


Вінницький національний технічний університет  
Факультет інтелектуальних інформаційних технологій та автоматизації  
Кафедра автоматизації та інтелектуальних інформаційних технологій

## МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА

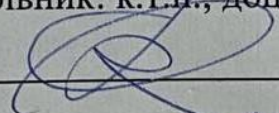
на тему:

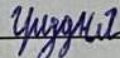
«Автоматизована система управління освітленням в приміщеннях»

Виконав: студент 2-ого курсу, групи 1АКІТ-22м  
спеціальності 151 – «Автоматизація та  
комп'ютерно- інтегровані технології»

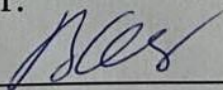
 Павло ГРИЩЕНКО

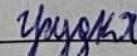
Керівник: к.т.н., доц. каф. АІТ

 Костянтин ОВЧИННИКОВ

« 4 »  2023 р.

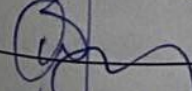
Опонент:

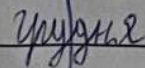
 Димчук Б.О.

« 7 »  2023 р.

Допущено до захисту

Завідувач кафедри АІТ

 д.т.н., проф. Олег БІСІКАЛО

« 11 »  2023 р.

Вінницький національний технічний університет  
Факультет інтелектуальних інформаційних технологій та автоматизації  
Кафедра автоматизації та інтелектуальних інформаційних технологій  
Рівень вищої освіти 2 -й (магістерський)  
Галузь знань 15 – Автоматика та приладобудування  
Спеціальність 151 – Автоматизація та комп'ютерно - інтегровані технології.  
Освітня програма – Інтелектуальні комп'ютерні системи

**ЗАТВЕРДЖУЮ**

Завідувач кафедри АІТ  
д.т.н., проф Олег БІСІКАЛО  
«20» 09 2023р.

**ЗАВДАННЯ**  
**НА МАГІСТРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ**  
Грищенко Павло Сергійович  
(ПІБ автора повністю)

1. Тема роботи: Автоматизована система управління освітленням в приміщеннях.

Керівник роботи: к.т.н., доцент кафедри АІТ Костянтин ОВЧИННИКОВ

Затверджені наказом ВНТУ від «18» 09 2023 року № 247

2. Строк подання студентом роботи «4» 09 2023 року.

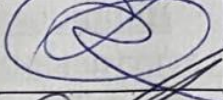
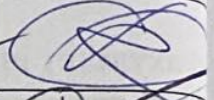

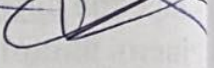
3. Вихідні дані до роботи: прошивка встановлена на плату Arduino; наявність 64Кб флеш пам'яті; 14 цифрових входів; вхідна напруга 7-12В; дані з датчиків що встановлені в підконтрольних приміщеннях; команди що віддаються системі через додаток.

4. Зміст текстової частини: вступ; аналіз систем контролю освітлення в приміщеннях; обґрунтування вибору апаратної частини системи контролю освітлення в приміщеннях; розробка програмного забезпечення центрального контроллера; розробка мобільного додатку для управління системою; тестування розробленого програмного забезпечення; економічний розділ; висновок; список використаних джерел.

5. Перелік ілюстративного (або графічного) матеріалу: схема плати Arduino UNO; Принципова схема Ethernet Shield W5100; Принципова електрична схема одноканального модулю реле; Принципова схема модуля освітленості LM393; Принципова схема датчику HC-SR501; Блок схема, що описує роботу системи.

6. Консультанти розділів роботи магістерської кваліфікаційної роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання	завдання


		видав	прийняв
1-5	Костянтин ОВЧИННИКОВ к.т.н., доцент кафедри АІТ		
6	Володимир КОЗЛОВСЬКИЙ к.е.н., проф. кафе. ЕПтаВМ		

7. Дата видачі завдання « 20 » 09 2023 року

### КАЛЕНДАРНИЙ ПЛАН

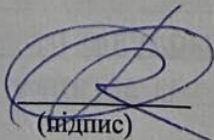
№ з/п	Назва етапів магістерської кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Аналіз систем контролю освітлення в приміщеннях	10.10.2023	Вик.
2	Обґрунтування вибору апаратної частини системи контролю освітлення в приміщеннях	15.10.2023	Вик.
3	Розробка програмного забезпечення центрального контролера	20.10.2023	Вик.
4	Розробка мобільного додатку для управління системою	30.10.2023	Вик.
5	Тестування розробленого програмного забезпечення	05.11.2023	Вик.
6	Економічний розділ	11.11.2023	Вик.
7	Оформлення пояснювальної зписки і графічного матеріалу	15.11.2023	Вик.
8	Попередній захист роботи	21.11.2023	Вик.
9	Остаточний захист роботи	12.12.2023	Вик.

Студент

  
(підпис)

Павло ГРИЩЕНКО

Керівник роботи

  
(підпис)

Костянтин ОВЧИННИКОВ

## АНОТАЦІЯ

УДК 681.51

Гріщенко П.С. Автоматизована система управління освітленням в приміщеннях. Магістерська кваліфікаційна робота зі спеціальності 151 – Автоматизація та комп'ютерно - інтегровані технології, освітня програма – Інтелектуальні комп'ютерні системи управління. Вінниця: ВНТУ, 2023. 110с.

На укр. мові. Бібліогр.: 30 назв; рис.: 34; табл.: 10.

Метою роботи є пошук доступного варіанту автоматизації системи контролю освітлення в приміщенні, який матиме зручне налаштування в поєднанні з високою надійністю та універсальністю. А також оцінка вже існуючих на ринку аналогів, формування основних вимог до таких продуктів.

В рамках роботи проведено дослідження систем контролю освітлення приміщення, проаналізовано недоліки і переваги існуючих продуктів від різних компаній. Визначено необхідний для зручного застосування функціонал.

Спираючись на отримані знання була створена власна концепція автоматизованої системи управління освітленням. Виконано підбір центрального контролера, оптимальних модулів та датчиків, а також розроблена програмна частина для роботи системи.

Ключові слова: освітлення, автоматизація, система управління, датчики, центральний контролер, мобільний додаток.

## **ABSTRACT**

Hrishchenko P.S. Automated indoor lighting control system. Master's qualification work on specialty 151 - Automation and computer-integrated technologies, educational program - Intelligent computer control systems. Vinnytsia: VNTU, 2023. 110c.

In Ukrainian speech Bibliography: 30 titles; Fig.: 34; tab.: 10.

The goal of the work is to find an affordable option for automating the lighting control system in the room, which will have a convenient setting combined with high reliability and versatility. As well as the assessment of analogs already existing on the market, the formation of basic requirements for such products.

As part of the work, a study of room lighting control systems was conducted, the shortcomings and advantages of existing products from various companies were analyzed. The necessary functionality for convenient use has been determined.

Based on the acquired knowledge, an own concept of an automated lighting control system was created. The selection of the central controller, optimal modules and sensors was carried out, as well as the software part for the system operation was developed.

Keywords: lighting, automation, control system, sensors, central controller, mobile application.

## ЗМІСТ

ВСТУП.....	4
1 АНАЛІЗ СИСТЕМ КОНТРОЛЮ ОСВІТЛЕННЯМ В ПРИМІЩЕННЯХ.....	7
1.1 Дослідження поняття системи контролю освітлення та її призначення .....	7
1.2 Аналіз норм освітлення в приміщеннях .....	11
1.3 Огляд існуючих систем контролю освітлення та їх аналіз.....	14
1.4 Висновки до першого розділу.....	19
2 ОБГРУНТУВАННЯ ВИБОРУ АПАРАТНОЇ ЧАСТИНИ СИСТЕМИ КОНТРОЛЮ ОСВІТЛЕННЯ В ПРИМІЩЕННЯХ.....	21
2.1 Обґрунтування вибору центрального контролера.....	21
2.2 Обґрунтування вибору датчиків та модулів.....	26
2.3 Висновки до другого розділу.....	39
3 ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ЦЕНТРАЛЬНОГО КОНТРОЛЛЕРУ .....	40
3.1 Принцип взаємодії елементів системи.....	40
3.2 Розробка програмного забезпечення центрального контролера.....	42
3.3 Застосування хмарного сервісу для роботи з даними .....	47
3.4 Висновки до третього розділу .....	52
4 РОЗРОБКА МОБІЛЬНОГО ДОДАТКУ ДЛЯ УПРАВЛІННЯ СИСТЕМОЮ .....	53
4.1 Обґрунтування вибору операційної системи .....	53
4.2 Вибір середовища для розробки додатку .....	56
4.3 Графічна частина додатку .....	59
4.4 Логіка подачі інформації в інтерфейсі.....	63
4.5 Обґрунтування вибору платформи для поширення додатку.....	68
4.6 Висновки до четвертого розділу.....	69
5. ТЕСТУВАННЯ РОЗРОБЛЕНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ...	70
5.1 Вибір способу тестування програми.....	70
5.2 Розробка тест кейсів .....	72
5.3 Висновок .....	75

6. ЕКОНОМІЧНИЙ РОЗДІЛ.....	76
6.1 Технологічний аудит розробленої автоматизованої системи управління освітленням в приміщеннях.....	76
6.2 Розрахунок витрат на розроблення автоматизованої системи управління освітленням в приміщеннях.....	80
6.3 Розрахунок економічного ефекту від можливої комерціалізації нашої розробки.....	84
ВИСНОВКИ.....	92
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	94
ДОДАТКИ.....	97
Додаток А (обов'язковий) Технічне завдання .....	98
Додаток Б (обов'язковий) ІЛЮСТРАТИВНА ЧАСТИНА .....	102
Додаток В (обов'язковий). Лістинг програми завантаження файлів... ..	109
Додаток Г (обов'язковий). ПРОТОКОЛ ПЕРЕВІРКИ ДИПЛОМНОЇ РОБОТИ НА НАЯВНІСТЬ ТЕКСТОВИХ ЗАПОЗИЧЕНЬ .....	110

## ВСТУП

*Актуальність роботи.* В сучасному світі технологій невпинно розвиваються, породжуючи нові концепції та їх реалізації, з'являються нові можливості для автоматизації колись рутинних процесів. Internet стає все більш доступним, а його швидкість передачі даних зростає. Все це дає поштовх для створення нових пристроїв, що збільшують комфорт та вирішують проблеми економії енергетичних ресурсів [1].

Нещодавно на території України з'явилася катастрофічна нестача електроенергії. Для подолання даної кризисної ситуації, почали масово застосовуватись планові відключення, що дозволяє тримати електромережі в робото спроможному стані, та хоч якось задовольняти потреби населення[2].

Але такий радикальний підхід управління електроенергією не завжди приносить очікувані результати для розподілу та економії цього ресурсу. А також, що не менш важливо при такій експлуатації електромереж, їх складові значно швидше зношується чи навіть ламаються, що призводить до нових аварій. Більш ефективним способом розвантаження електромереж є вимикання не потрібного освітлення та електрообладнання на місцях, що з тих чи інших причин вдається виконувати не усім жителям будинків чи персоналу підприємства [3-5].

Установка системи для управління освітленням є досить вдалим початком для створення сучасної, економної будівлі. Головна ідея даної системи полягає у створенні автономної сітки з освітлювальних приладів, що підвищує комфорт проживання та роботи, а також дозволяють більш ефективно використовувати електроенергію. Хоча існує досить багато готових систем, що реалізують дану ідею, їх доступність обмежена, а експлуатація потребує досить великих знань для установки і обслуговування. Однак широко представленні на ринку плати для автоматизації - Arduino, дозволяють налаштувати систему взаємодії контролерів та різноманітних датчиків на C-подібній мові скетчів без необхідності глибоко вивчення



самого процесу роботи модулів. Для плат Arduino вже створено досить велику кількість готових бібліотек, що забезпечують сумісність та надійну роботу різноманітної електроніки. Тому дипломна робота, що розглядає створення автоматизованої системи управління освітленням на базі плати Arduino є актуальною [6-9].

**Метою** магістерської кваліфікаційної роботи є підвищення економії енергоресурсів під час використання освітлювального обладнання. Для цього було розроблено доступну систему автоматизованого управління освітленням на базі Arduino, яка матиме зручне налаштування та управління за допомогою мобільного додатку. А також забезпечить високий потенціал для модифікації та розширення функціоналу [10-12].

**Об'єктом досліджень** є процес ефективного контролю освітлювального обладнання в житлових приміщеннях.

**Предметом роботи** є система управління та контролю освітлення в приміщеннях.

**Для досягнення мети сформульовані наступні задачі:**

- Провести аналіз готових рішень, що пропонуються на ринку;
- запропонувати власний концепт та основні вимоги до системи контролю освітленням;
- техніко-економічне та науково-технічне обґрунтування апаратної та програмної частин системи;
- обґрунтування вибору необхідного обладнання для створення апаратної частини системи управління освітленням;
- розробити принцип взаємодії приладів в системі та програмне забезпечення для центрального контролера;
- розробити мобільний додаток для зручного управління;
- тестування розробленого програмного забезпечення;

**Новизна** роботи полягає в застосуванні хмарного сервісу для збору та зберігання даних, що були отримані обладнанням в підконтрольних приміщеннях. А також їх передачі для обробки автоматизованій системі

управління освітленням. Даний функціонал для роботи з даними забезпечує надійну синхронізацію різних вузлів системи та за потреби дозволяє відновити знищені на стаціонарному носію дані. Використання хмарного сервіс дозволить організувати управління системою за допомогою додатку на будь-якій кількості Android пристрою [13-14].

**Практична цінність** даної роботи полягає в розробці системи автоматизованого управління освітленням, яка включає в себе апаратну основу та програмне забезпечення для неї. А також додаток, що забезпечить зручне управління.

**Апробація результатів та публікацій.** Дана робота була апробована у доповіді на науково-технічній конференції факультету інтелектуальних інформаційних технологій та автоматизації Вінницького національного технічного університету у 2023 році [1].

Наукові дослідження за темою магістерської кваліфікаційної роботи будуть проводитись на основі індивідуального завдання, складеного та затвердженого кафедрою автоматизації та інтелектуальних інформаційних технологій ВНТУ, а також розробленого технічного завдання на науково-дослідну роботу.

# 1 АНАЛІЗ СИСТЕМ КОНТРОЛЮ ОСВІТЛЕННЯМ В ПРИМІЩЕННЯХ

## 1.1 Дослідження поняття системи контролю освітлення та її призначення

У кожному сучасному будинку в тій чи й іншій мірі функціонує освітлювальне обладнання, що забезпечує комфорт і створює зручне робоче середовище. Інтеграція такого обладнання в єдину систему управління освітленням, дає можливість злагодженої роботи, збільшуючи тим самим ефективність і економічність кожного пристрою.

Система управління освітленням — це широкий комплекс різних пристроїв, які забезпечують освітлювальному обладнанню автономну роботу або можливість дистанційного управління. Це можна використати, щоб на відстані змінювати параметри освітлення в підконтрольному приміщенні, замість того, щоб застосовувати настінний вимикач чи панель управління, можна просто використати планшет або особистий смартфон на якому встановлений відповідний додаток. Для забезпечення автономної роботи системи освітлення, необхідні параметри часто контролюють за допомогою певних датчиків, що і збирають дані. Зазвичай, встановлюють датчики руху, датчиком рівня освітленості кімнати.

Основним завданням систем управління освітленням є підвищення ефективності приладів освітлення та економія електроенергії, також вони дозволяють користувачу забути про рутинні процеси включання, виключання світла в будинку та на робочому місці, тай в цілому гарантують додатковий комфорт, підвищивши експлуатаційні характеристик підконтрольної будівлі чи окремої кімнати. Система управління освітленням часто має функції автоматизованої чи навіть автономної роботи, які можуть без втручання власника підлаштовуватися під навколишню ситуацію, чи змінювати свої

налаштування, щоб зуміти правильно організувати освітлення контрольованого приміщення [2]. Автоматизовані рішення даних систем, як мінімум, повинні мати відповідний набір функціоналу для управління освітлювальним обладнанням:

- аналіз рівня поточного освітлення в приміщенні,
- можливість визначити день і ніч,
- перевірка присутності людини в підконтрольній зоні.

Сучасні запити до ринків освітлювального обладнання, вимагають тотальне переорієнтування з аналогового управління на цифрове. Основним плюсом цифрових рішень є зручність комунікації для функціонування в одній екосистемі з іншими пристроями будинку, побудова надійних зав'язків між окремими елементами, що об'єднуються в цілісну структуру. Зазвичай для створення зав'язків не використовують окремі дроти, оскільки передачу можна організувати за допомогою силових кабелів. Часто для цього не потрібні окремі дроти, а передача здійснюється за допомогою силових кабелів. Також все популярнішими стають рішення з використанням бездротових технологій.

Обладнання що контролює освітлення в сучасних будинках базується на кількох компонентах: автоматизованих системах керування та освітлювальному приладді. Щоб підвищити ефективність LED-освітлення в приміщенні потрібно використати ряд технологій [16].

1) Стандартний RGB-контролер. Даний пристрій є одно каналним і здатний управляти трьома кольорами, а також має ряд програм для управління, контроль якими зазвичай реалізують за допомогою додатків на мобільних пристроях або панелі управління.

2) Багатоканальний RGB-контролер. Має як мінімум три канали за допомогою яких здійснюється управління кольорами, їх можна об'єднати для побудови системи з RGB сценаріїв. Запрограмувати контролер можна за допомогою персонального комп'ютера.

3) DMX RGB-контролер. За допомогою даного пристрою здійснюється управління освітленням з використанням протоколу DMX 512, він підключається до комп'ютера через USB роз'єм. Даний протокол дає можливість підключити прилади до 512 доступних каналів, застосувавши даний функціонал, можна створити ряд світлових сценаріїв.

4) DALI RGB-контролер. Рішення, що базується на протоколі DALI та двох направленому обміні складовими системи. Перелік адрес, що застосовуються в рамках системи перевищують 12000, що за допомогою роутерів поєднуються в 200 окремих ліній.

Автоматизовані системи контролю освітлення необхідно комбінувати з різними датчиками та модулями, що додають чи розширюють функціонал, надаючи нові можливості власнику. Приклад системи контролю освітленням з модулями на базі протоколу DALI показує рис. 1.1.

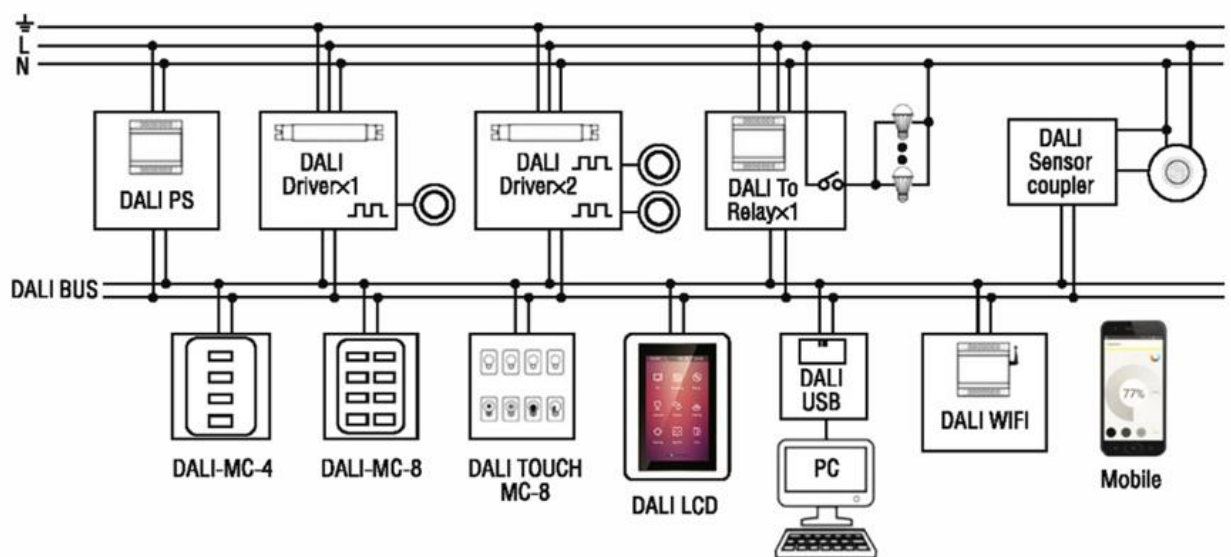


Рисунок 1.1 – Приклад системи контролю освітленням з модулями на базі протоколу DALI

Автоматичне управління освітлювальними приладами надає багато можливостей, таких як, імітування присутності власників в приміщенні,

алгоритми можуть самостійно знизити потужність роботи освітлення, якщо працює екран телевізора. Також, така система може бути запрограмованою, аби використовуватись як будильник, сигналізувати про надходження дзвінку чи повідомлення. Лампи в таких системах можуть загоратись самостійно, якщо користувач повернувся в будинок, та підкоряться голосовим командам власників [13].

Поєднавши систему контролю освітлення з набором сенсорів руху, можна налаштувати економне функціонування світильників у коридорі чи ванній кімнаті. Така збірка зможе суттєво збільшити комфорт та зменшить витрати на електроенергію, та підвищить ресурс роботи освітлювальних приладів. Датчики, що фіксують рух не допускають роботу лампочок в пустому приміщенні, оскільки автоматично вимикають їх, коли не може зафіксувати присутність людей. Крім цього, такий функціонал системи позбавить необхідності знаходити в п'яті перемикач – користувачу достатньо тільки увійти в кімнату або вийти на майданчик між сходами, а освітлення вже працює [17].

Спільна робота бездротової передачі даних та енергозберігаючих ламп в одній системі, може змінити не лише повсякденний уклад життя, але й дозволить створити чудовий інтер'єр в будинку. Найважливішим плюсом системи розумного освітлення є функціонал для створення набору світлових сценаріїв та ефектів. За допомогою них можна легко організувати дизайнерське оформлення для різноманітних закладів відпочинку(парків, торгових центрів, ресторанів).

В подібних проектах часто використовують різноманітні RGB світильники, в яких змінюється підсідка та переливаються тіні. Такі прилади, переважно, застосовуються комплексно, аби створити максимально вражаючі ефекти і зацікавити відвідувача.

Ще одним важливим елементом, що входить в систему розумного освітлення – світильники з кольором температури, який підлаштовується. Таке обладнання часто використовують для організації зручних умов роботи

тчи відпочинку. Підключивши такі лампи до інтелектуальної системи, отримаємо ідеальне для роботи яскраве світло та комфортне м'яке для дозвілля. Коли виникли LED-технології, управління світлодіодами стало простим і приємним процесом.

Частка електроенергії, що витрачається для освітлення є досить вагомою, тому розумне використання освітлювальних приладів може суттєво зменшити такі витрати, при цьому не доведеться терпіти незручності. Вирішити це завдання дозволяють автоматизовані системи управління освітленням, які часто інтегрують в обладнання для сучасного житла.

## **1.2 Аналіз норм освітлення в приміщеннях**

Правильне світло — річ надзвичайно важлива, адже кількість світла може вплинути на атмосферу в приміщенні: створити настрій для роботи й активної мозкової діяльності чи, навпаки, налаштувати на релакс та сон. Навіть хороше самопочуття залежить від рівня освітленості. Місця, де люди проводять найбільше часу — це дім та робота, і для того, щоб почуватися комфортно та впевнено на робочому місці та бути максимально продуктивним, а опісля, розслабитися і відпочити у власному будинку, варто провести правильний розрахунок освітлення приміщення, виважено підійти до вибору ламп та світильників та ще й розташувати їх так, щоб знайти золоту середину рівня освітлення [26].

Затишна атмосфера вдома чи добре освітлена робоча зона в офісі створюються за допомогою визначення коефіцієнту використання світлового потоку та правильного розрахунку освітлення приміщення. Саме від того якої цілі потрібно досягти — створити якісні умови праці, провівши правильний розрахунок освітлення виробничого приміщення чи забезпечити затишок вдома — і варто сконцентрувати свою увагу перед тим, як перейти до організації освітлення приміщення.

Основними технічними аспектами організації освітлення є:

- Вибір колірної температури джерел світла.
- Розмір кімнати та висота стелі;
- Потужність і світловий потік ламп, а також кількість світильників;
- Загальні рекомендації по освітленні різних типів кімнат.

Розглянемо деякі з цих аспектів детальніше.

Виділяють теплу колірну температуру - приємне жовте світло, що розслабляє і не створює зайвого навантаження на очі. Таке освітлення добре підійде для житлових приміщень, а також дитячих садочків, кафе, барів та ресторанів.

Прохолодна колірна температура - світло нейтрального кольору, досягти якого можна завдяки використанню світильників зі світлодіодними лампами. Саме це світло визначають сьогодні як таке, що сприяє найбільшій продуктивності. Таке освітлення можна використовувати для всіх приміщень, в яких ви проводите багато часу [24].

Холодне світло (його ще називають "денним") - має виражений синій колір і може здаватися суворим і неприємним. Його можна використовувати там, де потрібна висока концентрація уваги на короткотривалій термін, тобто для освітлення виробничих приміщень або там, де потрібно створити відчуття чистоти (ванна кімната, кухня, лікарняна палата, виробничі лінії промисловості). Холодний тон приміщенню надають флуоресцентні лампочки (використовують на 75% менше енергії, ніж лампи розжарювання), однак, слід враховувати, що вони містять ртуть і вимагають утилізації належним чином.

Не менш важливим фактором, коли проводиться розрахунок освітлення приміщення, є правильний вибір потужності лампочок, який визначається в залежності від того скільки світла отримує кожна частина кімнати. Цю характеристику зазвичай вказують на пакуванні, так само як і номінальну напругу. Тобто, якщо на лампочці написано 60/220, це означатиме, що при підключенні до мережі 220 потужність лампи буде 60. Це особливо важливо пам'ятати, щоб зробити правильний розрахунок освітленості.



Також, досить важливо при виборі світильників враховувати стандарти освітленості для різних кімнат:

- Вітальня: крім штучного, використовуйте підсвічування в одному куті кімнати. Зосередьтеся на об'єкті, наприклад, на творі мистецтва, цікавому комоді або стільці, адже правильне використання світлового потоку допоможе розставити акценти та підкреслити оригінальні рішення в дизайні.
- Кухня: світлодіодні лампи допомагають освітити робочу зону. Раковина також, як одна з найчастіше використовуваних робочих поверхонь, є гарним місцем для додавання освітлення.
- Загальноприйнято мати лампи у спальні на тумбочках. Рекомендовано спрямовувати будь-яке світло трохи далі від ліжка.
- Ванна кімната. Вам потрібно освітити дзеркало, але накладне світло може створювати тіні. Подумайте про освітлення дзеркала з обох боків. Потім використовуйте штучне освітлення, щоб повністю освітлити кімнату.

Таблиця 1.1 – Таблиці з усередненим значенням освітленості для різних типів приміщень

Норма освітленості кухні	150 ЛК
Вітальні	450 ЛК
Дитячої кімнати	200 ЛК
Кабінету	300 ЛК
Ванної кімнати	50-75 ЛК

### 1.3 Огляд існуючих систем контролю освітлення та їх аналіз

Розглянемо кілька прикладів таких систем, проаналізуємо їх переваги та недоліки:

Компанія SEA пропонує модулі і пристрої для створення інтелектуальної системи управління освітленням. Зокрема пристрої на протоколі DALI, які можуть вирішити ряд задач різного плану, наприклад організація керування освітленням в житлових приміщеннях, робочих зон, заводських або торгових будівлях [20]. Функціональність та легкість технології дозволяє швидко впроваджувати такі систем по всьому світі.

Ключові переваги даної системи контролю освітленням:

- Використано відкритий протоколом, який доступний всім розробникам та виробникам;
- Протокол DALI спеціально розроблявся для контролю освітленням, тому система SEA справляється з даною задачею набагато успішніше, порівняно з більшістю аналогів на ринку, при цьому пропонується по доступнішій ціні;
- Система базується на основі децентралізованої шини, це означає, що їй не потрібний центральний контролер, а також в ній може бути організована будь-яка топологія кабельної мережі, в тому числі і змішана;
- Щоб з формувати шину зв'язку всіх елементів однієї DALI-мережі знадобляться лише два кабелю а також не доведеться дотримуватись полярності підключення;
- Використання стандарту DALI, дає ряд переваг підчас монтування і налаштування системи;
- Система дозволяє вмикати чи вимикати освітлювальне обладнання, налаштовувати інтенсивність або колірну температуру випромінювати ламп для відповідних сцен освітлення чи світлових сценаріїв;

- Систему можна налаштувати за допомогою цілого ряду доступних програмних прошивок, залежно від потреб замовника, без необхідності вносити суттєві зміни в апаратну базу;

- Ще однією вагомою перевагою DALI, є можливість використовувати двосторонній комунікаційний протокол для збору даних про стан системи і обладнання, а також відслідковувати помилки чи несправності в роботі світильників;

- Використання в системі інтелектуальних технологій, дозволяє суттєво економити електроенергію та ресурс роботи лампочок.

Фізичний рівень системи контролю приставлений двопровідною шиною DALI, яку можна провести спільно з силовими лініями, це можна зробити навіть всередині одного кабелю, одним з варіантів кабелів, що можуть бути використані є стандартний 5-жильний кабель марки NYM в неметалевому екрані. Для якісного функціонування лінії, що передає дані, шина DALI потребує підключення до джерела живлення 16 В постійного струму. Межі робочої напруги шини знаходяться в діапазоні від 9.5 до 22.5 В, при струмі, що не має перевищувати показник 250 мА. Логічним «0» є діапазон напруги в рамках від -6.5 В до +6.5 В, а логічною «1» відповідно є діапазон напруги в рамках від 9.5 В ... 22.5 В. Діапазон від +6.5 В до +9.5 В непозначений, як корисний сигнал. Швидкість передачі даних по шині становить 1200 бод. Довжина всього кабелю залежить від падіння напруги вздовж лінії DALI, яке не має перевищити показник в 2 В. Від цих показників і залежить максимальна довжина кабелю з площею перетину 1.5/0.75/0.5 мм<sup>2</sup> і становить відповідно 300/150/100 м.

На рис. 1.2 наведено приклад системи, що пропонуються компанією SEA.

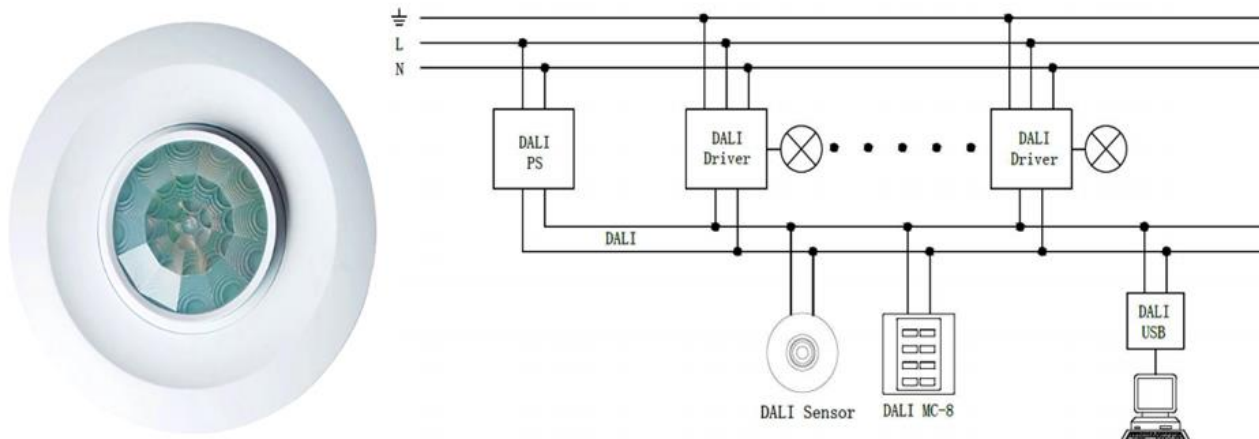


Рисунок 1.2 – Приклад системи контролю освітлення від компанії SEA

В проектах на базі DALI доступний ряд налаштувань, які не зникають навіть при довготривалому зникненні живлення в системі:

- сцени для освітлювального обладнання (максимум доступно 16 сцен);
- приналежність до груп (максимум доступно 16 груп);
- індивідуальну адресацію (доступно максимум 64 драйвера);
- виставлені параметри для регулювання світла;
- яскравість освітлення, навіть якщо перервані лінії управління, таке може статись в аварійному режимі;
- автоматичне відновлення налаштувань яскравість освітлення після тимчасового зникнення електропостачання;
- налаштування параметрів мінімального та максимального рівня інтенсивності освітлення.

Просте рішення від компанії SEA може в себе включати один DALI-драйвера та одне джерела живлення, що працюють з сильно обмеженим набором датчиків та кнопок DALI (див. рис. 1.3).

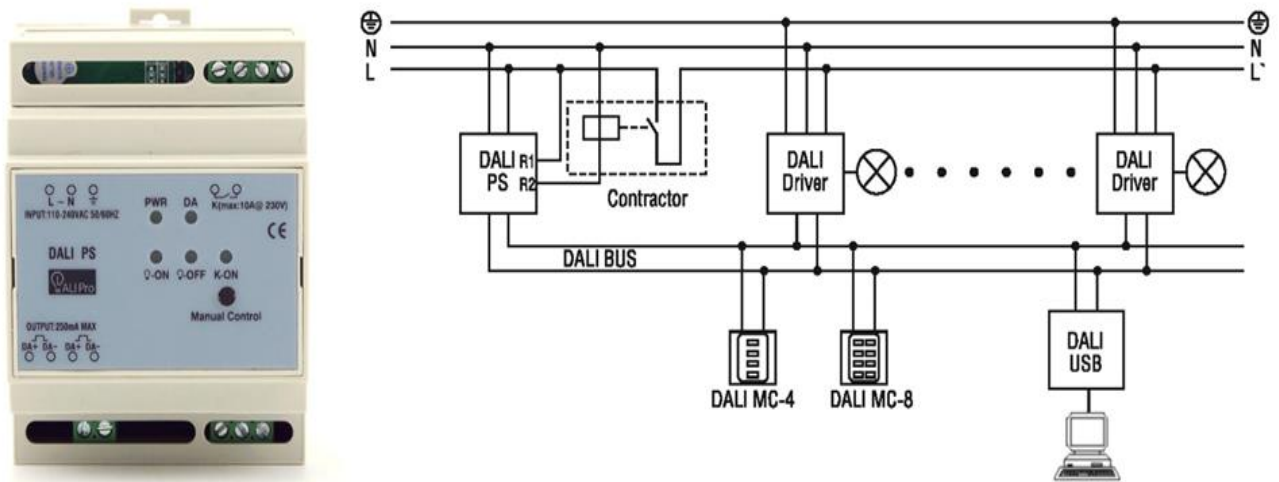


Рисунок 1.3 – Блок живлення шини

Система контролю освітлення від компанії Tervix - це також основа для розумного будинку, що направлена на автоматизацію системи освітлення в приміщенні. Приклад системи наведено на рисунку 1.4. Для роботи використовується енергоефективний протокол передачі даних ZigBee [3]. Після налаштування вся система може без проблем функціонувати автономно, незалежно від підключення до неї мережі WiFi та Інтернету. Розумний вимикач Tervix ZigBee встановлюється в підрозеточну коробку, разом з звичайний вимикач або у розподільчий щиток на DIN рейку. Він може працювати незалежно від звичайного вимикачем, при цьому вони обидва виконують свої функції, це забезпечує дублювання ручного та автоматичного вмикання/вимикання освітлення. Оскільки в системі присутні датчики освітлення, вона може відрізнити темну частину доби та світлу в залежності від природного освітлення на вулиці, якщо там темрява то система включає штучне освітлення, а коли наступить ранок, то освітлювальне обладнання вимкнеться. Також можна налаштувати, щоб світло горіло лише до першої години ночі, після чого відключалось. Якщо поставити датчики руху, то при наявності людини обладнання самостійно увімкнеться. Також світло можна включити/виключити застосувавши мобільний додаток, що пропонується

разом з системою, або завдяки голосу (Google Assistant.). Система працює з APP Tuuya Smart, що є найбільшим у світі провайдером IoT пристроїв.

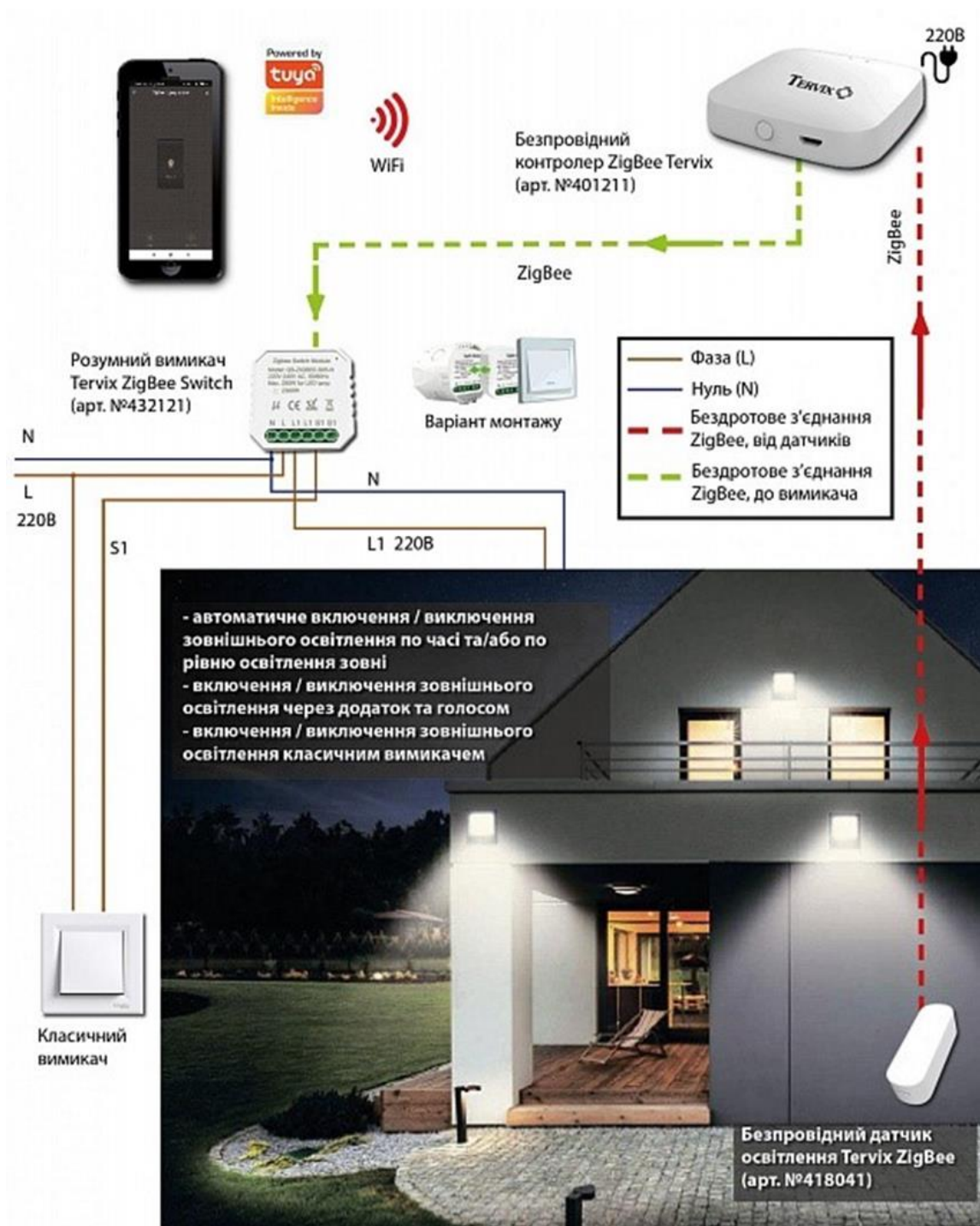


Рисунок 1.4 – Система контролю освітлення від компанії Tervix

Основні переваги [4]:

- частина розумного будинку, що легко масштабується
- система працює по протоколу ZigBee та WiFi
- контролер - ZigBee + WiFi
- сучасний дизайн складових системи
- управління - додаток TuYa / Smart Life
- створення Smart Scene
- детальна інструкція українською мовою
- управління за допомогою смартфона з будь-якої точки світу, де є інтернет
- після налаштування, система здатна працювати автономно (локально) – без інтернету та WiFi
- дублювання управління освітленням - вручну, через додаток, автоматично, голосом (Google Assistant, Alexa)
- можливість роботи у парі зі звичайним вимикачем
- гарантія - 2 роки

#### **1.4 Висновки до першого розділу**

Проведений аналіз показує, що сьогодні є багато способів організації системи освітлення, як за допомогою готових рішень від різноманітних компаній так і за допомогою плат керування, таких як: Raspberry Pi, NooLite, та інші. Ці плати дають можливість побудувати управління освітленням за допомогою модулів різних типів, таких як сенсори руху, що фіксуючи присутність людини, дозволяючи автоматично вмикати або вимикати освітлювальні пристрої, та датчиків навколишнього освітлення, що підлаштовують потужності штучного освітлення під поточний рівень.

Існує декілька варіантів організації зв'язку між компонентам в системах контролю освітленням, ці варіанти можна умовно розділити на 2 типи: дротове та бездротове з'єднання. До дротового з'єднання відноситься

стандарт RS-485, що працює за допомогою витії пари, а також з'єднання можна організувати за допомогою кабелю UTP 5. Бездротова організація системи адаптивного контролю може бути створена за допомогою радіозв'язку, наприклад завдяки модулю nRF24L01 або Wi-Fi. Зазвичай один тип з'єднання не може повністю забезпечити зв'язок між всіма елементами в системі, тому найкращим варіантом буде комбінація з кількох видів радіо - та Wi-Fi технологій.



## **2 ОБГРУНТУВАННЯ ВИБОРУ АПАРАТНОЇ ЧАСТИНИ СИСТЕМИ КОНТРОЛЮ ОСВІТЛЕННЯ В ПРИМІЩЕННЯХ**

Початком для створення автоматизованої системи управління освітлення є вибір апаратного забезпечення. Центральний контролер є основою такої системи і відіграє управляючу роль. Функціонал системи забезпечують датчики та модулі, що збирають та передають дані для обробки. Дуже важливо обрати найбільш якісне та доступне обладнання. Добре організовану технічну частину легше розширити чи запрограмувати в подальшому.

### **2.1 Обґрунтування вибору центрального контролера**

Як основа для апаратної системи був обраний Arduino UNO+WiFi R3, так як це доступний центральний контролер з відкритою архітектурою, він може зчитувати набір вхідних даних за допомогою своїх аналогових контактів у вигляді напруг.

При під'єднанні необхідних для системи датчиків до відповідних входів, вони зможуть програмними засобами отримувати потік інформації з контактів [18].

Arduino UNO+WiFi R3 чудово підходить для побудови різноманітних розробок, які функціонують по заданому завчано алгоритмі, маючи можливість відповідати на сигнали з зовнішніх пристроїв. Плата Arduino UNO+WiFi R3 має відкриту принципову схему, яку можна за потреби змінювати під власні потреби, вона допускає підключення сторонії пристроїв до системи.

Плата Arduino UNO+WiFi R3 наведена на рисунку 2.1



Рисунок 2.1 –Плата Arduino UNO+WiFi R3

Основою для Arduino Uno слугує ATmega328. На платі наявні 14 цифрові входи з яких лише шість можна застосувати в якості ШІМ виходів, також присутні шість аналогових входів, один кварцовий генератор 16 МГц та USB роз'єм, силове гніздо, а також кнопка для перезавантаження системи. Щоб почати роботу потрібно під'єднати платформу до персонального комп'ютера через USB роз'єм, або заживити систему через акумулятор.

Важливою перевагою моделі Arduino Uno є низька вартість, а також відносно проста експлуатація, вона серед всіх платформ найбільше підходить, щоб самостійно організувати невелику систему, це рішення часто користується популярністю в конструкторів, які нещодавно зайнялись програмуванням подібних плати та побудовою автоматичних систем.

Базові дані про плату Arduino Uno наведено в таблиці 2.1.

Таблиця 2.1 – Базові характеристики плати Arduino Uno

Характеристики	Значення
Мікроконтролер	Atmega328
Робоча напруга	5 В
Вхідна напруга (рекомендована)	7-12 В
Вхідна напруга (гранична)	6-20 В
Цифрові входи/виходи	14 (6 з яких можуть використовуватися як виходи ШІМ)
Аналогові входи	6
Постійний струм через вхід/вихід	40 мА
Постійний струм для виводу 3.3 В	50 мА
Флеш-пам'ять	32 Кб (Atmega328) з яких 0.5 Кб використовується для завантажувача
ОЗП	2 Кб (Atmega328)
EEPROM	1 Кб (Atmega328)
Тактова частота	16 МГц

АТmega328 виготовляється з вже вбудованим завантажувачем, таке рішення зрощує встановлення програмного забезпечення на мікроконтролер, тому зовнішній програматор є не потрібним. З'єднання забезпечується авторським протоколом STK500.

Також залити програмне забезпечення на мікроконтролер можна через висновки блоку ICSP (внутрішньо схемного програмування).

Плата Arduino Uno створена з розрахунком, що перед записом нового програмного забезпечення перезавантаження системи ініціює додаток Arduino на комп'ютері, тому натисканням спеціальної кнопки на платі не потрібне. Одна з ліній DTR мікросхеми АТmega8U2, керуючих потоком даних (DTR), підключена до виводу перезавантаження мікроконтролера АТmega328 через 100 нФ конденсатор. При активації цієї лінії, тобто під час передачі сигналу низького рівня відбувається перевантаження системи. При

роботі з платою Arduino, можна застосувати дану функцію, щоб натиснувши лише кнопку Upload в середовищі програмування Arduino, залити ПО в систему. Передача сигналу низького рівня по лінії DTR координується з початком запису ПО, що помітно скорочує час завантаження.

Корисною особливістю плати Arduino UNO+WiFi R3 є наявність інтегровано Wi-Fi модулю на ESP8266 з 8Mb flash пам'яті. Даний модуль може працювати спільно з платою Arduino Uno так і абсолютно незалежно. Такий функціонал забезпечує зв'язок рішення з інфраструктурою Wi-Fi.

На платах Uno наявний функціонал для відключення лінії автоматичного перезавантаження, розірвавши з'єднання лінії. Щоб відновити лінію потрібно просто з'єднати контакти на мікросхемі з двох кінців і сигнал запрацює знову. На лінії наявне маркування RESET-EN. Вимкнути автономне перезавантаження можна під'єднавши резистор 110 Ом між джерелом 5 В та потрібною лінією. Організація системи управління наведена на рисунку 2.2.

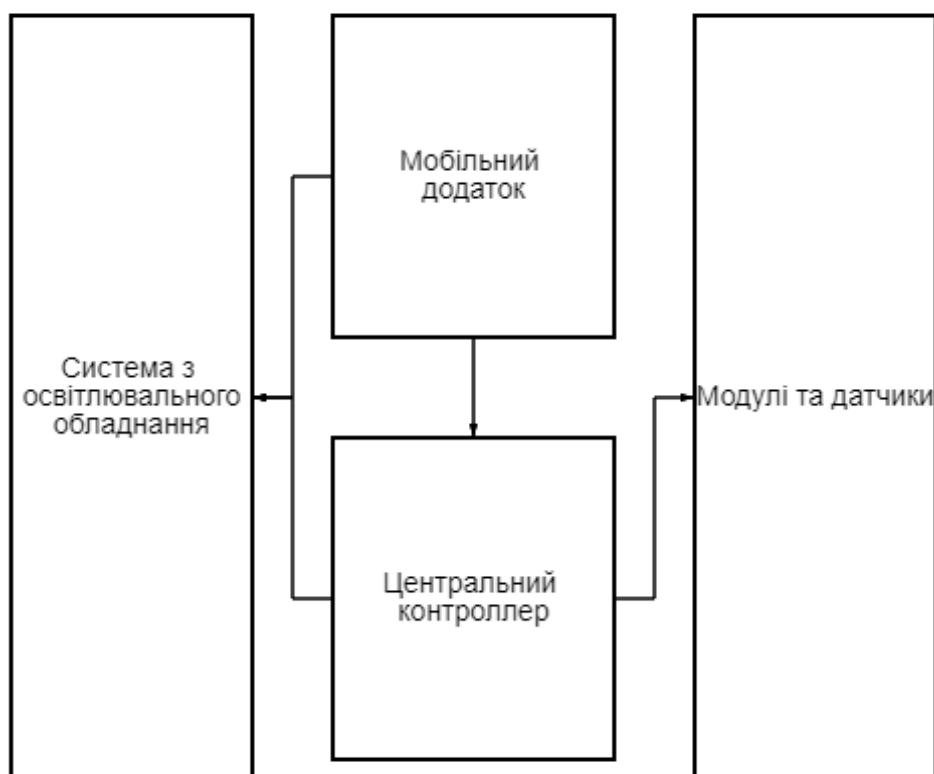


Рисунок 2.2 – Організація системи управління

Опис структури системи:

Рівень 1: Користувацький пристрій на який завантажується ПО для управління системою та відображення даних з датчиків, це може бути будь який мобільний гаджет з операційною системою Android.

Рівень 2: Контролер Arduino, який відповідає за збір та обробку даних з датчиків, а також здійснюють контроль освітлювального обладнання.

Рівень 3: Периферія, що проводить збір показників з навколишнього середовища. В автоматизованій системі управління освітлення на базі Arduino, дану функцію виконує набір датчиків та модулів, що були підібрані під конкретні задачі.

Датчики монтується в приміщеннях, освітлення в яких має контролювати система. Дані встановлених датчиків потрапляють на центральний контролер, який їх обробляє, та відштовхуючись від результатів проводить регуляцію підконтрольного обладнання. Інформація про стан освітлення передається на контролер через певний період часу або по запиту користувача.

Найпростішим способом отримати дані з Arduino, можна назвати підключення цієї плати через USB гніздо до персонального комп'ютеру, він зможе зчитати дані, як ніби їх передають по COM - порту. Також Arduino зможе функціонувати незалежно від персонального комп'ютера, якщо наявні джерела живлення, а також альтернативний канал для комунікації. Arduino підтримує підключення плат розширення (Shield).

## 2.2 Обґрунтування вибору датчиків та модулів

Для створення системи контролю освітлення крім контролера також знадобляться додаткові датчики та модулі. З'єднання мікроконтролера та локальної мережі забезпечить плата розширення W5100 Ethernet Shield з функціоналом для взаємодії з картами пам'яті microSD. Її зовнішній вигляд було наведено на рисунку 2.3. та рисунку 2.4.

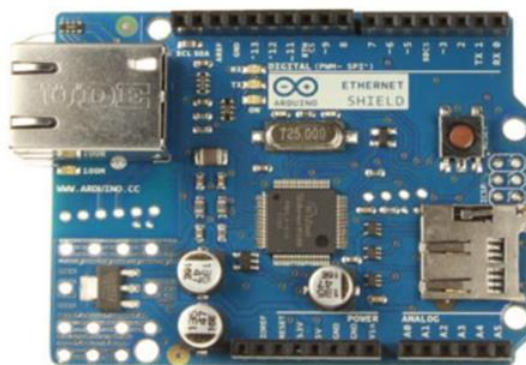


Рисунок 2.3 – Плата розширення W5100 Ethernet Shield вид зверху

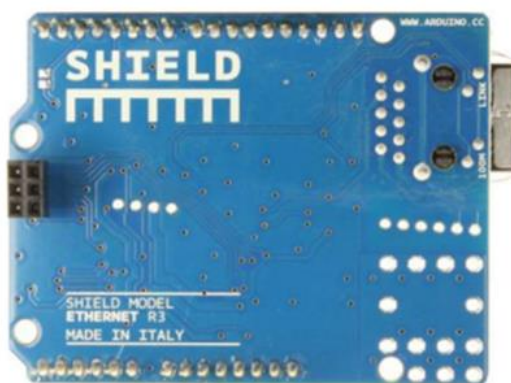


Рисунок 2.4 – Плата розширення W5100 Ethernet Shield вид знизу

Характеристики [16]:

- Робоча напруга 5В (подається безпосередньо від плати Arduino)
- Ethernet-контролер W5100 має вбудований буфером на 16 КБ
- Швидкість з'єднання 10/100 Мбіт
- Взаємодія з Ардуіно здійснюється через інтерфейс SPI

Так як файли з налаштуваннями ми будемо зберігати на карті пам'яті SD, то нам буде потрібен модуль SD Card Reader для роботи з нею. Зовнішній вигляд такого модуля наведений на рис. 2.5.

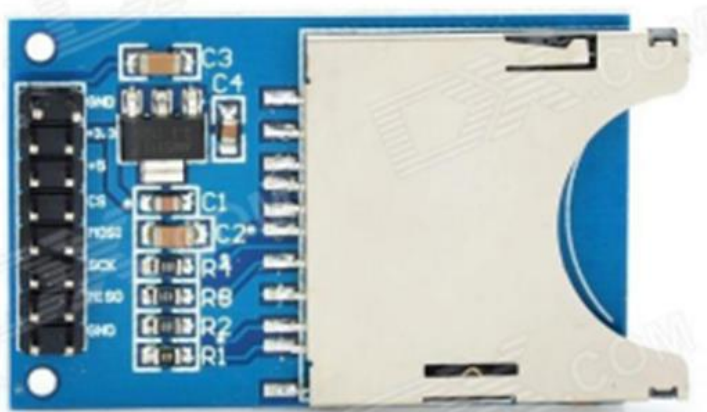


Рисунок 2.5 – SD Card Reader

Один з найбільш корисних модулів, що знадобиться для Arduino Uno, це елемент, що управляє напругою 230В. Arduino не може працювати з приладами освітлення, що працюють на високій напрузі без такого модуля, для забезпечення такого функціоналу було обрано реле 5В. Такі одноканальні та двоканальні модулі будуть використані в автоматизованій системі керування освітлення для керування різними типами навантажень. Вигляд модуля наведено на рисунок 2.6.



Рисунок 2.6 – Одноканальний модуль реле

Технічні характеристики:

- Напруга живлення: 5/12 В
- Струм: 15 мА ... 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



В конструкцію модуля входить одне реле фірми SONGLE модель якого SRD-05VDC-SL-C або SRD-05VDC-SL-C. Щоб швидко підключити модуль реле використовується дві групи контактів, слабкострумова з кроком 2,54 мм та силова група.

Що до принципу роботи модулю, після підключення до джерела живлення 5В, повинний загорітися червоний діод, але реле залишиться в незмінному стані. Щоб переключити модуль в інше положення, потрібно висновок IN під'єднати до землі, це здійснюється за допомогою контролера Arduino або просто замкнути IN1 з GND. Даний модуль не передбачає гальванічної розв'язки, висновок IN підключений напряму до керуючого транзистора, принципову схему одно каналного реле, наведено на рисунок 2.7.

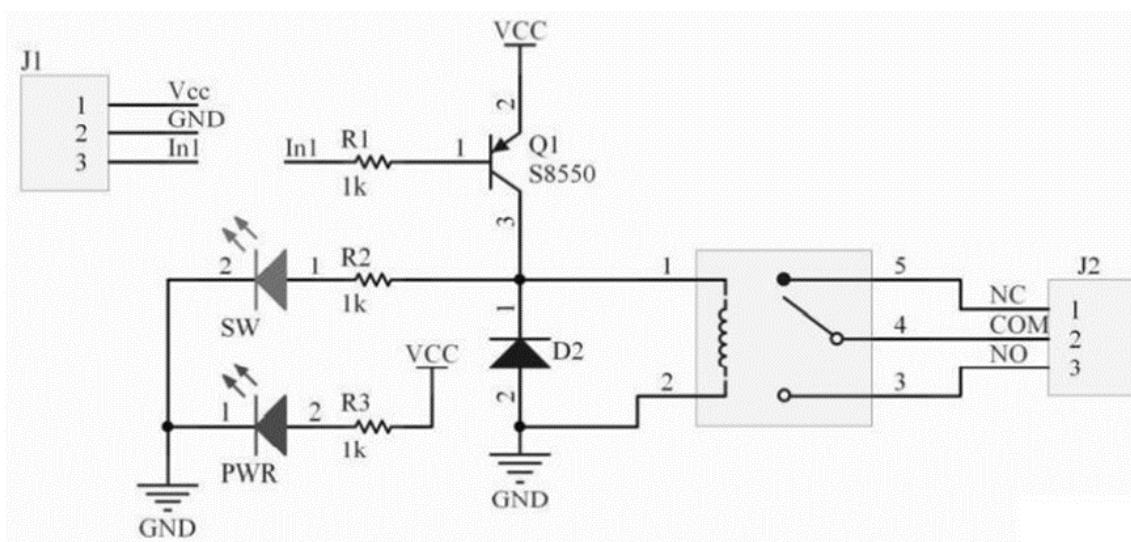


Рисунок 2.7 – Принципова електрична схема одноканального модулю реле

Модуль виміру освітленості на LM393, можна застосувати в різноманітних пристроях для вимірювання інтенсивності світла, наприклад для розширення функціоналу автоматизованої системи управління світла, або в роботах для розрізнення дня і ночі. Збір даних відбувається завдяки світлочутливому елементу (фоторезистора), в якому змінюється опір в

залежності від впливу на нього світла. Вигляд модуля наведено на рисунку 2.8.

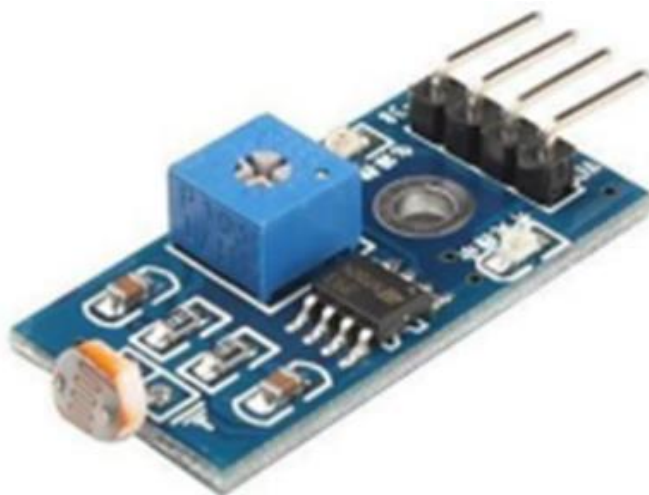


Рисунок 2.8 – Світлочутливий модуль датчика освітленості RSK205502

Даний модуль має чотири виводи з два вихідними контактами, аналоговий та цифровий і два контакти, щоб підключити живлення. Щоб зчитати аналоговий сигнал в модулі передбачений окремий вивід «АО», з нього можна взяти показання напруги з 0 В ... 3.3 В або 5 В залежно від обраного для системи джерела струму. Цифровий вихід DO, встановлюється в лог "0" або лог «1», залежно від яскравості світла, чутливість, можна змінювати завдяки поворотному потенціометру. Цифровий вихід, може видати більше 15 мА, це сильно спрощує застосування модуля, а також дає можливість для використання його минаючи плату Arduino, при цьому під'єднуючи його напряму до роз'єму одноканального реле або одного з входів двоканального реле [13]. Принципову схему датчика освітлювання на LM393 з 4 pin, показана на рисунку 2.9, дані про чутливість наведені в таблиці 2.2.

Таблиця 2.2 – Характеристики чутливості модулю освітленості

Освітлення	Опір фоторезистора	Приклад
0,1 лк	600 кОм	Безмісячна ніч
1 лк	70 кОм	Місячна ніч
10 лк	10 кОм	Темна кімната
100 лк	1,5 кОм	Дуже похмурий день
1000 лк	300 Ом	Похмурий день

Технічні характеристики модуля RСК205502:

- Напруга живлення: +3.3 В ~ +5.5 В
- Струм: 10 мА
- Формат сигналу цифрового виходу: TTL (0/1)
- Рівень сигналу аналогового виходу: 0..Vcc
- Підключається безпосередньо до мікроконтролера
- Робоча температура: від 0 °С ~ + 70 °С
- Розміри: 45 x 14 x 7 мм

Позначення виводів датчика:

- Vcc - плюс живлення
- GND - мінус живлення
- DO - цифровий вихід
- AO - аналоговий вихід

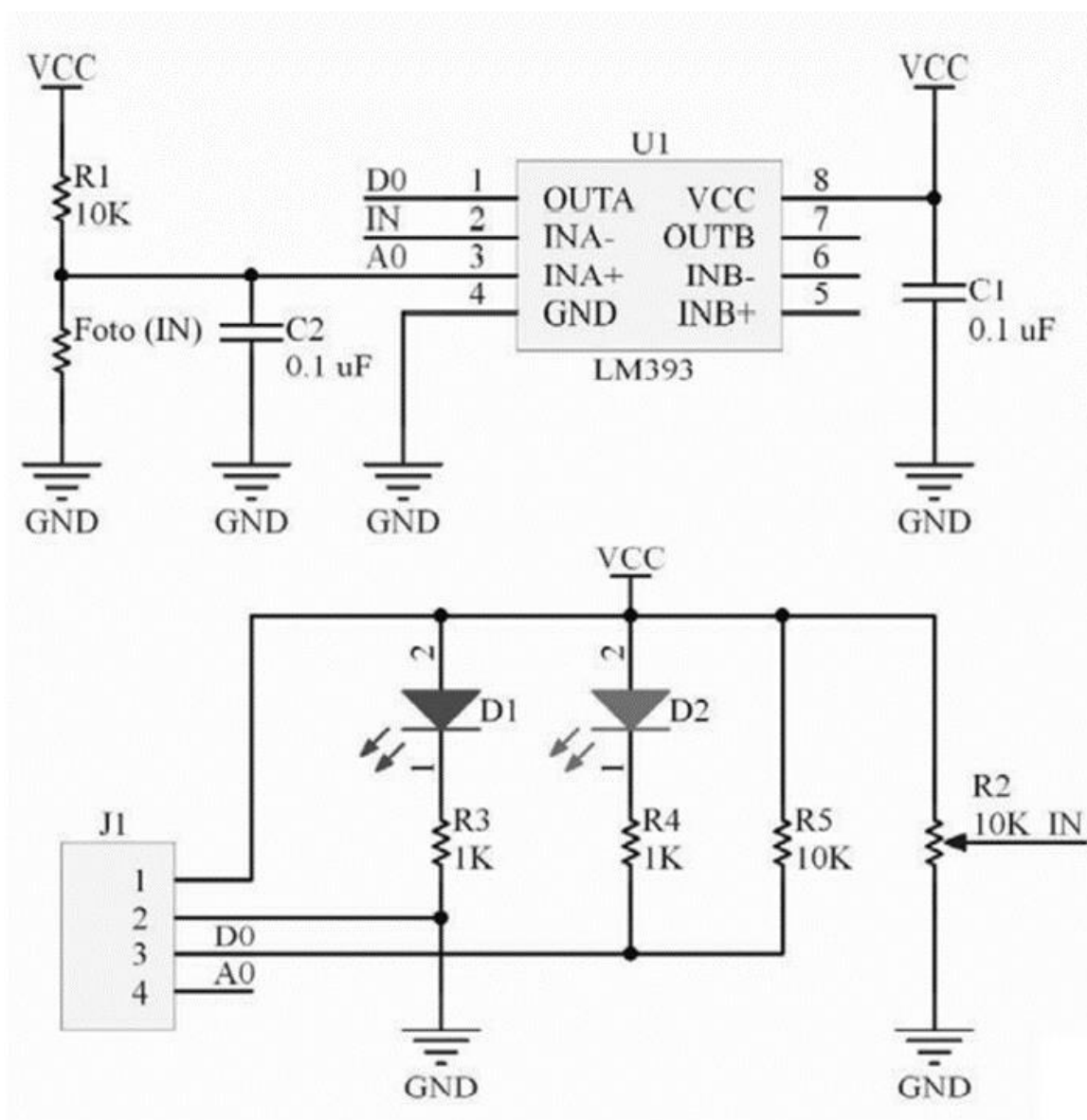


Рисунок 2.9 – Принципова схема модуля освітленості на LM393

Ультразвуковий датчик HC-SR04 працює по такому ж принципу, що і ультразвук в кажанів. Дана технологія працює так: датчик відправляє звуковий імпульс частотою 40 кГц після чого отримує відлуння з набором даних про навколишня середовище. Вагомим плюсом є те, що датчик HC-SR04 не реагує на сонячне світло, але недоліком є ризик, що пристрій може отримати неправдиві покази через тканину або тонкий предмет.

Передня частина HC-SR04, яка видно на рисунку 2.10, представлена р два ультразвукових датчиками, перший з написом Т (Transmitter), він

відправляє ультразвукову хвилю (ТСТ40-16Т), а другий з маркуванням R (Receiver), приймає відлуння від об'єктів поблизу (ТСТ40-16R), в центрі розташовується кварцовий генератор на 27 МГц.



Рисунок 2.10 – Ультразвуковий датчик HC-SR04

На іншому боці датчика HC-SR04, розташовується електрична обв'язка на якій виділяється набір основних елементів мікросхеми. Взаємодію з платою Arduino забезпечують чотири вивідні роз'єм, задачі контактів наведено нижче.

Призначення контактів модулю (у виконанні 4 pin):

- VCC: «+» живлення модуля
- Trig: вхід тригера
- Echo: вихід, відлуння.
- GND: «-» живлення модуля

На рисунку 2.11. продемонстровано принцип роботи модуля.

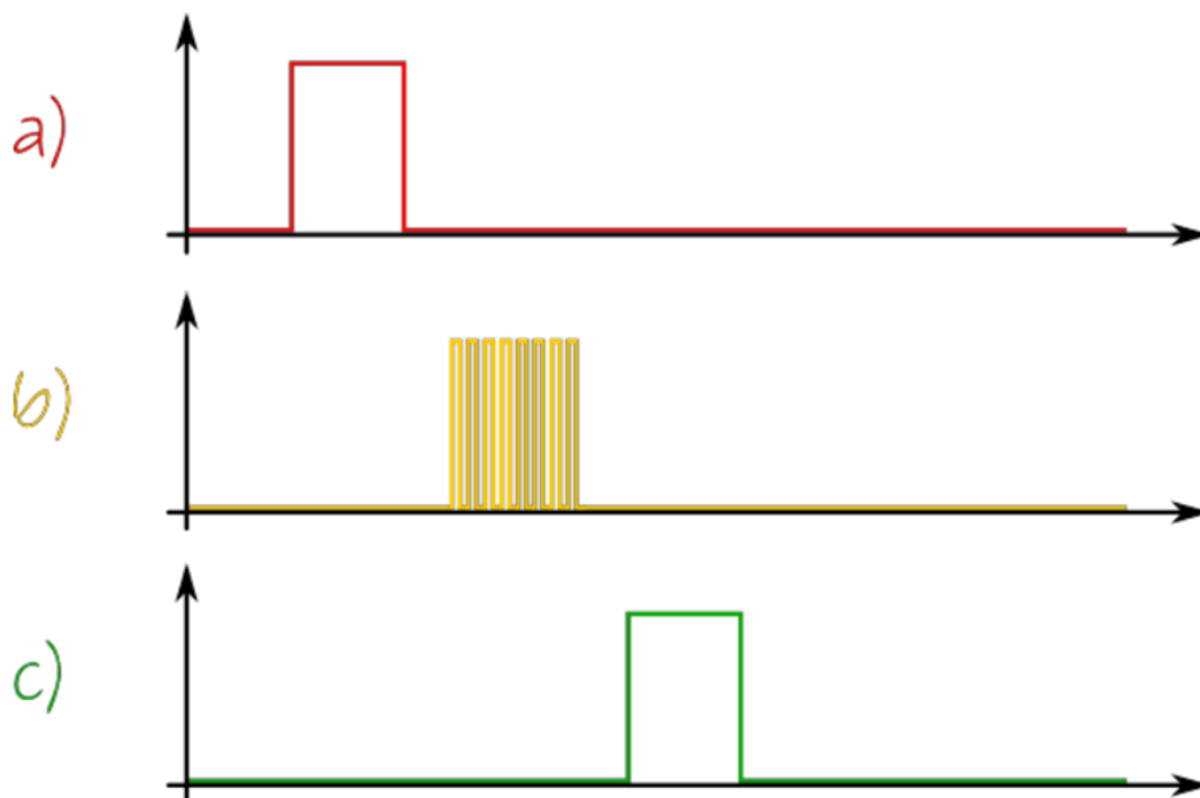


Рисунок 2.11 – Часові діаграми далекоміра

Щоб відбулась ініціалізація відправки сигнал з далекоміру, потребується подача високого сигнал тривалістю  $10\ \mu\text{s}$  на пін Trig. Коли модуль отримує необхідний сигнал тривалістю  $10\ \mu\text{s}$  на пін Trig, ним генерується пучок з восьми сигналів частотою  $40\ \text{кГц}$  та встановлюється високий рівень на пині Echo [10].

Коли модуль отримує відлуння відправленого сигналу, ним встановлюється на пін Echo низький рівень [15].

Характеристики ультразвукового далекоміру HC-SR04:

- Напруга живлення: 5 В;
- Споживання в спокійному режимі: 2 мА;
- Споживання підчас роботи : 15 мА;
- Діапазон відстаней: 2-400 см;

- Надійний кут спостереження: 15°;
- Стандартний кут спостереження: 30°;

Діаграма спрямованості далекоміра показана на рисунку 2.12.

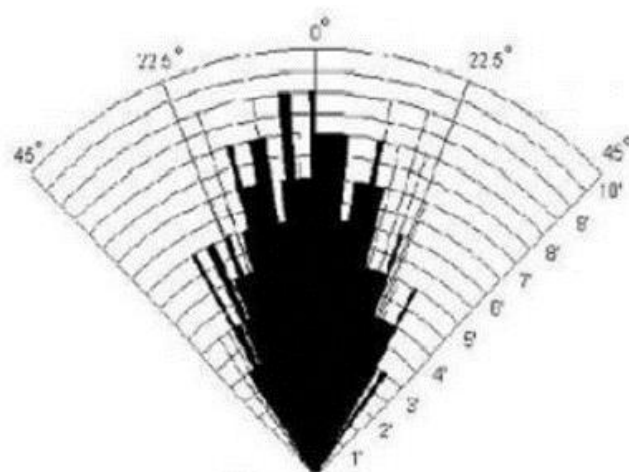


Рисунок 2.12 – Діаграма принципу роботи далекоміра

Будь-яка жива істота, що має температуру вище нуля, випромінює деяку кількість тепла. Виділення тепла живим тілом неможливо зафіксувати без телевізійного обладнання, тому що його випромінювання знаходиться в інфрачервоному спектрі хвиль, а даний спектр не бачить людське око. Дана енергія не є температурою, тому способи фіксації в них відмінні. Температура пов'язана з теплопровідністю, тому людина що увійшла до кімнат, не здатна миттєво змінити температуру в приміщенні. Тому методи, для виміру температури не можуть зафіксувати появу людини поблизу. Проте інфрачервоне випромінювання, від людського тіла може бути зафіксоване за допомогою датчика PIR [11].

Інфрачервоний датчик руху HC-SR501 простий, при використанні пристрій, який орієнтується на нормальне інфрачервоне випромінювання в рамках заданої зони фіксації. В включеному стані він фіксує зміну у випромінюванні, це може бути випадок коли людина пройшла або перемістився в рамках оглядової зони. Щоб зафіксувати інфрачервоне

випромінювання модуль застосовує вбудований піроелектричний датчик. По суті це пристрій, що генерує електричний струм під час фіксації інфрачервоного випромінювання [5]. Коли виявлено зміну в даних що надходять, датчик HC-SR501 змінює свій сигнал. Зовнішній вигляд датчика наведено на рисунку 2.13.

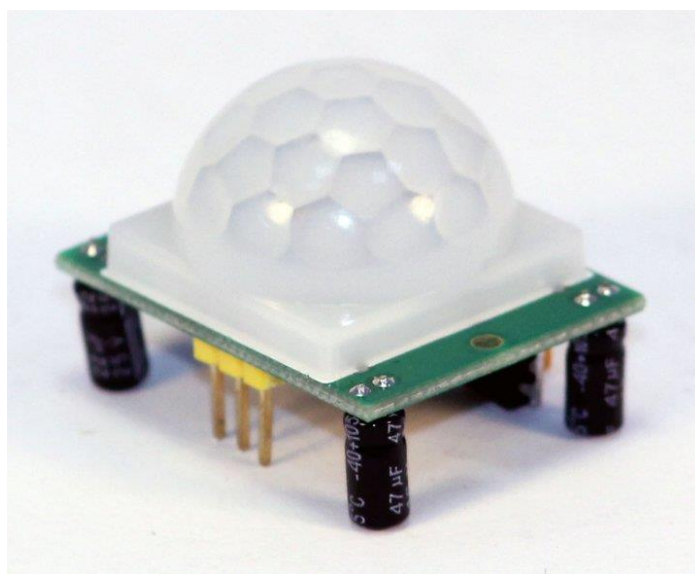


Рисунок 2.13 – Піроелектричний інфрачервоний (PIR) датчик руху HC-SR501

Щоб збільшити ефективність датчика HC-SR501 застосовують метод фокусування інфрачервоного лікування, використовуючи для цього Лінзи Френеля. Вона виготовляється з пластмаси у вигляді купола, що складається з певної кількості невеликих лінз Френеля. Пластик який використовується не прозорий для ока, проте чудово пропускає інфрачервоне світло, тому купол також фільтрує лише потрібне випромінювання[18].

Технічні характеристики:

- Напруга для датчика живлення: 4.8В ... 20В
- Статичний струм: 50 мА
- Рівень вихідного сигналу: 3.3 В / низький 0 В



- Час затримки: 0.5 - 200с, можна налаштувати
- Час блокування: 2.5с
- Кут для роботи: <100
- Межі фіксації температури: -15С ... + 70С
- Визначення об'єктів: 23 мм
- Розмір: 33мм x 25мм x 24мм

HC-SR501 – доступний датчик PIR, який може працювати автономно, або під управлінням контролера Arduino. Датчик може змінювати чутливість, що дозволяє зафіксувати рух в межах від трьох до семи метрів, також його вихід налаштовується таким чином, аби він був високим на протязі періоду від трьох секунд до п'яти хвилин. Також в датчику наявний вбудований стабілізатор напруги, тому його можна заживити від постійної напруги починаючи з 4,5 до 20 вольт, він буде споживати не велику кількість енергії. HC-SR501 має трьох контактний роз'єм, призначення виводів представлено на рисунку 2.14.

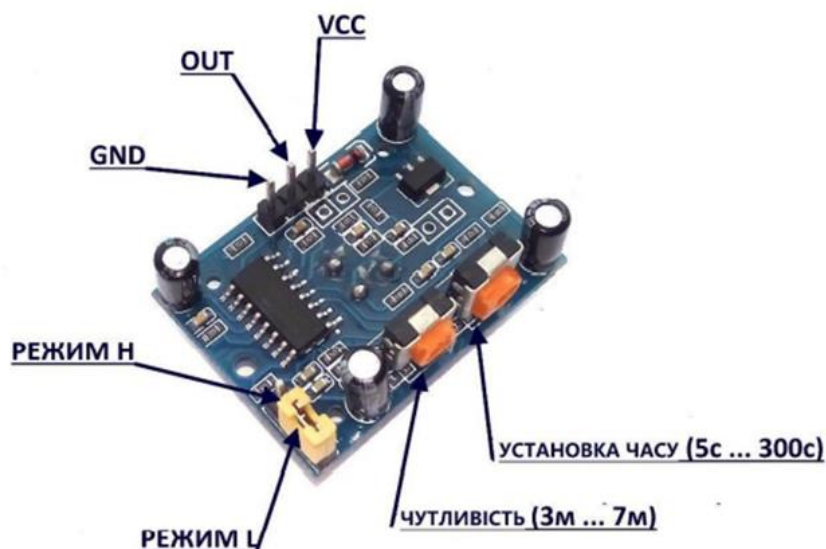


Рисунок 2.14 – Призначення виводів модулю HC-SR501

Призначення виводів:

- VCC - плюсова напруга постійного струму від 4,5 до 20 В.
- OUTPUT - логічний вихід на 3,3 вольт. LOW не вказує на виявлення, HIGH означає, що хтось був виявлений.
- GND - заземлення.
- На плату також вбудовано два потенціометра, що дає змогу налаштувати кілька параметрів:
- SENSITIVITY - встановлює максимальне і мінімальну відстань, від 3 до 7 метрів відповідно.
- TIME- час, на протязі якого вихід залишається HIGH після фіксації. мінімум 3 секунди, а максимум 300 секунд.

#### Призначення переминок:

- H - це настройка Hold або Repeat. У цьому положенні HC-SR501 буде продовжувати видавати сигнал HIGH, поки він продовжує виявляти рух.
- L - Це параметр переривання або без повтору. У цьому положенні вихід залишатиметься HIGH протягом періоду, встановленого налаштуванням потенціометра TIME. На платі HC-SR501 є додаткові отвори для двох компонентів, поруч розташовані маркування, подивитися на них можна знявши лінзу Френеля [19].

#### Призначення додаткових отворів:

- RT - це призначене для термістора або чутливого до температури резистора. Додавання цього дозволяє використовувати HC-SR501 в екстремальних температурах, а також в деякій мірі підвищує точність роботи детектора.
- RL - це з'єднання для світлозалежна резистора або фоторезистора. Додаючи компонент, HC-SR501 буде працювати тільки в темряві, що є загальним додатком для чутливих до руху систем освітлення.

### 2.3 Висновки до другого розділу

В даному розділі мною розглянута апаратна частина системи керування освітленням. За основу для системи було обрано контролер на базі готової плати Arduino UNO, який буде з'єднаний з всіма датчиками та модулями. Даний контролер чудово підходить для поєднання в єдину систему всіх елементів, та зможе збирати отриману інформацію для подальшого її аналізу та прийняття необхідних рішень. Функціональність і ефективність системи будуть забезпечувати різноманітні датчики у вигляді окремих модулів системи, вони будуть зчитувати параметри для контролю освітлення в приміщеннях, а також присутності користувача або «непрошених гостей». Інформація з цих модулів буде надіслана на контролер для обробки.

Вагомою перевагою автоматизованої системи контролю освітленням на базі Arduino UNO є досить доступна ціна самої плати та модулів до неї. В поєднанні з відносно легким збором апаратної частини, можна зробити висновок про оптимальність вибраних комплектуючих.

## 3 ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ЦЕНТРАЛЬНОГО КОНТРОЛЛЕРУ

### 3.1 Принцип взаємодії елементів системи

Вся архітектура робота пристроїв в системі автоматизованого управління освітленням спирається на основу, що складається з центрального контролера Arduino UNO. Робота освітлювального обладнання, буде автоматично коригуватись, в залежності від поточних сигналів з датчиків. Для ручного управління системою та пере налаштування деяких процесів, клієнт зможе застосовувати спеціальний мобільний додаток . Надійність та стабільність роботи забезпечить хмарний сервіс Dropbox, з яким буде синхронізовано проект. Схема взаємодії елементів в системі наведена на рисунку. 3.1.

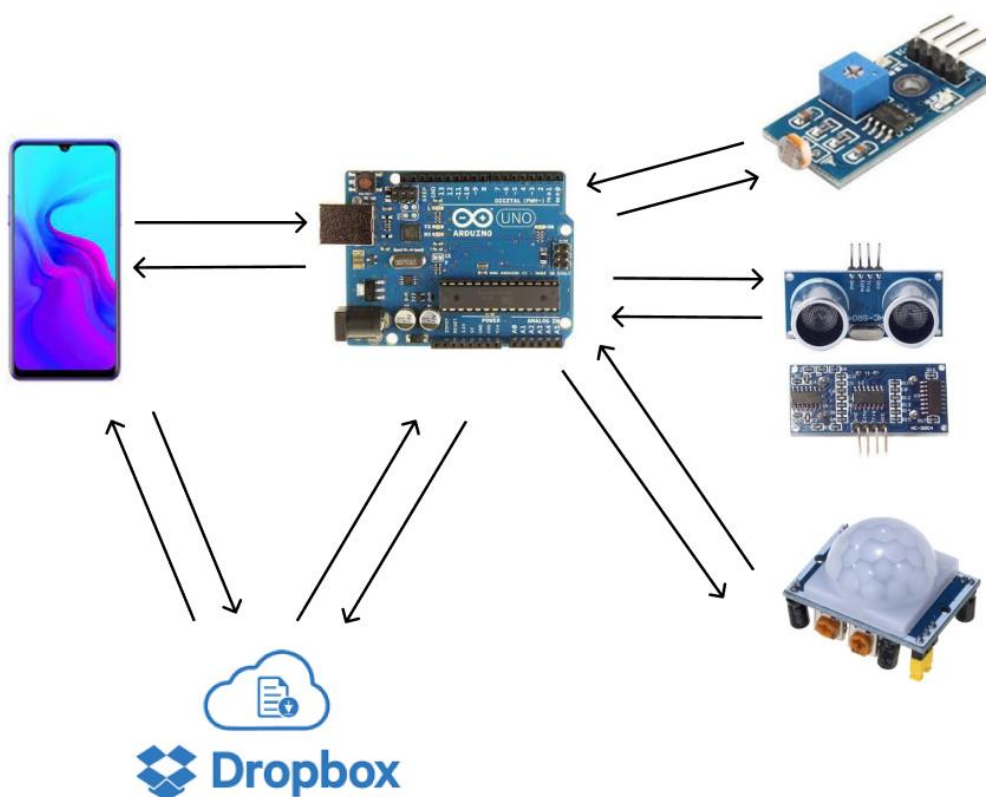


Рисунок 3.1 – Діаграма, що представляє принцип взаємодії приладів

Даний принцип організації системи має вагомі переваги:

- Надійність системи. При використанні хмарного сервісу як додаткового сховища для системи, її надійність суттєво збільшується. Оскільки дані, що зберігаються там мають додатковий рівень захищеності. Крім того образ системи збережений в такому сховищі можна використати для відновлення при пошкодженні ПО.

- Зменшення навантаження на мікроконтролер. Arduino UNO містить обмежений набір пам'яті в 32мб, тому прямий доступ до неї, так само як і зберігання з обробкою великої кількості налаштувань, негативно позначаються на її функціонуванні. У випадку з хмарним сервісом, частина логіки та потрібні для роботи дані зберігаються на ньому.

- Масштабованість та розширення системи. Використання Arduino як центрального контролера дає ряд переваг при масштабуванні системи управління освітленням, оскільки дана платформа підтримує велику кількість різноманітних модулів та датчиків. Також функціонал та масштаб такої системи можна суттєво розширити, поєднавши кілька плат Arduino в єдину систему.

- Зручність використання. Оскільки управління буде здійснюватися за допомогою спеціального мобільного додатку, користування функціоналом системи стає максимально зручним та зрозумілим.

Окрім вагомих переваг дана система також має деякі недоліки:

- Потрібно використовувати сторонній хмарний сервіс в якості додаткового сховища даних системи. Тому проект в деякій мірі стає залежним від такого сервісу.

- Хоча мобільний додаток забезпечує зручність використання системи, проте розширення апаратної частини та функціоналу призведе до потреби його оновлення чи навіть замінити.

### 3.2 Розробка програмного забезпечення центрального контролера

Щоб створити програмне забезпечення для розробленої апаратної частини, було використано Arduino IDE. Дане середовище орієнтоване на проекти, що будуються на базі Arduino і має чудовий інструментарій для роботи з цими платами.

Програмне забезпечення Arduino (IDE) з відкритим вихідним кодом дозволяє легко писати код і завантажувати його на плату. Це програмне забезпечення можна використовувати з будь-якою платою Arduino.

Основними перевагами Arduino IDE є :

- доступність;
- зручний для використання та розуміння інтерфейс;
- програма сумісна з усіма версіями операційної системи Windows;
- наявність широкої бази бібліотек;
- наявність необхідних для роботи інструментів;
- кілька варіантів мов програмування;
- дозволяє розширити знання C подібних мов;
- кросплатформенність;
- вбудований набір прикладів програм;
- відкритий вихідний код;
- функції збереження, експорту, перевірки, пошуку, заміни скетчів.

Інтерфейс середовища Arduino IDE показано на рисунку 3.2.

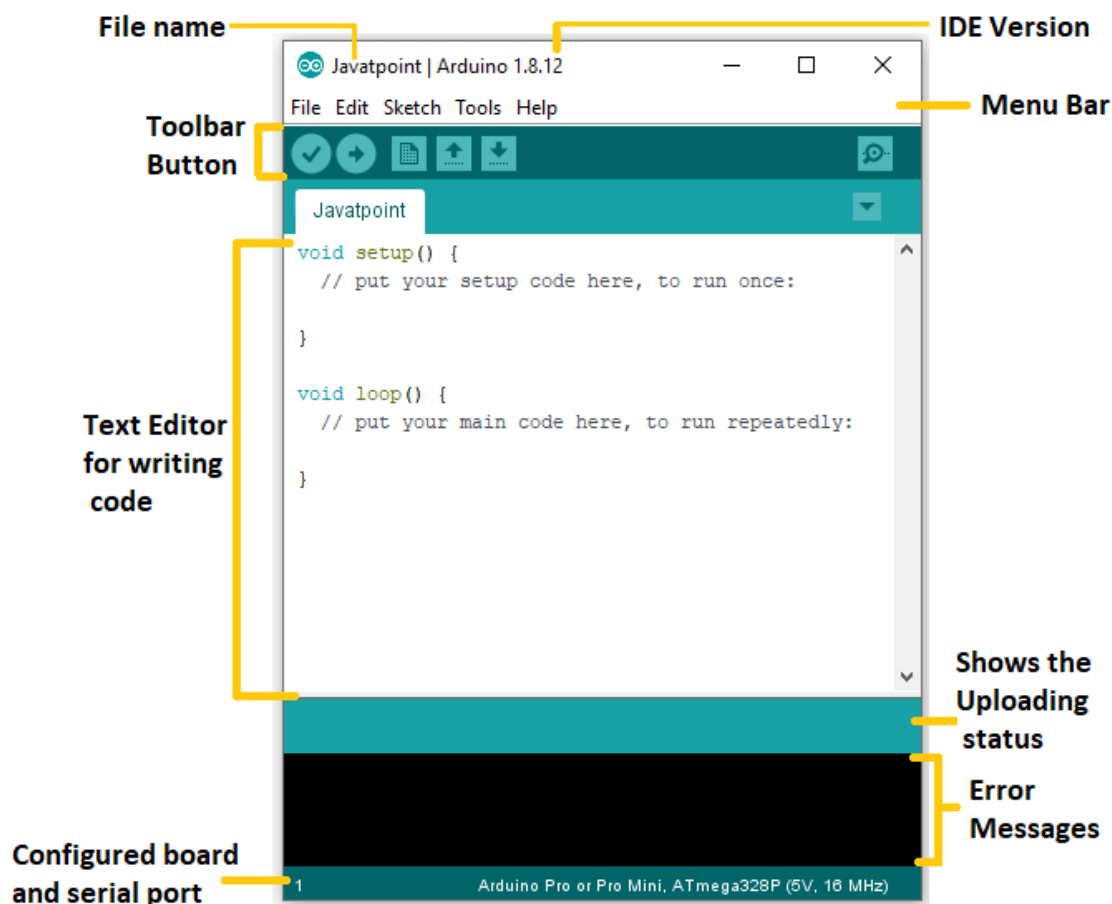


Рисунок 3.2 – Інтерфейс середовища програмування Arduino IDE

Як основна мова для створення програмної частини, чудово підходить C, вона підтримується середовищем розробки Arduino IDE і має широкий функціонал.

C – мова програмування високого рівня з підтримкою таких парадигм програмування як об'єктно-орієнтованість, узагальненість та процедурність. Серед переваг можна назвати: модульність, обробку винятків, роздільне компілювання, абстракцію даних. Також C є однією з найбільш поширених мов для створення програмних продуктів. Програми, написані на мові C, є мобільними, тобто можуть бути виконані на комп'ютерах різних виробників і в різних операційних системах. Проте вона застосовується не тільки при створенні програмного забезпечення, але і для розробки різноманітних

драйверів [14]. На даний момент C доволі популярний і зручний спосіб для програмування мікроконтролерів.

Під управлінням програмного забезпечення, центральний контролер збирає дані з датчиків та модулів, що приєднані до нього, обробляє їх і відштовхуючись від отриманого результату, відправляє команди на освітлювальне обладнання. За звичай умов система не потребує прямого втручання власника. Але за допомогою мобільного додатку користувач може змінити налаштування чи режими роботи системи, а також напряму впливати на підконтрольні пристрої. Система моніторить інформацію від своїх датчиків, через певні відрізки часу передаючи її на контролер Arduino UNO .

Основа контролер Arduino складається з блоків порівняння та прийняття рішення. Блок порівняння зберігає різницю чинного і еталонного показника керованого параметра. Відштовхуючись від значення і знака даної різниці у блоці прийняття рішення проводиться аналіз, після чого даються вказівки на певну дію, для виконавчого обладнання. Блок схема, що демонструє роботу системи наведено на рисунку 3.3.



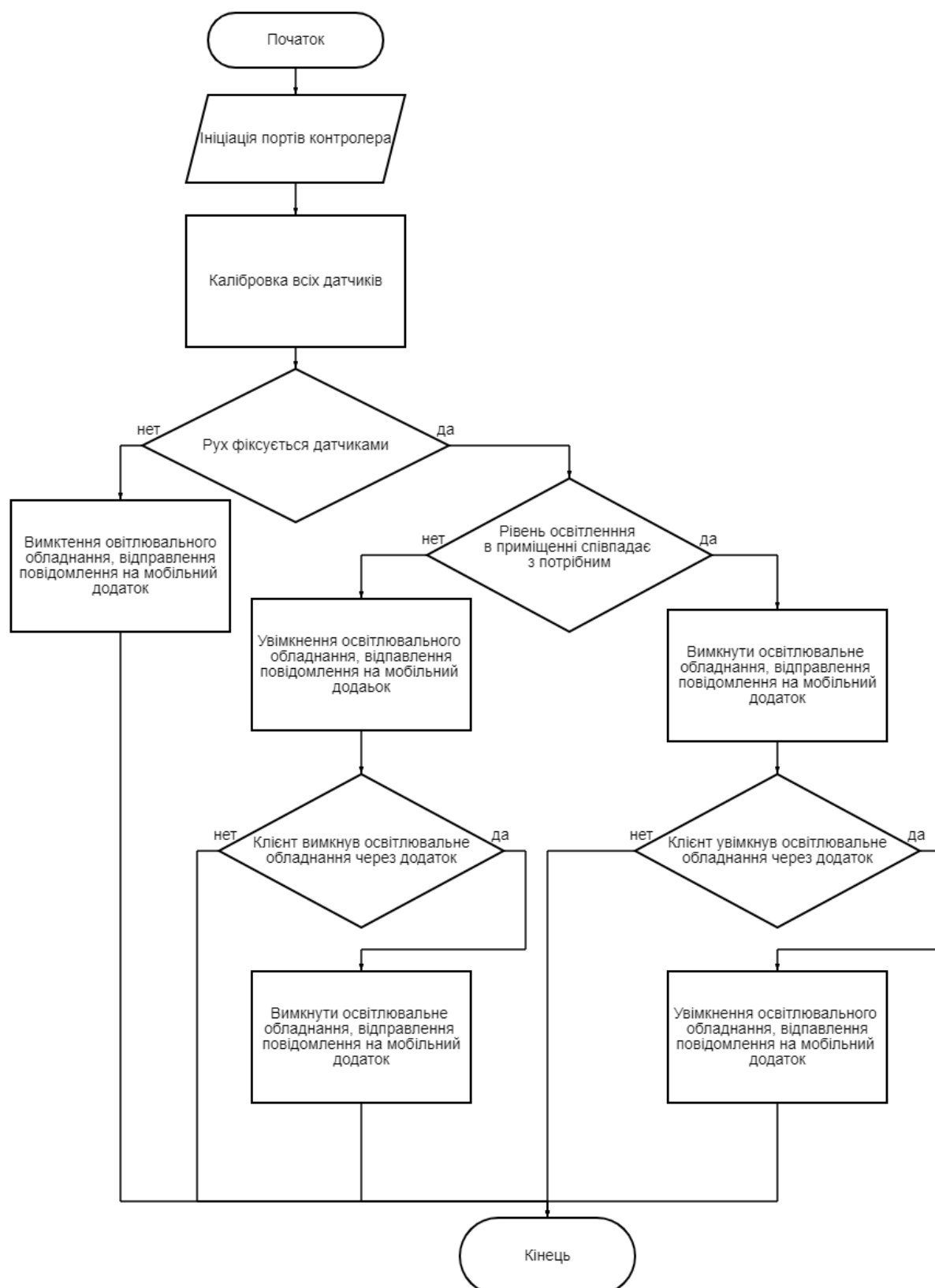


Рисунок 3.3 – Блок схема, що демонструє роботу системи

Підчас складання програмного забезпечення на мові С, спочатку було написано частину коду, що відповідає за опис вхідних даних системи. Для цього використано бібліотеки MFRC522.h та SPI.h

Далі був складений програмний код на мові С в середовищі розробки Arduino IDE. В кодї були використані бібліотеки MFRC522.h і SPI.h. Частина коду, яка описує вхідні дані, наведено на рисунку 3.4.

```

9  #include <SPI.h>
10 #include <MFRC522.h> // RFID library. int led = 4; // LED
11 int red_led = 5; int green_led = 6;
12 int p = 3; // LED
13 int pirPin = 8; // motion sensor
14 int pirState = LOW; //starting parameters of the motion sensor int val = 0;
15 char uidCard;
16 char incomingByte; int ihome = 0;
17 #define RST_PIN 9 // # define SS_PIN 10 //
18 MFRC522 mfrc522(SS_PIN, RST_PIN);
19 unsigned long uidDec, uidDecTemp ; // to store the label number in decimal format
20
21 void setup() { pinMode(led, OUTPUT); pinMode (15, OUTPUT);
22   pinMode (16, OUTPUT);
23   pinMode (17, OUTPUT);
24   pinMode (18, OUTPUT);
25   pinMode (green_led, OUTPUT);
26   pinMode (red_led, OUTPUT);
27   pinMode (p, OUTPUT);
28   pinMode (pirPin, INPUT);
29   Serial.begin(9600);
30   Serial.println("Waiting for card...");
31   SPI.begin(); // initialization SPI / Init SPI bus.
32   mfrc522.PCD_Init(); // initialization MFRC522 / Init MFRC522 card.
33 }
34

```

Рисунок 3.4 – Введення вхідних даних

На рисунку 3.5 показано елементи коду, що відповідають за контроль освітлювального обладнання.

```

34
35 // light control
36
37 void light()
38 {
39
40 digitalWrite(green_led, HIGH); analogWrite (p, 0);
41 val = digitalRead(pirPin); if (val == HIGH)
42 {
43 digitalWrite(15, HIGH); //turn on the LED
44 digitalWrite(16, HIGH); //turn on the LED
45 digitalWrite(17, HIGH); //turn on the LED
46 digitalWrite(18, HIGH); //turn on the LED
47 }
48 else
49 {
50 digitalWrite(15, LOW); //turn on the LED
51 digitalWrite(16, LOW); //turn on the LED
52 digitalWrite(17, LOW); //turn on the LED
53 digitalWrite(18, LOW); //turn on the LED
54 }
55 }
56
57 void loop() {
58 if (ihome == 1)
59 {
60 light();
61 }
62 else
63 {
64 signal ();
65 }

```

Рисунок 3.5 – Реалізація керування освітленням

### 3.3 Застосування хмарного сервісу для роботи з даними

Постійно відкриваються нові способи зберігати дані. Щоб обрати найбільш підходяще рішення, важливо зрозуміти, що таке хмарні сервіси. Кожен з таких сервісів є видом онлайн-сховищ. У таких сервісах відбувається зберігання інформації на різних серверах, якими компанії дозволяє користуватися клієнтам. При цьому клієнт не знайомий з внутрішньою структурою сервісу. Зберігання і обробка даних відбуваються в хмарі, яка являє собою Web сервер. Такі сервера в реальному світі можуть знаходитися в різних віддалених місцях [6]. Наприклад, в різних країнах чи навіть на різних континентах. Функціонал даного сервісу знадобився для підвищення

надійності роботи, оскільки є ризик збою апаратного і програмного забезпечення. Синхронізація ж системи з віддаленим сховищем, дозволяє відновити знищені на стаціонарному носію дані. Також дані, що зберігаються в хмарі захищені від зламу шифруванням.

Як провайдер для синхронізації з автоматизованою системою управління освітленням було обрано продукт від компанії Dropbox Inc.

Dropbox надає можливість централізованого зберігання файлів онлайн, послуги обміну файлами та синхронізації даних. Даний сервіс пропонує низку рішень хмарного зберігання, які дозволяють вибрати необхідний обсяг місця для зберігання даних без обмежень.

Функції:


- Доступ до будь-якого файлу у вашому обліковому записі навіть без підключення до Мережі та попередній перегляд файлів 175 з гаком типів, для якого не потрібно жодного спеціального ПЗ.
- Можна без проблем надсилати великі файли, просто скопіювавши та вставивши, можна надіслати будь-куди — навіть на гаджет, у якому немає облікового запису Dropbox.
- Скануйте документи, чеки, паспорти, фотографії та багато іншого за допомогою мобільного пристрою та перетворюйте на файли PDF високої роздільної здатності, які можна переглядати та надсилати куди завгодно.
- Синхронізувати папки між комп'ютерами Windows або Mac та Dropbox за допомогою резервного копіювання даних з комп'ютера; відновлення старих версій файлів або видалених файлів за допомогою журналу версій та функції відновлення файлів.


Після установки програмного забезпечення Dropbox на клієнтський пристрій створюється папка, синхронізації з Dropbox. Головні задачі, що ставиться перед технологією, це можливість синхронізації та обмін інформацією, Dropbox зберігає історію завантажень, тому після видалення потрібних даних з сервера чи збоїв на боці клієнта є можливість відновити втрачені дані. Крім цього сервіс моні торить історію зміни файлів, якою

можна скористатися протягом 30 діб, також доступна додаткова можливість безстрокового зберігання історії змін «Pack-Rat». Історія зміни файлів ведеться за принципом Diff-кодування, даний тип кодування дає можливість заощадити місце, для зберігання файлів. Історія змін зберігає лише відмінність поточної версії файлу від минулої. Файли, які були завантажені через клієнтську сторону, не обмежуються по розміру, проте файли, що завантажені за допомогою Web-інтерфейсу, обмежуються розміром 300 МБ. Крім цього сервіс дає можливість додавати файли для публічного доступу за допомогою папки «Public», тому дану функцію можна використовувати як файлообінника. Ще однією можливістю, яка з'явилася в нових версія Dropbox дозволяє надавати в загальний доступ папки з My Dropbox для подальшого користування через «shareable link», тобто за допомогою Web-інтерфейс. Спільну роботу для кількох осіб, Dropbox може забезпечити за допомогою «Shared» папок [25].

Технологій, що пропонуються хмарним сервісом Dropbox можна використати в Android додатках за допомогою Dropbox Core API. Безпека з'єднання організовується за допомогою токенів, тому потребує підтвердження від користувачем підчас першого застосування сервісу. Сторінка авторизації в хмарний сервіс Dropbox наведена на рисунку 3.6.

Увійти або [створити обліковку](#)

 Увійти за допомогою Google

 Увійти за допомогою Apple

---

або

Електронна пошта

Пароль

Запам'ятати мене

[Забули пароль?](#)

Рисунок 3.6 – Авторизація в хмарному сервісі Dropbox

Для уникнення подальших підтверджень під час завантаження чи передачі даних, клас `SharedPreferences` зберігає в собі потрібний токен. Щоб синхронізувати роботу в тлі було застосовано інтерфейс `AsyncTask`. Даний інтерфейс має можливість асинхронно виконувати код, при цьому ніяк не заважаючи роботі потоку графічного інтерфейсу, тому його рекомендують застосовувати під час виконання задач, що повинні виконуватись не більш 40 секунд, це забезпечить максимальну ефективність інтерфейсу. Також даний інтерфейс застосовується в методі завантаження програм. Метод завантаження файлу в Dropbox та клас `AsyncThread` наведено на рисунок 3.7.

```

/**
 * The method for downloading files from Dropbox
 * @param fileNamesInDropbox The name of files, which you want to get
 * @param files The initialized files, where you want to write files from dropbox
 * @return ArrayList with files from Dropbox
 * @throws ExecutionException in case of wrong file name or some other clauses, which may result
 * in not finding file with your filename in Dropbox folder
 * @throws InterruptedException in case of interrupting this thread by another one
 */
public ArrayList<File> download(String[] fileNamesInDropbox, File... files) throws ExecutionException, InterruptedException {
    DropboxConnector.DownloadFileFromDropbox dwl = new DropboxConnector.DownloadFileFromDropbox(
        DropboxConnector.FILE_DIR, mDBApi, fileNamesInDropbox);
    return dwl.execute(files).get();
}

//Async classes

/**
 * The Async class for downloading file from Dropbox in new Thread. Have to be invoked in UI
 * thread
 */
private class DownloadFileFromDropbox extends AsyncTask<File, Void, ArrayList<File>> {
    /**
     * The path to app folder in Dropbox
     */
    private String path;
    /**
     * A Dropbox API class
     */
    private DropboxAPI<?> dropbox;
    /**
     * The name of files, which you want to get
     */
    private String[] fileNamesInDropbox;

    /**
     * The constructor of async download class
     * @param path The path to app folder in Dropbox
     * @param dropbox A Dropbox API class
     * @param fileNamesInDropbox The name of files, which you want to get
     */
    public DownloadFileFromDropbox(String path, DropboxAPI<?> dropbox, String[] fileNamesInDropbox) {
        this.path = path;
        this.dropbox = dropbox;
        this.fileNamesInDropbox = fileNamesInDropbox;
    }

    @Override
    protected ArrayList<File> doInBackground(File... params) {
        ArrayList<File> files = new ArrayList<>(params.length);

        int count = 0;
        for(File file: params){
            FileOutputStream fileOutputStream;
            try {
                fileOutputStream = new FileOutputStream(file);
                dropbox.getFile(path + fileNamesInDropbox[count], null, fileOutputStream, null);
                files.add(file);
                count++;
            } catch (DropboxException | IOException e) {
                e.printStackTrace();
            }
        }
        return files;
    }

    @Override
    protected void onPostExecute(ArrayList<File> resFiles) {
    }
}

```

Рисунок 3.7 – Завантаження файлів в хмарне сховище Dropbox

### 3.4 Висновки до третього розділу

В рамках розділу було розроблено принцип взаємодії елементів системи, та їхнє програмне забезпечення. Дана частина автоматизованої системи управління освітленням відіграє таку ж важливу роль як апаратна основа. Тому підчас її розробки було необхідно створити код, що забезпечить необхідний для проекту функціонал, а також буде без проблем поєднуватись з всіма датчиками та модулями. Оскільки у центрального контролера є проблеми з вбудованою пам'яттю, в систему було інтегровано хмарний сервіс, який допоможе в зберіганні необхідних для роботи даних та забезпечить їх безпеку. Для цього було створено код, що відповідає за завантаження даних в хмару.



## 4 РОЗРОБКА МОБІЛЬНОГО ДОДАТКУ ДЛЯ УПРАВЛІННЯ СИСТЕМОЮ

### 4.1 Обґрунтування вибору операційної системи

Перед тим як почати розробку мобільного додатку, необхідно проаналізувати всі доступні платформи (операційні системи) та обрати найбільш оптимальну з них. Ту яка дозволить найкраще реалізувати функціонал необхідний мобільному додатку. Для цього потрібно розглянути саме поняття операційної системи та її існуючі реалізації для мобільних пристроїв.

Операційною системою (ОС) називають програмне забезпечення, яке діє як інтерфейс між кінцевим користувачем та апаратним забезпеченням комп'ютера або мобільного пристрою. Іншими словами, ОС надає користувачу можливість керувати власним пристроєм, не вимагаючи при цьому поглиблених знань в його архітектурі. Наявність ОС є необхідною частиною для запуску інших додатків (програм), так як вона забезпечує середовище, в якому вони будуть працювати та виконувати свої завдання [23].

Серед операційних систем для мобільних пристроїв є IOS, Android, Windows та Samsung. На рисунку 4.1 наведено відсоткове співвідношення між кількістю пристроїв на базі даних ОС(в період 2014-2023) на ринку України:

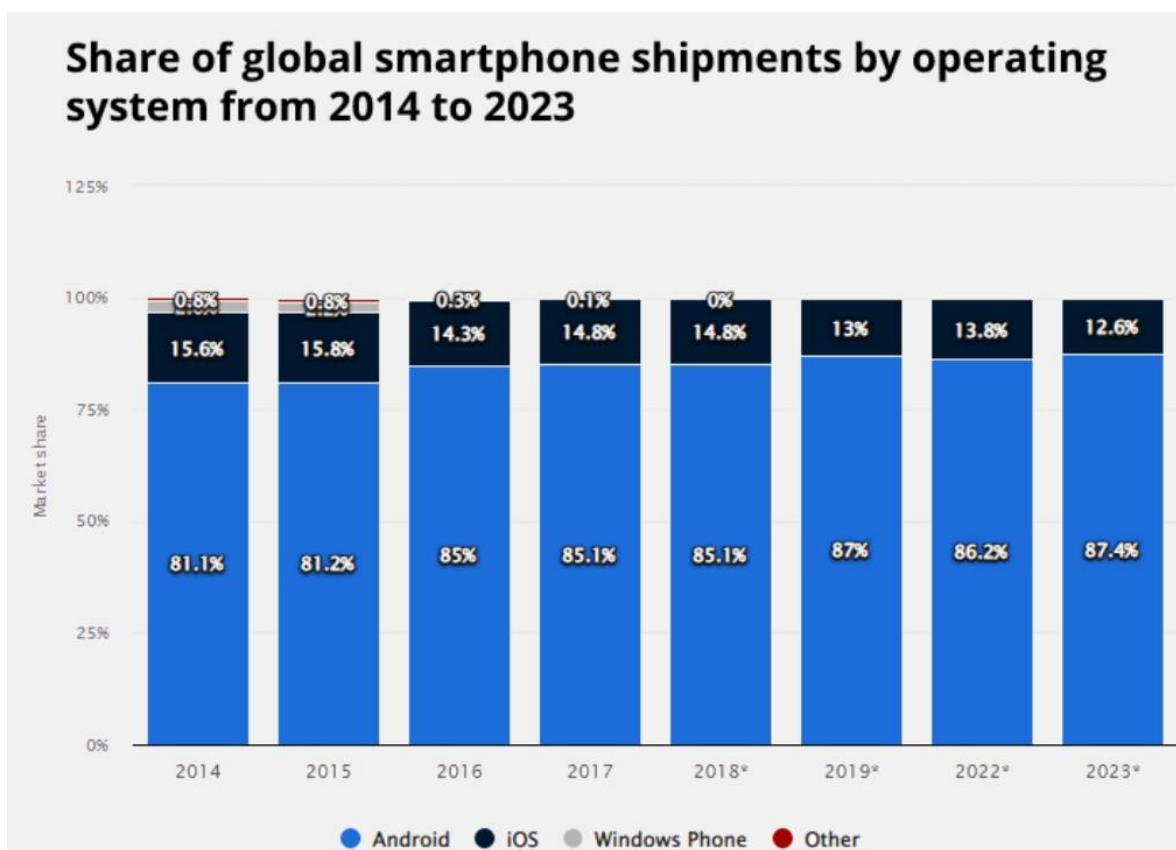


Рисунок 4.1 – Відсоткове співвідношення мобільних ОС на ринку України

Як можна побачити по статистиці, ОС Android опереджає найближчого свого конкурента більш ніж у 4 рази в межах України

Вибір операційної системи для реалізації мобільного додатку залежить від ряду чинників, які включають в себе цільову аудиторію, функціональність додатку, ресурси розробки та стратегії маркетингу. Android і iOS є двома основними операційними системами для мобільних пристроїв, кожна з яких має свої переваги. Платформа iOS - це операційна система, на якій працює популярна лінійка смартфонів Apple iPhone. Другою - Android від Google. Операційна система Android використовується не лише пристроями Google, а й багатьма іншими OEM-виробниками для створення власних смартфонів та інших інтелектуальних пристроїв.

Хоча між цими двома платформами є деяка схожість при створенні застосунків, розробка для iOS і розробка для Android припускають використання різних комплектів розроблення програмного забезпечення (SDK) і різних наборів інструментів розробки. Тоді як Apple використовує iOS виключно для своїх пристроїв, Google надає Android іншим компаніям за умови, що вони відповідають певним вимогам, таким як включення певних застосунків Google на пристрої, які вони постачають [19].

Якщо розглядати складність розробки, то розробка для IOS вважається порівняно легшою. Це, насамперед, пов'язано з мовою програмування Swift, що використовується при розробці для даної ОС. Порівняно з мовою програмування Java, на якій написано абсолютну більшість додатків на Android, Swift вимагає меншого об'єму коду для вирішення тих же задач, що зменшує час розробки програм.

В той же час, оскільки Android – це ОС з відкритим кодом, вона є більш гнучкою для розробки додатку, так як надає більш широкий спектр можливостей для створення власної унікальної функціональності з меншими обмеженнями.

Щодо інструментів та ціни розробки, то для Android існує інтегроване середовище розробки Android Studio, яке Google пропонує безкоштовно. Також розробники можуть безкоштовно розміщувати свої додатки в Google Play, хоча вони повинні сплатити одноразовий внесок у розмірі 25 доларів. У IOS офіційним інструментом розробки додатків є Xcode, котрий також доступний безкоштовно. Пакет для розробників Apple (компанії, що власне розробляє дану ОС) коштує 99 доларів на рік і включає можливість розміщення програм у iTunes App Store.

Підводячи підсумки, можна сказати, що обидві платформи надають користувачам можливість безкоштовної розробки додатків. Їх розміщення є ІАЛЦ.045490.004 ПЗ 16 Арк. Зм Лист № докум. Підп. Дата дорожчим у IOS, однак перевагою є більш простий процес розробки. Однак, ОС Android має 2 ключові переваги, що перехиляють вибір саме в її сторону:

1. Більша гнучкість розробки.
2. Значна перевага у поширеності мобільних пристроїв на локальному та всесвітньому ринку.

Останній компонент є критично важливим, так як додаток, що розробляється у даному проєкті, повинен бути доступним як можна більшої кількості користувачів.

Виходячи з усіх вище наведених аргументів, для розробки системи формування та керування електронної черги для мобільних пристроїв було обрано саме операційну систему Android.

## 4.2 Вибір середовища для розробки додатку

Android Studio – програма, яка є середовищем розробки додатків для мобільної платформи Android. Прямий конкурент найпопулярнішої утиліти для створення програмного забезпечення під Android - Eclipse. Середовище розробки з емулятором зображено на рисунку 4.2.

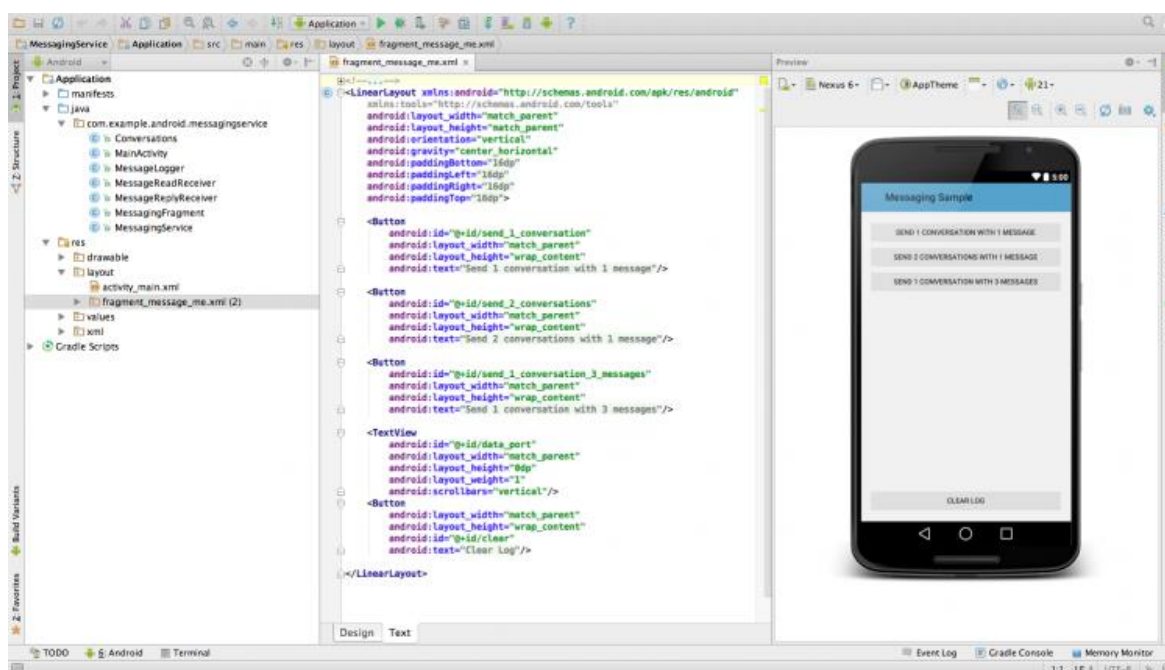


Рисунок 4.2 – Середовище розробки Android Studio

Android Studio перевершує конкурента за багатьма параметрами, до яких можна віднести:

- гнучкість середовища розробки;
- більший набір функцій;
- процес розробки, що підлаштовується під розробника.

Під час створення додатків та утиліт для операційної системи Android, користувач програмного забезпечення може спостерігати за змінами у проєкті, в режимі реального часу.

Android Studio вважається універсальне середовище розробки, тому що дозволяє оптимізувати роботу майбутніх програм для роботи не тільки на смартфонах, але і на планшетах, портативних ПК, які працюють на основі аналізованої операційної системи.

У програмі вбудований емулятор, що дозволяє перевірити коректну роботу програми на пристроях з різними екранами, із різними співвідношеннями сторін. Особливо актуальною ця функція стала після входу в тренди смартфонів, де встановлено екрани зі співвідношенням сторін 18:9.

Можна відмітити вагому особливість емулятора, що дозволяє переглянути приблизні показники продуктивності при запуску програми на найпопулярніших пристроях.

Середовище розробки для додатків Android Studio останньої версії стало по-справжньому зручним навіть для розробників-початківців. У програмі реалізовані всі сучасні засоби для пакування коду, його маркування. Затребувана багатьма авторами ПЗ функція Drag-n-Drop, що полегшує перенесення компонентів у середу розробки безпосередньо.

Локалізація програм стає значно простіше з функцією SDK, яка також входить до списку переваг Android Studio.

Основні переваги середовища розробки:

- середовище розробки підтримує роботу з кількома мовами програмування, до яких відносяться найпопулярніші - C/C++, Java.

- редактор коду, з яким зручно працювати;
- дозволяє розробляти програми не тільки для смартфонів/планшетів, а й для портативних ПК, приставок для телевізорів Android TV, пристроїв Android Wear, новомодних мобільних пристроїв із незвичайним співвідношенням сторін екрану;
- тестування коректності роботи нових ігор, утиліт, їхньої продуктивності на тій чи іншій системі, відбувається безпосередньо в емуляторі;
- ре факторинг готового коду;
- досить велика бібліотека з готовими шаблонами та компонентами для розробки ПЗ;
- розробка програми для Android N – найостаннішої версії операційної системи;
- попередня перевірка вже створеного додатка щодо помилок у ньому;
- великий набір засобів інструментів для тестування кожного елемента програми, ігри;
- для недосвідчених/початківців розробників спеціально створено посібник з використання Android Studio, розміщений на офіційному сайті утиліти.

Незважаючи на наявність вбудованого Android-емюлятора в середовищі розробки, з тестуванням ново розробленої програми можуть виникнути проблеми. Так, для його запуску необхідна досить велика апаратна основа ПК, на якому планується тестування.

Ще один недолік - це неможливість написати серверні проекти мовою Java для ПК, Android пристроїв.

ПЗ для розробки утиліт та програми на Android дійсно створює приємне перше враження. Android Studio оцінить як досвідчений розробник, так і початківець, який тільки освоює ази. Багатий набір інструментів, гнучкість у

розробці, можливості тестування, підтримка кількох мов програмування та вбудований емулятор роблять утиліту однією з найкращих у своїй ніші.

### 4.3 Графічна частина додатку

Віддалене керування платами Arduino через програму Android можливе завдяки графічному інтерфейсу користувача (GUI).

Додаток на пристроїв користувача є основним засобом взаємодії з автоматизованою системою управління освітленням, тому його інтерфейс є досить важливим [7]. Варто впевнитись, щоб додаток влаштує клієнта, оскільки його дизайн характеризує весь проект. Тому є основні вимоги яким повинні відповідати мобільні додатки, аби зацікавити клієнта:

- Якісне оформлення тексту за допомогою набору та верстки;
- Вірне розміщення об'єктів у просторі;
- Інтерфейс повинен бути інтуїтивно зрозумілим;
- Структура додатку не повинна бути перевантаженою;
- Додаток повинен легко налаштовуватись під потреби клієнта;

По суті, дизайн – це оформлення контенту, 80% якого передається через текст. Будь-яка інформація розташовується у вигляді ієрархії — головна та другорядна частини, що виділяються за допомогою певної стилізації тексту.

Наприклад, рівні заголовків 1-го або 2-го рівня повинні розрізняються за розміром шрифту і служать навігацією, а також роблять структурованим матеріал. Вирівнювання по лівому краю надає тексту акуратного вигляду тому його легко сприймати. Рядки починаються на одному рівні, і око швидко знаходить початок наступного рядка, тому користувач до тексту швидко звикає [21].

Інтуїтивна зрозумілість для інтерфейсу є основним показником його вдалості, оскільки вона досить сильно впливає на зручність його

застосування. Щоб визначити даний параметр, проводилось багато тестів і опитувань серед користувачів, відштовхуючись від отриманих даних було сформовано кінцеві вимоги для оформлення. Якщо слідувати рекомендаціям виробників Android пристроїв, що до дизайну додатків в їхній операційній системі, можна досить легко створити проект інтуїтивним.

Якщо структуру додатку не перевантажувати зайвими елементами, користувачеві буде легше в ньому орієнтуватися. Таку структуру можна створити дотримуючись внутрішньої логіки додатку, інтерфейс не потрібно наповнювати великою кількістю не обов'язкових полів введення, їх повинна бути мінімальна кількість, що забезпечить логіку програми. Правильне наповнення дозволить знизити поріг входження для користувачів.

Досить важливо, щоб додаток легко налаштовувався під потреби клієнта. Всі налаштування додатку мають знаходитись в одній окремій структурі, що призначена для цього. Тоді користувач зможе швидко змінити всі потрібні йому елементи під свої вимоги.

Правильне розміщення об'єктів у просторі є досить важливим, оскільки це допомагає користувачеві легко користуватись інтерфейсом, а також надає йому естетичний вигляд. Розроблений мною додаток повинен працювати в пристроях на операційній системі Android тому його дизайн повинен гармонічно виглядати в таких пристроях. Концепція такого дизайну має назву Material Design, його основна ознака – відгук на дотик користувача. Хоча вона здається примітивною, але складається з багатьох шарів. Ідея нагадує фізичну книгу, тобто користувачу доводиться переключати шари додатку.

Дизайнерські рішення операційної системи Android характеризуються адаптивною палітрою, що допомагає підібрати оптимальну кольорову гаму для оформлення додатку під потрібну тематику [21].

Ще однією особливістю Material design є анімація кнопок в інтерфейсі Android. Цей функціонал працює в всіх додатках на пристрої.



Головний екран додатку наведено на рисунку 4.3, екран з контактними даними наведено на рисунку 4.4, екран вибору кімнати 4.5, екран налаштування освітлення в обраній кімнаті наведено на рисунку. 4.6.



Рисунок 4.3 – Головний екран додатку

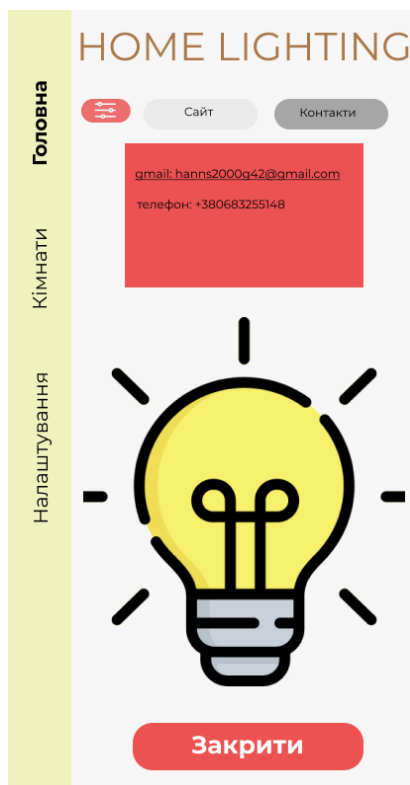


Рисунок 4.4 – Екран з контактними даними

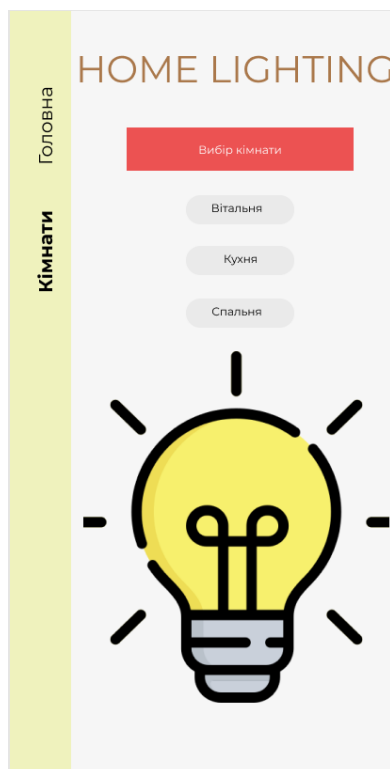


Рисунок 4.5 – Екран вибору кімнати

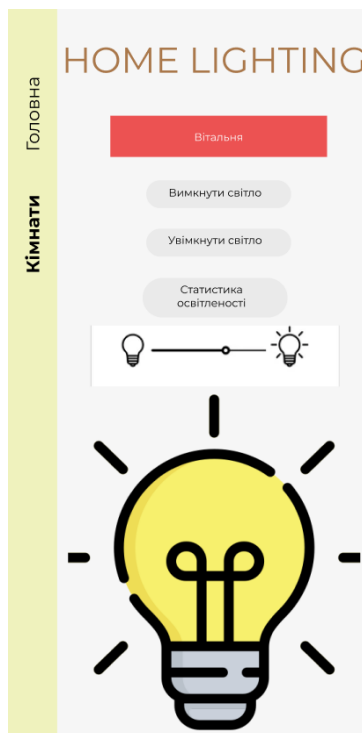


Рисунок 4.6 – Екран налаштування освітлення в кімнаті

Отже згідно перерахованими вище принципам і був сформований графічний інтерфейс мобільного клієнта. Як і планувалось він є простим і зрозумілим для користувача.

#### 4.4 Логіка подачі інформації в інтерфейсі

Правильність створених шляхів інтеграції, має вагоме значення для рівня інформативності додатку, а також впливають на сприйняття додатку користувачем в цілому. Вибір оптимального принципу подачі інформації в мобільному додатку досить сильно впливає на його рейтинг, а також є вагомим показником якості продукту [8].

Центральний контролер обробляє тільки інформацію з датчиків та запити, але не може відобразити графічний інтерфейс для управління процесами, тому для зручної взаємодії клієнта з системою було використано

додаток на пристрої Android. Статистика показує, що візуальна інформація в додатках сприймається користувачем на 60% краще порівняно з текстом. За статистикою, візуальна інформація сприймається на 60% краще, ніж текст, тому було прийнято рішення максимально використовувати графіки і діаграма.

Головна проблема, що з'явилась при реалізації даного підходу, є відсутність потрібних віджетів в Android SDK. Тому довилось шукати інформацію по даним типам view на сторонніх ресурсах [9].

Операційна система Android використовує логіку, що базується на інтерфейс-орієнтованому підході, саме тому центральним при створенні додатку є клас Activity з своїми атрибутами. Activity по суті відповідає за відображення окремої сторінки додатку, що поєднує xml-інтерфейс розмітки та логіку програмного забезпечення з алгоритмами, що базуються на елементах керування. Вона по суті є контейнером для виводу певних віджетів чи класів View.

Принцип побудови Material Design інтерфейсу базується на концепції карток, важливо щоб дизайн відповідав нормам виробника Android, тому було обрано динамічний шлях подання контенту в додатку. Програмно це можна реалізувати за допомогою застосування класу CardView та RecyclerView [11].

Віджет RecyclerView дозволяє легко та ефективно відображати великі набори даних. Можна надати дані та визначити, як виглядатиме кожен елемент, а бібліотека RecyclerView динамічно створює елементи, коли вони потрібні. RecyclerView переробляє ці окремі елементи. Коли елемент прокручується за межі екрана, RecyclerView не руйнує його подання. Замість цього RecyclerView повторно використовує подання для нових елементів, які прокручуються на екрані. Таке повторне використання значно покращує продуктивність, та реакцію програми та зменшуючи споживання енергії.

По суті віджет CardView імплементує картку, що є важливий для дизайну Material елементом. Для неї можна задавати радіус та заокруглення

кутів, а також обрати необхідний колір. Щоб його використати, достатньо вставити відповідний тег перед потрібним View в Xml-файлі [12].

Застосування внутрішнього ресурсу xml, щоб зберегти інформації про колір і мірки, має деякі переваг, серед яких централізація ресурсів та логіка їх зміни, напівтоновість кольорів, а також легка адаптація під відмінні співвідношення сторін. Також, можна використати теги XML для графічного View, аби мати можливість змінювати його зовнішній вигляд чи відображати статично через файл розмітки, не застосовуючи Java коду. Щоб реалізувати ці можливості були використано можливості класу Bundle для роботи з XML-параметрами. Вигляд View, для відображення графіків наведено на рис.4.7.

```
<view
    xmlns:graphic_view="http://schemas.android.com/apk/res/com.shometeam.ao.shome"
    android:layout_width="match_parent"
    android:layout_height="320dp"
    class="com.shometeam.ao.shome.GraphicViews.GraphicView"
    android:id="@+id/gr_view"
    android:layout_gravity="center_horizontal"
    graphic_view:mtext = "asd"
    graphic_view:mtype = "linear"
    graphic_view:mcolor = "red"/>
```

Рисунок 4.7 – View з власними XML тегами

Вигляд View, для відображення графіків наведено на рисунку 4.8, код xml-розмітки віджету графіку у обгортці CardView зображено на рисунку 4.9

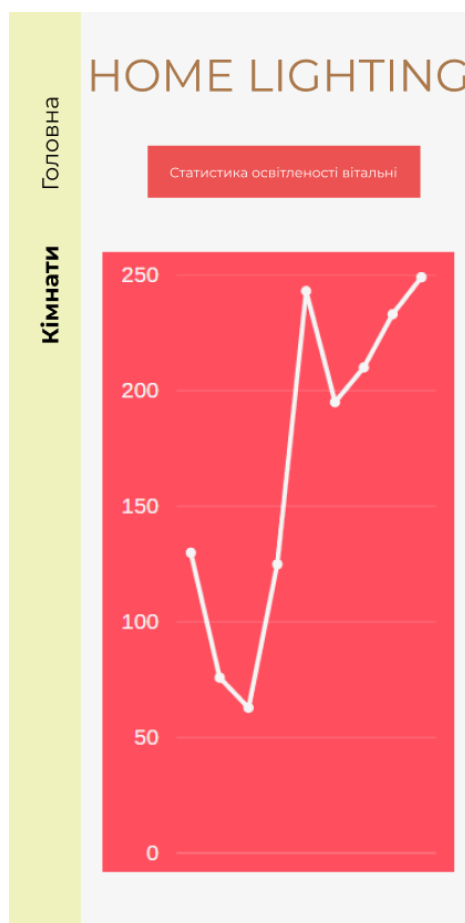


Рисунок 4.8 – View класс для відображення графіку

```

<android.support.v7.widget.CardView
    xmlns:card_view="http://schemas.android.com/apk/res-auto"
    android:id="@+id/card_view"
    android:layout_gravity="center"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_marginBottom="5dp"
    card_view:cardCornerRadius="5dp"
    card_view:contentPadding="10dp"
    card_view:cardElevation="2dp">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="380dp"
        android:orientation="vertical">

        <TextView
            android:layout_width="match_parent"
            android:layout_height="32dp"
            android:id="@+id/graphic_name"
            android:text="Humidity"
            android:layout_gravity="center"
            android:gravity="center"
            android:textAlignment="center"
            android:textSize="16sp"
            android:textColor="@color/text_color_light"
            android:elevation="4dp"
            android:background="@color/material_drawer_divider"
            />

        <view
            xmlns:graphic_view="http://schemas.android.com/apk/res/com.shometeam.ao.sh
            android:layout_width="match_parent"
            android:layout_height="320dp"
            class="com.shometeam.ao.shome.GraphicViews.GraphicView"
            android:id="@+id/gr_view"
            android:layout_gravity="center_horizontal"
            graphic_view:mtext = "asd"
            graphic_view:mtype = "linear"
            graphic_view:mcolor = "red"/>

    </LinearLayout>

</android.support.v7.widget.CardView>

```

Рисунок 4.9 – Код xml-розмітки віджету графіку у обгортці CardView

## 4.5 Обґрунтування вибору платформи для поширення додатку

Вибір платформи для поширення додатку є важливим етапом в розробці програмного продукту, її необхідно обрати, враховуючи багато факторів.

Google Play Market вважається однією з найкращих платформ для продажу Android-додатків, і це можна обґрунтувати декількома причинами:

- **Обширна аудиторія:** Google Play має найбільшу кількість активних користувачів серед магазинів додатків для Android. Це означає, що ваш додаток буде доступний для широкого кола користувачів, що може позитивно впливати на його популярність та завантаження [22].

- **Простота розміщення:** Процес розміщення додатків на Google Play досить простий та інтуїтивно зрозумілий. Розробники можуть легко завантажити свої додатки, оформити їхні сторінки та розпочати їхнє поширення.

- **Широкий функціонал:** Google Play пропонує розширений набір функцій для розробників, таких як можливість проведення різноманітних акцій, використання бета-тестувань, інструменти аналітики, а також можливість використовувати різні моделі монетизації.

- **Оновлення та підтримка:** Google регулярно вдосконалює та оновлює свій магазин додатків, надаючи розробникам нові можливості та користувачам — зручніші інструменти. Це дозволяє тримати додаток актуальним та враховувати останні тенденції ринку.

- **Гнучкість у монетизації:** Google Play дозволяє використовувати різні моделі монетизації, такі як оплата за додатковий контент, реклама, внутрішні покупки, або підписки. Це дає можливість розробникам вибрати той варіант, який найкраще підходить для їхнього продукту.

- **Високий рівень довіри:** Google Play є офіційним магазином додатків для Android, що створює високий рівень довіри серед користувачів.



Користувачі зазвичай вірять додаткам, розміщеним у цьому магазині, що сприяє успіху додатків.

Загалом, Google Play Market володіє великим сприятливим середовищем для розробки, розміщення та успішної монетизації Android-додатків.

#### **4.6 Висновки до четвертого розділу**

Мобільний додаток, що був розроблений для взаємодії з автоматизованою системою управління освітленням, є досить зручним при використанні. Його можна встановити майже на будь який пристрій з операційною системою Android.

Функціонал, що пропонується додатком, дозволяє проводити повноцінне керування всім обладнанням, навіть на відстані чи здійснювати моніторинг рівня освітлення в кімнатах та на вулиці. За потреби, можливості додатку можна розширити або змінити використавши середовища розробки Android Studio. Також дані нароби можуть слугувати для створення подібних додатків в подальшому, наприклад для організації управління розумним будинком.

## 5. ТЕСТУВАННЯ РОЗРОБЛЕНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

### 5.1 Вибір способу тестування програми

Зазвичай під час тестування програмного забезпечення протиставляються два шляхи ручне та автоматизоване тестування. Обидва засоби перевірки програмного забезпечення мають переваги та недоліки.

Основні переваги автоматизації тестування - швидкість та відсутність людського фактору - зробили широким його використання. Здається, що автоматизація застосовується для виконання все більшої кількості задач. Може здаватися, що вона може повністю витіснити роботу мануальних тестувальників у найближчому майбутньому [27].

Але спеціалісти, які здійснюють тестування веб-сайтів, десктопне та мобільне тестування, стверджують, що цього ніколи не станеться, тому що в деяких аспектах жоден інструмент не може зрівнятися з інженером з тестування.

Аспекти, в яких ручне тестування ПЗ краще автоматизованого:

- Тестувальники можуть вирішити перевірити області або виконати тести, які не враховані та не автоматизовані, і таким чином збільшити тестове покриття та знайти помилки, які б не знайшов інструмент;
- Тестувальники бачать дефекти, які не були передбачені в автоматизованих тестах. Все передбачити неможливо - інженери з тестування можуть імпровізувати, а інструменти такого не роблять;
- Інструменти не мають почуттів, які мають кінцеві користувачі, тому вони не можуть розраховувати на попередній досвід та інтуїцію. Тільки люди можуть повністю оцінити програму, а інструмент лише визначає, чи проходить програма тести.

Автоматизоване тестування передбачає таку взаємодію між програмним продуктом та інструментами тестування, за якої втручання людини зводиться до мінімуму.

Інструменти автоматизації корисні для багатьох видів діяльності. Вони дозволяють заощадити масу часу, виконуючи регресійне, модульне, навантажувальне та деякі інші види тестування [28].

Процеси, які можуть бути автоматизованими:

- регресійне тестування;
- димове тестування;
- тестування продуктивності;
- тестування навантаження;
- тестування методом білого ящика (за допомогою інструментів модульного тестування);
- аналіз покриття коду.

Але найбільшою помилкою розробників є автоматизація не тих процесів під час розробки програмного забезпечення. Не все можна запрограмувати та виконати автоматично.

Що ніколи не слід автоматизувати:

- нестабільне програмне забезпечення. Якщо програмний продукт знаходиться на стадії розробки та піддається змінам, автоматизоване тестування не буде доречним;
- сценарії тестування, які рідко використовуються. Не слід автоматизувати ті сценарії, що будуть виконуватися рідко;
- аналіз програмного кода і документації. Автоматизація цих процесів може в подальшому тільки додати клопоту.

Зазвичай для автоматизації найбільше підходять задачі, що повторюються. Саме тому деякі компанії вважають за краще спочатку провести ручне тестування, щоб визначити ті процеси, які виконуватимуться неодноразово, для їх подальшої автоматизації.

Аспекти, в яких автоматизоване тестування ПЗ краще за ручне:

- написаний скрипт можна виконувати скільки завгодно разів;
- автотести виконуються з більшою точністю, так як людський фактор виключений;
- час виконання автотестів набагато менший, ніж виконання тих самих перевірок вручну;
- тестовий прогін можна запланувати на будь-який час, запустити за розкладом, щоб він виконувався без участі людини.

Найбільш підходящі сценарії, до яких можна застосувати автоматизацію:

- важкодоступні місця в системі;
- функціональність, яка часто використовується, в якій ризики від помилок достатньо високі;
- рутинні операції, такі як перебори даних;
- перевірка валідаційних повідомлень;
- довгі end-to-end сценарії;
- перевірка даних, що потребують точних математичних розрахунків.

Обидва розглянуті підходи до тестування програмного забезпечення мають як певний перелік важливих переваг, так і недоліки. В більшості проектів прийнято використовується комбінацію з ручного та автоматизованого тестування, також рівень автоматизації часто залежить як від призначення проекту, так і від особливостей виробничих процесів в компанії. Комбінація обох видів забезпечить найвищу якість тестування та дозволить отримати максимальний результат.

## **5.2 Розробка тест кейсів**

Згідно до вимог продукту було проведено тестування програмного забезпечення та мобільного додатку. Для забезпечення якості програми необхідно створити сценарії тест кейсів для тестування головних

можливостей додатку [29]. Най важливіші тест кейси для перевірки роботи програми представлені у таблиці 5.1.

Таблиці 5.1 - Сценарії для перевірки функціоналу

Назва тесту	Крок	Очікування
Перевірка роботи програми на версіях Android не нижче 9.	Встановити програму на пристрій із Android 9 та вище, запустити додаток та перевірити відображення елементів інтерфейсу.	Додаток запускається з меню пристрою. Всі елементи інтерфейсу відображаються та функціонують коректно.
Перевірка програми на стабільність роботи.	Використати мобільний додаток аби включити та виключити усі підконтрольні освітлювальні прилади.	Освітлювальні прилади мають коректно включатись та виключатись реагуючи на сигнали з додатку.
Перевірка на функціональність системи.	Використати мобільний додаток аби змінювати налаштування усіх підконтрольних освітлювальних пристроїв.	Освітлювальні прилади мають правильно реагувати на сигнали з додатку змінюючи параметри освітлювання.
Перевірка коректності роботи системи вияву присутності.	Людина мають проявляти рухову активність в підконтрольних кімнатах аби викликати реакцію датчиків руху.	Датчики руху мають реагувати на появу людини та світло повинно викатись. Датчики присутності мають відчувати

	Людина має нерухомо знаходитись в підконтрольній кімнаті аби перевірити роботу датчиків присутності.	присутність людини та система має тримати світло включеним. Після покидання людиною кімнати усі датчики повинні відреагувати та світло має вимкнутись.
Перевірка системи моніторингу інтенсивності освітлення	Встановити в підконтрольних кімнатах пристрої для моніторингу інтенсивності освітлення(люксометр). Штучно змінювати інтенсивність освітлення в кімнатах, після чого порівняти дані з пристроїв для моніторингу та результати що зафіксувались в додатку.	Дані про інтенсивність освітлення в підконтрольних кімнатах зафіксовані люксометром та дані що потрапили і відобразились в додатку повинні співпадати.

Перевірка роботи програми на актуальних версіях Android показала, що вона коректно працює на більшості доступних на ринку пристроях. Але є не великі незручності при використанні смартфонів з маленьким екраном, оскільки елементи інтерфейсу на ньому відображаються досить не зручно а текст доводиться приближувати.

Тестування системи що фіксують присутність людини показало хороші результати в вияві людей різної комплекції та при різному рівні їхньої активності. Були виявлені певні особливості при роботі датчиків присутності, вони досить успішно реагують на присутність тварин. Такий функціонал може бути як корисним так і не бажаним для користувача, але за потреби це можна змінити при налаштуванні під потреби власника.

### **5.3 Висновок**

В рамках даному розділі було проаналізовано різні види тестування програмного забезпечення та обґрунтовано найкращий спосіб тестування мобільного додатку. Створено сценарії для тест кейсів та описано процес тестування програмного забезпечення. Під час тестування, було перевірено стабільність роботи програми на різних моделях смартфонів. Тест показав стабільність роботи додатку з різними версіями Android, але були виявлені певні незручності при використанні інтерфейсу на смартфонах с маленькими екранами. Також було проведено тестування функціональності системи та управлінських можливостей програми. За допомогою мобільного додатку вдалось здійснити повний контроль над освітлювальним обладнанням та перевірити функціонал всієї системи контролю освітлення.

## 6. ЕКОНОМІЧНИЙ РОЗДІЛ

### 6.1 Технологічний аудит розробленої автоматизованої системи управління освітленням в приміщеннях

Як було зазначено раніше, зростання кількості електричних ламп, які використовуються в сучасних приміщеннях для їх освітлення, спричиняє зростання кількості спожитої електроенергії (та інших ресурсів) і як наслідок – збільшення потреби в нових електричних потужностях. Тому актуальним постало питання забезпечення економії електричної енергії (особливо в умовах військової агресії росії), організації чіткого обліку спожитих ресурсів, підвищення контролю за використанням електроенергії, раціоналізації споживання електричної енергії протягом доби, попередження несанкціонованого відбору електроенергії тощо.

Розв'язати це питання у ручному режимі просто неможливо. Тому все більшої популярності набуває впровадження в приміщеннях автоматизованих систем управління освітленням, застосування яких дозволяє оперативно приймати рішення стосовно регулювання споживання електроенергії, зменшуючи тим самим обсяги її споживання та відповідні витрати.

Тому метою виконаної нами магістерської кваліфікаційної роботи була розробка автоматизованої системи управління освітленням в приміщеннях, а також удосконалення обліку та контролю за споживанням електричної енергії.

Для досягнення поставленої мети було проведено огляд науково-літературних джерел за тематикою досліджуваного об'єкту; сформульовано та описано вимоги до програмної частини; визначено технологію та підхід до її реалізації; створено тестові сценарії для тестування системи; описано архітектуру розробленого додатку з висновками, його переваги та недоліки тощо



В результаті виконаної роботи було розроблено сучасне програмне забезпечення та відповідні алгоритми.

Для встановлення комерційного потенціалу розробленої нами автоматизованої системи управління освітленням в приміщеннях було запрошено 3-х відомих експертів – кандидатів технічних наук, доцентів Кулика Я.А. та Кривогубченка С.Г., а також професора Папінова В.М. Визначення комерційного потенціалу розробленої автоматизованої системи управління освітленням в приміщеннях було здійснено за критеріями, наведеними в таблиці 6.1.

Таблиця 6.1 – Рекомендовані критерії оцінювання рівня комерційного потенціалу будь-якої розробки і їх бальна оцінка

Критерії оцінювання та бали (за 5-ти бальною шкалою)					
	0	1	2	3	4
Технічна здійсненність концепції:					
1	Достовірність концепції не підтверджена	Концепція підтверджена експертними висновками	Концепція підтверджена розрахунками	Концепція перевірена на практиці	Перевірено роботоздатність продукту в реальних умовах
Ринкові переваги (недоліки):					
2	Багато аналогів на малому ринку	Мало аналогів на малому ринку	Кілька аналогів на великому ринку	Один аналог на великому ринку	Продукт не має аналогів на великому ринку
3	Ціна продукту значно вища за ціни аналогів	Ціна продукту дещо вища за ціни аналогів	Ціна продукту приблизно дорівнює цінам аналогів	Ціна продукту дещо нижче за ціни аналогів	Ціна продукту значно нижче за ціни аналогів
4	Технічні та споживчі властивості продукту значно гірші, ніж в аналогів	Технічні та споживчі властивості продукту трохи гірші, ніж в аналогів	Технічні та споживчі властивості продукту на рівні аналогів	Технічні та споживчі властивості продукту трохи кращі, ніж в аналогів	Технічні та споживчі властивості продукту значно кращі, ніж в аналогів
Ринкові перспективи					
5	Експлуатаційні витрати значно	Експлуатаційні витрати	Експлуатаційні витрати	Експлуатаційні витрати трохи	Експлуатаційні витрати значно

	вищі, ніж в аналогів	дещо вищі, ніж в аналогів	на рівні експлуатаційних витрат аналогів	нижчі, ніж в аналогів	нижчі, ніж в аналогів
6	Ринок малий і не має позитивної динаміки	Ринок малий, але має позитивну динаміку	Середній ринок з позитивною динамікою	Великий стабільний ринок	Великий ринок з позитивною динамікою
7	Активна конкуренція великих компаній на ринку	Активна конкуренція	Помірна конкуренція	Незначна конкуренція	Конкурентів немає
8	Відсутні фахівці як з технічної, так і з комерційної реалізації ідеї	Необхідно наймати фахівців або витратити значні кошти та час на навчання наявних фахівців	Необхідне незначне навчання фахівців та збільшення їх штату	Необхідне незначне навчання фахівців	Є фахівці з питань як з технічної, так і з комерційної реалізації ідеї
9	Потрібні значні фінансові ресурси, які відсутні. Джерела фінансування ідеї відсутні	Потрібні незначні фінансові ресурси. Джерела фінансування відсутні	Потрібні значні фінансові ресурси. Джерела фінансування є	Потрібні незначні фінансові ресурси. Джерела фінансування є	Не потребує додаткового фінансування
10	Необхідна розробка нових матеріалів	Потрібні матеріали, що використовуються у військово-промисловому комплексі	Потрібні дорогі матеріали	Потрібні досяжні та дешеві матеріали	Всі матеріали для реалізації ідеї відомі та давно використовуються у виробництві
11	Термін реалізації ідеї більший за 10 років	Термін реалізації ідеї більший за 5 років. Термін окупності інвестицій більше 10-ти років	Термін реалізації ідеї від 3-х до 5-ти років. Термін окупності інвестицій більше 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій від 3-х до 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій менше 3-х років
12	Необхідна розробка	Необхідно отримання	Процедура отримання	Необхідно тільки	Відсутні будь-які регламентні

регламентних документів та отримання великої кількості дозвільних документів на виробництво та реалізацію продукту	великої кількості дозвільних документів на виробництво та реалізацію продукту, що вимагає значних коштів та часу	дозвільних документів для виробництва та реалізації продукту вимагає незначних коштів та часу	повідомлення відповідним органам про виробництво та реалізацію продукту	обмеження на виробництво та реалізацію продукту
--	--	---	---	---

Запрошені нами шановні експерти оцінили розроблену нами автоматизовану систему управління освітленням в приміщеннях таким чином (див. таблицю 6.2):

Таблиця 6.2 – Результати технологічного аудиту розробленої автоматизованої системи управління освітленням в приміщеннях (за шкалою оцінювання 0-1-2-3-4)

Критерії	Прізвище, ініціали експертів		
	Кулик А.Я.	Кривогубченко С.Г.	Папінов В.М.
	Бали, що їх виставили експерти:		
1	4	3	3
2	4	3	4
3	3	3	3
4	4	4	4
5	3	3	3
6	4	4	3
7	3	3	4
8	3	3	3
9	4	4	3
10	3	3	4
11	4	4	3
12	3	3	4
Сума балів	СБ <sub>1</sub> = 42	СБ <sub>2</sub> = 40	СБ <sub>3</sub> = 41
Середньоарифметична сума балів $\overline{СБ}$	$\overline{СБ} = \frac{\sum_{i=1}^3 СБ_i}{3} = \frac{42 + 40 + 41}{3} = \frac{123}{3} = 41,00$		

Для встановлення комерційного потенціалу розробленої нами автоматизованої системи управління освітленням в приміщеннях скористуємося рекомендаціями, які наведено в таблиці 6.3 [30].

Таблиця 6.3 – Рівні комерційного потенціалу будь-якої наукової розробки

Середньоарифметична сума балів $\overline{СБ}$ , розрахована на основі висновків експертів	Рівень комерційного потенціалу розробки
0 – 10	Низький
11 – 20	Нижче середнього
21 – 30	Середній
31 – 40	Вище середнього
41 – 48	Високий

Оскільки середньоарифметична сума балів, що їх виставили експерти, складає 41,00 балів, то це свідчить, що розроблена нами автоматизована система управління освітленням в приміщеннях має рівень комерційного потенціалу, який вважається «високим».

Це пояснюється тим, що розроблена нами автоматизована система управління освітленням в приміщеннях базується на сучасних підходах, методах та алгоритмах, що дозволяє більш ефективно організувати процес управління, обліку та аналізу споживання електроенергії в приміщеннях.

## **6.2 Розрахунок витрат на розроблення автоматизованої системи управління освітленням в приміщеннях**

При розробленні автоматизованої системи управління освітленням в приміщеннях були зроблені певні витрати. Зокрема:

А). Основна заробітна плата  $Z_0$  розробників, яка визначається за формулою:

$$Z_o = \frac{M}{T_p} \cdot t \text{ грн,} \quad (6.1)$$

де  $M$  – місячний посадовий оклад розробника, грн; прийmemo, що

$M = (6700 \dots 23000)$  грн/місяць;

$T_p$  – число робочих днів в місяці; прийmemo  $T_p = 21$  день;

$t$  – число днів роботи розробників.

Зроблені розрахунки зведемо до таблиці 6.4:

Таблиця 6.4 – Основна заробітна плата розробників

Найменування посади виконавця	Місячний посадовий оклад, грн	Оплата за робочий день, грн	Число днів роботи	Витрати на оплату праці, грн
1. Науковий керівник магістерської роботи	19700	938,09	20 годин	$\approx 3127$
2. Магістрант-студент-виконавець	2000 (беремо 6700)	319,04	77	$\approx 24566$
3. Консультант з економічної частини	17650	840,48	1,5 години	$\approx 210$ (при 6-годинному робочому дні)
Загалом				$Z_o = 27903$ грн

Б). Додаткова заробітна плата  $Z_d$  розробників розраховується як (10...12)% від величини їх основної заробітної плати, тобто:

$$Z_d = \alpha \cdot Z_o = (0,1 \dots 0,12) \cdot Z_o. \quad (6.2)$$

Прийmemo, що  $\alpha = 0,11$ . Тоді для нашого випадку отримаємо:

$$Z_d = 0,11 \times 27903 = 3069,33 \approx 3070 \text{ грн.}$$

В). Нарахування на заробітну плату  $НЗП_{зп}$  розробників (дослідників) розраховуються за формулою:

$$НЗП_{зп} = (Z_o + Z_d) \cdot \frac{\beta}{100}, \quad (6.3)$$

де  $\beta$  – ставка обов'язкового єдиного внеску на державне соціальне страхування, %.  $\beta = 22\%$ . Тоді:

$$\text{НЗН}_{\text{зп}} = (27903 + 3070) \times 0,22 = 6814,06 \approx 6815 \text{ грн.}$$

Г). Амортизація основних засобів А, які використовувались під час виконання цієї роботи:

$$A = \frac{\text{Ц} \cdot \text{Н}_a}{100} \cdot \frac{\text{Т}}{12} \text{ грн,} \quad (6.4)$$

де Ц – загальна балансова вартість основних засобів, грн;

$\text{Н}_a$  – річна норма амортизаційних відрахувань. Для нашого випадку можна прийняти, що  $\text{Н}_a = (2,5 \dots 25)\%$ ;

Т – термін використання основних засобів, місяці.

Зроблені розрахунки зведено в таблицю 6.5.

Таблиця 6.5 – Розрахунок амортизаційних відрахувань

Найменування обладнання, приміщень тощо	Балансова вартість, грн.	Норма амортизації, %	Термін використання, міс.	Величина амортизаційних відрахувань, грн
1. Комп'ютерна техніка, обладнання тощо	74000	25	3,2 (при 70% використанні)	3453,33 $\approx$ 3454
2. Приміщення університету, кафедри	32000	3,5	3,2 при 20% використанні	59,73 $\approx$ 60
Всього				A = 3514 грн

Д). Витрати на матеріали М розраховуються за формулою:

$$M = \sum_1^n \text{Н}_i \cdot \text{Ц}_i \cdot \text{К}_i - \sum_1^n \text{В}_i \cdot \text{Ц}_в \text{ грн.,} \quad (6.5)$$

де  $\text{Н}_i$  – витрати матеріалу  $i$ -го найменування, кг;  $\text{Ц}_i$  – вартість матеріалу  $i$ -го найменування;  $\text{К}_i$  – коефіцієнт транспортних витрат,  $\text{К}_i = (1,1 \dots 1,15)$ ;  $\text{В}_i$  –

маса відходів матеріалу  $i$ -го найменування;  $\Pi_v$  – ціна відходів матеріалу  $i$ -го найменування;  $n$  – кількість видів матеріалів.

Е). Витрати на комплектуючі  $K$  розраховуються за формулою:

$$K = \sum_1^n N_i \cdot \Pi_i \cdot K_i \text{ грн,} \quad (6.6)$$

де  $N_i$  – кількість комплектуючих  $i$ -го виду, шт.;  $\Pi_i$  – ціна комплектуючих  $i$ -го виду;  $K_i$  – коефіцієнт транспортних витрат,  $K_i = (1, 1 \dots 1, 15)$ ;  $n$  – кількість видів комплектуючих.

Під час виконання роботи загальні витрати на матеріали та комплектуючі склали приблизно 2200 грн.

Ж). Витрати на силову електроенергію  $V_e$  розраховуються за формулою:

$$V_e = \frac{V \cdot \Pi \cdot \Phi \cdot K_{\Pi}}{K_d}, \quad (6.7)$$

де  $V$  – вартість 1 кВт-год. електроенергії, в 2023 р.  $V \approx 4,5$  грн/кВт;

$\Pi$  – установлена потужність обладнання, кВт;  $\Pi = 1,25$  кВт;

$\Phi$  – фактична кількість годин роботи обладнання, годин.

Прийmemo, що  $\Phi = 315$  годин;

$K_{\Pi}$  – коефіцієнт використання потужності;  $K_{\Pi} < 1 = 0,82$ .

$K_d$  – коефіцієнт корисної дії,  $K_d = 0,78$ .

Тоді витрати на силову електроенергію будуть дорівнювати:

$$V_e = \frac{V \cdot \Pi \cdot \Phi \cdot K_{\Pi}}{K_d} = \frac{4,5 \cdot 1,25 \cdot 315 \cdot 0,82}{0,78} = 1862,74 \approx 1863 \text{ грн.}$$

И). Інші витрати  $V_{\text{інш}}$  можна прийняти як (50...300)% від основної заробітної плати розробників, тобто:

$$V_{\text{інш}} = (0,5 \dots 3) \times 3_0. \quad (6.8)$$

Для нашого випадку отримаємо:

$$V_{\text{інш}} = 1,20 \times 27903 = 33483,60 \approx 33484 \text{ грн.}$$

К). Сума всіх попередніх статей витрат складає витрати на виконання роботи безпосередньо розробником-магістрантом – В.

$$B = 27903 + 3070 + 6815 + 3514 + 2200 + 1863 + 33484 = 78849 \text{ грн.}$$

Л). Загальні витрати на розробку автоматизованої системи управління освітленням в приміщеннях  $B_{\text{заг}}$  розраховуються за формулою:

$$B_{\text{заг}} = \frac{B}{\beta}, \quad (6.9)$$

де  $\beta$  – коефіцієнт, який характеризує етап (стадію) виконання цієї роботи. Можна прийняти, що,  $\beta \approx 0,9$  [30], оскільки робота практично завершена.

$$\text{Тоді: } B_{\text{заг}} = \frac{78849}{0,9} = 87610,00 \text{ грн або приблизно 88 тисяч грн.}$$

Тобто прогнозовані загальні витрати на розробку автоматизованої системи управління освітленням в приміщеннях становлять приблизно 88 тисяч грн.

### **6.3 Розрахунок економічного ефекту від можливої комерціалізації нашої розробки**

Економічний ефект від впровадження та можливої комерціалізації розробленої автоматизованої системи управління освітленням в приміщеннях пояснюється її значно кращими функціональними можливостями. Тому нашу розробку можна реалізовувати на ринку дещо дорожче, ніж аналогічні за функціями розробки. Так, якщо подібна за функціями автоматизована система у 2023 році коштувала на ринку приблизно 5,2 тисяч грн, то нашу розробку можна буде реалізовувати на ринку приблизно за 5,5 тисяч грн або на 0,3 тисяч грн дорожче.

Аналіз місткості ринку показує, що на сьогодні в Україні попит на подібні системи автоматизованого управління освітленням в приміщеннях може бути дуже великим. Це, насамперед, численні домогосподарства,



заклади харчування, офісні приміщення тощо. Тому можна очікувати зростання попиту на нашу розробку принаймні протягом 3-х років після її впровадження.

Тобто, якщо наша розробка буде впроваджена з 1 січня 2024 року, то її результати будуть виявлятися протягом 2024-го, 2025-го та 2026-го років.

Прогноз зростання попиту на нашу розробку складає по роках:

2023 рік – базовий попит 1000 шт.

а) 2024 р. – приблизно +400 шт. до базового року;

б) 2025 р. – +700 шт. до базового року;

в) 2026 р. – +1000 шт. до базового року.

Можливе збільшення чистого прибутку  $\Delta\Pi_i$ , що його може отримати потенційний інвестор від комерціалізації, тобто виведення нашої розробки на ринок, становитиме:

$$\Delta\Pi_i = \sum_1^n (\Delta C_o \cdot N + C_o \cdot \Delta N)_i \cdot \lambda \cdot \rho \cdot \left(1 - \frac{v}{100}\right), \quad (6.10)$$

де  $\Delta C_o$  – покращення основного якісного показника від впровадження результатів нашої розробки у цьому році. Для нашого випадку це є збільшення ціни реалізації нашої розробки  $\Delta C_o = (5,5 - 5,2) = +0,3$  тисячі грн;

$N$  – основний кількісний показник, який визначає обсяг діяльності у році до впровадження результатів розробки;  $N = 1000$  шт.;

$\Delta N$  – покращення основного кількісного показника від впровадження результатів розробки. Таке покращення становитиме: у 2024 році – + 400 шт., у 2025 році – +700 шт., у 2026 році – + 1000 шт. (відносно базового 2023 року);

$C_o$  – основний якісний показник (тобто ціна), який визначає обсяг діяльності у році після впровадження результатів розробки, грн;  $C_o = 5,5$  тисяч грн;

$n$  – кількість років, протягом яких очікується отримання позитивних результатів від впровадження розробки; для нашого випадку  $n = 3$ ;

$\lambda$  – коефіцієнт, який враховує сплату податку на додану вартість;  
 $\lambda = 0,8333$ ;

$\rho$  – коефіцієнт, який враховує рентабельність продукту.  
 Рекомендується приймати  $\rho = (0,2...0,5)$ ; візьмемо  $\rho = 0,5$ ;

$\nu$  – ставка податку на прибуток. У 2023 і наступних роках  $\nu = 18\%$ .

Тоді можливе зростання чистого прибутку  $\Delta\Pi_1$  для потенційного інвестора протягом першого року від можливого впровадження нашої розробки (2024 р.) складе:

$$\Delta\Pi_1 = [0,3 \cdot 1000 + 5,5 \cdot 400] \cdot 0,8333 \cdot 0,5 \cdot \left(1 - \frac{18}{100}\right) \approx 854 \text{ тис. грн.}$$

Можливе зростання чистого прибутку  $\Delta\Pi_2$  для потенційного інвестора від можливого впровадження нашої розробки протягом другого (2025 р.) року складе:

$$\Delta\Pi_2 = [0,3 \cdot 1000 + 5,5 \cdot 700] \cdot 0,8333 \cdot 0,5 \cdot \left(1 - \frac{18}{100}\right) \approx 1418 \text{ тис. грн.}$$

Можливе зростання чистого прибутку  $\Delta\Pi_3$  для потенційного інвестора від можливого впровадження нашої розробки протягом третього (2026 р.) року складе:

$$\Delta\Pi_3 = [0,3 \cdot 1000 + 5,5 \cdot 1000] \cdot 0,8333 \cdot 0,5 \cdot \left(1 - \frac{18}{100}\right) \approx 1982 \text{ тис. грн.}$$

Приведена вартість зростання всіх чистих прибутків від можливого впровадження нашої розробки становитиме:

$$\text{ПП} = \sum_1^t \frac{\Delta\Pi_i}{(1 + \tau)^t}, \quad (6.11)$$

де  $\Delta\Pi_i$  – збільшення чистого прибутку у кожному із років, протягом яких виявляються результати виконаної та впровадженої роботи, грн;

$t$  – період часу, протягом якого виявляються результати впровадженої роботи, роки. Для нашого випадку  $t = 3$  роки;

$\tau$  – ставка дисконтування. Приймемо  $\tau = 0,10$  (10%);

$t$  – період часу від моменту початку розроблення автоматизованої системи до моменту отримання можливих чистих прибутків потенційним інвестором.

Тоді приведена вартість зростання всіх можливих чистих прибутків ПП, що їх може отримати потенційний інвестор від комерціалізації нашої розробки, складе:

$$\text{ПП} = \frac{854}{(1+0,1)^2} + \frac{1418}{(1+0,1)^3} + \frac{1982}{(1+0,1)^4} \approx 706 + 1066 + 1354 = 3126 \text{ тисяч грн.}$$

Теперішня вартість інвестицій  $PV$ , що повинні бути вкладені для реалізації нашої розробки:  $PV = (1,0\dots5) \times V_{\text{заг}}$ .

Для нашого випадку  $PV = (1,0\dots5,0) \times 88 = 5 \times 88 = 440$  тисяч грн.

Розраховуємо абсолютний ефект від можливих вкладених інвестицій  $E_{\text{абс}}$ .

$$E_{\text{абс}} = \text{ПП} - PV, \quad (6.12)$$

де ПП – приведена вартість збільшення всіх чистих прибутків для інвестора від можливого впровадження нашої розробки, грн;

$PV$  – теперішня вартість інвестицій  $PV = 440$  тисяч грн.

Абсолютний ефект від можливого впровадження нашої розробки (при прогнозованому ринку збуту) за три роки складе:

$$E_{\text{абс}} = 3126 - 440 = 2686 \text{ тисяч грн.}$$

Оскільки  $E_{\text{абс}} > 0$ , то комерціалізація нашої розробки може бути доцільною.

Далі розрахуємо внутрішню дохідність  $E_v$  вкладених інвестицій:

$$E_v = T_{\text{ж}} \sqrt[1 + \frac{E_{\text{абс}}}{PV}]{} - 1, \quad (6.13)$$

де  $E_{\text{абс}}$  – абсолютний ефект вкладених інвестицій;  $E_{\text{абс}} = 2686$  тис. грн;

$PV$  – теперішня вартість початкових інвестицій  $PV = 440$  тис. грн;

$T_{\text{ж}}$  – життєвий цикл розробки, роки.

$T_{\text{ж}} = 4$  років (2023-й, 2024-й, 2025-й, 2026-й роки)

Для нашого випадку отримаємо:

$$E_B = \sqrt[4]{1 + \frac{2686}{440}} - 1 = \sqrt[4]{1 + 6,1045} - 1 = \sqrt[4]{7,1045} - 1 = 1,633 - 1 = 0,633 = 63,3\%.$$

Далі визначимо ту мінімальну дохідність, нижче за яку потенційному інвестору не вигідно буде займатися комерціалізацією нашої розробки.

Мінімальна дохідність або мінімальна (бар'єрна) ставка дисконтування  $\tau_{\text{мін}}$  визначається за формулою:

$$\tau_{\text{мін}} = d + f, \quad (6.14)$$

де  $d$  – середньозважена ставка за депозитними операціями в комерційних банках; в 2023 році в Україні  $d = (0,10 \dots 0,12)$ ;

$f$  – показник, що характеризує ризикованість вкладень;  $f = (0,05 \dots 0,30)$ .

Для нашого випадку отримаємо:

$$\tau_{\text{мін}} = 0,12 + 0,30 = 0,42 \text{ або } \tau_{\text{мін}} = 42\%.$$

Оскільки величина  $E_B = 63,3\% > \tau_{\text{мін}} = 42\%$ , то потенційний інвестор у принципі може бути зацікавлений у фінансуванні та комерціалізації розробленої нами автоматизованої системи управління освітленням в приміщеннях.

Далі розраховуємо термін окупності коштів, вкладених у можливу комерціалізацію розробленої автоматизованої системи управління освітленням в приміщеннях.

Термін окупності  $T_{\text{ок}}$  розраховується за формулою:

$$T_{\text{ок}} = \frac{1}{E_B}. \quad (6.15)$$

Для нашого випадку термін окупності  $T_{\text{ок}}$  коштів становитиме:

$$T_{\text{ок}} = \frac{1}{0,633} = 1,58 \text{ років} < 3 \text{ років},$$

що свідчить про потенційну доцільність комерціалізації розробленої нами автоматизованої системи управління освітленням в приміщеннях.

Далі проведено моделювання залежності величини внутрішньої дохідності вкладених потенційним інвестором коштів в комерціалізацію

розробленої нами автоматизованої системи управління освітленням в приміщеннях від рівня інфляції в країні.

Як відомо, деякі фахівці прогнозують зростання рівня інфляції (в межах 30% і вище), що обумовлюється військовою агресією росії проти України.

Прийнявши рівень інфляції у 30% отримаємо:

$$\text{ПП} = \frac{854}{(1+0,3)^2} + \frac{1418}{(1+0,3)^3} + \frac{1982}{(1+0,3)^4} \approx 505 + 646 + 694 = 1845 \text{ тисяч грн.}$$

Тоді абсолютний ефект від можливого впровадження нашої розробки за три роки складе:

$$E_{\text{абс}} = 1845 - 440 = 1405 \text{ тисяч грн.}$$

Внутрішня дохідність  $E_v$  вкладених інвестицій становитиме:

$$E_v = \sqrt[4]{1 + \frac{E_{\text{абс}}}{PV}} - 1,$$

де  $E_{\text{абс}}$  – абсолютний ефект вкладених інвестицій;  $E_{\text{абс}} = 1405$  тисяч грн;

$PV$  –теперішня вартість початкових інвестицій  $PV = 440$  тисяч грн.

Для нашого випадку отримаємо:

$$E_v = \sqrt[4]{1 + \frac{1405}{440}} - 1 = \sqrt[4]{1 + 3,1932} - 1 = \sqrt[4]{4,1932} - 1 = 1,431 - 1 = 0,431 = 43,1\%.$$

Оскільки величина  $E_v = 43,1\% > \tau_{\text{мін}} = 42\%$ , то потенційний інвестор у принципі також може бути зацікавлений у фінансуванні та комерціалізації розробленої нами автоматизованої системи управління освітленням в приміщеннях.

Прийнявши рівень інфляції у 50% отримаємо:

$$\text{ПП} = \frac{854}{(1+0,5)^2} + \frac{1418}{(1+0,5)^3} + \frac{1982}{(1+0,5)^4} \approx 380 + 420 + 392 = 1192 \text{ тисяч грн.}$$

Тоді абсолютний ефект від можливого впровадження нашої розробки за три роки складе:

$$E_{\text{абс}} = 1192 - 440 = 752 \text{ тисяч грн.}$$

Внутрішня дохідність  $E_v$  вкладених інвестицій становитиме:

$$E_B = \sqrt[4]{1 + \frac{E_{\text{абс}}}{PV}} - 1,$$

де  $E_{\text{абс}}$  – абсолютний ефект вкладених інвестицій;  $E_{\text{абс}} = 752$  тисяч грн;  
 $PV$  – теперішня вартість початкових інвестицій  $PV = 440$  тисяч грн.

Для нашого випадку отримаємо:

$$E_B = \sqrt[4]{1 + \frac{752}{440}} - 1 = \sqrt[4]{1 + 1,7091} - 1 = \sqrt[4]{2,7091} - 1 = 1,283 - 1 = 0,283 = 28,3\%.$$

Оскільки величина  $E_B = 28,3\% < \tau_{\text{мін}} = 42\%$ , то потенційний інвестор у принципі може бути і незацікавлений у фінансуванні та комерціалізації розробленої нами автоматизованої системи управління освітленням в приміщеннях, але остаточне рішення щодо цього питання буде прийматися при врахуванні інших обставин (наприклад, шляхом зниження рівня прийнятого ризику з  $f = 30\%$  до меншої величини, або шляхом підняття ціни реалізації нашої розробки тощо).

Зроблені розрахунки у вигляді графіків наведено на рис. 6.1.

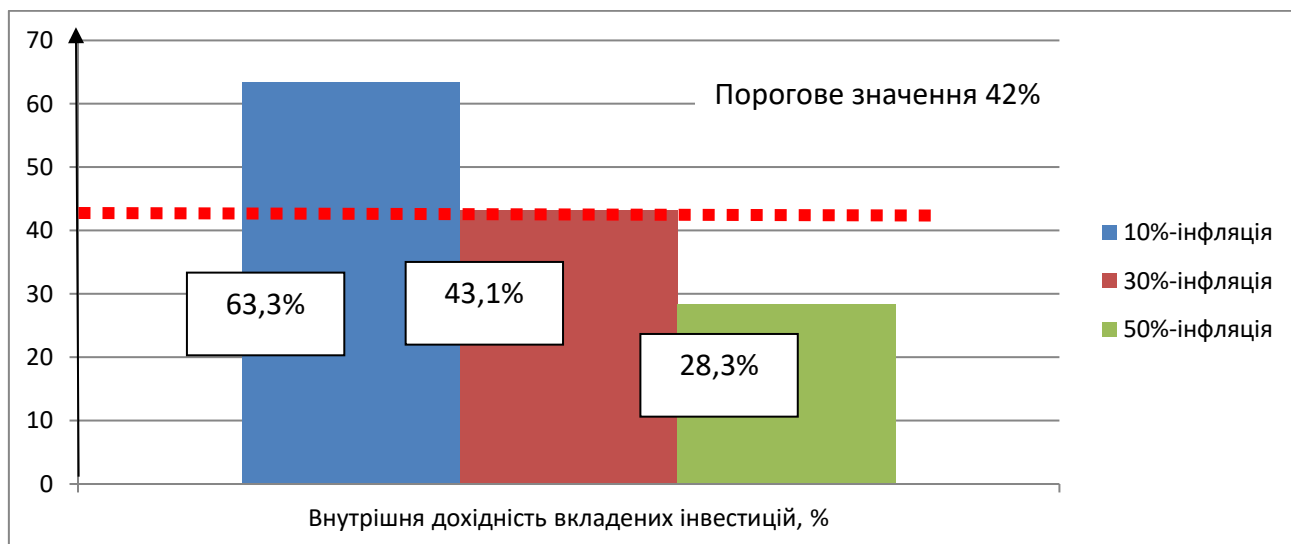


Рисунок 6.1 – Моделювання залежності величини внутрішньої дохідності потенційних інвестицій від рівня інфляції в країні при її значеннях у 10%, 30% та 50%

Аналіз графіка на рис 6.1 показує, що при рівні інфляції в 10% величина внутрішньої дохідності інвестицій становить  $E_B = 63,3\%$ , що значно більше

порогового значення  $\tau_{\text{мін}} = 42\%$  і тому комерціалізація нашої розробки може бути доцільною. При рівні інфляції в 30% величина внутрішньої дохідності інвестицій, вкладених в комерціалізацію нашої розробки, становить  $E_v = 43,1\%$ , що також більше порогового значення  $\tau_{\text{мін}} = 42\%$ , і тому комерціалізація нашої розробки може бути для інвестора доцільною. І тільки при інфляції у 50% величина внутрішньої дохідності інвестицій, вкладених в комерціалізацію нашої розробки, становить  $E_v = 28,3\%$ , що менше порогового значення  $\tau_{\text{мін}} = 42\%$ , і комерціалізація розробленої нами автоматизованої системи управління освітленням в приміщеннях може бути проблематичною і потребує додаткових розрахунків.

Результати виконаної економічної частини магістерської кваліфікаційної роботи зведено у таблицю:

Показники	Задані у ТЗ	Досягнуті у магістерській кваліфікаційній роботі	Висновок
1. Витрати на розробку	Не більше 100 тис. грн	88 тис. грн.	Досягнуто
2. Абсолютний ефект від впровадження розробки, тисяч грн	Не менше 2500 тисяч грн (за три роки)	2686 тисяч грн (при 10%-інфляції)	Виконано
3. Внутрішня дохідність інвестицій, %	не менше 42%	63,3% (при 10%-інфляції); 43,1% (при 30%-інфляції);	Досягнуто
4. Термін окупності інвестицій, роки	до 3-ти років	1,58 років	Виконано

Таким чином, основні техніко-економічні показники розробленої нами автоматизованої системи управління освітленням в приміщеннях, визначені у технічному завданні, виконані.

## ВИСНОВКИ

Виконуючи даний проект, було розглянуто вимоги до систем управління освітленням, а також проаналізовано готові варіанти таких пристроїв, що приставлені на ринку. На даний час є багато однотипних рішень, які базуються на технологіях компаній виробників та потребують регулярне обслуговування. Визначено основні норми освітлення в приміщеннях, які складають 150 ЛК для кухні, 450 ЛК для вітальні, 300 ЛК для спальні, але не більше 200 ЛК для дитячої кімнати.

Відштовхуючись від отриманих даних було розроблено власну концепцію автоматизованої системи управління освітленням в приміщеннях. Основою проекту є апаратна частина, що базується на мікроконтролері Arduino, що повинен мати не менше 64Кб флеш пам'яті, 14 цифрових входів та 7-12В вхідної напруги, функціонал системи забезпечать датчики та модулі.

Програмне забезпечення було розроблене в спеціалізованому середовищі розробки Arduino IDE. Воно максимально підходить для кодування обраного контролера, оскільки пропонує широкий набір інструментів для роботи з Arduino, а також дозволяє без проблем завантажити готову прошивку через USB роз'єм на приєднану плату. Створене ПО відповідає за логіку роботи і дозволяє системі працювати в автономному режимі.

Щоб користувач міг самостійно взаємодіяти з розробкою та проводити необхідні йому налаштування, було розроблено мобільний додаток. Він дозволить на відстані впливати на роботу освітлення, та робить використання системи максимально зручним. Також функціонал додатку дозволить відстежувати рівень освітлення в підконтрольних зонах будинку та двору.

Отже розроблений концепт автоматизованої системи управління освітленням, пропонує досить широкий спектр можливостей, що забезпечать комфорт та зекономлять час користувача. А також він може слугувати основою для “розумного будинку” в подальшому.



Відштовхуючись від отриманих економічних показників можна зробити висновки що витрати на розробку не перевищать 100 тис. грн. Абсолютний ефект від впровадження розробки передбачається в розмірі 2500 тисяч грн (за три роки). Очікується внутрішня дохідність інвестицій не менше 42 відсотка. Таким чином, основні техніко-економічні показники розробленої нами автоматизованої системи управління освітленням в приміщеннях, виконані.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Гріщенко П.С., Овчинников К.В. Ефективність систем контролю освітлення в приміщеннях: науково-технічна конференція факультету інтелектуальних інформаційних технологій та автоматизації 2023рік. – Вінниця: ВНТУ.
2. Розумне освітлення. URL: <https://milight.com.ua/ua/umnoe-osveshchenie/> (дата звернення : 19.05.2023).
3. Що таке Zigbee і чому це важливо для вашого розумного будинку. URL: <https://xterm.com.ua/novosti/что-takoe-zigbee-i-pochemu-eto-vazhno-dlia-vashego-umnogo-doma> (дата звернення : 19.05.2023).
4. Федько А.О., Зеленцова І.Я., Тягунова М.Ю. Способи підвищення надійності системи "розумний будинок" на платформі Arduino Nano : міжнар. наук.- практ. конф.: «Technical sciences: history, the present time, the future, EU experience», 27-28 september, 2019: тези доп., - Wlowlawek, Republic of Poland, 2019. – Р. 49-52
5. Датчики руху: види та призначення. URL: <https://oib.com.ua/datchiki-dvizheniya-vidy-i-prednaznachenie/> (дата звернення : 19.05.2023.)
6. Хмарні сховища. Nic.ua. URL: <https://info.nic.ua/uk/blog-uk/cloudstorage-2/> (дата звернення : 19.05.2023).
7. Vasic M. Mastering Android Development with Kotlin: Deep dive into the world of Android to create robust applications with Kotlin. Packt Publishing, 2017, 378 с
8. Особливості Фрагментації Android. URL: <http://naukam.triada.in.ua/index.php/konferentsiji/46-shistnadtsyata-vseukrajinska-praktichno-piznavalna-internet-konferentsiya/346-osoblivosti-fragmentatsiji-android> (дата звернення : 19.05.2023.)
9. Kotlin Language. URL: <https://kotlinlang.org/docs/home.html>. (дата звернення : 19.05.2023.)
10. Гаджет-інфо. URL: <https://uk.gadget-info.com/> (дата звернення : 19.05.2023).

11. Безпека передачі даних для інтернету речей. URL: <https://www.csecurity.kubg.edu.ua/index.php/journal/article/view/107>
12. Android Fragmentation Vizualized. URL: [https://cdn.opensignal.com/public/data/reports/global/data-2015-08/2015\\_08\\_fragmentation\\_report.pdf](https://cdn.opensignal.com/public/data/reports/global/data-2015-08/2015_08_fragmentation_report.pdf)  
(дата звернення : 20.05.2023).
13. Природнє і штучне освітлення : ДБН В.2.5-28:2006. – К. : Міністерство будівництва, архітектури та житлово-комунального господарства України, 2006. – 68 с. – (Національні стандарти України).
14. Розробка програмного забезпечення: система «Вікіпедія». URL: [https://uk.wikipedia.org/wiki/Розробка\\_програмного\\_забезпечення](https://uk.wikipedia.org/wiki/Розробка_програмного_забезпечення). (дата звернення 19.05.2023).
15. Darwin F. Android Cookbook: Problems and Solutions for Android Developers – 2-е видавництво. — O'Reilly Media, 2017, 772с.
16. HomeKit smart home device | FIBARO. URL: <https://www.fibaro.com/en/homekit/> (дата доступу: 22.05.2023)
17. Brandt, A.; Hui, J.; Kelsey, R.; Levis, P.; Pister, K.; Struik, R.; Vasseur, J.P.; Alexander, R. RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks; RFC 6550; Winter, T., Thubert, P., Eds.; Internet Engineering Task Force: Fremont, CA, USA, 2012.
18. . Кривонос О. М., Кузьменко Є. В., Кузьменко С. В. Огляд та перспективи використання платформи Arduino у вищій школі. Інформаційні технології і засоби навчання. - 2016. - Т. 56, вип. 6.- С. 77-87. URL: [http://nbuv.gov.ua/UJRN/ITZN\\_2016\\_56\\_6\\_9](http://nbuv.gov.ua/UJRN/ITZN_2016_56_6_9) (дата доступу: 22.05.2023)
19. Як налаштувати панель навігації на Android. URL: <https://topinfoweb.com/10158823-how-to-customize-the-navigation-bar-on-android-how-to-remove-or-return-the-buttons-at-the-bottom-of-the-screen> (дата звернення 22.05.2023).

20. Економія електроенергії на підприємстві: основні способи. URL: <https://eenergy.com.ua/energoefectyvnist/ekonomiya-elektroenergiyi-na-pidpryyemstvi/> (дата звернення 14.05.2023).
21. Розумний будинок — з чого він складається та чи потрібен вам. URL: <https://nachasi.com/tech/2018/06/25/smart-house-faq/> (дата звернення : 19.05.2023).
22. Darwin F. Android Cookbook: Problems and Solutions for Android Developers – 2-е видавництво. O'Reilly Media, 2017, 772с.
23. Modern Android development. URL: <https://developer.android.com/jetpack/androidx/releases/room>. (дата звернення 24.05.2022).
24. ДБН В.2.5-28:2018 .Природне і штучне освітлення. Технічні норми.– Чинні з 28.02.2019. – К.: 2018.– 133с.
25. The International Dark-Sky Association (IDA) is the recognized authority on light pollution and is the leading organization combating light pollution worldwide. URL: <https://www.darksky.org/about/> (дата звернення 1.08.2023).
26. Освітлення виробничих приміщень — види промислових світильників. URL: <https://stolb.com.ua/osvitlennya-virobnichikh-primishchen-vidi-promislovikh-svitilnikiv/>( дата звернення 16.08.2023).
27. F. Wu, C. Rüdiger, M.R. Yuce, “Real-Time Performance of a Self-Powered Environmental IoT Sensor Network System,” Sensors, 17:282, pp. 184-253, 2017
28. Cem Kaner, Jack Falk, Hung Q. Nguyen. Testing Computer Software, 2nd Edition. Wiley, 2014, 480с.
29. James A. Whittaker. Exploratory Software Testing: Tips, Tricks, Tours, and Techniques to Guide Test Design. Addison-Wesley, 2018, 247с.
30. В.О. Козловський, О.Й. Лесько, В.В.Кавецький. Методичні вказівки до виконання економічної частини магістерських кваліфікаційних робіт. Вінниця : ВНТУ, 2021. – 42 с.

## **ДОДАТКИ**

**Додаток А (обов'язковий)****Технічне завдання**

Міністерство освіти і науки України  
Вінницький національний технічний університет  
Факультет інтелектуальних інформаційних технологій та автоматизації

ЗАТВЕРДЖУЮ

Завідувач кафедри АІТ

\_\_\_\_\_ д.т.н., проф. Олег БІСІКАЛО

«\_\_» \_\_\_\_\_ 2023 року

**ТЕХНІЧНЕ ЗАВДАННЯ**

на магістерську кваліфікаційну роботу

**АВТОМАТИЗОВАНА СИСТЕМА УПРАВЛІННЯ ОСВІТЛЕННЯМ В  
ПРИМІЩЕННЯХ**

08-31.МКР.004.02.000 ТЗ

Керівник: к.т.н., доц. каф. АІТ

\_\_\_\_\_ Константин ОВЧИННИКОВ

«\_\_» \_\_\_\_\_ 2023 р.

Розробив студент гр. 1АКІТ-22м

\_\_\_\_\_ Павло ГРІЩЕНКО

«\_\_» \_\_\_\_\_ 2023 р.

Вінниця 2023

### 1. Назва та галузь застосування

Данна магістерська кваліфікаційна робота присвячена розробці автоматизованої системи управління освітленням в приміщеннях. Результати роботи, можна застосувати для створення та модернізації автоматизованих систем управління освітленням в приміщеннях. Такі системи забезпечать зручне управління та економну роботу освітлювального обладнання.

### 2. Підстава для проведення розробки

Розробка системи здійснювати на підставі наказу по університету №\_\_ по ВНТУ від «\_\_» \_\_\_\_\_ 2023р. та завдання до магістерської кваліфікаційної роботи, складено та затвердженого кафедрою “Автоматизації та інтелектуальних інформаційних технологій”.

### 3. Мета та призначення розробки

Метою магістерської кваліфікаційної роботи є підвищення економії енергоресурсів підчас використання освітлювального обладнання. Для цього було розроблено доступну систему автоматизованого управління освітленням на базі Arduino, яка матиме зручне налаштування та управління за допомогою мобільного додатку. А також забезпечить високий потенціал для модифікації та розширення функціоналу.

### 4. Джерела розробки

- 1) Методичні вказівки до виконання магістерської дипломних робіт (проектів) для студентів спеціальностей 126 – «Інформаційні системи та технології», 151 – «Автоматизація та комп'ютерно-інтегровані технології» / Уклад. Р. Н. Кветний, О. М. Бевз, О. В. Бісікало. – Вінниця : ВНТУ, 2019. – 26 с.
- 2) ДСТУ Б В.2.2-6-97 (ГОСТ 249940-96) Методи вимірювання освітленості.
- 3) ДСТУ EN 12464-1:2016 Світло та освітлення. Освітлення робочих місць. Частина 1. Внутрішні робочі місця (EN 12464-1:2011, IDT).
- 4) ДСТУ Б EN 15232:2011 Енергоефективність будівель. Вплив автоматизації, моніторингу та управління будівлями (EN 15232:2007, IDT).

## 5. Показники призначення

Основні технічні вимоги та мінімальні системні вимоги до програми:

- наявність 64Кб флеш пам'яті;
- стабільний зв'язок з мережею Інтернет;
- 1Кб EEPROM пам'яті;
- коректно встановлена прошивка на плату Android;
- процесор як мінімум з 1 ядром Kryo 680 Prime (Cortex-X1) на 2840 МГц;
- наявність запасу пам'яті на девайсі для можливого оновлення додатку;
- 15Гб вільного місця в хмарному сховищі .

Вхідні дані для проведення роботи:

- дані з датчиків що встановленні в підконтрольних приміщеннях;
- команди що віддаються системі через додаток;
- вхідна напруга 7-12В;
- наявність 14 цифрових входів;

Результати роботи програми:

- вмикання та вимикання освітлювального обладнання;
- зміна рівня яскравості освітлення в підконтрольному приміщенні;
- моніторинг рівня освітлення в приміщенні;
- реагування на присутність користувача в кімнаті;
- виведення статистики рівня освітлення в приміщенні через додаток;

## 6. Економічні показники

До економічних показників входять:

- Витрати на розробку – не більше 100 тис. Грн;
- Абсолютний ефект від впровадження розробки – не менше 2500 тисяч грн (за три роки)
- Внутрішня дохідність інвестицій – не менше 42%
- Термін окупності інвестицій – близько 3-х років

## 7. Стадії розробки

а) «Аналіз систем контролю освітлення в приміщеннях» має бути виконаний до 6.10.2023 – 12.10.2023.



- б) « Обґрунтування вибору апаратної частини системи контролю освітлення в приміщеннях» має бути виконаний до 12.10.2023 – 19.10.2023.
- в) « Розробка програмного забезпечення центрального контролера» має бути виконаний до 19.10.2023 – 03.11.2023.
- г) «Розробка мобільного додатку для управління системою» має бути виконаний до 03.11.2023 – 15.11.2023.
- д) «Тестування розробленого програмного забезпечення» має бути виконаний до 15.11.2023 – 20.11.2023.
- е) Економічний розділ має бути виконаний до 20.11.2023 – 26.11.2023.
- є) Оформлення матеріалів до захисту МКР 26.11.2023 – 12.12.2023.

#### 8. Порядок контролю та приймання

- 1) Рубіжний контроль. Провести до 11.11.2023.
- 2) Попередній захист магістерської кваліфікаційної роботи. Провести до 21.11.2023.
- 3) Захист магістерської кваліфікаційної роботи. Провести в період з 12.12.2023 до 14.12.2023

**Додаток Б (обов'язковий)****ІЛЮСТРАТИВНА ЧАСТИНА**

АВТОМАТИЗОВАНА СИСТЕМА УПРАВЛІННЯ ОСВІТЛЕННЯМ В  
ПРИМІЩЕННЯХ

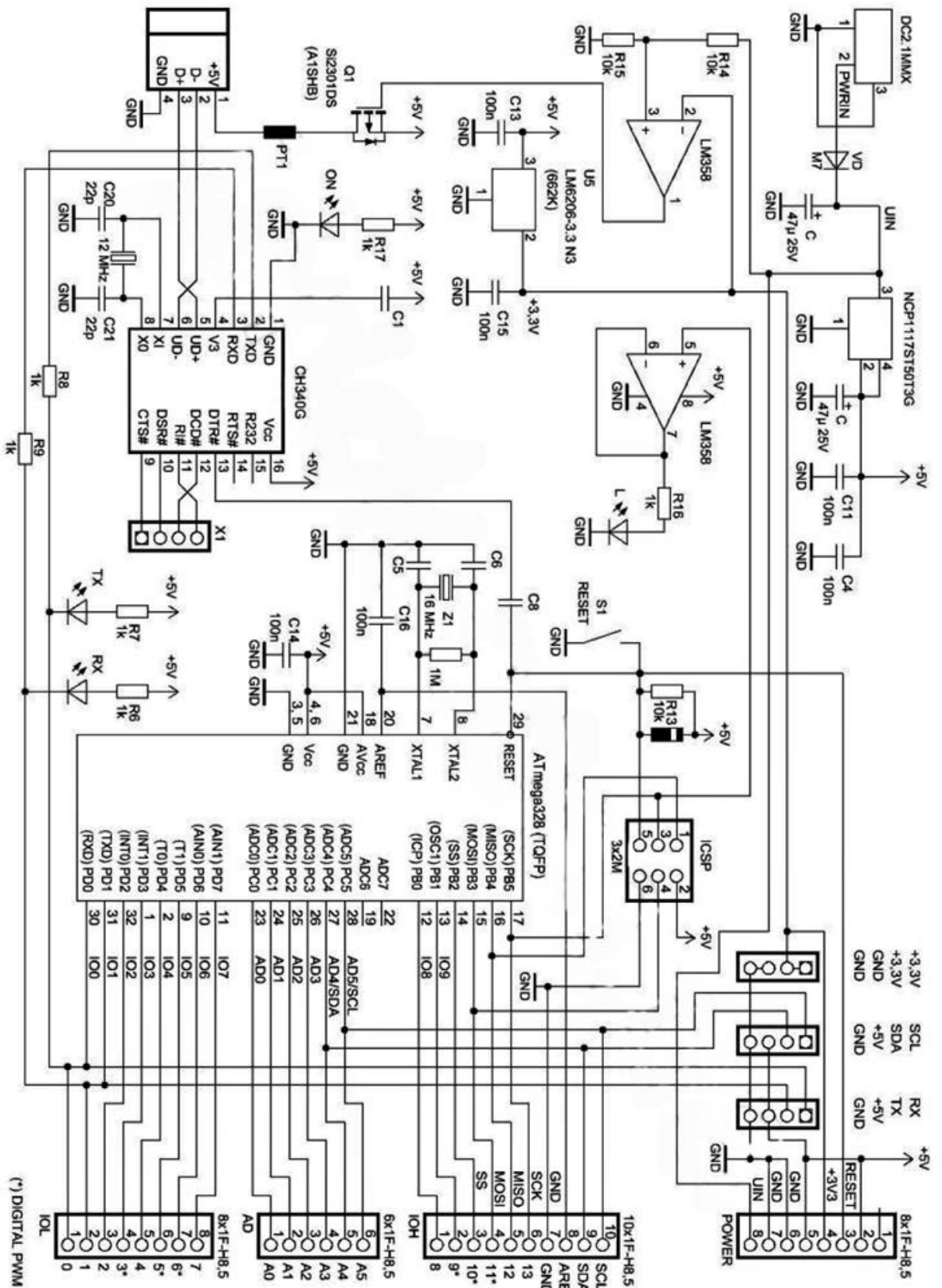


Рисунок Б.1 – Схема платы Arduino UNO

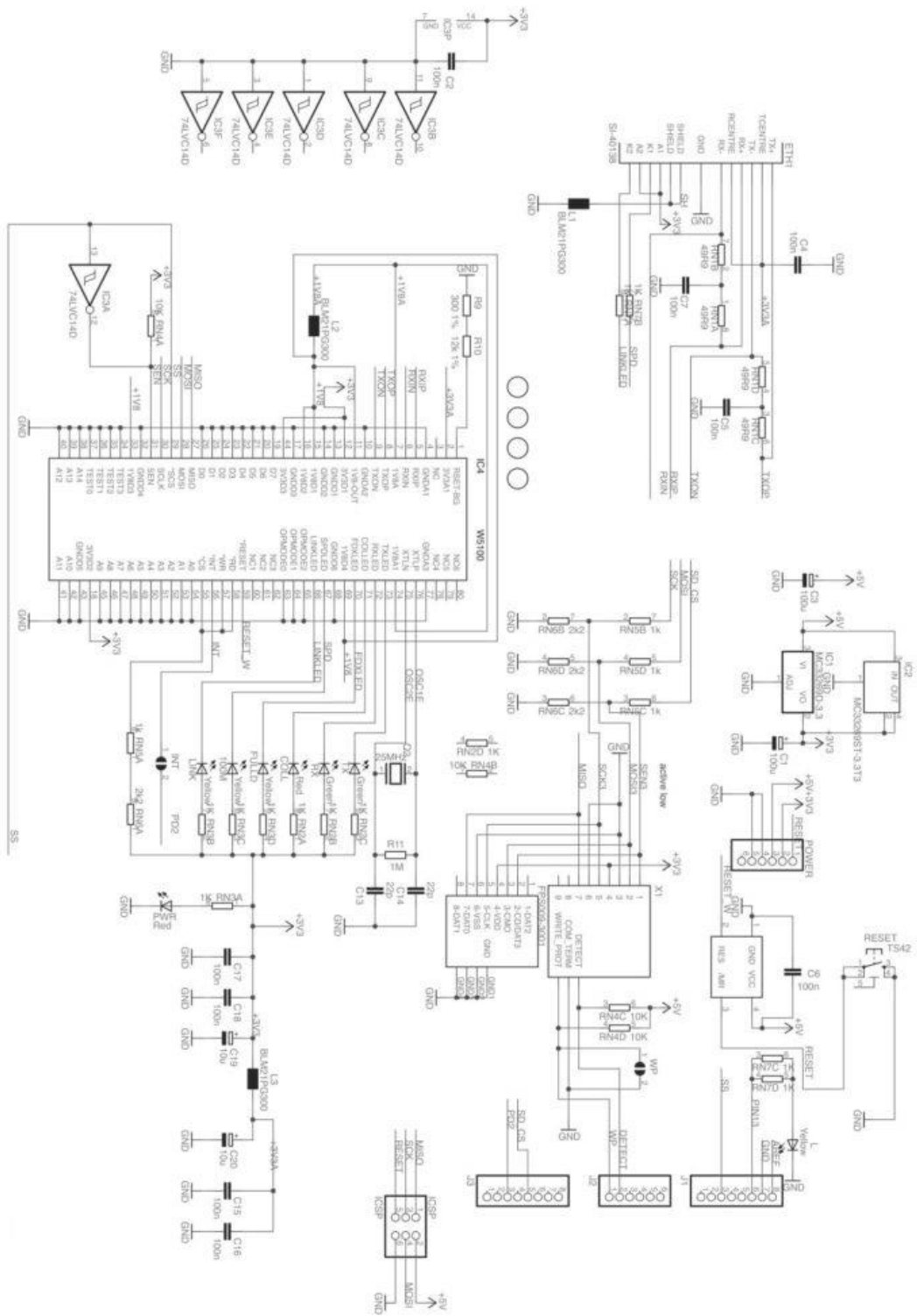


Рисунок Б.2 – Принципова схема Ethernet Shield W5100

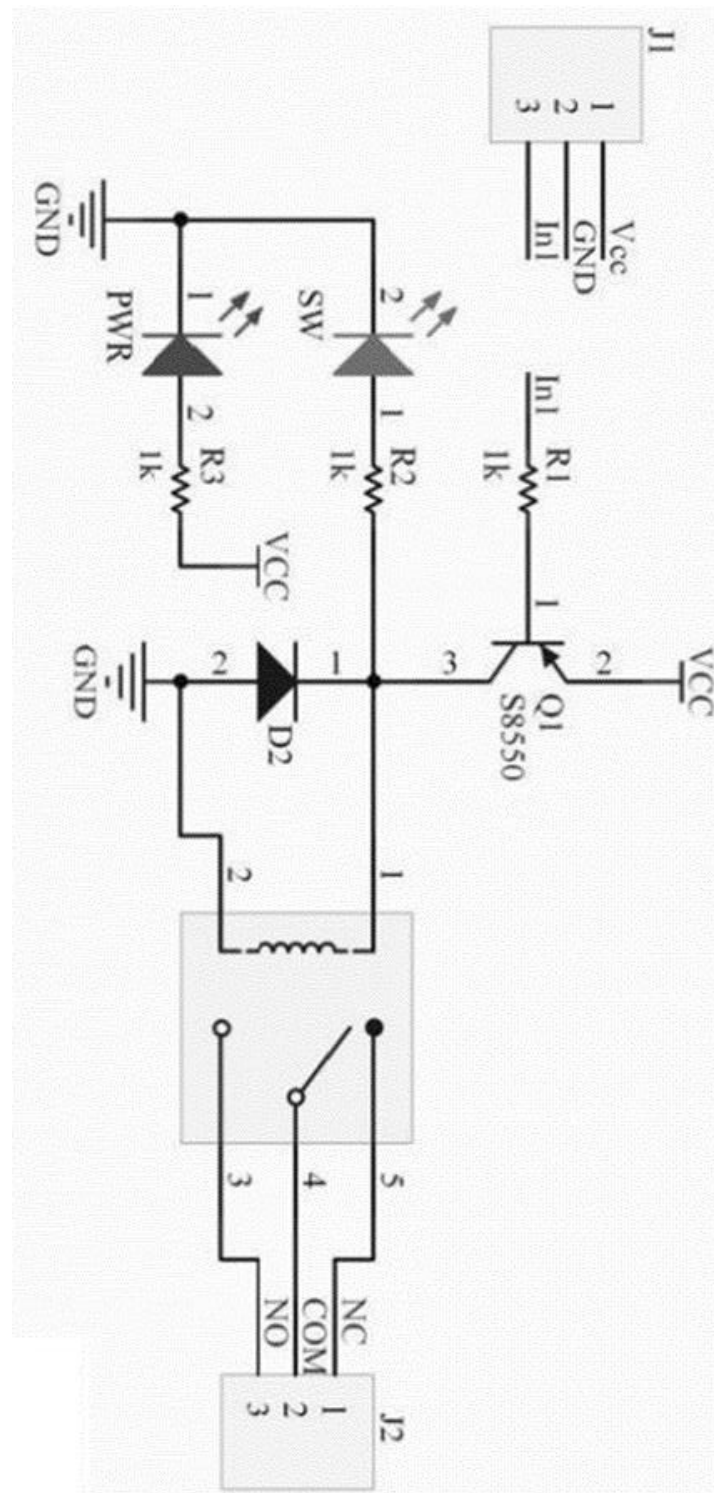


Рисунок Б.3 – Принципова електрична схема одноканального модулю реле

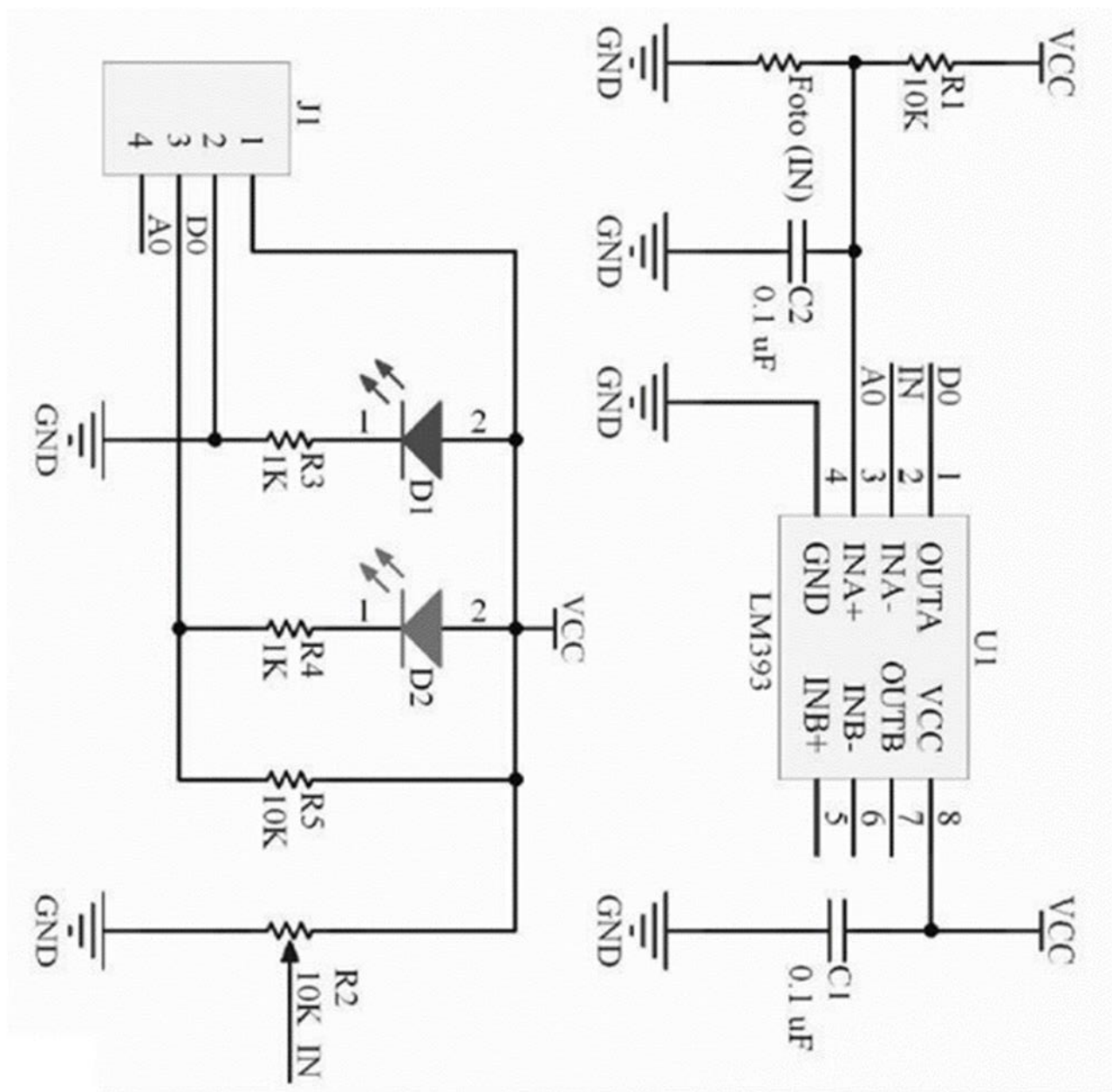


Рисунок Б.4 – Принципова схема модуля освітленості LM393

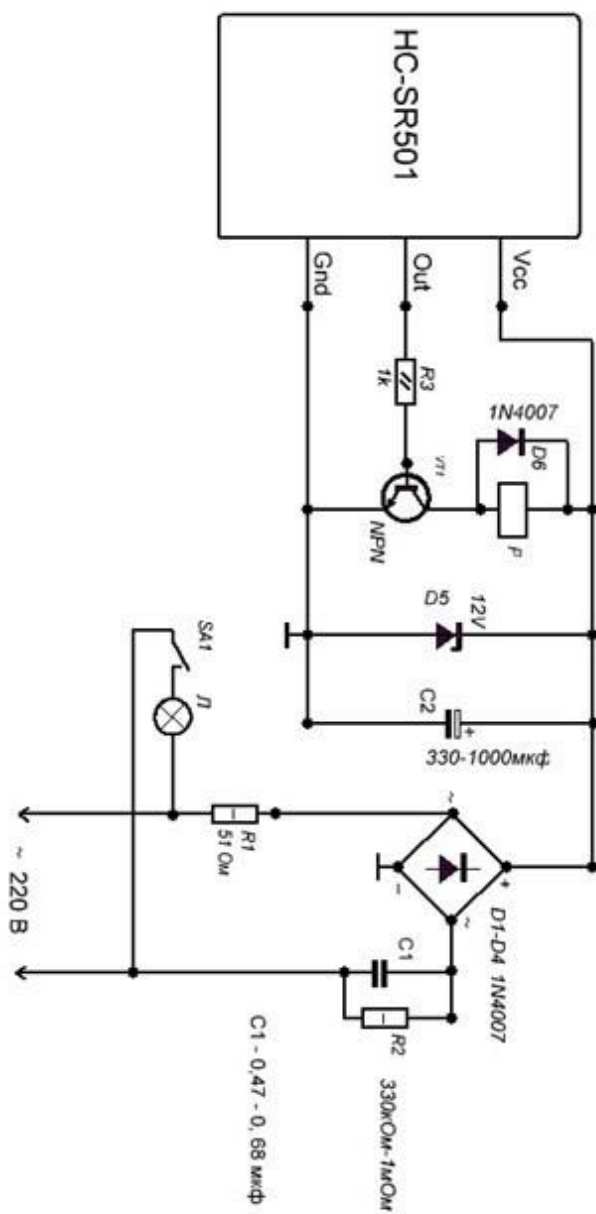


Рисунок Б.5 – Принципова схема датчику HC-SR501

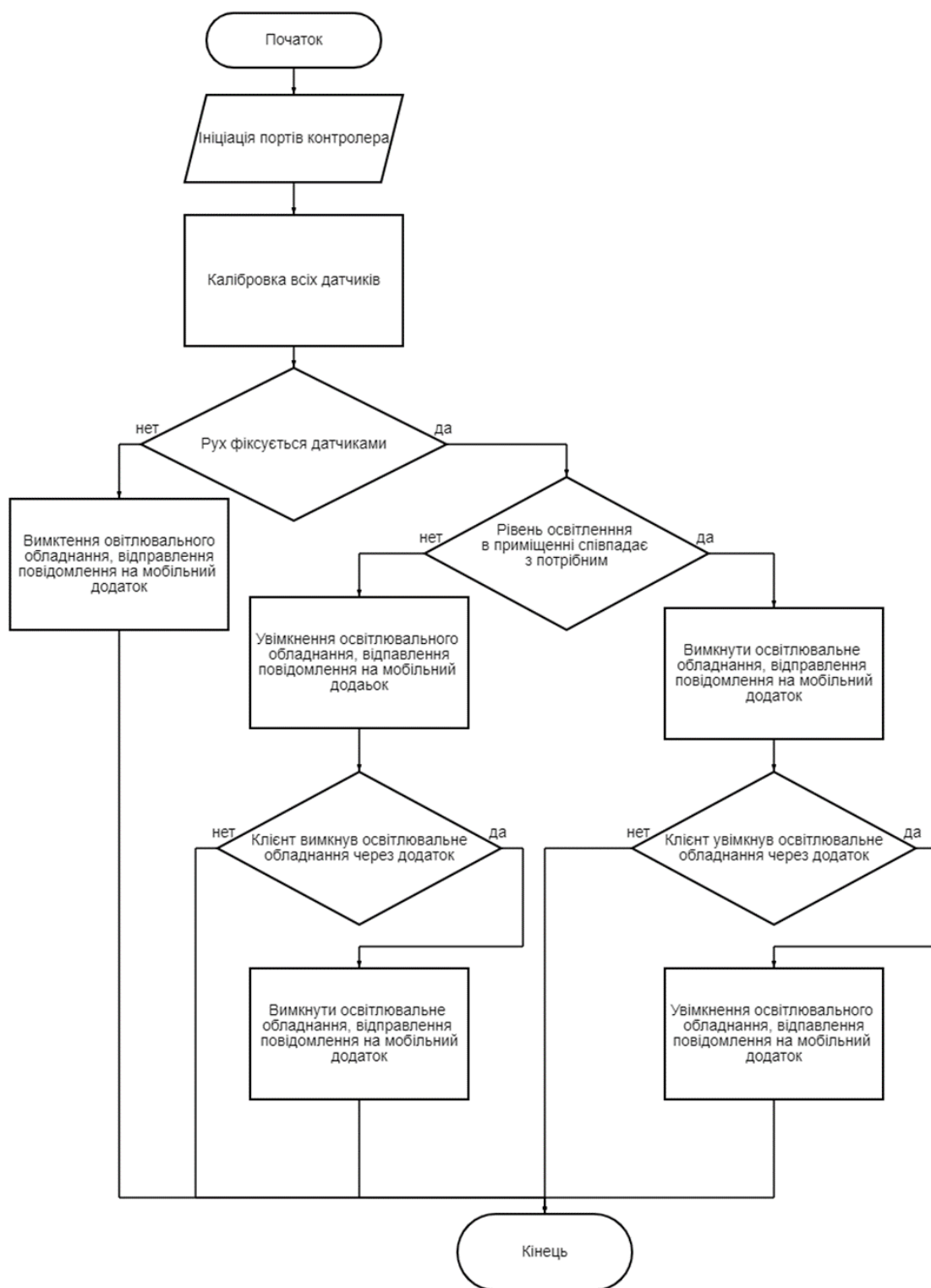


Рисунок Б.6 – Блок схема, що описує роботу системи



## Додаток В (обов'язковий).

## Лістинг програми завантаження файлів

```

/**
 * The method for downloading files from Dropbox
 * @param fileNamesInDropbox The name of files, which you want to get
 * @param files The initialized files, where you want to write files from dropbox
 * @return ArrayList with files from Dropbox
 * @throws ExecutionException in case of wrong file name or some other clauses, which may result
 * in not finding file with your filename in Dropbox folder
 * @throws InterruptedException in case of interrupting this thread by another one
 */
public ArrayList<File> download(String[] fileNamesInDropbox, File... files) throws ExecutionException, InterruptedException {
    DropboxConnector.DownloadFileFromDropbox dwl = new DropboxConnector.DownloadFileFromDropbox(
        DropboxConnector.FILE_DIR, mDBApi, fileNamesInDropbox);
    return dwl.execute(files).get();
}

//Async classes

/**
 * The Async class for downloading file from Dropbox in new Thread. Have to be invoked in UI
 * thread
 */
private class DownloadFileFromDropbox extends AsyncTask<File, Void, ArrayList<File>> {
    /**
     * The path to app folder in Dropbox
     */
    private String path;
    /**
     * A Dropbox API class
     */
    private DropboxAPI<?> dropbox;
    /**
     * The name of files, which you want to get
     */
    private String[] fileNamesInDropbox;

    /**
     * The constructor of async download class
     * @param path The path to app folder in Dropbox
     * @param dropbox A Dropbox API class
     * @param fileNamesInDropbox The name of files, which you want to get
     */
    public DownloadFileFromDropbox(String path, DropboxAPI<?> dropbox, String[] fileNamesInDropbox) {
        this.path = path;
        this.dropbox = dropbox;
        this.fileNamesInDropbox = fileNamesInDropbox;
    }

    @Override
    protected ArrayList<File> doInBackground(File... params) {
        ArrayList<File> files = new ArrayList<>(params.length);

        int count = 0;
        for(File file: params){
            FileOutputStream fileOutputStream;
            try {
                fileOutputStream = new FileOutputStream(file);
                dropbox.getFile(path + fileNamesInDropbox[count], null, fileOutputStream, null);
                files.add(file);
                count++;
            } catch (DropboxException | IOException e) {
                e.printStackTrace();
            }
        }
        return files;
    }

    @Override
    protected void onPostExecute(ArrayList<File> resFiles) {
    }
}
}

```

## Додаток Г (обов'язковий).

### ПРОТОКОЛ ПЕРЕВІРКИ ДИПЛОМНОЇ РОБОТИ НА НАЯВНІСТЬ ТЕКСТОВИХ ЗАПОЗИЧЕНЬ

Назва роботи: Автоматизована система управління освітленням в приміщеннях

Тип роботи: магістерська кваліфікаційна робота

Підрозділ: кафедра автоматизації та інтелектуальних інформаційних технологій, факультет інтелектуальних інформаційних технологій та автоматизації

#### Показники звіту подібності Plagiat.pl (StrikePlagiarism)

Оригінальність 82.9 % Схожість 17.1 %

Аналіз звіту подібності (відмітити потрібне)

- Запозичення, виявлені у роботі, оформлені коректно і не містять ознак плагіату
- Виявлені у роботі запозичення не мають ознак плагіату, але їх надмірна кількість викликає сумніви щодо цінності роботи і відсутності самостійності її автора. Роботу направити на розгляд експертної комісії кафедри.
- Виявлені у роботі запозичення є недобросовісними і мають ознаки плагіату та/або в ній містяться навмисні спотворення тексту, що вказують на спроби приховування недобросовісних запозичень

Особа, відповідальна за перевірку \_\_\_\_\_ Роман МАСЛІЙ  
(підпис)

Ознайомлені з повним звітом подібності, який був згенерований системою Unicheck щодо роботи

Автор роботи \_\_\_\_\_ Павло ГРІЩЕНКО  
(підпис)

Керівник роботи \_\_\_\_\_ Костянтин ОВЧИННИКОВ  
(підпис)