КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
БУДІВНИЦТВА І АРХІТЕКТУРИ

автоматизації і інформаційних технологій

(факультет)

інформаційних технологій

(кафедра)

ПОЯСНЮВАЛЬНА ЗАПИСКА
ДО АТЕСТАЦІЙНОЇ ВИПУСКНОЇ РОБОТИ
НА ЗДОБУТТЯ ОСВІТНЬОГО РІВНЯ «БАКАЛАВР»

на тему: «Розробка підсистеми управління "розумним" домом»

ФІСЬКОВ ІЛЛЯ СЕРГІЙОВИЧ

(прізвище, ім’я та по батькові студента повністю)

Київ 2023 р.

КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
БУДІВНИЦТВА І АРХІТЕКТУРИ

автоматизації і інформаційних технологій

(факультет)

інформаційних технологій

(кафедра)

ЗАТВЕРДЖУЮ
Завідувач кафедри ІТ
д.т.н., професор Цюцюра С.В.

«___» _____20___ року

ПОЯСНЮВАЛЬНА ЗАПИСКА
ДО АТЕСТАЦІЙНОЇ ВИПУСКНОЇ РОБОТИ
НА ЗДОБУТТЯ ОСВІТНЬОГО РІВНЯ «БАКАЛАВР»

на тему: «Розробка підсистеми управління "розумним" домом»

Виконав:студент 4-го курсу, групи КН-41

Спеціальності: 122 «Комп`ютерні науки»

Спеціалізація: _ «Інформаційні управляючі системи та технології»
(шифр і назва напряму підготовки, спеціальності)

Фіськов І.С. .
(прізвище та ініціали)

Керівник к.т.н., доц. Горда О.В.
(прізвище та ініціали)

Рецензент к.т.н., доц. Шутовський О.М.
(прізвище та ініціали)

Київ, 2023 р.

КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
БУДІВНИЦТВА І АРХІТЕКТУРИ

Факультет: __автоматизації і інформаційних технологій
Кафедра:___інформаційних технологій
Освітній рівень: «бакалавр» за ОПП
Спеціальність:_122 «Комп`ютерні науки»
Спеціалізація:__Інформаційні управляючі системи та технології

ЗАТВЕРДЖУЮ
Завідувач кафедри ІТ
д.т.н., професор Цюцюра С.В.
_____
„___” _____2023 року

З А В Д А Н Н Я
ДО ВИКОНАННЯ АТЕСТАЦІЙНОЇ ВИПУСКНОЇ РОБОТИ
НА ЗДОБУТТЯ ОСВІТНЬОГО РІВНЯ «БАКАЛАВР»

Фіськов Ілля Сергійович
(прізвище, ім’я, по батькові)

1. Тема роботи: Розробка підсистеми управління "розумним" домом
керівник роботи: Горда Олена Володимирівна, д.т.н,
затверджені наказом ректора КНУБА № 1811/2 від « 17» листопада 2022 р.

2. Термін подачі студентом роботи до захисту: 01 червня 2023 .
3. Вихідні дані до роботи _____
4. Зміст пояснювальної записки: Вступ 1. Аналіз проблеми. 2. Проєктування програмного забезпечення. 3. Розробка програмного забезпечення. 4. Техніко-економічне обґрунтування розробки підсистеми (Бізнес-план)
5. Перелік презентаційно-інформаційних слайдів: 1. Розробка підсистеми управління «розумним» домом. 2. Мета, предмет, об’єкт та актуальність . 3. Дерево цілей . 4. Аналіз типів підсистем управління. 5. Постановка задачі. 6. Вибір архітектури систем IoT. 7. Макет інтерфейсу керування. 8. Функціональні блоки інтерфейсу та use case діаграма. 9. Структурна схема функцій. 10. Вибір програмного інструментарію. 11. Приклад інтерфейсу програми на ПК. 12. Структурна схема управління. 13. Бізнес план. 14. Висновок

6. Консультанти розділів атестаційної випускної роботи

Розділ	Прізвище, ініціали та посада консультанта, представника комісії	дата	підпис
Техніко-економічне обґрунтування розробки підсистеми (Бізнес-план)	д.т.н. проф. Цюцюра С.В.		
Прийом програмного продукту	к.т.н., доц. Єрукаєв А.В.		

7. Дата видачі завдання: 15 лютого 2023 р.

КАЛЕНДАРНИЙ ПЛАН

Види робіт та їх зміст	Дата виконання
Р. 1. Аналіз та дослідження проблеми	17.04.2023-22.04.2023
Р. 2. Проєктування інформаційного забезпечення	22.04.2023-08.05.2023
Р. 3. Практична реалізація	08.05.2023-12.05.2023
Р. 4. Бізнес план	09.04.2023-12.04.2023
Остаточне оформлення роботи	21.05.2023–22.05.2023
Направлення роботи на рецензування	24.05.2023-28.05.2023
Попередній захист роботи на кафедрі	13.06.2023–14.06.2023

Бакалавр _____ Фіськов І.С.
(підпис) (прізвище та ініціали)

Керівник _____ Горда О.В.
(підпис) (прізвище та ініціали)

АНОТАЦІЯ

Фіськов І.С. Розробка підсистем управління «розумного» дому

Дипломна бакалаврська робота за спеціальністю – «Комп’ютерні науки» - Київський національний університет будівництва та архітектури, Київ, 2023 рік.

Бакалаврську роботу присвячено дослідженню основних тенденцій, принципів та реалізацій підсистем управління «розумних» будинків.

У роботі досліджено основні принципи та технології розробки систем управління "розумного" будинку, створені та проаналізовані, основні методи забезпечення теоретичних засобів управління

Ключовими словами є: "Управління", "розумний будинок", "системи управління", "підсистема", "функціональні можливості".

ABSTRACT

Fiskov I.S. Development of "smart" home control subsystems

Bachelor's diploma work in the specialty - "Computer Science" - Kyiv National University of Construction and Architecture, Kyiv, 2023.

The bachelor's thesis is devoted to the study of the main trends, principles and implementations of management subsystems of "smart" buildings.

The work examines the main principles and technologies of developing control systems of a "smart" house, created and analyzed the main methods of providing theoretical control tools

Key words are: "Management", "smart house", "control systems", "subsystem", "functionality".

ЗМІСТ

ВСТУП 9

1 АНАЛІЗ ПРОБЛЕМИ 12

1.1. Загальні відомості 12

1.2. Проблематика розумних підсистем 13

1.3. Дерево цілей 15

1.4. Аналіз типів підсистем управління 19

1.5. Аналіз виробників систем розумних речей 23

1.6. Використання штучного інтелекту в підсистемах розумного дому 25

1.7. Постановка задачі 28

2. ПРОЕКТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ 30

2.1. Вимоги програмного забезпечення 30

2.2. Архітектура програмного забезпечення розумного дому 31

2.2.1. Вибір архітектурної основи 31

2.2.2. Вибір системи управління базою даних 33

2.2.3. Включення інших архітектурних моментів 34

2.3. Розробка інтерфейсу користувача 35

2.3.1. Розробка інтерфейсу користувача 35

2.3.2. Вибір технологій та інструментів для розробки інтерфейсу 38

2.4. Проектування візуальної частини інтерфейсу з урахуванням зручності та ефективності використання 42

2.5. Розробка функціональності інтерфейсу 43

2.6. Use case діаграма інтерфейсу 45

2.7. Представлення структурної схеми функцій 46
2.8. Постановка питань розробки системи управління 48
2.9. Тестування та валідація програмного забезпечення 50
2.10. Управляючі функції та їх структура 51
2.10.1. Структура управляючої підсистеми 49
2.10.2. Управляючі функції 53
2.10.3. Взаємодія з користувачем 54
3. РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ 56
3.1. Вибір програмного інструментарію 56
3.1.1. Вибір мови програмування 56
3.1.2. Вибір середовища розробки: 59
3.2. Розробка інтерфейсної частини підсистеми управління ррозумним домом 63
3.3. Розробка програмної частини розробки підсистем управління розумного будинку 67
3.4. Реалізація управляючих функцій 69
3.4.1. Реалізація комунікаційного протоколу 69
3.4.2. Реалізація підключення до бази даних 68
3.4.3. Реалізація підключення датчиків та акураторів 74
4. БІЗНЕС ПЛАН 76
4.1. Резюме 76
4.1.1. Впровадження продукту 76
4.1.2. Визначення визначних сторін продукту 77
4.1.3. Очікувані фінансові результати 79
4.2. Проектований продукт 80

4.2.1. Опис продукту проєкту 80
4.3. Оцінка ринку збуту 79
4.3.1. Умови постачання та понеціал продукту 79
4.3.2. Аналіз даних 83
4.4. Конкуренція 86
4.4.1. Найбільші виробники аналогічного продукту 86
4.4.2. Сутність продукту конкуренту 87
4.5. Стратегія маркетингу 88
4.5.1. Схема розповсюдження товарів 88
4.5.2. Ціноутворення та реклама 89
4.6. Організаційний план 87
4.6.1. Кваліфікаційні вимоги 87
4.7. Юридичний план 89
4.8. Оцінка ризиків і страхування 94
4.8.1. Типи ризиків 94
4.9. Стратегія фінансування 96
4.9.1. План отримання засобів для створення або розширення підриємства 96
ВИСНОВОК 99
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ 101
Додаток А 101

ВСТУП

Актуальність дослідження. В сучасному світі все більшу популярність набуває концепція "розумного" дому, що передбачає використання різноманітних технологій та пристроїв для автоматизації різних процесів у житловому приміщенні. Розробка підсистеми управління "розумним" домом є актуальною темою дослідження, оскільки це дозволяє забезпечити зручне та ефективне управління всіма системами та пристроями в домі, такими як опалення, кондиціонування повітря, освітлення, вентиляція, безпека та інші.

Така підсистема управління може значно зменшити споживання електроенергії, підвищити комфорт та безпеку проживання, а також забезпечити зниження витрат на ремонт та обслуговування пристроїв у домі. Крім того, розробка "розумного" дому може бути корисною для людей з обмеженими можливостями, оскільки така система може забезпечити додаткові можливості контролю та управління різними системами у житловому приміщенні.

У світі вже існує значна кількість побудованих "розумних" будинків, але їхня кількість постійно збільшується, тому розробка нових технологій та підсистем управління є дуже важливим напрямом досліджень у галузі "розумного" житла.

Мета дослідження. Покращення комфорту, безпеки та енергоефективності "розумного" дому, а також зменшення витрат на його обслуговування та ремонт, можливе створення універсальності в підсистемі.

Об’єкт дослідження. Підсистеми та пристрої автоматизації у житловому приміщенні.

Предмет дослідження. Розробка ефективної та зручної підсистеми управління різними системами та пристроями в житловому приміщенні з використанням сучасних технологій та методів.

Методи дослідження.

- Аналіз технічних можливостей та особливостей різних систем автоматизації.
- Вивчення популярних технологій та стандартів управління "розумними" домами.
- Визначення потреб та вимог користувачів щодо функціональності та зручності системи управління.
- Розробка та тестування програмного забезпечення для управління підсистемами автоматизації.
- Встановлення та налаштування обладнання для роботи підсистеми управління.
- Експериментально-теоретичне тестування та оцінка ефективності підсистеми.

Завдання дослідження.

- Аналіз ринку технологій автоматизації житлових приміщень та визначення найбільш популярних та ефективних рішень.
- Розробка програмного забезпечення для управління системами автоматизації та інтеграція різних систем управління в єдину платформу.
- Вивчення потреб користувачів та визначення функціональних вимог до підсистеми управління.
- Встановлення та налаштування обладнання для роботи підсистеми управління.
- Тестування та оцінка ефективності.
- Розробка методів зменшення витрат на енергопостачання та оптимізація енергоспоживання в житловому приміщенні.
- Забезпечення безпеки користування системою управління та захисту від несанкціонованого доступу.

Наукова новизна. Розробка цілісної підсистеми управління різними функціями житла на основі інтернет-платформ та додатків забезпечує зручний та ефективний контроль за станом будинку і становить наукову новизну у цій галузі. Крім того, дослідження можуть включати нові підходи до оптимізації витрат енергії та забезпечення безпеки користування системою управління, що також представляють наукову новизну.

Практична значимість. Проаналізована підсистема може допомогти значно зменшити витрати на енергопостачання та підвищити комфортність проживання в житловому приміщенні. Розумна підсистема управління може автоматично регулювати температуру, вологість, освітлення та інші параметри відповідно до потреб користувача. Крім того, така підсистема може забезпечувати безпеку будинку, контролювати рівень води та газу, відслідковувати стан електричних приладів та багато іншого.

Що важливо, відкриває можливості в будівництві нового житла та модернізації існуючого. Розумний будинок може бути привабливим для потенційних покупців житла, що збільшує його комерційну цінність. Крім того, впровадження такої підсистеми може збільшити енергоефективність будинку та зменшити його вплив на довкілля.

Отже, практична значимість даної теми полягає в покращенні комфорту проживання та зменшенні витрат на енергопостачання в житлових приміщеннях для людей. І в теоретичному створені стандартизованості та універсальності в системі для розробників.

Результати дослідження. Проаналізована розробка прототипу системи управління, що дозволяє контролювати освітлення, опалення, вентиляцію, системи безпеки та інші параметри житлового приміщення. Крім того, дослідження включають вивчення технологічних рішень для підвищення енергоефективності будинків, зменшення витрат на енергопостачання та покращення екологічної стійкості будівель.

1 АНАЛІЗ ПРОБЛЕМИ

1.1.Загальні відомості

За останні кілька років зростає попит на технології "розумних" речей, що створює безліч можливостей для розширення функціоналу та зручності в повсякденному житті людей. Такі технології називаються Internet of Things (IoT) – це мережа фізичних пристроїв, які мають можливість збирати та обмінюватися даними між собою без участі людини. Ці пристрої можуть виконувати різноманітні функції, такі як контроль за станом пристроїв, збір інформації про оточуюче середовище, аналіз отриманих даних та прийняття відповідних рішень[1].

Одним з найпопулярніших застосувань IoT є "розумний" будинок, який забезпечує автоматизацію та контроль за електроприладами, освітленням, системами безпеки та іншими пристроями в будинку. Ці системи дозволяють людям економити час та гроші, знижуючи витрати на енергію та інші ресурси.

В цьому дипломному проекті досліджені та розглянуті загальні та теоретичні аспекти систем розумних речей та IoT, з фокусом на розробці підсистеми управління "розумним" будинком. Дослідження були проведені з використанням актуальних даних та відповідної літератури в цій галузі. Серед проаналізованої інформації були, до прикладу, дані області:

- Архітектура систем розумних речей та IoT
- Технології та пристрої, які використовуються в системах та підсистемах IoT
- Застосування систем IoT в "розумних" будинках
- Проектування та розробка підсистем IoT для "розумних" будинків

- Проблеми та виклики, пов'язані з використанням підсистем розумних речей та IoT в "розумних" будинках, такі як проблеми приватності та безпеки.

Метою дипломного проекту є дослідження та аналіз систем розумних речей та IoT, зосередження на їх застосуванні в "розумних" будинках та розробка прототипу системи IoT для будинку з використанням датчиків, мікроконтролерів та хмарних сервісів. В результаті проекту очікується отримати відповідні знання та навички з проектування та розробки систем IoT, а також здатність застосовувати ці знання на практиці.

У процесі виконання дипломного проекту були використані методи дослідження та аналізу, а також методи проектування та розробки програмного забезпечення. Також були використані відповідні інструменти та технології для розробки прототипу системи IoT для "розумного" будинку.

Загальні та теоретичні знання про системи розумних речей та IoT, їх застосування в "розумних" будинках та проблеми, пов'язані з їх використанням, є дуже важливими для розвитку цієї галузі. Дані технології можуть значно полегшити та зручніше зробити повсякденне життя людей, але вони також мають певні виклики та ризики, які необхідно враховувати та усувати. Виконання цього дипломного проекту сприятиме поглибленню знань в цій галузі та розумінню її потенціалу та обмежень.

1.2. Проблематика розумних підсистем

Розробка підсистеми управління "розумним" домом – це процес створення комплексної системи, яка забезпечує автоматизацію та контроль різних функцій в будинку, таких як освітлення, опалення, охорона, звукова та візуальна інформація, а також багато інших.

Одна з основних проблем, яка виникає при розробці такої підсистеми – це технічна складність. Управління "розумним" домом вимагає інтеграції різних пристроїв, сенсорів, платформ та протоколів в єдину систему, що може бути складно і дорого. Крім того, це вимагає досконалого розуміння технічної структури інтерфейсів між різними підсистемами та узгодження різних форматів даних.

Ще одна проблема, яка пов'язана з управлінням "розумним" домом – це забезпечення безпеки. Оскільки дані, які обробляються у системі, є конфіденційними, важливо забезпечити їх захист від несанкціонованого доступу та зламу системи. Це може бути досягнуто за допомогою шифрування даних, використання системи автентифікації та інших методів.

Ще одна проблема, пов'язана з управлінням "розумним" домом, – це визначення потреб та очікувань користувачів. Якщо система не відповідає потребам користувачів, то вона може бути неприйнятною для використання або навіть небезпечною. Тому важливо розробляти систему з урахуванням потреб та очікувань користувачів.

Крім того, для багатьох людей ця технологія може бути неприйнятною через недостатню освіту або необізнаність у технологічних рішеннях. Тому важливо проводити навчання та надавати користувачам підтримку після впровадження системи.

Отже, розробка підсистеми управління "розумним" домом потребує рішення технічних, безпекових та організаційних проблем. Необхідно забезпечити сумісність та інтеграцію різних систем "розумного" дому на різних платформах та протоколах, а також забезпечити надійність та безпеку управління. Для досягнення цих цілей можуть бути використані технології штучного інтелекту, блокчейн, захисту даних та інші технології.

Крім технічних аспектів, розробка підсистеми управління "розумним" домом також повинна враховувати потреби та очікування користувачів. Наприклад, забезпечення простоти та зручності управління

системами "розумного" дому може зробити їх більш доступними та прийнятними для широкого кола користувачів.

Також важливо мати на увазі економічний аспект розробки підсистеми управління "розумним" домом. Інвестування у розробку та впровадження "розумного" будинку може бути високим, тому важливо забезпечити ефективне використання ресурсів та знизити витрати на енергопостачання.

Отже, розробка підсистеми управління "розумним" домом вимагає рішення багатьох складних технічних, безпекових та організаційних проблем, а також врахування потреб та очікувань користувачів і економічного аспекту. Розв'язання цих проблем може допомогти створити більш ефективні та доступні системи "розумного" дому, що поліпшить якість життя та знизять витрати на енергопостачання.

1.3. Дерево цілей

Так як основна мета: розробити підсистему управління "розумним" домом, яка може охоплювати безліч модулів і їх процесів роботи, вважаю доцільним внести більше цілей для їх охоплення. Необхідно зауважити, що робота виконана в науково дослідницьких цілях, тож робота не несе за собою цілі виконання кожного аспекту наведеного в рисунку.

Дерево цілей даного дипломного проєкту наведено на рисунку 1.1.

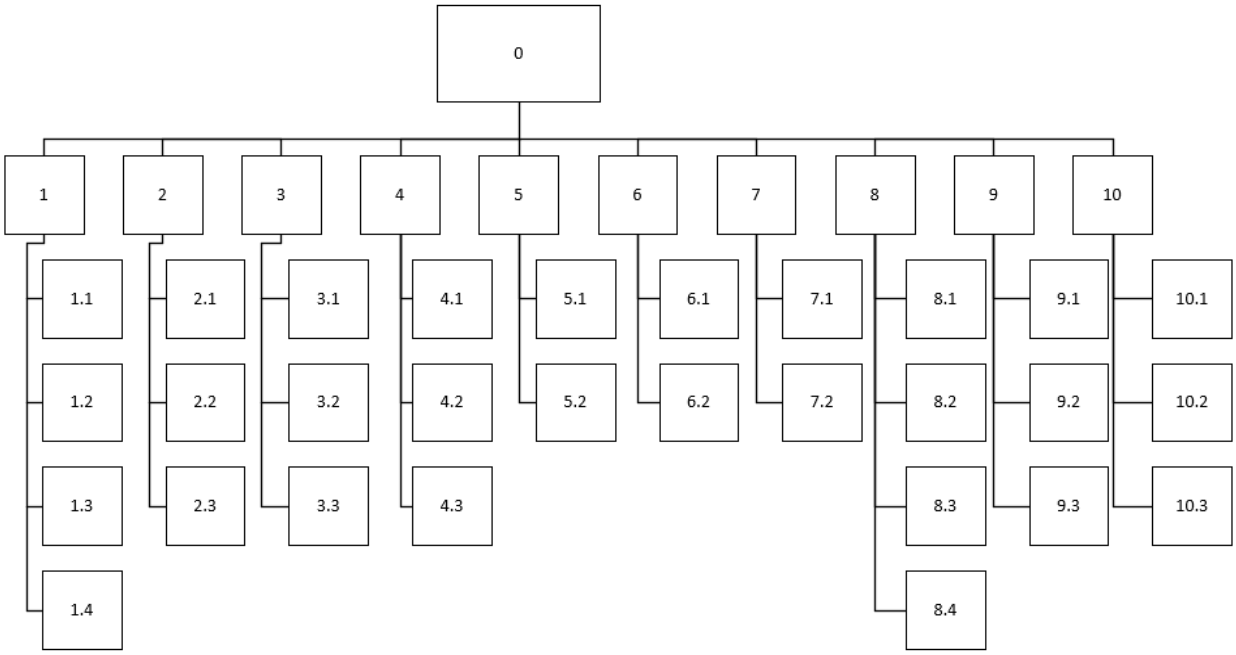


Рисунок 1.1 – Дерево цілей

1. Функціональні вимоги
   1.1. Забезпечити можливість керування приладами в будинку через мобільний додаток.
   1.2. Забезпечити можливість відслідковування стану різних підсистем в будинку, таких як підсистема опалення, підсистема водопостачання тощо.
   1.3. Забезпечити можливість автоматичного управління підсистемою опалення та кондиціонування повітря.
   1.4. Забезпечити можливість налаштування графіка роботи різних пристроїв в будинку, таких як світильники, пристрої для приготування їжі тощо.
2. Надійність та безпека
   2.1. Забезпечити надійне збереження даних про стан підсистем в будинку.
   2.2. Забезпечити захист підсистеми від зламів та злочинних дій.
   2.3. Забезпечити можливість автоматичного вимкнення електроприладів та інших пристроїв в разі виникнення небезпечних ситуацій.

3. Енергоефективність
   3.1. Забезпечити можливість енергозбереження шляхом автоматичного вимикання електроприладів та інших пристроїв в разі їх непотрібності.
   3.2. Забезпечити можливість налаштування графіка роботи підсистеми опалення та кондиціонування повітря з урахуванням енергоефективності.
   3.3. Забезпечити можливість відстеження споживання електроенергії та інших ресурсів в будинку.
4. Інтегрованість
   4.1. Забезпечити можливість інтеграції підсистеми управління "розумним" домом зі створення дерева основних цілей теми "Розробка підсистеми управління "розумним" домом
   4.2. Забезпечити можливість інтеграції з іншими підсистемами в будинку, такими як підсистема безпеки, підсистема відеонагляду, підсистема контролю доступу тощо.
   4.3. Забезпечити можливість інтеграції з хмарними сервісами для забезпечення можливості керування будинком віддалено та забезпечення безпеки даних.
5. Зручність та комфорт
   5.1. Забезпечити можливість налаштування різних сценаріїв для різних ситуацій, наприклад, вечірнє освітлення, автоматичне включення підсистеми опалення перед поверненням додому тощо.
   5.2. Забезпечити можливість автоматичного виконання рутинних завдань, таких як включення світла або забезпечення оптимальної температури в будинку.
6. Масштабованість та розширюваність
   6.1. Забезпечити можливість розширення функціональності підсистеми управління "розумним" домом, наприклад, додавання нових пристроїв або інтеграцію з новими підсистемами.

6.2. Забезпечити можливість розширення площі будинку або додавання нових приміщень, не впливаючи на роботу підсистеми управління "розумним" домом.

7. Вартість та доступність
   7.1. Забезпечити можливість розробки та впровадження підсистеми управління "розумним" домом за доступною ціною для користувачів.
   7.2. Забезпечити можливість використання доступних технологій та рішень для розробки підсистеми управління "розумним" домом.
8. Підтримка та обслуговування
   8.1. Забезпечити можливість підтримки та обслуговування підсистеми управління "розумним" домом протягом всього її життєвого циклу
   8.2. Забезпечити можливість оновлення програмного та апаратного забезпечення підсистеми управління "розумним" домом.
   8.3. Забезпечити можливість ремонту та заміни несправних компонентів підсистеми управління "розумним" домом.
   8.4. Забезпечити користувачів інформацією щодо правильного використання та обслуговування підсистеми управління "розумним" домом.
9. Безпека
   9.1. Забезпечити захист від злому та хакерських атак на підсистему управління "розумним" домом.
   9.2. Забезпечити захист персональних даних користувачів та даних про стан будинку.
   9.3. Забезпечити можливість аварійного вимикання підсистеми управління "розумним" домом у разі виявлення загроз для безпеки користувачів та будинку.
10. Екологічність

10.1. Забезпечити можливість ефективного використання енергії та ресурсів, зокрема шляхом автоматичного вимикання світла та підсистем опалення у неактивних зонах будинку.
10.2. Забезпечити можливість використання екологічно чистих джерел енергії, наприклад, сонячних панелей та вітрогенераторів, для живлення підсистеми управління "розумним" домом.
10.3. Забезпечити можливість моніторингу витрат енергії та ресурсів в будинку, щоб користувачі могли ефективніше використовувати їх.

1.4. Аналіз типів підсистем управління

Однією з ключових складових розумного дому є підсистеми управління, які дозволяють автоматизувати роботу з різними пристроями та системами в будинку. Нижче описує найпопулярніші підсистеми управління розумним домом, їхні можливості та переваги, а також можливість їх використання в комплексі з іншими системами для створення більш ефективної та зручної системи управління:

1. Підсистеми управління освітленням. Ця підсистема дозволяє віддалено керувати освітленням в будинку. Зазвичай, вона працює за допомогою датчиків руху та світла, які автоматично вмикають світильники, коли людина заходить в кімнату, і вимикають, коли він покидає її.
2. Підсистеми управління системами опалення та кондиціювання повітря. Ця підсистема дозволяє регулювати температуру в будинку, використовуючи різні методи, такі як таймери, датчики температури, і дистанційне керування.
3. Підсистеми управління безпекою. Ця підсистема має різні функції, такі як відеоспостереження, системи безпеки входів та виходів, а також системи пожежної безпеки. Вона дозволяє користувачам

забезпечувати безпеку свого дому віддалено, використовуючи різні методи.

4. Підсистеми управління розетками та електроприладами. Ця підсистема дозволяє керувати розетками та електроприладами в будинку, використовуючи дистанційне керування. Користувач може віддалено вимикати та включати прилади, такі як пральні машини, холодильники, телевізори та інші.
5. Підсистеми управління звуком та музикою. Ця підсистема дозволяє керувати звуковою системою в будинку, включаючи звукові плеєри, динаміки та інші. Користувач може віддалено вибирати музику та налаштовувати гучність звуку в різних кімнатах будинку.
6. Підсистеми управління водопостачанням та водовідведенням. Ця підсистема дозволяє віддалено керувати системою водопостачання та водовідведення в будинку, включаючи різні функції, такі як регулювання тиску води, вимикання та включання води в різних кімнатах та інші.
7. Підсистеми управління господарськими потребами. Ця підсистема дозволяє користувачам віддалено керувати господарськими потребами, такими як видалення сміття, прибирання підлоги та інші. Зазвичай ця підсистема працює за допомогою роботів–помічників.
8. Підсистеми управління енергоспоживанням. Ця підсистема дозволяє віддалено керувати споживанням енергії в будинку, включаючи функції енергозбереження, автоматичне вимикання електричних приладів та інші.
9. Підсистеми управління присутністю. Ця підсистема дозволяє відслідковувати присутність людей в будинку та автоматично вмикати та вимикати різні системи в залежності від їх наявності.
10. Підсистеми управління меблями та інтер'єром. Ця підсистема дозволяє користувачам віддалено керувати меблями та інтер'єром

будинку, включаючи різні функції, такі як налаштування положення меблів, контроль за освітленням та інші.

11. Підсистеми управління кліматом. Ця підсистема дозволяє віддалено керувати системою опалення та кондиціонування повітря в будинку. Користувач може налаштовувати температуру в будь–якій кімнаті та включати або вимикати систему з будь–якого місця.
12. Підсистеми безпеки. Ця підсистема дозволяє користувачам віддалено керувати системами безпеки в будинку, такими як системи відеоспостереження, системи сигналізації та інші. Користувач може відстежувати події, що відбуваються в будинку, та отримувати сповіщення в разі потенційних проблем.
13. Підсистеми управління комунікаціями. Ця підсистема дозволяє користувачам віддалено керувати системами комунікацій в будинку, такими як телефон, інтернет, телевізор та інші. Користувач може віддалено включати або вимикати ці системи, налаштовувати їх на різні канали та інші функції.
14. Підсистеми управління розвагами. Ця підсистема дозволяє користувачам віддалено керувати розважальними системами в будинку, такими як кінотеатр, системи гри та інші. Користувач може віддалено вибирати фільми, налаштовувати звук та інші функції.
15. Підсистеми управління автомобілями. Ця підсистема дозволяє користувачам віддалено керувати системами автомобіля в будинку, такими як відкривання та закривання дверей, запуск двигуна та інші функції. Користувач може віддалено контролювати автомобіль, перевіряти стан автомобіля та інші функції.
16. Підсистеми управління енергоспоживанням. Ця підсистема дозволяє користувачам контролювати споживання електроенергії в будинку. Користувач може віддалено включати та вимикати пристрої, налаштовувати графіки використання енергії та інші функції.

17. Підсистеми управління водою. Ця підсистема дозволяє користувачам віддалено керувати системами водопостачання та каналізації в будинку. Користувач може віддалено включати та вимикати пристрої, налаштовувати температуру води та інші функції.
18. Підсистеми управління здоров'ям. Ця підсистема дозволяє користувачам віддалено керувати системами моніторингу здоров'я в будинку. Користувач може віддалено контролювати свій стан здоров'я та отримувати сповіщення в разі потенційних проблем.

Ці підсистеми роблять розумний дім більш комфортним та безпечним для користувачів. Кожна з них має свої переваги та може адаптуватись в залежності від потреб користувача. Важливо зазначити, що не всі підсистеми будуть корисними для кожного користувача, і вибір підсистеми повинен бути зроблений на основі індивідуальних потреб і можливостей.

Крім того, підсистеми управління розумним домом можуть взаємодіяти одна з одною, що дозволяє забезпечувати більш повну і ефективну систему управління. Наприклад, підсистема управління освітленням може автоматично вимикатися, коли користувач підходить до виходу з кімнати, а підсистема управління камерами безпеки може відображати зображення зі спальні на телевізорі у вітальні в разі спрацювання датчика руху.

Загалом, розумний дім – це інноваційний підхід до життя, який дозволяє забезпечити комфорт, ефективність та безпеку користувачів. Із зростанням технологій та розвитком інтернету речей, розумний дім стає все доступнішим та популярнішим серед користувачів.

Рисунок 1.2 – Стандартні групи наборів підсистем

1.5. Аналіз виробників систем розумних речей

Ринок підсистем управління розумним домом розширюється з кожним роком, тому існує велика кількість виробників цих систем. Найпопулярніші з них:

1. Amazon Alexa – це популярний голосовий помічник, який може керувати домашніми пристроями з підтримкою Alexa. Він може здійснювати розмову з користувачем та виконувати задачі, такі як включення телевізора або регулювання температури.
2. Google Home – ця система є конкурентом Amazon Alexa. Вона також працює з голосовим помічником та може керувати різними домашніми пристроями, включаючи освітлення, двері та систему опалення.
3. Apple HomeKit – це інтегрована система керування домашніми пристроями для користувачів Apple. Вона працює з Apple HomePod або iPhone та дозволяє керувати освітленням, термостатом, розетками та іншими пристроями.
4. Samsung SmartThings – ця система керування розумним домом має декілька компонентів, включаючи хаб, сенсори та інші пристрої.

Вона дозволяє керувати різними домашніми пристроями, включаючи камери спостереження, термостати та розумні пристрої.

5. Philips Hue – ця система освітлення є однією з найпопулярніших на ринку. Вона дозволяє керувати освітленням з мобільного пристрою та підключати його до інших систем керування розумним домом.
6. Nest – ця система термостатів дозволяє регулювати температуру в будинку з мобільного пристрою або голосовим помічником. Вона також може вивчати режими життя користувачів і працювати відповідно до їх звичок, що дозволяє економити енергію.
7. Ecobee – ця система термостатів також дозволяє регулювати температуру в будинку з мобільного пристрою. Вона має додаткові функції, такі як датчики руху та розумні камери, що дозволяє відслідковувати присутність в будинку та керувати системою безпеки.
8. Lutron – ця система освітлення та жалюзів дозволяє керувати різними освітленням та жалюзів з мобільного пристрою або голосовим помічником. Вона також має можливості автоматичного керування освітленням та жалюзів на основі зовнішнього освітлення та часу доби.
9. Wink – ця система керування розумним домом підтримує багато пристроїв від різних виробників та дозволяє керувати ними з одного додатку. Вона має розширені можливості настройки, що дозволяє користувачам створювати свої власні сценарії керування.
10. Control4 – ця система керування розумним домом підтримує багато пристроїв від різних виробників та дозволяє керувати ними з одного додатку. Вона має багато розширених можливостей настройки, що дозволяє створювати складні сценарії керування.
11. Savant – ця система керування розумним домом має великий вибір пристроїв та дозволяє керувати ними з одного додатку. Вона має

інтуїтивний інтерфейс та дозволяє створювати персоналізовані сценарії керування.

12. Crestron – ця система керування розумним домомпідтримує багато пристроїв та дозволяє керувати ними з одного додатку. Вона має багато можливостей настройки та інтеграції з іншими системами, що дозволяє створювати складні сценарії керування.

Загалом, розумні системи управління домом є дуже різноманітними, тому при виборі підсистеми необхідно враховувати потреби та особливості свого будинку та родини. Важливо визначити, які функції потрібні, які пристрої використовуються, та які інтеграції з іншими системами можуть бути необхідні. Також варто звернути увагу на інтерфейс та можливості настройки, щоб система була зручною у використанні та можливо налаштувати під свої потреби.

1.6. Використання штучного інтелекту в підсистемах розумного дому

Розробка підсистем управління розумним домом використовує штучний інтелект для автоматизації процесів управління домашньою технікою та забезпечення комфортного життя. Штучний інтелект є однією з найбільш прогресивних технологій в цій галузі, яка дозволяє значно поліпшити функціональність та ефективність систем управління розумним домом.

Проте, існують деякі проблеми, які необхідно вирішити, щоб використання штучного інтелекту в підсистемах управління стало ефективним та безпечним.

Одна з основних проблем пов'язаних з використанням штучного інтелекту в системах управління розумним домом – це проблема захисту персональних даних користувачів. Штучний інтелект вимагає доступу до великої кількості даних для свого навчання та роботи, що може призвести

до порушення приватності користувачів та можливого використання цих даних зі злочинними цілями.

Крім того, іншою проблемою є складність розуміння та взаємодії з системами управління розумним домом для користувачів, які не мають достатнього рівня технічної грамотності. Не всі користувачі можуть зрозуміти, як працюють підсистеми управління розумним домом та як взаємодіяти з ними, що може створити додаткові проблеми.

Також, існують проблеми з впровадженням штучного інтелекту в підсистеми управління розумним домом, пов'язані зі створенням потрібних алгоритмів та програмного забезпечення для роботи з даними та управління системами. Ці проблеми можуть включати складність розробки та інтеграції штучного інтелекту з іншими системами управління розумним домом, які використовують різні стандарти та протоколи.

Незважаючи на ці проблеми, застосування штучного інтелекту в системах управління розумним домом має багато переваг. Однією з основних переваг є покращення енергоефективності, що досягається за рахунок автоматичного вимкнення електроприладів, коли їх не використовують. Це дозволяє зменшити споживання електроенергії та скоротити рахунки за комунальні послуги.

Іншою перевагою є зручність управління домашньою технікою та іншими електронними пристроями, яка забезпечується за рахунок зв'язку між різними системами управління та користувачем. Наприклад, користувач може віддалено включити підігрів підлоги або налаштувати температуру у кімнаті через мобільний додаток.

Крім того, штучний інтелект може допомогти вирішити проблеми з підвищенням рівня безпеки в домі, наприклад, розпізнавати та відслідковувати незвичайну активність в будинку, таку як вторгнення.

Відповідно, можна зробити висновок, що використання штучного інтелекту в системах управління розумним домом має багато переваг, але також потребує вирішення деяких проблем зі захистом персональних

даних та складністю взаємодії з системами для користувачів. Потенційні переваги включають покращення енергоефективності, зручне управління технікою та підвищення рівня безпеки в домі. Тому, розробка підсистем управління розумним домом з використанням штучного інтелекту є актуальним та перспективним напрямом.

Однією з основних проблем, які потрібно вирішити при розробці підсистем управління розумним домом з використанням штучного інтелекту, є забезпечення безпеки даних та зменшення можливості їх зламу. Для цього потрібно розробити відповідні алгоритми шифрування та механізми захисту даних. Також важливо забезпечити правильну настройку системи та регулярно оновлювати її програмне забезпечення для уникнення можливих кібератак.

Ще однією проблемою є інтеграція штучного інтелекту з іншими системами управління розумним домом, які використовують різні стандарти та протоколи. Для розв'язання цієї проблеми необхідно встановити єдиний стандарт та протокол взаємодії між різними системами, або розробити адаптери, які забезпечать сумісність з різними стандартами та протоколами.

Також важливо враховувати етичні питання використання штучного інтелекту в системах управління розумним домом, зокрема, забезпечення приватності користувачів та уникнення дискримінації на основі використання алгоритмів машинного навчання.

Отже, розробка підсистем управління розумним домом з використанням штучного інтелекту має багато переваг, але потребує вирішення деяких технічних та етичних проблем. Розумне використання цієї технології може допомогти покращити якість життя людей, зменшити витрати на енергопостачання та забезпечити більшу безпеку в домі.

Однією з переваг використання штучного інтелекту в підсистемах управління розумним домом є здатність системи адаптуватись до змін у використанні приміщень та потреб користувачів. Наприклад, система може

виявити, що користувачі зазвичай вмикають світло в певних приміщеннях в певний час дня, і автоматично налаштувати світло на цей час. Також, система може на основі аналізу даних про використання приміщень зменшувати витрати на опалення та кондиціонування повітря, наприклад, за рахунок автоматичного вимикання опалення в тих приміщеннях, які не використовуються.

Іншою перевагою використання штучного інтелекту в підсистемах управління розумним домом є можливість дистанційного управління системою з будь–якого місця, де є доступ до Інтернету. Наприклад, користувач може віддалено управляти системою опалення, кондиціонування повітря та освітлення, що забезпечить йому більший комфорт та зменшить витрати на енергопостачання.

Також важливо зазначити, що використання штучного інтелекту може сприяти підвищенню ефективності та економічності підсистем управління розумним домом. Наприклад, система може прогнозувати витрати на енергопостачання та регулювати їх у реальному часі, що дозволить зменшити витрати та підвищити ефективність використання енергії.

Отже, використання штучного інтелекту в підсистемах управління розумним домом має багато переваг, однак є деякі проблеми, які потрібно вирішувати.

1.7.Постановка задачі

Задача полягає у розробці підсистеми управління "розумним" домом з використанням сучасних технологій інтернету речей (IoT) та штучного інтелекту. Метою є створення підсистеми, яка забезпечуватиме зручне, ефективне та енергоефективне управління різними пристроями та

системами в домі, такими як опалення, кондиціонування повітря, освітлення, вентиляція та інші.

Основні завдання, які потрібно вирішити для реалізації даної системи, включають:

1. Аналіз та, при необхідності, розробки, архітектури системи: визначення структури, взаємозв'язків та функцій підсистеми управління "розумним" домом.
2. Аналіз та, при необхідності, розробки додаткових пристроїв: створення датчиків руху, температури, вологості, які будуть відповідати за збір інформації про стан будинку.
3. Аналіз та, при необхідності, розробки програмного забезпечення: розробка програмного забезпечення для збору, аналізу та обробки даних, а також для взаємодії з пристроями в домі.
4. Аналіз та, при необхідності, розробки інтерфейсу користувача: створення зручного інтерфейсу для користувача, який дозволить відстежувати та керувати станом різних систем в домі.
5. Оптимізацію енергоспоживання: розробка алгоритмів, які будуть дозволяти оптимізувати енергоспоживання в домі, зокрема за допомогою планування режимів опалення та кондиціонування повітря.
6. Забезпечення безпеки: розробка системи, яка буде забезпечувати безпеку в домі, наприклад, виявлення пожеж, протікання води та інших аварійних ситуацій

2. ПРОЕКТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

2.1. Вимоги програмного забезпечення

Вимоги до програмного забезпечення для розумного дому включають різні аспекти, такі як функціональність, продуктивність, безпеку, масштабованість та інші. Описані нижче вимоги можуть бути корисними при розробці програмного забезпечення для розумного дому.

1. Функціональність. ПО для розумного дому має забезпечувати ефективне та надійне управління всіма пристроями в домі. Воно повинно бути здатне інтегруватись з усіма пристроями та системами, що використовуються в розумному домі. Також важливо забезпечити можливість налаштування системи, щоб вона відповідала потребам користувача.
2. Продуктивність. ПО для розумного дому повинно бути швидким та ефективним. Це дозволить уникнути затримок при виконанні команд та забезпечити надійне та швидке виконання запитів користувача. Крім того, програмне забезпечення повинно бути здатним працювати зі значною кількістю пристроїв та систем.
3. Безпека. ПО для розумного дому повинно забезпечувати високий рівень безпеки. Воно повинно бути здатним захистити систему від зломів, а також забезпечити захист персональних даних користувачів. Для цього можна використовувати різні методи захисту, такі як шифрування та аутентифікація.
4. Масштабованість. ПО для розумного дому повинно бути здатним масштабуватись під зростаючі потреби користувачів. Воно повинно бути здатним працювати зі значною кількістю пристроїв та систем, а також забезпечувати просте додавання нових функцій та можливостей.
5. Керованість. ПО для розумного дому повинно бути легким у використанні та керуванні. Користувач повинен мати зручний

інтерфейс для управління всіма системами та пристроями в розумному домі. Також важливо забезпечити можливість керування системою з різних пристроїв та з різних місць, включаючи мобільні пристрої.

6. Надійність. ПО для розумного дому повинно бути надійним та стабільним. Воно повинно бути здатним працювати без збоїв та перебоїв в роботі. Важливо забезпечити можливість відновлення роботи системи після можливих збоїв.
7. Інтегрованість. ПО для розумного дому повинно бути здатним інтегруватись з іншими системами та пристроями в домі. Важливо забезпечити можливість співпраці з пристроями та системами, що використовують різні протоколи та стандарти.
8. Підтримка. ПО для розумного дому повинно мати відповідну підтримку від розробників. Важливо забезпечити регулярні оновлення та підтримку користувачів у випадку проблем.

Усі ці вимоги є дуже важливими для розробки програмного забезпечення для розумного дому. Розробники повинні забезпечити, щоб програмне забезпечення стало ефективним, безпечним, масштабованим та легким у використанні та керуванні. Важливо також забезпечити надійність та інтегрованість програмного забезпечення з іншими системами та пристроями в розумному домі.

2.2. Архітектура програмного забезпечення розумного дому

2.2.1. Вибір архітектурної основи

В створенні підсистеми керування "розумним" домом застосовуються різні підходи до архітектури, залежно від вимог до системи. Деякі з можливих підходів включають централізовану, децентралізовану, гібридну, архітектуру з подіями та архітектуру з мікросервісами.

У таблиці нижче зібрані характеристики кожної з архітектур для порівняння:

Таблиця 2.1 – Порівняльна характеристика архітектури систем IoT

Архітектура	Характеристики
Централізована	Всі датчики та пристрої зв'язані з центральним контролером
Децентралізована	Різні пристрої та датчики можуть зв'язуватися між собою та приймати рішення автономно, без центрального контролера
Гібридна	Використовуються як централізовані, так і децентралізовані елементи
Архітектура з подіями	Система реагує на зміни через спеціальні "події". При цьому всі пристрої в системі можуть бути зв'язані між собою за допомогою протоколів обміну повідомленнями.
Архітектура з мікросервісами	Система будується з набору незалежних компонентів, які можуть взаємодіяти між собою за допомогою мережевих інтерфейсів

Враховуючи характеристики, зроблений висновок, що гібридна архітектура є оптимальним вибором для створення підсистеми керування "розумним" домом в ході даної дипломної роботи. Використання як централізованих, так і децентралізованих елементів дозволить забезпечити гнучкість управління підсистемою, зберігаючи при цьому певний рівень контролю над системою в цілому.

Але варто розуміти, що централізований контролер може бути використаний для координації та контролю за основними функціями "розумного" дому, такими як освітлення, опалення та безпека. У той же час, децентралізовані елементи, такі як датчики руху, температури та

вологості можуть збирати дані та взаємодіяти з іншими елементами без необхідності постійної координації з централізованим контролером.

Гібридна архітектура дозволяє зберігати рівень контролю над системою, забезпечуючи при цьому достатню гнучкість та масштабованість. Наприклад, якщо будуть додані нові елементи до системи, такі як пристрої для контролю вологості ґрунту або системи автоматичного поливу, їх можна буде додати як децентралізовані елементи, не впливаючи при цьому на роботу централізованого контролера.

Отже, гібридна архітектура є оптимальним вибором для створення підсистеми керування "розумним" домом, оскільки вона дозволяє зберігати рівень контролю над системою, забезпечуючи при цьому гнучкість та масштабованість. Однак, при виборі архітектури необхідно враховувати конкретні потреби та вимоги до системи, а також можливість інтеграції з іншими системами управління та зв'язку.

2.2.2. Вибір системи управління базою даних

Для порівняння різних СУБД можна використовувати такі параметри, як тип даних, швидкість роботи, масштабованість, надійність та вартість.

Ось порівняльна таблиця найпопулярніших СУБД:

Таблиця 2.2 – Порівняльна характеристика СУБД систем IoT

СУБД	Тип даних	Швидкість роботи	Масштабованість	Надійність	Вартість
MySQL	Реляційні	Висока	Середня	Висока	Безкоштовно
PostgreSQL	Реляційні	Висока	Висока	Висока	Безкоштовно
MongoDB	NoSQL	Середня	Висока	Середня	Безкоштовно
Cassandra	NoSQL	Висока	Висока	Висока	Безкоштовно
Microsoft	Реляційні	Висока	Висока	Висока	Платно

SQL Server					
Oracle	Реляційні	Висока	Висока	Висока	Платно

За результатами порівняння зроблений висновок, що вибір конкретної СУБД для підсистеми управління "розумним" домом залежить від багатьох факторів, включаючи тип даних, масштабованість та доступність функцій.

Якщо для зберігання даних "розумного" дому використовуються переважно реляційні дані та вимагається висока швидкість роботи, то MySQL, PostgreSQL, Microsoft SQL Server або Oracle можуть бути оптимальним вибором.

Якщо ж вимагається гнучкість та масштабованість в зберіганні даних "розумного" дому, то використання NoSQL баз даних, таких як MongoDB або Cassandra, може бути більш вигідним. Вони зазвичай мають кращу масштабованість та гнучкість в порівнянні з реляційними СУБД, що робить їх ідеальними для "розумних" домів, де зберігається велика кількість різноманітних даних.

Окрім цього, в залежності від бюджету проекту, можуть бути вибрані безкоштовні або платні СУБД. Якщо підприємство має обмежений бюджет, то безкоштовні СУБД, такі як MySQL, PostgreSQL або MongoDB, можуть бути оптимальним вибором. З іншого боку, якщо підприємство може дозволити собі платну СУБД, то Microsoft SQL Server або Oracle можуть бути кращим вибором.

Кращий вибір залежить від конкретних потреб проекту, і може бути зроблений після ретельного аналізу вищезгаданих параметрів. Тож обрана система управління базами даних MySql, яка покриває всі умови виконання даної дипломної роботи.

2.2.3. Включення інших архітектурних моментів

Окрім розподіленої архітектури та СУБД, іншими ключовими компонентами програмного забезпечення для розумного дому є:

1. Модуль збору та обробки даних: забезпечує збір даних з різних пристроїв та систем у розумному домі, а також їх обробку та аналіз.
2. Модуль управління системами та пристроями: забезпечує управління різними системами та пристроями в розумному домі, що дозволяє налаштувати та керувати різними параметрами, такими як температура, освітлення, безпека тощо.
3. Модуль управління користувачами: забезпечує інтерфейс для взаємодії користувачів з системою, включаючи можливість налаштування та контролю різних параметрів у розумному домі.
4. Модуль безпеки: забезпечує захист системи та даних, що збираються та оброблюються у розумному домі. Це може включати різні заходи безпеки, такі як автентифікація користувачів, захист від вірусів та зломів, шифрування даних тощо.

Загалом, архітектура програмного забезпечення для розумного дому повинна бути розподіленою, масштабованою та гнучкою, забезпечуючи високу продуктивність, безпеку та легку керованість. Для цього вона повинна включати різні компоненти, такі як СУБД, модуль збору та обробки даних, модуль управління системами та пристроями, модуль управління користувачами та модуль безпеки.

2.3.Розробка інтерфейсу користувача

2.3.1. Розробка інтерфейсу користувача

Для визначення вимог до інтерфейсу проведено аналіз потреб користувачів, їх звичок та вимог до функціоналу системи. При цьому звернено увагу на такі аспекти:

1. Функціональність: потрібно визначити, які функції повинен мати інтерфейс, щоб забезпечити комфортну та ефективну

роботу з системою управління розумним домом. Наприклад, можна врахувати необхідність вмикання та вимикання різних пристроїв, регулювання освітлення та температури в приміщеннях тощо.

2. Інтерфейс користувача: необхідно визначити, які інтерфейсні елементи потрібно використовувати, щоб забезпечити максимально зручну та інтуїтивно зрозумілу взаємодію з системою. Наприклад, можна використовувати кнопки, перетягування, використання голосових команд тощо.
3. Відповідність стандартам: необхідно врахувати вимоги до інтерфейсу, які встановлені стандартами. Наприклад, стандартами може бути визначено розмір кнопок, яскравість та контрастність кольорів тощо.
4. Адаптивність: необхідно врахувати можливість адаптування інтерфейсу до різних типів пристроїв та екранів. Наприклад, якщо користувач взаємодіє з системою через мобільний телефон або планшет, інтерфейс повинен бути адаптований до розміру екрану та можливостей цих пристроїв.
5. Естетика: необхідно врахувати естетичні аспекти інтерфейсу. Його дизайн повинен бути привабливим та зручним для користувача.

У процесі визначення вимог до інтерфейсу використані різні методи, наприклад, опитування користувачів, фокус–групи, аналіз конкурентів та їхніх інтерфейсів, дослідження вимог до інтерфейсу, які встановлені стандартами тощо.

В результаті визначення вимог до інтерфейсу складено вимоги до інтерфейсу, які повинні бути виконані під час розробки інтерфейсу системи управління розумним домом, а саме:

1. Інтерфейс повинен бути інтуїтивно зрозумілим та легко користуватися для користувачів різного рівня технічної освіти та досвіду.
2. Інтерфейс повинен бути адаптивним та респонсивним, тобто відображати інформацію на різних пристроях та екранах різного розміру, забезпечувати швидку реакцію на дії користувачів.
3. Інтерфейс повинен мати привабливий та сучасний дизайн, який буде відповідати останнім трендам та стандартам дизайну.
4. Інтерфейс повинен бути безпечним та забезпечувати захист від несанкціонованого доступу до даних користувачів та системи управління розумним домом.
5. Інтерфейс повинен підтримувати багатомовність та мати можливість зміни мови в залежності від потреб користувача.
6. Інтерфейс повинен забезпечувати зручний та швидкий доступ до всіх функцій системи управління розумним домом, включаючи можливість керування освітленням, опаленням, кондиціонерами, системою безпеки, розкриття та закриття вікон та дверей тощо.
7. Інтерфейс повинен мати можливість налаштування та персоналізації, тобто користувачі повинні мати можливість вибирати теми, додавати власні шаблони та налаштовувати інші параметри інтерфейсу.
8. Інтерфейс повинен мати можливість показувати статистику використання системи управління розумним домом, включаючи інформацію про споживання електроенергії, газу, води тощо, а також інформацію про стан системи та попередження про можливі несправності.
9. Інтерфейс повинен забезпечувати можливість отримання сповіщень про стан системи та повідомлення про події, такі як

спрацювання сигналізації, відкриття дверей, заповнення бака з газом або водою тощо.

10. Інтерфейс повинен забезпечувати можливість керування системою управління розумним домом з дистанції, наприклад, через мобільний додаток або веб–інтерфейс.
11. Інтерфейс повинен забезпечувати можливість підключення до інших систем управління, таких як системи розпізнавання голосу, інтернету речей тощо.
12. Інтерфейс повинен мати можливість налаштування прав доступу для різних користувачів, щоб забезпечити безпеку та контроль доступу до системи управління розумним домом.
13. Інтерфейс повинен мати можливість збереження журналу дій користувачів та системи управління розумним домом, щоб забезпечити можливість відстеження дій та аналізу в разі необхідності.
14. Інтерфейс повинен мати можливість резервного копіювання та відновлення налаштувань системи управління розумним домом, щоб забезпечити захист від втрати даних та можливість відновлення після непередбачуваних ситуацій.
15. Інтерфейс повинен мати можливість оновлення та покращення функцій системи управління розумним домом, щоб забезпечити можливість адаптації до нових технологій та потреб користувачів.

Наведені вимоги не повинні бути виконані в повній мірі реалізації програмування самого продукту, але при певному рівні інвестування, кожен пункт стане потрібним функціональним розширенням найпростішого продукту.

2.3.2. Вибір технологій та інструментів для розробки інтерфейсу

Вибір технологій та інструментів для розробки інтерфейсу системи управління розумним став важливим етапом у проектуванні програмного забезпечення. Для розробки інтерфейсу розглянуті різні технології та інструменти, які мають свої переваги та недоліки. Наприклад:

1. Web–технології: розробка інтерфейсу з використанням web–технологій, таких як HTML, CSS, JavaScript та інші, дозволяє створювати кросплатформенні інтерфейси, які можуть працювати на різних пристроях та операційних системах. Однак, ці технології можуть вимагати значних ресурсів для роботи на пристроях з обмеженими можливостями, а також можуть бути менш ефективними у взаємодії з датчиками та іншими пристроями у розумному домі.
2. Нативні додатки: розробка нативних додатків для кожної платформи (iOS, Android, Windows і т.д.) дозволяє створювати інтерфейси, які оптимізовані для певної платформи та її функцій. Ці додатки можуть бути більш ефективними взаємодії з датчиками та іншими пристроями у розумному домі, але вимагають більше ресурсів для розробки та підтримки.
3. Хмарні технології: використання хмарних технологій, таких як Amazon Web Services, Microsoft Azure або Google Cloud Platform, дозволяє створювати інтерфейси, які можуть бути доступні з будь–якого місця, де є інтернет–з'єднання. Однак, використання хмарних технологій може вимагати високої швидкості Інтернет–з'єднання, щоб забезпечити ефективну роботу інтерфейсу, а також може залежати від стійкості і доступності хмарних сервісів.

Для вибору технологій та інструментів для розробки інтерфейсу враховані наступні фактори:

1. Функціональні вимоги до інтерфейсу: необхідно зрозуміти, які функції повинен виконувати інтерфейс системи управління

розумним домом, які елементи керування повинні бути присутні, які дані повинні бути відображені тощо.

2. Ресурси пристроїв: необхідно враховувати можливості та обмеження пристроїв, на яких буде запущена система управління розумним домом. Наприклад, важливо визначити, яку кількість пам'яті, обчислювальних потужностей та інших ресурсів може виділити кожен пристрій для роботи з інтерфейсом.
3. Досвід користувачів: необхідно враховувати очікування та вимоги користувачів щодо інтерфейсу системи управління розумним домом. Наприклад, якщо більшість користувачів використовують мобільні пристрої, то слід визначити, які мобільні платформи є популярними серед цієї аудиторії та які функції повинні бути присутні в мобільному додатку.
4. Бюджет та терміни: необхідно враховувати можливості замовника проекту щодо бюджету та термінів розробки. Вибір технологій та інструментів повинен бути оптимальним з точки зору вартості та швидкості розробки.

Основними технологіями та інструментами, які можуть бути використані для розробки інтерфейсу системи управління розумним домом, є:

1. HTML, CSS та JavaScript: ці технології є стандартними для розробки веб–інтерфейсів та можуть бути використані для створення веб–інтерфейсу системи управління розумним домом. Вони дозволяють створювати багатофункціональний та динамічний інтерфейс.
2. React та React Native: ці технології є популярними для розробки мобільних додатків. React дозволяє створювати інтерактивні інтерфейси за допомогою JavaScript, а React Native дозволяє розробляти мобільні додатки для Android та iOS за допомогою React.
3. Flutter: це фреймворк для розробки мобільних додатків, що дозволяє створювати кросплатформні додатки для Android та iOS. Flutter

використовує мову програмування Dart та має багатий набір готових компонентів для створення інтерфейсу.

4. Windows forms c#: це технологія для розробки віконних застосунків мовою C#, але вона також може бути використана для розробки інтерфейсу. WF має вбудований редактор, що дозволяє створювати інтерфейси.
5. Adobe XD: це інструмент для дизайну інтерфейсів, який дозволяє створювати прототипи та макети інтерфейсу. Adobe XD дозволяє швидко створювати дизайн та перевіряти його на різних пристроях та розмірах екрану.

Під час вибору технологій та інструментів для розробки інтерфейсу також враховані функціональні вимоги, обмеження ресурсів пристроїв, очікування користувачів, бюджет, терміни розробки. Також на наявність готових рішень, наприклад, фреймворків та бібліотек, які дозволять швидко та ефективно розробити інтерфейс. Враховані можливість інтеграції з іншими системами управління розумним домом, які вже використовуються в проекті.

При виборі технологій та інструментів розглянуті питання безпеки даних, оскільки інтерфейс зазвичай забезпечує доступ до важливих даних про стан системи управління розумним домом. Тому обрані технології та інструменти, мають високий рівень безпеки та можуть захистити дані від несанкціонованого доступу.

Крім того, не проігноровано питання про підтримку технологій та інструментів, оскільки проект розробляється на довгий термін, хоч і в демонстраційному, елементарному виді.

Отже, вибір технологій та інструментів для розробки інтерфейсу системи управління розумним домом важливий етап проектування програмного забезпечення. Тому з урахуванням всієї інформаційної бази, мовою та інструментом стали C# з використанням WINDOWS FORMS.

2.4. Проектування візуальної частини інтерфейсу з урахуванням зручності та ефективності використання

Першим етапом проектування візуальної частини інтерфейсу є аналіз потреб користувачів та їх поведінки при взаємодії з системою управління розумним домом. Для цього проведені дослідження з визначенням пріоритетів користувачів та їх очікувань від системи. На основі отриманих даних визначено основні функції та інтерфейсні елементи, які найбільше важливі для користувачів.

Далі необхідно створено макет інтерфейсу з урахуванням принципів зручності та ефективності використання. Макет відображає основні функції системи та інтерфейсні елементи, які дозволяють користувачеві зручно та швидко взаємодіяти з системою. На цьому етапі дотримано принципу мінімалізму та забезпечено максимально простий та зрозумілий інтерфейс.

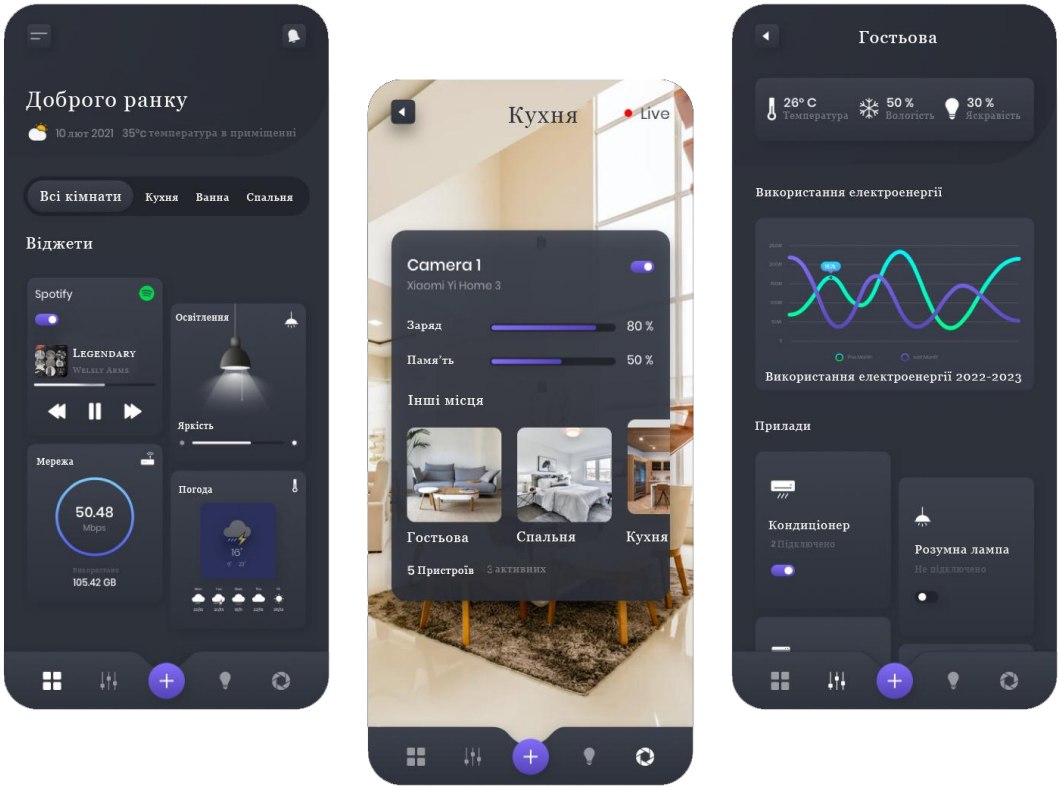


Рисунок 2.1 – Макет інтерфейсу керування у вигляді мобільного додатку

Після створення макету проведено тестування інтерфейсу на різних групах користувачів. Тестування дозволило виявити недоліки та помилки в інтерфейсі, які вплинули на зручність та ефективність використання системи. На основі результату отриманого під час тестування був дороблений макет інтерфейсу та введені необхідні зміни. Проблеми:

1. Неінтуїтивний інтерфейс: Інтерфейс мобільного додатку управління розумним домом не був простим та інтуїтивним для користувача.

Наступним етапом проектування візуальної частини інтерфейсу стала розробка дизайну інтерфейсу. Дизайн став зрозумілим та привабливим для користувача, а також відображає концепцію розумного дому. Використані правильні кольорові схеми та шрифти, та забезпечено максимальну зручність та ефективність використання системи.

2.5. Розробка функціональності інтерфейсу

Інтерфейс повинен бути зручним і інтуїтивно зрозумілим для користувача, а також повинен дозволяти легко взаємодіяти з різними пристроями і системами дому.

Перш за все, інтерфейс повинен бути доступним для користувача на різних пристроях, таких як смартфон, планшет, комп'ютер, голосовий асистент і т.д. Для цього використовуються різні технології, наприклад, веб–інтерфейс, мобільні додатки, голосові команди, а також інтерфейс з використанням жестів.

Основна функціональність інтерфейсу повинна дозволяти користувачеві керувати різними пристроями і системами в домі. Для цього використовується графічний інтерфейс з використанням кнопок, перемикачів, регуляторів, список команд і т.д. Крім того, є можливість

розширення налаштування таймерів, розкладів, сценаріїв, що дозволить автоматизувати процес управління домом.

Інтерфейс також може доповнюватися додатковою функціональністю, такою як моніторинг систем безпеки, керування терморегулюванням, контроль за витратами електроенергії та води. Це дозволить користувачеві отримувати повну інформацію про стан дому і ефективно керувати його ресурсами.

Крім того, інтерфейс забезпечений відповідними засобами безпеки, що дозволить захистити систему від несанкціонованого доступу та злому. Для цього використовується авторизацію, шифрування даних та інші заходи забезпечення безпеки. Для цього використовуються різні технології, такі як SSL/TLS протоколи для захисту від перехоплення трафіку, парольні системи та системи двофакторної автентифікації.

Важливо забезпечити можливість збереження історії взаємодії користувача з системою управління розумним домом. Це дозволить користувачеві переглядати історію своїх дій, а також відновлювати попередні налаштування у випадку потреби.

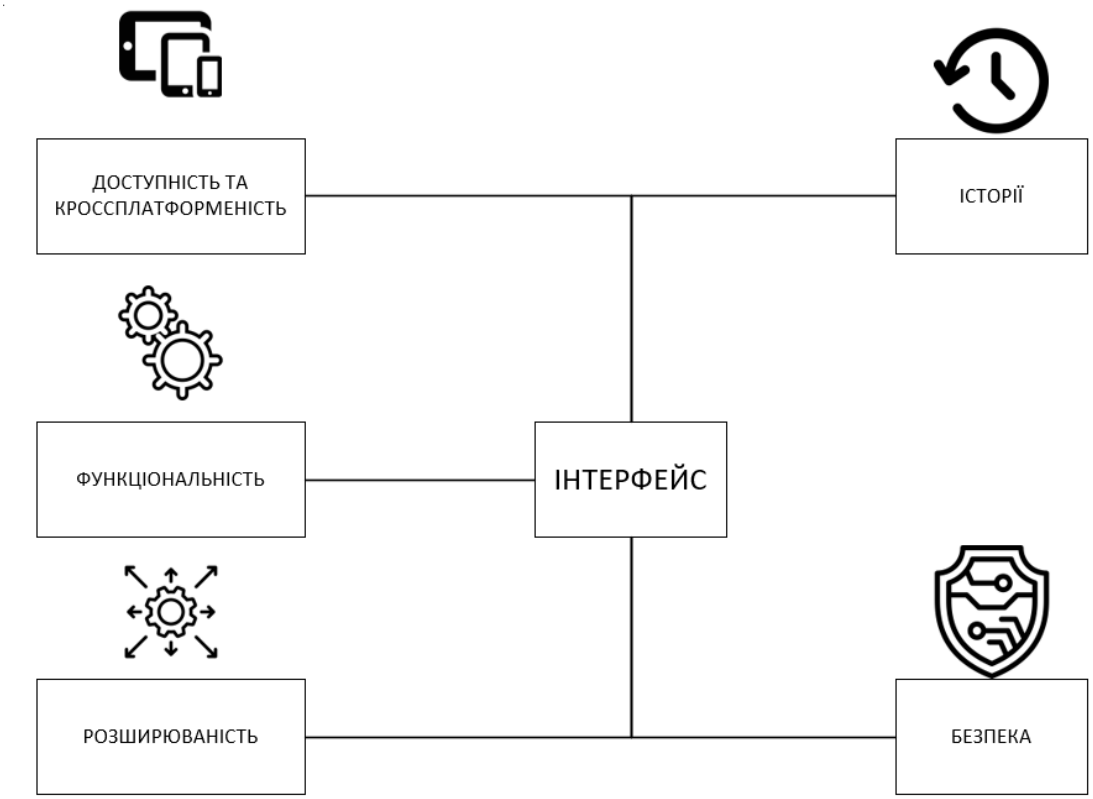


Рисунок 2.2 – функціональні блоки інтерфейсу

Загалом, розробка функціональності інтерфейсу стала важливою складовою підсистеми управління розумним домом. Вона дозволяє користувачам зручно та ефективно керувати різними пристроями та системами в домі, а також забезпечує захист від несанкціонованого доступу та злому. При розробці інтерфейсу враховувано потреби та особливості користувачів, а також використано найновіші технології та підходи для забезпечення оптимальної роботи системи.

2.6. Use case діаграма інтерфейсу

Ідея Use case діаграми інтерфейсу полягає в ідентифікації та моделюванні різних сценаріїв використання системи, а також визначення основних функцій та можливостей, доступних користувачам. Цей аналіз допомагає розробити інтуїтивно зрозумілий та ефективний інтерфейс, що забезпечує зручне керування різними аспектами розумного дому, такими як освітлення, опалення, безпека та інші.

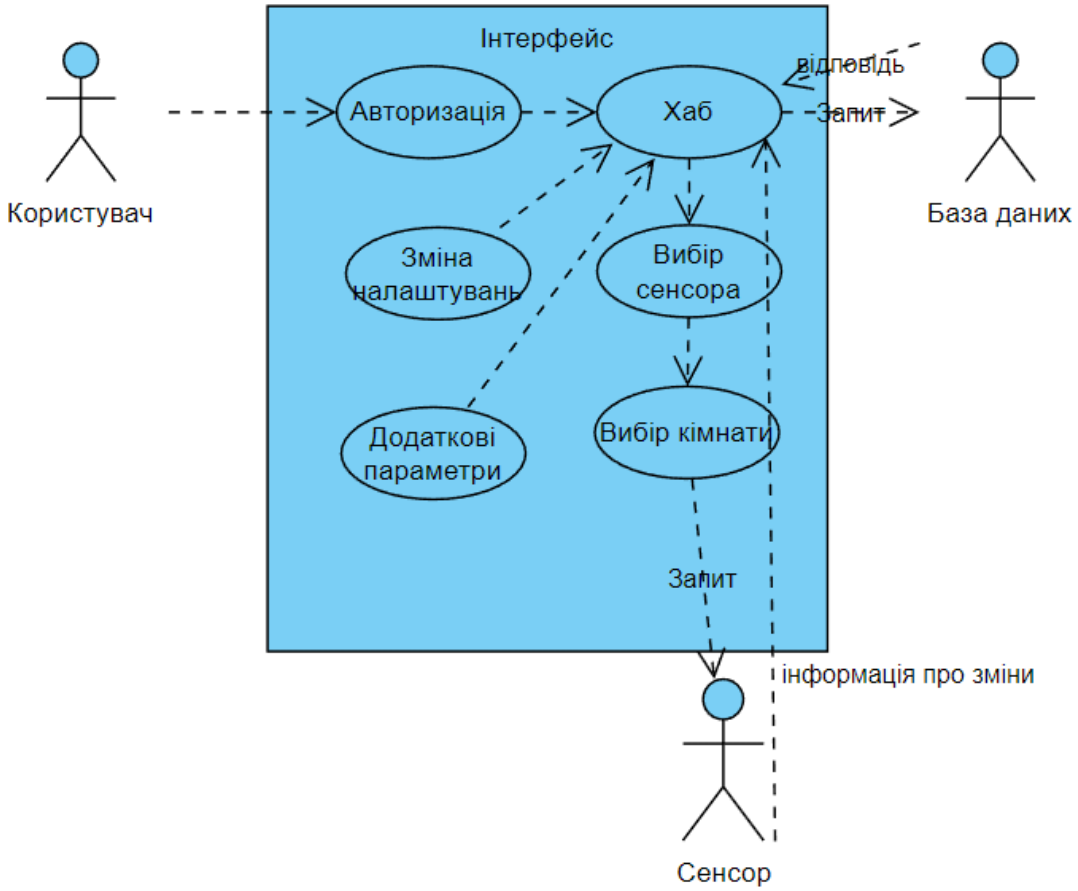


Рисунок 2.3 – Use case діаграма інтерфейсу

2.7.Представлення структурної схеми функцій

Структурна схема описана нижче та відображена на рисунку 2.4

1. Головна система управління розумним домом
   1.1.Керування освітленням
   1.2.Керування опаленням
   1.3.Керування вентиляцією
   1.4.Керування безпекою
   1.5.Керування енергозбереженням
2. Підсистема керування освітленням
   2.1.Управління освітленням за розкладом
   2.2.Регулювання яскравості освітлення
   2.3.Детекція руху для автоматичного включення/вимикання

3. Підсистема управління термостатом
   3.1. Налаштування температури
   3.2. Регулювання режимів опалення та охолодження
   3.3. Розклад температурних режимів Зв'язок з датчиками температури
   3.4. Автоматичне вимкнення/увімкнення опалення/охолодження при відсутності людей в будинку
4. Підсистема управління камерами
   4.1. Запис відео та збереження зображень
   4.2. Налаштування режимів відеоспостереження (нагляд/розпізнавання облич/розпізнавання руху)
   4.3. Зв'язок з датчиками руху для автоматичного включення/вимикання запису
   4.4. Доступ до відео з віддаленого пристрою через мобільний додаток або веб-інтерфейс
5. Підсистема управління датчиками відкриття та закриття
   5.1. Моніторинг стану дверей, вікон, гаражних воріт тощо
   5.2. Відображення статусу датчиків відкриття/закриття на інтерфейсі
   5.3. Надсилання повідомлень про відкриті двері/вікна, коли немає присутності в будинку
   5.4. Інтеграція зі системою безпеки для активації сигналізації при несанкціонованому відкритті
6. Підсистема управління датчиками газу та диму
   6.1. Виявлення наявності газу та диму
   6.2. Відображення статусу датчиків газу/диму на інтерфейсі
   6.3. Надсилання сповіщень та активація сигналізації при виявленні газу/диму
   6.4. Автоматичне вимкнення електроприладів при виявленні газу/диму
7. Підсистема управління водою
   7.1. Керування системою поливу саду або газону
   7.2. Моніторинг рівня води в басейні або резервуарі

7.3. Автоматичне вимкнення подачі води при досягненні заданого рівня

8. Підсистема управління електроенергією

8.1. Моніторинг споживання електроенергії за різними пристроями та зонами

8.2. Управління освітленням за розкладом або на основі датчиків руху

8.3. Вимкнення електроприладів у режимі очікування або при відсутності присутності в приміщенні

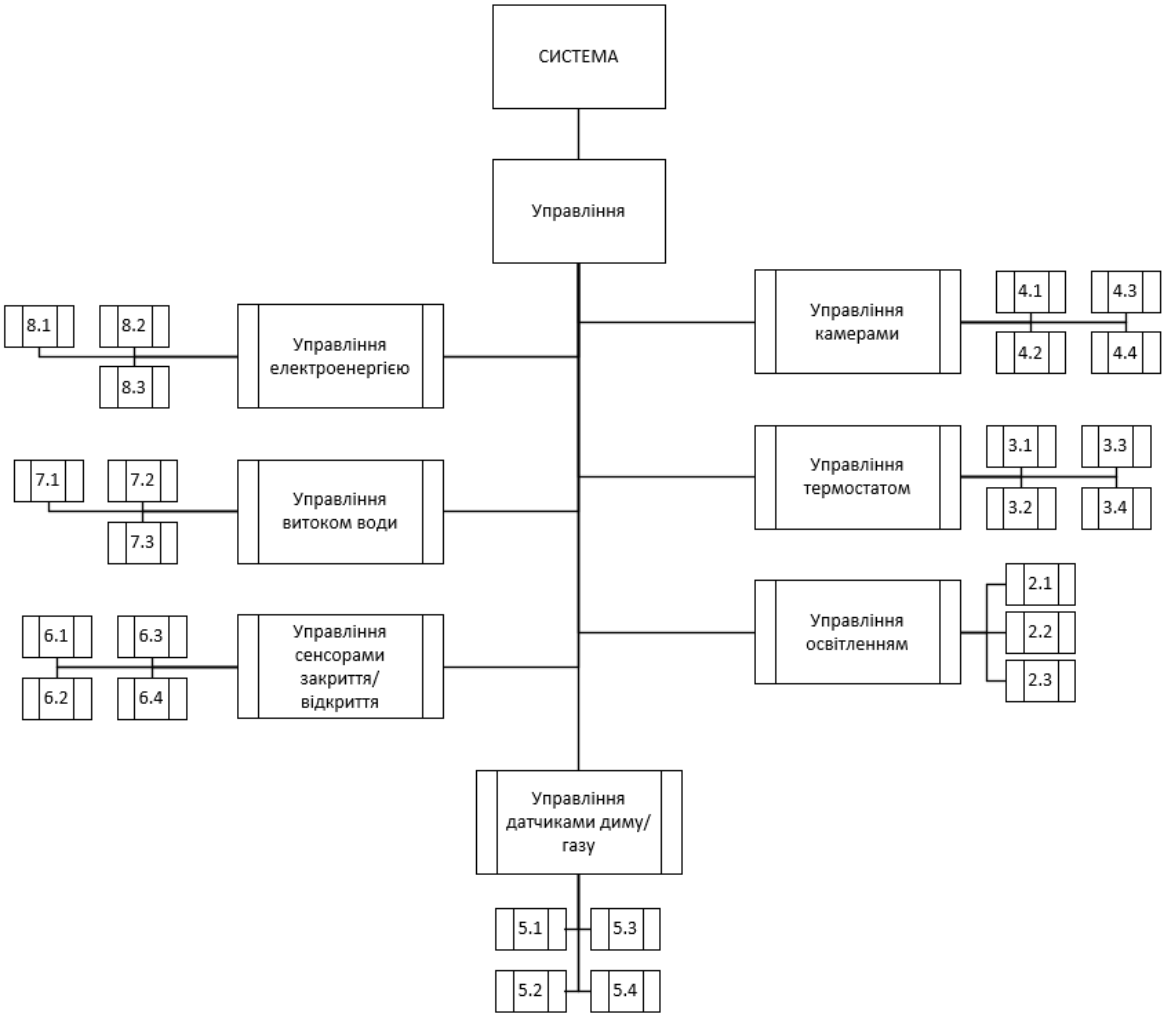


Рисунок 2.4 – представлення структурної схеми функцій

2.8. Постановка питань розробки системи управління

Для створення простої підсистеми управління розумним домом необхідно виконати наступні кроки:

1. Визначити функціональні вимоги до підсистеми управління, тобто визначити, які пристрої вона повинна керувати і які операції повинна виконувати.
2. Обрати платформу, на якій буде розроблятися підсистема управління. Найпоширеніші платформи для розумного дому – це Raspberry Pi, Arduino та ESP8266. Вибір платформи залежить від функціональних вимог до системи та ресурсів, які вона повинна мати.
3. Розробити алгоритми управління для підсистеми. Алгоритми повинні визначати, як система повинна реагувати на події, наприклад, як ввімкнути світло, коли ввімкнути кондиціонер, як контролювати температуру в приміщенні та інші події.
4. Розробити інтерфейс користувача, який дозволяє взаємодіяти з підсистемою управління. Інтерфейс може бути реалізований за допомогою мобільного додатку, веб–інтерфейсу або інтерфейсу на екрані.
5. Підключити пристрої до системи управління. Для цього необхідно з'єднати пристрої з платформою та написати програмне забезпечення, яке дозволяє управляти пристроями.
6. Провести тестування системи. Тестування повинно включати перевірку коректності роботи системи та її стабільності.
7. Внести необхідні корективи та вдосконалення у систему на основі результатів тестування.

В результаті цих кроків буде розроблена проста підсистема управління розумним домом, яка зможе керувати деякими пристроями та виконувати деякі базові операції, наприклад, вмикати та вимикати освітлення, контролювати температуру в приміщенні, відкривати та закривати двері та інші дії.

У процесі розробки підсистеми управління розумним домом, необхідно дотримуватися стандартів та протоколів зв'язку, які

використовуються в розумному домі, щоб забезпечити сумісність з іншими пристроями та системами.

Також важливо забезпечити безпеку підсистеми управління, щоб запобігти несанкціонованому доступу до системи та пристроїв, які підключені до неї.

У процесі розробки підсистеми управління розумним домом, можуть виникати труднощі, пов'язані з підключенням та взаємодією з пристроями, проблемами з електропостачанням та інші проблеми. Тому, важливо забезпечити належну підтримку та обслуговування підсистеми управління після її впровадження.

У загальному, розробка підсистеми управління розумним домом – це складний процес, який вимагає знань з різних областей, таких як програмування, електроніка, мережі та безпека. Однак, він може бути виконаний успішно за допомогою правильного планування та належної експертизи. В основі диплому, як науково дослідницької роботи, можуть бути пропущені деякі з цих підпитань, так як вони потребують фінансування.

2.9. Тестування та валідація програмного забезпечення

Першим етапом тестування є розробка тестових сценаріїв, які дозволяють відтворити різні ситуації, які можуть статися при використанні програмного забезпечення. Наприклад, тестові сценарії можуть включати в себе тестування реакції системи на різні події, такі як ввімкнення певного пристрою, зміна налаштувань тощо.

Після розробки тестових сценаріїв проводиться тестування програмного забезпечення. Цей процес може бути автоматизованим або виконуватися вручну. Автоматизоване тестування дозволяє швидко та ефективно перевіряти роботу програмного забезпечення в різних

сценаріях. Вручну ж тестування проводять, коли необхідно протестувати взаємодію користувача з системою.

Під час тестування відслідковуються та фіксуються всі помилки та недоліки, які були виявлені під час процесу. Після завершення тестування виконується валідація програмного забезпечення, що полягає у перевірці його відповідності вимогам та очікуванням замовника та користувачів.

У процесі різні методи, наприклад, методи контролю якості, тестування відповідності функціональності та тестування реакції на стандарти безпеки, а також тестування на забезпечення захисту від злому та витоку даних.

Для забезпечення ефективного тестування та валідації програмного забезпечення використані різні інструменти, такі як програмне забезпечення для автоматизації тестування та інструменти для аналізу коду.

У разі виявлення помилок та недоліків під час тестування, розробник виправляє чи перетворює на фічу їх та перевіряє, чи не вплинули вони на інші функції та взаємодії системи. Після виправлення помилок необхідно повторно протестувати програмне забезпечення та перевірити, чи були виправлені всі помилки та недоліки.

Таким чином, процес тестування та валідації програмного забезпечення розумного дому дозволяє забезпечити його надійну та безперебійну роботу, а також відповідність вимогам та очікуванням користувачів.

2.10. Управляючі функції та їх структура

У цьому розділі досліджується структура та функціональні можливості розробленої підсистеми управління розумного дому. Основною метою підсистеми є забезпечення користувачам зручного та ефективного керування всіма пристроями та системами, що входять до

складу розумного дому. Розглянемо докладніше управляючі функції та їх структуру.

2.10.1. Структура управляючої підсистеми

Управляюча підсистема складається з декількох компонентів, які взаємодіють між собою для забезпечення коректної роботи системи. Основними компонентами структури є:

1. Керуючий центр (Control Center): Цей компонент є мозком системи управління розумного дому. Він відповідає за збір, обробку та аналіз даних від різних датчиків і пристроїв у розумному домі. Керуючий центр також забезпечує інтерфейс для взаємодії з користувачем та приймає команди керування.
2. Датчики та актуатори (Sensors and Actuators): Ці компоненти відповідають за збір інформації про стан різних систем у розумному домі (наприклад, температура, освітлення, вологість) за допомогою датчиків, а також за виконання дій на пристроях (наприклад, вмикання світла, регулювання температури) за допомогою актуаторів.
3. Комунікаційний протокол (Communication Protocol): Цей компонент відповідає за передачу даних між різними пристроями та системами в розумному домі. Він забезпечує надійну та безпечну комунікацію між всіма компонентами системи і забезпечує синхронізацію даних. Комунікаційний протокол може використовувати різні технології, такі як Wi-Fi, Bluetooth, Zigbee або протоколи Інтернету речей (IoT).
4. База даних (Database): Цей компонент зберігає інформацію про стан різних пристроїв та систем в розумному домі. Він дозволяє зберігати історичні дані, налаштування, розклади та інші важливі параметри для подальшого аналізу та керування.

5. Інтерфейс користувача (User Interface): Цей компонент забезпечує зручний інтерфейс для взаємодії з користувачем. Це може бути мобільний додаток, веб-інтерфейс або голосовий помічник, який дозволяє користувачеві контролювати та налаштовувати різні пристрої та системи в розумному домі.

2.10.2. Управляючі функції

Управляючі функції підсистеми управління розумного дому включають:

1. Управляючі функції підсистеми управління розумного дому включаюМоніторинг та збір даних: Підсистема здійснює постійний моніторинг стану різних пристроїв та систем в розумному домі за допомогою датчиків. Вона збирає дані про температуру, освітлення, вологість, енергоспоживання та інші параметри.
2. Аналіз та обробка даних: Зібрані дані проходять через процес аналізу та обробки. Застосовуються алгоритми та методи аналізу даних, щоб отримати корисну інформацію, наприклад, прогнозування споживання енергії, розпізнавання патернів споживання, виявлення аномалій тощо.
3. Прийняття рішень: На основі оброблених даних та заданих параметрів робумовлень, підсистема управління розумного дому приймає рішення щодо керування пристроями та системами. Ці рішення можуть включати автоматичне регулювання температури, освітлення, вентиляції, безпеки та інших аспектів забезпечення комфорту та ефективності розумного дому.
4. Керування пристроями та системами: Після прийняття рішення підсистема управління розумного дому виконує необхідні дії

на пристроях та системах. Вона може надсилати команди до актуаторів для вмикання або вимикання пристроїв, регулювання параметрів, запуску задач тощо. Контроль та зворотний зв'язок: Після виконання команд підсистема управління розумного дому здійснює контроль за станом пристроїв та систем. Вона отримує зворотний зв'язок щодо успішності виконання команд, стану пристроїв та споживання ресурсів. Це дозволяє системі виявляти проблеми, вирішувати конфлікти та забезпечувати надійну роботу розумного дому.

2.10.3. Взаємодія з користувачем

1. Управляюча підсистема розумного дому забезпечує зручний інтерфейс для взаємодії з користувачем. Це може бути мобільний додаток, веб-інтерфейс або голосовий помічник, залежно від вибраної платформи та доступних технологій. Користувач має можливість:
2. Керувати пристроями та системами: Користувач може виконувати різні команди для керування пристроями та системами в розумному домі. Він може вмикати або вимикати пристрої, налаштовувати параметри, створювати розклади обумовлення та автоматизовані сценарії.
3. Налаштування параметрів: Користувач може налаштовувати параметри різних пристроїв та систем, наприклад, регулювати температуру, освітлення, режими безпеки, розклади роботи та інші важливі налаштування.
4. Моніторинг та статистика: Підсистема управління розумного дому надає користувачу доступ до моніторингу стану пристроїв, споживання ресурсів, статистики та звітів. Користувач може переглядати дані про споживання енергії,

температурні графіки, журнали подій та іншу інформацію, яка допомагає краще керувати розумним домом.

5. Розклади та автоматизація: Користувач може налаштовувати розклади для автоматичного виконання певних дій. Наприклад, встановлення часу вмикання або вимикання світла, температурних режимів в різні періоди доби, активація систем безпеки під час відсутності.
6. Зворотній зв'язок та сповіщення: Підсистема управління може надсилати повідомлення користувачеві про стан пристроїв, сповіщати про виникнення подій або незвичайні ситуації. Наприклад, попередження про витік води, сповіщення про незакриті двері або спрацювання пожежної сигналізації.

Всі ці функції та взаємодія з користувачем забезпечують зручний та ефективний спосіб керування розумним домом, надають користувачеві повний контроль та можливість налаштування різних аспектів життя в будинку.

3. РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

3.1.Вибір програмного інструментарію

3.1.1. Вибір мови програмування

У цьому розділі розглянуті три популярні мови програмування – C#, Python та C++ – для використання при розробці підсистем управління "розумним домом". Оцінені їх особливості, переваги та недоліки, а також створена порівняльна таблицю, яка допомогла прийняти рішення щодо вибору найбільш підходящої мови програмування даного проекту.

1. C#

C# – це об'єктно–орієнтована мова програмування, розроблена компанією Microsoft. Вона широко використовується для розробки додатків під платформу .NET і має декілька переваг для розробки підсистем управління "розумним домом":

- Легкість використання: C# має зрозумілу синтаксичну структуру, що дозволяє швидко створювати програми і зменшує кількість можливих помилок.
- Підтримка .NET Framework: C# надає доступ до багатьох потужних бібліотек і фреймворків .NET, які спрощують розробку рішень "розумного дому".
- Висока продуктивність: Хоча C# є мовою програмування з високим рівнем абстракції, вона дозволяє отримувати високу продуктивність завдяки оптимізаціям компілятора та ефективному використанню ресурсів.
- Масштабованість: C# має підтримку для розробки розподілених систем, що дозволяє створювати розумні доми зі складними архітектурами та великою кількістю компонентів.

2. Python

Python – це інтерпретована мова програмування, яка має велику популярність в галузі розробки програмного забезпечення, включаючи системи управління "розумним домом". Основні переваги Python для таких проектів включають:

- Простота та легкість використання: Python має просту синтаксичну структуру, яка дозволяє швидко розробляти програми. Вона також відома своєю читабельністю, що сприяє зручності розуміння коду і співпраці між розробниками.
- Багатий екосистема: Python має велику кількість сторонніх бібліотек і модулів, які полегшують розробку "розумних домів". Наприклад, бібліотека RPi.GPIO дозволяє взаємодіяти з Raspberry Pi для керування різними пристроями.
- Підтримка різних платформ: Python підтримується на різних операційних системах і пристроях, включаючи Linux, Windows і macOS. Це дає можливість розгортати рішення "розумних домів" на різних пристроях.
- Інтеграція з іншими мовами: Python легко інтегрується з мовами програмування низького рівня, такими як C і C++, що дає можливість використовувати оптимізований код для вимогливих завдань.

3. C++

C++ – це мова програмування загального призначення, яка використовується для розробки високоефективного програмного забезпечення. Деякі переваги використання C++ для підсистем управління "розумним домом" включають:

- Висока продуктивність: C++ дозволяє писати код з високою швидкодією та ефективністю. Це особливо корисно, коли ми маємо справу зі швидкими потоками даних або великою кількістю обчислень.

- Близькі до мови машинного коду: C++ надає можливість прямого доступу до пам'яті та низькорівневих операцій, що дозволяє оптимізувати код для вимогливих за ресурсами задач.
- Розширені можливості: C++ має широкий набір функцій і бібліотек, які дозволяють розробникам ефективно виконувати завдання з управління "розумним домом". Наприклад, бібліотека Boost надає інструменти для роботи з мережами, паралельним програмуванням та багатьма іншими задачами.
- Переносимість: C++ код може бути скомпільований для різних платформ, що дозволяє розгортати рішення "розумного дому" на різних пристроях і операційних системах.

Щоб краще зрозуміти відмінності між цими мовами програмування, створена порівняльна таблицю, яка підкреслить їх основні особливості:

Таблиця 3.1 – Порівняння мов для використання їх у розробці підсистеми управління розумним домом

Особливості	C#	Python	C++
Синтаксис	Схожий до Java	Простий і зрозумілий	Складний і гнучкий
Екосистема	Розширена, .NET Framework	Багата, сторонні бібліотеки	Широкий вибір бібліотек
Продуктивність	Висока	Зазвичай помірна	Висока
Переносимість	Залежить від .NET Framework	Добра	Добра
Масштабованість	Добра	Залежить від використання	Добра
Інтеграція з іншими мовами	Через .NET	Легка	Легка

З урахуванням особливостей кожної мови програмування, а також вимог проекту, зроблено висновок, що C# є найбільш підходящим варіантом для розробки підсистем управління "розумним домом". Вибір C# обґрунтований наступними причинами:

- Легкість використання: C# має простий і зрозумілий синтаксис, що полегшує розробку програм та зменшує кількість можливих помилок. Це особливо важливо, коли маєте справу зі складними системами "розумного дому".
- Підтримка .NET Framework: C# є основною мовою для розробки під платформу .NET. Це надає доступ до багатофункціональних бібліотек і фреймворків, які значно спрощують розробку систем управління "розумним домом".
- Масштабованість: C# має підтримку для розробки розподілених систем, що дає можливість створювати розумні доми зі складними архітектурами та великою кількістю компонентів.
- Інтеграція з іншими мовами: C# добре інтегрується з мовами програмування низького рівня, що дозволяє використовувати оптимізований код для вимогливих завдань.
- Підтримка різних платформ: C# код може бути виконаний на різних платформах, включаючи Windows, Linux і macOS. Це дає можливість розгортати рішення "розумного дому" на різних пристроях і операційних системах.

Загалом, C# має необхідні можливості і інструменти для розробки підсистем управління "розумним домом".

3.1.2. Вибір середовища розробки:

Розглянемо три популярні інструменти: Microsoft Visual Studio, SharpDevelop та Rider. Для кращого порівняння цих середовищ наведено характеристики та переваги кожного з них.

1. Microsoft Visual Studio:

Опис: Microsoft Visual Studio є інтегрованою середовище розробки (IDE) від Microsoft, яке підтримує різні мови програмування, включаючи C#, Visual Basic, C++, F# та інші.

1.1.Характеристики:

- Можливості: Велика кількість інструментів для розробки, розширення та налагодження програмного забезпечення.
- Сумісність: Повна підтримка платформи .NET, що дозволяє розробляти програми для різних операційних систем, включаючи Windows, macOS та Linux.
- Екосистема: Велика спільнота розробників, багато сторонніх розширень та плагінів.

1.2.Переваги:

- Могутній інструмент для розробки програмного забезпечення в середовищі .NET.
- Інтеграція з іншими продуктами та сервісами Microsoft.
- Широкий набір інструментів для розробки, відладки та тестування програм.

1.3.Недоліки:

- Вимагає наявності платної ліцензії для повного функціоналу.
- Великі вимоги до апаратних ресурсів комп'ютера.

2. SharpDevelop:

Опис: SharpDevelop є безкоштовним середовищем розробки для мови програмування C# та інших мов, розроблених на платформі .NET.

2.1.Характеристики:

- Можливості: Підтримка основних функцій розробки, таких як редактор коду, відладка, система керування версіями.
- Сумісність: Розробка програмного забезпечення для операційних систем Windows.

- Екосистема: Хоча SharpDevelop не має такої великої спільноти, як Visual Studio, все ж доступні деякі сторонні розширення та плагіни.

2.2.Переваги:

- Безкоштовне програмне середовище з базовими функціями розробки.
- Добре підходить для початківців або проектів з невеликими вимогами.
- Простий та легкий у використанні.

2.3.Недоліки:

- Обмежений функціонал порівняно з Visual Studio.
- Менша кількість інструментів та розширень.

3. Rider:

Опис: Rider – це інтегроване середовище розробки, розроблене компанією JetBrains, яке підтримує різні мови програмування, включаючи C#, VB.NET, F#, JavaScript та багато інших.

3.1.Характеристики:

- Можливості: Має широкий набір функцій, включаючи потужний редактор коду, відлагування, рефакторинг, підтримку Git та інші інструменти розробки.
- Сумісність: Підтримка платформи .NET, що дозволяє розробляти програми для різних операційних систем, включаючи Windows, macOS та Linux.
- Екосистема: JetBrains має широку спільноту розробників і підтримує багато сторонніх плагінів та розширень.

3.2.Переваги:

- Могутній та швидкий інструмент розробки з багатьма продуктивними функціями.
- Висока інтеграція з іншими продуктами JetBrains.
- Підтримка кросплатформеного розробки.

3.3.Недоліки:

- Вимагає платної ліцензії для повного функціоналу.
- Великі вимоги до апаратних ресурсів комп'ютера.

Таблиця 3.2 – Порівняльна таблиця для середовищ

Характеристика	Microsoft Visual Studio	SharpDevelop	Rider
Мови програмування	C#, Visual Basic, C++, F# та інші	C# та інші мови на платформі .NET	C#, VB.NET, F#, JavaScript та інші
Ціна	Платна	Безкоштовна	Платна
Платформи підтримки	Windows, macOS, Linux	Windows	Windows, macOS, Linux
Інструменти та функціонал	Велика кількість інструментів, розширень та плагінів	Базовий функціонал розробки	Широкий набір продуктивних функцій
Екосистема	Велика спільнота розробників, підтримка сторонніх розширень та плагінів	Обмежена спільнота розробників, деякі сторонні розширення	Велика спільнота розробників, багато сторонніх плагінів та розширень
Легкість використання	Потребує певного часу для ознайомлення, але забезпечує широкий функціонал	Простий та легкий у використанні, ідеальний для початківців	Має деяку крутішу криву навчання, але надає більше продуктивних можливостей

Апаратні вимоги	Високі	Невисокі	Високі

У порівнянні цих трьох програмних середовищ для розробки підсистем управління "розумним домом", Microsoft Visual Studio є найкращим вибором. Воно забезпечує широкий набір інструментів та функціоналу, є популярним серед розробників, має велику спільноту та підтримує розробку на різних платформах. Враховуючи, що розробка "розумного дому" може потребувати великої функціональ ності та інтеграції з іншими сервісами, Microsoft Visual Studio надає потужність та гнучкість для реалізації проекту.

3.2. Розробка інтерфейсної частини підсистеми управління ррозумним домом

Розділ є важливою складовою процесу створення розумних систем, оскільки надає зручні та інтуїтивно зрозумілі засоби комунікації користувача з системою. Метою цього розділу є детальне описання процесу розробки інтерфейсної частини програмного забезпечення для розумного дому. Тож загальний вигляд розробленої, теоретичної та простої системи, виглядає так:

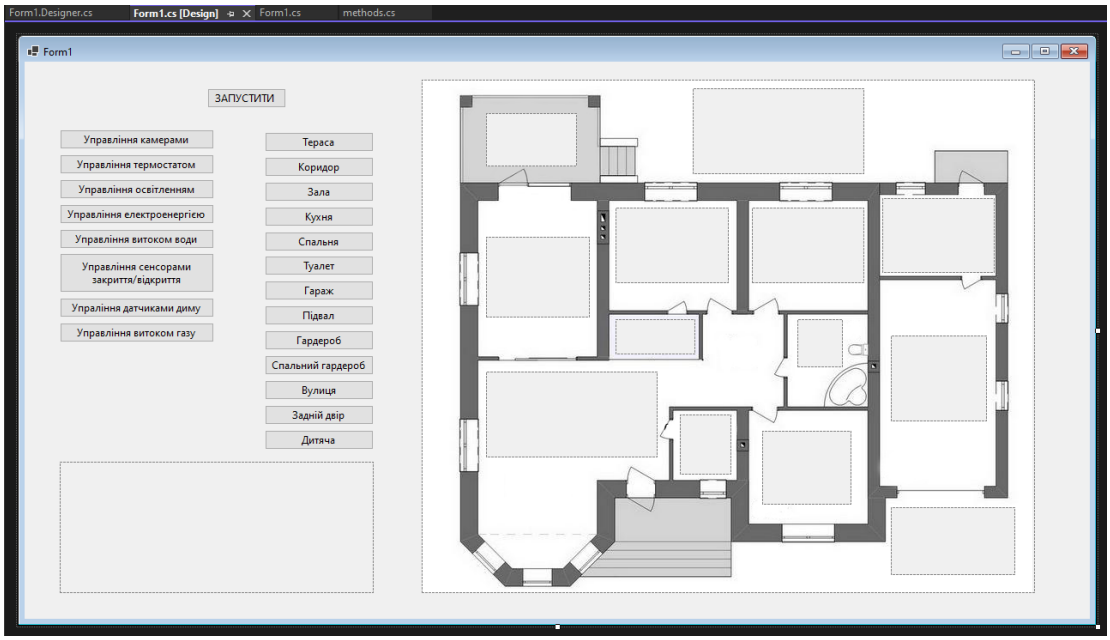


Рисунок 3.1 – Загальний вид інтерфейсу системи

Далі доречно запустити і протестувати інтерфейс користувача:

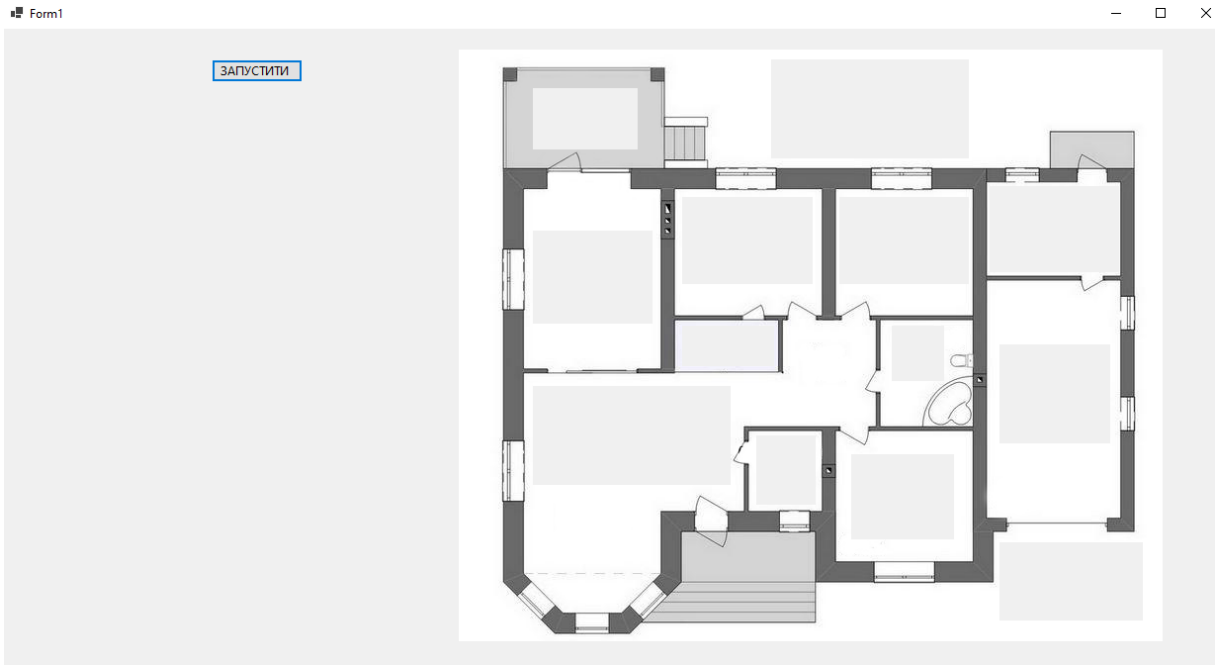


Рисунок 3.2 – Вид тільки запущено інтерфейсу системи

Далі користувач натискає на клавішу запуску системи, та обирає необхідну підсистему управління сенсорами розумного будинку:

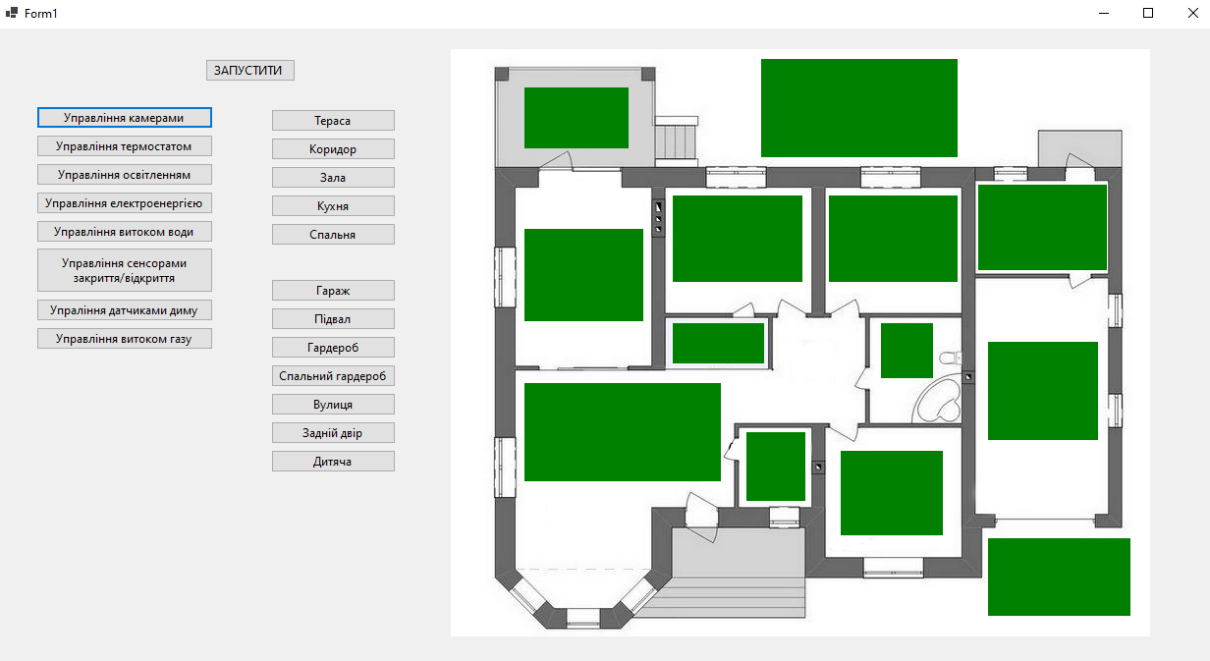


Рисунок 3.3 – Меню вибору потребуючої підсистеми з підсвіткою доступних сенсорів в кімнатах будинку

Наступним кроком стає вибір кімнати для подальшого налаштування. Кімната, обрана для налаштування має підсвітку фіолетовим кольором. Також є можливість вибору декількох кімнат для налаштування. Приклади зображені на рисунках 3.4 та 3.5.

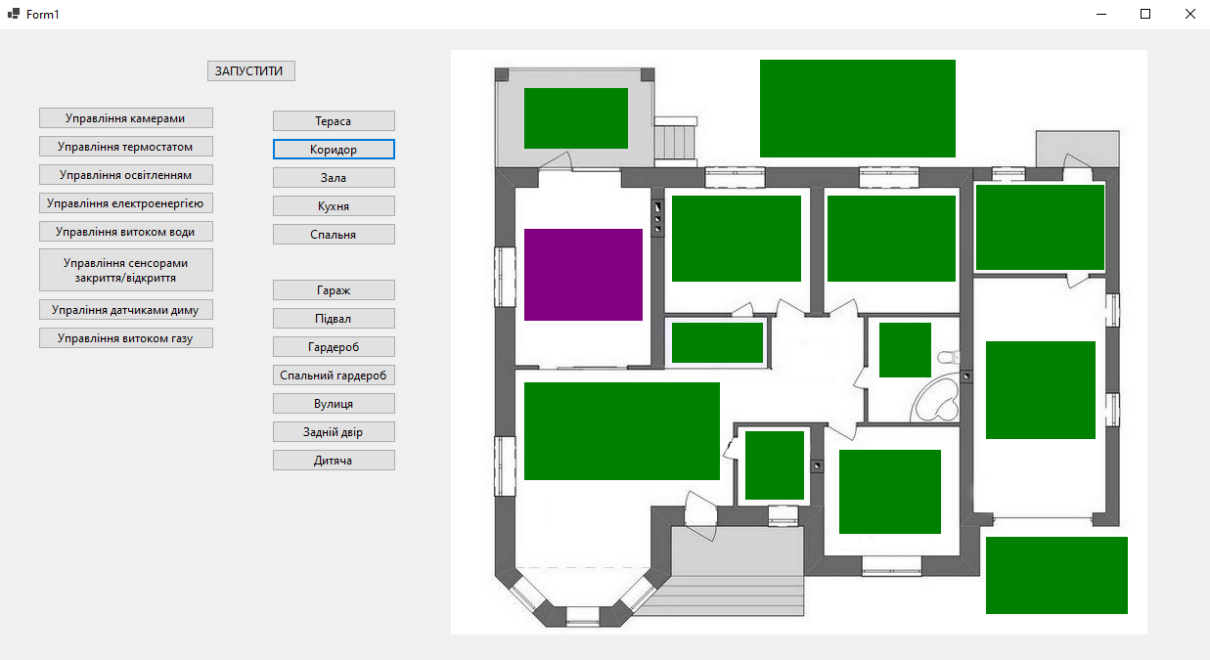


Рисунок 3.4 – Меню вибору потребуючої підсистеми в певній кімнаті з підсвіткою доступних налаштувань в будинку

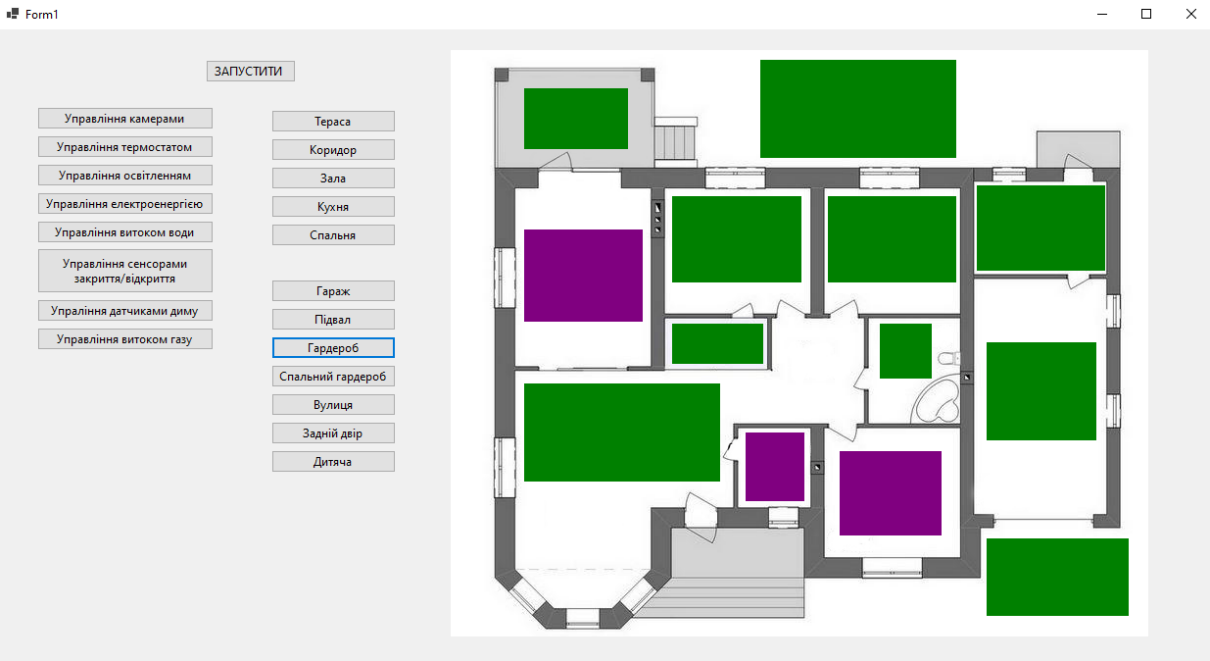


Рисунок 3.5 – Меню вибору потребуючої підсистеми в певних кімнатах з підсвіткою доступних налаштувань в будинку

Реалізація вибору і підсвітки створена за допомогою найпростіших методів розробки. Функції з методом Params продемонстровані нижче:

```
1 reference
public void chooseControlMethod(params Button[] Controller)
{
    for(int i = 0; i < Controller.Length; i++) {
        Controller[i].Visible = true;
    }
}
```

Рисунок 3.6 – Функція вибору підсистеми управління з подальшою підсвіткою

```
8 references
public void sensorUsage(params Button[] SensorArray)
{
    for (int i = 0; i < SensorArray.Length; i++)
    {
        SensorArray[i].Visible = true;
    }
}
```

Рисунок 3.7 – Функція вибору певних сенсорів з подальшою підсвіткою

```
8 references
public void highlighter(params Panel[] availableRoom)
{
    for (int i = 0; i < availableRoom.Length; i++)
    {
        availableRoom[i].BackColor = Color.Green;
    }
}
```

Рисунок 3.8 – Функція підсвічування на інтерфейсі

```
8 references
public void hideroom(params Button[] room)
{
    for (int i = 0; i < room.Length; i++)
    {
        room[i].Visible = false;
    }
}
```

Рисунок 3.9 – Функція схову підсвічування на інтерфейсі

```
13 references
public void chosenRoom(params Panel[] chosenRoom)
{
    for (int i = 0; i < chosenRoom.Length; i++)
    {
        chosenRoom[i].BackColor = Color.Purple;
    }
}
```

Рисунок 3.10 – Функція вибору певної кімнати для подальшого налаштування

3.3. Розробка програмної частини розробки підсистем управління розумного будинку

Аспект написання програмної частини, програмного забезпечення, є фундаментальним при створенні розумних систем, оскільки він відповідає за збір та обробку даних з датчиків, керування підключеними пристроями та забезпечення взаємодії з іншими підсистемами. Метою цього розділу є теоретичне описання процесу розробки програмної частини, включаючи архітектуру системи, вибір технологій та методів реалізації, а також

особливості інтеграції з існуючими системами. Але важливо розуміти, що робота написана в дослідницькому вигляді, та не має бюджету на покупку датчиків та їх повного програмування, тож подальший програмний код використовувати, як інструкцію для подальшого створення своїх систем.[1,2,4,5,6,7]

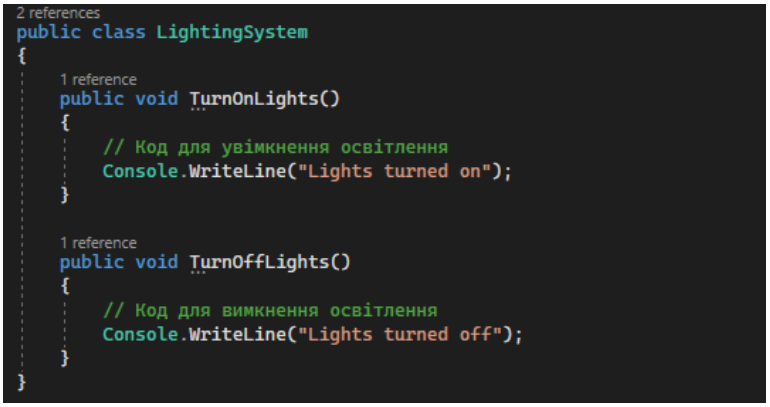


Рисунок 3.11 – Функціональна частина можливих налаштувань підсистем освітлення

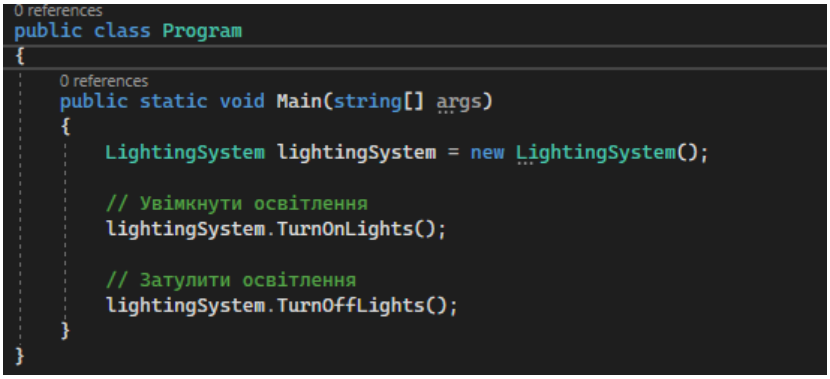


Рисунок 3.12 – Функціональна частина управління підсистем освітлення

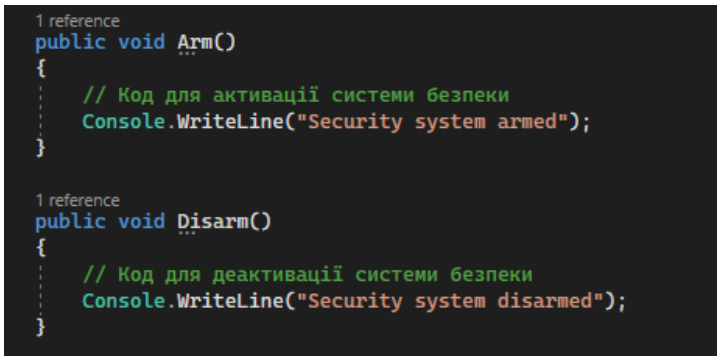


Рисунок 3.13 – Функціональна частина можливих налаштувань підсистем захисту

```
0 references
public static void Main(string[] args)
{
    SecuritySystem securitySystem = new SecuritySystem();

    // Активувати систему безпеки
    securitySystem.Arm();

    // Деактивувати систему безпеки
    securitySystem.Disarm();
}
```

Рисунок 3.14 – Функціональна частина управління підсистем захисту розумного дому

```
2 references
public class TemperatureControlSystem
{
    1 reference
    public void SetTemperature(int temperature)
    {
        // Код для встановлення заданої температури
        Console.WriteLine("Temperature set to: " + temperature + "°C");
    }
}
```

Рисунок 3.15 – Функціональна частина можливих налаштувань підсистем температурних налаштувань

```
0 references
public static void Main(string[] args)
{
    TemperatureControlSystem temperatureControlSystem = new TemperatureControlSystem();

    // Встановити температуру на 25°C
    temperatureControlSystem.SetTemperature(25);
}
```

Рисунок 3.16 – Функціональна частина управління підсистем температурних датчиків

3.4. Реалізація управляючих функцій

3.4.1. Реалізація комунікаційного протоколу

Для роботи встановлюю бібліотеку MQTT у C#. Одним з популярних варіантів є бібліотека M2Mqtt, яка надає можливість здійснювати з'єднання

з брокером MQTT, публікувати повідомлення та підписуватися на повідомлення, що надійшли від інших пристроїв.

Основні етапи для реалізації комунікаційного протоколу управління включають:

1. Підключення до брокера MQTT: Створено клієнт MQTT, використовуючи бібліотеку M2Mqtt, та підключаємось до брокера MQTT, вказавши адресу та порт брокера.

```
MqttClient client = new MqttClient("broker_address");
client.Connect("client_id");
```

Рисунок 3.17 – Підключення до брокеру

2. Публікація повідомлень: ВІдправляємо повідомлення до брокера MQTT, вказуючи тему (topic) та дані, які необхідно передати.

```
string topic = "home/bedroom/light";
string message = "on";
client.Publish(topic, Encoding.UTF8.GetBytes(message), MqttMsgBase.QOS_LEVEL_AT_LEAST_ONCE, false);
```

Рисунок N.N – Публікація повідомлення завдяки комунікаційному протоколу

3. Підписка на повідомлення: Підпишіться на теми (topics), щоб отримувати повідомлення від інших пристроїв. Використовуйте делегат для обробки отриманих повідомлень.

```
string[] topics = { "home/bedroom/light", "home/kitchen/temperature" };
byte[] qosLevels = { MqttMsgBase.QOS_LEVEL_AT_LEAST_ONCE, MqttMsgBase.QOS_LEVEL_AT_LEAST_ONCE };
client.Subscribe(topics, qosLevels);
client.MqttMsgPublishReceived += Client_MqttMsgPublishReceived;
```

Рисунок 3.18 – початкові дані для підписку на топіки

```
private static void Client_MqttMsgPublishReceived(object sender, MqttMsgPublishEventArgs e)
{
    // Обробка отриманого повідомлення
    string topic = e.Topic;
    string message = Encoding.UTF8.GetString(e.Message);

    // Виклик методу для обробки отриманого повідомлення
    ProcessReceivedMessage(topic, message);
}
```

Рисунок 3.19 – Метод обробки отриманих повідомлень

```
private static void ProcessReceivedMessage(string topic, string message)
{
    // Ваш код для обробки отриманого повідомлення
    // Наприклад, визначення пристрою та виклик відповідних дій у підсистемі "розумного" дому
    if (topic == "home/bedroom/light")
    {
        if (message == "on")
        {
            // Включити освітлення у спальні
        }
        else if (message == "off")
        {
            // Вимкнути освітлення у спальні
        }
    }
    else if (topic == "home/kitchen/temperature")
    {
        // Отримати дані про температуру у кухні та здійснити відповідні дії
    }
}
```

Рисунок 3.20 – процес реакції на отримане повідомлення

У цьому прикладі використовується бібліотека M2Mqtt для реалізації комунікаційного протоколу MQTT у мові C# з використанням програмного середовища Microsoft Visual Studio (MVS). Але важливо розуміти що цей приклад надає загальну структуру комунікаційного протоколу і обробки повідомлень. Користувач може вільно адаптувати його до конкретних потреб і вимог системи "розумного" дому.

3.4.2. Реалізація підключення до бази даних

Для підключення до бази даних у підсистемі "розумного" дому, реалізованій мовою C# через програмне середовище Microsoft Visual Studio (MVS), використовується технологія ADO.NET для взаємодії з реляційною базою даних, такою як Microsoft SQL Server, MySQL або PostgreSQL, тощо.

Основні кроки для підключення до бази даних у C#:

1. Створюється посилання на необхідну бібліотеку ADO.NET до проекту. Наприклад, для підключення до Microsoft SQL Server використовують бібліотеку “System.Data.SqlClient”.

2. Встановлюється рядок підключення (connection string), який містить інформацію про сервер бази даних, назву бази даних, облікові дані користувача та інші необхідні параметри.

```
string connectionString = "Server=server_name;Database=database_name;User Id=username;Password=password;";
```

Рисунок 3.21 – рядок підключення до бази даних

3. Створюється об'єкт підключення до бази даних, використовуючи рядок підключення:

```
SqlConnection connection = new SqlConnection();
```

Рисунок 3.22 – об’єкт підключення до бази даних

4. Відкривається з'єднання з базою даних:

```
connection.Open();
```

Рисунок 3.23 – Процес підключення до бази даних

5. Виконується запит до бази даних, наприклад, вибірку даних або вставку нового запису. Використовується об'єкт SqlCommand для виконання запитів

```
    string query = "SELECT * FROM table_name";
    SqlCommand command = new SqlCommand(query, connection);

    // Виконати запит і отримати результати
    SqlDataReader reader = command.ExecuteReader();
while (reader.Read())
{
    // Обробка отриманих даних
    string column1 = reader.GetString(0);
    int column2 = reader.GetInt32(1);
    // і так далі...
}

// Закрити читача даних після виконання запиту
reader.Close();
```

Рисунок 3.24 – Функція відсилання запиту в бд

6. Після завершення роботи з базою даних, з'єднання закривається:

```
connection.Close();
```

Рисунок 3.25 – Процес відключення від бази даних

7. Для виконання параметризованих запитів, де значення параметрів можуть змінюватися, використовуються параметри команди SqlCommand. Наприклад, для виконання вставки запису з параметрами:

```
string query = "INSERT INTO table_name (column1, column2) VALUES (@value1, @value2)";
SqlCommand command = new SqlCommand(query, connection);

// Додайте параметри до команди
command.Parameters.AddWithValue("@value1", value1);
command.Parameters.AddWithValue("@value2", value2);

// Виконати запит
int rowsAffected = command.ExecuteNonQuery();
```

Рисунок 3.26 – Параметризований запит

8. Якщо виокристовується ORM, таке як Entity Framework, створюються моделі даних (класи) для представлення таблиць бази даних та використовуються LINQ (Language-Integrated Query) для виконання запитів до бази даних. ORM дозволяє зробити роботу з базою даних більш зручною та абстрагує вас від написання SQL-запитів вручну.

9. Забезпечується правильне керування підключенням до бази даних, використовуючи конструкцію using або користувачу необхідно впевнитися, що вручну закрито підключення після використання.

```
using (SqlConnection connection = new SqlConnection(connectionString))
{
    connection.Open();

    // Виконайте запити до бази даних

}           // Після завершення блоку using підключення буде автоматично закриточ
```

Рисунок 3.27 – Параметризований запит

Це загальна структура підключення до бази даних у підсистемі "розумного" дому, реалізованій мовою C# через програмне середовище MVS. Залежно від конкретних потреб та вимог, можуть бути використані інші технології або підходи.

3.4.3. Реалізація підключення датчиків та акураторів

Підключення датчиків до підсистеми "розумного" дому, реалізованої мовою C# через програмне середовище Microsoft Visual Studio (MVS), варіюються залежно від типу датчиків та їх інтерфейсів. Нижче наведено загальний підхід для підключення датчиків за допомогою різних технологій:

1. Датчики, що використовують фізичний інтерфейс (наприклад, GPIO, I2C, SPI): Для підключення таких датчиків до підсистеми "розумного" дому, знадобиться апаратне забезпечення, яке підтримує вказаний інтерфейс, а також відповідна бібліотека для взаємодії з ним. Зазвичай виробники пристроїв надають документацію та бібліотеки для використання їхніх датчиків у вашому проекті. Наприклад, для підключення датчиків через GPIO до Raspberry Pi у мові C#, необхідно використовувати бібліотеку RaspberryIO, яка забезпечує можливість роботи з GPIO пінами.

```
    // Приклад роботи з GPIO пінами
    using System.Device.Gpio;

    GpioController gpio = new GpioController();
int pinNumber = 17; // Номер GPIO піна
gpio.OpenPin(pinNumber, PinMode.Input); // Налаштування піна як вхідного

    // Отримання стану піна
    PinValue pinValue = gpio.Read(pinNumber);
```

Рисунок 3.28 – Підключення датчика через фізичний інтерфейс

Для датчиків, що використовують інші інтерфейси, такі як I2C або SPI, знадобиться відповідна бібліотека для взаємодії з цими інтерфейсами, наприклад, бібліотека I2CDevLib для I2C.

2. Датчики, що використовують бездротові технології (наприклад, Wi-Fi, Bluetooth, Zigbee):

Для підключення датчиків, що використовують бездротові технології, до підсистеми "розумного" дому, знадобиться відповідна бібліотека або SDK для роботи з цими технологіями.

Наприклад, для бездротового підключення датчиків через Wi-Fi, ви можете використовувати бібліотеку .NET Core System.Net.Sockets, яка надає можливості для створення TCP або UDP з'єднань із сервером або пристроєм за його IP-адресою та портом.

```
// Приклад підключення до сервера за його IP-адресою та портом
string ipAddress = "192.168.0.100";
int port = 1234;

try
{
    // Створення TCP клієнта
    TcpClient client = new TcpClient();
client.Connect(IPAddress.Parse(ipAddress), port);

    // Відправка та отримання даних
    NetworkStream stream = client.GetStream();
// Отримання або відправка даних через stream

// Закриття з'єднання
stream.Close();
    client.Close();
}
catch (Exception ex)
{
Console.WriteLine("Помилка підключення: " + ex.Message);A
```

Рисунок 3.29 – Підключення датчика через безпровідний інтерфейс

Для бездротових технологій, таких як Bluetooth або Zigbee, можуть бути доступні спеціалізовані бібліотеки або SDK від виробників пристроїв. Рекомендується звернутися до документації та прикладів, наданих виробником датчика, який планується використовувати.

Залежно від типу бездротової технології та конкретного датчика, використовуються різні підходи до підключення та обробки даних з цих датчиків. Рекомендується вивчання документації, прикладівта ресурсів, пов'язаних з використанням конкретної бездротової технології та датчика, щоб зрозуміти, як правильно реалізувати підключення в певному випадку.

4. БІЗНЕС ПЛАН

4.1.Резюме

4.1.1. Впровадження продукту

Підсистема управління "розумним" домом впроваджуватиметься з метою поліпшення зручності, комфорту, безпеки та енергоефективності життя в домашньому середовищі. Основні аргументи впровадження продукту стають наступними:

1. Зручність і комфорт: Підсистема управління "розумним" домом дозволить власникам дому з легкістю контролювати різні аспекти свого домашнього середовища, такі як освітлення, опалення, вентиляція, кондиціювання повітря, аудіо/відео системи та інші підсистеми. Завдяки інтеграції різних "розумних" пристроїв та їх управлінню за допомогою однієї централізованої системи, користувачі зможуть налаштовувати режими роботи свого дому відповідно до своїх потреб, створювати розклади автоматичного включення/вимикання певних пристроїв, а також віддалено керувати ними з допомогою мобільного додатка або віддаленого доступу до веб-інтерфейсу.
2. Безпека: Підсистема управління "розумним" домом включає ряд заходів забезпечення безпеки, таких як системи відеоспостереження, виявлення вторгнень, датчики витоку води, вогню, газу, системи контролю доступу, автоматичні оповіщення про небезпеку тощо. Це дозволить власникам дому відслідковувати події, що відбуваються в їхньому будинку в режимі реального часу, а також приймати заходи для попередження можливих аварійних ситуацій.
3. Енергоефективність: Підсистема управління "розумним" домом допоможе в енергоефективному використанні різних ресурсів, таких як електроенергія, вода, газ тощо. Завдяки можливості програмування

розкладів роботи пристроїв, автоматичного вимикання непотрібних пристроїв відповідно до розкладу або присутності мешканців в будинку, а також контролю рівня енергоспоживання, користувачі зможуть раціональніше використовувати енергію та ресурси, що в результаті може призвести до зменшення витрат на комунальні послуги та зниження екологічного впливу.

4. Інтеграція та розширення можливостей: Підсистема управління "розумним" домом може бути розширена та інтегрована з різними "розумними" пристроями та системами, такими як розумні термостати, датчики руху, системи аудіо/відео розваг, системи автоматизованого поливу та інші. Це дозволить користувачам масштабувати та розширювати функціональні можливості своєї системи управління "розумним" домом залежно від їхніх потреб та вимог.

Таким чином, впровадження підсистеми управління "розумним" домом забезпечує зручність, комфорт, безпеку та енергоефективність в домашньому середовищі, а також можливість розширення та інтеграції з різними "розумними" пристроями та системами для забезпечення більш високого рівня автоматизації та керування різними аспектами домашнього життя.

4.1.2. Визначення визначних сторін продукту

Декілька визначних рис підсистеми управління "розумним" домом, які відрізняють її від продуктів конкурентів, включають:

1. Гнучкість та розширюваність: Підсистема управління "розумним" домом розроблена з урахуванням гнучкості та розширюваності, що дозволяє користувачам налаштовувати систему відповідно до їхніх вимог та потреб. Вона повинна підтримувати різні типи пристроїв, комунікаційні протоколи, та інтегруватися з різними системами, що дає користувачам більший вибір та можливість розширювати

функціональні можливості системи управління "розумним" домом в майбутньому.

2. Інтелектуальні аналітичні можливості: Підсистема управління "розумним" домом має розумні аналітичні можливості, такі як аналіз даних з різних датчиків, машинне навчання, аналіз споживання енергії та ресурсів, що дозволяє системі автоматично приймати рішення, прогнозувати поведінку системи та оптимізувати роботу пристроїв в реальному часі. Це забезпечує більш ефективне та розумне управління різними аспектами домашнього середовища.
3. Забезпечення безпеки та конфіденційності: Підсистема управління "розумним" домом повинна мати високі стандарти безпеки та захисту даних. Вона включає захист від несанкціонованого доступу до системи, шифрування даних, аутентифікацію користувачів та інші заходи безпеки, що забезпечують захист персональної інформації та даних користувачів, а також захист від потенційних кібератак.
4. Інтеграція зі смарт-екосистемами: Підсистема управління "розумним" домом здатна інтегруватися з різними смарт-екосистемами, такими як голосові асистенти (наприклад, Amazon Alexa, Google Assistant), смарт-девайси (наприклад, смарт-телевізори, смарт-розетки, смарт-термостати) та іншими системи "розумного" дому. Це може забезпечити зручність та легкість управління різними пристроями та системами в домашньому середовищі, дозволяючи користувачам керувати всіма аспектами "розумного" дому з одного інтерфейсу.
5. Інтуїтивний та зручний інтерфейс: Підсистема управління "розумним" домом має інтуїтивний та зручний інтерфейс користувача, який дозволяє легко налаштовувати та керувати різними пристроями та системами. Це може включати мобільні додатки, веб-інтерфейси, графічні інтерфейси та інші зручні способи взаємодії з системою.
6. Гнучкість та розширюваність: Підсистема управління "розумним" домом може бути гнучкою та розширюваною, дозволяючи додавати

нові функції, пристрої та системи в майбутньому. Це може включати можливість оновлень програмного забезпечення, додавання нових сенсорів, розширення підтримки протоколів зв'язку та інші можливості, що дозволяють адаптувати продукт до змінних потреб користувачів та ринкових вимог.

Ці визначні риси роблять продукт управління "розумним" домом відмінним від продуктів конкурентів, забезпечуючи вищий рівень зручності, функціональності, екологічної сталості, гнучкості та розширюваності, надійності, безпеки та сумісності з різними пристроями та системами. Такі визначні риси забезпечують конкурентну перевагу продукту на ринку розумних домів, дозволяючи користувачам отримати більш високу якість, зручність та ефективність в управлінні своїм домом, порівняно з аналогічними продуктами конкурентів.

4.1.3. Очікувані фінансові результати

Прогнозні об'єми продажу на найближчі роки були теоретично визначені на основі дослідження ринку, аналізу конкуренції, оцінки попиту та інших факторів. Наприклад, на перші роки може бути спрогнозовано помірний зріст продажів, з врахуванням введення продукту на ринок та його популяризації, а наступні роки можуть передбачатися стабільні обсяги продажу зі збільшенням наслідком зростання клієнтської бази та лояльності клієнтів.

Прибуток від продажів залежить від ціни продукту, витрат на виробництво, обсягу продажів та маржі прибутку, встановленої компанією. Наприклад, висока ціна продукту при відносно низьких витратах на виробництво та великому обсягу продажів може забезпечити високий прибуток від продажу.

Витрати на виробництво включають витрати на сировину, компоненти, працю, виробничі умови, технічне обладнання та інші

витрати, пов'язані з виробництвом продукту. Наприклад, витрати на сировину можуть бути залежні від її ціни на ринку, обсягу виробництва та рівня технологічності продукту.

Валовий прибуток може бути визначений як різниця між прибутком від продажу та витратами на виробництво. Рівень прибутковості вкладень в майбутню справу розрахований відношенням валового прибутку до вкладених коштів або капіталу. Наприклад, високий валовий прибуток та рівень прибутковості можуть бути привабливими для потенційних інвесторів.

Термін повернення банківського кредиту залежатиме від умов кредитного договору, розміру кредиту, процентної ставки, погашення кредиту та фінансової потужності компанії. Наприклад, більш короткий термін повернення може бути вигідним з точки зору витрат на відсотки, але може вимагати високих щомісячних платежів, тоді як довший термін повернення може забезпечити більш низькі щомісячні платежі, але загалом призвести до більших витрат на відсотки.

Однак важливо зазначити, що точні числові дані щодо прогнозних об'ємів продажу, прибутку, витрат на виробництво, валового прибутку та терміну повернення банківського кредиту варіюватимуться в залежності від багатьох факторів, таких як ринкові умови, конкуренція, рівень попиту, внутрішні фінансові ресурси компанії та інші фактори. Детальний аналіз і розрахунки можуть бути виконані на основі конкретної ситуації та бізнес-плану компанії.

4.2.Проектований продукт

4.2.1. Опис продукту проєкту

Для повної міри опису продукту, доцільно використати вісім наступних питань:

1. Які потреби повинен задовольнити продукт проекту;

Забезпечення зручного та ефективного управління освітленням, опаленням, безпекою, енергоефективністю, комфортом та життєвим циклом "розумного" дому, що забезпечить зручність, ефективність та комфорт користувачів.

2. Його особливості і відмінні риси, які дозволять віддати перевагу даному продукту проекту, а не продукту конкурентів;

Висока інтеграцію систем, використання інноваційних технологій, зручний інтерфейс, можливості розширення та адаптації до потреб користувачів, що робить його більш привабливим в порівнянні з продуктами конкурентів.

3. Наявність патентів або авторських свідоцтв на продукт проекту;

Немає.

4. Чи є наочне зображення продукту проекту (фотографії, малюнки);

Наявність фотографій, малюнків або візуалізацій продукту проекту, що допоможе візуалізувати його вигляд та функції.

5. Яка попередня оцінка реалізації ціни реалізації продукту проекту і витрат, на його виробництво;

Продукт проекту включає детальний розрахунок витрат на розробку, виробництво, тестування, маркетинг, логістику тощо, що допоможе визначити економічну доцільність проекту.

6. Очікувана величина прибутку, який приноситиме продукт проекту;

На основі бізнес-моделі та ринкових умов, що допоможе визначити потенційний дохід від реалізації продукту

7. Характеристика якісних показників продукту проекту, його переваги;

Високий рівень автоматизації та інтелектуалізації, можливість дистанційного управління та моніторингу, енергоефективність, надійність та безпеку системи, можливість інтеграції з різноманітними пристроями та

платформами "розумного" дому, що робить продукт привабливим на ринку.

8. Організація сервісу продукту проекту (якщо це технічні вироби);

Так як це технічний виріб, включає планування післяпродажного обслуговування, технічну підтримку, оновлення програмного забезпечення, можливість віддаленого діагностування та ремонту, надання гарантійного обслуговування, що забезпечує надійну роботу продукту та задоволення потреб користувачів.

Отже, продукт проекту "Розробка підсистеми управління "розумним" домом" задовільняє потреби у зручному та ефективному управлінні домашніми системами, має високий рівень інтеграції, інноваційні технології, можливості розширення та адаптації, можливість патентного захисту, наявність наочних зображень, оцінку вартості реалізації та витрат, очікувану величину прибутку, характеристики якісних показників, а також організацію сервісу для забезпечення успішної реалізації та задоволення потреб користувачів.

4.3.Оцінка ринку збуту

4.3.1. Умови постачання та понеціал продукту

Умови постачання, виробництва та збуту продукту проекту "Розробка підсистеми управління "розумним" домом" включають:

1.1. Постачання компонентів та матеріалів для виробництва продукту, забезпечення їх якості та вчасності поставок.

1.2. Виробництво продукту відповідно до встановлених технічних вимог, забезпечення високої якості та відповідності продукту вимогам ринку.

1.3. Збут продукту, включаючи розробку стратегії маркетингу, визначення каналів розповсюдження, просування продукту на

ринку, взаємодію зі спеціалізованими дистриб'юторами або роздрібними мережами, встановлення цін на продукт, управління логістикою та постачанням продукту.

Потенційні конкуренти проекту можуть бути інші компанії або розробники, що також пропонують рішення для "розумного" дому. Оцінка їх потенціалу може включати:

1.1. Продукт конкурентів: Які продукти вже існують на ринку "розумного" дому, які функції та можливості вони пропонують, яка їх якість та надійність, чим вони відрізняються від продукту проекту.
1.2. Якість продукту проекту: Які переваги та особливості має продукт проекту в порівнянні з конкурентами, чи він задовольняє потреби ринку, чи має відмінні характеристики, які можуть забезпечити йому конкурентну перевагу.
1.3. Попередні ціни на продукт проекту: Яка вартість продукту проекту в порівнянні з цінами конкурентів, чи він конкурентоспроможний з точки зору ціни, який рівень прибутку може бути досягнутий.

4.3.2. Аналіз даних

Аналіз даних, включає в себе багато аспектів, що включають: визначення потенційних споживачів продукту проекту для дослідження ринку, вивчення поведінки споживачів, їхніх потреб та вимог. Необхідність відповіді на питання, хто може бути зацікавлений у продукті, чому вони можуть бути зацікавлені, скільки продукт може коштувати для цих споживачів, та коли вони можуть бути готові купувати продукт, яка є найближча та подальша перспектива. Дослідження споживачів яке включає опитування, фокус-групи, аналіз соціальних медіа та взаємодії з

потенційними клієнтами, щоб зрозуміти їхні потреби, побажання та готовність купувати продукт.

Визначення зразкової реалізаційної ціни продукту проекту в умовах конкуренції, що вимагає аналізу ринку, вивчення цін на подібні продукти конкурентів та врахування витрат на виробництво, маржинальної прибутковості та стратегії ціноутворення. Дослідження цін конкурентів яке включає аналіз цін на аналогічні або подібні продукти на ринку, оцінку їхньої якості та функціональних характеристик, а також врахування рівня конкуренції в галузі "розумного" дому. Врахування витрат на виробництво, управління та маржинальної прибутковості допоможе визначити оптимальну ціну продукту, що забезпечить прибутковість проекту в умовах конкуренції. Врахування стратегії ціноутворення може бути також важливим аспектом, де ви можете визначити, чи будете ви встановлювати ціну продукту на рівень нижче, рівною або вищою за ціну конкурентів, чи ж обрати стратегію цінової конкуренції, встановивши ціну нижче за конкурентів з метою залучення більшої кількості клієнтів.

Тож для визначення зразкової реалізаційної ціни продукту проекту в умовах конкуренції можуть бути корисними наступні кроки:

- Дослідження ринку: Вивчення цін на подібні або конкуруючі продукти на ринку, аналіз їхніх особливостей, якості та цінової політики конкурентів.
- Аналіз витрат на виробництво: Оцінка витрат на виробництво продукту, включаючи сировинні матеріали, працю, енергію, виробничі витрати та інші витрати, пов'язані з виробництвом продукту.
- Врахування маржинальної прибутковості: Визначення необхідного рівня прибутку та маржинальної прибутковості продукту, яка забезпечить прибутковість проекту в умовах конкуренції.
- Врахування стратегії ціноутворення: Визначення стратегії ціноутворення, яка може бути високої ціни з високою якістю

продукту, середньої ціни з нормальною якістю, або низької ціни з меншою якістю. Вибір стратегії ціноутворення може впливати на визначення зразкової реалізаційної ціни.

- Врахування факторів попиту: Оцінка попиту на продукт проекту, включаючи оцінку готовності споживачів купувати продукт за певну ціну в найближчій та подальшій перспективі.

Після виконання досліджень та аналізу цих факторів, можна визначити зразкову реалізаційну ціну продукту проекту в умовах конкуренції. Зразкова реалізаційна ціна повинна враховувати витрати на виробництво, бажану прибутковість проекту, стратегію ціноутворення та фактори попиту на ринку.

Для встановлення ціни можна застосувати різні підходи, такі як:

- Витратний підхід: Додавання витрат на виробництво продукту до бажаної прибутковості проекту. Наприклад, якщо витрати на виробництво продукту складають $100,000, а бажана прибутковість - 20%, то зразкова реалізаційна ціна буде $100,000 + ($100,000 * 20%) = $120,000.
- Конкурентоспрямований підхід: Встановлення ціни на основі цін конкурентів. Наприклад, якщо середня ціна конкурентів на подібний продукт складає $150,000, то зразкова реалізаційна ціна може бути приблизно на рівні або трохи вище цієї ціни, залежно від бажаної прибутковості проекту.
- Попитоспрямований підхід: Встановлення ціни на основі оцінки готовності споживачів купувати продукт за певну ціну в найближчій та подальшій перспективі. Це може включати дослідження ринку, аналіз попиту, опитування споживачів тощо. Наприклад, якщо дослідження показує, що споживачі готові купувати продукт за ціною до $130,000, то зразкова реалізаційна ціна може бути в цьому діапазоні.

Важливо також враховувати стратегію ціноутворення, яку обрав проект, та його позицію на ринку. Наприклад, якщо проект має стратегію ціної лідера, то зразкова реалізаційна ціна може бути нижчою за ціну конкурентів, з метою привернення більшого попиту. З іншого боку, якщо проект має стратегію ціної диференціації, то зразкова реалізаційна ціна може бути вищою за ціну конкурентів, з метою позиціонування продукту як преміум-варіанту.

Важливо також враховувати фактори внутрішнього та зовнішнього середовища, такі як витрати на виробництво, конкуренція на ринку, попит споживачів, наявність аналогічних продуктів та їхні ціни, економічні тенденції та інші фактори, які можуть впливати на встановлення зразкової реалізаційної ціни.

У підсумку, визначення зразкової реалізаційної ціни продукту проекту в умовах конкуренції вимагає ретельного аналізу витрат на виробництво, бажаної прибутковості, стратегії ціноутворення, факторів попиту та конкурентних умов ринку. Ретельне дослідження ринку, аналіз конкурентів та опитування споживачів можуть бути корисними інструментами при встановленні зразкової реалізаційної ціни продукту проекту.

4.4.Конкуренція

4.4.1. Найбільші виробники аналогічного продукту

На 2023 рік немає одного єдиного найбільшого виробника аналогічного продукту у темі "Розробка підсистеми управління 'розумним' домом", оскільки ця галузь розвивається досить швидко, і на ринку присутні багато компаній, як великих, так і менших, що займаються розробкою технологій "розумного" дому.

Деякі з відомих компаній, які можуть бути виробниками аналогічних продуктів у цій галузі на 2023 рік, включають:

- Google (з системою Google Home та платформою Google Assistant).
- Amazon (з продуктами Amazon Echo та платформою Amazon Alexa).
- Apple (з системою Apple HomeKit та голосовим помічником Siri).
- Samsung (з платформою SmartThings).
- Philips (з системою Philips Hue для "розумного" освітлення).
- Nest Labs (підрозділ Google, що розробляє "розумні" термостати та системи безпеки).
- Xiaomi (китайський виробник "розумної" техніки для дому).
- Honeywell (з системами управління освітленням, температурою та безпекою).

LG (з "розумними" пристроями для дому, такими як пральні машини, холодильники, телевізори та ін.).

Варто зауважити, що цей список не є вичерпним, оскільки на ринку присутні багато інших компаній, які також активно розробляють продукти для "розумного" дому. Ринок "розумних" домівок є динамічним і продовжує зростати, тому можуть бути інші виробники, які також мають значну частку на ринку у 2023році.

4.4.2. Сутність продукту конкуренту

Продукт конкурентів є комплексом характеристик, які включають основні технічні та функціональні параметри, які відрізняють його від інших продуктів на ринку. Основні характеристики продукту конкурентів можуть включати такі елементи:

- Основні технічні та функціональні параметри: Це можуть бути такі параметри, як розміри, вага, продуктивність, споживання енергії, можливості підключення до мережі, робота з певними операційними

системами або платформами, наявність додаткових функцій та можливостей.

- Рівень якості: Це може включати використання високоякісних матеріалів, високотехнологічне виробництво, наявність сертифікацій та стандартів якості, наявність гарантій та підтримки після продажу, оцінки якості від споживачів та незалежних організацій.
- Дизайн: Це може включати зовнішній вигляд продукту, його ергономіку, функціональний дизайн, використання кольорів, форм та інших елементів дизайну, які можуть впливати на сприйняття продукту споживачами.
- Думка зацікавлених сторін: Це може включати думку клієнтів, партнерів, експертів галузі, аналітиків, журналістів та інших зацікавлених сторін. Вони можуть мати різні думки про продукт конкурентів, включаючи його переваги, недоліки, можливості та загрози на ринку.

Зазначена інформація варіюється в залежності від конкретної галузі та продукту, а також від оцінки різних джерел. Рекомендується проводитти детальне дослідження конкурентів на основі різних джерел, включаючи офіційні дані виробників, відгуки споживачів, аналітичні звіти, статистику ринку та інші доступні джерела, що наразі неможливо. Це допоможе отримати більш повну інформацію про продукти конкурентів, їх характеристики, якість, дизайн та думку зацікавлених сторін, що може бути важливим при розробці власного продукту та стратегії маркетингу.

4.5.Стратегія маркетингу

4.5.1. Схема розповсюдження товарів

Найвигідніші шляхи розповсюдження даної техніки:

- Через власні фірмові магазини:

Продукти будуть розповсюджені через фірмові магазини, які знаходяться на різних ринках або в різних місцях. Це можуть бути власні роздрібні магазини, бутіки або фірмові сторінки в онлайн-магазинах.

- Через оптові організації:

Продукти будуть розповсюджені через оптові організації, які забезпечують постачання продуктів іншим оптовим та роздрібним покупцям. Це можуть бути дистриб'ютори, оптові постачальники, агенти, або інші проміжні посередники, які займаються оптовим продажем продуктів.

Схема розповсюдження товарів є абсолютно різною для різних компаній залежно від їхніх бізнес-моделей, цілей та ринкових стратегій. Деякі компанії можуть використовувати комбінацію власних фірмових магазинів та оптових організацій, в той час як інші можуть спрямовуватись виключно на одну з цих стратегій.

4.5.2. Ціноутворення та реклама

Методика визначення цін на товари:

1. Маркетингові дослідження: Дослідження ринку та аналіз конкурентів можуть визначити оптимальний рівень цін на товари.
2. Витрати на виробництво: Включення витрат на сировину, працю, виробництво, упаковку, логістику та інші фактори можуть вплинути на визначення цін.
3. Маржинальний аналіз: Врахування маржі на кожен товар або послугу може бути використано для визначення оптимальних цін.
4. Очікуваний рівень рентабельності на вкладені засоби: Це може бути різний для різних компаній, але взагалі розраховується як відношення прибутку до вкладених засобів. Очікуваний рівень рентабельності може залежати від багатьох факторів, таких як галузь, ринкові умови, конкуренція та стратегії компанії.

5. Методи організації реклами:
   5.1. Реклама в ЗМІ: Телебачення, радіо, газети, журнали та інші ЗМІ можуть використовуватися для розміщення рекламних оголошень.
   5.2. Інтернет-реклама: Реклама в Інтернеті може включати контекстну рекламу, банери, соціальні медіа, електронну пошту та інші цифрові канали.
   5.3. Пряма реклама: Організація спеціальних заходів, презентацій, демонстраційних заходів та інших акцій для залучення уваги споживачів.
6. Величина засобів, необхідних для реалізації реклами:
   6.1. Величина засобів, необхідних для реалізації реклами, варіюється в залежності від обсягу рекламної кампані, виду рекламної діяльності та маркетингових цілей компанії. Витрати на рекламу включають витрати на розробку рекламного матеріалу, купівлю рекламного майданчика, оплату рекламних послуг, організацію рекламних заходів та інші витрати, пов'язані з просуванням товарів чи послуг.
   6.2. Розмір бюджету на рекламу визначений на основі багатьох факторів, включаючи фінансові можливості компанії, розмір ринку, конкуренція, маркетингові цілі та стратегії компанії. Оптимальний розмір бюджету на рекламу може бути визначений шляхом аналізу ефективності рекламних кампаній та їх впливу на продажі та прибуток компанії.

Важливо враховувати, що реклама повинна бути ретельно спланована, має відповідати цілям компанії та бути націленою на цільову аудиторію. Ефективна рекламна кампанія може сприяти підвищенню свідомості про бренд, залученню нових клієнтів та підвищенню продажів, але вимагає витрат на ресурси та управління відповідними бюджетами.

4.6.Організаційний план

4.6.1. Кваліфікаційні вимоги

Для успішного ведення справ в галузі розробки підсистеми управління "розумним" домом знадобляться фахівці з наступних профілів:

1. Інженер з електроніки або електротехніки: для розробки електронних компонентів, включаючи схеми, плати, сенсори та інші електронні пристрої.
2. Програміст з вбудованими системами: для розробки програмного забезпечення, що працює на вбудованих системах, таких як мікроконтролери, процесори та інші.
3. Програміст зі штучним інтелектом та аналітик даних: для розробки алгоритмів машинного навчання та аналізу даних з сенсорів "розумного" дому.
4. Інженер зі зв'язку: для розробки систем зв'язку між різними компонентами "розумного" дому, такими як бездротові мережі, мережі передачі даних, Інтернет речей (IoT) тощо.
5. Фахівець з тестування та якості: для виконання вимог контролю якості продукції на різних етапах виробництва.

Заробітна плата фахівців буде варіюватися в залежності від рівня освіти, досвіду, компетенцій та ринкових умов. У галузі розробки "розумних" домів зазвичай заробітна плата фахівців є конкурентоспроможною та залежить від рівня відповідності профілю, освіти та досвіду працівника вимогам компанії.

Умови прийому на роботу можуть різнитися залежно від внутрішніх політик підприємства. Це може бути як постійна робота, так і співпраця з зовнішніми експертами на підставі окремих контрактів, угод або проектних замовлень. Залежно від потреб і ресурсів підприємства, можуть

бути різні форми працевлаштування, такі як повна зайнятість, неповна зайнятість, фріланс або контрактна робота.

До того ж, можливо скористатися послугами організацій з найму фахівців в галузі розробки "розумних" домів. Це можуть бути рекрутингові агенції, HR-консалтингові компанії, фахові онлайн-платформи та інші агенції, які спеціалізуються на рекрутингу IT-фахівців.

Щодо біографічних даних співробітників, це може бути наступна інформація:

1. Кваліфікація: освіта, наукові ступені, сертифікації, професійні кваліфікації, вміння та навички, що відповідають вимогам роботодавця.
2. Досвід роботи: перелік попередніх робочих місць, тривалість роботи на кожному місці, посади, обов'язки, досягнення, проекти, на яких працював співробітник.
3. Корисність для підприємства: опис внеску співробітника в діяльність підприємства, досягнення, впроваджені проекти, вміння вирішувати проблеми, комунікативні навички, здатність працювати в команді та інші фактори, які можуть бути корисні для успішного ведення справ.

Важливо забезпечити конфіденційність даних співробітників та дотримуватися вимог законодавства щодо захисту персональних даних.

4.7.Юридичний план

Юридичний план теми "Розробка підсистеми управління "розумним" домом" в контексті приватної власності може включати наступні аспекти:

1. Реєстрація та охорона інтелектуальної власності: розробка підсистеми може включати розробку програмного забезпечення, апаратного забезпечення, алгоритмів та інших інноваційних рішень, які можуть бути об'єктом інтелектуальної власності, такої як патенти,

авторські права, торгові марки тощо. Юридична охорона інтелектуальної власності може бути важливим аспектом захисту інвестицій підприємства.

2. Укладення контрактів з розробниками та співпраця з партнерами: розробка підсистеми може включати співпрацю з розробниками програмного забезпечення, постачальниками апаратного забезпечення, технічними експертами та іншими сторонами. Укладення контрактів, які визначають права та обов'язки сторін, може бути важливим аспектом забезпечення юридичної стабільності та захисту інтересів підприємства.

3. Забезпечення дотримання законодавства: розробка підсистеми повинна відповідати вимогам законодавства, таким як закони про захист даних, інтелектуальну власність, безпеку та стандарти якості. Забезпечення відповідності законодавчим вимогам може бути важливим аспектом уникнення юридичних ризиків та забезпечення репутації підприємства.

4. Регулювання відносин з клієнтами: розробка підсистеми може включати взаємодію з клієнтами, управанням їх даними, доступом до системи, збереженням приватності та іншими аспектами взаємодії. Регулювання відносин з клієнтами може включати укладення угод, політик конфіденційності, управління ризиками та захист прав споживачів.

5. Податкове планування: розробка підсистеми може впливати на податкові обов'язки підприємства, такі як оподаткування прибутку, ПДВ та інших податків. Податкове планування може включати оцінку податкових наслідків розробки, оптимізацію податкових обов'язків та взаємодію з податковими органами.

6. Ліцензування та розповсюдження підсистеми: підприємство може розглядати варіанти ліцензування та розповсюдження розробленої підсистеми на ринку. Це може включати розробку ліцензійних угод, управління авторськими правами, контроль за використанням підсистеми та отриманням відповідних дозволів.

7. Захист прав власності: як приватна власність, підсистема управління "розумним" домом може включати захист прав власності, таких як права

власності на програмне забезпечення, апаратне забезпечення, технічні рішення та інші інновації. Забезпечення захисту прав власності може включати вжиття відповідних юридичних заходів, таких як патентування, реєстрація торгових марок, захист авторських прав тощо.

Це загальний огляд юридичного плану теми "Розробка підсистеми управління "розумним" домом" в контексті приватної власності. З огзазначеного можна вважати, що розробка підсистеми управління "розумним" домом як приватної власності також може включати додаткові юридичні аспекти, такі як розгляд варіантів фінансування розробки, врегулювання відносин між розробниками, розробниками та клієнтами, а також взаємодії з регуляторними органами, стандартизацією технічних рішень, вирішенням спорів та конфліктів.

Додатково, в контексті приватної власності можуть виникати питання щодо відповідальності за використання підсистеми, включаючи можливі ризики щодо безпеки, конфіденційності та захисту даних користувачів. Додатково, можуть виникати вимоги до забезпечення безпеки даних, включаючи відповідність законодавству про захист персональних даних та інших відповідних нормативних вимог.

У загальному, юридичний план розробки підсистеми управління "розумним" домом в контексті приватної власності включає ряд важливих аспектів, таких як укладення угод, врегулювання відносин з клієнтами, податкове планування, ліцензування та розповсюдження підсистеми.

4.8.**Оцінка ризиків і страхування**

4.8.1. Типи ризиків

Розробка підсистеми управління "розумним" домом в контексті приватної власності може відноситися до різних типів ризиків. Деякі з можливих ризиків, пов'язаних з даною темою, включають:

1. Технічні ризики: це можуть бути пов'язані з технічними аспектами розробки підсистеми, такими як проблеми з апаратним забезпеченням, програмним забезпеченням, комунікаційними мережами, безпекою та захистом даних, інтеграцією з іншими системами тощо. Неправильне функціонування технічних компонентів може призвести до зниження ефективності та ефективності підсистеми.

2. Безпекові ризики: розумні домові системи можуть бути піддаються різного роду кібератакам, злому, витокам даних, включаючи персональні дані користувачів, та іншим видам кіберзагроз. Недостатні заходи безпеки можуть призвести до порушення конфіденційності, цілісності та доступності даних, а також вплинути на довіру користувачів до системи.

3. Юридичні ризи: розробка підсистеми управління "розумним" домом може відноситися до різних юридичних ризиків, таких як порушення авторських прав, патентних прав, торговельних марок, невідповідність законодавству про захист персональних даних, регуляторних вимог та стандартів безпеки, неправильне укладення угод з розробниками, клієнтами та іншими сторонами.

4. Економічні ризи: розробка підсистеми управління "розумним" домом може пов'язуватися з економічними ризиками, такими як непередбачувані витрати на дослідження та розробку, високі витрати на впровадження та інтеграцію системи в реальному середовищі, незавершеність ринку "розумних" домових систем, непередбачувані зміни в економічному середовищі та конкуренція на ринку.

5. Операційні ризи: включають ризики пов'язані з ефективним функціонуванням розробленої підсистеми управління "розумним" домом, такі як збої в роботі системи, несправності технічних компонентів, невідповідність режиму експлуатації вимогам та очікуванням користувачів.

6. Ризики стосовно користувачів: пов'язані зі здатністю користувачів належним чином використовувати та налаштовувати систему, розумінням

функціональності та можливостей системи, ризиками збитку або травм, що можуть виникнути в результаті неправильного використання системи або некоректної реакції на автоматичні команди.

7. Ризики стосовно змін у вимогах ринку та технологій: зміни в ринкових вимогах, технологічному розвитку, зміні попиту користувачів та конкуренції можуть вплинути на успішність та прибутковість розробленої підсистеми управління "розумним" домом.

Це лише кілька можливих типів ризиків, які можуть відноситися до розробки підсистеми управління "розумним" домом в контексті приватної власності. Конкретні ризики можуть варіюватися в залежності від конкретних умов та контексту проекту. Ретельний аналіз ризиків та впровадження відповідних стратегій управління

4.9.**Стратегія фінансування**

4.9.1. План отримання засобів для створення або розширення підриємства

Кількість засобів, необхідних для реалізації проекту "Розробка підсистеми управління 'розумним' домом", залежить від ряду факторів, таких як обсяг та складність проекту, рівень технологічної розвиненості, доступні ресурси та бюджет проекту.

Основні категорії засобів, які можуть бути потрібні для реалізації такого проекту, включають:

1. Людські ресурси: це можуть бути розробники, інженери, дизайнери, тестувальники, менеджери проекту та інші спеціалісти, які беруть участь у розробці, тестуванні та впровадженні системи управління "розумним" домом.
2. Технічні засоби: це можуть бути сервери, комп'ютери, периферійні пристрої, мережеве обладнання, програмне забезпечення, датчики,

актуатори та інші технічні засоби, необхідні для розробки та впровадження підсистеми управління "розумним" домом.

3. Матеріали та компоненти: це можуть бути різноманітні електронні компоненти, кабелі, датчики, актуатори, корпуси, джерела живлення та інші матеріали, необхідні для збирання та встановлення підсистеми управління "розумним" домом.
4. Фінансові ресурси: це може бути бюджет проекту, який включає кошти на оплату заробітної плати спеціалістів, закупівлю технічних засобів та матеріалів, проведення досліджень та розробок, впровадження та випробування підсистеми управління "розумним" домом, рекламу та маркетинг, а також резервні фонди на непередбачуванівитрати та ризики.

Загальна кількість засобів, необхідних для реалізації проекту "Розробка підсистеми управління 'розумним' домом", буде залежати від ряду факторів, таких як розмір проекту, його складність, технічні вимоги, рівень автоматизації, рівень інтеграції з існуючими системами та інше. Детальну оцінку необхідних засобів можна визначити шляхом проведення детального аналізу вимог проекту, розробки плану реалізації, розрахунку вартості ресурсів та оцінки бюджету проекту.

Необхідність розробки "розумного" дому може включати розробку апаратної та програмної інфраструктури, розробку спеціального програмного забезпечення, налаштування датчиків, актуаторів, мережевого обладнання та інших компонентів, тестування та валідацію системи, а також впровадження та підтримку розробленої системи. Залежно від розміру та складності проекту, можуть знадобитися різні види спеціалістів, таких як програмісти, електронні інженери, інженери з мережевого обладнання, тестувальники, проектні менеджери та інші.

Для реалізації проекту також можуть знадобитися матеріали та компоненти, такі як датчики, актуатори, електронні компоненти, кабелі,

корпуси та інше. Вартість цих матеріалів також варіюватиме в залежності від їх типу, кількості та якості.

Крім того, фінансові ресурси, такі як бюджет проекту, також є важливим фактором. Бюджет проекту повинен враховувати витрати на заробітну плату спеціалістів, вартість матеріалів та компонентів, витрати на рекламу, маркетинг, підтримку та обслуговування після впровадження системи, а також можливі резервні фонди на випадок непередбачених ситуацій.

Ризики, пов'язані з реалізацією проекту, можуть включати зміну вимог замовника, затримки в розробці, проблеми з інтеграцією з існуючими системами, технічні проблеми, зміни на ринку, конкуренцію, фінансові ризики та інші. Врахування ризиків та розробка планів їх управління також є важливою складовою частиною оцінки необхідних засобів для реалізації проекту.

Загалом, оцінка необхідних засобів для реалізації проекту "Розробка підсистеми управління 'розумним' домом" вимагає детального аналізу вимог проекту, розробки плану реалізації, оцінки вартості ресурсів та ризиків, а також врахування бюджету проекту. Рекомендується взяти до уваги досвід фахівців з відповідними компетенціями та врахувати можливі зміни в процесі реалізації проекту.

ВИСНОВОК

У результаті виконання даного дипломного проекту розроблена найпростіша теоретична підсистема управління систем розумного будинку. Для досягнення цієї мети було проведено аналіз та дослідження, проектування задачі, програмування програмного продукту та розробку бізнес-плану. У кожному з розділів були виконані конкретні завдання, які розглянуті далі.

Аналіз та дослідження. У розділі проведено широке дослідження ринку систем розумного будинку, а також аналіз основних принципів, технологій та стандартів, які використовуються у цій галузі. Проведений огляд існуючих рішень та їх переваг та недоліків. Крім того, визначені потреби та вимоги до розроблюваної підсистеми, що послужило основою для подальшого проектування.

Проектування задачі. У розділі розроблено архітектуру системи розумного будинку, визначено функціональні та нефункціональні вимоги до підсистеми управління. Проведені детальні дослідження та проектування інтерфейсу користувача, а також розроблено моделі бази даних для зберігання інформації про будинок, пристрої та їх стан.

Програмування програмного продукту. У розділі "Програмування програмного продукту" було реалізовано підсистему управління систем розумного будинку. Використано сучасні програмні інструменти та технології для розробки функціональних модулів та їх інтеграції. Реалізована система забезпечує взаємодію з різними пристроями управління, включаючи освітлення, опалення, системи безпеки та інші. Під час програмування було звернуто особливу увагу на забезпечення надійності, безпеки та ефективності роботи системи. Проведені тестування та валідація розробленого програмного продукту з метою перевірки його функціональності та відповідності вимогам, встановленим на етапі проектування.

Бізнес-план. Останній розділ дипломного проекту присвячений розробці бізнес-плану. В ньому проведено аналіз ринку систем розумного будинку та визначено потенційні цільові аудиторії та їх потреби. Враховані фінансові аспекти, включаючи розрахунок вартості розробки, впровадження та підтримки системи. Також проведена стратегічна оцінка конкурентного середовища та розроблені маркетингові стратегії для просування продукту на ринку.

Загалом, цей дипломний проект зосереджувався на розробці підсистеми управління систем розумного будинку. Результатом роботи є функціональний програмний продукт, який задовольняє вимоги та потреби цільової аудиторії. Крім того, розроблений бізнес-план, що дає змогу оцінити перспективи комерціалізації розробленої системи. Даний проект може стати основою для подальшого розвитку та вдосконалення систем розумного будинку, сприяючи зручності, безпеці та енергоефективності життя в будинках.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Андрущак, І. І., & Бабій, І. В. (2018). Розробка підсистеми управління розумним будинком на основі інтернету речей. Збірник наукових праць Кам'янець-Подільського національного університету імені Івана Огієнка, 29, 5-10.
2. Войт, С. В., & Коваленко, О. О. (2019). Розробка підсистеми керування розумним будинком на базі мікроконтролера Arduino. Електротехніка і електромеханіка, 3(93), 84-90.
3. Деркач, М. В., & Максимов, В. О. (2017). Розробка підсистеми управління розумним будинком на основі контролера Arduino. Вісник Харківського національного університету Повітряних Сил імені Івана Кожедуба. Серія: Радіоелектроніка та інформатика, 3(51), 51-54.
4. Катич, О. В., & Кривцов, О. А. (2019). Розробка підсистеми управління розумним будинком на основі мікроконтролера NodeMCU. Збірник наукових праць Харківського національного університету радіоелектроніки. Серія: Технічна фізика та приладобудування, 27, 33-38.
5. Коваленко, Ю. В., & Куценко, Ю. О. (2017). Розробка підсистеми управління розумним будинком на основі контролера Arduino та платформи Android. Вісник Київського національного університету технологій та дизайну, 2(104), 104-110.
6. Корнієнко, С. М., & Булічев, А. С. (2018). Розробка підсистеми управління розумним будинком на основі платформи Raspberry Pi. Вісник Національного технічного університету "Харківський політехнічний інститут". Серія: Нові рішення в сучасних технологіях, 2(28), 67-73.
7. Марченко, О. В., & Полюга, А. А. (2020). Розробка підсистеми управління розумним будинком на основі платформи Arduino. Вісник

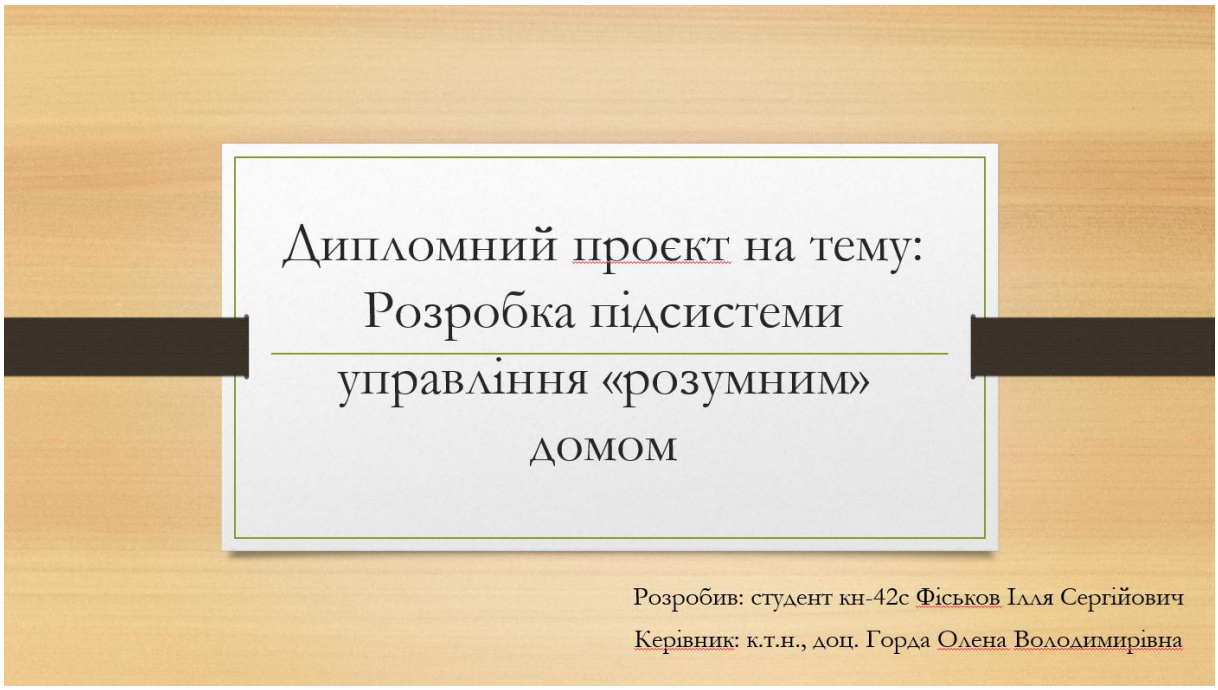
Харківського національного університету радіоелектроніки. Серія: Радіотехніка і телекомунікації, 190, 150-155.

8. Павленко, І. В., & Марченко, В. О. (2019). Розробка підсистеми управління розумним будинком на основі платформи Arduino. Вісник НТУ "ХПІ". Серія: Нові рішення в сучасних технологіях, 1(33), 105-112.
9. Смаглій, О. В., & Литвинов, О. О. (2018). Розробка підсистеми управління розумним будинком на основі мікроконтролера NodeMCU. Вісник Харківського національного університету радіоелектроніки. Серія: Технічна фізика та приладобудування, 27, 28-32.
10. Угриновський, С. В., & Самойлов, Ю. Ю. (2017). Розробка підсистеми управління розумним будинком на основі платформи Raspberry Pi. Електротехніка і електромеханіка, 3(89), 45-51.
11. Adamczyk, J., Chodarev, V., & Snasel, V. (Eds.). (2018). Intelligent Systems in Production Engineering and Maintenance—ISPEM 2017. Springer.
12. Fadlullah, Z. M., & Kato, N. (2016). IoT-based smart home: Architecture, potential applications, and challenges. IEEE Network, 30(2), 84-90.
13. Hancke, G. P., De Groot, G., & Silva, B. J. (2012). The role of wireless sensor networks in smart grids. In Smart Grid Communication Infrastructures (pp. 53-74). John Wiley & Sons.
14. Huang, X., Wang, X., & Yu, J. (2017). Intelligent home system based on internet of things. In Proceedings of the 3rd International Conference on Multimedia Systems and Signal Processing (pp. 312-317).
15. Lee, J., & Kim, E. (2015). Smart home energy management system using IEEE 802.15. 4 and ZigBee. IEEE Transactions on Consumer Electronics, 61(4), 480-488.
16. Liu, D., & Keally, M. (2019). Intelligent home control system based on the Internet of Things. In Proceedings of the 2019 2nd International Conference on Computer Communication and the Internet (pp. 214-217).

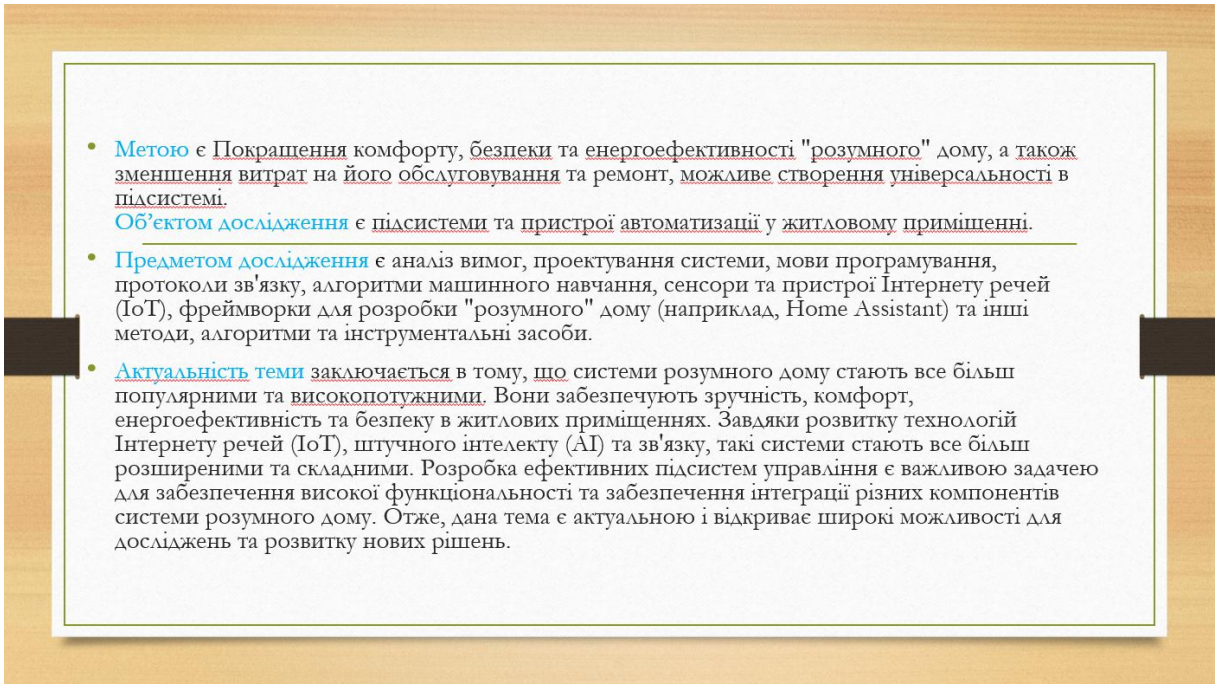
17. Mohan, N., Sahu, S. K., & Kumar, A. (2020). Design and implementation of smart home automation system using IoT. In Proceedings of the International Conference on Smart Intelligent Computing and Applications (pp. 353-361).
18. Ravelo-Medina, T. R., Jiménez-García, A. L., & Mendoza-Ramírez, A. (2020). Smart home control system using IoT and machine learning. Journal of Sensor and Actuator Networks, 9(3), 39.
19. Sim, K. M., & Jeon, Y. B. (2016). Intelligent home energy management system based on ZigBee and cloud computing. Journal of Electrical Engineering and Technology, 11(1), 1-9.
20. Wang, C., Zhang, Y., & Li, W. (2019). Design and implementation of a smart home control system based on ZigBee technology. In Proceedings of the 2019 4th International Conference on Automation, Control and Robots (pp. 360-364).

Додаток А

Презентація Атестаційної випускної роботи



Слайд 1 – Розробка підсистеми управління «розумним» домом

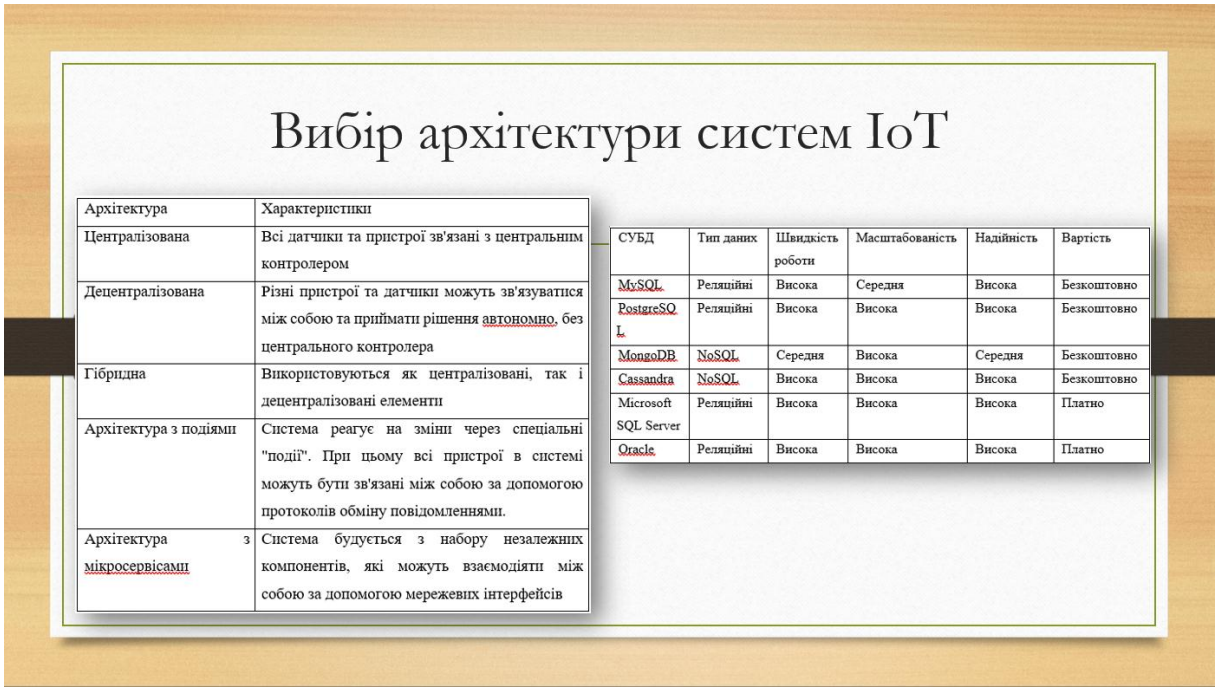


Слайд 2 – Мета, об'єкт, предмет та актуальність дослідження

Слайд 3 – Дерево цілей

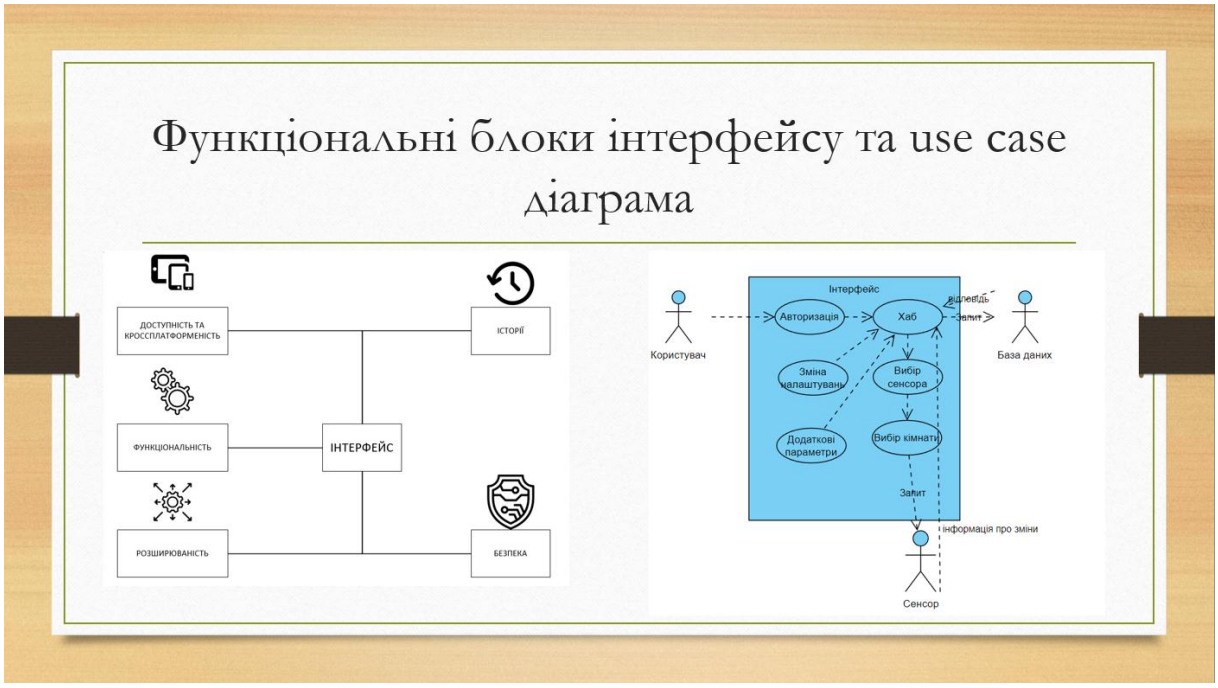
Слайд 4 – Аналіз типів підсистем управління

Слайд 5 – Постановка задачі



Слайд 6 – Вибір архітектури систем IoT

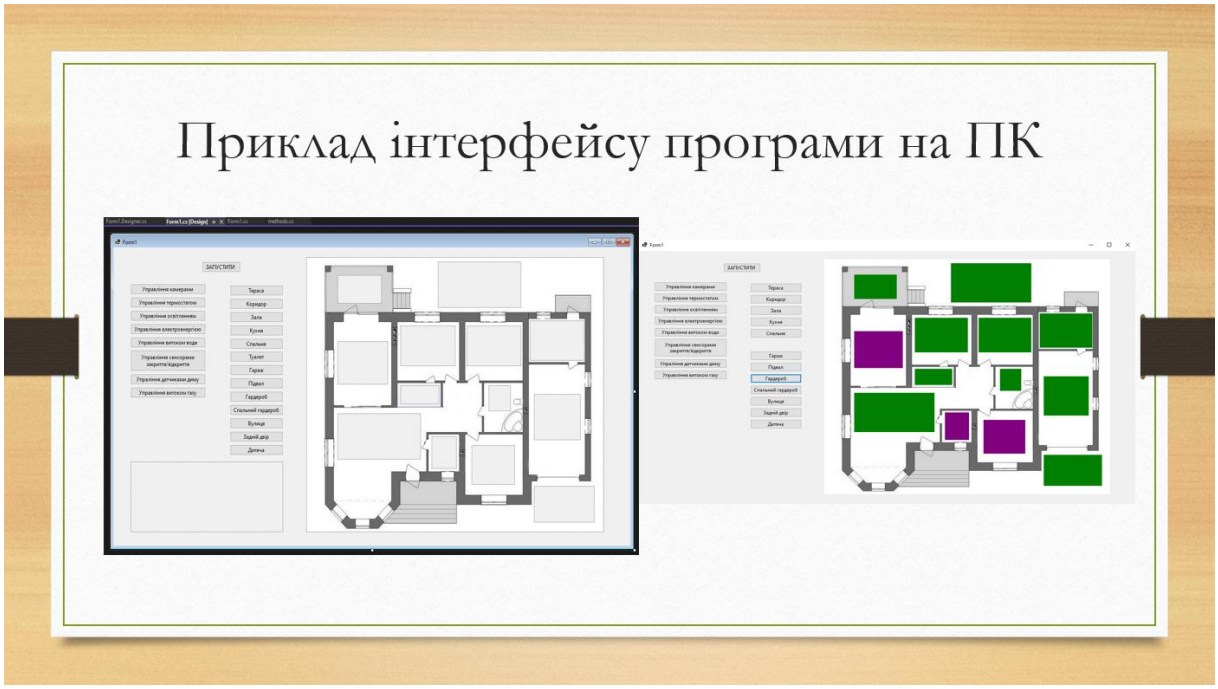
Слайд 7 – Макет інтерфейсу керування



Слайд 8 – Функціональні блоки інтерфейсу та use case діаграма

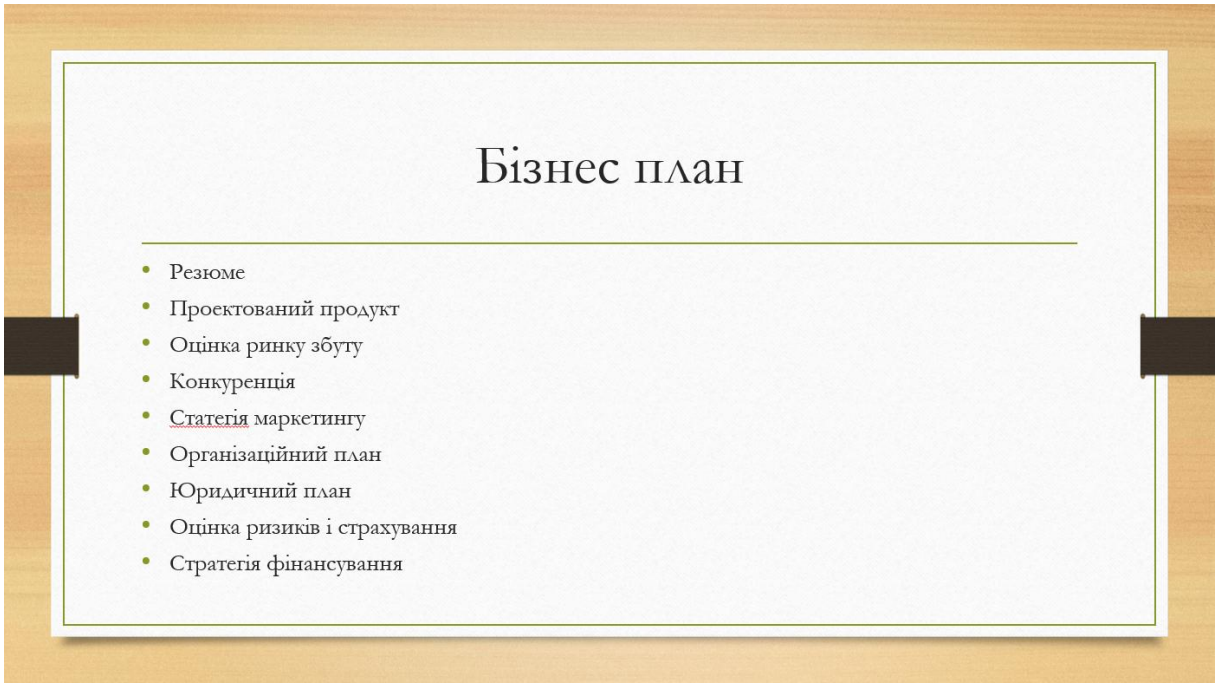
Слайд 9 – Структурна схема функцій

Слайд 10 – Вибір програмного інструментарію

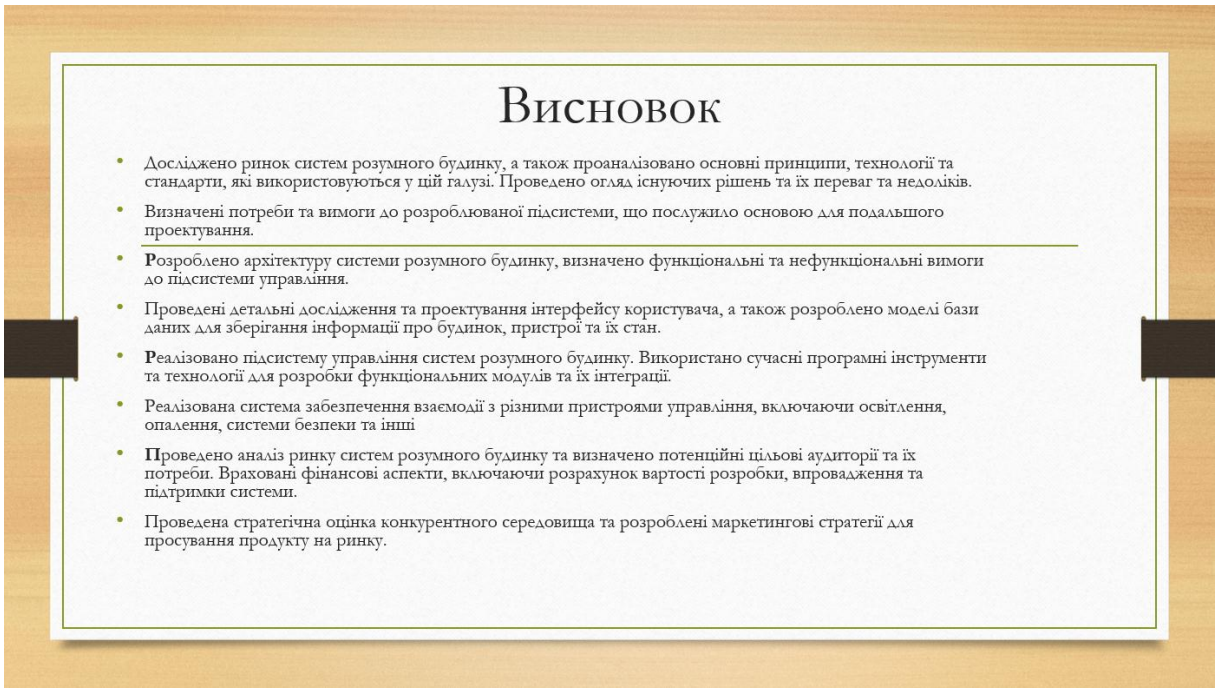


Слайд 11 – Приклад інтерфейсу програми на ПК

Слайд 12 – Структурна схема управління



Слайд 13 – Бізнес план



Слайд 14 – Висновок