

ОДЕСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ І. І. МЕЧНИКОВА

(повне найменування закладу вищої освіти)

Факультет математики, фізики та інформаційних технологій

(повне найменування факультету)

Кафедра комп'ютерних систем та технологій

(повна назва кафедри)

Кваліфікаційна робота

на здобуття ступеня вищої освіти «бакалавр»

«Побудова елементів системи "Розумний будинок"»

(тема дипломної роботи українською мовою)

«Construction of elements of the "Smart Home" system»

(тема дипломної роботи англійською мовою)

Виконав: здобувач денної форми навчання
спеціальності 123 – Комп'ютерна інженерія

(код, назва спеціальності)

Освітня програма Комп'ютерна інженерія

(назва)

Іванов Володимир Сергійович

(прізвище, ім'я, по-батькові здобувача)

Керівник д.т.н., проф. Гунченко Ю.О.

(науковий ступінь, вчене звання, прізвище, ініціали)

(підпис)

Рецензент Єпик М. О.

(науковий ступінь, вчене звання, прізвище, ініціали)

Рекомендовано до захисту:
Протокол засідання кафедри

№ від . . 2024 р.

Завідувач кафедри

Юрій ГУНЧЕНКО

(підпис)

(прізвище, ім'я)

Захищено на засіданні ЕК №
протокол № від . . 2024 р.

Оцінка / /
(за національною шкалою/шкалою ECTS/ бали)

Голова ЕК

Алла КОБОЗЄВА

(підпис)

(прізвище, ім'я)

Одеса 2024

АНОТАЦІЯ

У дипломній роботі досліджується проблема проектування та дослідження елементів системи "розумного будинку", які б сприяли покращенню енергоефективності та безпеки житла.

Мета даної роботи полягає у проектуванні простої та ефективної системи для розумного будинку для покращення умов життя, зменшення енергозатрат та безпеки житла, шляхом побудови деяких елементів системи та реалізації зручного додатку для управління цими елементами, що допоможе користувачам легко та зручно налаштовувати необхідні параметри.

Предметом дослідження є проектування та дослідження елементів системи "розумного будинку" для покращення енергоефективності та безпеки житла.

Об'єктом дослідження є системи "розумного будинку".

Для досягнення поставленої мети вирішуються наступні завдання:

- вивчено принципи роботи основних компонентів систем;
- проведено аналіз існуючих методів управління системою;
- розроблено власну систему управління "розумним будинком";
- спроектовані елементи системи;
- проведено експерименти та тестування розробленої системи;
- сформульовано рекомендації щодо вдосконалення та впровадження розробленого прототипу.

В результаті роботи розроблено елементи системи "розумного будинку", та зручний додаток що поєднує в собі енергоефективність, безпеку та економічність.

Очікується, що дана дипломна робота зробить значний внесок у розвиток систем "розумного будинку" та сприятиме їх ширшому впровадженню в життя людей.

Ключові слова: розумний будинок, енергоефективність, безпека житла.

ABSTRACT

The thesis examines the problem of developing and researching the elements of the "smart house" system, which would contribute to the improvement of energy efficiency and safety of housing.

The purpose of this work is to develop a simple and effective system for a smart home to improve living conditions, reduce energy consumption and home security, by building some elements of the system and implementing a convenient application for managing these elements, which will help users easily and conveniently adjust the necessary parameters.

The subject of the research is the development and research of elements of the "smart house" system to improve energy efficiency and home security.

The object of research is "smart home" systems.

To achieve the goal, the following tasks are solved:

- the principles of operation of the main system components are studied;
- analysis of existing system management methods was carried out;
- developed own "smart home" management system;
- designed elements of the system;
- experiments and testing of the developed system were conducted;
- recommendations for improvement and implementation of the developed prototype were formulated.

As a result of the work, the elements of the "smart house" system and a convenient application combining energy efficiency, safety and economy were developed.

It is expected that this thesis will make a significant contribution to the development of "smart home" systems and will contribute to their wider implementation in people's lives.

Keywords: smart house, energy efficiency, home security.

ЗМІСТ

ВСТУП.....	5
1 ОГЛЯД ІСНУЮЧИХ СИСТЕМ	8
1.1 Apple HomeKit	8
1.2 Google Home	9
1.3 Amazon Alexa.....	9
1.4 Xiaomi Aqara	10
1.5 KNX.....	11
1.6 Уточнення постановки задачі	12
2 ПРОЄКТУВАННЯ СИСТЕМИ	13
2.1 Вибір комплектуючих	14
2.2 Рівень контролерів	21
2.3 Рівень додатку	22
2.4 Проектування бази даних.....	23
2.5 Вибір програмного забезпечення	25
3 РЕАЛІЗАЦІЯ СИСТЕМИ	28
3.1 Схема роботи	28
3.2 Опис класів та архітектури додатку	30
3.3 Опис роботи системи	33
3.4 Опис роботи інтерфейсу	35
ВИСНОВКИ	40
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ	42

ВСТУП

Людство постійно прагне до оптимізації побуту, створюючи та вдосконалюючи пристрої, що автоматизують рутинні задачі. Їх можливості іноді навіть перевищують межі людських здібностей. Наприклад, сучасні системи "розумного будинку" здатні керувати всіма аспектами житла набагато швидше, точніше та економніше, ніж сама людина.

Електронні системи управління будинком вже міцно увійшли до нашого життя. Вони дозволяють дистанційно керувати освітленням, опаленням, побутовою технікою та іншими функціями.

Окрім комфорту, ці системи забезпечують безпеку житла. При вторгненні вони включають світло, сирени та надсилають повідомлення на смартфон власника. Завдяки цьому можна не перейматись за безпеку навіть при від'їзді з дому. Спеціальні датчики допоможуть вчасно виявити пожежу, витік газу, затоплення та інші небезпечні ситуації. Більшість систем "розумного будинку" оснащені відеоспостереженням та підключаються до персональної консолі безпеки.

Переваги розумного будинку:

- економія часу та ресурсів: автоматизація рутинних завдань звільняє ваш час та заощаджує енергію;
- підвищений комфорт: налаштування освітлення, температури та інших параметрів відповідно до потреб;
- безпека: система попереджає про небезпеку і допомагає запобігти її виникненню;
- дистанційне керування: контролюйте ваш будинок з будь-якої точки світу за допомогою смартфона або планшета;
- простота використання: управління системою "розумного дому" не потребує спеціальних знань чи навичок.

Сфери використання розумних приладів можна умовно поділити на декілька типів:

- освітлення: автоматичне або дистанційне управління освітленням – вимкнення всього освітлення в квартирі одним натисканням, приглушене світло під час перегляду кіно, закривання жалюзі тощо;
- клімат контроль: керування кондиціонерами, теплою підлогою з потрібною температурою та регулювання роботи вентиляцій;
- управління побутовими приладами: більшість сучасної побутової техніки можна підключити до «Дому» – можна дистанційно контролювати холодильник, робот-пилосос, телевізори, пральну та посудомийну машини, чайник та кавоварку, тощо;
- безпека: інтелектуальна протипожежна, захист від протікання та охоронна системи з датчиками руху, камери спостереження та розумні замки працюють цілодобово.

Але досить складно зробити високоякісну систему, яка буде працювати ідеально. Сьогоднішні розробки мають ряд своїх недоліків:

- висока ціна: всі обладнання та витрати на монтаж і налаштування такої системи дорого коштують, а також повторне обслуговування;
- залежність від Інтернету: відсутність Інтернету блокує керування через програму, роблячи недоступними важливі функції;
- несумісність пристроїв: часто пристрої різних виробників не сумісні між собою, а додавання нових функцій може бути проблематичним;
- складання та монтаж: встановлення обладнання вимагає час та часто виклик спеціалістів;
- безпека даних: злом системи може призвести до крадіжки конфіденційних даних та банківських реквізитів;
- швидке старіння: при розробці нових систем постійно виникають, нові можливості, які несумісні зі старим обладнанням;
- складність налаштування та управління: багатьом людям важко розібратися в налаштуваннях та керуванні системами;
- інтелектуальність і автономність: багато систем розумного будинку потребують ручного керування, що знижує їхню автономність;

- персоналізація та адаптивність: багато систем не здатні до самонавчання і не можуть підлаштовуватися під індивідуальні потреби користувачів.

Розвиток систем розумного будинку знаходиться на ранній стадії, ця сфера дуже стрімко розвивається і набуває все більшої актуальності. Тому дана робота присвячена розробці елементів системи «Розумного будинку», що буде більш простою у використанні та не менш ефективною за інші. Основну увагу буде зосереджено на формуванні комфортного та безпечного проживання в будинку. Це включає освітлення, опалення та контроль за безпекою житла.

Мета даної роботи полягає у проектуванні простої та ефективної системи для розумного будинку для покращення умов життя, зменшення енергозатрат та безпеки житла, шляхом побудови деяких елементів системи та реалізації зручного додатку для управління цими елементами, що допоможе користувачам легко та зручно налаштовувати необхідні параметри.

Предметом дослідження є розробка та дослідження елементів системи "розумного будинку" для покращення енергоефективності та безпеки житла.

Об'єктом дослідження є системи "розумного будинку".

Для реалізації мети необхідно виконати наступні задачі:

- 1) вивчення принципів роботи основних компонентів систем;
- 2) аналіз існуючих методів та алгоритмів управління системою;
- 3) проектування власної системи управління;
- 4) проектування і моделювання елементів системи;
- 5) проведення експериментів та тестування спроектованої системи.

1 ОГЛЯД ІСНУЮЧИХ СИСТЕМ

Ринок систем "розумного будинку" динамічно розвивається, пропонуючи користувачам все нові й нові рішення. Вибір оптимальної системи залежить від потреб, бюджету та особистих вподобань. Розглянемо декілька з найпопулярніших систем на сьогодні.

1.1 Apple HomeKit

Максимальна зручність, надійність і безпека – ось до чого прагнуть усі виробники гаджетів і програмного забезпечення. Компанія Apple зі своєю системою HomeKit лише підтверджує ці цінності. Завдяки цій системі, кожен користувач може стати справжнім диригентом побутової техніки у власному домі. Для цього достатньо мати один із пристроїв Apple – iPhone, iPad, комп'ютер Mac або приставку Apple TV[1].

Apple передбачила спеціальну програму видачі ліцензій на використання певних девайсів у системі HomeKit. Крім того, існує безліч пристроїв, для яких розроблено спеціальні рішення у вигляді сервера-емулятора з відкритим вихідним кодом HomeBridge або міні-комп'ютера Raspberry Pi, з однією платою та розміром з кредитну картку. Також є різні п्लюзи для інтеграції з HomeKit. Отже, існує багато способів забезпечити повноцінне функціонування системи HomeKit.

Переваги:

- проста інтеграція з пристроями Apple;
- високий рівень безпеки;
- широкий спектр сумісних пристроїв.

Недоліки:

- працює лише з пристроями Apple;
- може бути дорогою для деяких користувачів.

1.2 Google Home

Програма Google Home допомагає налаштовувати пристрої розумного будинку та керувати ними [2]. Програму можна завантажити на пристрій Android, iPhone або iPad. Потім додати пристрої Google, наприклад Google Nest Mini або Nest Thermostat. Також можна підключити інші сумісні продукти для розумного будинку, наприклад світильники, камери та пристрої з підтримкою Matter.

Переваги:

- голосове управління за допомогою Google Assistant;
- широкий спектр сумісних пристроїв;
- доступна ціна.

Недоліки:

- може бути менш безпечною, ніж інші системи;
- деякі функції доступні лише в США.

1.3 Amazon Alexa

Amazon Alexa - це віртуальний асистент, створений для змагання з іншими аналогічними помічниками, такими як Siri від Apple і Google Now [3]. Розроблена в Lab126, Alexa здатна сприймати ваші голосові команди та відповідати на них, виконуючи різні завдання. Вона може відтворювати музику зі стрімінгового сервісу Spotify, створювати списки справ і керувати пристроями інтернету речей у домі, наприклад, світильниками Philips Hue або термостатом Google Nest.

Спочатку Alexa набула популярності завдяки колонці Amazon Echo, яка функціонує не лише як динамік, але й як хаб для пристроїв розумного дому. Сьогодні Alexa сумісна з багатьма іншими пристроями. Цей віртуальний асистент постійно вдосконалюється та оновлюється. Хоча вона може мати труднощі з розпізнаванням команд, як і будь-який інший асистент, завдяки

підключенню до хмарного сховища Amazon оновлення відбуваються автоматично. Технологія машинного навчання допомагає робити Alexa більш корисною для користувачів з часом.

Переваги:

- широкий спектр сумісних пристроїв;
- потужні голосові можливості;
- доступ до Amazon Prime та інших сервісів.

Недоліки:

- може бути складною для налаштування;
- деякі функції доступні лише за підпискою.

1.4 Xiaomi Aqara

Aqara є суббрендом китайської компанії Xiaomi, відомої своєю жорсткою прив'язкою пристроїв до регіонів [4]. Китайські смартфони компанії мають багато регіональних обмежень, яких немає в європейських версіях. Наприклад, вони не мають доступу до популярного магазину додатків Play Market, що робить установку програм практично неможливою. Ситуація з системами розумного дому аналогічна. Хоча додаткові програми для них встановлювати не потрібно, це не означає, що ви зможете використовувати їх без обмежень. У випадку з розумними пристроями Aqara виробник поділяє пристрої на різні регіони використання, такі як материковий Китай, Європа, США, Корея та інші. Україна входить до європейського регіону, що надає пристроям додаткову перевагу – максимальну безпеку.

Переваги:

- доступна ціна;
- простий монтаж та налаштування;
- широкий спектр сумісних пристроїв.

Недоліки:

- немає голосового управління;

- деякі функції доступні лише в Китаї.

1.5 KNX

Завдяки розвитку технологій інтелектуального управління, з'явилося багато неймовірних систем, які раніше існували лише в кіно. Наприклад, відчинення дверей за допомогою мобільного телефону або увімкнення світла за допомогою голосових команд. Сьогодні ці інтелектуальні пристрої стали частиною нашого повсякденного життя. Однак незахищені або вразливі системи можуть бути піддані несанкціонованому втручанню злоумисників, що може призвести до витоку даних і порушення приватного життя. Таким чином, під загрозою опиняються особиста безпека та збереження майна. Для забезпечення безпеки передачі даних була розроблена система "KNX Data Secure", яка вирішує поточні та майбутні завдання, пов'язані з кібербезпекою при автоматизації будівель [5].

Переваги:

- відкрита та гнучка система;
- високий рівень надійності та безпеки;
- широкий спектр сумісних пристроїв від різних виробників.

Недоліки:

- складна та дорога в інсталяції;
- потребує допомоги кваліфікованого фахівця.

Існують інші, менш популярні системи в яких існують такі недоліки:

- часті випадки збоїв у системі та втрата зв'язку, рідко виходять оновлення ПЗ;
- досить висока ціна, зрідка можливі помилкові спрацьовування, слабкі елементи живлення;
- тривалі відповіді тех-підтримки, система довго завантажується немає фіксаторів у комплекті, складний інтерфейс програми.

1.6 Уточнення постановки задачі

У цій теоретичній частині дипломної роботи ми провели огляд сучасних систем. На основі цього огляду можна зробити наступні висновки та уточнення постановки задачі дипломної роботи.

Метою дипломної роботи є розробка та дослідження системи "розумного будинку", яка буде:

- енергоефективною: допоможе зменшити споживання енергії в житловому приміщенні;
- безпечним: забезпечить захист житла від загроз;
- економічним: буде доступним за ціною та простим в монтажі.

Тож для досягнення мети поставлено наступні завдання:

- провести огляд існуючих систем "розумного будинку" та елементів, що входять до їх складу;
- вивчити принципи роботи систем "розумного будинку";
- проаналізувати системи та елементи, що забезпечують безпеку житлового приміщення;
- розробити прототип системи "розумного будинку", яка буде відповідати поставленим вимогам;
- провести аналіз розробленого прототипу.
- сформулювати рекомендації щодо вдосконалення та впровадження розробленого прототипу.

Надалі розроблений прототип системи "розумного будинку" може бути використано для вдосконалення існуючих систем та створення нових.

2 ПРОЄКТУВАННЯ СИСТЕМИ

На сьогоднішній день найбільш популярні пристрої розумного будинку працюють на базі технологій Wi-Fi, Bluetooth, ZigBee та Z-Wave. Кожна з цих технологій має свої переваги та недоліки, і їх можна використовувати разом, щоб компенсувати недоліки кожної окремо. Проте для різних завдань і типів розумних пристроїв застосовуються різні технології. Розглянемо кожен більш детально.

Wi-Fi [6] незамінний для IP-камер, телевізорів, аудіо/медіа-плеєрів та іншої техніки, що передає відеосигнал. Wi-Fi можна використовувати з різними датчиками та термостатами, але є проблема що немає ретрансляції сигналу. Через це важко створити датчики, що працюють роками. Кожен виробник випускає свою власну програму для керування Wi-Fi-пристроєм, будь то розумна лампочка, чайник, холодильник або робот-пилосос, що ускладнює управління всією технікою з однієї програми. Це робить «розумний будинок» на базі Wi-Fi менш зручним.

Bluetooth [6] відрізняється низьким енергоспоживанням, завдяки чому працюють маленькі бездротові навушники, колонки і різні датчики на батарейках. Проблеми аналогічні до Wi-Fi: відсутність загального стандарту управління, що змушує кожного виробника створювати власний додаток, незручний для користувача.

ZigBee [6] був розроблений для мереж датчиків, таких як лічильники електроенергії, води, газу, датчики температури. Мережа може мати різну топологію, включаючи чарункову (mesh), що дозволяє будь-якому датчику передавати сигнал через інші датчики, значно підвищуючи надійність передачі. Більшість елементів для розумних будинків випускаються з ZigBee: термостати, датчики, реле і так далі. Проте побутові прилади, такі як холодильники і телевізори, з ZigBee не випускаються. Порівняно з іншими протоколами для «розумного будинку», у ZigBee-пристроїв найбільш

привабливі ціни, але відсутність повної сумісності між пристроями і хабами різних виробників не дозволяє побудувати розумний будинок лише на ZigBee.

Z-Wave [6]— це ще один бездротовий протокол, що використовується для автоматизації "розумного будинку". Його головна перевага полягає в тому, що його можна без проблем комбінувати та поєднувати компоненти від різних брендів, не турбуючись про сумісність. На ринку представлено широкий спектр Z-Wave пристроїв, які охоплюють всі аспекти "розумного будинку", від освітлення та термостатів до датчиків та замків. Він також використовує мережеву топологію mesh. Однак, важливою вадою Z-Wave є його ціна, що приблизно вдвічі дорожча, ніж аналоги на базі ZigBee.

Для максимально ефективної роботи розумного будинку ми будемо використовувати різні протоколи керування для різних елементів.

Саму структуру системи «розумний будинок» можна поділити на 3 рівні:

- рівень датчиків та елементів;
- рівень контролера, до якого приєднуються всі елементи;
- рівень додатку для управління системою.

2.1 Вибір комплектуючих

Перед вибором пристроїв потрібно визначитися з завданнями, які буде вирішувати розумний будинок. У цій дипломній роботі буде реалізована безпека і часткова автоматизація. Для цього розглянемо:

- автоматизоване освітлення;
- передбачення небезпечних ситуацій з сигналізацією;
- контроль використання електричної енергії;
- управління параметрами середовища;
- підтримка мікроклімату.

2.1.1 Датчик освітленості

BH1750 – це цифровий датчик освітленості, який вимірює інтенсивність світла в люксах [7] (Рис. 2.1.1). Він використовується в багатьох застосунках для автоматизації освітлення, завдяки своїй точності та простоті використання.



Рис. 2.1.1 – Світлочутливий модуль датчика освітленості BH1750

Основні характеристики:

- діапазон вимірювань: 1-65535 люкс;
- інтерфейс: I2C, що дозволяє легко підключати сенсор до мікроконтролерів, таких як Arduino;
- напруга живлення: 2.4 - 3.6V;
- висока точність, завдяки цифровому виходу;
- компактний, що дозволяє легко інтегрувати його в різні проекти.

Принцип роботи: BH1750 працює на основі фотодіода, який перетворює світловий потік на електричний сигнал. Цей сигнал обробляється внутрішнім аналого-цифровим перетворювачем (ADC), який надає результат у цифровому вигляді через I2C інтерфейс.

2.1.2 Датчик температури і вологості

Для вимірювання температури і вологості у системі розумного будинку можна використовувати популярний сенсор DHT11 [8] (Рис. 2.1.2). Цей сенсор широко використовуються і легко інтегруються з Arduino.

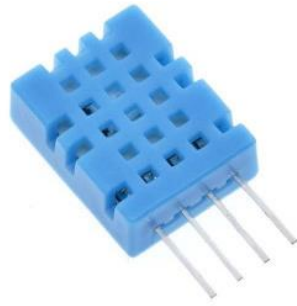


Рис. 2.1.2 – Модуль вимірювання температури і вологості

Основні характеристики DHT11:

- температурний діапазон 0-50°C ($\pm 2^\circ\text{C}$);
- діапазон вологості 20-80% RH ($\pm 5\% \text{ RH}$);
- частота вимірювань раз на 1 секунду;
- недорогий.

Принцип роботи: DHT11 складається з ємнісного датчика вологості і термістора для вимірювання температури. Датчик періодично вимірює температуру та вологість і передає ці дані у вигляді цифрового сигналу на мікроконтролер.

2.1.3 Піроелектричний інфрачервоний (PIR) датчик руху

Для виявлення руху в системі можна використовувати PIR-датчик (Passive Infrared Sensor). PIR-датчики зазвичай використовуються для виявлення руху, оскільки вони можуть визначити зміну інфрачервоного випромінювання, що виходить від людини. [9] (рис. 2.1.3)



Рис. 2.1.3 – Піроелектричний інфрачервоний (PIR) датчик руху HC-SR501

Основні характеристики HC-SR501:

- дальність виявлення: 0-7 м;
- кут спрацьовування: 110 ° на дистанції до 7 м;
- напруга живлення (рекомендована): 4.5-12 В;
- вихідна напруга логічного рівня: 0-3.3 В;
- час затримки: 0.3-300 секунд (регулюється);
- метод спрацьовування: L неповторне перемикання, Н повторюване перемикання;
- максимальний вихідний струм: 65 мА;
- робочі температури: -20-50 град. Цельсія;
- розміри: 32x24 мм.

Принцип роботи: PIR (Passive Infrared Sensor) датчики виявляють інфрачервоне випромінювання, яке випромінюють об'єкти з температурою вище абсолютного нуля. Датчик має дві послідовні піроелектричні елементи. Коли рух виявляється, інфрачервоне випромінювання змінюється і створює різницю сигналів між двома елементами, що призводить до зміни вихідного сигналу. Цей сигнал передається на мікроконтролер для подальшої обробки.

З цими компонентами можна відстежувати та реагувати на рух у розумному будинку. PIR-датчик у поєднанні з Arduino і веб-додатком дозволить створити систему безпеки, яка повідомлятиме про виявлення руху та може виконувати додаткові дії, наприклад, включення світла або сигналізації.

2.1.4 Датчик природного газу

Для виявлення природного газу можна використовувати газовий датчик MQ-9 [10]. Цей датчик чутливий до монооксиду вуглецю (CO) та зрідженого газу (LPG), метану (CH₄), пропану (C₃H₈). MQ-9 має аналоговий вихід, який можна підключити до аналогового піну Arduino або іншого мікроконтролера. (Рис. 2.1.4)



Рис. 2.1.4 – Датчик природного газу MQ-9

Основні характеристики MQ-9:

- напруга живлення 5В;
- напруга нагрівача мінімальна 1.4В;
- напруга нагрівача максимальна: 5В;
- опір нагрівача: 33Ом;
- потужність нагрівача: 340мВт;
- час розігріву: від 60 до 90 с;
- опір датчика: від 2 до 20 кОм;
- параметри вимірювання: температура $-20 \pm 2^\circ$ відносна вологість $65\% \pm 5\%$ опір навантаження датчика: $10\text{k}\Omega \pm 5\%$

Принцип роботи: MQ-9 використовує нагрівальний елемент і оксид металу для виявлення газів. Коли певний газ (наприклад, монооксид вуглецю або метан) присутній, він взаємодіє з поверхнею оксиду металу і змінює його електричний опір. Датчик видає аналоговий сигнал, який пропорційний концентрації газу. Цей сигнал можна прочитати за допомогою мікроконтролера.

2.1.5 Датчик полум'я

Для виявлення присутності вогню або полум'я в приміщенні використовується датчик полум'я KY-026 [11]. Він може бути корисним для системи пожежної безпеки у розумному будинку. Зовнішній вигляд датчика показаний на рис. 2.1.5.



Рис. 2.1.5 – Модуль полум'я на датчику KY-026

Основні характеристики KY-026:

- діапазон виявлення: 760 нм (інфрачервоне випромінювання);
- живлення: 3.3V-5V;
- споживаний струм: 20 мА;
- вихідний сигнал: Аналоговий (АО) і цифровий (DO);
- кут виявлення: 60 градусів;
- робоча температура: -25°C до 85°C.

Принцип роботи: датчик полум'я зазвичай використовує інфрачервоний (ІЧ) приймач для виявлення інфрачервоного випромінювання, яке випромінює полум'я. Датчик полум'я реагує на інфрачервоні промені з довжиною хвилі приблизно 760 нм, що відповідає спектру випромінювання полум'я. Коли полум'я виявляється, сигнал на цифровому виході (DO) змінюється, або на аналоговому виході (АО) змінюється рівень напруги пропорційно інтенсивності полум'я.

2.1.6 Модуль реле

Реле – це електро cơханічний пристрій, який використовується для керування електричними навантаженнями за допомогою малих електричних сигналів. Вони можуть бути використані для управління освітленням, електричними розетками, насосами тощо.

Принцип роботи: Реле має спіральну обмотку, яка вмикається при подачі струму. Це створює магнітне поле, яке зміщує контакти у реле, з'єднуючи або роз'єднуючи їх. Коли обмотка вимикається, магнітне поле зникає, і контакти повертаються у вихідне положення.

Технічні характеристики:

- напруга живлення: 5/12 В;
- струм: 15-20 мА;
- сигнал включення: 0 В (низький рівень), 1 (високий рівень);
- оптична ізоляція: немає;
- кількість реле: 1 шт;
- тип реле: електромеханічне;
- номінальний струм навантаження: 10 А;
- комутована напруга: 250VAC, 30VDC;
- габарити: 43мм x 20мм x 18мм.

2.1.7 Джерело безперебійного живлення

Для живлення електронних компонентів системи розумного будинку потрібні блоки живлення. Ці блоки перетворюють вхідну напругу (зазвичай з розетки) на необхідний рівень напруги для живлення мікроконтролерів, датчиків та інших пристроїв. Для цього можуть використовуватися стандартні блоки живлення з вихідними напругами 5V або 3.3V.

Для забезпечення неперервного живлення важливих компонентів системи розумного будинку під час перебоїв у живленні можуть використовуватися аварійні джерела живлення, такі як акумуляторні батареї або джерела безперебійного живлення (UPS). Це дозволяє системі продовжувати працювати навіть під час відключення живлення.

Щоб захистити електронні компоненти від можливих перенапруг, можуть використовуватися реле або запобіжники. Ці пристрої автоматично

відключають живлення в разі перенапруги, щоб запобігти пошкодженню електроніки.

2.2 Рівень контролерів

В якості контролера використаємо Arduino Uno – це відкрита електронна платформа, що базується на простому апаратному та програмному забезпеченні. Вона включає в себе мікроконтролер, який можна програмувати для виконання різноманітних завдань, від управління світлодіодами до зчитування даних з датчиків.

Характеристики:

- процесор: ATmega328P (Uno);
- частота: 16 MHz;
- вхідна напруга: 7-12V (рекомендовано);
- виходи: 14 цифрових входів/виходів (6 з них можуть використовуватись як PWM виходи), 6 аналогових входів (Uno);
- переваги: легкий у використанні, велика кількість доступних бібліотек і прикладі.

Принцип роботи: Arduino виконує програму, яку ви написали на мові Arduino (базованій на C/C++). Запрограмований код за допомогою Arduino IDE завантажується на плату через USB. Коли код завантажений, Arduino виконує інструкції, включаючи зчитування вхідних сигналів від датчиків, виконання обчислень та керування виводами для контролю акторів.

Для комунікації використовується WiFi модуль ESP8266 – це мікроконтролер, що має вбудований WiFi модуль. Він використовується для побудови IoT пристроїв, які можуть підключатися до Інтернету безпосередньо через WiFi.

Характеристики:

- частота: 2.4 GHz
- протоколи: 802.11 b/g/n

- переваги: широко використовуваний стандарт, легкість інтеграції з інтернетом, підтримка багатьох протоколів і бібліотек (MQTT, HTTP).

Принцип роботи: ESP8266 також програмується, але він має вбудований WiFi. Після програмування він може зчитувати дані з датчиків, відправляти їх на сервер або обробляти локально та керувати акторами.

Zigbee – це бездротовий протокол, спеціально призначений для мереж з багатьма вузлами. Zigbee модулі використовуються для побудови бездротових мереж сенсорів і акторів.

Характеристики:

- частота: 2.4 GHz;
- протоколи: Zigbee (IEEE 802.15.4);
- переваги: низьке енергоспоживання, надійність і стабільність в мережах з багатьма вузлами, підтримка мережових топологій (mesh, star);
- недоліки: вища вартість порівняно з WiFi, необхідність спеціального координаційного пристрою (hub).

Принцип роботи: кожен пристрій у мережі є вузлом і може спілкуватися з іншими вузлами. Вони можуть використовувати різні топології мережі, такі як зірка або сітка. Коли пристрої взаємодіють, вони можуть відправляти дані один одному або до координаційного пристрою, який потім може передати дані на сервер або інші пристрої.

2.3 Рівень додатку

Необхідно передбачити віддалене управління розумним будинком. Людина може знаходитись за тисячі кілометрів від дому і мати можливість перевіряти, що все гаразд: чи не вимкнене світло, чи немає витoku газу. Датчики температури, вологості, освітлення та інші пристрої збирають інформацію, яку користувач може бачити на смартфоні чи планшеті. За

допомогою смартфона користувач може керувати будь-яким елементом підключеним до системи. А також система негайно повідомить користувача про будь-які проблеми, будь то витік газу, пожежа.

У зв'язку з цим реалізація підключення Arduino здійснено у вигляді веб-додатку за шаблоном MVC за допомогою python flask:

- простота та гнучкість Flask у поєднанні з чіткою організацією коду MVC роблять додаток легким у розробці, підтримці та розширенні;
- гнучкість MVC допомагає зберігати проект структурованим та організованим, навіть якщо додаток стає більш складним;
- гнучкість Flask дозволяє інтегрувати будь-які необхідні компоненти, не накладаючи обмежень на архітектуру додатку.

2.4 Проектування бази даних

Для того щоб користувач міг самостійно налаштовувати свій будинок та зберігати дані спроектовано базу даних (рис. 2.4.1).

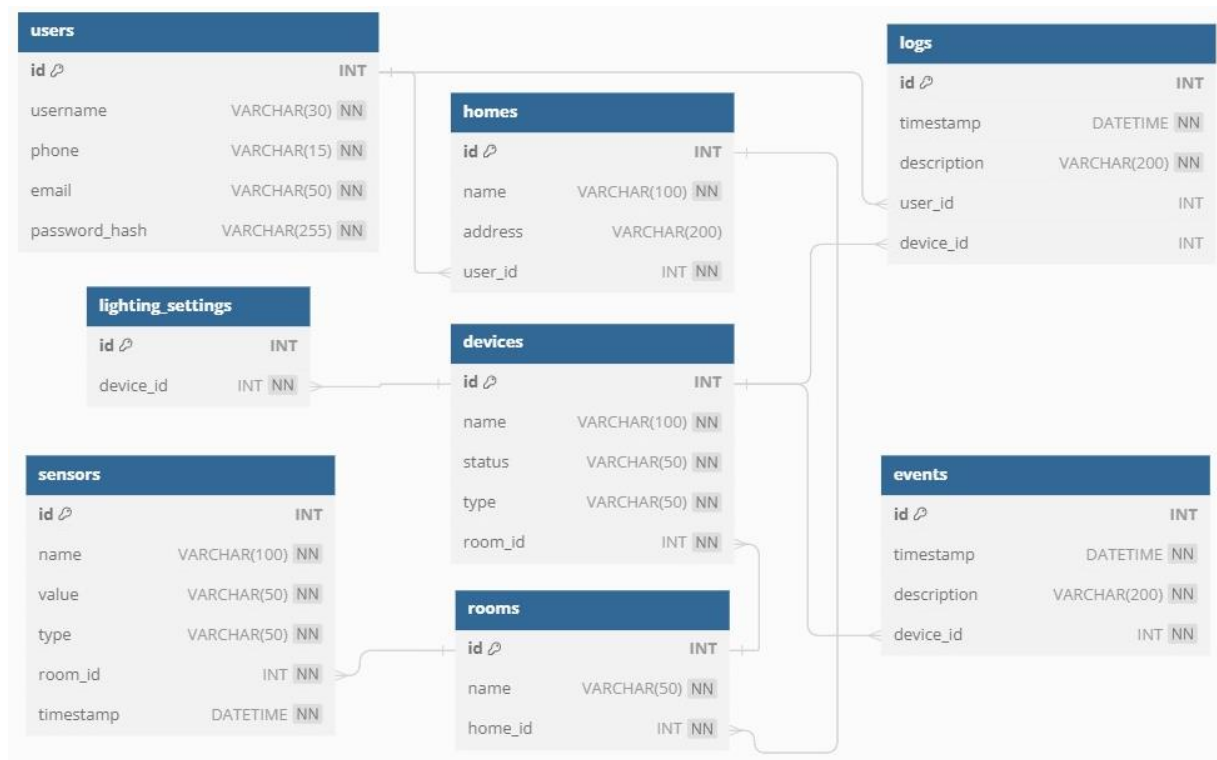


Рисунок 2.4.1 – UML-діаграма бази даних

База даних для системи розумного будинку включає кілька таблиць, які взаємопов'язані між собою для забезпечення зручного управління будинками, кімнатами, пристроями та логами подій.

Таблиця `users` зберігає інформацію про користувачів системи. Кожен користувач має унікальний ідентифікатор (`id`), ім'я користувача (`username`), номер телефону (`phone`), електронну пошту (`email`) та хеш паролю (`password_hash`). Ці дані забезпечують аутентифікацію та ідентифікацію користувачів у системі.

Таблиця `homes` містить дані про будинки, якими володіють користувачі. Кожен будинок має унікальний ідентифікатор (`id`), назву (`name`), адресу (`address`) та посилання на користувача (`user_id`), якому належить цей будинок. Це дозволяє кожному користувачу мати один або кілька будинків у системі.

Таблиця `rooms` містить інформацію про кімнати в будинках. Кожна кімната має унікальний ідентифікатор (`id`), назву (`name`) та посилання на будинок (`home_id`), до якого вона належить. Це дозволяє відстежувати структуру кімнат у кожному будинку.

Таблиця `devices` зберігає інформацію про пристрої, встановлені в кімнатах. Кожен пристрій має унікальний ідентифікатор (`id`), назву (`name`), статус (`status`), тип (`type`) та посилання на кімнату (`room_id`), в якій він знаходиться. Це дозволяє керувати різними типами пристроїв, такими як освітлення, клімат-контроль тощо.

Таблиця `lighting_settings` містить специфічні налаштування для пристроїв освітлення. Вона включає унікальний ідентифікатор (`id`) та посилання на пристрій (`device_id`), для якого ці налаштування застосовуються.

Таблиця `sensors` зберігає дані про сенсори, встановлені в кімнатах. Кожен сенсор має унікальний ідентифікатор (`id`), назву (`name`), значення (`value`), тип (`type`), посилання на кімнату (`room_id`) та час отримання даних (`timestamp`). Це дозволяє відстежувати показники сенсорів, такі як температура, вологість, дим тощо.

Таблиця logs містить інформацію про дії користувачів у системі. Кожен запис має унікальний ідентифікатор (id), час події (timestamp), опис дії (description), посилання на користувача (user_id), який виконав дію, та пристрій (device_id), на якому ця дія була виконана. Це дозволяє зберігати історію взаємодій користувачів з системою.

Таблиця events зберігає інформацію про події, пов'язані з пристроями. Кожна подія має унікальний ідентифікатор (id), час події (timestamp), опис події (description) та посилання на пристрій (device_id), який генерував цю подію. Це дозволяє відстежувати автоматичні та системні події, що відбуваються в системі.

2.5 Вибір програмного забезпечення

Обираючи СУБД для реалізації рівня управління ресурсами додатку розумний дім вибір став на PostgreSQL за такими причинами [12]:

- відкритість та безкоштовність: PostgreSQL є вільною і відкритою СУБД, що полегшує розробку проекту через відсутність необхідності сплати коштів за користування ПЗ;
- підтримка різноманітних типів даних: PostgreSQL підтримує багато типів даних, включаючи географічні типи, JSON, XML, і багато інших, що дозволяє ефективно зберігати та опрацьовувати різноманітні дані в рамках застосунку;
- надійність та стабільність: PostgreSQL відомий своєю надійністю та стабільністю. Він має розгорнуті механізми резервного копіювання, реплікації та відновлення, що забезпечують безпеку даних і готовність системи до відновлення в разі виникнення проблем;
- активна спільнота та підтримка: PostgreSQL має велику та активну спільноту користувачів та розробників, яка постійно вносить внески у розвиток та підтримку цієї СУБД, що забезпечує доступність

різноманітних ресурсів, включаючи документацію, форуми підтримки та сторонні розширення;

- PGAdmin – це програмне забезпечення для адміністрування баз даних PostgreSQL, що розроблене спеціально для роботи з PostgreSQL і надає ряд зручних інструментів для управління базами даних, виконання запитів SQL, адміністрування користувачами та доступом, моніторингу процесів та багато іншого.

Веб-застосунок розумний дім розроблено за допомогою мови програмування Python з використанням фреймворку Flask [13], ORM SQLAlchemy та шаблонного движку Jinja2.

Вибір Flask як головного фреймворку обумовлений його простотою, легкістю вивчення та використання, а також активною спільнотою розробників, яка забезпечує підтримку, навчальні матеріали та регулярні оновлення.

Flask-SQLAlchemy: забезпечує простий та ефективний спосіб взаємодії з базами даних з використанням SQLAlchemy ORM. Цей модуль дозволяє здійснювати операції з базою даних за допомогою об'єктів Python, що спрощує розробку та підтримку коду.

Також для взаємодії інтерфейсу користувача із серверною частиною застосунку використано Jinja2. Цей шаблонний движок дозволяє вбудовувати динамічні дані у HTML-сторінки та зручно керувати їх відображенням.

Основні переваги Jinja2:

- легка інтеграція з Flask: Jinja2 входить до складу Flask і має інтуїтивно зрозумілий синтаксис, що робить його ідеальним вибором для розробки веб-застосунків з використанням Flask;
- гнучкість та розширюваність: Jinja2 дозволяє створювати складні шаблони, використовуючи умови, цикли та інші конструкції, що робить його потужним інструментом для веб-розробки;

- активна спільнота розробників: Jinja2 має велику та активну спільноту розробників, яка надає підтримку, документацію та регулярні оновлення.

В якості середовища розробки для програмування платформ Arduino використана Arduino IDE [14]. Вона має такі переваги:

- проста у використанні і налаштуванні, Arduino IDE має дуже простий інтерфейс, що дозволяє швидко налаштувати новий проект і почати програмувати без зайвих складнощів;
- підтримка мови програмування C/C++, Arduino IDE використовує мову програмування C/C++, що є однією з найпоширеніших мов для мікроконтролерів. Це дозволяє розробникам використовувати потужні можливості мови програмування для створення складних проектів;
- кросплатформенність, Arduino IDE підтримується на різних операційних системах, включаючи Windows, macOS та Linux, що робить його доступним для широкого кола розробників;
- відладка та моніторинг серійного порту, Arduino IDE має вбудовані інструменти для відладки вашого коду та моніторингу виводу через серійний порт, що дозволяє швидко виявляти та виправляти помилки;
- бібліотеки та приклади коду, Arduino IDE поставляється з великою кількістю стандартних бібліотек та прикладів коду, що спрощує розробку різноманітних проектів;
- інтеграція з платформою Arduino, Arduino IDE працює з платформою Arduino, що означає, що виникає можливість легко програмувати та завантажувати ваш код на плати Arduino без зайвих складнощів.

Отже, Arduino IDE - це потужний інструмент для розробки проектів на платформі Arduino, який дозволяє розробникам швидко та ефективно створювати різноманітні електронні пристрої та системи.

3 РЕАЛІЗАЦІЯ СИСТЕМИ

3.1 Схема роботи

Для керування освітленням розроблена схема (рис. 3.1.1), яка використовує датчик освітлення, руху та датчик відстані. Такий елемент допоможе автоматизувати включення та виключення світла. Завдяки датчику руху світло буде вмикатись, якщо хтось зайде до кімнати. Датчик відстані допоможе не вимикати світло, якщо хтось все ще знаходиться в кімнаті, але не рухається. А датчик освітлення буде запобігати вмиканню світла вдень, коли це не потрібно, навіть якщо виявлено рух.

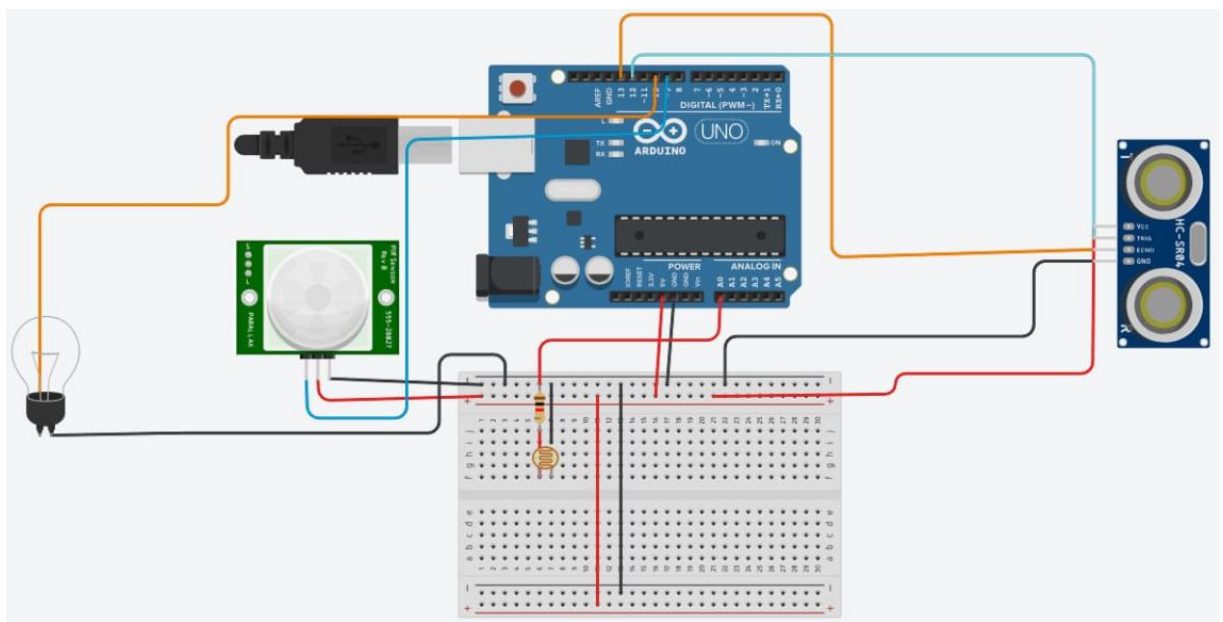


Рисунок 3.1.1 – Макетна схема елемента автоматизації освітлення

Принципова схема зображена на рис. 3.1.2.

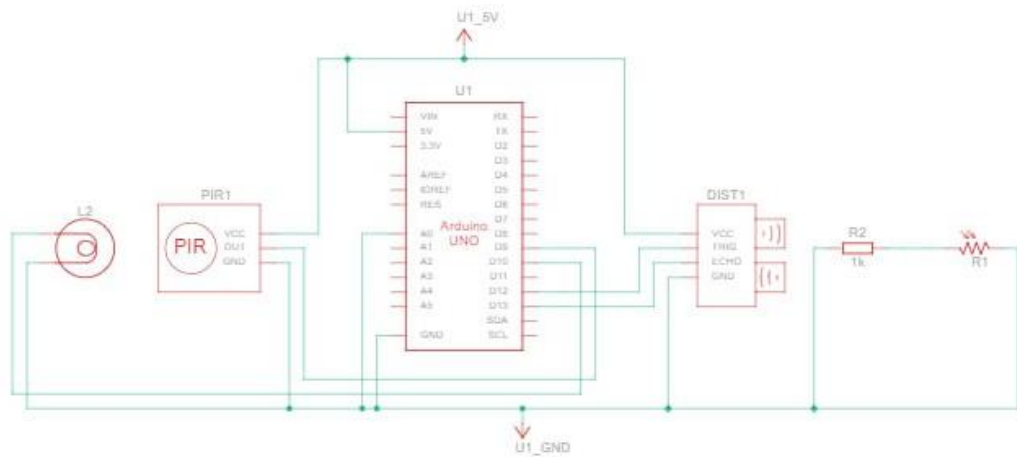


Рисунок 3.1.2 – Принципова схема елемента автоматизації освітлення

Для системи безпеки розроблена інша схема (рис. 3.1.3). Схема включає кілька компонентів, таких як датчики температури, газу, диму та вогню, а також базер для сигналізації. За допомогою цих датчиків можна буде відстежувати показники через додаток. А також при виникненні якихось небезпечних ситуацій спрацює сигналізація та клієнт отримає повідомлення на телефон.

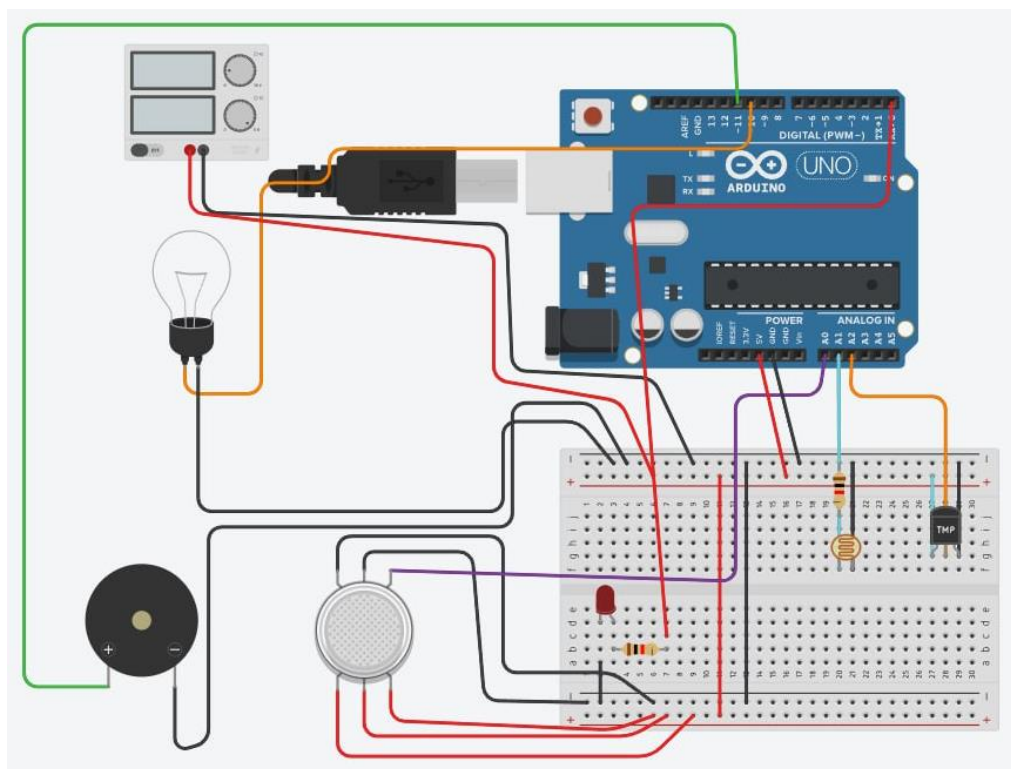


Рисунок 3.1.3 – Макетна схема елемента автоматизації освітлення

На рис. 3.1.4 зображена принципова схема цього елементу.

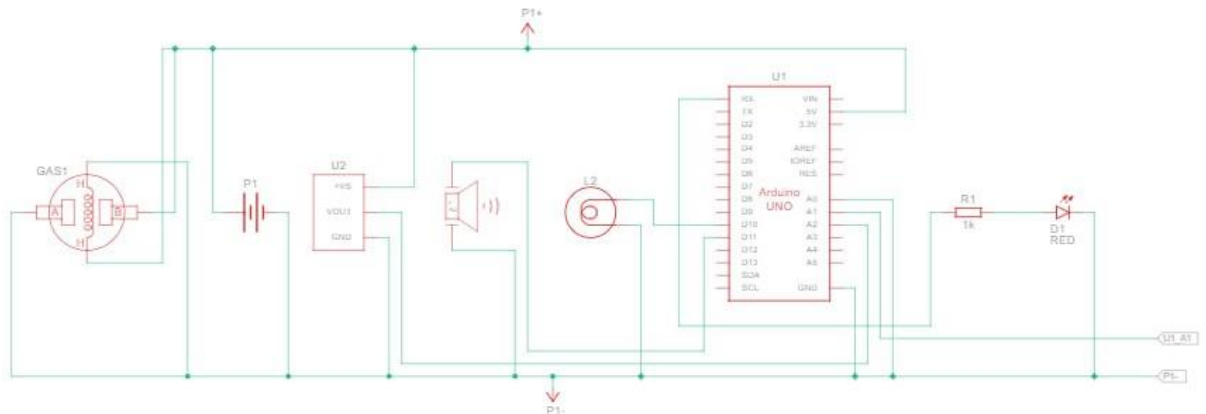


Рисунок 3.1.4 – Принципова схема елементу автоматизації освітлення

Підключення всіх компонентів системи розумного будинку розділених на окремі модулі виконується за комунікацією через Wi-Fi. Це залежить від кількох факторів:

- масштабованість: можна легко додавати нові компоненти до системи без значних змін у вже існуючих налаштуваннях;
- гнучкість: кожен модуль може бути розміщений у найбільш зручному місці, що особливо корисно для великих будинків;
- легкість в обслуговуванні: у разі проблеми з одним модулем, інші можуть продовжувати працювати без перешкод;
- мінімізація кабельного з'єднання: зменшується кількість проводів, що проходять через ваш будинок.

3.2 Опис класів та архітектури додатку

Додаток користувача був реалізований в наступним чином:

- `__init__.py` – це файл ініціалізації для пакета `app`. Він створює об'єкт `Flask` додатку та налаштовує маршрути;
- `models/` – містить моделі даних для додатку для зберігання стану пристроїв;

- controllers/ – містить логіку управління додатком, обробляє запити від користувачів, змінює стан моделей і викликає відповідні функції для відображення;
- templates/ – містить HTML-шаблони для додатку, index.html є основною сторінкою;
- smart_home.py – це головний файл для запуску Flask-додатку. Він створює додаток і запускає його на локальному сервері.

Розглянемо створення моделі, наприклад, для взаємодії з пристроями створено модель Device (лістинг 3.2.1). Клас має усі необхідні поля для збереження даних про пристрій (name, status, type, room_id) а також допоміжні поля для з'єднання з іншими таблицями.

```
class Device(db.Model):
    id = db.Column(db.Integer, primary_key=True)
    name = db.Column(db.String(100), nullable=False)
    status = db.Column(db.String(50), nullable=False)
    type = db.Column(db.String(50), nullable=False)
    room_id = db.Column(db.Integer, db.ForeignKey('room.id'),
        nullable=False)
    events = db.relationship('Event', backref='device',
        lazy=True)
    logs = db.relationship('Log', backref='device', lazy=True)
    lighting_settings = db.relationship('LightingSettings',
        backref='device', lazy=True, uselist=False)
```

Лістинг 3.2.1 – Приклад створення моделі

Далі розглянемо контролер, наприклад, для налаштування параметру освітлення (лістинг 3.2.2). Наприклад для зміни параметру освітлення користувач відкриває додаток і переходить до сторінки керування кімнатами. Він вибирає кімнату, в якій встановлено пристрій освітлення, та переходить до налаштувань освітлення. На сторінці користувач бачить поточний стан пристрою освітлення та його параметри, включаючи поточну яскравість. Він використовує повзунок (slider) для зміни рівня яскравості освітлення. Після зміни положення повзунка, додаток надсилає HTTP POST запит на сервер, вказуючи нове значення яскравості. Для цього використовується форма, дані з

якої записуються до БД та передаються в шаблон, який у свою чергу викликає функцію із запитом до пристрою за допомогою протоколу. Пристрій отримує команду, змінює рівень яскравості світла і підтверджує виконання команди, надсилаючи відповідь серверу. Сервер надсилає відповідь додатку про успішну зміну яскравості. Додаток оновлює інтерфейс користувача, відображаючи новий рівень яскравості. Всі дії користувача та пристрою логуються в таблиці `logs` для забезпечення повної історії взаємодій.

```
@app.route('/lighting/settings/<int:device_id>', methods=['GET',
'POST'])
@login_required
def lighting_settings(device_id):
    device = Device.query.get_or_404(device_id)
    settings =
LightingSettings.query.filter_by(device_id=device.id).first()
    if request.method == 'POST':
        settings.light_level = request.form['light_level']
        settings.motion_detected = 'motion_detected' in
request.form
        settings.presence_detected = 'presence_detected' in
request.form
        settings.auto_on = 'auto_on' in request.form
        settings.auto_off = 'auto_off' in request.form
        db.session.commit()
        return redirect(url_for('lighting_room',
room_id=device.room_id))
    return render_template('lighting_settings.html',
device=device, settings=settings)
```

Лістинг 3.2.2 – Приклад функції-контролера для зміни освітлення

Також необхідно запрограмувати самі елементи системи. У лістингу 3.2.3 наведено приклад роботи елемента для автоматизації освітлення. Після завантаження програми ESP8266 підключається до WiFi-мережі. Запускається веб-сервер, який слухає запити на порті 80. Коли на маршрут `/set_brightness` надходить HTTP POST-запит з параметром `brightness`, сервер отримує значення параметра, змінює рівень яскравості LED і надсилає відповідь про успішне виконання. Усі запити обробляються у безперервному циклі `loop()`.

```
#include <ESP8266WiFi.h>
#include <ESP8266WebServer.h>
```

```

const char* ssid = "SSID";
const char* password = "PASSWORD";
ESP8266WebServer server(80);
int brightness = 0;
void setup() {
    Serial.begin(115200);
    pinMode(D1, OUTPUT);

    WiFi.begin(ssid, password);
    while (WiFi.status() != WL_CONNECTED) {
        delay(500);
        Serial.print(".");
    }
    Serial.println("WiFi connected");

    server.on("/set_brightness", HTTP_POST, []() {
        if (server.hasArg("brightness")) {
            brightness = server.arg("brightness").toInt();
            analogWrite(D1, map(brightness, 0, 100, 0, 1023));
            server.send(200, "application/json",
"{\"success\":true}");
        } else {
            server.send(400, "application/json",
"{\"success\":false}");
        }
    });
    server.begin();
    Serial.println("HTTP server started");
}
void loop() {
    server.handleClient();
}

```

Лістинг 3.2.3 – Приклад програмування елемента автоматизації освітлення

3.3 Опис роботи системи

Система розумного будинку дозволяє користувачам керувати своїми будинками та підключеними пристроями через веб-інтерфейс. Користувач починає взаємодію із системою, увійшовши в обліковий запис за допомогою своїх облікових даних. Після успішної аутентифікації він потрапляє на головну сторінку, де може переглядати список усіх своїх будинків. Користувач може додавати нові будинки, редагувати інформацію про них або видаляти існуючі будинки разом з усіма пов'язаними пристроями.

Приклад роботи системи можна розглянути на рис. 3.3.1.

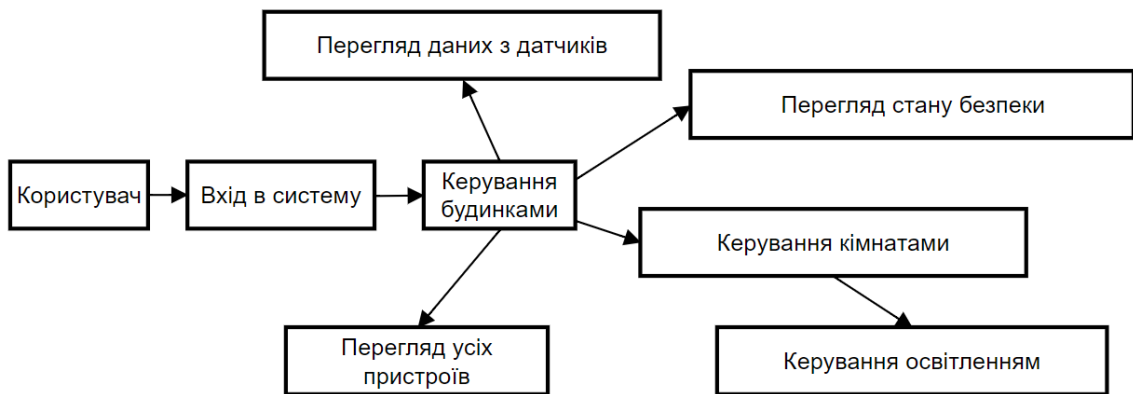


Рисунок 3.3.1 – Схема роботи системи

У кожному будинку є кілька кімнат, якими користувач може також керувати. Він може додавати нові кімнати, змінювати їхні параметри або видаляти кімнати, якщо вони більше не потрібні. У кожній кімнаті користувач може бачити перелік підключених пристроїв, таких як освітлення або клімат-контроль. Для кожного пристрою є можливість керування його параметрами: ввімкнення/вимкнення, регулювання яскравості світла тощо.

Система також дозволяє переглядати всі підключені до будинку пристрої на одній сторінці. Це дозволяє користувачеві швидко отримати уявлення про стан всіх пристроїв у будинку та оперативно ними керувати. Окрім цього, користувач може переглядати дані з різних датчиків, встановлених у будинку. Це можуть бути датчики температури, вологості, газу або вогню. Інформація з цих датчиків відображається у реальному часі, що дозволяє користувачу швидко реагувати на зміни.

Ще однією важливою функцією є система безпеки. Користувач може бачити загальний стан безпеки будинку, включаючи спрацювання датчиків газу або вогню. У разі виявлення небезпеки система автоматично надсилає користувачеві повідомлення на електронну пошту, щоб оперативно повідомити про проблему.

Уся діяльність користувача в системі, така як додавання або видалення пристроїв, зміна параметрів, вмикання/вимикання пристроїв, логуються в базу даних. Це дозволяє користувачу переглядати історію дій та виявляти можливі проблеми чи відстежувати використання системи.

Загалом система розумного будинку забезпечує зручне та інтуїтивне керування будинками та пристроями, підвищуючи комфорт та безпеку користувачів.

3.4 Опис роботи інтерфейсу

Додаток реалізовано з простим та привабливим інтерфейсом. При вході на сайт, користувач відразу потрапляє на сторінку «Вхід до системи» (рис.3.4.1).

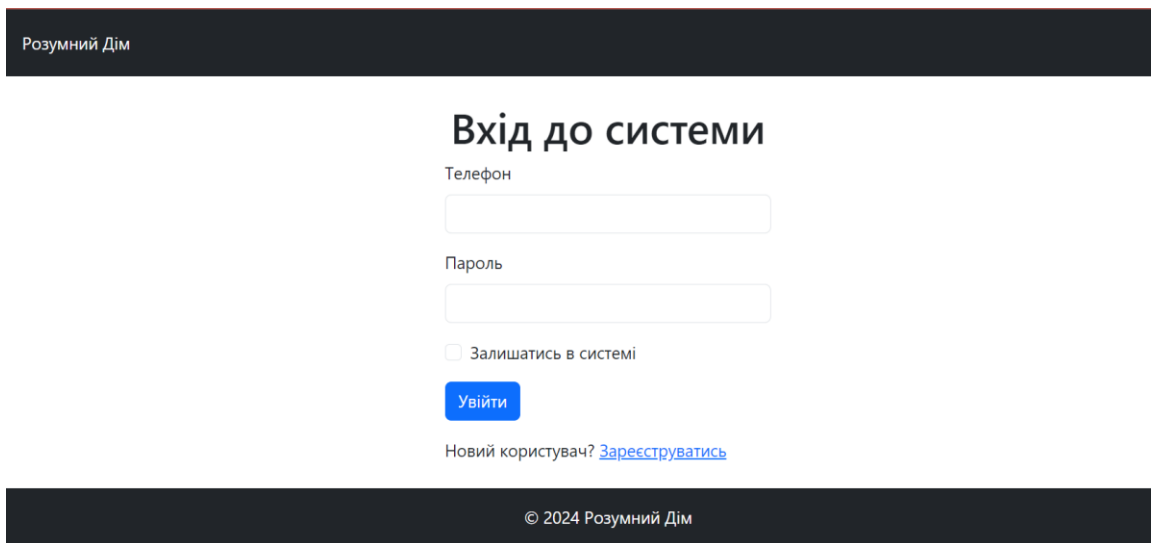


Рисунок 3.4.1 – Сторінка вхід до системи

Якщо у користувача ще немає облікового запису, він може перейти на сторінку реєстрації (рис. 3.4.2)

Реєстрація

Ім'я

Телефон

Email

Пароль

Повторіть пароль

[Зареєструватися](#)

© 2024 Розумний Дім

Рисунок 3.4.2 – Сторінка реєстрації

Після входу в систему, користувач потрапляє на головну сторінку де може бачити всі будинки додані до його системи (рис. 3.4.3). На цій сторівці він має можливість додати новий будинок (рис. 3.4.4), змінити або видалити вже існуючий.

Розумний Дім [Головна](#) [Профіль](#) [Вийти](#)

Мої будинки

[Додати новий будинок](#)

Дім 1
Одеська обл., м. Одеса, вул. Єлісаветинська, буд. 3

[Переглянути](#) [Редагувати](#) [Видалити](#)

Дім 2
Одеська обл., м. Доброслав, пров. Шевченка, буд. 5

[Переглянути](#) [Редагувати](#) [Видалити](#)

Рисунок 3.4.3 – Сторінка з переглядом будинків

Розумний Дім

ГоловнаПрофільВийти

Додати будинок

Назва

Адреса

Додати

Рисунок 3.4.4 – Форма додавання нового будинку

Також на сторінці з будинками користувач може перейти на перегляд конкретного будинку (рис. 3.4.5). На цій сторінці користувач може перейти до перегляду якоїсь кімнати, додати нову кімнату та переглянути дані з датчиків температури та вологості, а також стан системи. У разі небезпеки, користувач отримає повідомлення на пошту.

Розумний Дім

ГоловнаПрофільВийти

Будинок: Дім 1

Кімнати

Зала

Вітальня

Спальня

Кухня

Додати кімнату

Дані з датчиків

Температура: 26 °C

Вологість: 35 %

Стан безпеки: В безпеці

Рисунок 3.4.5 – Перегляд інформації конкретного будинку

Також є можливість додати нову кімнату (рис. 3.4.6) та переглянути пристрої які є в конкретній кімнаті (рис. 3.4.7).

The screenshot shows the 'Додати кімнату' (Add Room) form. At the top, there is a dark navigation bar with the text 'Розумний Дім' on the left and 'Головна', 'Профіль', and 'Вийти' on the right. Below the navigation bar, the title 'Додати кімнату' is displayed. The form contains three input fields: 'Назва кімнати' (Room Name), 'Будинок' (Building), and 'Дім 1' (Apartment 1). A blue button labeled 'Додати' (Add) is positioned below the 'Дім 1' field.

Рисунок 3.4.6 – Форма додавання нової кімнати

The screenshot shows the 'Кімната: Зала' (Room: Hall) control interface. At the top, there is a dark navigation bar with the text 'Розумний Дім' on the left and 'Головна', 'Профіль', and 'Вийти' on the right. Below the navigation bar, the title 'Кімната: Зала' is displayed. Under the title, the section 'Освітлення' (Lighting) is shown. This section contains two rows of controls. The first row is for 'Лампа' (Lamp), featuring a 'Увімкнути' (Turn On) button, a slider, and a 'Видалити' (Delete) button. The second row is for 'Нічник' (Nightstand), also featuring a 'Увімкнути' (Turn On) button, a slider, and a 'Видалити' (Delete) button. A blue button labeled 'Додати пристрій освітлення' (Add lighting device) is located below the 'Нічник' controls. At the bottom of the interface, there is a dark footer bar with the text '© 2024 Розумний Дім'.

Рисунок 3.4.7 – Перегляд та керування пристроями якоїсь кімнати

Також користувач має можливість переглядати статистику (рис. 3.4.8)

Перегляд даних з датчиків

Дата від: 15.06.2024 ☐ Дата до: 16.06.2024 ☐

Дата	Температура (°C)	Вологість (%)	Загазованість	Статус системи
2024-06-15 12:50:33	28.00C	34.00%	463	Безпека
2024-06-15 12:50:38	28.00C	34.00%	463	Безпека
2024-06-15 12:50:43	28.00C	34.00%	463	Безпека
2024-06-15 12:50:49	28.00C	34.00%	463	Безпека
2024-06-15 12:50:54	28.00C	34.00%	463	Безпека
2024-06-15 12:50:59	28.00C	34.00%	463	Безпека
2024-06-15 12:51:04	28.00C	33.00%	463	Безпека
2024-06-15 12:51:09	28.00C	33.00%	463	Безпека
2024-06-15 12:51:14	28.00C	33.00%	463	Безпека
2024-06-15 12:51:19	28.00C	33.00%	463	Безпека
2024-06-15 12:51:24	28.00C	34.00%	463	Безпека
2024-06-15 12:51:30	28.00C	34.00%	463	Безпека
2024-06-15 12:51:35	28.00C	34.00%	463	Безпека
2024-06-15 12:51:40	28.00C	34.00%	463	Безпека
2024-06-15 12:51:45	28.00C	34.00%	463	Безпека
2024-06-15 12:51:50	28.00C	34.00%	463	Безпека
2024-06-15 12:51:55	28.00C	34.00%	463	Безпека
2024-06-15 12:52:00	28.00C	34.00%	463	Безпека

Рисунок 3.4.8 – Перегляд даних з датчиків за період

ВИСНОВКИ

У дипломній роботі проведено дослідження проблеми розробки та вдосконалення елементів системи "розумного будинку" з метою підвищення енергоефективності та безпеки житла. В рамках роботи було досягнуто основної мети — спроектувано проста та ефективна система для розумного будинку, яка покращує умови життя, знижує енергозатрати та підвищує рівень безпеки.

Вивчено принципи роботи основних компонентів систем "розумного будинку", таких як датчики освітлення, руху, температури, вологості, та датчики безпеки (полум'я, витік газу).

Проаналізовано існуючі системи управління розумним будинком, що дозволило обрати оптимальні компоненти для розробки.

Спроектовано кілька ключових елементів системи "розумного будинку", включаючи систему освітлення з автоматичним включенням/вимкненням лампочки на основі руху та рівня освітлення, а також систему перевірки безпеки будинку.

Спроектовано і змодельовано інтегровану систему, яка дозволяє керувати умовами проживання через зручний веб-додаток, розроблений за шаблоном MVC на Python Flask.

Проведено експериментальні випробування спроектованої системи, які підтвердили її ефективність та відповідність поставленим вимогам. Система продемонструвала високу надійність у реальних умовах, забезпечуючи стабільну роботу та легке налаштування параметрів.

Переваги спроектованої системи:

- енергоефективність: автоматизація освітлення та інших елементів дозволила значно зменшити енергоспоживання;
- безпека: інтегровані датчики безпеки забезпечують своєчасне виявлення небезпечних ситуацій (пожежа, витік газу), що підвищує загальний рівень безпеки житла;

- економічність: використання доступних компонентів та спрощена архітектура системи дозволили знизити вартість впровадження.

Пропонована система є доступною та легко адаптується під індивідуальні потреби користувачів, що робить її привабливою для широкого кола споживачів.

У подальшому розвитку систем "розумного будинку" варто приділяти увагу удосконаленню алгоритмів автоматизації, підвищенню рівня безпеки даних, а також забезпеченню сумісності з новими пристроями та технологіями. Це дозволить створювати більш інтелектуальні, автономні та зручні у використанні системи, які ще більше покращать якість життя людей.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. HomeKit і розумний дім: інтеграція техніки Apple у розумні системи
https://ti.ua/ua/news/homekit_i_umnyy_dom_integratsiya_tekhniki_apple_v_umnye_sistemy/
2. Google home
<https://support.google.com/chromecast/answer/7071794?hl=co=GENIE.Platform%3DAndroid>
3. Що таке Amazon Alexa <https://ipkey.com.ua/faq/937-amazon-alexa.html>
4. СИСТЕМА РОЗУМНИЙ ДІМ AQARA ВІД XIAOMI: СЕРТИФІКОВАНІ ВЕРСІЇ ЄВРОПЕЙСЬКОГО РЕГІОНУ
<https://www.itbox.ua/ua/blog/Sistema-Rozumniy-Dim-Aqara-vid-Xiaomi-sertifikovani-versiyi-evropeyskogo-regionu/>
5. Що таке захист даних KNX? <https://www.sea.com.ua/ua/istochniki-pitaniya/news/so-take-zahist-danih-knx/>
6. Smart Home Protocols Explained: Wi-Fi vs Bluetooth vs Zigbee vs Z-Wave Vs Thread and Matter <https://www.dfrobot.com/blog-13453.html>
7. Датчик освітленості цифровий GY-302 BH1750FVI
<https://arduino.ua/prod1116-datchik-osveshennosti-cifrovoy-bh1750fvi>
8. Датчик температури і вологості <https://arduino.ua/prod185-datchik-vlajnosti-i-temperatyri-dht11>
9. Піроелектричний інфрачервоний (PIR) датчик руху
<https://arduino.ua/prod193-ik-datchik-dvijeniya-dlya-arduino-hc-sr501>
10. Датчик природного газу MQ-9 <https://arduino.ua/prod1238-modul-datchika-gaza-mq-9>
11. Датчик полум'я KY-026 <https://radiostore.com.ua/ua/p68092321-nabor-datchikov-dlya.html>
12. PostgreSQL: The world's most advanced open source relational database
 [Електронний ресурс] – Режим доступу: <http://www.postgresql.org/>

13. Розробка веб-додатків з використанням Flask фреймворку <https://it-rating.ua/rozrobka-veb-dodatkov-z-vikoristannyam-flask-freymvorku>
14. Arduino Integrated Development Environment (IDE) v1 <https://docs.arduino.cc/software/ide-v1/tutorials/arduino-ide-v1-basics/>