

Центральноукраїнський національний технічний університет
Механіко-технологічний факультет
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 автоматизації
будинку на основі технології ВІоТ”

КБПЗ - 2024

Виконав здобувач вищої освіти
ІІ курсу, групи КІ-23М
ОПП «Комп’ютерна інженерія»
спеціальності 123 «Комп’ютерна інженерія»
_____ Серeda O.O.
« ____ » _____ 2024 р.

Керівник проекту
кандидат технічних наук, доцент
_____ Дреєв O.M.
« ____ » _____ 2024 р.
Рецензент _____

Центральноукраїнський національний технічний університет
Факультет Механіко-технологічний
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

Середі Олександр Олександровичу

(прізвище, ім'я, по батькові)

1. Тема роботи *Дослідження та програмна реалізація системи автоматизації будинку на основі технології ВІоТ*

2. Керівник роботи *Дреєв Олександр Миколайович, канд. техн. наук, доцент*
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу № 19-13 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 автоматизації будинку на основі технології ВІоТ*

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

- | | |
|--|--|
| <i>1. Призначення та область використання.</i> | <i>6. Наукова новизна.</i> |
| <i>2. Перегляд аналогічних існуючих систем.</i> | <i>7. Маркетингове та економічне обґрунтування ІТ-проєкту.</i> |
| <i>3. Опис і обґрунтування проектних рішень.</i> | <i>8. Заходи з охорони праці та техніки безпеки.</i> |
| <i>4. Етапи програмування системи.</i> | <i>9. Висновки.</i> |
| <i>5. Впровадження системи в промислову експлуатацію</i> | |

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

- | | |
|--|-----------------|
| <i>Наукова новизна</i> | <i>1 аркуш</i> |
| <i>Структурна схема системи</i> | <i>1 аркуш</i> |
| <i>Функціональна схема системи</i> | <i>1 аркуш</i> |
| <i>Діаграма процесів</i> | <i>1 аркуш</i> |
| <i>Блок-схема алгоритму роботи додатку</i> | <i>2 аркуша</i> |
| <i>Показники економічної ефективності</i> | <i>1 аркуш</i> |

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

Середа О.О. Дослідження та програмна реалізація системи автоматизації будинку на основі технології ВІоТ. 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 автоматизації будинку на основі технології ВІоТ.

Метою розробки є дослідження та програмна реалізація системи автоматизації будинку на основі технології ВІоТ.

Об'єктом дослідження є процес автоматизації будинку на основі технології ВІоТ.

Предметом дослідження є методи автоматизації будинку на основі технології ВІоТ.

Методи дослідження базуються на методах Інтернету речей, методах математичної статистики, методах розробки програмного забезпечення.

Результат роботи – програмна реалізація системи автоматизації будинку на основі технології ВІоТ.

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, автоматизація будинку, ВІоТ

ABSTRACT

Sereda O.O. Research and software implementation of a home automation system based on BIoT technology. 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 a home automation system based on BIoT technology.

The goal of development is research and software implementation of a home automation system based on BIoT technology.

The object of research is the process of home automation based on BIoT technology.

The subject of research is home automation methods based on BIoT technology.

Research methods are based on Internet of Things methods, mathematical statistics methods, and software development methods.

The result of the work is the software implementation of the home automation system based on BIoT technology.

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, home automation, BIoT

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ	3
ВСТУП.....	4
1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ	6
1.1 Призначення системи.....	6
1.2 Область застосування.....	11
2 ПЕРЕГЛЯД АНАЛОГІЧНИХ ІСНУЮЧИХ СИСТЕМ	12
2.1 Огляд існуючих систем, технологій, архітектур та програмних рішень за профілем теми випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти.....	12
2.2 Обґрунтування вибору засобів для побудови системи та мови програмування.....	19
2.3 Розгорнута постановка завдання	21
3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ	23
3.1 Опис функціонування системи	23
3.2 Розробка структурної схеми.....	32
3.3 Розробка функціональної схеми	36
3.4 Розробка діаграми процесів.....	39
4 РЕАЛІЗАЦІЯ РОБОТИ. РОЗРАХУНКИ І ЕКСПЕРИМЕНТАЛЬНІ ДАНІ, ЩО ПІДТВЕРДЖУЮТЬ ВІРНІСТЬ ПРОЕКТНИХ ТА ПРОГРАМНИХ РІШЕНЬ.....	41
4.1 Розробка блок-схем та опис алгоритмів функціонування системи.....	41
4.2 Захист розробленого програмного забезпечення.....	57
5 ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ	58
6 НАУКОВА НОВИЗНА	64

						ВКРМ-123.24.0040.00.00.ПЗ		
Вим	Арк.	№ докум.	Підп.	Дата		Літ.	Аркуш	Аркушів
Розроб.	Середа О.О.				Дослідження та програмна реалізація системи автоматизації будинку на основі технології ВІоТ	М	1	91
Перев.	Дресв О.М.							
Н.контр.	Коваленко А.С.					ЦНТУ КІ-23М		
Затв.	Смірнов О.А.							

7	МАРКЕТИНГОВЕ ТА ЕКОНОМІЧНЕ ОБҐРУНТУВАННЯ ІТ-ПРОЄКТУ	65
7.1	Визначення цільової аудиторії кінцевого готового продукту	65
7.2	Оцінка привабливості шляхом застосування методів експертних оцінок ...	66
7.3	Вибір методу оцінки вартості ПЗ	67
7.4	Розрахунок економічної ефективності від впровадження реалізованого ПЗ як фактору його привабливості.....	68
7.5	Пропозиція алгоритму просування проєкту розробки ПЗ	68
7.6	Оптимізація каналів збуту та шляхів реалізації ПЗ	71
7.7	Визначення ключових факторів успіху конкретного проєкту.....	72
8	ЗАХОДИ З ОХОРОНИ ПРАЦІ ТА ТЕХНІКИ БЕЗПЕКИ	74
8.1	Вступ.....	74
8.2	Аналіз умов праці на робочому місці ІТ-фахівця	75
8.3	Пропозиції щодо підвищення працездатності ІТ-фахівця	77
8.4	Пожежна безпека	79
8.5	Розрахункова частина	80
9	ОСНОВНІ ВИСНОВКИ.....	83
	СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	85

КБПЗ-2024

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		2

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ

АРМ	—	автоматизоване робоче місце
АСУ	—	автоматизована система управління
ДБЖ	—	джерело безперебійного живлення
ДКС	—	домашня кабельна мережа
ДУ	—	дистанційне управління
ЕОМ	—	електронно-обчислювальна машина
ЕФ	—	екранна форма
ЗТО	—	звукова трансляція й оповіщення
ІБ	—	інтелектуальний будинок
ІЧ	—	інфрачервоний
ОВК	—	управління опаленням, вентиляцією й кондиціонуванням
ОДС	—	оперативна диспетчерська система
ОПС	—	охоронно-пожежна сигналізація
ПДУ	—	пульти дистанційного управління
ПЗ	—	програмне забезпечення
ПЛК	—	програмувальні логічні контролери
ПМО	—	програмно-математичного забезпечення
РК	—	рідкокристалічний
СКК	—	система кабельних комунікацій
ТЗ	—	технічне завдання
ЕІВ	—	європейська інсталяційна шина
X10	—	технологія інтелектуального дому

ВСТУП

Актуальність теми. У жодному з визначень інтелектуального будинку не піддається сумніву, що ключовим елементом такого будинку є сучасна мережна інфраструктура, що забезпечує підключення й взаємодію різних систем. Протягом багатьох років різні організації, компанії-виробники й окремі експерти вживали спроби сформулювати універсальне визначення «інтелектуального» будинку. У результаті на світло з'явилася безліч визначень, у яких «інтелект» будинку розглядається часом з кардинально різних точок зору. Проте в жодному з них не піддається сумніву, що ключовим елементом такого будинку є сучасна мережна інфраструктура, що забезпечує підключення й взаємодію різних систем. Важливою тенденцією в області побудови «інтелектуальних» будинків є організація єдиної комунікаційної інфраструктури, здатної підтримувати функціонування провідної й бездротової мережі, а також різних інженерних систем і додатків. Наявність такої інфраструктури спрощує обмін інформацією в системах «людина – людина», «людина – пристрій» і «пристрій – пристрій» як у межах будинку, так і при взаємодії із зовнішнім миром. Вона забезпечує функціонування провідної локальної мережі, Wi-Fi, стільникового зв'язку усередині будинку, а також передачу аудіо- і відеоданих, підключення різних датчиків і виконавчих механізмів, роботу систем керування будинком.

Мета й завдання дослідження. Метою роботи є дослідження та програмна реалізація системи автоматизації будинку на основі технології ВІоТ.

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

- Огляд існуючих систем автоматизації будинку на основі технології ВІоТ.
- Дослідження системи автоматизації будинку на основі технології ВІоТ.
- Програмна реалізація системи автоматизації будинку на основі технології ВІоТ.

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		4

Об'єктом дослідження є процес автоматизації будинку на основі технології ВІоТ.

Предметом дослідження є методи автоматизації будинку на основі технології ВІоТ.

Методи дослідження базуються на методах Інтернету речей, методах математичної статистики, методах розробки програмного забезпечення.

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

- Удосконалено метод автоматизації будинку на основі технології ВІоТ.
- Розроблено вітчизняний продукт автоматизації будинку на основі технології ВІоТ, який має більш широкі можливості, на відміну від існуючих аналогів.

Практична цінність отриманих результатів полягає в тому, що розроблені алгоритми дозволяють успішно вирішувати задачі автоматизації будинку на основі технології ВІоТ.

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 автоматизації будинку на основі технології ВІоТ, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти.

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		5

1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ

1.1 Призначення системи

Всі існуючі системи можна дуже узагальнено розділити на 3 категорії – це:

- Централізовані системи.
- Децентралізовані (шинні) системи.
- Системи, що працюють по радіоканалу й силовій проводці.

Централізовані система "Розумний будинок"

До них відносяться розробки компаній AMX, Crestron, Lutron і ін. Складаються із центрального контролера, панелей керування й безлічі виконавчо-командних блоків. Системи споконвічно розроблялися як домашні системи керування.

Центральний контролер у цій системі автоматизації "Розумний будинок" виконує функції "мозку" – до нього підключаються всі інші системи. Різні компоненти мають свої мікроконтролери, але програма взаємодії перебуває в одному – головному. Від головного контролера сигнали керування можуть іти до виконавців по різних каналах.

Переваги:

- Доскональне пророблення ядра й програмного забезпечення.
- Можливість зібрати в єдиний комплекс всі системи життєзабезпечення й звести керування ними на єдиний пульт.
- Можливість включати як виконавців пристрою від самих різних виробників "розумного будинку".
- Гарні засоби створення графічного інтерфейсу.
- Відмінно продумана робота з аудіо-відеотехнікою, мультирум системами, живим відео сигналом.

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		6

– Дуже респектабельний дизайн і функціональність панелей керування, впізнаваність і престижність.

Недоліки:

– Досить висока ціна.

– Для свого фізичного розміщення вимагає технологічних обсягів або технологічних приміщень, для установки 19" стійок і інших компонентів системи.

– У випадку виходу з ладу центрального контролера – вся система перестає працювати.

Проектування й інсталяція доступні тільки кваліфікованим фахівцям. Монтаж здійсимо в ході ремонту або будівлі будинку.

Децентралізовані (шинні) системи "Розумний будинок"

До них можна віднести системи Instabus (EIB), LonWorks, C-BUS, BacNet і ін.

Instabus (EIB)

Відмінністю шинні технології, зокрема системи Instabus (EIB) є те, що в системи "розумного будинку" немає головного керуючого центра.

Вся система складається із сенсорів і активаторів. Сенсори призначені для виявлення якої-небудь активності в будинку або виміру яких-небудь характеристик. Наприклад: натискання на кнопку, рух, зміну яскравості освітлення, вимір температури, вологості й т.д. Сенсори посилають телеграми активаторам (виконавчим пристроям), які, у свою чергу, виконують підходящі команди. Виконавчим пристроєм може бути світлорегулятор, реле керування жалюзі, клапан системи опалення й ін..

Кожний сенсор і виконавчий пристрій мають свій контролер, підключений до керуючої шини й діючий цілком самостійно, оскільки в нього закладена програма з таблицею керуючих сигналів.

За допомогою двожильного кабелю типу "кручена пара" або універсального кабелю забезпечується як електроживлення контролерів і

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		7

виконавчих пристроїв (24В, постійний струм), так і обмін керуючими сигналами за особливими правилами. Одна шина поєднує всі електричні пристрої будинку.

Керування шинною системою "Розумний будинок" здійснюється або із клавішних панелей, розташованих у певних приміщеннях, або централізовано – з пульта керування або комп'ютера. Головними перевагами цієї системи є її відкритість і здатність управляти більшою кількістю елементів, що підключаються.

Радіошинні системи (GIRA, LEGRAND, BTCINO)

Радіошина система – набір компонентів, використання яких дозволяє дистанційно (за допомогою радіосигналів) управляти різними джерелами світла (включати й виключати, а також регулювати освітлення), приводами жалюзі й іншими пристроями. Радіошина система, забезпечуючи зручний спосіб керування із застосуванням сучасних технічних рішень, розроблена спеціально для додаткового оснащення сучасною шинною технікою вже існуючих будинків і є ідеальним рішенням для невеликих об'єктів. Завдяки тому, що система не має потреби в створенні якої-небудь інфраструктури, вона становить інтерес для локального застосування. У будь-який час систему можна доповнити новими елементами, поступово підвищуючи комфортність будинку. Для установки радіошиної системи "Розумний будинок" не буде потрібно спеціального навчання і якихось спеціальних програмних засобів, тому що її програмування здійснюється досить просто.

Всі елементи радіошиної системи діляться на дві частини – радіопередаючі й радіоприймальні (виконавчі) пристрою. Принцип дії системи простий: радіопередаючий пристрій відправляє керуючу телеграму, всі виконавчі пристрої її приймають, але виконують команду тільки ті пристрої, які налаштовані на прийом цієї телеграми від даного радіопередаючого пристрою. Передавачі, що одержують живлення від батареї, роблять систему дуже гнучкою. Їх можна встановлювати навіть там, де немає проводів електромережі 230 В. Кожен, з виконавчих пристроїв може бути налаштований не більше ніж на 30

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		8

керуючих команд від різних передавальних пристроїв. Дальність дії системи становить 100 метрів у чистому полі.

MyHome BTICINO

Система MY HOME складається з декількох складових частин, які в комплексі контролюють всі необхідні для автоматизації будинку функції: забезпечення комфорту, безпека, енергозбереження, зв'язок і керування. Всі пристрої системи MY HOME використовують одну технологію, засновану на використанні загальної цифрової шини, завдяки чому компоненти різних частин можуть бути з'єднані воедино, згідно вашому бажанню й запитам. Модульність і можливість спільної установки різних частин системи позитивно позначається на оптимізації вартості системи. Крім цього, ви також завжди можете вибрати, яку функцію системи ви хочете використовувати зараз, і, яку функцію – у майбутньому. Більше того, система MY HOME також надає можливість зв'язуватися із зовнішнім миром за допомогою спеціальних пристроїв, які взаємодіють із системою за допомогою телефонних ліній, а також через локальний або глобальну (Інтернет) комп'ютерну мережу.

Переваги:

– Можливість реалізації в рамках однієї системи "Розумний будинок" і в одному дизайні з електроустановкою й керування інженерними системами, і багатозону систему озвучування й функції забезпечення безпеки.

Недоліки:

- Система закрита, тобто протокол, по якому спілкуються пристрої в системі, доступний тільки пристроям виробництва Vticino.
- Обмежена адресна ємність системи, застосовна тільки для невеликих проектів.
- Обмежені функції по керуванню деякими інженерними системами.

InOne by Legrand

Система керування на власному протоколі InOneByLegrand. Єдина із систем, що працює по існуючій проводці. Закритий протокол і унікальна система

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		9

передачі інформації із силових проводів забезпечує надійну роботу системи, як у приватних будинках, так і в офісних приміщеннях. Дозволяє управляти будь-якими інженерними системами будинку. Розрахована на автоматизацію щодо невеликих приміщень із меншими витратами як на встаткування, так і на монтаж. Програмування здійснюється як вручну, так і за допомогою комп'ютера.

Пристрою керування являють собою добре нам з вами знайомі механізми вимикачів, розеток і модульної апаратури з єдиною відмінністю: кожне з них містить у собі електронний пристрій. Вся система будується й проектується як стандартна електрична мережа, при цьому пристрої з'єднуються між собою стандартними електричними проводами. По проводам одночасно подається живляча напруга 220 В і інформаційно керуючий сигнал. Система "Розумний будинок" будується за принципом ведучий-ведомий. Можлива побудова різноманітних сцен і сценаріїв. Програмне забезпечення для роботи даної системи не використовується, оскільки, як ми вже сказали, всі пристрої керування являють собою модернізовані традиційні механізми.

Система може бути побудована з використанням трьох основних технологій, комплексно об'єднаних між собою:

- обмін інформацією із силової лінії з напругою 220 В;
- інфрачервона технологія;
- радіотехнологія.

Система In One by Legrand являє собою набір невід'ємних функцій, що забезпечують зниження витрат, зручність експлуатації й належний комфорт:

- керування освітленням;
- керування рольставнями й рухливими тентами;
- система сигналізації;
- технічна сигналізація;
- аудіо-, відеодомофоні системи;
- телефонна, інформаційна й ТБ-мережа;
- домашній Інтернет-сервер.

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		10

1.2 Область застосування

Звичайно ж, "розумним будинком" зараз називати систему домашньої автоматизації неправильно. У більшості випадків це саме система домашньої автоматизації – система, що допомагає заощадити часові ресурси на керування всіма іншими інженерними й розважальними системами. І чим більше систем, тим потрібніше буде використовувати систему автоматизації у вашій квартирі або заміському будинку.

Таким чином, виходячи з вищеперерахованого, дослідження та програмна реалізація системи автоматизації будинку на основі технології ВІоТ, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти.

КБПЗ – 2024

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		11

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

На початку 2024 року концепція розумного будинку перетворилася з футуристичної мрії на відчутну реальність. Завдяки розвитку технологій розумні будинки стають більш доступними, ефективними та інтегрованими, ніж будь-коли раніше.

У цьому посібнику розглядатимуться найновіші функції розумного дому, які стануть домінувати у 2024 році, і ви зможете зрозуміти, як перетворити свій житловий простір на осередок зручності та енергоефективності.

1. Голосові помічники

Голосові помічники вже кілька років є наріжним каменем розумних будинків, і їхні можливості продовжують розширюватися. У 2024 році ці помічники стануть більш інтуїтивно зрозумілими, оперативнішими та інтегрованими з більшим набором пристроїв. Такі популярні опції, як Amazon Alexa, Google Assistant і Siri від Apple, стали ще більш досконалими, пропонуючи безперебійний контроль над домашнім середовищем.

Ці помічники можуть краще розуміти контекст, дозволяючи вести більш природні розмови. Наприклад, ви можете запитати: «Яка сьогодні погода?» і запитайте: "Чи потрібна мені парасолька?" без повторення контексту. Крім того, вони можуть керувати більш широким набором інтелектуальних пристроїв, від світильників і термостатів до кухонних приладів.

2. Управління енергією та стійкість

Зі зростанням занепокоєння щодо зміни клімату та зростання цін на енергію управління енергією та сталість стали критично важливими питаннями

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		12

для власників будинків. Технологія розумного дому в 2024 році пропонує інноваційні рішення, які допоможуть вам зменшити вуглецевий слід і заощадити на рахунках за електроенергію.

Розумні термостати, такі як Nest Learning Thermostat і Ecobee, стали розумнішими. Вони вивчають ваші вподобання та відповідно регулюють температуру. Вони також можуть інтегруватися з прогнозами погоди для оптимізації опалення та охолодження. Системи розумного освітлення, такі як Philips Hue, дозволяють контролювати яскравість і колір світла, зменшуючи споживання енергії.

Крім того, системи управління енергією розумного будинку можуть відстежувати споживання енергії в режимі реального часу, надаючи інформацію про те, як можна зменшити споживання. Ці системи також можна інтегрувати з відновлюваними джерелами енергії, такими як сонячні батареї, щоб максимізувати ефективність і стійкість.

3. Розумна техніка

Кухню часто вважають серцем дому, і розумна техніка змінює те, як ми готуємо їжу, прибираємо та керуємо нашими повсякденними справами. У 2024 році розумні холодильники, духовки, посудомийні та пральні машини стануть більш підключеними та інтуїтивно зрозумілими, ніж будь-коли.

Розумні холодильники тепер можуть відстежувати ваші продукти, пропонувати рецепти на основі інгредієнтів і навіть розміщувати замовлення на продукти, що закінчуються. Розумними духовками можна керувати дистанційно, дозволяючи розігрівати їх по дорозі додому або отримувати сповіщення, коли ваша їжа буде готова. Посудомийні та пральні машини можуть оптимізувати використання води та енергії, забезпечуючи більш ефективне та екологічне рішення.

4. Домашні розважальні системи

Домашні розваги також досягли значного прогресу завдяки інтеграції інтелектуальних технологій. У 2024 році смарт-телевізори, звукові системи та

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		13

застосування.

Датчики руху, такі як Philips Hue Motion Sensor і Samsung SmartThings Motion Sensor, можуть виявляти рух і запускати дії на основі ваших уподобань. Наприклад, ви можете налаштувати датчик руху, щоб вмикати світло при вході в кімнату або надсилати сповіщення при виявленні руху в певній зоні.

Екологічні датчики, такі як Awair Element і Netatmo Weather Station, можуть контролювати якість повітря, температуру, вологість та інші фактори навколишнього середовища. Ці датчики надають дані в реальному часі та можуть ініціювати дії для підвищення комфорту та безпеки у вашому домі. Наприклад, ви можете налаштувати автоматику на вмикання очищувача повітря, коли якість повітря падає нижче певного рівня.

11. Інтеграція «Розумного дому» з носіями

Носимі технології стають все більш популярними, і в 2024 році вони запропонують повну інтеграцію з системами розумного дому. Розумні годинники, фітнес-трекери та інші пристрої, які можна носити, тепер можуть взаємодіяти з пристроями вашого розумного дому, забезпечуючи більш персоналізований і зручний досвід.

Наприклад, такі розумні годинники, як Apple Watch і Samsung Galaxy Watch, можуть безпосередньо керувати вашими розумними домашніми пристроями з вашого зап'ястка. Ви можете відрегулювати термостат, увімкнути світло або замкнути двері простим дотиком або голосовою командою. Носимі пристрої також можуть надавати дані про здоров'я та фізичну форму вашій системі розумного дому, забезпечуючи більш персоналізовану автоматизацію та процедури.

12. Розумний дім підключення та мережі

Надійне підключення та підключення до мережі необхідні для безперебійної роботи розумного дому. У 2024 році прогрес у технології Wi-Fi та мережевих рішеннях спростив підключення та керування кількома розумними пристроями.

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		17

Висновок

Функції розумного будинку 2024 року пропонують широкий спектр переваг, від підвищення енергоефективності до покращеної зручності та персоналізації. Використовуючи ці технології, ви можете перетворити свій дім на сучасний, підключений, інтелектуальний житловий простір, який відповідає вашим потребам і вподобанням. Незалежно від того, чи хочете ви оновити наявний розумний дім чи почати з нуля, цей посібник містить вичерпний огляд останніх тенденцій та інновацій у технології розумного дому.

2.2 Обґрунтування вибору засобів для побудови системи та мови програмування

Як мова програмування обрана Python. Python – високорівнева мова програмування загального призначення з акцентом на продуктивність розроблювача й читаність коду. Синтаксис ядра Python мінімалістичний. У той же час стандартна бібліотека включає великий обсяг корисних функцій.

Python підтримує кілька парадигм програмування, у тому числі структурне, об'єктно-орієнтоване, функціональне, імперативне й аспектно-орієнтоване. Основні архітектурні риси – динамічна типізація, автоматичне керування пам'яттю, повна інтроспекція, механізм обробки виключень, підтримка багатопоточні обчислень і зручні високорівневі структури даних. Код у Python організовується у функції й класи, які можуть поєднуватися в модулі (які у свою чергу можуть бути об'єднані в пакети).

Еталонною реалізацією Python є інтерпретатор CPython, що підтримує більшість активно використовуваних платформ. Він поширюється вільно під дуже ліберальною ліцензією, що дозволяє використовувати його без обмежень у будь-яких застосунках, включаючи пропрієтарні. Є реалізації інтерпретаторів для JVM (з можливістю компіляції), MSIL (з можливістю компіляції), LLVM і інших. Проект PyPy пропонує реалізацію Python на самому Python, що зменшує витрати

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		19

на зміни мови й постановку експериментів над новими можливостями. Python – мова програмування, що активно розвивається, нові версії (з додаванням/зміною мовних властивостей) виходять приблизно раз у два з половиною року. Внаслідок цього й деяких інших причин на Python відсутні ANSI, ISO або інші офіційні стандарти, їхню роль виконує CPython.

Python портований і працює майже на всіх відомих платформах – від КПК до мейнфреймів. Існують порти під Microsoft Windows, практично всі варіанти UNIX (включаючи FreeBSD і Linux), Plan 9, Mac OS і Mac OS X, iPhone OS 2.0 і вище, Palm OS, OS/2, Amiga, AS/400 і навіть OS/390, Symbian і Android.

При цьому, на відміну від багатьох портуємих систем, для всіх основних платформ Python має підтримку характерних для даної платформи технологій (наприклад, Microsoft COM/DCOM). Більше того, існує спеціальна версія Python для віртуальної машини Java – Jython, що дозволяє інтерпретаторові виконуватися на будь-якій системі, що підтримує Java, при цьому класи Java можуть безпосередньо використовуватися з Python й навіть бути написаними на Python. Також кілька проектів забезпечують інтеграцію із платформою Microsoft .NET, основні з яких – IronPython і Python.Net.

Python підтримує динамічну типізацію, тобто тип змінної визначається тільки під час виконання. Тому замість «присвоювання значення змінної» краще говорити про «зв'язування значення з деяким ім'ям». У Python є убудовані типи: бульові, рядки, Unicode-рядки, цілі числа довільної точності, числа із плаваючою комою, комплексні числа й деякі інші. З колекцій Python підтримує кортежі (*tuples*), списки, словники (асоціативні масиви) і, починаючи з версії 2.4, безлічі. Всі значення в Python є об'єктами, у тому числі функції, методи, модулі, класи.

Додати новий тип можна або написавши клас (*class*), або визначивши новий тип у модулі розширення (наприклад, написаному мовою C). Система класів підтримує спадкування (одиначне й множинне) і метапрограмування. Можливе спадкування від більшості убудованих типів і типів розширень.

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		20

Всі об'єкти діляться на посилальні й атомарні. До атомарного ставляться `int`, `long`, `complex` і деякі інші. При присвоюванні атомарних об'єктів копіюється їхнє значення, у той час як для посилальних копіюється тільки покажчик на об'єкт, таким чином, обидві змінні після присвоювання використовують те саме значення. Посилальні об'єкти бувають змінювані й незмінні. Наприклад, рядки й кортежі є незмінними, а списки, словники й багато інших об'єктів – змінюваними. Кортеж у Python є, по суті, незмінним списком. У багатьох випадках кортежі працюють швидше списків, тому якщо ви не плануєте змінювати послідовність, то краще використовувати саме їх.

Мова має чіткий і послідовний синтаксис, продуману модульність й масштабованість, завдяки чому вихідний код написаних на Python програм легко читаємий.

Python – стабільна й розповсюджена мова. Він використовується в багатьох проектах і в різних якість: як основна мова програмування або для створення розширень і інтеграції застосунків. На Python реалізоване велика кількість проектів, також він активно використовується для створення прототипів майбутніх програм. Python використовується в багатьох великих компаніях.

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 автоматизації будинку на основі технології ВІоТ.

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

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		21

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.

КБПЗ-2024

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		22

3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ

3.1 Опис функціонування системи

Комфорт, що може надати розумний будинок, зрівняти із чимсь дуже важко. Це можливість автоматизувати будь-які дані відео й охоронної системи спостереження, світло, опалення, а також можливості медіа. Крім цього до системи можна підключити будь-яку побутову техніку на вибір, все це входить у можливості розумного будинку.

Затишний комфорт розумних технологій

Більшість людей навіть не представляють, як багато повсякденних справ можна перекласти на автоматизовані системи. Наприклад, при від'їзді можна задати програму поливати квіти, контролювати обігрів житла, кількість снігу на даху або в інших місцях. Може проводитися кондиціонування приміщень у жаркий період або обігрів у холодний. Клімат контроль, налаштований під конкретного клієнта дозволяє не тільки жити у звичних умовах, але й підтримувати їх навіть під час своєї відсутності, якщо в будинку перебувають екзотичні рослини або тварини.

Переваги:

– Дана система розумний будинок може контролювати той об'єкт, що побажає вказати користувач. Набрати воду в басейн, включити лазню до приїзду, перед пробудженням скип'ятити чайник, включити тостер – все це легко за допомогою розумного будинку.

– При цьому контролюється вся техніка і її справність. При несправній техніці вона просто не ввімкнеться, а система пошле сигнал «sos» хазяїнові у вигляді дзвінка або смс.

– Оптимальне освітлення, температура, інші комфортні умови, все це буде підібрано з побажаннями клієнта.

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		23

– Економія електроенергії при такому підключенні може скласти до 40%.

Раніше така система вимагала спеціальної проводки, що було проблемою для власників будинків і квартир, тому що робити ремонт заново хотілося не всім, а потрібна було укладання перешкодозахищеного радіо кабелю. Зараз же всі подібні утруднення в минулому.

Унікальна Ні-Фі система допомагає управляти будь-яким приладом без штроблення стін. Звичайні вимикачі, постачені радіопередавачами, можуть замінити всю складну систему проводки.

Охорона будинку й периметра

Особливо добре допомагають можливості автоматизації при несанкціонованому проникненні в житло або на територію ділянки. Контролери відразу сповістять, що в будинку сторонній. Програма може сповістити хазяїна, сфотографувати порушника, викликати поліцію, заблокувати двері або виконати ті команди, які в неї закладені. Обов'язково ведеться відеоспостереження, що дозволить зафіксувати зловмисників.

Та ж картина відбувається при витокі газу або води, пожежі або іншій проблемі. Датчики можуть заблокувати водопровід, щоб уникнути затоплення. Автоматично викличуть потрібну службу, повідомлять власника про виниклу проблему. При виникненні задимлення спрацьовують датчики гасіння пожежі, після чого блокується електроенергія. При події номер, що заданий в охоронній системі, набирається автоматично, і якщо він зайнятий, то набирається наступний.

Можливості для створення затишку:

– При наявності в будинку дітей може бути записано аудіо або відео послання для них від батьків.

– При приході персоналу пролунають розпорядження по збиранню або обслуговуванні будинку.

– При скупченні гостей може автоматично включатися кондиціонер, музика, прилад або освітлення на кухні.

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		24

Насправді все, що хоче власник запрограмувати, він це зможе зробити сам, якщо в нього будуть підключені датчики й налагоджений вся система контролю. Природно установку такого програмного забезпечення й інших датчиків краще довірити фахівцям, тому що вони можуть гарантувати, що всі функції системи розумний будинок працюють нормально.

Цінові пропозиції

У середньому у вартість системи до 50 тисяч гривень можуть входити такі послуги як:

– GSM канал – спрацьовування охоронної й пожежної сигналізації, відключення енергопостачання.

– СМС – обігрів, освітлення, вентиляція.

– Керування розетками за допомогою пульта.

Від 50 до 250 тисяч гривень перелік значно розширюється:

– Керування електропостачанням і відключення його при необхідності при від'їзді власника.

– Домофон і відеоспостереження.

– Попередження про можливу аварію.

Від 250 до 600 тисяч гривень мають на увазі такі пропозиції:

– Охоронна й пожежна сигналізація із захватом периметра.

– Відеоспостереження шляхом інтернету.

– Автоматизоване керування освітленням.

– Автоматика вхідних воріт.

– Аварійні повідомлення про витік води, газу й т.д.

Від 600 до 1 млн. 200 тисяч гривень:

– Керування всіма системами єдиною системою контролю

– Можливість керування із сенсорних панелей і КПК.

– Індивідуальний підрозділ керування опаленням, вентиляцією, теплою підлогою й т.д. у кожній кімнаті.

– Контроль якості повітря і його вологості (ручне або автомат).

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		25

- Керування розетками й освітленням (ручне або автомат).
- Керування каналізацією й водопостачанням.

Від 1 млн.200 тисяч гривень:

- Контроль промерзання зовнішнього кабелю і його обігрів.
- Обігрів стоків при необхідності щоб уникнути бурульок і полою.
- Мультирум.
- Повна автоматизація всіх систем у будинку.
- Контроль і керування через інтернет.

Дані цінові діапазони наведені орієнтовно для приблизного розуміння ситуації. Також та або інша послуга може зустрічатися й у більше недорогому ціновому сегменті – все залежить від особливостей, складності й кількості необхідних пристроїв у системі.

Розрахунок і технічне завдання

Проектування розумного будинку краще планувати заздалегідь, щоб не тільки врахувати всі потреби замовника, але й виконати їх у найбільш якісному виді. Технічне завдання на розрахунок компанії часто пропонують безкоштовно, тому довідатися, у скільки обійдеться та або інша установка контролюючих систем можна швидко й без особливих витрат. Фахівці роз'яснять, що саме можна зробити й проконтролювати, а також якими можливостями володіє дана система розумного будинку.

Досить сказати, що цифрова інтеграція є частиною нашого способу життя. Від смартфонів до соціальних медіа, від фінансів до фітнесу, прориви в технологіях постійно влітаються в наші повсякденні справи. Здавалося б, план із майбутнього, розумні будинки є продовженням нашого прагнення до зручності, із зростаючими можливостями, які дозволяють власникам будинків перетворювати свої житлові простори на чутливе середовище, яке повністю відповідає їхнім потребам.

Автоматизовані прилади, передові системи безпеки, моніторинг здоров'я та інші технічні зручності можуть здатися фантазією, але правда полягає в тому,

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		26

що розумні будинки швидко стають нашою реальністю. Перетворюючи наше повсякденне життя, персоналізуючи комфорт, інновації в домашньому господарстві спрямовані не так на вау-фактор, а більше на те, щоб полегшити наше життя завдяки легкому контролю.

Отже, якщо ви шукаєте способи покращити своє оточення та звільнити час для занять, які вам справді подобаються, давайте розглянемо головні тенденції розумного дому, на які ви можете звернути увагу у 2024 році.

Пріоритет сталого розвитку

Пам'ятаєш, як повертався додому, але зрозумів, що не ввімкнув світло? Або інтуїція підказує вам, що спринклери потрібно відрегулювати перед відпусткою? Або будь-який із сотні подібних неприємностей, які викликають стрес і марну енергію? З розумними будинками стає зеленим, зробити внесок у екологічні зусилля легше, ніж будь-коли.

Інноваційні функції, починаючи від інтелектуальних систем освітлення до енергоефективних приладів і збереження води, можуть перетворити ваші житлові простори з неорганізованих споживачів на стійкі гавані. Якщо ви прагнете зменшити вплив на навколишнє середовище, зверніть увагу на розумні термостати, які автоматично регулюють температуру залежно від кількості людей, подовжувачі живлення, які вимикають живлення неактивних пристроїв, системи освітлення, які вмикаються лише тоді, коли це потрібно, водонагрівачі, які підлаштовуються під ваші потреби. візерунки, прилади з енергозберігаючими функціями, інтуїтивно зрозуміле оновлення вікон, інтеграція сонячної енергії або стійкі будівельні матеріали. Сприяйте ефективному та невимушеному способу життя, впровадивши ці оновлення у свій дім.

Звичайно, це не можна зробити за одну ніч і коштує недешево. Але це не повинно вас знеохочувати, оскільки ці функції можна впроваджувати поступово, починаючи з доступних і простих у використанні варіантів і поступово переходячи до більш вимогливих покращень. У довгостроковій перспективі інвестиції в екологічно чисті розумні будинки з низьким рівнем впливу корисні

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		27

як для планети, так і для вашого гаманця; початкові витрати можуть бути окуплені з часом завдяки значному скороченню ваших рахунків за електроенергію.

Моніторинг здоров'я

Сучасний спосіб життя змушує нас все більше і більше зосереджуватися на здоров'ї та благополуччі. Довгий час житлові приміщення не були частиною рівняння; це було більше про фітнес-трекери, катушки для харчування та програми для усвідомленості. Однак розумні будинки стають надійним і проактивним партнером у нашій подорожі до здоров'я, пропонуючи функції, які дають нам можливість брати на себе відповідальність за своє благополуччя. Ваша скромна вітальня може перетворитися на повноцінний центр моніторингу. як?

Подумайте про розумні матраци, які відстежують якість вашого сну, частоту серцевих скорочень і дихання, щоб ви знали, чому іноді вранці відчуваєте себе погано. Подумайте про розумні ваги, які вимірюють вашу вагу, відсоток жиру в організмі та структуру м'язової маси, тобто точні статистичні дані вдома. Подумайте про домашні системи, які підлаштовують плани тренувань на основі даних, отриманих із носимих пристроїв. Для тих, хто має хронічні захворювання, дозатори ліків, які видають ліки у встановлений час, або монітори артеріального тиску, які попереджають вас про можливі проблеми, можуть виявитися неоціненними. Монітори якості повітря, термометри чи системи освітлення, які імітують природні цикли, сприяють більш здоровому та комфортному середовищу життя. Керуйте своїм благополуччям, не виходячи з дому.

Керування голосом і жестами

Концепція розумного дому може хвилювати багатьох, але постійна взаємодія з програмами, сенсорними екранами та датчиками може бути ідеальною не для всіх. На щастя, для тих, хто шукає спосіб навігації по дому без використання рук, доступні варіанти взаємодії. Елементи керування голосом і жестами виходять у центр уваги, забезпечуючи ще ширший діапазон керування.

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		28

Це так само просто, як запрограмувати слово, щоб увімкнути світло, налаштувати температуру або засунути штори, щоб швидко подрімати, не возячись із кнопками. Використання голосових команд для активації складних процедур забезпечує рівень контролю та налаштування, з яким традиційні методи не можуть зрівнятися. Управління жестами, з іншого боку, робить крок далі: вкажіть на пристрій, щоб увімкнути його, проведіть пальцем, щоб відрегулювати гучність, або старим добрим плеском, щоб вимкнути світло. Для людей з обмеженою рухливістю простим помахом можна закрити жалюзі або відрегулювати термостат, а нахил голови призупинить перегляд фільму. Це дає їм змогу жити більш незалежно та комфортніше у своїх розумних будинках.

Слово мудрому. Якими б революційними вони не були, все ще є місце для вдосконалення: фоновий шум може впливати на голосові команди, тоді як складні жести можуть неправильно інтерпретуватися. Голосовим помічникам потрібен доступ до даних користувача, тому ознайомтеся з політикою конфіденційності та керуйте налаштуваннями даних для плавної та безпечної інтеграції.

Передові системи безпеки

Міцні двері та собака, що гавкає, – перевірені часом рецепти безпеки будинку, але сьогодні вони просто не підійдуть. Будь то відпочинок на вихідних чи довгий день на роботі, власники будинків знають, що покинути дім або заховатися вночі, це викликає явне відчуття вразливості. Тож знати, що ви в безпеці від злому, надзвичайних ситуацій чи будь-яких інших тривог, справді, просто важливо. Саме тут на допомогу приходять розумні системи безпеки, які пропонують багаторівневий підхід до захисту вашого будинку.

Уважно стежачи за вашим приміщенням, поки вас немає, виявляючи потенційні проблеми до їх загострення або пропонуючи можливості дистанційного керування, сучасні системи безпеки оптимізовані, щоб бачити всередині та зовні, дозволяючи вам контролювати безпеку вашого дому, де б ви не були. Переживаєте через незамкнені двері, димову сигналізацію чи небажаних

						ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата			29

гостей? Розумні датчики в режимі реального часу повідомлять вам, чи є про що турбуватися. Прилад залишився запущеним, і ви не можете до нього підійти – програма перевірить його стан і вимкне його за вас. Вогні спрацювали через рух вночі? Камери на місці, знімають і записують. Ця проактивна та чуйна екосистема гарантує душевний спокій.

Роботизована допомога

Все почалося з роботизованих пилососів, але майбутнє автоматизації виходить далеко за межі бездоганної підлоги та ідеально звареної кави. Звісно, косити газон, чистити жолоб або витирати пил із будинку може бути приємно, але нещодавні досягнення дозволяють роботам взяти на себе роботу, коли вихідні здаються занадто короткими для напруженої роботи або, чесно кажучи, коли вам просто не хочеться.

Повторювані завдання, які поглинають наш час, є основними цілями роботизованої допомоги, що приносить відчуття легкості та ефективності у ваше повсякденне життя. Незалежно від того, чи це прання, приготування їжі чи миття підлоги, цих робіт можна налаштувати відповідно до ваших потреб і вподобань, виконуючи завдання, які вам найбільше не подобаються. Інтеграція цих технологій може означати створення цілісної екосистеми, яка справді персоналізована та відповідає поточним потребам вашого дому.

Коли ви досліджуєте ці роботизовані рішення, також варто розглянути їх ширшетенденції індустрії поліпшення дому, які формують майбутнє житлових приміщень.

Що стосується майбутнього, то воно рухається в проактивному напрямку: роботи, які передбачають ваші потреби та виконують завдання без ваших підказок. Уявіть, що вони запускають посудомийну машину після обіду або попередньо розігрівають духовку, коли ви йдете додому, поки ви відпочиваєте. Однак майбутнє розумних будинків із роботами полягає не лише в автоматизації; йдеться про створення простору, який дає вам змогу жити повноцінним життям.

						ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата			30

Автономні розумні пристрої

Вибір правильного циклу пральної машини, чищення преса для сміття або регулювання температури в духовці – це найчастіше клопітка. Не більше. **Розумні прилади** стають дедалі більш автономними, керуючи ними, а ви дбаєте про ширшу картину.

Скажімо, ви кидаєте білизну в машину, а розумна пральна машина, залежно від типу тканини та ваги, подбає про оптимальний цикл. Або ваш холодильник знає, що продукти в'януть, і рекомендує рецепт із їх використанням. Ці функції мають не стільки спільного зі зручністю, скільки оптимізацію та персоналізацію. По суті, це те, чим насправді є «розумний дім» – майбутнє, де технології дають змогу жити більш ефективним, комфортним і насиченим життям.

Персоналізація та підключення AI

В основі всіх цих тенденцій лежить складна мережа взаємопов'язаних систем, яка постійно розширюється. На перший погляд, йдеться про керування світлом і відтворення музики. Однак, дивлячись далі, розумний дім розроблений, щоб думати наперед, вивчаючи ваші звички, уподобання та рутину. Чим більше ви з ним взаємодієте, тим більше він навчається та адаптується, створюючи справді персоналізоване середовище життя.

Створені для вирішення проблем, ці алгоритми навчені розпізнавати шаблони та адаптуватися до ваших примх. Скажімо, це звичайний будній день – ви завжди вмикаєте світло й кавоварку о 7 ранку? Коли ви повертаєтеся додому, ви спочатку поливаєте свій сад чи вмикаєте музику? О котрій годині починає працювати сушарка? Ви віддаєте перевагу тьмяному освітленню під час перегляду фільму чи відкриваєте віконниці під час сну? Ви можете не знати, але ваш розумний дім може. І це, в двох словах, те, як підключення розширює можливості даних, а дані дозволяють персоналізувати.

3.2 Розробка структурної схеми

Від BAS до ВІоТ

Об'єднання й підключення датчиків і інших пристроїв у рамках єдиної інфраструктури, так само як і організація міжмашинних комунікацій, є фундаментальними передумовами концепції Інтернету, що набирає популярність, речей (IoT). Стосовно до систем інтелектуальних будинків останнім часом все частіше говорять про Інтернет речей у будинках – Building Internet of Things (ВІоТ). Експерти Метаногі визначають ВІоТ як «накладену ІР-мережу, що забезпечує підключення всіх систем будинку, а також моніторинг, аналіз і контроль цих систем без втручання людини». Вони прогнозують, що в найближчі п'ять років традиційні системи автоматизації будинків (BAS) будуть еволюціонувати в напрямку ВІоТ, що, у свою чергу, забезпечить підвищення рівня «інтелекту» будинку й, як наслідок, ефективності його використання.

Очевидно, що, коли датчики, електронні компоненти й засоби підключення до мереж додаються в різні пристрої для вилученого одержання даних і керування, загальна ефективність функціонування будинку підвищується. Багато промислових і комерційних об'єктів уже використовують ВІоТ в обмежених масштабах, підключаючи до єдиної мережі засобу опалення, вентиляції й кондиціонування (ОВК), освітлення й інші системи будинку. Однак поки в мережу об'єднані всього близько 1% пристроїв, установлених у будинках. Щоб повністю реалізувати потенціал ВІоТ, потрібно приєднати інші пристрої за допомогою Ethernet, стільникової мережі, Bluetooth з низьким енергоспоживанням (Bluetooth LE), ZigBee, Wi-Fi або інших протоколів, роблячи вибір з урахуванням конкретного завдання.

Переваги єдиної інфраструктури

У міру розвитку мережних технологій з'являються нові можливості по інтеграції систем для керування нерухомістю, ІТ і інженерним устаткуванням у єдину мережну інфраструктуру, побудовану на базі кабельної системи з мідними

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		32

крученими парами. Так, наприклад, розвиток технології подачі електроживлення по мережі Ethernet (PoS) дозволяє вже сьогодні використовувати слабкострумову проводку для підключення світлодіодних світильників, а технологія HDBase-T відкриває дорогу передачі по тій же мережі високоякісного аудіо- і відеоконтента.

Загальна інфраструктура – економічно вигідний засіб підтримки безлічі різноманітних додатків. З погляду зручності експлуатації така інфраструктура переважніше набору різнорідних провідних і бездротових мереж, для кожної з яких потрібне окреме керування. Вирішуючи всі завдання по передачі даних і керуванню ними, єдина інтелектуальна магістраль дозволяє вдвічі скоротити капітальні витрати, а також зменшити поточні витрати.

Скорочення числа окремих мереж підвищує надійність і час безперебійної роботи. Крім того, загальна інфраструктура є гнучкою й адаптуємою, тому вона дозволяє легко замінити або розширити підключені системи відповідно до потреб бізнесу.

Універсальна зонна структура

Один з найбільш перспективних способів побудови єдиної мережі припускає організацію універсальної зонної структури – Universal Connectivity Grid (UCG). Цей підхід заснований на традиційному зонному принципі формування кабельної системи і є його подальшим розвитком.

При такому підході горизонтальна кабельна система формується із зон покриття, а не з кінцевих точок або конкретних пристроїв. Зонна організація підвищує гнучкість інфраструктури й зменшує загальну вартість володіння протягом життєвого циклу. Вона дозволяє власникові будинку й або орендареві максимально ефективно використовувати приміщення й різні системи, при цьому підключення нових робочих станцій і інших пристроїв здійснюється легко й просто. Така архітектура забезпечує побудову гнучкої й масштабованої інфраструктури, здатної задовольняти майбутні вимоги до підключень.

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		33

Універсальна структура UCG, як уже говорилося, розвиває ідею зонної архітектури: об'єкт розбивається по сітці на області обслуговування однакового розміру (осередку). Кожна з них містить точку консолідації (TE), що забезпечує підключення об'єктів до магістральної мережі у своїй області. Незалежно від того, чи прокладається сітка UCG при будівництві нового будинку або організується у вже існуючому, переміщення, додавання або зміна підключень здійснюється просто й зрозуміло. Зміни можна вносити без використання додаткових матеріалів і без витрати зайвих засобів, при цьому простої, що знижують продуктивність, мінімальні.

Точний вибір розміру осередків сітки UCG залежить від специфіки конкретного проекту, але в загальному випадку експерти CommScore радять керуватися рекомендаціями стандартів TR-24704 і TSB-162 щодо проектування кабельної системи для підключення точок доступу Wi-Fi.

В 2004 році міжнародні організації ISO і IEC представили технічний звіт TR-24704, де детально розглядалися питання «універсальної кабельної інфраструктури для підключення бездротових точок доступу». У ньому запропонована оптимальна схема розміщення таких точок. Як модель мережі приймається масив із прилягаючий друг до друга шестикутних стільників (осередків). Щоб забезпечити більше високу ємність, зона покриття кожної стільники обмежений радіусом 12 м. TR-24704 рекомендує розміщати телекомунікаційні розетки якнайближче до центра стільники. Тим самим досягається максимальна гнучкість в установці окремих точок доступу й забезпечується оптимальне покриття.

Незабаром після появи документа TR-24704 американська асоціація TIA представила свої рекомендації з реалізації кабельної інфраструктури для підключення бездротових точок доступу. Документ TIA TSB-162 припускає організацію сітки із квадратних осередків зі стороною 18 м. Такий підхід більше відповідає типовій забудові, він спрощує проектування й інсталяцію.

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		34

Число точок підключення в кожному осередку буде залежати від підтримуваних систем, типів і числа пристроїв, що підключаються, особливостей установленної в приміщенні меблів, розташування стін, дверей і безлічі інших факторів. При формуванні осередків меншого розміру число портів змінюється відповідно.

Можливий варіант, коли універсальна зонна структура UCG буде використовуватися тільки для підключення точок доступу бездротової мережі, а для пристроїв інших систем по старинці будуть застосовуватися окремі кабельні з'єднання «точка – точка». Однак такий підхід не дає повних переваг загальної мережної інфраструктури, описаної вище.

Доцільно задіяти точки консолідації для підключення пристроїв тих систем, які розміщуються в стельовій зоні. Це можуть бути блоки керування системами охолодження, вентиляції й кондиціонування, засоби контролю доступу в приміщення, камери відеоспостереження, а також низьковольтні світлодіодні світильники. При цьому для підключення ПК може використовуватися традиційна кабельна проводка локальної мережі, наприклад, організована під фальшполом.

Нарешті, кабельна інфраструктура на базі UCG буде дійсно єдиною, коли через точки консолідації будуть підключатися й пристрою, розташовані на робочих місцях користувачів, як показано на рис. 3.1.

Для передачі даних з різноманітних систем і підтримки зростаючих потреб у пропускній здатності загальна інфраструктура повинна мати високопродуктивні канали зв'язку. Щоб задовольнити ці запити й забезпечити можливість наступного розширення, для приєднання прикінцевих пристроїв варто використовувати горизонтальні кабелі Категорії 6А. Як магістральні кабелі (для підключення поверхових кросів до головного кросу) бажано використовувати волоконно-оптичні кабелі OM4.

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		35

– блок функцій, які відповідають за роботу з мультимедіа, тобто дозволяють по локальній мережі всередині будинку передавати дані на різного виду кінцеві пристрої (плазменний телевізор, сенсорні панелі й т.і.);

– блок функцій, які відповідають за безпеку, що дозволяє в автоматичному режимі знімати дані з датчиків і по мережі передавати їх до охоронного сервера, що приймає рішення, яку функцію виконати ;

– блок функцій, які відповідають за ручне управління інтелектуальним домом, тобто дозволяють через розроблене програмне забезпечення за допомогою пультів, або персонального комп'ютера управляти усіма блоками інтелектуального дому;

– блок функцій, які відповідають за автоматизацію, що дозволяють незалежно від людини виконувати роботи по підтримці дому у належному стані, тобто сервер автоматизації приймає рішення включати той або інший прибор.

Блок функцій, які відповідають за автоматизацію, включає в себе наступні функціональні блоки:

- макроси та сценарії;
- за температурою повітря;
- по руху людини;
- за рівнем освітленості;
- по сухому контакту;
- за часом доби.

Блок функцій, які відповідають за ручне управління інтелектуальним домом включає в себе наступні функціональні блоки:

- управління з екрану ЕОМ;
- релейне управління;
- диммування;
- «проходні» схеми вимикання;
- бездротове управління електрикою;
- для аудіо/відео та світла один пульт ДУ;
- віддалене управління за смартфоном.

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		37

Блок функцій, які відповідають за безпеку, включає в себе наступні функціональні блоки:

- звукова тривожна сигналізація;
- охоронна сигналізація;
- світлова тривожна сигналізація;
- тривожне оповіщення по смартону;
- світлова імітація наявності людей;
- функція «Паніка».

Блок функцій, які відповідають за роботу з мультимедіа, включає в себе наступні функціональні блоки:

- бездротове аудіо/відео;
- універсальні навчаємі пульти ІЧ ДУ;
- радіотрансляція ІЧ ДУ;
- бездротове аудіо/відео спостереження;
- радіоуправління ПЗ ЕОМ.

Розглянувши усі блоки функціональної схеми перейдемо до розгляду діаграми взаємодії процесів, які відбуваються у системі.

3.4 Розробка діаграми процесів

Діаграма взаємодії процесів використовується для візуалізації процесів обробки даних (структурне проектування). Для розробника вважається звичним спочатку креслити діаграму взаємодії процесів даних рівня контексту, завдяки чому буде показано взаємодію системи. Ця діаграма в подальшому підлягає уточненню шляхом деталізації процесів та потоків даних з метою показати систему що розробляється. Діаграма процесів розробленої системи зображена на рисунку 3.3. При детальному її розгляді можна побачити як саме проходить взаємодія у розробленій системі. Використовується модель проектування, графічне представлення «потоків» даних в інформаційній системі.

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		39

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 автоматизації будинку на основі технології ВІоТ.

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-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		41

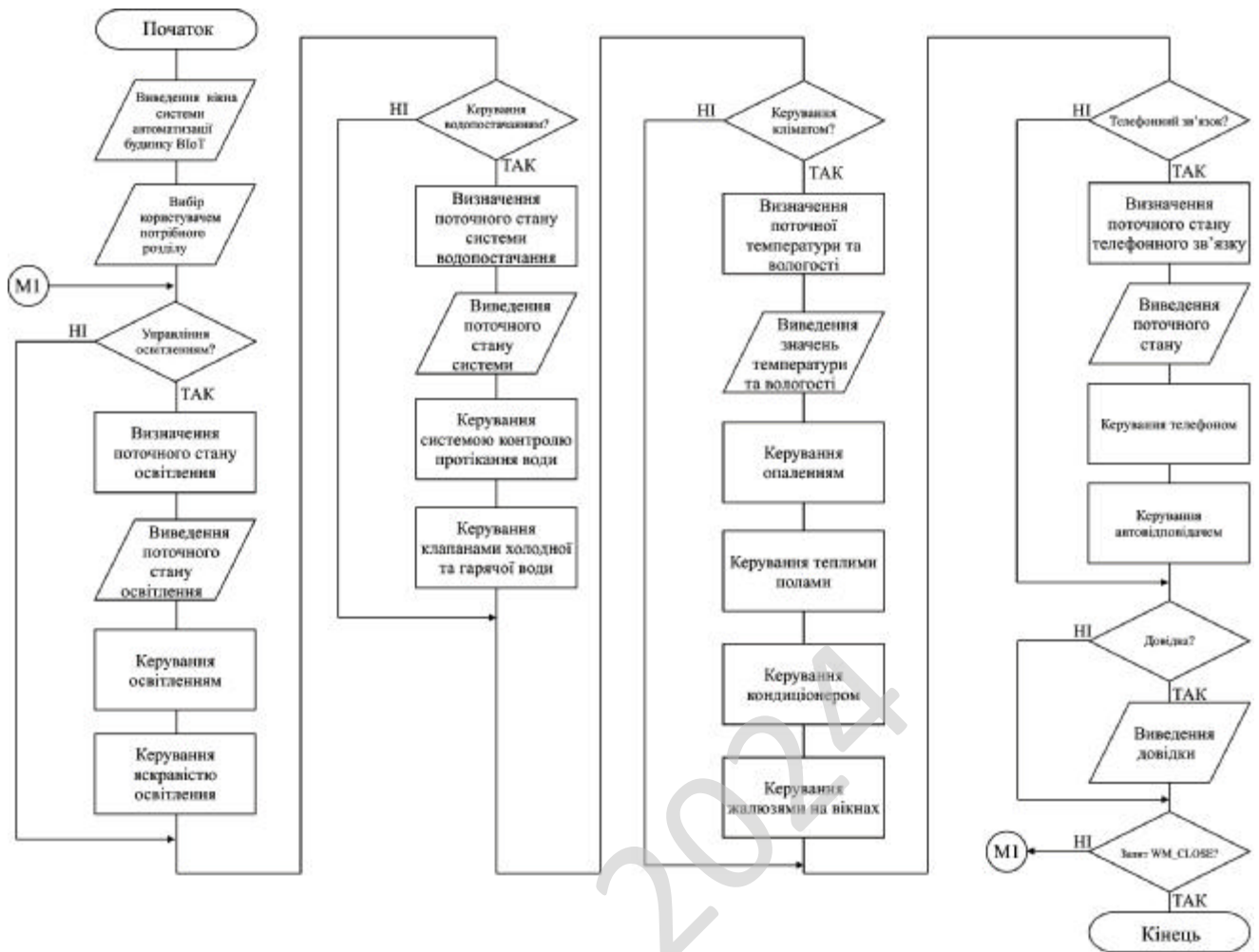


Рисунок 4.1 – Блок-схема основної програми

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, названої UML-моделлю.

UML був створений для визначення, візуалізації, проектування й документування в основному програмних систем. UML не є мовою

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

```
# Імпорт основних модулів для ВІoT системи
import time
import random

# Модуль для сенсорів
class Sensor:
    def __init__(self, name, unit, value=0):
        self.name = name
        self.unit = unit
        self.value = value

    # Метод для отримання показників сенсора
    def read(self):
        self.value = random.uniform(20, 25) # симуляція показників
        return self.value

# Модуль для керування приладами
class Device:
    def __init__(self, name, status=False):
        self.name = name
        self.status = status

    # Метод для увімкнення пристрою
    def turn_on(self):
        self.status = True
        print(f"{self.name} увімкнено")

    # Метод для вимкнення пристрою
    def turn_off(self):
        self.status = False
        print(f"{self.name} вимкнено")

# Модуль для контролера системи
class Controller:
    def __init__(self):
        self.sensors = []
        self.devices = []
```

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		44

```

# Метод для додавання сенсора
def add_sensor(self, sensor):
    self.sensors.append(sensor)

# Метод для додавання пристрою
def add_device(self, device):
    self.devices.append(device)

# Метод для моніторингу та прийняття рішень
def monitor_and_control(self):
    for sensor in self.sensors:
        value = sensor.read()
        print(f"{sensor.name}: {value} {sensor.unit}")
# Приклад рішення: якщо температура більша за 24 градуси,
# увімкнути кондиціонер
        if sensor.name == "Температура" and value > 24:
            for device in self.devices:
                if device.name == "Кондиціонер":
                    device.turn_on()

# Модуль для мережевих розрахунків системи ВІоТ
def calculate_network_load(sensors, devices):
    sensor_data_size = len(sensors) * 0.5
# кожен сенсор передає 0.5 КБ
    device_data_size = len(devices) * 0.2
# кожен пристрій передає 0.2 КБ
    total_data_size = sensor_data_size + device_data_size
    print(f"Загальне навантаження на мережу: {total_data_size} КБ")

# Створюємо об'єкти сенсорів
temperature_sensor = Sensor("Температура", "°C")
humidity_sensor = Sensor("Вологість", "%")

# Створюємо об'єкт пристрою
ac_unit = Device("Кондиціонер")

# Створюємо контролер
home_controller = Controller()

# Додаємо сенсори та пристрій до контролера
home_controller.add_sensor(temperature_sensor)
home_controller.add_sensor(humidity_sensor)

```

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		45

```

home_controller.add_device(ac_unit)

# Моніторинг системи
home_controller.monitor_and_control()

# Розрахунок навантаження на мережу
calculate_network_load(home_controller.sensors, home_controller.devices)

# Тестові дані для оцінки проектних рішень
def project_validation():
    # Додаємо тести для перевірки ефективності роботи системи
    average_temperature = sum([sensor.read() for sensor in home_controller.sensors
if sensor.name == "Температура"]) / len(home_controller.sensors)
    print(f"Середнє значення температури: {average_temperature} °C")

# Розрахунок енерговитрат
device_energy_use = 1.5 # Кондиціонер використовує 1.5 кВт/год
estimated_runtime = 5 # Пристрій працює 5 годин на добу
total_energy_use = device_energy_use * estimated_runtime

print(f"Загальні енерговитрати на день: {total_energy_use} кВт/год")

# Валідація проекту
project_validation()

```

Хоча сигнали управління ВІоТ передаються й приймаються безпосередньо по електричній мережі, для під'єднання пристроїв до комп'ютера можна використовувати різні топології, такі як наприклад Ethernet, RS-232, RS-485 та інші. Для вирішення поставленого у дипломному проекті завдання, було обрано інтерфейс Ethernet.

Стандарт ВІоТ визначає метод і протокол передачі керуючих сигналів-команд (включити, виключити, яскравіше, темніше й т.д.) по силовій електропроводці на електронні модулі, до яких підключені керовані електропобутові й освітлювальні прилади. Усього в мережу ВІоТ може бути об'єднане до 256 груп пристроїв з різними адресами.

З погляду логіки організації внутрімережної взаємодії всі пристрої ВІоТ можна розбити на дві більші групи: контролери й виконавчі модулі.

Контролери відповідають за генерацію команд ВІоТ і, крім ручного

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		46

кнопкового керування, можуть мати убудований таймер або спеціалізований пристрій уведення зовнішнього впливу (датчик освітленості, фотоприймач інфрачервоного випромінювання від пульта дистанційного керування й т.д.).

Виконавчий модуль, виконуючи команди, передані тим або іншому контролеру, управляє комутацією електроживлення побутового або освітлювального приладу, відіграючи роль "розумного" вимикача. Найпоширенішийо модулі двох типів: лампові (lamp module) і приладові (appliance module).

Конструктивно лампові модулі являють собою тиристорні регулятори потужності й забезпечують, крім функцій включення й вимикання, плавну регулювання яскравості світіння електроламп (функція диммера, від англійського слова dimmer – "реостат", "темнитель"). Приладові модулі оснащені електромагнітними реле для перемикаання живлення й не призначені для плавного регулювання подаваної на навантаження потужності.

Для передачі команди X10 потрібно одинадцять циклів (періодів) силової напруги. Перші два цикли передають стартовий код, наступні чотири цикли представляють код будинку (з А по Р) і останні п'яти циклів передають код приладу (з 1 по 16) або код функції (ВМК, ВИМК і т.д.), тобто ключовий код. Цей повний код (стартовий код + код будинку + ключовий код) завжди передається двічі безперервним блоком. Між блоками різних команд завжди повинен бути перерва в три цикли силової напруги. Виключенням із цього правила є блоки команд ЯСКРАВИШЕ/ТЕМНІШЕ, які передаються послідовно (мінімум два блоки) без затримок .

Усередині кожного блоку, код будинку й ключовий код повинні передаватися з кодами, що доповнюють до одиниці, у суміжних напівперіодах силової напруги. Наприклад, якщо одиничний імпульс переданий у першій половині періоду, то в другий не повинне бути ніякого сигналу (нульовий біт).

Нижче наведені можливі значення коду будинку й ключового коду і їхні двійкові подання. Стартовий код – це унікальний код, завжди рівний 1110 і не має

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		47

Команди ВІоТ

У протоколі ВІоТ передбачено шість базових команд:

1. Включити (On)
2. Виключити (Off)
3. Яскравіше (Bright)
4. Темніше (Dim)
5. Включити все світло (ALL Ligts ON)
6. Виключити всі (ALL Units OFF)

Сигнали ВІоТ

Технологія ВІоТ використовує цифрове подання сигналів керування. Інформація кодується двійковим кодом і передається по електричній мережі за допомогою високочастотних імпульсів. Кожний переданий імпульс відповідає одному біту інформації сї значенням “1”. Передача чергового імпульсу відбувається в момент часу, коли сіткова напруга приймає нульове значення.

Стандартна команда ВІоТ передається протягом, приблизно, 50 періодів мережної напруги частоти 50Гц або 1 сек.

Більшість переданих по мережі ВІоТ повідомлень містить, принаймні, два інформаційні поля: адреса пристрою, якому ця команда адресована, і властиво команду. Підключені до електромережі пристрої ВІоТ приймають передані повідомлення, декодують поле адреси й, якщо він збігається з їх власним, виконують команду.

Інтеграція ВІоТ

Основним засобом інтеграції мережі ВІоТ із зовнішнім устаткуванням є PLC інтерфейс ХМ10. Будь-який контролер, що підтримує відкритий протокол обміну з ХМ10, може відправляти команди ВІоТ в електромережу й, навпаки, одержувати з мережі інформацію про стан пристроїв ВІоТ.

Redmine – вільне серверне ПЗ для управління проектами та відстежування помилок. До системи входить календар-планувальник та діаграми Ганта для візуального представлення ходу робіт за проектом та строків виконання. Redmine

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		49

написано на мові Ruby і є ПЗ розробленим з використанням відомого веб-фреймворку Ruby on Rails, що означає легкість в розгортанні системи та її адаптації під конкретні вимоги. Для кожного проекту можна вести свої вікі та форуми.

Функціональні можливості:

- Ведення декількох проектів.
- Гнучка система доступу з використанням ролей.
- Система відстеження помилок.
- Діаграми Ганта та календар.
- Ведення новин проекту, документів та управління файлами.
- Сповіщення про зміни за допомогою RSS-потоків та електронної пошти.
- Власна Wiki для кожного проекту.
- Форуми для кожного проекту.
- Облік часових витрат.
- Налаштування власних (custom) полів для задач, затрат часу, проектів та користувачів.

– Легка інтеграція із системами керування версіями (SVN, CVS, Git, Mercurial, Vazaar и Darcs).

- Створення записів про помилки на основі отриманих листів
- Підтримка LDAP автентифікації.
- Можливість самореєстрації нових користувачів.
- Багатомовний інтерфейс (у тому числі українська мова).
- Підтримка СКБД: MySQL, PostgreSQL, SQLite.

Діаграма Ганта (Gantt chart, також стрічкова діаграма, графік Ганта) – це популярний тип діаграм, який використовується для ілюстрації плану, графіка робіт за будь-яким проектом. Є одним з методів планування та управління проектами.

Діаграма Ганта являє собою відрізки (графічні плашки), розміщені на горизонтальній шкалі часу. Кожен відрізок відповідає окремому завданню або

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		50

підзадачі. Завдання і підзадачі, складові плану, розміщуються по вертикалі. Початок, кінець і довжина відрізка на шкалі часу відповідають початку, кінцю і тривалості завдання. На деяких діаграмах Ганта також показується залежність між завданнями.

Діаграма може використовуватися для представлення поточного стану виконання робіт: частина прямокутника, що відповідає завданню, заштриховується, відзначаючи відсоток виконання завдання; показується вертикальна лінія, що відповідає моменту «сьогодні».

Часто діаграма Ганта використовується спільно з таблицею зі списком робіт, рядки якої відповідають окремо взятій задачі, зображеній на діаграмі, а стовпці містять додаткову інформацію про задачу.

Система відстеження помилок Багтрекер – прикладна програма для допомоги розробникам програмного забезпечення (програмістам, тестувальникам тощо) враховувати і контролювати помилки, знайдені у програмах, питання щодо функціональності, рішення та оновлення, побажання користувачів, а також стежити за процесом їх виконання.

Кожному, хто розробляв програмні продукти, добре знайоме співвідношення «20/80» – останні 20 % роботи тривають 80 % часу.

Як це не парадоксально, але нічого дивного в цій пропорції немає, адже саме на завершальній стадії починається тестування проекту, коли виявляються помилки, і що більший проект, то більше буде знайдено помилок.

Водночас досить часто виявляється, що більшість цих помилок були відомі та могли бути виправлені з меншими витратами на попередніх стадіях роботи, але не були вчасно описані, а потім загубилися серед інших важливих завдань.

Отже, система відстеження помилок у найпростішому варіанті – це процес, що включає в себе виявлення помилки, її опис, виправлення і перевірку цього виправлення, тобто процес «стеження» за багом протягом всього як його життєвого циклу, так і життєвого циклу розробки в цілому.

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		51

Сукупність інформації про дефект. Головний компонент такої системи – база даних, що містить відомості про виявлені дефекти. Ці відомості можуть включати в себе:

- номер (ідентифікатор) дефекту;
- хто повідомив про дефект;
- дата і час виявлення дефекту;
- версія продукту, в якій виявлено дефект;
- серйозність (критичність) дефекту та пріоритет рішення;
- опис кроків для відтворення дефекту (неправильної поведінки програми);
- відповідальний за усунення дефекту;
- обговорення можливих рішень та їх наслідків;
- поточний стан виправлення дефекту;
- версії продукту, в якій дефект виправлений.

Крім того, розвинені системи надають можливість прикріплювати файли, які допомагають описати проблему, наприклад, дамп пам'яті або скріншот.

Використання. Основна перевага систем відстеження помилок полягає в забезпеченні чітких централізованих оглядів, запитів на розробку (включаючи помилки і виправлення) та їх стан. У корпоративному середовищі, системи відстеження помилок можуть бути використані для генерації звітів по продуктивності програмістів виправлення помилок. Однак, це може іноді приводити до неточних результатів, тому що різні помилки можуть мати різні ступені пріоритету та серйозності, що пов'язано з складністю їх фіксації.

Життєвий цикл дефекту. Як правило, система відстеження помилок використовує той чи інший варіант «життєвого циклу» помилки, стадія якого визначається поточним станом помилки.

Типовий життєвий цикл дефекту:

1. Новий – дефект зареєстрований тестувальником.
2. Призначений – призначений відповідальний за виправлення дефекту.

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		52

3. Дозволений – дефект переходить назад у сферу відповідальності тестувальника. Як правило, супроводжується резолюцією, наприклад:

- Виправлено (виправлення включені у версію таку-то).
- Дубль (повторює дефект, що вже знаходиться в роботі).
- Не виправлено (працює відповідно до специфікації, має занадто низький пріоритет, виправлення відкладено до наступної версії тощо).
- «В мене все працює» (запит додаткової інформації про умови, в яких дефект проявляється).

4. Далі тестувальник проводить перевірку виправлення, залежно від чого дефект або знову переходить у стан «Призначений» (якщо він описаний як виправлений, але не виправлений), або у стан «Закрито».

5. Відкрито повторно – дефект знайдено знову в іншій версії.

Система може надавати адміністраторові можливість налаштування користувачі, які можуть переглядати і редагувати помилки залежно від їх стану, переводити їх в інший стан або видаляти.

У корпоративному середовищі, система відстеження помилок може використовуватися для отримання звітів, що показують продуктивність програмістів при виправленні помилок. Однак, часто такий підхід не дає достатньо точних результатів через те, що різні помилки мають різну ступінь серйозності та складності. При цьому серйозність проблеми прямо не стосується складності її усунення.

При розробці ПЗ було використано V-Model (або VEE модель) є моделлю розробки інформаційних систем (ІС), спрямованої на спрощення розуміння складнощів, пов'язаних з розробкою систем. Вона використовується для визначення єдиної процедури розробки програмного забезпечення, апаратного забезпечення та людино-машинного інтерфейсу.

Концепція V-подібної моделі була розроблена Німеччиною та США в кінці 1980-х років незалежно один від одного:

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		53

– Німецька V-модель була розроблена аерокосмічною компанією IAVG в Оттобрунні поряд з Мюнхеном у сприянні з Федеральним департаментом з закупівлі озброєнь в Кобленці, для Міністерства оборони Німеччини. Модель була прийнята німецькою федеральною адміністрацією для цивільних потреб влітку 1992.

– Американська V-Model (VEE) була розроблена національною радою з системної інженерії (міжнародна – з 1995 року) для супутникових систем, включаючи обладнання, програмне забезпечення та взаємодію з користувачами.

Сучасною версією V-Model є V-Model XT, яка була затверджена в лютому 2005 року. V-модель використовується для управління процесом розробки програмного забезпечення для німецької федеральної адміністрації.

Зараз вона є стандартом для німецьких урядових і оборонних проектів, а також для виробників ПЗ в Німеччині. V-Model являє собою скоріше набір стандартів у галузі проектів, що стосуються розробки нових продуктів. Ця модель багато в чому схожа з Prince2 і описує методи як для проектного управління, так і для системного розвитку.

Основні принципи

Основний принцип V-подібної моделі полягає в тому, що деталізація проекту зростає при русі зліва направо, одночасно з плином часу, і ні те, ні інше не може повернути назад. Ітерації в проекті виробляються по горизонталі, між лівою і правою сторонами літери.

Стосовно до розробки інформаційних систем V-Model – варіація каскадної моделі, в якій завдання розробки йдуть зверху вниз по лівій стороні букви V, а завдання тестування – вгору по правій стороні букви V. Усередині V проводяться горизонтальні лінії, що показують, як результати кожної з фаз розробки впливають на розвиток системи тестування на кожній із фаз тестування.

Модель базується на тому, що приймально-здавальні випробування ґрунтуються, насамперед, на вимогах, системне тестування – на вимогах та

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		54

архітектури, комплексне тестування – на вимогах, архітектурі та інтерфейсах, а компонентне тестування – на вимогах, архітектурі, інтерфейсах та алгоритмах

Цілі

V-модель забезпечує підтримку у плануванні та реалізації проекту. В ході проекту ставляться такі завдання:

1. Мінімізація ризиків: V-подібна модель робить проект більш прозорим і підвищує якість контролю проекту шляхом стандартизації проміжних цілей і опису відповідних їм результатів та відповідальних осіб. Це дозволяє виявляти відхилення в проекті і ризики на ранніх стадіях і покращує якість управління проектом.

2. Підвищення та гарантії якості: V-Model – стандартизована модель розробки, що дозволяє домогтися від проекту результатів бажаної якості. Проміжні результати можуть бути перевірені на ранніх стадіях. Універсальне документування полегшує читаність, зрозумілість та контрольованість.

3. Зменшення загальної вартості проекту: Ресурси на розробку, виробництво, управління і підтримку можуть бути заздалегідь прораховані та проконтрольовані. Отримувані результати також універсальні і легко прогножуються. Це зменшує витрати на подальші стадії та проекти.

4. Підвищення якості комунікації між учасниками проекту: Універсальний опис усіх елементів та умов полегшує взаєморозуміння всіх учасників проекту. Таким чином, зменшуються неточності у розумінні між користувачем, покупцем, постачальником і розробником.

Переваги:

– Користувачі V-Model беруть участь у розробці та підтримці V-моделі. Комітет з контролю за змінами підтримує проект і збирається раз на рік для обробки всіх отриманих запитів на внесення змін до V-Model.

– На старті будь-якого проекту V-подібна модель може бути адаптована під цей проект, так як ця модель не залежить від типів організацій та проектів.

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		55

– V-model дозволяє розбити діяльність на окремі кроки, кожен з яких буде включати в себе необхідні для нього дії, інструкції до них, рекомендації та докладне пояснення діяльності.

Обмеження. Наступні моменти не враховуються в V-моделі, але можуть бути розглянуті окремо, або можливо адаптувати модель під них:

– Не регулюється розміщення контрактів на обслуговування.

– Організація і виконання управління, обслуговування, ремонту та утилізації системи не враховуються в V-моделі. Однак, планування і підготовка до цих операцій моделлю розглядаються.

– V-подібна модель більше стосується розробки програмного забезпечення в проекті, ніж всієї організації процесу.

Переваги:

– У моделі особливе значення надається плануванню, спрямованому на верифікацію та атестацію розроблювального продукту на ранніх стадіях його розробки. Фаза модульного тестування підтверджує правильність деталізованого проектування. Фази інтеграції та тестування реалізують архітектурне проектування або проектування на вищому рівні. Фаза тестування системи підтверджує правильність виконання етапу вимог до продукту і його специфікації.

– У моделі передбачені атестація та верифікація всіх зовнішніх і внутрішніх отриманих даних, а не тільки самого програмного продукту.

– У V-подібної моделі визначення вимог виконується перед розробкою проекту системи, а проектування ПО – перед розробкою компонентів.

– Модель визначає продукти, які повинні бути отримані в результаті процесу розробки, причому кожен отриманий дани повинні піддаватися тестуванню.

– Завдяки моделі менеджери проекту можуть відслідковувати хід процесу розробки, так як в даному випадку цілком можливо скористатися тимчасовою шкалою, а завершення кожної фази є контрольною точкою.

Недоліки:

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		56

- Модель не передбачає роботу з паралельними подіями.
- У моделі не передбачено внесення вимоги динамічних змін на різних етапах життєвого циклу.
- Тестування вимог в життєвому циклі відбувається занадто пізно, внаслідок чого неможливо внести змін, не вплинувши при цьому на графік виконання проекту.
- У модель не входять дії, спрямовані на аналіз ризиків.
- Деякий результат можна отримати тільки при досягненні низу букви V.

4.2 Захист розробленого програмного забезпечення

Для захисту розробленого програмного забезпечення запропоновано використовувати алгоритм Khufu. Khufu – це 64-бітовий блоковий шифр. 64-бітовий відкритий текст спочатку розщеплюється на дві 32-бітові половини, L і R . Над обома половинами й певними частинами ключа виконується операція XOR. Потім, аналогічно DES, результати проходять деяку послідовність раундів. У кожному раунді молодший значущий байт L використовується як вхід S-блоку. У кожного S-блоку 8 вхідних біт і 32 вихідних біта. Далі обраний в S-блоці 32-бітовий елемент піддається операції XOR з R . Потім L циклічно зрушується на число, кратним восьми біткам, L і R міняються місцями, і раунд завершується. Сам S-блок не статичний, він міняється кожні вісім раундів. Нарешті, по закінченні останнього раунду, над L і R виконується операція XOR з іншими частинами ключа, і половини поєднуються, утворюючи блок шифртексту.

Хоча частини ключа використовуються для операції XOR із блоком шифрування на початку й кінці виконання алгоритму, головне призначення ключа – генерація S-блоків. Ці S-блоки секретні, по суті, це частина ключа. Повний розмір ключа алгоритму Khufu дорівнює 512 біт (64 байт), алгоритм надає спосіб генерації S-блоків по ключу.

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		57

5 МЕТОДИКА ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ

Розглянемо розроблене ПЗ системи автоматизації будинку на основі технології ВІоТ яке зображено на рисунку 5.1. Консольний додаток відповідає за виконання обчислень, а результати пересилаються в інтерфейс для подальшого використання. З рисунку можна побачити що інтерфейс головного вікна розподілено на наступні функціональні розділи:

- Навігаційне меню: Файл; Студент; Опції; Довідка.
- Функції представлені у графічному вигляді (іконки).
- Вікна обрання групи .
- Вікно виведення результату роботи системи.
- Навігаційного меню яке викликається натисканням правої клавіші маніпулятора миші.
- Функціональних кнопок ПЗ.

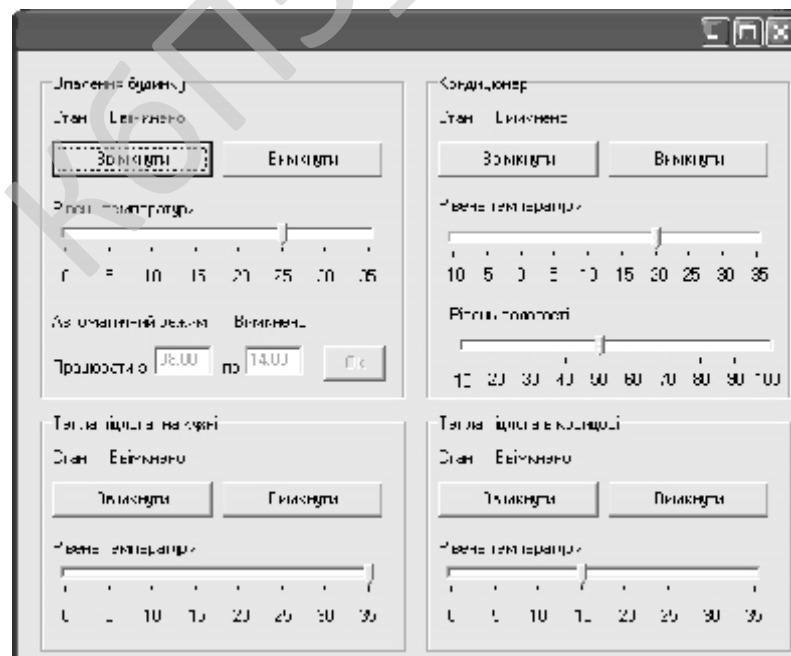


Рисунок 5.1 – Головне вікно ПЗ

Звичайно ж, "розумним будинком" зараз називати систему домашньої автоматизації неправильно. У більшості випадків це саме система домашньої автоматизації – система, що допомагає заощадити часові ресурси на керування всіма іншими інженерними й розважальними системами. І чим більше систем, тим потрібніше буде використовувати систему автоматизації у вашій квартирі або заміському будинку. 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.

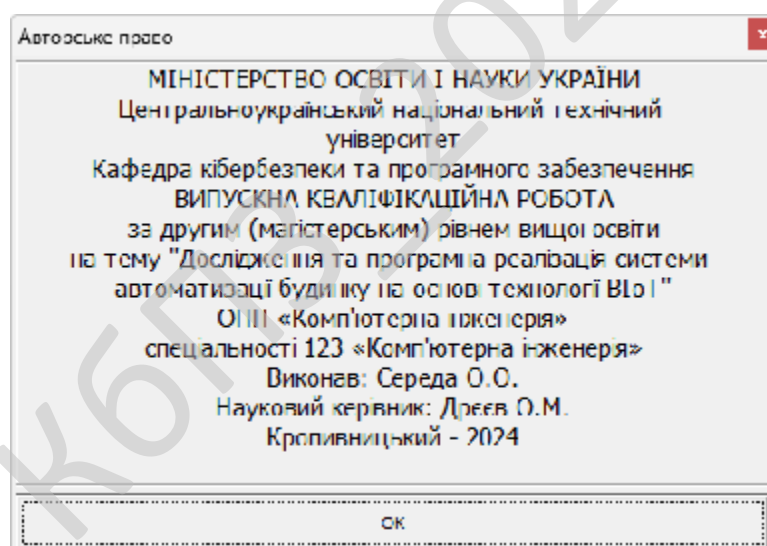


Рисунок 5.2 – Авторське право

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

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		59

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 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;

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		60

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

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		61

- Помилки інтерфейсу;
- Помилки у зовнішніх структурах даних або в доступі до зовнішньої бази даних;
- Помилки характеристик (необхідна ємність пам'яті і т.д.);
- Помилки ініціалізації та завершення.

Обрано умови розповсюдження – Shareware.

Під умовно-безплатним програмним забезпеченням можна розуміти спосіб або метод розповсюдження комерційного ПЗ на ринку (тобто на шляху до кінцевого користувача), при якому випробувачеві пропонується обмежена за можливостями (не повнофункціональна або демонстраційна версія), терміном дії (тріал версія) або версія з вбудованим набридливим нагадуванням про необхідність оплати використання програми.

В угоді про використання (ліцензії для кінцевого користувача, EULA) також може бути обумовлена заборона на комерційне або професійне (не тестове) її використання.

Основний принцип умовно-безплатного ПЗ – «спробуй, перш ніж купити» (try before you buy). ПЗ що поширюється як умовно-безплатний, надається користувачам безоплатно. Звичайно користувач платить тільки за час завантаження файлів через Інтернет або за носій (CD диск, флешку, ключ). Протягом певного терміну, що становить зазвичай тридцять днів, він може користуватися програмою, тестувати її, освоювати її можливості.

Якщо після закінчення цього терміну користувач вирішить продовжити використання ПЗ, він зобов'язаний купити його (zareєструватися), заплативши авторові певну суму. В іншому випадку користувач повинен припинити використання ПЗ та видалити його зі свого комп'ютера.

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		63

6 НАУКОВА НОВИЗНА

У випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти розроблено програмне забезпечення, яке призначено для системи автоматизації будинку на основі технології ВІоТ.

Метою розробки є дослідження та програмна реалізація системи автоматизації будинку на основі технології ВІоТ.

Об'єктом дослідження є процес автоматизації будинку на основі технології ВІоТ.

Предметом дослідження є методи автоматизації будинку на основі технології ВІоТ.

Методи дослідження базуються на методах Інтернету речей, методах математичної статистики, методах розробки програмного забезпечення.

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

- Удосконалено метод автоматизації будинку на основі технології ВІоТ.
- Розроблено вітчизняний продукт автоматизації будинку на основі технології ВІоТ, який має більш широкі можливості, на відміну від існуючих аналогів.

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		64

7 МАРКЕТИНГОВЕ ТА ЕКОНОМІЧНЕ ОБГРУНТУВАННЯ ІТ-ПРОЄКТУ

7.1 Визначення цільової аудиторії кінцевого готового продукту

Результати дослідження та програмної реалізації системи автоматизації будинку на основі технології ВІоТ можуть бути цікавими для теоретиків-дослідників цього питання, приватних власників житла, будівельних компаній, фахівців з електросистем та інших, що ми звели на рисунку 7.1.

Власники житла: Люди, які цікавляться поліпшенням комфорту та безпеки у своїх домівках, можуть знайти ці результати корисними для впровадження сучасних технологій.

Будівельні компанії: Компанії, що займаються будівництвом та ремонтом, можуть використовувати дані для інтеграції розумних технологій у нові проекти.

Електрики та системні інтегратори: Фахівці, які займаються встановленням та налаштуванням автоматизаційних систем, можуть отримати корисну інформацію про нові технології та рішення.

Дослідники та науковці: Люди, які займаються дослідженнями в галузі автоматизації, ІоТ та smart home технологій, можуть бути зацікавлені в отриманих результатах для подальшого аналізу та розвитку.

Постачальники технологій: Компанії, що виробляють компоненти або програмне забезпечення для автоматизації, можуть бути зацікавлені в нових рішеннях та інноваціях.

Споживачі енергоресурсів: Організації, що займаються управлінням енергетичними ресурсами, можуть використовувати результати для оптимізації споживання енергії в будинках.

Державні органи: Влада може бути зацікавлена в реалізації політик, що стосуються енергоефективності та розумних міст, на основі отриманих даних.

Рисунок 7.1 – Цільова аудиторія

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		65

Таким чином, результати вашого дослідження можуть знайти застосування в багатьох сферах, впливаючи на різні аспекти сучасного життя та технологічного прогресу.

7.2 Оцінка привабливості шляхом застосування методів експертних оцінок

Оцінка привабливості програмної реалізації системи автоматизації будинку на основі технології ВІоТ може бути здійснена за допомогою експертних оцінок. Перш ніж залучати експертів, потрібно визначити критерії, за якими буде проводитися оцінка привабливості проекту (рис. 7.2).

технічна доцільність: наскільки технологія біот відповідає вимогам проекту.

економічна ефективність: вартість впровадження та очікувана економія витрат (енергія, обслуговування тощо).

користувацький досвід: зручність та інтуїтивність системи для користувачів.

безпека: захищеність системи від зовнішніх загроз і витоків даних.

сумісність: можливість інтеграції з існуючими системами та пристроями.

Рисунок 7.2 – Критерії експертної оцінки

Далі залучаємо експертів у галузі автоматизації, ІоТ, програмування та споживчого досвіду. Кількість експертів може варіюватися, але бажано мати щонайменше 5-7 осіб для отримання більш об'єктивних результатів.

Кожен експерт оцінює визначені критерії за шкалою, наприклад, від 1 до 5, де: 1 – дуже низький рівень, 2 – низький рівень, 3 – середній рівень, 4 – високий рівень, 5 – дуже високий рівень. Експерти можуть заповнити анкету, в якій зазначають свої оцінки та обґрунтування.

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		66

Після збору оцінок від усіх експертів, можна обчислити середнє значення для кожного критерію.

Таблиця 7.1 – Зведені результати експертних оцінок

Критерій	Оцінка експерт 1	Оцінка експерт 2	Оцінка експерт 3	Середня оцінка
Технічна доцільність	4	5	4	4.33
Економічна ефективність	3	4	3	3.33
Користувацький досвід	5	4	4	4.33
Безпека	4	5	4	4.33
Сумісність	3	3	4	3.33

На основі отриманих середніх оцінок можна оцінити загальну привабливість проекту. Вищі оцінки свідчатимуть про високу привабливість реалізації системи автоматизації, тоді як нижчі можуть вказувати на потребу в доопрацюванні концепції.

На основі оцінок експертів можна сформулювати рекомендації щодо покращення проекту, зокрема, звернути увагу на ті критерії, які отримали нижчі оцінки, і розглянути шляхи їх поліпшення. Цей процес дозволяє не лише об'єктивно оцінити проект, а й залучити до нього фахівців з різних областей, що підвищує його якість та ефективність.

7.3 Вибір методу оцінки вартості ПЗ

Для оцінки вартості програмної реалізації системи автоматизації будинку на основі технології ВІоТ можна використовувати кілька методів: метод аналогій, експертних оцінок, метод часткових оцінок, метод триангулювання, метод функціонального оцінювання. Для проекту автоматизації будинку на основі ВІоТ,

ймовірно, найкраще буде використовувати метод часткових оцінок у поєднанні з методом аналогій. Це дозволить вам отримати як детальну, так і загальну оцінку вартості, що є важливим для таких складних систем. Залучення експертів також може підвищити точність оцінки, особливо в специфічних аспектах, пов'язаних з ВІоТ.

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

Економічна ефективність від впровадження системи автоматизації будинку на основі технології ВІоТ може бути проілюстрована через різні аспекти, такі як зниження витрат, підвищення комфорту та покращення енергоефективності. Проведемо розрахунок впровадження ВІоТ системи автоматизації будинку.

Впровадження системи автоматизації будинку на основі ВІоТ може забезпечити значні економічні вигоди, включаючи зменшення витрат на енергію та охорону, підвищення комфорту та зменшення витрат на обслуговування. За допомогою даного прикладу можна побачити, що інвестиції можуть окупитися протягом приблизно 6 років, після чого користувачі отримують чистий прибуток від системи.

7.5 Пропозиція алгоритму просування проєкту розробки ПЗ

Алгоритм просування проєкту програмної реалізації системи віртуалізації додатків у ОС Linux візуалізовано

Алгоритм просування проєкту програмної реалізації системи автоматизації будинку на основі технології ВІоТ. Цей алгоритм подано на на рисунок 7.3. і включає він етапи, які допоможуть залучити клієнтів, підвищити впізнаваність бренду та успішно реалізувати проєкт.

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		68

Таблиця 7.2 – Основні показники впровадження проєкту

Первинні інвестиції:	
1. Витрати на впровадження системи	
Кошти на обладнання (датчики, контролери, камер спостереження) – 2,500 USD	
Витрати на встановлення та налаштування системи – 1,000 USD	
Програмне забезпечення (ліцензії, налаштування) – 500 USD	
Загальні первинні витрати: 4,000 USD	
2. Очікувані щорічні заощадження	
Економія на витратах на енергію:	
Завдяки автоматизації освітлення та обігріву – 300 USD на рік	
Зниження витрат на охорону:	
Завдяки інтеграції системи безпеки (датчики руху, камери) – 200 USD на рік	
Зменшення витрат на обслуговування:	
Системи можуть передбачати поломки та запобігати їм, що дозволяє знизити витрати на ремонти – 150 USD на рік	
Загальні щорічні заощадження: 300 + 200 + 150 = 650 USD на рік	
3. Термін окупності	
Термін окупності = $\frac{650 \text{ USD/рік}}{4,000 \text{ USD}} \approx 6.15$ років	
Додаткові переваги	
Покращений комфорт: Користувачі можуть контролювати різні аспекти будинку (освітлення, температура, безпека) зі своїх смартфонів, що підвищує якість життя.	
Зниження викидів CO2: Енергоефективність системи допомагає зменшити викиди, що є важливим для навколишнього середовища.	
Підвищення вартості нерухомості: Наявність системи автоматизації може підвищити вартість нерухомості на ринку.	

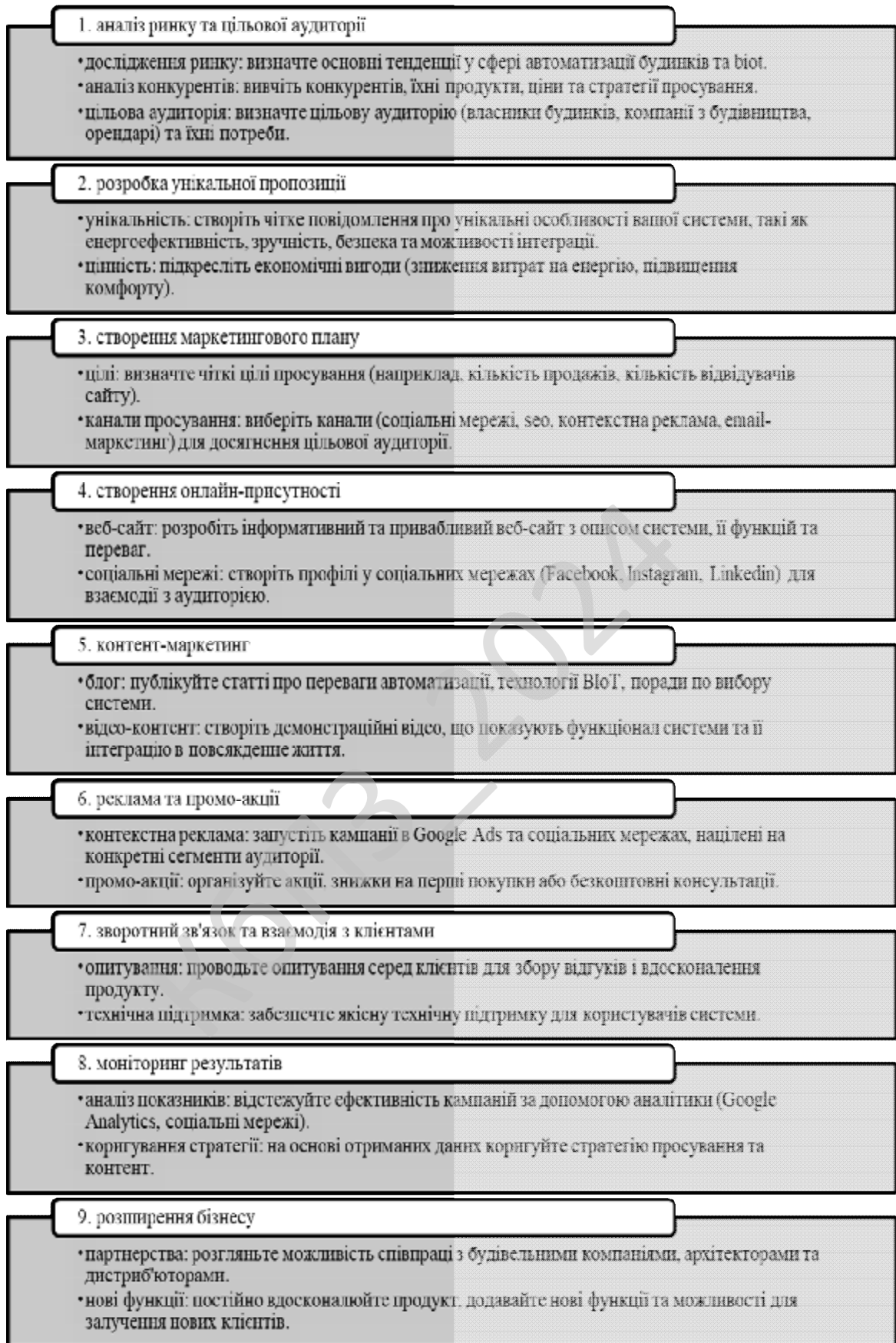


Рисунок 7.3 – Алгоритм просування проекту

Цей алгоритм дозволяє систематично підходити до просування проекту автоматизації будинку на основі технології ВІоТ, залучаючи цільову аудиторію та підвищуючи впізнаваність продукту.

7.6 Оптимізація каналів збуту та шляхів реалізації ПЗ

Оптимізація каналів збуту та шляхів реалізації проекту програмної реалізації системи автоматизації будинку на основі технології ВІоТ може суттєво підвищити ефективність продажів і задоволеність клієнтів:

- створіть власний інтернет-магазин, де користувачі можуть ознайомитися з продуктами, отримати консультації та зробити замовлення.
- співпрацюйте з компаніями, які вже мають досвід у продажу технологій biot, щоб розширити доступ до нових клієнтів.
- відкрийте демонстраційні стенди у магазинах електроніки чи будівельних супермаркетах, щоб потенційні клієнти могли безпосередньо ознайомитися з продуктом.
- реєструйтеся на популярних платформах (Amazon, Ebay, Etsy) для продажу ваших рішень. це дозволить залучити нових клієнтів через вже наявну аудиторію.
- використовуйте платформи, такі як Facebook, Instagram Та LinkedIn, для просування продукту, а також для організації кампаній з продажу через ці канали.
- забезпечте простоту замовлення на вашому сайті, щоб користувачі могли легко знайти потрібну інформацію та здійснити покупку.
- розробіть мобільний додаток, який дозволяє користувачам замовляти продукцію, отримувати консультації та отримувати підтримку.
- пропонуйте пакетні рішення, які включають кілька компонентів системи автоматизації. це може зменшити вартість для клієнтів і стимулювати їх до покупки більше одного продукту.

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		71

- забезпечте послуги установки та налаштування системи для підвищення зручності для клієнтів.
- створіть програми лояльності для постійних клієнтів, які передбачають знижки на повторні покупки або безкоштовні консультації.
- заохочуйте клієнтів рекомендувати ваші рішення іншим, надаючи їм знижки за кожного нового клієнта, якого вони залучать.
- організуйте вебінари та семінари, щоб навчити потенційних клієнтів користуватися вашою системою та її перевагами.
- створюйте інформаційний контент (блоги, статті, відео) про переваги ВІоТ і способи автоматизації, щоб зацікавити потенційних клієнтів.
- збирайте та аналізуйте відгуки клієнтів про ваш продукт, щоб виявити проблеми та можливості для вдосконалення.
- використовуйте аналітичні інструменти для моніторингу ефективності каналів збуту та виявлення найбільш продуктивних стратегій.
- впроваджуйте можливість моніторингу та контролю системи через смартфони чи інші пристрої для забезпечення зручності та простоти використання.

Оптимізація каналів збуту та реалізації проєкту дозволить вам не лише збільшити продажі, а й створити позитивний імідж компанії серед споживачів, підвищуючи їх задоволеність і лояльність.

7.7 Визначення ключових факторів успіху конкретного проєкту

Ключові фактори успіху проєкту (рисунок 7.4) програмної реалізації системи автоматизації будинку на основі технології ВІоТ можуть суттєво вплинути на результати та довгострокову життєздатність проєкту.

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		72

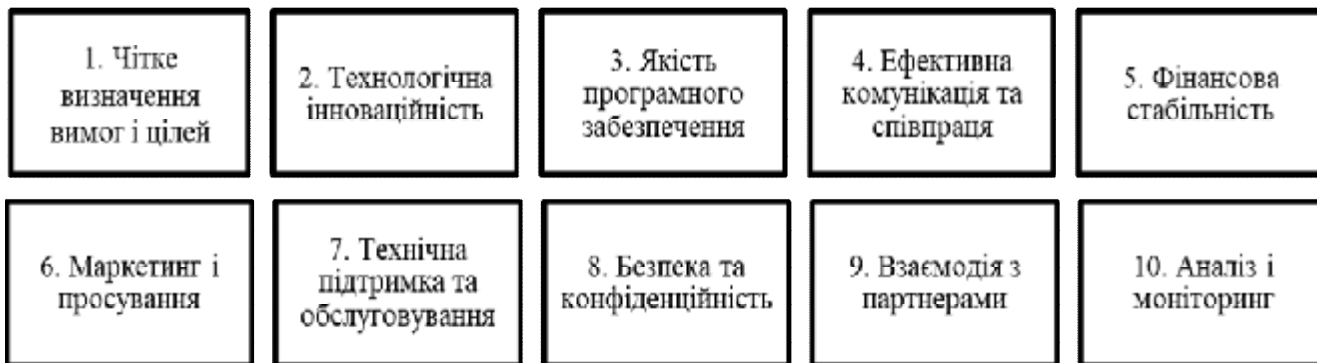


Рисунок 7.4 – Ключові фактори успіху проєкту

Врахування цих факторів може суттєво підвищити шанси на успіх проєкту програмної реалізації системи автоматизації будинку на основі технології ВІоТ, забезпечуючи зручність, безпеку та ефективність для користувачів.

КБПЗ_2024

8 ЗАХОДИ З ОХОРОНИ ПРАЦІ ТА ТЕХНІКИ БЕЗПЕКИ

8.1 Вступ

Охорона праці – система збереження життя і здоров'я працівників у процесі трудової діяльності, що включає правові, соціально-економічні, організаційні, технічні, санітарно-гігієнічні, лікувально-профілактичні, реабілітаційні та інші заходи. Загальні положення державної політики, щодо галузі охорони праці зазначені у Законі України “Про охорону праці”. Цей Закон визначає основні положення щодо реалізації конституційного права працівників на охорону їх життя і здоров'я у процесі трудової діяльності, на належні, безпечні і здорові умови праці, регулює за участю відповідних органів державної влади відносини між роботодавцем і працівником з питань безпеки, гігієни праці та виробничого середовища і встановлює єдиний порядок організації охорони праці в Україні [20]. Законодавство про працю містить норми і вимоги з техніки безпеки і виробничої санітарії, норми, що регулюють робочий час і час відпочинку, звільнення та переведення на іншу роботу, норми праці щодо жінок, молоді, гігієнічні норми і правила тощо. Загальний нагляд за додержанням норм охорони праці покладено на прокуратуру, спеціальний – на професійні спілки. Контроль за безпекою праці здійснюють також, державні й відомчі спеціалізовані інспекції. Науково-технічний прогрес вніс серйозні зміни в умови виробничої діяльності робітників розумової діяльності. Їх праця стала більш інтенсивною, напруженою і вимагає значних витрат розумової, емоційної і фізичної енергії. Це призвело до необхідності у знаходженні комплексного рішення проблем ергономіки, гігієни і організації праці, регламентації режимів праці та відпочинку. Охорона здоров'я робітників, забезпечення безпеки умов праці, ліквідація та профілактика професійних захворювань і виробничого травматизму складає одну з головних турбот людського суспільства.

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		74

8.2 Аналіз умов праці на робочому місці ІТ-фахівця

На робочому місці ІТ-фахівця (або програміста) виникають небезпечні та шкідливі для безпечної життєдіяльності фактори:

- підвищений рівень шуму;
- несприятливі мікрокліматичні умови;
- недостатній рівень освітленості;
- шкідливі речовини;
- підвищений рівень електромагнітних випромінювань радіочастот;
- висока напруга електричної мережі;
- статична електрика та інші.

Робота програміста супроводжується також підвищеним ступенем напруженості трудового процесу. При систематичному впливі виробничих факторів, які не відповідають нормативним показникам, зростає рівень професійно зумовленої захворюваності працюючих та можуть виникнути професійні захворювання органів зору, руху, нервової системи. Таким чином, вивчення умов праці на робочому місці програміста є необхідною умовою запобігання негативних наслідків впливу небезпечних та шкідливих факторів. Робоче місце, добре пристосоване до трудової діяльності інженера, правильно і доцільно організоване, щодо простору, форми, розміру забезпечує йому зручне положення при роботі і високу продуктивність праці при найменшому фізичному і психічному напруженні.

Нормування параметрів проводиться в залежності від періоду року та категорії важкості виконуваних робіт. Для постійних робочих місць, якими є робочі місця ІТ-фахівців, встановлені оптимальні параметри мікроклімату, а за неможливості їх дотримання використовують допустимі параметри. Робота ІТ-фахівця за важкістю відноситься до Іа (роботи, що виконуються сидячи і не потребують фізичного напруження) та Іб (роботи, що виконуються сидячи, стоячи або пов'язані з ходінням та супроводжуються деяким фізичним

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		75

Створення сприятливих умов праці і правильне естетичне оформлення робочих місць на виробництві має велике значення як для полегшення праці, так і для підвищення його привабливості, позитивно впливає на продуктивність праці. Забарвлення приміщень і меблів повинні сприяти створенню сприятливих умов для зорового сприйняття, гарного настрою. У службових приміщеннях, у яких виконується одноманітна розумова робота, що вимагає значної нервової напруги і великого зосередження, забарвлення повинно бути спокійних тонів – малонасичені відтінки холодного зеленого або блакитного кольорів.

При розробці оптимальних умов праці програміста необхідно враховувати освітленість. Раціональне освітлення робочого місця є одним з найважливіших факторів, що впливають на ефективність трудової діяльності людини, що попереджають травматизм і професійні захворювання. Правильно організоване освітлення створює сприятливі умови праці, підвищує працездатність і продуктивність праці. Освітлення на робочому місці програміста повинно бути таким, щоб працівник міг без напруги зору виконувати свою роботу. Стомлюваність органів зору залежить від ряду причин: недостатність освітленості; надмірна освітленість; неправильний напрям світла. Недостатність освітлення приводить до напруги зору, ослабляє увагу, приводить до настання передчасної стомленості. Надмірно яскраве освітлення викликає засліплення, роздратування і різь в очах. Неправильний напрямок світла на робочому місці може створювати різкі тіні, відблиски, дезорієнтувати працюючого. Всі ці причини можуть призвести до нещасного випадку або профзахворювань. [21]

8.3 Пропозиції щодо підвищення працездатності ІТ-фахівця

Практичне значення заходів щодо підвищення працездатності впливає із закономірностей її динаміки і зводиться ось до чого:

- збільшення фази стійкого стану у фонді робочого часу;
- прискорення процесу опрацювання;

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		77

- віддалення фази розвитку втоми;
- забезпечення високої продуктивності праці за нормальних фізіологічних затрат.

Комплекс заходів щодо підвищення і збереження працездатності працівників на оптимальному рівні реалізується на техніко-організаційному, соціально-економічному, санітарно-гігієнічному, медико-біологічному, психологічному напрямках.

Вагомим фактором високої працездатності і продуктивності праці є оптимізація трудових навантажень на основі механізації і автоматизації виробничих процесів, удосконалення технології, скорочення і ліквідації важкої ручної праці. Доведено, що при правильній організації праці на легких роботах спостерігається найбільша тривалість фази стійкого стану, а на важких роботах вона нетривала.

Високий рівень працездатності безпосередньо залежить від умов праці, оскільки поліпшення їх супроводжується зменшенням енергетичних затрат організму на подолання несприятливого впливу факторів виробничого середовища.

Важливим напрямком підвищення працездатності працюючих є ритмізація трудових процесів, оптимізація темпу роботи, а також раціоналізація трудових рухів на фізіологічній основі, що сприяє формуванню і закріпленню робочих динамічних стереотипів, а отже зменшенню м'язових і вольових зусиль. Ритмічна робота підвищує функціональні можливості організму, сприяє його тренуваності і забезпечує економізацію енергетичних затрат. [1]

Багатьом програмістам постійно доводиться працювати з великою кількістю програм одночасно. Часте перемикання туди-сюди між IDE та довідкою суттєво зменшує продуктивність фахівця. Однак вирішення цієї проблеми досить просте та очевидне: встановлення більшої кількості моніторів.

Оптимальним варіантом є два монітори. Все ж таки це найпростіший з апаратної точки зору варіант. Крім того, якби їх було більше, то ними було б

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		78

важче керувати, та й столі просто не вистачить місця на ще один монітор. Але тут ще залежить розміру моніторів. Є системи із 4 або 6 відносно невеликими екранами, які кріпляться на кронштейні. Але оптимальним є два 27-дюймові монітори, на яких все добре видно, особливо коли працювати доводиться в основному з текстом [3].

8.4 Пожежна безпека

Вимоги до пожежної безпеки на підприємстві неухильно повинен дотримуватися кожен співробітник, а організаційна складова при цьому покладається на посадових осіб за відповідним рішенням керівництва і прописується в посадових інструкціях і положеннях по структурним підрозділам.

Зокрема, вказуються конкретні території, ділянки, зони, об'єкти, цілі будівлі і їх частини, поверхи, на яких відповідального співробітника повинне проводити такі організаційні роботи.

Відповідальні особи зобов'язуються розробити, впровадити та підтримувати в певному інструкцією і положенням на ввірених їм об'єктах протипожежний режим і інструкції відповідно до вимог, викладених в нормативних актах.

Передбачено також створення підрозділу добровільної пожежної охорони та пожежно-рятувальної команди в його складі.

Встановлений режим включає порядки з описом місць спеціального призначення та правила їх користування та утримання, наприклад:

- евакуаційних шляхів;
- так званих «курилок»;
- місць складування продукції та сировини;
- стоянки транспорту.

Також встановлюється порядок роботи та технічного обслуговування:

- вентиляційного устаткування;

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		79

- засобів пожежогасіння і захисту від загорянь;
- нагрівальних приладів;
- електрообладнання.

Розробляються і впроваджуються правила роботи з відкритим вогнем і горючими матеріалами. Створюються графіки проходження інструктажів з пожежної безпеки співробітників, а також порядок і терміни перевірок знань пожежно-технічного мінімуму, в тому числі, тих працівників, які відповідальні за цю ділянку роботи на підприємстві. При цьому можуть передбачатися внутрішні лекції, семінари, тренінги та практичні заняття на підприємстві, а також зовнішні – на базі спеціалізованих навчальних центрів з професійними викладачами.

Важливою складовою протипожежного режиму на будь-якому об'єкті є розробка і впровадження порядку дій при виникненні пожежі. Неодмінно має бути план евакуації, описано, як повинні відключатися електроустановки, що і в якій послідовності необхідно робити співробітникам.

Відповідно, для кожного об'єкта, кожного приміщення (крім коридорів, санвузлів, басейнів і подібних приміщень), окремих видів робіт складаються інструкції, за якими повинен працювати персонал, залучений на певних ділянках і в виконанні окремих видів робіт. За інструкціями проводиться навчання (інструктаж) персоналу з подальшим контролем знань.

Детально про те, як розробити протипожежний режим, прописати порядки та інструкції, пояснюють на тематичних курсах і семінарах. [24]

8.5 Розрахункова частина

Проведемо розрахунок штучного освітлення за методом коефіцієнта використання світлового потоку для приміщення, ширина якого складає 6 м, довжина – 7 м, висота – 3,4 м.

У зазначеному приміщенні працює 7 людей.

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		80

коефіцієнтів використання світлового потоку (n) світильників з відповідним типом ламп) [8]. Підставимо всі значення у формулу, визначемо світловий потік: $F=90391$ Лм.

Для розрахунку дудемо використовувати світлодіодні стельові панелі *Призма-72 6400K*, світловий потік яких $F_{л} = 7200$ Лм.

Число ламп визначається по формулі:

$$N=F/F_{л}$$

де: F – світловий потік,

$F_{л}$ – світловий потік однієї лампи.

Підставимо всі значення у формулу та визначемо індекса приміщення:

$$N= 90391 / 7200=12,5 \text{ шт.}$$

Приймаємо необхідну кількість світлодіодних світильників 13 шт.

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 умов праці, призначеного для праці програмістів, проведено розгляд небезпечних та шкідливих факторів, що негативно впливають на програмістів під час роботи. Виконано розрахунок штучного освітлення, як одного з ключових факторів впливу на працездатність та здоров'я програміста. Розроблено заходи з умов поліпшення охорони праці.

					ВКРМ-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		82

9 ОСНОВНІ ВИСНОВКИ

Програмне забезпечення, створене в результаті виконання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти, призначено для системи автоматизації будинку на основі технології ВІоТ.

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 автоматизації будинку на основі технології ВІоТ.

Рішення даного завдання полягало у вирішенні наступних задач:

- Був проведений огляд існуючих систем автоматизації будинку на основі технології ВІоТ.
- Досліджена система автоматизації будинку на основі технології ВІоТ.
- На основі отриманих результатів досліджень створена програмна реалізація системи автоматизації будинку на основі технології ВІоТ.

Розроблені під час виконання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти алгоритми дозволяють успішно вирішувати завдання автоматизації будинку на основі технології ВІоТ.

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-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		83

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 Khufu.

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-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		84

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Серета О.О. Дослідження та програмна реалізація системи автоматизації будинку на основі технології ВІоТ // 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., Yenchey, 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-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		85

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., Kundyz, 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.

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-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		89

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-123.24.0040.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		90

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-123.24.0040.00.00.ТЗ			
Вим.	Арк.	№ документа	Підпис	Дата				
Розробив	Середа О.О.				Дослідження та програмна реалізація системи автоматизації будинку на основі технології ВІоТ	Літ.	Аркуш	Аркушів
Перевірів	Дресв О.М.					М	1	6
Н. Контр.	Коваленко А.С.				ЦНТУ КІ-23М			
Затв.	Смірнов О.А.							

1 Найменування та область застосування

Це технічне завдання розповсюджується на дослідження та програмну реалізацію системи автоматизації будинку на основі технології ВІоТ.

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. № 19-13 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 автоматизації будинку на основі технології ВІоТ.

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-123.24.0040.00.00.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 автоматизації будинку на основі технології ВІоТ;
- 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-123.24.0040.00.00.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-123.24.0040.00.00.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 повинна бути розглянутий аналіз умов праці на робочому місці ІТ-фахівця.

					ВКРМ-123.24.0040.00.00.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 – 91 аркуш.

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 16.12.2024 р.

					ВКРМ-123.24.0040.00.00.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
_____ Дреєв О.М.

*Дослідження та програмна реалізація
системи автоматизації будинку на основі технології ВІоТ*

Лістинг програми

Код документу 12

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

Загальна кількість аркушів: 19

Літера: РП

Кропивницький – 2024 року

Основна програма

```
# Імпорт бібліотек для роботи з датчиками, пристроями та мережею
import time
import random
import requests

# Клас сенсора для отримання показників з різних пристроїв
class Sensor:
    def __init__(self, name, unit, min_value=0, max_value=100):
        self.name = name
        self.unit = unit
        self.min_value = min_value
        self.max_value = max_value
        self.value = 0

# Метод для симуляції зчитування значень з сенсора
def read(self):
    self.value = random.uniform(self.min_value, self.max_value)
    return self.value

# Клас сенсора руху з власною логікою
class MotionSensor(Sensor):
    def __init__(self, name):
        super().__init__(name, "Detected/Not Detected")
        self.motion_detected = False

# Метод для визначення руху
def read(self):
    self.motion_detected = random.choice([True, False])
    return "Рух виявлено" if self.motion_detected else "Рух не виявлено"

# Клас пристрою для керування різними приладами
class Device:
    def __init__(self, name, power_usage=1.0):
        self.name = name
        self.status = False
        self.power_usage = power_usage

# Метод для увімкнення пристрою
def turn_on(self):
    self.status = True
    print(f"{self.name} увімкнено")

# Метод для вимкнення пристрою
def turn_off(self):
    self.status = False
    print(f"{self.name} вимкнено")

# Метод для отримання енерговитрат
def get_energy_usage(self, hours):
```

```

        return self.power_usage * hours

# Клас освітлювального приладу з власною логікою
class Light(Device):
    def __init__(self, name, brightness=100):
        super().__init__(name, 0.2)
        self.brightness = brightness

# Метод для регулювання яскравості
    def set_brightness(self, level):
        self.brightness = level
        print(f"Яскравість {self.name} встановлена на рівень {self.brightness}")

# Клас контролера для керування сенсорами та пристроями
class Controller:
    def __init__(self):
        self.sensors = []
        self.devices = []
        self.energy_log = []

# Метод для додавання сенсора
    def add_sensor(self, sensor):
        self.sensors.append(sensor)

# Метод для додавання пристрою
    def add_device(self, device):
        self.devices.append(device)

# Метод для моніторингу та прийняття рішень
    def monitor_and_control(self):
        for sensor in self.sensors:
            value = sensor.read()
            print(f"{sensor.name}: {value} {sensor.unit}")
            # Логіка керування кондиціонером
            if sensor.name == "Температура" and value > 24:
                for device in self.devices:
                    if device.name == "Кондиціонер":
                        device.turn_on()
            # Логіка керування світлом на основі сенсора руху
            if isinstance(sensor, MotionSensor):
                if sensor.motion_detected:
                    for device in self.devices:
                        if isinstance(device, Light):
                            device.turn_on()

# Метод для вимірювання енергоспоживання
    def log_energy_usage(self):
        total_energy = 0
        for device in self.devices:
            if device.status:
                energy_used = device.get_energy_usage(5) # умовно працює 5
                self.energy_log.append((device.name, energy_used))

```

годин

```

        total_energy += energy_used
    print(f"Загальна енерговитрата: {total_energy} кВт/год")

# Метод для отримання даних від зовнішнього API (наприклад, погодного)
def get_external_weather_data(self):
    try:
        response =
requests.get("https://api.openweathermap.org/data/2.5/weather?q=Kyiv&appid=your
api_key")

        data = response.json()
        temperature = data['main']['temp']
        print(f"Зовнішня температура: {temperature} °C")
        return temperature
    except Exception as e:
        print(f"Помилка отримання даних: {str(e)}")
        return None

# Клас для розрахунку мережевого навантаження
class NetworkCalculator:
    def __init__(self):
        self.data_usage = 0

# Метод для розрахунку трафіку на основі кількості сенсорів та пристроїв
def calculate_data_usage(self, sensors, devices):
    sensor_data_size = len(sensors) * 0.5
    device_data_size = len(devices) * 0.2
    self.data_usage = sensor_data_size + device_data_size
    print(f"Загальний трафік: {self.data_usage} КБ")

# Додатковий клас для інтеграції з іншими пристроями через протокол MQTT
class MqttClient:
    def __init__(self, broker_url):
        self.broker_url = broker_url

# Метод для підключення до брокера
def connect(self):
    print(f"Підключення до брокера MQTT за адресою {self.broker_url}")

# Метод для відправки даних
def publish(self, topic, message):
    print(f"Відправка повідомлення '{message}' на тему '{topic}'")

# Основна логіка системи

# Створюємо об'єкти сенсорів
temperature_sensor = Sensor("Температура", "°C", 15, 30)
humidity_sensor = Sensor("Вологість", "%", 30, 70)
motion_sensor = MotionSensor("Сенсор руху")

# Створюємо об'єкти пристроїв
ac_unit = Device("Кондиціонер", 1.5)
light = Light("Освітлення", 80)

```

```
# Створюємо контролер
home_controller = Controller()

# Додаємо сенсори та пристрої до контролера
home_controller.add_sensor(temperature_sensor)
home_controller.add_sensor(humidity_sensor)
home_controller.add_sensor(motion_sensor)
home_controller.add_device(ac_unit)
home_controller.add_device(light)

# Створюємо об'єкт для розрахунку мережевого трафіку
network_calculator = NetworkCalculator()

# Викликаємо методи контролера для моніторингу та управління
home_controller.monitor_and_control()

# Розраховуємо навантаження на мережу
network_calculator.calculate_data_usage(home_controller.sensors,
home_controller.devices)

# Логуювання енергоспоживання
home_controller.log_energy_usage()

# Отримання зовнішніх даних
home_controller.get_external_weather_data()

# Створюємо MQTT клієнт та підключаємось до брокера
mqtt_client = MqttClient("mqtt://broker.hivemq.com")
mqtt_client.connect()

# Відправляємо повідомлення через MQTT
mqtt_client.publish("home/automation", "Система активна")
```

Файл sensors.py

```
# Файл: sensors.py

import random

# Клас сенсора для базових параметрів, таких як температура, вологість і т.д.
class Sensor:
    def __init__(self, name, unit, min_value=0, max_value=100):
        self.name = name
        self.unit = unit
        self.min_value = min_value
        self.max_value = max_value
        self.value = 0

# Метод для отримання даних з сенсора
    def read(self):
        self.value = random.uniform(self.min_value, self.max_value)
        return self.value

# Клас сенсора руху для виявлення активності
class MotionSensor(Sensor):
    def __init__(self, name):
        super().__init__(name, "Detected/Not Detected")
        self.motion_detected = False

# Метод для визначення руху
    def read(self):
        self.motion_detected = random.choice([True, False])
        return "Рух виявлено" if self.motion_detected else "Рух не виявлено"
```

```
# Файл: devices.py

# Клас пристрою, який можна контролювати (увімкнути, вимкнути, тощо)
class Device:
    def __init__(self, name, power_usage=1.0):
        self.name = name
        self.status = False
        self.power_usage = power_usage

# Метод для увімкнення пристрою
    def turn_on(self):
        self.status = True
        print(f"{self.name} увімкнено")

# Метод для вимкнення пристрою
    def turn_off(self):
        self.status = False
        print(f"{self.name} вимкнено")

# Метод для отримання енерговитрат
    def get_energy_usage(self, hours):
        return self.power_usage * hours

# Клас освітлення для керування яскравістю світла
class Light(Device):
    def __init__(self, name, brightness=100):
        super().__init__(name, 0.2)
        self.brightness = brightness

# Метод для регулювання яскравості
    def set_brightness(self, level):
        self.brightness = level
        print(f"Яскравість {self.name} встановлена на рівень {self.brightness}")
```

Файл controller.py

```
# Файл: controller.py

from sensors import Sensor, MotionSensor
from devices import Device, Light

# Клас контролера для моніторингу сенсорів та керування пристроями
class Controller:
    def __init__(self):
        self.sensors = []
        self.devices = []
        self.energy_log = []

# Метод для додавання сенсора
    def add_sensor(self, sensor):
        self.sensors.append(sensor)

# Метод для додавання пристрою
    def add_device(self, device):
        self.devices.append(device)

# Метод для моніторингу та прийняття рішень
    def monitor_and_control(self):
        for sensor in self.sensors:
            value = sensor.read()
            print(f"{sensor.name}: {value} {sensor.unit}")
            # Логіка керування кондиціонером на основі температури
            if sensor.name == "Температура" and value > 24:
                for device in self.devices:
                    if device.name == "Кондиціонер":
                        device.turn_on()
            # Логіка керування освітленням на основі сенсора руху
            if isinstance(sensor, MotionSensor):
                if sensor.motion_detected:
                    for device in self.devices:
                        if isinstance(device, Light):
                            device.turn_on()

# Метод для запису енергоспоживання
    def log_energy_usage(self):
        total_energy = 0
        for device in self.devices:
            if device.status:
                energy_used = device.get_energy_usage(5)
                self.energy_log.append((device.name, energy_used))
                total_energy += energy_used
        print(f"Загальна енерговитрата: {total_energy} кВт/год")
```

Файл network.py

```
# Файл: network.py

# Клас для розрахунку мережевого навантаження
class NetworkCalculator:
    def __init__(self):
        self.data_usage = 0

# Метод для розрахунку трафіку на основі кількості сенсорів та пристроїв
def calculate_data_usage(self, sensors, devices):
    sensor_data_size = len(sensors) * 0.5
    device_data_size = len(devices) * 0.2
    self.data_usage = sensor_data_size + device_data_size
    print(f"Загальний трафік: {self.data_usage} КБ")
```

КБПЗ_2024

Файл mqtt_client.py

```
# Файл: mqtt_client.py

# Клас для підключення та керування протоколом MQTT
class MqttClient:
    def __init__(self, broker_url):
        self.broker_url = broker_url

# Метод для підключення до брокера
    def connect(self):
        print(f"Підключення до брокера MQTT за адресою {self.broker_url}")

# Метод для відправки повідомлень
    def publish(self, topic, message):
        print(f"Відправка повідомлення '{message}' на тему '{topic}'")
```

КБПЗ_2024

Файл notification.py

```
# Файл: notification.py

import smtplib
from email.mime.text import MIMEText

# Клас для відправки сповіщень
class NotificationSystem:
    def __init__(self, email_host, email_port, email_user, email_password):
        self.email_host = email_host
        self.email_port = email_port
        self.email_user = email_user
        self.email_password = email_password

# Метод для відправки email
    def send_email(self, to_email, subject, message):
        try:
            msg = MIMEText(message)
            msg['Subject'] = subject
            msg['From'] = self.email_user
            msg['To'] = to_email

            with smtplib.SMTP(self.email_host, self.email_port) as server:
                server.starttls()
                server.login(self.email_user, self.email_password)
                server.sendmail(self.email_user, to_email, msg.as_string())
            print(f"Сповіщення надіслано на {to_email}")
        except Exception as e:
            print(f"Не вдалося надіслати сповіщення: {str(e)}")
```

Файл voice_assistant.py

```
# Файл: voice_assistant.py

from gtts import gTTS
import os

# Клас для голосового відгуку через Google Assistant API
class VoiceAssistant:
    def __init__(self, language='uk'):
        self.language = language

# Метод для створення голосового повідомлення
    def speak(self, text):
        tts = gTTS(text=text, lang=self.language)
        tts.save("response.mp3")
        os.system("mpg321 response.mp3")
```

КБПЗ_2024

Файл dashboard.py

```
# Файл: dashboard.py

from flask import Flask, render_template, jsonify
from sensors import Sensor
from devices import Device

app = Flask(__name__)

# Створюємо об'єкти сенсорів та пристроїв
temperature_sensor = Sensor("Температура", "°C", 15, 30)
ac_unit = Device("Кондиціонер", 1.5)

# Головна сторінка панелі керування
@app.route('/')
def home():
    return render_template('index.html')

# API для отримання даних сенсорів
@app.route('/api/sensors')
def get_sensors():
    temp_value = temperature_sensor.read()
    return jsonify({'temperature': temp_value})

# API для керування пристроями
@app.route('/api/devices/ac/on')
def turn_ac_on():
    ac_unit.turn_on()
    return jsonify({'status': 'Кондиціонер увімкнено'})

@app.route('/api/devices/ac/off')
def turn_ac_off():
    ac_unit.turn_off()
    return jsonify({'status': 'Кондиціонер вимкнено'})

if __name__ == '__main__':
    app.run(debug=True)
```

Файл automation_rules.py

```
# Файл: automation_rules.py

class AutomationRules:
    def __init__(self, controller):
        self.controller = controller

# Метод для перевірки правил автоматизації
def check_rules(self):
    for sensor in self.controller.sensors:
        value = sensor.read()
        if sensor.name == "Температура" and value > 30:
            self.controller.get_device_by_name("Кондиціонер").turn_on()
        if sensor.name == "Температура" and value < 20:
            self.controller.get_device_by_name("Кондиціонер").turn_off()
```

КБПЗ_2024

Файл auth.py

```
# Файл: auth.py

from flask import Flask, request, jsonify, session
from werkzeug.security import generate_password_hash, check_password_hash

app = Flask(__name__)
app.secret_key = 'your_secret_key'

# Простий словник для зберігання користувачів
users = {}

# Реєстрація користувача
@app.route('/register', methods=['POST'])
def register():
    username = request.form['username']
    password = generate_password_hash(request.form['password'], method='sha256')
    users[username] = password
    return jsonify({'message': 'Реєстрація успішна'})

# Логін користувача
@app.route('/login', methods=['POST'])
def login():
    username = request.form['username']
    password = request.form['password']
    if username in users and check_password_hash(users[username], password):
        session['username'] = username
        return jsonify({'message': 'Логін успішний'})
    return jsonify({'message': 'Невірні дані'})

# Логат користувача
@app.route('/logout')
def logout():
    session.pop('username', None)
    return jsonify({'message': 'Вихід успішний'})
```

```
# Файл: camera.py

import cv2

# Клас для роботи з IoT камерами
class Camera:
    def __init__(self, camera_id=0):
        self.camera_id = camera_id
        self.cap = cv2.VideoCapture(camera_id)

# Метод для отримання потоку відео
def start_stream(self):
    while True:
        ret, frame = self.cap.read()
        if not ret:
            print("Помилка отримання відео")
            break
        cv2.imshow('Live Stream', frame)
        if cv2.waitKey(1) & 0xFF == ord('q'):
            break
    self.cap.release()
    cv2.destroyAllWindows()
```

Файл remote_access.py

```
# Файл: remote_access.py

from flask import Flask, jsonify
from devices import Device

app = Flask(__name__)

# Створюємо об'єкт пристрою
light = Device("Освітлення", 0.5)

# API для віддаленого керування пристроєм
@app.route('/remote/light/on')
def turn_light_on():
    light.turn_on()
    return jsonify({'status': 'Освітлення увімкнено'})

@app.route('/remote/light/off')
def turn_light_off():
    light.turn_off()
    return jsonify({'status': 'Освітлення вимкнено'})

if __name__ == '__main__':
    app.run(host='0.0.0.0', port=8080, debug=True)
```

Файл air_quality.py

```
# Файл: air_quality.py

import random

# Клас для сенсора якості повітря
class AirQualitySensor:
    def __init__(self, name="CO2", unit="ppm"):
        self.name = name
        self.unit = unit
        self.value = 0

# Метод для отримання даних
    def read(self):
        self.value = random.uniform(300, 800) # Симуляція даних
        return self.value

# Клас для вентиляції
class Ventilation(Device):
    def __init__(self, name="Вентиляція", power_usage=1.0):
        super().__init__(name, power_usage)
```

Файл scheduler.py

```
# Файл: scheduler.py

import schedule
import time
from devices import Device

# Клас для розкладу
class Scheduler:
    def __init__(self):
        self.jobs = []

# Метод для додавання задачі
    def add_job(self, job, time):
        self.jobs.append(schedule.every().day.at(time).do(job))

# Метод для запуску розкладу
    def run(self):
        while True:
            schedule.run_pending()
            time.sleep(1)

# Створюємо пристрій та планувальник
light = Device("Освітлення")
scheduler = Scheduler()

# Додаємо задачу
scheduler.add_job(light.turn_off, "22:00")
scheduler.run()
```