина смуги пропускання, що гарантує певну смугу пропускання для черги з більш високим пріоритетом. Для критичних ситуацій, коли обсяг даних у черзі перевищує розміри смуги пропускання, в алгоритмах обслуговування звичайно передбачається передача трафіку з високим пріоритетом на смугу пропускання, “приналежну” чергам з низьким пріоритетом, і навпаки. Найпростіші алгоритми обслуговують кожен чергу за принципом FIFO. При цьому передача кадрів великого розміру, що мають високий пріоритет, може приводити до затримок трафіку іншого додатка з настільки ж високим пріоритетом, але меншим обсягом.

У більш складних алгоритмах уживає спроба “справедливої” обробки черг. Наприклад, алгоритм рівномірного пропорційного (або зваженого) обслуговування (WFQ – Weighted Fair Queuing), розроблений компанією Cisco, підрозділяє додатки на потребуючі великої й малої ширини смуги пропускання, а сама смуга пропускання розподіляється між всіма додатками нарівно. Слід зазначити, що основні виробники маршрутизаторів самі розробляють алгоритми обслуговування черг і використовують для їхнього опису власну термінологію.

Істотним недоліком сучасних маршрутизаторів і комутаторів є те, що вони підтримують мале число черг. Найчастіше виробники організують служби системи керування інфраструктурою на основі Cisco Digital Ceiling, що використовують чотири черги, хоча чим більше черг, тим більше різних пріоритетів можна привласнити переданим пакетам і тим “справедливіше” розподілити смугу пропускання між додатками. Наприклад, адміністратор у стані

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		39

задати пріоритети таким чином, щоб перевага при передачі віддавалося пакетам, адресованим на більше віддалені вузли.

Блок управління навантаженням

Служба системи керування інфраструктурою на основі Cisco Digital Ceiling дає можливість використовувати для керування мережею два важливих механізми – керування в умовах перевантаження й запобігання перевантажень. Перший з них дозволяє кінцевій станції відразу знижувати швидкість передачі даних, коли в мережі починається втрата пакетів. У протоколах TCP/IP і SNA цей механізм підтримується вже протягом декількох років. І хоча сам по собі він не гарантує якості передачі, при його використанні разом з механізмом запобігання перевантажень результати виявляються набагато кращими. У мережах TCP/IP механізм запобігання перевантажень застосовується досить давно, але лише в останні роки він стає стандартом “де-факто” для маршрутизаторів телекомунікаційних мереж і Internet.

Стандартним способом запобігання перевантажень у мережі стало застосування механізму випадкового виділення пакетів (Random Early Detection, RED). При заповненні черг вище певної критичної оцінки цей механізм змушує маршрутизатор вибирати із черги за випадковим законом деякі пакети й “втрачати” їх. Швидкість передачі даних станціями-відправниками знижується, що й дозволяє уникнути переповнення черги.

Механізм пропорційного випадкового виділення пакетів – WRED (Weighted RED) – можна вважати наступною, більше зробленою “версією” RED. Він передбачає, що вибір пакетів, які повинні “втратитися”, буде відбуватися з обліком їх пріоритезації згідно IP TOS.

Блок формування трафіку

Формування трафіку – це загальний термін, яким прийнято позначати різні способи маніпулювання даними для підвищення якості їхньої передачі. Один їхній таких способів – сегментація пакетів. У мережах АТМ гарантовано високий рівень системи керування інфраструктурою на основі Cisco Digital Ceiling досягається в тому числі й за рахунок малого розміру переданих пакетів

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		40

(осередків – у термінології АТМ). Максимальний час затримки при передачі будь-якого пакета мережі АТМ – це час передачі одного осередку. Запозичаючи корисні механізми технології АТМ, виробники маршрутизаторів і комутаторів починають забезпечувати у своїх продуктах можливість сегментації пакетів. Деякі пристрої, призначені для мереж frame relay, сегментують пакети, передані по каналах глобальних мереж, щоб гарантувати конкретний час передачі й мінімізувати затримки. Ще один спосіб формування трафіку – його “вирівнювання”. Для таких протоколів, як наприклад, AppleTalk, характерна нерівномірна передача пакетів, що часом приводить до появи в мережі послідовностей або ланцюжків пакетів, а отже – до її перевантаження. Процедура вирівнювання трафіку дозволяє розчленувати ланцюжки шляхом розміщення пакетів у буфері перед їхньою передачею в мережу. Для забезпечення більше рівномірної передачі даних можна також вирівнювати трафік кінцевих вузлів мережі.

Блок визначення параметрів системи керування інфраструктурою на основі Cisco Digital Ceiling

Призначений для визначення існуючих параметрів якості обслуговування (системи керування інфраструктурою на основі Cisco Digital Ceiling). До них відносяться:

– Bandwidth (BW) – смуга пропускання, описує номінальну пропускну здатність середовища передачі інформації, визначає ширину каналу. Вимірюється в bit/s (bps), kbit/s (kbps), mbit/s (mbps).

– Delay – затримка при передачі пакета.

– Jitter – коливання (варіація) затримки при передачі пакетів.

– Packet Loss – втрати пакетів. Визначає кількість пакетів, що відкидаються мережею під час передачі.

Блок примусового завдання параметрів системи керування інфраструктурою на основі Cisco Digital Ceiling

Призначений для примусового завдання одного, або декількох параметрів якості обслуговування (системи керування інфраструктурою на основі Cisco

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		41

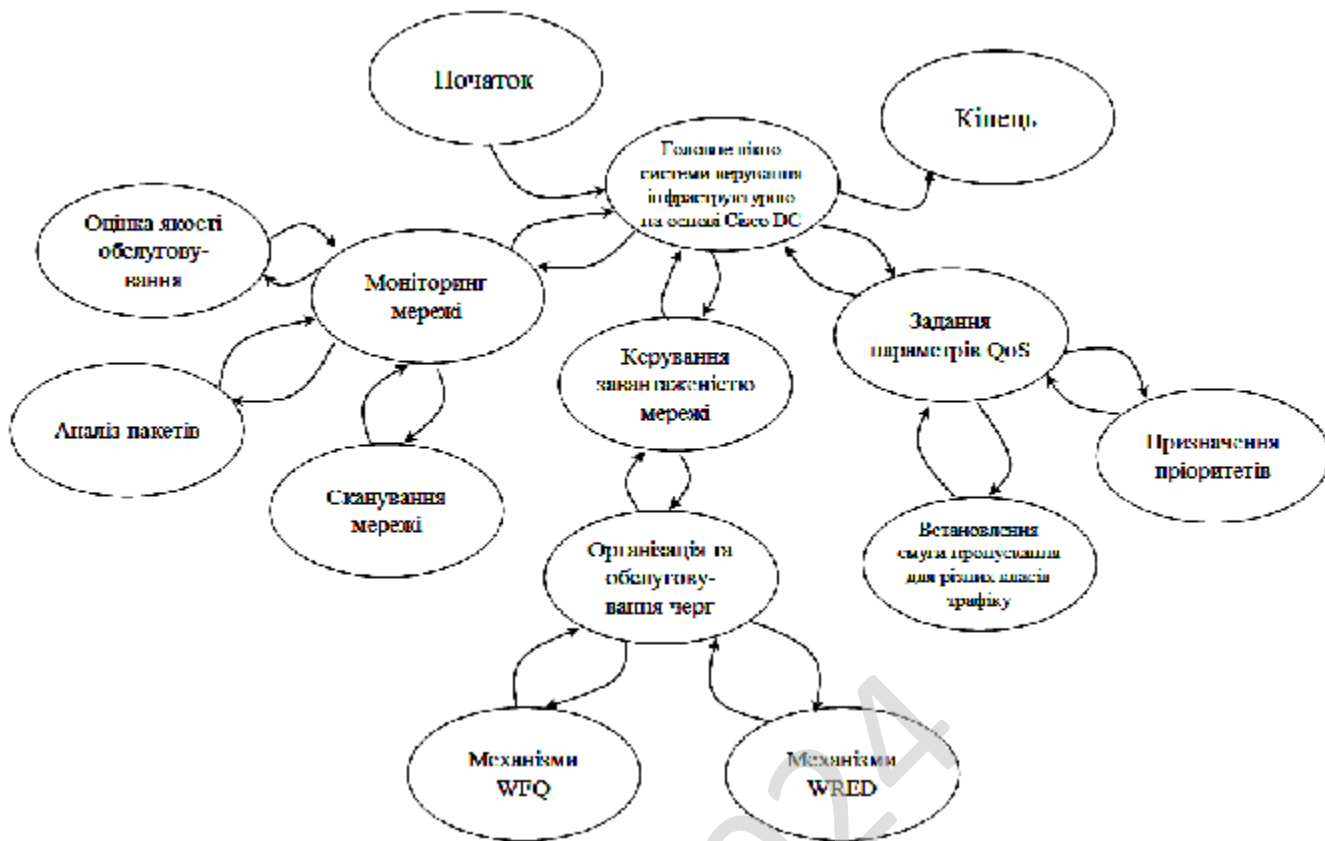


Рисунок 3.3 – Діаграма взаємодії процесів

Таким чином, розглянувши опис системи, структурну, функціональну схеми системи, та діаграму взаємодії процесів перейдемо до опису блок-схем основної програми, та підпрограм, які використовуються, для реалізації системи.

4 РЕАЛІЗАЦІЯ ПРОЕКТУ. РОЗРАХУНКИ І ЕКСПЕРИМЕНТАЛЬНІ ДАНІ, ЩО ПІДТВЕРДЖУЮТЬ ПРАВИЛЬНІСТЬ ПРОЕКТНИХ РІШЕНЬ

4.1 Блок-схеми та опис алгоритмів функціонування системи

Блок-схеми є основою ПЗ. Тому від точності і детальності проробки блок-схеми залежить результат всієї програми.

При виборі початкової точки відліку при побудові схем було враховано, що виходячи з вибору мови програмування і інших технічних засобів, програма буде об'єктно-орієнтована що вимагає оптимізації, також те, що при розробці програми слід надати особливу увагу модулю системи керування інфраструктурою на основі Cisco Digital Ceiling.

Функціональні блоки на схемі позначають прямокутниками, всередині яких надписують їх найменування відповідно до функцій, що виконуються. Зв'язки між функціональними блоками (внутрішні впливи) позначаються лініями зі стрілками, які вказують напрям впливів.

Функціональні блоки можуть виконуватися в укрупненому і розгорненому вигляді. У першому випадку на схемі зображають найважливіші блоки системи і зв'язки між ними.

На рисунку 4.1 зображена основна блок-схема програми, на рисунку 4.2 зображено роботу підпрограми.

З яких видно що робота основної програми складається з початкових етапів ініціалізації ПЗ, перевірки наявності ресурсів системи, блоку початку основного циклу з чеканням запиту від користувача в якому відбувається виклик підпрограми та останньої стадії – перевірки поточного стану та поверненням на початок схеми чи з завершенням роботи розробленого ПЗ.

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		44

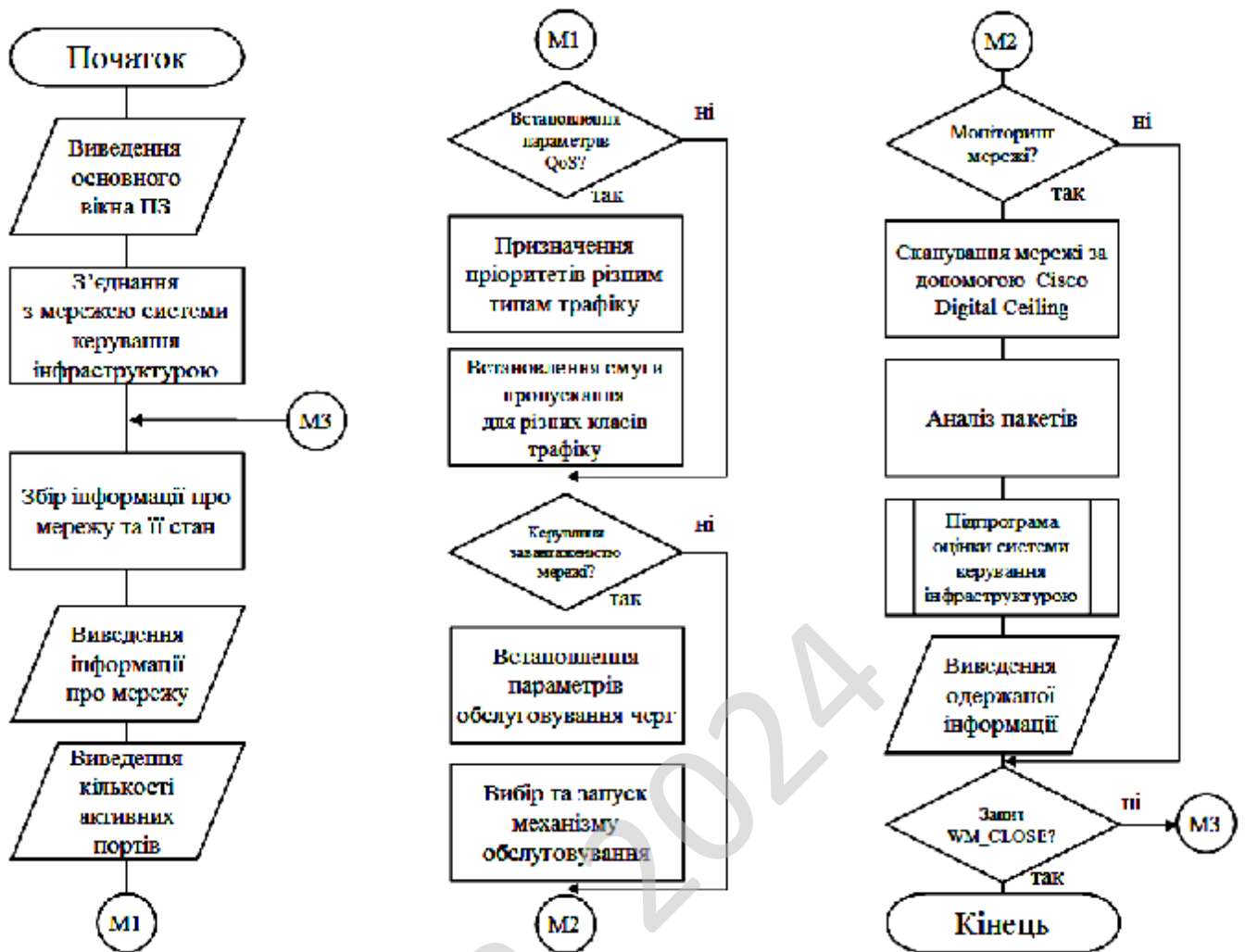


Рисунок 4.1 – Блок-схема основної програми

При роботі підпрограми виконується основний функціонал системи з циклічними послідовностями, перевіркою поточного стану та поверненням в основну програму прапорів стану виконання.

Було використано підходи з використанням UML, це уніфікована мова моделювання, використовується у парадигмі об'єктно-орієнтованого програмування. Є невід'ємною частиною уніфікованого процесу розробки програмного забезпечення. UML є мовою широкого профілю, це відкритий стандарт, що використовує графічні позначення для створення абстрактної моделі системи, називаної UML-моделлю. UML був створений для визначення, візуалізації, проектування й документування в основному програмних систем.

UML не є мовою програмування, але в засобах виконання UML-моделей як інтерпретованого коду можлива кодогенерація.

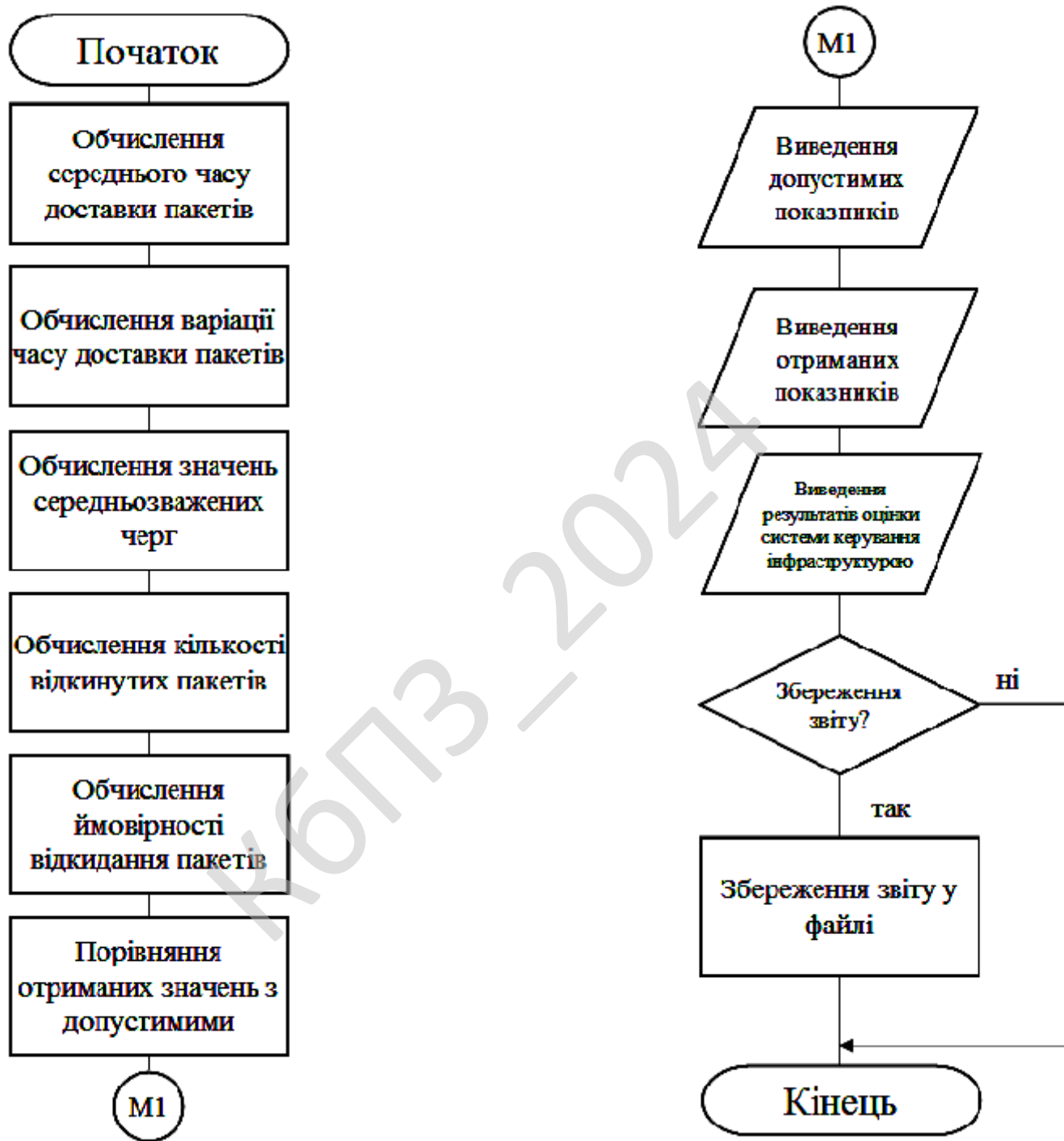


Рисунок 4.2 – Блок-схема роботи підпрограми

Розглянемо використані технології та їх основні компоненти що підтверджують правильність використаних проектних рішень.

Була використана водоспадна (каскадна) модель життєвого циклу ПЗ (waterfall model) – послідовний метод розробки програмного забезпечення, названий так через діаграму схожу на водоспад.

Ця модель розробки запозичена з системної інженерії у виробництві та будівництві – областях, в яких зміни на пізніх етапах дуже дорогі, або неможливі. Наприклад, для створення складних інженерних конструкцій (споруд, літаків, мостів і т.п.). Зміни в проекті фундаменту будинку після того, як покладений дах коштують дуже дорого, тому перфекціонізм на початкових етапах проектування просто необхідний. Інженери, які починали займатись розробкою програмного забезпечення перейшовши з інших галузей, просто адаптували звичну модель, тому що на ранніх етапах розвитку комп'ютерної техніки не було методологій створених саме для програмування. Проте, схожі методології застосовуються для програмного забезпечення й далі, у випадках коли вимоги фіксовані, і вимагається висока якість та надійність, наприклад в системах для військових чи медичних потреб.

Перший формальний опис водоспадної моделі, після якої вона стала популярною був здійснений В. В. Ройсом у 1970. Попри те, що стаття містить переважно критику методу, на неї часто посилаються.

Переваги методу:

- Ніяких переробок.
- Гарна специфікація перетікає в гарну документацію.
- Зрозуміла модель.
- Розробники можуть мати низьку кваліфікацію.

Недоліки:

- Необхідний перфекціонізм на кожному етапі.
- Важко вносити зміни (якщо взагалі можливо).
- Надлишкове проектування.

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		47


```

# Модуль для розрахунку ефективності керування енергоспоживанням
class EnergyEfficiencyCalculator:
    def __init__(self, energy_data):
        self.energy_data = energy_data

    def calculate_efficiency(self):
# Виконуємо розрахунок енергоефективності на
# основі споживання та вказаних показників
        baseline_consumption = np.mean(self.energy_data["baseline"])
        actual_consumption = np.mean(self.energy_data["actual"])
        efficiency = (baseline_consumption - actual_consumption) /
baseline_consumption * 100
        return efficiency

# Модуль для керування температурою приміщень
class TemperatureController:
    def __init__(self, connection):
        self.connection = connection

    def set_temperature(self, room_id, temperature):
# Встановлюємо температуру в кімнаті
        cisco.set_temperature(self.connection, room_id, temperature)

    def monitor_temperature(self, room_id):
# Моніторимо поточну температуру в кімнаті
        return cisco.get_temperature(self.connection, room_id)

# Архітектура системи в цілому

def main():
# Ініціалізуємо контролер Cisco Digital Ceiling
    ceiling_controller = CiscoDigitalCeilingController("192.168.1.1", "admin",
"password")

# Отримуємо стан інфраструктури
    infrastructure_status = ceiling_controller.get_infrastructure_status()
    print("Infrastructure status:", infrastructure_status)

# Контролюємо освітлення в аудиторії
    ceiling_controller.control_lighting("room_101", 75)

# Моніторимо енергоспоживання та розраховуємо ефективність

```

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		49

```

energy_data = {
    "baseline": [100, 105, 102, 110],
    "actual": [95, 98, 96, 97]
}
energy_calculator = EnergyEfficiencyCalculator(energy_data)
efficiency = energy_calculator.calculate_efficiency()
print("Energy efficiency:", efficiency, "%")

# Керування температурою в аудиторії
temperature_controller = TemperatureController(ceiling_controller.connection)
current_temperature = temperature_controller.monitor_temperature("room_101")
print("Current temperature in room 101:", current_temperature)

# Встановлюємо комфортну температуру
temperature_controller.set_temperature("room_101", 22)

# Викликаємо головну функцію
if __name__ == "__main__":
    main()

```

Ця система представляє собою інтегровану платформу для керування інфраструктурою через Cisco Digital Ceiling, яка дозволяє контролювати освітлення, температуру та моніторити енергоспоживання. Основні розрахунки проводяться для підвищення енергоефективності, з використанням методів порівняння базового та фактичного споживання енергії.

При розробці ПЗ було використано підходи ризик-менеджменту – це система управління ризиками, яка включає в себе стратегію та тактику управління, направлені на досягнення основних цілей. Ефективний ризик-менеджмент включає:

- систему управління;
- систему ідентифікації і вимірювання;
- систему супроводження (моніторингу та контролю).

Сучасна наука представляє ризик як вірогідну подію, в результаті настання якої можуть відбутися позитивні, нейтральні або негативні наслідки. Якщо ризик припускає наявність як позитивних, так і негативних результатів, він відноситься до спекулятивних ризиків. Якщо ж наслідки негативні, або відсутні взагалі, такий ризик іменується чистим.

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		50

- на третьому етапі вибираються методи і інструменти управління виявленим ризиком;
- на четвертому етапі проводиться безпосереднє управління ризиком;
- на завершальному етапі оцінюються досягнуті результати і коректується ризик-стратегія.

За ключовий етап ризик-менеджменту вважається етап вибору методів і інструментів управління ризиком.

Методи і інструментарій ризик-менеджменту

Базовими методами ризик-менеджменту є відмова від ризиків, зниження, передача і ухвалення.

Ризик-інструментарій значно ширший. Він включає політичні, організаційні, правові, економічні, соціальні інструменти, причому ризик-менеджмент як система допускає можливість одночасного застосування декількох методів і інструментів ризик-управління.

Найбільш часто вживаним інструментом ризик-менеджменту є страхування. Страхування припускає передачу відповідальності за відшкодування передбачуваного збитку сторонній організації (страхової компанії).

Прикладами інших інструментів можуть бути відмова від надмірно ризикової діяльності (метод відмови), профілактика або диверсифікація (метод зниження), аутсорсинг витратних ризикових функцій (метод передачі), формування резервів або запасів (метод ухвалення).

4.2 Захист розробленого програмного забезпечення

Для захисту розробленого програмного забезпечення запропоновано використовувати алгоритм REDOC III, який оперує з 80-бітовим блоком. Довжина ключа може змінюватися й досягати 2560 байт (204800 біт). Алгоритм складається тільки з операцій XOR над байтами ключа й відкритого тексту, перестановки й підстановки не використовуються.

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		52

1. Створюють таблицю ключів з 256 10-байтових ключів, використовуючи секретний ключ.

2. Створюють два 10-байтових блоки масок M1 і M2. M1 являє собою результат операції XOR перших 128 10-байтових ключів, а M2 – результат операції XOR других 128 10-байтових ключів.

3. Для шифрування 10-байтового блоку:

a. Виконують операцію XOR з першим байтом блоку даних і першим байтом M1. Вибирають ключ у таблиці ключів, розрахованої в раунді 1. Використовують обчислене значення XOR як індекс таблиці. Виконують операцію XOR з кожним, крім першого, байтом блоку даних і відповідним байтом обраного ключа.

b. Виконують операцію XOR із другим байтом блоку даних і другим байтом M1. Вибирають ключ у таблиці ключів, розрахованої в раунді 1. Використовують обчислене значення XOR як індекс таблиці. Виконують операцію XOR з кожним, крім другого, байтом блоку даних і відповідним байтом обраного ключа.

c. Продовжують ці дії з усім блоком даних (з 3-10 байтами), поки не буде використаний кожний байт для вибору ключа з таблиці після виконання операції XOR з ним і відповідним значенням M1. Потім виконують операцію XOR з кожним, крім використаного для вибору ключа, байтом, і ключем.

d. Повторюють етапи а-с для M2.

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		53

5 МЕТОДИКА ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ

Розглянемо розроблене ПЗ системи керування інфраструктурою на основі Cisco Digital Ceiling яке зображено на рисунку 5.1. Процес розрахунків здійснюється в консольному додатку з передачею отриманих результатів у користувацький інтерфейс. З рисунку можна побачити що інтерфейс головного вікна розподілено на наступні функціональні розділи:

- Навігаційне меню: Файл; Дії; Параметри; Довідка.
- Розділу введення параметрів.
- Вікно виведення результату роботи системи.
- Навігаційного меню яке викликається натисканням правої клавіші маніпулятора миші.
- Функціональних кнопок ПЗ.

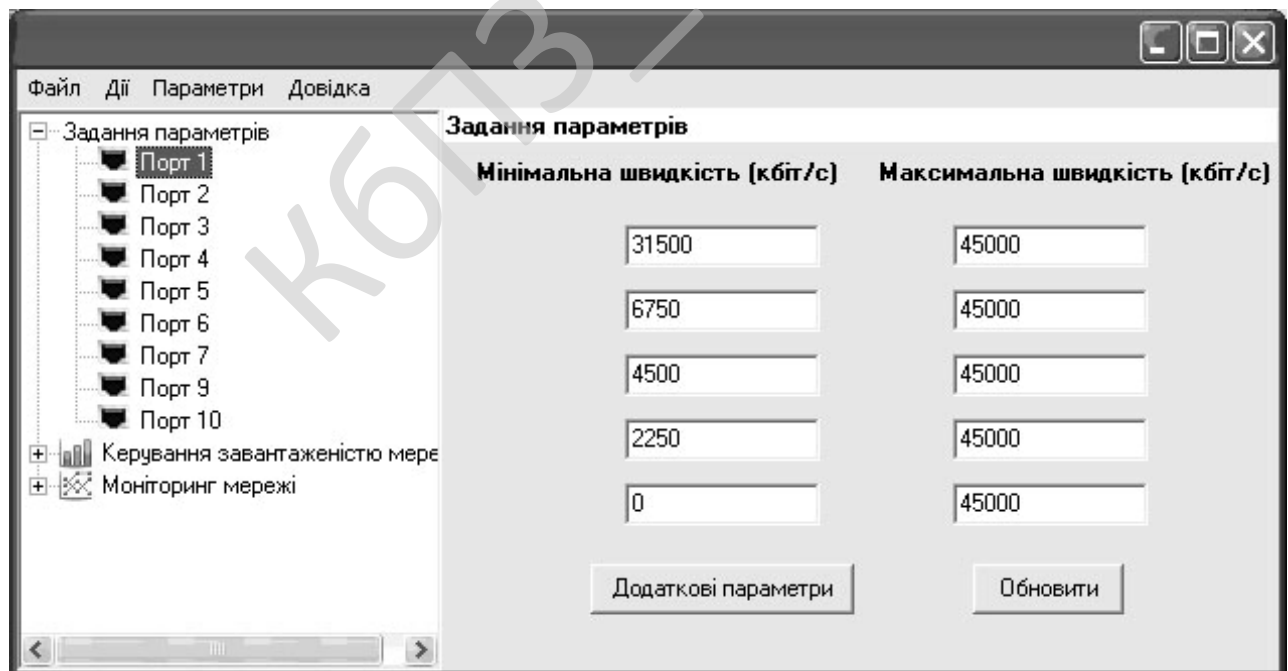


Рисунок 5.1 – Головне вікно ПЗ

Розроблена програма має дуже простий і інтуїтивно зрозумілий інтерфейс з користувачем. Кожен, хто в достатньому обсязі володіє операційним середовищем Windows без особливих складностей освоїть і цю програму, оскільки її інтерфейс інтуїтивно зрозумілий.

Якщо програма не видала ніяких помилок, і працює, то можна використовувати, інакше слід слідувати інструкціям, які пропонує програма.

На рисунку 5.2 зображено авторські дані розробленого програмного забезпечення.

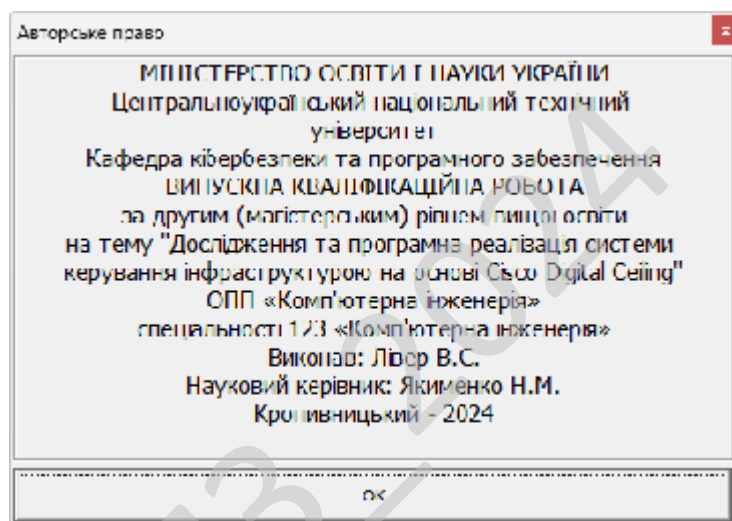


Рисунок 5.2 – Авторське право

Розглянемо процес впровадження програмного забезпечення, це процес налаштування програмного забезпечення під певні умови використання, а також навчання користувачів роботі з програмним продуктом. Впровадження програмного забезпечення це усі дії, що роблять розроблену програмну систему готовою до використання. Даний процес є частинною життєвого циклу програмного забезпечення.

Загалом процес розгортання складається з кількох взаємопов'язаних дій із можливими переходами між ними. Ця активність може відбуватися як з боку виробника так і з боку споживача. Оскільки кожна програмна система є

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		55

унікальною, то усі процеси та процедури під час розгортання важко передбачити. Тому, "розгортання" можна трактувати як загальний процес відповідно до певних вимог та характеристик. Розгортання може здійснюватись програмістом і в процесі розробки програмного забезпечення.

До діяльностей пов'язаних із розгортанням програмного забезпечення відносять:

- Випуск.
- Встановлення та активація.
- Деактивація.
- Адаптація.
- Обновлення.
- Вмонтування.
- Відстежування версій.
- Видалення.
- Вилучення з обігу.

При впровадженні програмного забезпечення потрібно урахувати наступні дії:

– Виділення критичних, з точки зору загального результату, процедур в діяльності організації. Коли набір таких процедур визначений, необхідно в першу чергу використовувати ІТ рішення для автоматизації операцій усередині саме цих процедур. Таким чином, розроблене ІТ рішення автоматично стає життєво важливим і затребуваним для організації, а також буде забезпечена публічність процесу впровадження;

– Розширення нормативної бази організації шляхом включення до неї регламентів, що описують порядок виконання процедур автоматизованих процесів. В іншому випадку є небезпека виникнення неузгодженості між автоматизованими процедурами та іншими процесами організації.

– Виконання робіт з загальної стандартизації існуючої діяльності організації, коли виділяються кращі практики виконання процедур і включаються

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		56

в IT рішення за принципом найбільшої корисності для більшості учасників. Відсоток таких процедур щодо загального обсягу автоматизації може бути невеликий, але це надає процесу побудови рішення вагу в організації за рахунок збільшення його необхідності.

Під час роботи над програмою було проведено тестування програмного забезпечення, тобто технічне дослідження, призначене для виявлення інформації про якість продукту відносно контексту, в якому воно має використовуватись.

Тестування включає як процес пошуку помилок або інших дефектів, так і випробування програмних складових з метою їх оцінки.

Проводилась оцінка:

- відповідності поставленим вимогам;
- правильна відповідь для усіх можливих вхідних даних;
- виконання функцій за прийнятний час;
- практичність;
- сумісність з ОС та стороннім ПЗ.

Оскільки число можливих тестів для програмних компонент практично нескінченне, тому стратегія тестування полягала в тому, щоб провести всі можливі тести з урахуванням наявного часу та ресурсів.

Як результат ПЗ тестувалось стандартним виконанням програми з метою виявлення помилок або інших дефектів.

Проводилось тестування форматом чорної скриньки. Основне місце програми тестів «чорної скриньки» – інтерфейс ПЗ. Відомі: функції програми. Досліджується: робота кожної функції на всій області визначення.

Ці тести демонструють:

- Як виконуються функції програми.
- Як приймаються вихідні дані.
- Як виробляються результати.
- Як зберігається цілісність зовнішньої інформації.

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		57

– Помилки ініціалізації та завершення.

Обрано умови розповсюдження – Freeware.

Це власницьке програмне забезпечення, котре можна Безоплатно використовувати протягом необмеженого терміну без обмежень у функціональності, і поширюване без сирцевих кодів.

Автори такого програмного забезпечення, як правило, хочуть «дати щось спільноті», але хочуть також контролювати його подальшу розробку. Іноді, коли програмісти вирішують припинити розробку, вони передають сирцевий код іншим програмістам, або ж спільноті як вільне програмне забезпечення.

Дуже часто плутають поняття «безплатне програмне забезпечення» та «вільне програмне забезпечення», хоча вони суттєво відрізняються.

Безплатне програмне забезпечення можна безоплатно встановлювати та використовувати (іноді з певними обмеженнями, як, наприклад, «безплатне для домашнього або некомерційного вжитку»), в той час як вільне програмне забезпечення можна продавати за будь-яку суму, але при тому, у користувача, котрий його отримує, повинні бути права на вивчення, модифікацію та поширення сирцевих кодів одержаної програми.

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		59

6 НАУКОВА НОВИЗНА

У випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти розроблено програмне забезпечення, яке призначено для системи керування інфраструктурою на основі Cisco Digital Ceiling.

Метою розробки є дослідження та програмна реалізація системи керування інфраструктурою на основі Cisco Digital Ceiling.

Об'єктом дослідження є процес керування інфраструктурою на основі Cisco Digital Ceiling.

Предметом дослідження є методи керування інфраструктурою на основі Cisco Digital Ceiling.

Методи дослідження базуються на методах теорії комп'ютерних мереж, методах математичної статистики, методах розробки програмного забезпечення.

Наукова новизна отриманих результатів. У процесі рішення завдань, обумовлених цілями дослідження, отримані наступні результати:

- Удосконалено метод керування інфраструктурою на основі Cisco Digital Ceiling.
- Розроблено вітчизняний продукт керування інфраструктурою на основі Cisco Digital Ceiling, який має більш широкі можливості, на відміну від існуючих аналогів.

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		60

7 МАРКЕТИНГОВЕ ТА ЕКОНОМІЧНЕ ОБГРУНТУВАННЯ ІТ-ПРОЄКТУ

7.1 Визначення цільової аудиторії кінцевого готового продукту

Результати дослідження та програмна реалізація системи керування інфраструктурою на основі Cisco Digital Ceiling можуть бути цікаві різноманітним організаціям та професіоналам, які зацікавлені в управлінні інтелектуальними будівлями та оптимізації інфраструктурних рішень. Ось основні зацікавлені сторони:

Девелопери та будівельні компанії	Можуть використовувати Cisco Digital Ceiling як частину колекції "розумних" будівель щоб підвищити провідність об'єктів нерухомості.
Компанії з управління нерухомістю	Застосування таких систем допомагає знизити експлуатаційні витрати, підвищити енергоефективність і спрощувати обслуговування об'єктів.
Офісні центри та бізнес-парки	Для цих об'єктів Cisco Digital Ceiling дозволяє підвищити комфорт співробітників, керувати ресурсами освітлення, вентиляції та інших систем, забезпечуючи економію енергії та зменшення експлуатаційних витрат.
Компанії, орієнтовані на екологічність	Такі організації можуть використовувати цю інфраструктуру для покращення екологічної ефективності будівель що відповідає стандартам Green Building і LEED.
Університети та школи	Cisco Digital Ceiling допомагає створити зручне та безпечне освітнє середовище і можливість контролю мікроклімату освітлення, систем обігріву.
Тікети та медичні заклади	У таких закладах ефективне управління інфраструктурою має важливе значення для забезпечення безпеки пацієнтів, контролю умов середовища та оптимізації витрат.
Системні інтегратори	Компанії, що спеціалізуються на впровадженні рішень IoT, можуть використовувати Cisco Digital Ceiling для створення та інтеграції інтелектуальних рішень на основі IoT і технологій автоматизації.
Постачальники IoT-обладнання та програмного забезпечення	Вони можуть бути цікаві у розширенні функціоналу Digital Ceiling інтегруючи додаткові пристрої або розробляючи рішення для сумісності з іншими платформами.
Інженери та інженери з управління інфраструктурою	Значність у централізованому управлінні та автоматизації процесів робить систему привабливою для спеціалістів, які відповідають за оптимізацію технологічних рішень у компаніях.
Менеджери з енергетичного менеджменту	Для цих фахівців можливість зниження енергоспоживання і скорочення витрат завдяки ефективному управлінню інфраструктурою.
Муніципалітети та органи місцевого самоврядування	Система Cisco Digital Ceiling може використовуватися в рамках ініціатив "розумних міст" для управління освітленням, безпекою, комунікаційними послугами та іншими аспектами міської інфраструктури.
Державні установи, відповідальні за енергоефективність	Установи, які контролюють стандарти енергоефективності, можуть підтримувати такі проєкти для досягнення позитивних ефектів у різних установках.

Рисунок 7.1 – Цільова аудиторія

Ці аудиторії можуть побачити вигоду у впровадженні Cisco Digital Ceiling для підвищення ефективності управління інфраструктурою, зниження витрат та покращення умов середовища у своїх будівлях та на об'єктах.

7.2 Оцінка привабливості шляхом застосування методів експертних оцінок

Оцінка привабливості проєкту програмної реалізації системи керування інфраструктурою на основі Cisco Digital Ceiling може проводитися за допомогою методу експертних оцінок. Це передбачає залучення групи експертів для надання незалежної оцінки, що дозволяє отримати комплексне уявлення про доцільність і перспективи проєкту. Проводиться відбір 5-10 експертів, які володіють знаннями у сферах інтелектуального управління інфраструктурою, будівельних технологій, енергоменеджменту та ІТ. До групи можна включити представників компаній, що займаються управлінням нерухомістю, девелоперів, системних інтеграторів і фахівців з ІоТ.

Далі визначаються критерії оцінювання (рисунок 7.2).



Рисунок 7.2 – Критерії оцінювання проєкту експертами

Залежно від стратегічних цілей, кожному критерію присвоюється вага: технічна життєздатність – 0.3, економічна доцільність – 0.2, зручність у використанні – 0.15, масштабованість – 0.15, екологічність – 0.1, інноваційність – 0.1.

Кожен експерт оцінює критерії за шкалою від 1 до 5, де 1 – найнижчий бал, а 5 – найвищий. Середні оцінки обчислюються шляхом підсумовування оцінок усіх експертів для кожного критерію та ділення на кількість експертів. Для кожного критерію середній бал множиться на його ваговий коефіцієнт. Отримані значення підсумовуються, що дає підсумкову оцінку привабливості проєкту.

Таблиця 7.1 – Зведені результати оцінювання

Критерій	Середній бал	Вага	Вага x Середній бал
Технічна життєздатність	4.2	0.3	1.26
Економічна доцільність	4.8	0.2	0.96
Зручність у використанні	4.0	0.15	0.6
Масштабованість	3.7	0.15	0.555
Екологічність	4.5	0.1	0.45
Інноваційність	4.7	0.1	0.47
Підсумкова оцінка			4.295

Загальна оцінка привабливості 4.295 свідчить про високу перспективність проєкту.

7.3 Вибір методу оцінки вартості ПЗ

Для оцінки вартості програмної реалізації системи керування інфраструктурою на основі Cisco Digital Ceiling можна використовувати кілька методів. Найбільш підходящими є: метод витрат та порівняльний аналіз.

Метод витрат (Cost Approach) передбачає оцінку вартості проєкту на основі підрахунку витрат, пов'язаних з його реалізацією. Він включає: прямі витрати (витрати на розробку програмного забезпечення, ліцензії на програму, обладнання та інфраструктуру), непрямі витрати (витрати на навчання персоналу, технічну підтримку, обслуговування та адміністрування системи), витрати на інтеграцію (витрати на інтеграцію системи з існуючою інфраструктурою). Цей метод корисний, коли є точні дані про витрати, але може не враховувати потенційні вигоди.

Метод порівняльного аналізу (Comparative Market Analysis) включає порівняння з іншими подібними рішеннями на ринку: аналіз конкурентів (оцінка вартості аналогічних рішень, їх можливостей та функціоналу), ринкова позиція (визначення, як ваше рішення позиціонується на ринку в порівнянні з конкурентами, що вплине на його цінність). Цей підхід дозволяє зрозуміти, як ваше рішення вписується в ринкові тренди та цінові пропозиції.

7.4 Розрахунок економічної ефективності від впровадження реалізованого ПЗ як фактору його привабливості

Приклад економічної ефективності впровадження системи керування інфраструктурою на основі CDC можна розглянути на основі різних аспектів, які стосуються зниження витрат, підвищення продуктивності, оптимізації ресурсів та покращення умов роботи. Ключові фактори, які можуть бути враховані в оцінці економічної ефективності подано на рисунку 7.3.

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		64

7.5 Пропозиція алгоритму просування проєкту розробки ПЗ

Алгоритм просування проєкту програмної реалізації системи керування інфраструктурою на основі Cisco Digital Ceiling можна розбити на кілька етапів(рисунок 7.4).

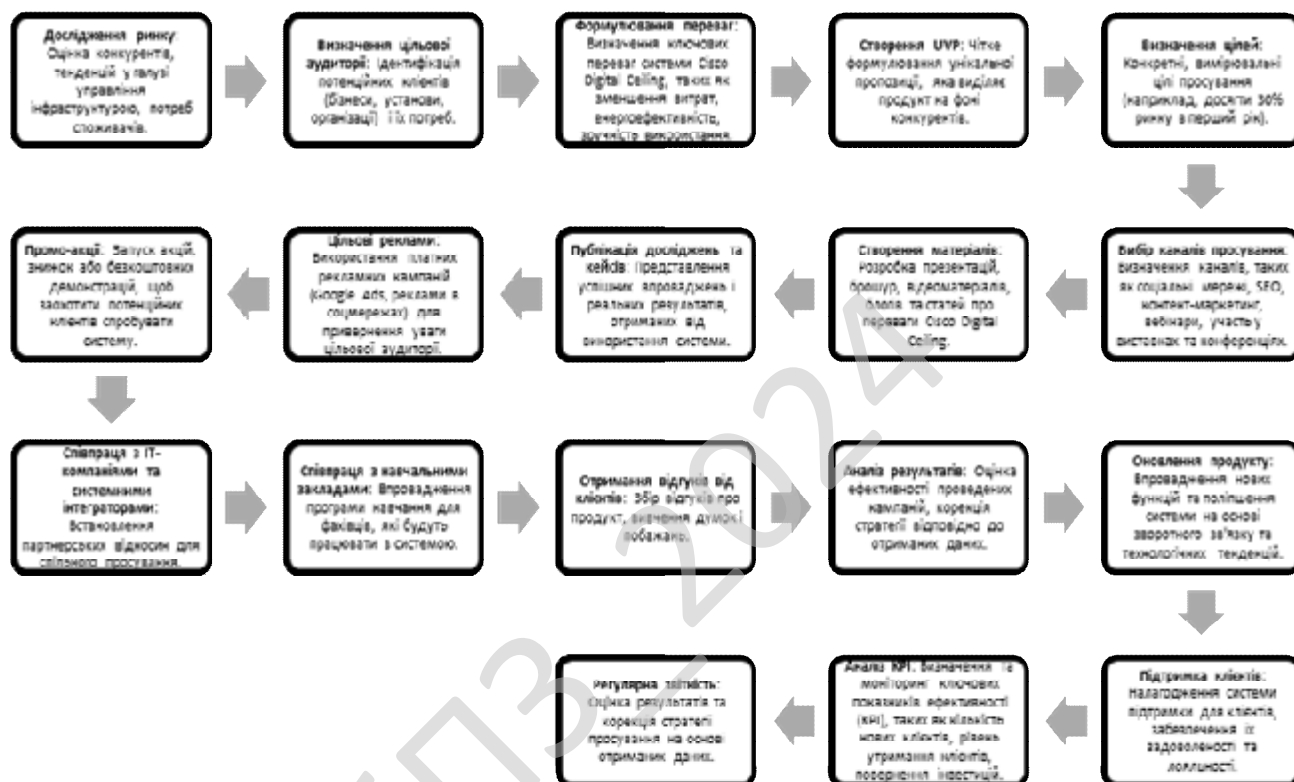


Рисунок 7.4 – Алгоритм просування проєкту

Кожен з цих етапів включає специфічні дії, спрямовані на досягнення успішного впровадження та популяризацію системи. Цей алгоритм просування проєкту програмної реалізації системи керування інфраструктурою на основі Cisco Digital Ceiling допоможе створити комплексний підхід до популяризації системи, залучення нових клієнтів і підвищення обізнаності про продукт.

7.6 Оптимізація каналів збуту та шляхів реалізації ПЗ

Для оптимізації каналів збуту та шляхів реалізації проекту програмної реалізації системи керування інфраструктурою на основі Cisco Digital Ceiling можна застосувати стратегії, представлені на рисунку 7.5.

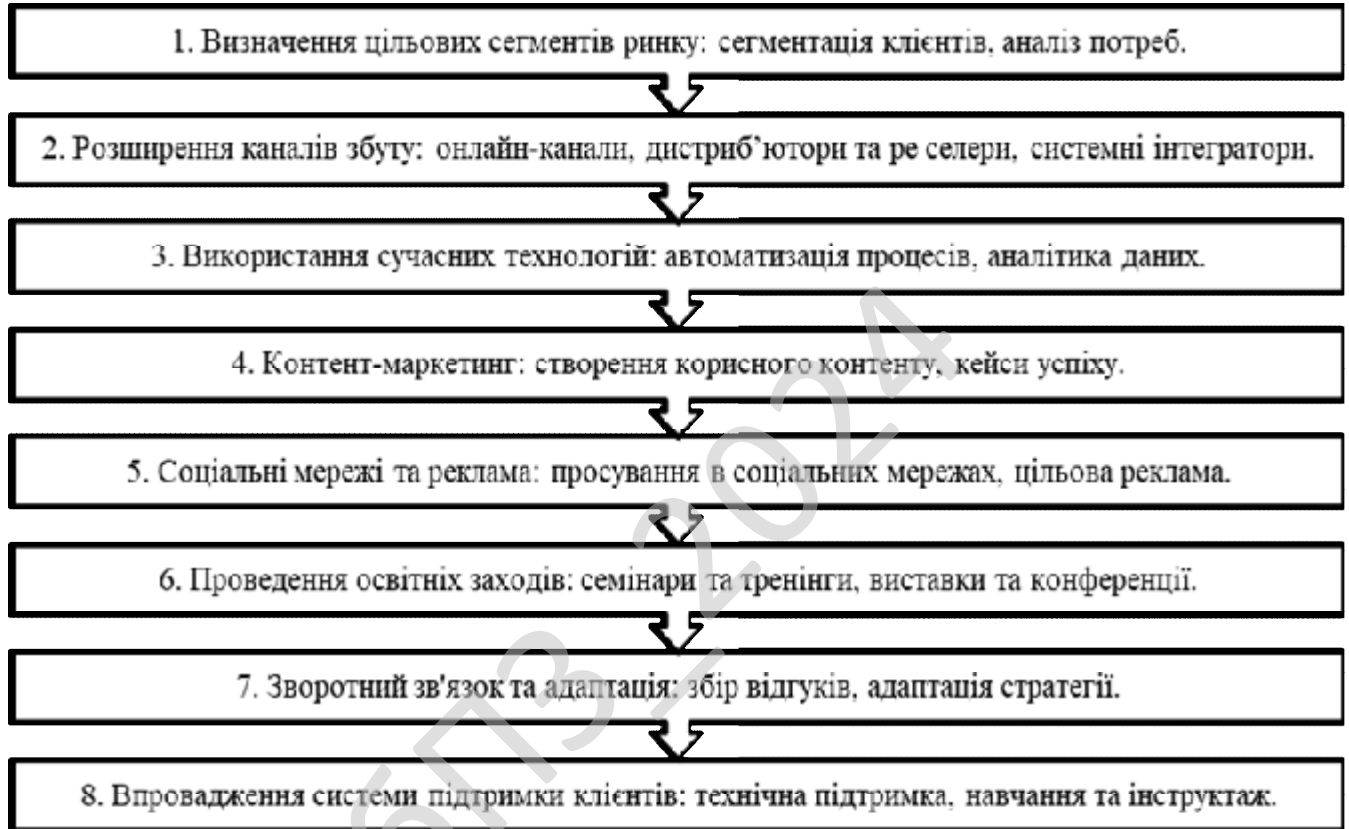


Рисунок 7.5 – Напрями оптимізації каналів збуту

Ці стратегії допоможуть оптимізувати канали збуту та реалізації проекту Cisco Digital Ceiling, покращити взаємодію з клієнтами, а також підвищити загальну ефективність бізнесу.

7.7 Визначення ключових факторів успіху конкретного проєкту

Ключові фактори успіху проєкту програмної реалізації системи керування інфраструктурою на основі **Cisco Digital Ceiling** включають кілька важливих аспектів, які можуть визначити успішність впровадження та ефективність роботи системи (рисунок 7.6).



Рисунок 7.6 – Ключові фактори успіху проєкту

Ці ключові фактори успіху сприяють ефективному впровадженню системи керування інфраструктурою на основі Cisco Digital Ceiling, забезпечуючи досягнення поставлених цілей і задоволення потреб бізнесу.

8 ЗАХОДИ З ОХОРОНИ ПРАЦІ ТА ТЕХНІКИ БЕЗПЕКИ

8.1 Вступ

Охорона здоров'я працівників, забезпечення безпеки умов праці, ліквідація професійних захворювань і виробничого травматизму повинна складати одну з головних завдань роботодавця.

Основою охорони праці є науковий аналіз умов праці, технологічних процесів, виробничого обладнання, робочих місць, трудових операцій, організації виробництва з метою виявлення шкідливих і небезпечних виробничих факторів, їх властивостей, особливостей впливу на організм людини. На підставі такого аналізу розробляються заходи та засоби, спрямовані на мінімізацію несприятливого впливу виробничих факторів, створення безпечних та нешкідливих умов праці.

Для того, щоб об'єктивно проаналізувати відповідність умов праці діючим нормативно-правовим актам, необхідно здійснити санітарно-гігієнічну характеристику умов праці відділу, в якому працює програміст, над розробкою даного програмного продукту.

В зв'язку з цим необхідно сконцентрувати увагу на небезпечних і шкідливих чинниках пов'язаних з постійною роботою за комп'ютером.

Електробезпека є одним із критичних питань для співробітників, що працюють із технікою, яка одержує живлення з електричної мережі. При невиконанні норм електробезпеки можлива поразка електричним струмом.

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		69

Проведений аналіз показує, що показники мікроклімату в приміщенні відповідають установленим нормам. Штучне опалення застосовується у холодний період року. В літню пору застосовується кондиціонер.

Для боротьби з пилом робляться регулярні провітрювання та вологі прибирання приміщенні.

У приміщенні знаходяться наступні джерела шуму: принтер HP 1100, електродвигуни вентиляторів ЕОМ.

Одним з найважливіших факторів, які впливають на ефективність трудової діяльності людини, та попереджають травматизм і професійні захворювання програмістів є освітлення на робочому місці.

З 2019 року діють Державні будівельні норми України “Природне і штучне освітлення” – ДБН В.2.5-28:2018 [1], у яких прописані вимоги до використання всіх освітлювальних приладів, у т.ч. світлодіодних.

Працю працівника, який постійно працює за комп'ютером, згідно ДБН В.2.5-28:2018 [1], можна віднести до роботи з малою точністю (найменший розмір об'єкта розрізнення від 1 до 5 мм) V-го розряду зорової роботи, з великою контрастністю об'єкта розрізнення (символів на екрані дисплея), з темним тлом (під розряд зорової роботи В). Приміщення можна віднести до 1-ої групи приміщень, у яких проводиться розрізнення об'єктів зорової роботи при фіксованому напрямку лінії зору того, що працює на робочу поверхню. Для такого типу приміщень і розряду зорової роботи нормоване значення коефіцієнта природної освітленості (КПО) робочої поверхні (при поєднаному, спільному освітленні), повинен становити не більше 1,5%, освітленість при штучному висвітленні повинна становити 300 Лк. [1], Крім того все поле зору повинне бути освітлено достатньо рівномірно – ця основна гігієнічна вимога. Так як яскраве світло на ділянці периферійного зору значно збільшує напруженість очей і, як наслідок, призводить до їх швидкої стомлюваності, ступінь освітлення приміщення і яскравість екрану комп'ютера повинні бути приблизно однаковими.

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		72

8.3 Розробка заходів з умов поліпшення охорони праці

Згідно аналізу умов праці в розглянутому приміщенні, ми одержали наступні результати:

- розмірі приміщення, у розрахунку на одному працюючого, відповідають нормативам;
- мікроклімат відповідає нормативному значенню;
- акустичні умови роботи не перевищують нормативних значень;

Таким чином можна припустити, що основною причиною можливого зниження працездатності програміста є психофізіологічний фактор, тому основна пропозиція буде така: дотримання позитивної психологічної атмосфери в колективі та регламентованого режиму праці та відпочинку, організація робочого місця з урахуванням ергономічних вимог.

Рекомендовані заходи: регулярні періодичні наочні огляди персоналом шляхів для евакуації людей із приміщення, відповідно до плану евакуації (який повинен розташовуватись на видному місці у приміщенні), включення до колективного договору мінімально можливого вмісту аптечок з обов'язково наявністю масок-клапанів, або іншого спорядження для штучного дихання. Регулярна періодична перевірка параметрів заземлення та занулення (вимірювання опору ланцюга).

Регулярна наочне знайомство персоналу із шляхами для евакуації людей із приміщення відповідно до плану евакуації, забезпечення розподільних щитів спеціальними розетками з заземлюючими контактами; організація заземлення всіх приладів і пристроїв, які працюють при нарузі вище 36 В.

Так як при ураженні електричним струмом у людини може статися фібриляція шлуночків серця, в організації бажано мати дефібрилятор і підготовлений персонал для роботи з ним.

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		73

Загальний опір розтіканню електричного струму заземлювача [11]:

$$R = (R_0 \cdot R_{\Pi}) / (R_0 \cdot \eta_{\Pi} + N \cdot R_{\Pi} \cdot K_{ев}) = \\ = (21,94 \cdot 15,13) / (21,94 \cdot 0,75 + 6,85 \cdot 15,13 \cdot 0,8) = 3,33 \text{ Ом.}$$

де $\eta_{\Pi} = 0,75$ – табличне значення коефіцієнта екранування з'єднуючої полоси [11].

Умова $R \leq R_{3н}$ виконується ($3,33 \leq 4$).

Так як R суттєво більше $R_{3н}$, зменшимо кількість вертикальних електродів до 6 і виконаємо перерахунок. У результаті остаточно отримали: кількість вертикальних електродів дорівнює 6 при $R = 3,8 \text{ Ом}$.

При необхідності можна зменшити кількість електродів заземлювача, зменшивши загальний опір розтіканню електричного струму заземлювача методом зменшення питомого опору ґрунта, домішуючи у ґрунт безпосередньо навколи електродів заземлювача розчини солей NaCl , CaCl , сажу, соду, шлак, коксову дрібницю, або спеціальні суміші.

8.5 Висновки до розділу

Дотримання всіх необхідних умов праці не лише сприяє збереженню здоров'я працівників, а також підвищує ефективність виробництва в цілому.

З цих міркувань було здійснено аналіз приміщення, призначеного для праці програмістів, проведено розгляд небезпечних та шкідливих факторів, що негативно впливають на програмістів під час роботи. Виконано розрахунок захисного штучного заземлення, як одного з ключових факторів безпеки програміста. Розроблено заходи з охорони праці.

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		76

9 ОСНОВНІ ВИСНОВКИ

Програмне забезпечення, створене в результаті виконання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти, призначено для системи керування інфраструктурою на основі Cisco Digital Ceiling.

В межах України в недостатній мірі представлені вітчизняні розробки в цій області.

У випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти наведені теоретичне узагальнення й рішення наукового завдання дослідження методів керування інфраструктурою на основі Cisco Digital Ceiling.

Рішення даного завдання полягало у вирішенні наступних задач:

- Був проведений огляд існуючих систем керування інфраструктурою на основі Cisco Digital Ceiling.
- Досліджена система керування інфраструктурою на основі Cisco Digital Ceiling.
- На основі отриманих результатів досліджень створена програмна реалізація системи керування інфраструктурою на основі Cisco Digital Ceiling.

Розроблені під час виконання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти алгоритми дозволяють успішно вирішувати завдання керування інфраструктурою на основі Cisco Digital Ceiling.

Проведено аналіз предметної галузі в ході якого були виявлені об'єкти, взаємодія яких носить істотний характер для функціональної діяльності предметної галузі, і їхні основні характеристики; побудована алгоритм і вибраний середовище розробки.

Розроблене програмне забезпечення має простий, дружній та зручний інтерфейс користувача, що забезпечує легкість у освоєнні роботи програмного продукту, зручність у використанні, і не потребує особливих спеціальних знань.

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		77

При створенні програмного забезпечення було використано об'єктно-орієнтований підхід, що відповідає сучасним тенденціям у галузі розробки комерційних програмних систем.

Програма реалізована на мові високого рівня Python. Дана мова програмування дозволяє найбільш ефективно обробляти дані. Це дозволило мінімізувати строк розробки програмного забезпечення, і, як слід, зменшити витрати на його розробку. Запропоноване програмне забезпечення ділиться на загальне програмне забезпечення, що поставляється із засобами обчислювальної техніки й спеціальне програмне забезпечення, що спеціально розроблене для даної конкретної системи й включає програми, що реалізують її функції.

Програма призначена для виконання під управлінням багатозадачної операційної системи Windows 10/11.

Даються необхідні рекомендації з установки розробленого програмного забезпечення.

Для підвищення рівня безпеки запропоновано застосовувати алгоритм REDOC III.

В цілому створене програмне забезпечення підтверджує правильність використаних проектних рішень та повністю відповідає вимогам технічного завдання. Створене програмне забезпечення має потенційну можливість для подальшого вдосконалення і застосування у різних галузях.

Проведено маркетингове та економічне обґрунтування IT-проєкту, що дозволило визначити ключові фактори успіху даного проєкту.

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		78

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Лівєр В.С. Дослідження та програмна реалізація системи керування інфраструктурою на основі Cisco Digital Ceiling // Збірник праць молодих науковців ЦНТУ. – Вип. 14. – Кропивницький: ЦНТУ, 2024.
2. Jack Ganssle and Michael Barr. 2003. Embedded Systems Dictionary. CMP Books.
3. Технології інтернету речей. Навчальний посібник [Електронний ресурс]: / Б. Ю. Жураковський, І.О. Зенів; КПІ ім. Ігоря Сікорського. – Електронні текстові дані (1 файл: 12,5 Мбайт). – Київ: КПІ ім. Ігоря Сікорського, 2021. – 271 с.
4. Greg Dunko, Joydeep Misra, Josh Robertson, Tom Snyder “A reference guide to the Internet of Things” / 2017 Bridgera LLC, RIoT.
5. Donald Norris “Programming with STM32. Getting started with the Nucleo Board and C/C++” 416 p. 2018.
6. Neil Kolban “Kolban’s book on ESP32”. Texas, USA. 951 p.
7. Al-Mudhafar Aqeel, A.M., Smirnova, T., Buravchenko, K., Smirnov, O. «The method of assessing and improving the user experience of subscribers in software-configured networks based on the use of machine learning». *Advanced Information Systems*, 2023, 7(2), pp. 49-56.
8. Kuznetsov, O., Kuznetsova, Y., Smirnov, O., Kostenko, O., Zvieriev, V. «Evaluating Hashing Algorithms in the Age of ASIC Resistance». *CEUR Workshop Proceedings*, 2023, 3628, pp. 93-105.
9. Smirnov, O., Sydorenko, V., Aleksander, M., Zhyharevych, O., Yenchев, S. «Simulation of the cloud IoT-based monitoring system for critical infrastructures». *CEUR Workshop Proceedings*, Volume 3530, 2023, pp. 256-265.
10. Smirnov, O., Odarchenko, R., Smirnova, T., Bondar, S., Volosheniuk, D. «Optimal Structure Construction of Private 5G Network for the Needs of

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		79

Enterprises». *Lecture Notes on Data Engineering and Communications Technologies*, 2023, 178, pp. 208–223.

11. Smirnova, T., Gnatyuk, S., Yudin, O., Sydorenko, V., Polozhentsev, A., «The Model for Calculating the Quantitative Criteria for Assessing the Security Level of Information and Telecommunication Systems». *CEUR Workshop Proceedings Volume 3156*, 2022, Pages 390-399.

12. Smirnova T., Gnatyuk S., Berdibayev R., Avkurova Zh., Iavich M. «Cloud-Based Cyber Incidents Response System and Software Tools». *Communications in Computer and Information Science*, 2021, vol 1486. Springer, Cham. pp 169-184.

13. Smirnov O., Kuznetsov A., Kiian A., Kuznetsova T. «Non-binary constant weight coding technique». *CEUR Workshop Proceedings*. Volume 2740, 2020, Pages 102-114.

14. Smirnov O., Alimseitova Zh., Adranova A., Akhmetov B., Lakhno V., Zhilkishbayeva G. «Models and algorithms for ensuring functional stability and cybersecurity of virtual cloud resources». *Journal of theoretical and applied information technology* Vol.98. No 21, 2020, P. 3334-3346.

15. Smirnov O., Kuznetsov A., Kiian A., Cherep A., Kanabekova M., Chepurko I. «Testing of code-based pseudorandom number generators for post-quantum application». *2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT)*, Ukraine, Kyiv, May 14-18. 2020. P. 172-177.

16. Smirnov O., Kuznetsov A., Pushkar'ov A., Serhiienko R., Babenko V., Kuznetsova T., «Representation of Cascade Codes in the Frequency Domain». In: Radivilova T., Ageyev D., Kryvinska N. (eds) *Data-Centric Business and Applications. Lecture Notes on Data Engineering and Communications Technologies*, vol 48. Springer, Cham. 2021. pp 557-587.

17. Smirnov, O., Markovets, O. Vovk, N., Turchyn, Y., «Model of informational support for social network administrators' content creation». *CEUR Workshop Proceedings* Volume 2616, 2020, Pages 125-136.

18. Smirnov, O., Drieieva, H., Drieiev, O., Polishchuk, Y., Brzhanov, R., Aleksander, M. «Method of fractal traffic generation by a model of generator on the graph». *CEUR Workshop Proceedings* Volume 2616, 2020, Pages 366-379.

19. Smirnov, O., Drieieva, H., Drieiev, O., Simakhin, V., Bondar, S., Odarchenko, R. «Managing multifractal properties of the binary sequence generated with the Markov chains», *CEUR Workshop Proceedings* Volume 2608, 2020, Pages 633-645.

20. Smirnov O. Kuznetsov A., Zaichenko Yu., Pastukhov M., Oleshko O., Kuznetsova K., «Formation of Discrete Signals with Special Correlation Properties». *International Conference on Information and Telecommunication Technologies and Radio Electronics, UkrMiCo 2019*; Odessa; Ukraine; 9-13 September 2019. P.22-28.

21. Smirnov, O., Kuznetsov, A., Kolovanova, I., Kuznetsova, T., «Noise immunity of the algebraic geometric codes». *International Journal of Computing*; 2019, Volume 18, Issue 4 – Research Institute for Intelligent Computer Systems – 2019. – P. 393-407.

22. Smirnov, O., Kuznetsov, A., Reshetniak, O., Ivko, N., Katkova, T., Kuznetsova, T., «Generators of Pseudorandom Sequence with Multilevel Function of Correlation». *2019 IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology (PIC S&T)*, Kyiv, Ukraine, 8 – 11 October 2019 . P.517-522.

23. Smirnov, O., Odarchenko, R., Abakumova, A., Usik, P., Kundyzy, M., «QoE optimization technique for media delivery in 5G networks». *2019 IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology (PIC S&T)*, Kyiv, Ukraine, 8 – 11 October 2019. P.597-601.

24. Smirnov, O., Krasnobayev, V., Yanko, A., Kuznetsova, T. «Methods of nulling numbers in the system of residual classes». *CEUR Workshop Proceedings*, Vol 2588, P. 90-106, 2019.

25. Smirnov, O., Kuznetsov, A., Kovalchuk, D., Averchev, A., Pastukhov, M., Kuznetsova, K., «Formation of Pseudorandom Sequences with Special Correlation Properties», *2019 3rd International Conference on Advanced Information and Communications Technologies, AICT-2019/ Lviv, Ukraine, 2-6 July, 2019*, P. 395-399.

26. Smirnov, O., Kuznetsov, A., Kiian, A., Zamula, A., Rudenko, S., Hryhorenko, V., «Variance Analysis of Networks Traffic for Intrusion Detection in Smart Grids», *2019 IEEE 6th International Conference On Energy Smart Systems (2019 IEEE ESS)*, Kyiv, Ukraine April 17-19, 2019 P. 353-358.

27. Smirnov, O., Kuznetsov, A., Kavun, S., Babenko, B., Nakisko, O., Kuznetsova, K., «Malware Correlation Monitoring in Computer Networks of Promising Smart Grids», *2019 IEEE 6th International Conference On Energy Smart Systems (2019 IEEE ESS)*, Kyiv, Ukraine April 17-19, 2019 P. 347-352.

28. Smirnov, O., Kuznetsov, A., Kovalchuk, D., Pastukhov, M., Kuznetsova, K., Prokopovych-Tkachenko, D., «Discrete Signals with Special Correlation Properties», *CEUR Workshop Proceedings Volume 2353, CEUR Workshop Proceedings 2019*, Pages 618-629.

29. Smirnov A.A., Kuznetsov A.A., Danilenko D.A., Berezovsky A., «The statistical analysis of a network traffic for the intrusion detection and prevention systems», *Telecommunications and Radio Engineering*. – Volume 74, Issue 1. – Begel House Inc. – 2015. – P. 61-78.

30. Вінтенко Б.Ю., Смірнов О.А., Коваленко А.С., Смірнов С.А., Буравченко К.О. «Дослідження вимог міжнародних стандартів IEC60880 та IEC62138 з розробки програмного забезпечення інформаційно-керуючих систем АЕС, важливих для безпеки». *Системи управління, навігації та зв'язку*, 2023, вип. 3(73), С. 155-166.

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		82

31. Аль-Мудхафар Акіл Абдулхуссейн М., Смірнова Т.В., Буравченко К.О., Смірнов О.А. «Метод оцінки та підвищення користувальницького досвіду абонентів в програмно-конфігурованих мережах на основі використання машинного навчання». *Сучасні інформаційні системи*, 2023, том 7, № 2, С. 49-56.

32. Смірнова Т.В., Гнатюк С.О., Сидоренко В.М., Юдін О.Ю., Сидоренко С.Ю., «Модель визначення критичності галузевих інформаційно-телекомунікаційних систем». *Проблеми інформатизації та управління*, № 2(70). 2022. С. 28-37.

33. Смірнов О.А., Смірнова Т.В., Якименко Н.М., Смірнов С.А., Поліщук Л.І., «Дослідження стійкості до диференціального криптоаналізу запропонованої функції гешування удосконаленого модуля криптографічного захисту в інформаційно-комунікаційних системах» *Системи управління, навігації та зв'язку*, 2022, № 3(69). С. 93-98.

34. Смірнов О.А., Смірнова Т.В., Якименко Н.М., Поліщук Л.І., Смірнов С.А. «Дослідження статистичної стійкості та швидкісних характеристик запропонованої функції гешування удосконаленого модуля криптографічного захисту в інформаційно-комунікаційних системах» *Вісник Хмельницького національного університету. Серія: «Технічні науки»*, № 2 (307). С. 46-52. 2022.

35. Смірнов О.А., Смірнова Т.В., Константинова Л.В., Смірнов С.А., Якименко Н.М., «Дослідження стійкості до лінійного криптоаналізу запропонованої функції гешування удосконаленого модуля криптографічного захисту в інформаційно-комунікаційних системах» *Системи управління, навігації та зв'язку*, 2022, № 1(67). С. 84-89.

36. Смірнов О.А., Смірнова Т.В., Буравченко К.О., Кравченко С.С., Горбов В.О., «Хмарна система підтримки прийняття рішень технологічного процесу відновлення поверхонь конструкцій і деталей машин». *Сучасні інформаційні системи*. 2021. Т. 5, № 4. С. 79-95

37. Смірнов О.А., Усік П.С., Миронець І.В., Буравченко К.О., Якименко Н.М. «Метод підвищення ефективності розподіленої обробки даних у

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		83

комп'ютерних системах операторів стільникового зв'язку» *Вісник Черкаського державного технологічного університету. Технічні науки.* №4. С. 103-110. 2020.

38. О.А.Смірнов, Т.В.Смірнова, Л.І. Поліщук, К.О. Буравченко, А.О.Макевнін, «Дослідження хмарних технологій як сервісів», *Кібербезпека: освіта, наука, техніка.* № 3(7). С. 43-62. 2020.

39. Смірнов О.А., Коноплицька-Слободенюк О.К., Смірнов С.А., Буравченко К.О., Смірнова Т.В., Поліщук Л.І. Інформаційна безпека в комп'ютерних мережах. Навчальний посібник – Кропивницький: вид. Лисенко В.Ф. 2020. – 294 с.

40. О.А. Смірнов, П.С. Усік, «Дослідження перспектив використання технологічних рішень в мережах 5G» у *Кібербезпека та інформаційні технології: монографія.* – Х. : ТОВ «ДІСА ПЛЮС», 2020.С. 122-135.

41. Смірнов О.А., Дреєва Г.М., Дреєв О.М., Смірнова Т.В. «Фрактальний аналіз генератора самоподібного трафіку на основі ланцюга Маркова». *Центральноукраїнський науковий вісник. Технічні науки.* № 2(33). с. 161-172, 2019.

42. Смірнов О.А., Коноплицька-Слободенюк О.К., Смірнов С.А., Буравченко К.О., Смірнова Т.В. Поліщук Л.І. Проектування комп'ютерних систем та мереж. Навчальний посібник – Кропивницький: вид. Лисенко В.Ф. 2019. – 264 с.

43. Smirnov, O., Kuznetsov, A., Kuznetsova., K. Synthesis of Discrete Signals with Improved Correlation Properties. Монографія: In.: ISCI'2019: Information Security in Critical Infrastructures. Collective monograph. Edited by Ivan D. Gorbenko and Alexandr A. Kuznetsov, ASC Academic Publishing, USA, 2019, pp. 281-299. – ISBN: 978-0-9989826-8-7 (Hardback), ISBN: 978-0-9989826-9-4 (Ebook).

44. Смірнов О.А., Дреєва Г.М. Метод генерування фрактального трафіку за допомогою моделі генератора на графі. Монографія: Інформаційна безпека та інформаційні технології : монографія / за заг. ред. В. С. Пономаренка. – Х. : Вид. Рожко С.Г. 2019. С. 123-139

					ВКРМ-123.24.0003.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		84

45. Дреєва Г.М., Смірнов О.А., Дреєв О.М. Метод генерування фрактальноподібної числової послідовності на основі скінченного автомату для моделювання трафіку у мережі. Центральноукраїнський науковий вісник. Технічні науки. № 1(32). с. 173-183, 2019.

46. Смірнова Т.В., Солових Є.К., Смірнов О.А., Дреєв О.М. Побудова хмарних інформаційних технологій оптимізації технологічного процесу відновлення та зміцнення поверхонь деталей. Центральноукраїнський науковий вісник. Технічні науки. № 1(32). с. 184-194, 2019.

47. Смірнов О.А., Смірнов С.А., Поліщук Л.І., Смірнова Т.В., Коноплицька-Слободенюк О.К. Метод формування антивірусного захисту даних з використанням безпечної маршрутизації метаданих. Кібербезпека: освіта, наука, техніка. – Том 3 № 3. – Київ: КУ ім. Бориса Грінченка. – 2019. – С. 63-87.

48. Смірнов О.А., Гнатюк С.О., Кавун С.В., Терейковський І.А., Жмурко Т.О., Смірнов С.А., Коваленко А.С. Основи безпеки в комп'ютерних мережах. Навчальний посібник – Кропивницький: вид. Лисенко В.Ф. 2018. – 177 с.

49. Смірнов О.А., Котелянець В.В. Стійкі до колізій стохастичні моделі функціонування безпроводових сенсорних мереж. Вісник інженерної академії України, №3, с. 145-152, 2018

50. Смірнов О.А., Смірнов С.А., Дідик А.К., Дреєв А.М. Алгоритми формування безлічі маршрутів передачі метаданих у антивірусні хмарні системи. Збірник наукових праць "Системи обробки інформації". – Випуск 5 (142). – Х.: ХУПС – 2016. – С. 148-152.

51. Смірнов О.А., Смірнов С.А. Дідик А.К., Дреєв О.М. Моделі системи нейромережових експертів безпечної маршрутизації у хмарних антивірусних системах. Збірник наукових праць "Системи обробки інформації". – Випуск 3 (140). – Х.: ХУПС – 2016. – С. 36-39.

Додаток А
(обов'язковий)

Технічне завдання

Зміст

1 Найменування та область застосування.....	2
2 Підстава для розробки.....	2
3 Мета та призначення розробки.....	2
4 Джерела розробки.....	2
5 Технічні вимоги.....	2
5.1 Вміст проекту.....	2
5.2 Показники призначення.....	3
5.3 Вимоги до функціональних характеристик.....	3
5.4 Вимоги до архітектури.....	3
5.5 Вимоги до надійності.....	3
5.6 Умови експлуатації.....	4
5.7 Вимоги до складу та параметрів технічних засобів.....	4
5.8 Вимоги до інформаційної і програмної сумісності.....	4
5.8.1 Обладнання.....	4
5.8.2 Мова програмування.....	4
5.8.3 Вхідні дані.....	5
5.8.4 Вихідні дані.....	5
6 Вимоги до програмної документації.....	5
7 Економічні вимоги.....	5
8 Вимоги щодо охорони праці.....	5
9 Перелік документів, що розробляються.....	6
10 Етапи розробки.....	6
11 Порядок контролю та приймання.....	6

					ВКРМ-123.24.0003.00.00.ТЗ			
Вим.	Арк.	№ документа	Підпис	Дата				
Розробив	Лівер В.С.				<i>Дослідження та програмна реалізація системи керування інфраструктурою на основі Cisco Digital Ceiling</i>	Літ.	Аркуш	Аркушів
Перевірів	Якименко Н.М.					М	1	6
Н. Контр.	Коваленко А.С.				ЦНТУ КІ-23Мз			
Затв.	Смірнов О.А.							

1 Найменування та область застосування

Це технічне завдання розповсюджується на дослідження та програмну реалізацію системи керування інфраструктурою на основі Cisco Digital Ceiling.

2 Підстава для розробки

Підставою для розробки служить завдання на випускню кваліфікаційну роботу за другим (магістерським) рівнем вищої освіти, видане на кафедрі кібербезпеки та програмного забезпечення (нак. № 21-13 від 07.08.2024 року).

3 Мета та призначення розробки

Метою випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти є дослідження та програмна реалізація системи керування інфраструктурою на основі Cisco Digital Ceiling.

4 Джерела розробки

Джерелом цієї випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти є стосовна до теми література і існуючі аналоги.

5 Технічні вимоги

5.1 Склад продукції

Складниками розробки є:

- вибір і обґрунтування методів реалізації проекту;
- розробка програмної частин системи, а також розробка взаємодії системи з ОС та з користувачем;

					ВКРМ-123.24.0003.00.00.ТЗ	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		2

- техніко-економічне обґрунтування доцільності прийнятого до розробки програмного забезпечення;
- аналіз умов праці;
- розробка програми, що реалізує спроектовані алгоритми роботи системи.

5.2 Показники призначення

Система повинна забезпечувати:

- програмну реалізацію системи керування інфраструктурою на основі Cisco Digital Ceiling;
- цілісність даних у процесі роботи та при зберіганні;
- простий, інтуїтивно зрозумілий інтерфейс.

5.3 Вимоги до функціональних характеристик

Розроблене програмне забезпечення не повинно мати обмежень на версію драйверів та операційної системи.

5.4 Вимоги до архітектури

Компонент, що розробляється повинен використовувати системні засоби та апаратні засоби, що на даному етапі розвитку обчислювальної техніки найбільше поширені.

5.5 Вимоги до надійності

Програмні модулі написані по всім правилам, які стосуються стандартних викликів процедур, функцій, методів і форм, визначених технічною документацією на середовище розробки.

					ВКРМ-123.24.0003.00.00.ТЗ	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		3

5.6 Умови експлуатації

Робочі місця користувачів ПЗ повинні задовольняти наступним умовам експлуатації:

- температура повітря: 19-20 град. по Цельсію;
- відносна вологість повітря до 80%;
- атмосферний тиск 107 кПа.

5.7 Вимоги до складу та параметрів технічних засобів

Програмне забезпечення повинно бути реалізоване на ПЕОМ, працювати в ОС Windows 10/11 і з сумісними з цією платформою пристроями і прикладним програмним забезпеченням.

5.8 Вимоги до інформаційної і програмної сумісності

Переносність програмного забезпечення повинна бути забезпечена за рахунок його реалізації стандартного інтерфейсу взаємодії з ОС, що працюють під управлінням ОС Windows 10/11.

5.8.1 Обладнання

Комп'ютер Intel® Celeron/8 Mb/1.2 Gb/SVGA 14" 1Mb або сумісні з ним.

5.8.2 Мова програмування

Середовище Python.

					ВКРМ-123.24.0003.00.00.ТЗ	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		2

5.8.3 Вхідні дані

Опис алгоритму роботи запропонованої системи.

5.8.4 Вихідні дані

Робоча програма.

6 Вимоги до програмної документації

Програмна продукція повинна бути представлена у виді опису структури даних, схем та опису алгоритму, а також текстів вихідних модулів програмного забезпечення згідно ЄСПД .

7 Економічні вимоги

7.1 Для ПЗ необхідно виробити функціонально-вартісний аналіз варіантів розробки.

7.2 Виконати маркетингове та економічне обґрунтування ІТ-проєкту з урахуванням цін на 3 вересня 2024 року.

8 Вимоги щодо охорони праці

В частині охорони праці випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти повинен бути розглянутий аналіз санітарно-гігієнічних умов праці на робочому місці програміста.

					ВКРМ-123.24.0003.00.00.ТЗ	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		5

9 Перелік документів, що розробляються

- Наукова новизна – 1 аркуш.
- Структурна схема системи – 1 аркуш.
- Функціональна схема системи – 1 аркуш.
- Діаграма процесів – 1 аркуш.
- Блок-схема алгоритму роботи програми – 2 аркуша.
- Показники економічної ефективності – 1 аркуш.
- Пояснювальна записка – 85 аркушів.

10 Етапи розробки

10.1 Збір і обробка інформації по темі випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти. Постановка задачі на виконання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти (складання ТЗ).

10.2 Проведення досліджень або експериментальних робіт для уточнення основних положень випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти.

10.3 Розробка функціональних схем, блок схем алгоритмів роботи програмного забезпечення.

10.4 Побудова схем взаємодії даних.

10.5 Створення прототипу ПЗ.

10.6 Віднаходження ПЗ, аналіз отриманих результатів.

10.7 Робота над питанням охорони праці і техніки безпеки.

10.8 Маркетингове та економічне обґрунтування ІТ-проєкту.

10.9 Оформлення пояснювальної записки і виконання робіт по графічній частині.

11 Порядок контролю та приймання

11.1 Подання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти на попередній захист 02.12.2024 р.

11.2 Подання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти на захист 17.12.2024 р.

					ВКРМ-123.24.0003.00.00.ТЗ	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		6

Додаток Б
(обов'язковий)

Міністерство освіти і науки України
Центральноукраїнський національний технічний університет

ЗАТВЕРДЖУЮ

Керівник випускної кваліфікаційної роботи за
другим (магістерським) рівнем вищої освіти

_____ Якименко Н.М.

*Дослідження та програмна реалізація
системи керування інфраструктурою на основі Cisco Digital Ceiling*

Лістинг програми

Код документу 12

Носій: CD/DVD-диск / USB-флеш-накопичувач

Загальна кількість аркушів: 19

Літера: РП

Кропивницький – 2024 року

Основна програма

```
import cisco_apis as cisco
import numpy as np
import time

# Модуль для підключення та контролю Digital Ceiling
class CiscoDigitalCeilingController:
    def __init__(self, ip_address, username, password):
        self.ip_address = ip_address
        self.username = username
        self.password = password
        self.connection = self.connect_to_ceiling()
        self.rooms = {}

    def connect_to_ceiling(self):
        # Підключаємося до Cisco Digital Ceiling API
        return cisco.connect(self.ip_address, self.username, self.password)

    def add_room(self, room_id, initial_brightness, initial_temperature):
        # Додаємо кімнату до системи керування
        self.rooms[room_id] = {
            "brightness": initial_brightness,
            "temperature": initial_temperature
        }

    def set_lighting(self, room_id, brightness_level):
        # Встановлюємо рівень освітлення
        if room_id in self.rooms:
            self.rooms[room_id]["brightness"] = brightness_level
            cisco.set_lighting(self.connection, room_id, brightness_level)

    def set_temperature(self, room_id, temperature):
        # Встановлюємо температуру в кімнаті
        if room_id in self.rooms:
            self.rooms[room_id]["temperature"] = temperature
            cisco.set_temperature(self.connection, room_id, temperature)

    def get_room_status(self, room_id):
        # Отримуємо стан кімнати
        if room_id in self.rooms:
            return self.rooms[room_id]
        return None

    def monitor_energy_consumption(self):
        # Моніторимо енергоспоживання інфраструктури
        return cisco.get_energy_consumption(self.connection)

    def automate_lighting(self, room_id, time_of_day):
        # Автоматичне налаштування освітлення в залежності від часу дня
        if 6 <= time_of_day <= 18:
```

```

        self.set_lighting(room_id, 100)
    else:
        self.set_lighting(room_id, 50)

    def schedule_temperature(self, room_id, schedule):
# Плануємо зміну температури в залежності від розкладу
        current_hour = time.localtime().tm_hour
        if current_hour in schedule:
            self.set_temperature(room_id, schedule[current_hour])

    def monitor_and_optimize_energy(self):
# Автоматичний моніторинг і оптимізація енерговитрат
        energy_data = self.monitor_energy_consumption()
        for room_id in self.rooms:
            if energy_data[room_id] > 50:
                self.set_lighting(room_id, 70)

    def shutdown_infrastructure(self):
# Вимикаємо всю інфраструктуру для економії енергії
        for room_id in self.rooms:
            self.set_lighting(room_id, 0)
            self.set_temperature(room_id, 18)

# Модуль розрахунку енергоефективності
class EnergyEfficiencyCalculator:
    def __init__(self, energy_data):
        self.energy_data = energy_data

    def calculate_efficiency(self):
# Виконуємо розрахунок енергоефективності
        baseline_consumption = np.mean(self.energy_data["baseline"])
        actual_consumption = np.mean(self.energy_data["actual"])
        efficiency = (baseline_consumption - actual_consumption) /
baseline_consumption * 100
        return efficiency

# Модуль моніторингу температурних показників
class TemperatureMonitoring:
    def __init__(self, connection):
        self.connection = connection

    def monitor_temperature(self, room_id):
# Моніторимо температуру в заданій кімнаті
        return cisco.get_temperature(self.connection, room_id)

    def detect_anomalies(self, room_id):
# Виявлення аномальних показників температури
        temperature = self.monitor_temperature(room_id)
        if temperature > 28 or temperature < 16:

```

```
        print(f"Warning: Temperature anomaly detected in {room_id}:  
{temperature}°C")  
        return True  
    return False  
  
# Модуль керування розкладом вентиляції  
class VentilationController:  
    def __init__(self, connection):  
        self.connection = connection  
  
    def control_ventilation(self, room_id, status):  
# Керування вентиляцією в залежності від показників якості повітря  
    if status == "on":  
        cisco.set_ventilation(self.connection, room_id, True)  
    else:  
        cisco.set_ventilation(self.connection, room_id, False)  
  
    def automate_ventilation(self, room_id, air_quality):  
  
# Автоматичне включення вентиляції за показниками якості повітря  
    if air_quality < 60:  
        self.control_ventilation(room_id, "on")  
    else:  
        self.control_ventilation(room_id, "off")  
  
# Модуль для інтеграції з системою безпеки  
class SecurityIntegration:  
    def __init__(self, connection):  
        self.connection = connection  
  
    def arm_system(self):  
# Активуємо систему безпеки  
        cisco.arm_security(self.connection)  
  
    def disarm_system(self):  
# Вимикаємо систему безпеки  
        cisco.disarm_security(self.connection)  
  
    def monitor_intrusion(self):  
# Моніторимо вторгнення  
        intrusion = cisco.detect_intrusion(self.connection)  
        if intrusion:  
            print("Intrusion detected! Notifying authorities.")  
            return True  
        return False  
  
# Модуль для інтеграції з розумним освітленням  
class SmartLightingSystem:  
    def __init__(self, connection):  
        self.connection = connection  
  
    def set_lighting_color(self, room_id, color):  
# Встановлюємо колір освітлення
```

```

    cisco.set_lighting_color(self.connection, room_id, color)

    def adapt_lighting_by_activity(self, room_id, activity):
# Адаптація освітлення в залежності від активності
        if activity == "presentation":
            self.set_lighting_color(room_id, "blue")
        elif activity == "meeting":
            self.set_lighting_color(room_id, "warm_white")
        else:
            self.set_lighting_color(room_id, "neutral_white")

# Модуль для керування мультимедіа
class MultimediaController:
    def __init__(self, connection):
        self.connection = connection

    def play_media(self, room_id, media_url):
# Програвання медіа у вибраній кімнаті
        cisco.play_media(self.connection, room_id, media_url)

    def stop_media(self, room_id):
# Зупинка відтворення
        cisco.stop_media(self.connection, room_id)

# Модуль для обробки даних клімату
class ClimateDataProcessor:
    def __init__(self, connection):
        self.connection = connection

    def collect_climate_data(self):
# Збір даних з датчиків клімату
        return cisco.get_climate_data(self.connection)

    def analyze_climate_trends(self):
# Аналіз кліматичних трендів
        climate_data = self.collect_climate_data()
        avg_temperature = np.mean([data["temperature"] for data in
climate_data])
        print(f"Average temperature: {avg_temperature}°C")

# Основна програма

def main():
# Ініціалізація системи Digital Ceiling
    ceiling_controller = CiscoDigitalCeilingController("192.168.1.1", "admin",
"password")

# Додавання кімнат до системи
    ceiling_controller.add_room("room_101", 70, 22)
    ceiling_controller.add_room("room_102", 60, 24)

# Керування освітленням та температурою
    ceiling_controller.set_lighting("room_101", 80)

```

```
ceiling_controller.set_temperature("room_101", 21)

# Автоматичне налаштування освітлення
ceiling_controller.automate_lighting("room_101", 14)

# Розрахунок енергоефективності
energy_data = {
    "baseline": [120, 115, 110],
    "actual": [105, 102, 100]
}
energy_calculator = EnergyEfficiencyCalculator(energy_data)
efficiency = energy_calculator.calculate_efficiency()
print("Energy efficiency:", efficiency, "%")

# Моніторинг температурних показників
temperature_monitor = TemperatureMonitoring(ceiling_controller.connection)
temperature_monitor.monitor_temperature("room_101")

# Інтеграція з системою безпеки
security_system = SecurityIntegration(ceiling_controller.connection)
security_system.arm_system()

# Інтеграція з мультимедіа системою
multimedia_controller = MultimediaController(ceiling_controller.connection)
multimedia_controller.play_media("room_101", "http://example.com/video.mp4")

# Автоматизація вентиляції
ventilation_controller =
VentilationController(ceiling_controller.connection)
ventilation_controller.automate_ventilation("room_102", 55)

# Викликаємо головну функцію
if __name__ == "__main__":
    main()
```

Файл cisco_digital_ceiling_controller.py

```
# cisco_digital_ceiling_controller.py
import cisco_apis as cisco

class CiscoDigitalCeilingController:
    def __init__(self, ip_address, username, password):
        self.ip_address = ip_address
        self.username = username
        self.password = password
        self.connection = self.connect_to_ceiling()
        self.rooms = {}

    def connect_to_ceiling(self):
# Підключаємося до Cisco Digital Ceiling API
        return cisco.connect(self.ip_address, self.username, self.password)

    def add_room(self, room_id, initial_brightness, initial_temperature):
# Додаємо кімнату до системи керування
        self.rooms[room_id] = {
            "brightness": initial_brightness,
            "temperature": initial_temperature
        }

    def set_lighting(self, room_id, brightness_level):
# Встановлюємо рівень освітлення
        if room_id in self.rooms:
            self.rooms[room_id]["brightness"] = brightness_level
            cisco.set_lighting(self.connection, room_id, brightness_level)

    def set_temperature(self, room_id, temperature):
# Встановлюємо температуру в кімнаті
        if room_id in self.rooms:
            self.rooms[room_id]["temperature"] = temperature
            cisco.set_temperature(self.connection, room_id, temperature)

    def get_room_status(self, room_id):
# Отримуємо стан кімнати
        if room_id in self.rooms:
            return self.rooms[room_id]
        return None

    def monitor_energy_consumption(self):
# Моніторимо енергоспоживання інфраструктури
        return cisco.get_energy_consumption(self.connection)

    def automate_lighting(self, room_id, time_of_day):
# Автоматичне налаштування освітлення в залежності від часу дня
        if 6 <= time_of_day <= 18:
            self.set_lighting(room_id, 100)
        else:
            self.set_lighting(room_id, 50)
```

```
def schedule_temperature(self, room_id, schedule):  
# Плануємо зміну температури в залежності від розкладу  
    current_hour = time.localtime().tm_hour  
    if current_hour in schedule:  
        self.set_temperature(room_id, schedule[current_hour])  
  
def monitor_and_optimize_energy(self):  
# Автоматичний моніторинг і оптимізація енерговитрат  
    energy_data = self.monitor_energy_consumption()  
    for room_id in self.rooms:  
        if energy_data[room_id] > 50:  
            self.set_lighting(room_id, 70)  
  
def shutdown_infrastructure(self):  
# Вимикаємо всю інфраструктуру для економії енергії  
    for room_id in self.rooms:  
        self.set_lighting(room_id, 0)  
        self.set_temperature(room_id, 18)
```

КБПЗ_2024

Файл energy_efficiency_calculator.py

```
# energy_efficiency_calculator.py
import numpy as np

class EnergyEfficiencyCalculator:
    def __init__(self, energy_data):
        self.energy_data = energy_data

    def calculate_efficiency(self):
# Виконуємо розрахунок енергоефективності
        baseline_consumption = np.mean(self.energy_data["baseline"])
        actual_consumption = np.mean(self.energy_data["actual"])
        efficiency = (baseline_consumption - actual_consumption) /
baseline_consumption * 100
        return efficiency
```

КБПЗ_2024

Файл temperature_monitoring.py

```
# temperature_monitoring.py
import cisco_apis as cisco

class TemperatureMonitoring:
    def __init__(self, connection):
        self.connection = connection

    def monitor_temperature(self, room_id):
        # Моніторимо температуру в заданій кімнаті
        return cisco.get_temperature(self.connection, room_id)

    def detect_anomalies(self, room_id):
        # Виявлення аномальних показників температури
        temperature = self.monitor_temperature(room_id)
        if temperature > 28 or temperature < 16:
            print(f"Warning: Temperature anomaly detected in {room_id}:
{temperature}°C")
            return True
        return False
```

КБПЗ_2024

Файл ventilation_controller.py

```
# ventilation_controller.py
import cisco_apis as cisco

class VentilationController:
    def __init__(self, connection):
        self.connection = connection

    def control_ventilation(self, room_id, status):
# Керування вентиляцією в залежності від показників якості повітря
        if status == "on":
            cisco.set_ventilation(self.connection, room_id, True)
        else:
            cisco.set_ventilation(self.connection, room_id, False)

    def automate_ventilation(self, room_id, air_quality):
# Автоматичне включення вентиляції за показниками якості повітря
        if air_quality < 60:
            self.control_ventilation(room_id, "on")
        else:
            self.control_ventilation(room_id, "off")
```

Файл security_integration.py

```
# security_integration.py
import cisco_apis as cisco

class SecurityIntegration:
    def __init__(self, connection):
        self.connection = connection

    def arm_system(self):
        # Активуємо систему безпеки
        cisco.arm_security(self.connection)

    def disarm_system(self):
        # Вимикаємо систему безпеки
        cisco.disarm_security(self.connection)

    def monitor_intrusion(self):
        # Моніторимо вторгнення
        intrusion = cisco.detect_intrusion(self.connection)
        if intrusion:
            print("Intrusion detected! Notifying authorities.")
            return True
        return False
```

Файл multimedia_controller.py

```
# multimedia_controller.py
import cisco_apis as cisco

class MultimediaController:
    def __init__(self, connection):
        self.connection = connection

    def play_media(self, room_id, media_url):
# Програвання медіа у вибраній кімнаті
        cisco.play_media(self.connection, room_id, media_url)

    def stop_media(self, room_id):
# Зупинка відтворення
        cisco.stop_media(self.connection, room_id)
```

КБПЗ_2024

Файл main.py

```
# main.py
from cisco_digital_ceiling_controller import CiscoDigitalCeilingController
from energy_efficiency_calculator import EnergyEfficiencyCalculator
from temperature_monitoring import TemperatureMonitoring
from ventilation_controller import VentilationController
from security_integration import SecurityIntegration
from multimedia_controller import MultimediaController
import time

def main():
    # Ініціалізація системи Digital Ceiling
    ceiling_controller = CiscoDigitalCeilingController("192.168.1.1", "admin",
"password")

    # Додавання кімнат до системи
    ceiling_controller.add_room("room_101", 70, 22)
    ceiling_controller.add_room("room_102", 60, 24)

    # Керування освітленням та температурою
    ceiling_controller.set_lighting("room_101", 80)
    ceiling_controller.set_temperature("room_101", 21)

    # Автоматичне налаштування освітлення
    ceiling_controller.automate_lighting("room_101", 14)

    # Розрахунок енергоефективності
    energy_data = {
        "baseline": [120, 115, 110],
        "actual": [105, 102, 100]
    }
    energy_calculator = EnergyEfficiencyCalculator(energy_data)
    efficiency = energy_calculator.calculate_efficiency()
    print("Energy efficiency:", efficiency, "%")

    # Моніторинг температурних показників
    temperature_monitor = TemperatureMonitoring(ceiling_controller.connection)
    temperature_monitor.monitor_temperature("room_101")

    # Інтеграція з системою безпеки
    security_system = SecurityIntegration(ceiling_controller.connection)
    security_system.arm_system()

    # Інтеграція з мультимедіа системою
    multimedia_controller = MultimediaController(ceiling_controller.connection)
    multimedia_controller.play_media("room_101", "http://example.com/video.mp4")

    # Автоматизація вентиляції
    ventilation_controller =
VentilationController(ceiling_controller.connection)
    ventilation_controller.automate_ventilation("room_102", 55)
```

```
if __name__ == "__main__":  
    main()
```

КБПЗ_2024

Файл `lighting_color_controller.py`

```
# lighting_color_controller.py
import cisco_apis as cisco

class LightingColorController:
    def __init__(self, connection):
        self.connection = connection

    def set_lighting_color(self, room_id, color):
        # Встановлюємо колір освітлення в кімнаті
        cisco.set_lighting_color(self.connection, room_id, color)

    def set_automatic_color_scheme(self, room_id, time_of_day):
        # Автоматично змінюємо колір освітлення в залежності від часу доби
        if 6 <= time_of_day <= 12:
            self.set_lighting_color(room_id, "warm_white")
        elif 12 < time_of_day <= 18:
            self.set_lighting_color(room_id, "cool_white")
        else:
            self.set_lighting_color(room_id, "dim_white")

    def adapt_color_for_activity(self, room_id, activity):
        # Адаптація освітлення до активності
        if activity == "study":
            self.set_lighting_color(room_id, "cool_white")
        elif activity == "relax":
            self.set_lighting_color(room_id, "warm_white")
        else:
            self.set_lighting_color(room_id, "neutral")
```

Файл occupancy_detection.py

```
# occupancy_detection.py
import cisco_apis as cisco

class OccupancyDetectionSystem:
    def __init__(self, connection):
        self.connection = connection

    def detect_occupancy(self, room_id):
        # Виявлення наявності людей у приміщенні
        return cisco.get_occupancy_status(self.connection, room_id)

    def automatic_lighting_control(self, room_id):
        # Автоматичне керування освітленням на основі виявлення присутності
        if self.detect_occupancy(room_id):
            cisco.set_lighting(self.connection, room_id, 100)
        else:
            cisco.set_lighting(self.connection, room_id, 0)

    def automatic_shutdown(self):
        # Автоматичне вимкнення інфраструктури у разі відсутності людей
        rooms = cisco.get_all_rooms(self.connection)
        for room_id in rooms:
            if not self.detect_occupancy(room_id):
                cisco.set_lighting(self.connection, room_id, 0)
                cisco.set_temperature(self.connection, room_id, 18)
```

Файл climate_control.py

```
# climate_control.py
import cisco_apis as cisco

class ClimateControlSystem:
    def __init__(self, connection):
        self.connection = connection

    def set_humidity(self, room_id, humidity_level):
        # Встановлюємо рівень вологості у приміщенні
        cisco.set_humidity(self.connection, room_id, humidity_level)

    def monitor_climate_conditions(self, room_id):
        # Моніторимо кліматичні умови
        temperature = cisco.get_temperature(self.connection, room_id)
        humidity = cisco.get_humidity(self.connection, room_id)
        return {"temperature": temperature, "humidity": humidity}

    def optimize_climate(self, room_id, desired_temp, desired_humidity):
        # Оптимізація кліматичних умов для досягнення бажаних параметрів
        current_conditions = self.monitor_climate_conditions(room_id)
        if current_conditions["temperature"] > desired_temp:
            cisco.set_temperature(self.connection, room_id, desired_temp)
        if current_conditions["humidity"] < desired_humidity:
            self.set_humidity(room_id, desired_humidity)
```

Файл reporting_system.py

```
# reporting_system.py
import json

class ReportingSystem:
    def __init__(self, connection):
        self.connection = connection

    def generate_energy_report(self):
# Генерація звіту по енерговитратам
        energy_data = cisco.get_energy_data(self.connection)
        report = {
            "average_consumption": sum(energy_data) / len(energy_data),
            "max_consumption": max(energy_data),
            "min_consumption": min(energy_data)
        }
        with open("energy_report.json", "w") as report_file:
            json.dump(report, report_file)
        return report

    def generate_climate_report(self, room_id):
# Генерація звіту по клімату у приміщенні
        climate_data = cisco.get_climate_data(self.connection, room_id)
        report = {
            "average_temperature": sum(climate_data["temperature"]) /
len(climate_data["temperature"]),
            "average_humidity": sum(climate_data["humidity"]) /
len(climate_data["humidity"])
        }
        with open(f"climate_report_{room_id}.json", "w") as report_file:
            json.dump(report, report_file)
        return report

    def generate_security_report(self):
# Генерація звіту по безпеці
        security_events = cisco.get_security_events(self.connection)
        report = {
            "total_events": len(security_events),
            "critical_events": len([e for e in security_events if e["severity"]
== "critical"])
        }
        with open("security_report.json", "w") as report_file:
            json.dump(report, report_file)
        return report
```