

Міністерство освіти і науки України
Криворізький національний університет
Факультет інформаційних технологій
Кафедра комп'ютерних систем та мереж

Пояснювальна записка
до кваліфікаційної роботи бакалавра
за спеціальністю 123 «Комп'ютерна інженерія»

на тему: РОЗУМНА СИСТЕМА КОНТРОЛЮ ДОГЛЯДУ ЗА РОСЛИНАМИ

Проектував	_____	Д. Д. Духов
Керівник роботи	_____	Д. І. Кузнецов
Нормоконтроль	_____	Д. І. Кузнецов
Завідувач кафедри	_____	А. І. Купін

Криворізький національний університет
Факультет інформаційних технологій
Кафедра комп'ютерних систем та мереж

Ступінь вищої освіти
Спеціальність

бакалавр
123 «Комп'ютерна інженерія»

ЗАТВЕРДЖУЮ

Завідувач кафедри, голова циклової комісії

_____ А. І. Купін

“ ____ ” _____ 20__ року

З А В Д А Н Н Я
НА КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ

_____ (прізвище, ім'я, по батькові)

1. Тема роботи _____

керівник роботи _____,

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

затверджені наказом вищого навчального закладу від “ ____ ” _____ 20__ року №__

2. Строк подання студентом роботи _____

3. Вихідні дані до роботи _____

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

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

6. Консультанти розділів роботи

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

7. Дата видачі завдання _____

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Строк виконання етапів роботи	Примітка

Студент _____
(підпис) (прізвище та ініціали)Керівник роботи _____
(підпис) (прізвище та ініціали)

РЕФЕРАТ

Пояснювальна записка: 60 сторінок, 34 рисунків, 1 таблиця, 1 додаток, 32 використаних джерел.

Об'єкт аналізу – розумна система догляду за кімнатними рослинами.

Проект складається з трьох розділів.

Перший розділ присвячений аналізу систем догляду за рослинами. Надано детальний огляд ринку на наявність готових рішень, опис доступних технологій та мобільних додатків, що пов'язані з доглядом рослин в домашніх умовах та їх аналіз.

У другому розділі розкриті питання вибору апаратної частини для моделі виконавчого модулю системи розумного догляду за рослинами. Реалізація елементу керування розумної системи через мобільний додаток, який з'єднується з виконавчою частиною через Bluetooth.

Третій розділ є заключним. Опис правил використання спроектованої моделі догляду за рослинами вдома. Проводиться тестування готової системи.

ІНТЕРНЕТ РЕЧЕЙ, ПРОГРАМУВАННЯ МІКРОКОНТРОЛЕРА, ПРОГРАМУВАННЯ МОБІЛЬНОГО ДОДАТКА, BLUETOOTH, ARDUINO, REACT, EXPO, JAVASCRIPT, РЕАЛІЗАЦІЯ СИСТЕМИ КОТОРОЛЮ.

					КНУ.РБ.123.24.06.Р			
Змн.	Арк.	№ документа	Підпис	Дата				
Розробив		Духов			РЕФЕРАТ	Літера	Аркуш	Аркушів
Перевірив		Кузнецов						
Н.контроль		Кузнецов				КІ-20		
Затвердив		Купін						

Bachelor's qualifying paper: 60 pages, 34 figures, 1 table, 1 addition, 32 used sources.

Object of analysis is the implementation of a smart indoor plant care system through the Internet of Things.

The project consists of three sections.

The first chapter is devoted to the description of the problem of plant care. A detailed review of the market is conducted for the availability of ready-made solutions, a description of available technologies and mobile applications related to plant care at home and their analysis.

In the second section, the issues of choosing the hardware parts of the model for the implementation of the executive module of the smart plant care system, as well as the implementation of the control element of the smart system through a mobile application that is disconnected from the executive part via Bluetooth, are revealed.

The third section is the final one, where the finished system is analyzed and the rules for using the designed model of plant care at home are described.

INTERNET OF THINGS, MICROCONTROLLER PROGRAMMING, MOBILE APPLICATION PROGRAMMING, BLUETOOTH, ARDUINO, REACT, EXPO, JAVASCRIPT, CONTROL SYSTEM IMPLEMENTATION.

					КНУ.РБ.123.24.06.Р	Арк.
Арк.	№ документа	Підпис	Дата			

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ.....	7
ВСТУП.....	8
1 АНАЛІЗ ТА ПОРІВНЯННЯ СИСТЕМ ГОТОВИХ РІШЕНЬ НА РИНКУ ІНТЕРНЕТ РЕЧЕЙ	9
1.1 Дослідження проблеми догляду за кімнатними рослинами та пошук його рішення	9
1.2 Дослідження пристроїв догляду за рослинами	9
1.2.1 Пристрій розумний вазон	10
1.2.2 Пристрій гідропонна установка	11
1.2.3 Пристрій краплинний полив для кімнатних рослин Wi Drop M12	12
1.3 Дослідження мобільних додатків догляду за рослинами.....	13
1.3.1 Додаток Plant Care Reminder	13
1.3.2 Додаток Flora Incognita	14
1.3.3 Додаток PlantIn: Визначник Рослин	15
Висновки	16
2 ПРОЄКТУВАННЯ ДИСТАНЦІЙНОЇ СИСТЕМИ ДОГЛЯДУ ЗА РОСЛИНАМИ	17
2.1 Реалізація виконавчого модулю системи	17
2.1.1 Вибір обладнання для реалізації системи.....	18
2.1.2 Алгоритм роботи виконавчої частини системи	27
2.1.3 Електрична схема модулю та поетапна збірка модулю	29
2.2 Програмування керуючого елемента системи	39
2.2.1 Технологія з'єднання Bluetooth та його використання.....	39
2.2.2 Програмування мікроконтролера Arduino для виконавчого модулю.....	41
2.2.3 Програмування мобільного додатку для керування системою в середовищі програмування EXPO	41
2.2.4 UML діаграма програмної частини системи	45
Висновки	46
3 РЕЗУЛЬТАТИ ТЕСТУВАННЯ МОДЕЛІ ТА ПРАВИЛА ЕКСПЛУАТАЦІЇ СИСТЕМИ	47
3.1 Тестування схеми на Arduino	47
3.2 Тестування мобільного додатка.....	51
3.3 Інструкція з використання системи.....	54
Висновки	56
ВИСНОВКИ.....	57
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	58
Додаток А Схема модулю системи контролю догляду за рослинами	61

					КНУ.РБ.123.24.06.3			
Змн.	Арк.	№ документа	Підпис	Дата				
Розробив		Духов			ЗМІСТ	Літера	Аркуш	Аркушів
Перевірив		Кузнєцов						
Н.контроль		Кузнєцов			KI-20			
Затвердив		Купін						

ПЕРЕЛІК СКОРОЧЕНЬ

БЖ – блок живлення;

МК – мікроконтролер;

BT – Bluetooth (технологія передачі даних між пристроями);

IoT – Internet of Things (Інтернет речей).

					КНУ.РБ.123.24.06.ПС	Арк.
Арк.	№ документа	Підпис	Дата			

ВСТУП

У сучасних умовах значний акцент робиться на впровадженні автоматизації та інтелектуалізації процесів, що стосуються повсякденного життя, зокрема догляду за рослинами у домашніх умовах. Даний підхід дозволяє оптимізувати ці процеси, покращуючи ефективність догляду за рослинами та забезпечуючи їм комфортні умови для зростання.

У цій роботі представлено розробку власної моделі розумної системи контролю догляду за рослинами, що здійснена на підставі аналізу готових рішень наявних систем догляду за кімнатними рослинами на ринку інтернет речей, а також відомих мобільних додатків з метою створення дистанційного керування. Програмування керуючого елемента системи з використанням МК Arduino. Зручний у використанні мобільний додаток, створений у середовищі Expo. З'єднання між системою та додатком здійснюється за допомогою технології Bluetooth, що є частиною концепції розумного будинку.

Основні компоненти цієї системи включають автоматичний полив рослин за допомогою насосу з напругою 5 В з датчиком вологості та освітлення за допомогою спеціальної запрограмованої фітострічки. Автоматичний полив дозволяє рослинам отримувати достатню кількість води у потрібний час, що запобігає їх пересиханню або надмірному зволоженню. Водночас, освітлення з використанням запрограмованої фітострічки забезпечує оптимальний спектр світла для рослин, що сприяє їх здоровому зростанню та розвитку.

Крім основних функціональних можливостей, система також надає користувачам можливість встановлювати індивідуальні параметри догляду за рослинами, включаючи режим поливу за показниками вологості ґрунту чи задати інтервал між поливом та налаштувати тривалість додаткового освітлення. Це дозволяє користувачам налаштовувати систему під специфічні потреби кожного виду рослин, забезпечуючи їм найкращі умови для зростання.

Розробка цієї системи є важливим кроком до підвищення рівня автоматизації у догляді за рослинами та впровадження концепції розумного будинку. Тема проекту є актуальною, тому що використання таких технологій сприяє покращенню якості життя та знижує навантаження на власників будинків, дозволяючи їм зосередитися на інших важливих аспектах повсякденного життя.

Даний проєкт створено з метою подальшого удосконалення системи контролю догляду за рослинами у сфері інтернет речей та має відкритий код викладений на платформі GitHUB для залучення однодумців та бажаючих реалізувати дану систему у себе.

					КНУ.РБ.123.24.06.ВС			
Змн.	Арк.	№ документа	Підпис	Дата				
Розробив		Духов			ВСТУП	Літера	Аркуш	Аркушів
Перевірив		Кузнєцов						
Н.контроль		Кузнєцов			KI-20			
Затвердив		Купін						

1 АНАЛІЗ ТА ПОРІВНЯННЯ СИСТЕМ ГОТОВИХ РІШЕНЬ НА РИНКУ ІНТЕРНЕТ РЕЧЕЙ

1.1 Дослідження проблеми догляду за кімнатними рослинами та пошук його рішення

Догляд за кімнатними рослинами - це важлива тема для багатьох людей, особливо тих, хто прагне створити приємний та зелений простір у своїй оселі. Кімнатні рослини не тільки додають краси та затишку в будинок, але й сприяють покращенню якості повітря, підвищенню вологості та загальному благополуччю мешканців. Однак, догляд за кімнатними рослинами може бути складним завданням, особливо для тих, хто не має у цьому досвіду. Тому важливо розглянути проблеми, які можуть виникати при догляді за кімнатними рослинами, та шукати шляхи їх вирішення.

Одна з головних проблем з якою стикаються власники кімнатних рослин - це вибір правильного режиму поливу. Вирішення цієї проблеми полягає у встановленні графіку поливу, який підходить кожній конкретній рослині. Це може вимагати експериментування з частотою поливу та кількістю води, а також спостереження за станом рослин. Інша поширена проблема - недостатнє освітлення. Кімнатні рослини потребують різних рівнів світла залежно від їх виду.

Для розв'язання зазначених проблем власникам кімнатних рослин важливо бути обізнаними про потреби своїх рослин та створювати сприятливі умови для їх росту. Регулярне спостереження за станом рослин, коригування умов утримання та використання відповідних інструментів і методів догляду допоможуть забезпечити здоровий розвиток кімнатних рослин.

У сучасних умовах актуальними стали дистанційна робота та дистанційне навчання. Люди все більше часу проводять вдома. У зв'язку з цим виникає попит на вирощування рослин у домашніх умовах. Шалений темп життя спонукає до постійної автоматизації процесів життєдіяльності. Це надихнуло на дослідження ринку інтернет речей стосовно полегшення догляду за рослинами.

1.2 Дослідження пристроїв догляду за рослинами

На ринку представлені наступні пристрої догляду за рослинами:

- розумний вазон;
- гідропонна установка;
- краплинний полив для кімнатних рослин.

					КНУ.РБ.123.24.06.01.АПСТР			
Змн.	Арк.	№ документа	Підпис	Дата				
Розробив		Духов			АНАЛІЗ ТА ПОРІВНЯННЯ СИСТЕМ ГОТОВИХ РІШЕНЬ	Літера	Аркуш	Аркушів
Перевірив		Кузнєцов						
Н.контроль		Кузнєцов			КІ-20			
Затвердив		Купін						

1.2.1 Пристрій розумний вазон



Рисунок 1.1 – Розумний вазон

Принцип дії.

Система автоматичного поливу «Розумний вазон»(рис. 1.1) гарантує, що рослини в залежності від розміру горщика отримають достатню кількість води та поживних речовин, необхідних для їх оптимального зростання протягом терміну до 12 тижнів. Зручний у використанні для догляду за кімнатними рослинами. Принцип дії зображено на малюнку [1].

Переваги:

- Легко використати;
- Об'єм рідини в контейнері: 1 л;
- Є рівень ємності води в вазоні;
- Малий розмір.

Недоліки:

- Немає дистанційного керування;
- Немає графіку поливу;
- Немає контролю вологості;
- Немає контролю поливу.

Ціна товару на 2024 рік становить від 1300 грн. до 3000 грн.

Рейтинг на ринку інтернет речей складає 5 із 5.

1.2.2 Пристрій гідропонна установка



Рисунок 1.2 – Гідропонна установка

Принцип дії.

Smart Garden(рис. 1.2) – це ретельно розроблений, інноваційний розумний сад для будинку. Все, що вимагається - вставити рослинні капсули, що розкладаються, заповнити резервуар водою і включити кабель у розетку. Лампи забезпечують оптимальні спектри за 10000 lux. Завдяки світлодіодному освітленню, розумний вазон Click&Grow споживає мінімум енергії. Лампа легко регулюється по висоті, за допомогою подовжувачів. Вбудований таймер забезпечує ідеальну кількість світла протягом 16 годин щодня і вимикається на 8. Автоматичного поливу вистачає на місяць [2].

Переваги:

- Легко використати;
- Об'єм рідини в контейнері на місяць;
- Є лампа для освітлення;
- Догляд за декількома рослинами.

Недоліки:

- Немає дистанційного керування;
- Немає графіку поливу;
- Немає контролю вологості;
- Немає контролю поливу.

Ціна товару на 2024 рік становить від 5800 грн. до 12700 грн.

Рейтинг на ринку інтернет речей складає 4.8 із 5.

					КНУ.РБ.123.24.06.01. АПСГР	Арк.
Арк.	№ документа	Підпис	Дата			

1.2.3 Пристрій краплинний полив для кімнатних рослин Wi Drop M12



Рисунок 1.3 – Краплинний полив для кімнатних рослин

Принцип дії.

Краплинний полив(рис. 1.3) зручний тим, що можна залишити рослини на час від'їзду, відпустки. Достатньо запрограмувати пристрій на певний інтервал та інтенсивність поливу. Краплинний полив Wi Drop M12 працює від вбудованого акумулятора. У середньому одного заряду вистачає на 1-2 місяці. Немає необхідності підключати автоматичну систему поливу до крана. Шланг просто поміщається у будь-яку ємність із водою залежно від необхідної продуктивності. Крім таймера обирається і тип розпилення: звичайний полив чи крапельне зрошення. У пристрій вбудована система виявлення відсутності води, а при низькому відсотку заряду акумулятора прилад просигналізує світловим мерехтінням [3].

Переваги:

- Компактність;
- Є графік поливу;
- Різні режими роботи;
- Догляд за декількома рослинами.

Недоліки:

- Немає дистанційного керування;
- Немає контролю вологості.

Ціна товару на 2024 рік становить від 1100 грн. до 2100 грн.

Рейтинг на ринку інтернет речей складає 3.2 із 5.

1.3 Дослідження мобільних додатків догляду за рослинами

Розповсюдження системи розумний дім нараховує певну кількість мобільних додатків для спрощення догляду за рослинами. Для аналізу обрано:

- Plant Care Reminder;
- PictureThis - Plant Identifier;
- PlantIn: Визначник Рослин.

1.3.1 Додаток Plant Care Reminder

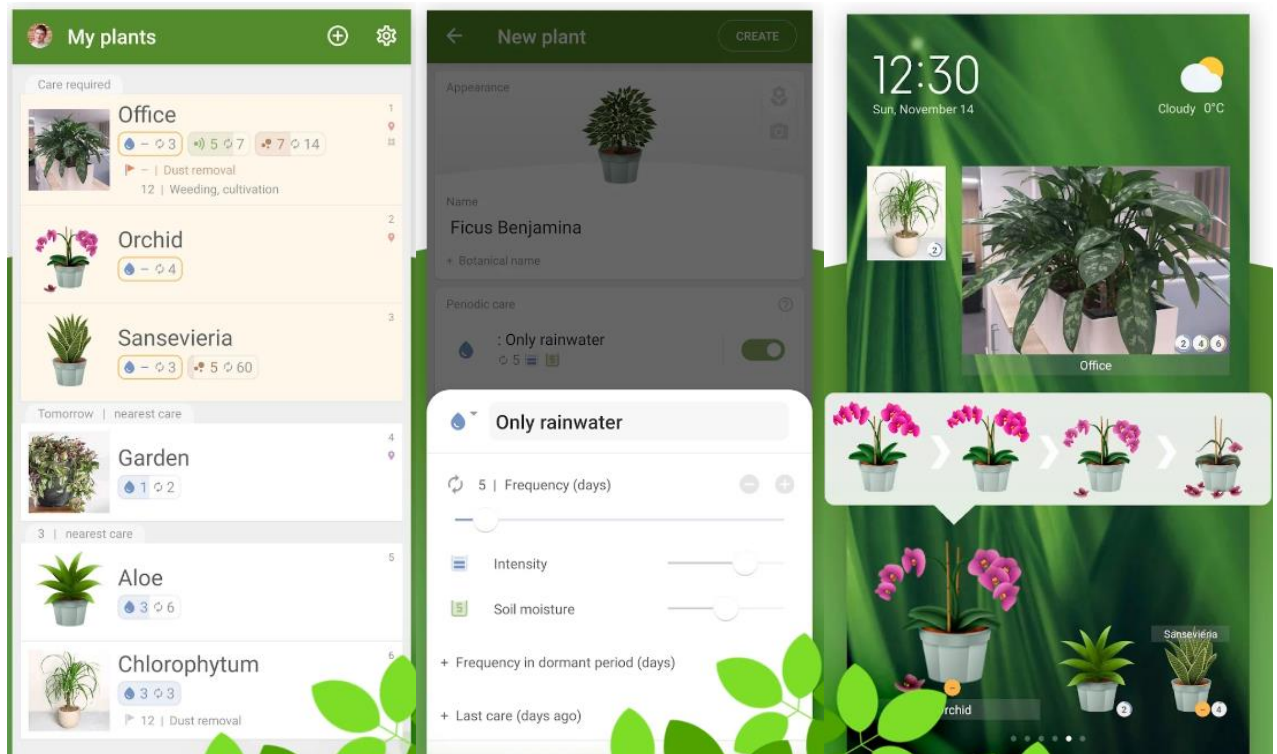


Рисунок 1.4 – Додаток Plant Care Reminder

Програма Plant Care Reminder(рис. 1.4) - інструмент для моніторингу стану рослин та своєчасного їх догляду. Додаток дозволяє створити віджет для кожної рослини окремо або для всіх рослин одночасно у будинку, саду чи на робочому місці.

Отримання інформації про рослини без необхідності відкривати додаток - все необхідне на головному екрані пристрою.

Віджет відображає зображення рослини, яке змінюється відповідно до стану рослини, а також календар з підрахунком днів з останнього поливу. У налаштуваннях є можливість обрати зовнішній вигляд рослини, надати їй назву, додати коментарі та інші опції [4].

На 2024 рік додаток безкоштовний та має рейтинг 4.2 із 5 на більш ніж 100 тисяч кількість завантажень.

					КНУ.РБ.123.24.06.01. АПСР	Арк.
Арк.	№ документа	Підпис	Дата			

1.3.2 Додаток Flora Incognita

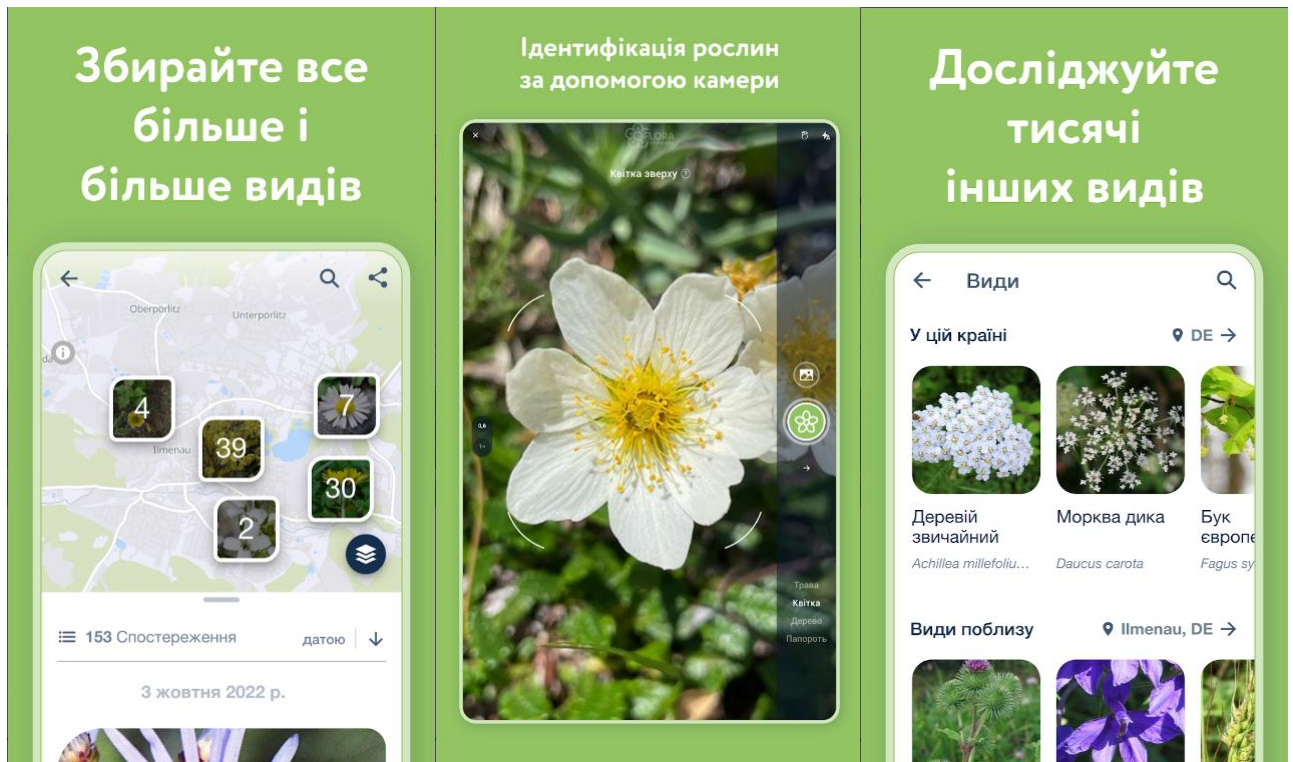


Рисунок 1.5 – Додаток Flora Incognita

Flora Incognita (рис. 1.5) забезпечує швидку відповідь на питання шляхом фотографування рослини. Високоточні алгоритми, засновані на штучному інтелекті, дозволяють ідентифікувати дикі рослини, навіть якщо вони ще не квітнуть.

Flora Incognita це безкоштовний та безрекламний додаток, який є частиною науково-дослідного проекту, спрямованого на покращення охорони природи. Зібрані спостереження використовуються для наукових досліджень, таких як дослідження поширення інвазивних видів або вплив зміни клімату.

Ідентифікація видів рослин у додатку Flora Incognita базується на алгоритмах глибокого навчання, що забезпечують точність понад 90%. Для максимальної точності ідентифікації рекомендується робити чіткі фотографії рослин, таких як квіти, листя, кора чи фрукти.

Додаток розроблено вченими з Технічного університету Ільменау та Інституту біогеохімії Макса Планка в Єні. Його розробка була підтримана Федеральним міністерством освіти і досліджень Німеччини, Німецьким Федеральним агентством з охорони природи за кошти Федерального міністерства довкілля, охорони природи та ядерної безпеки [5].

Зручний і доступний у використанні.

На 2024 рік додаток безкоштовний та має рейтинг 4.9 із 5 на більш ніж 1 мільйон кількість завантажень.

1.3.3 Додаток PlantIn: Визначник Рослин

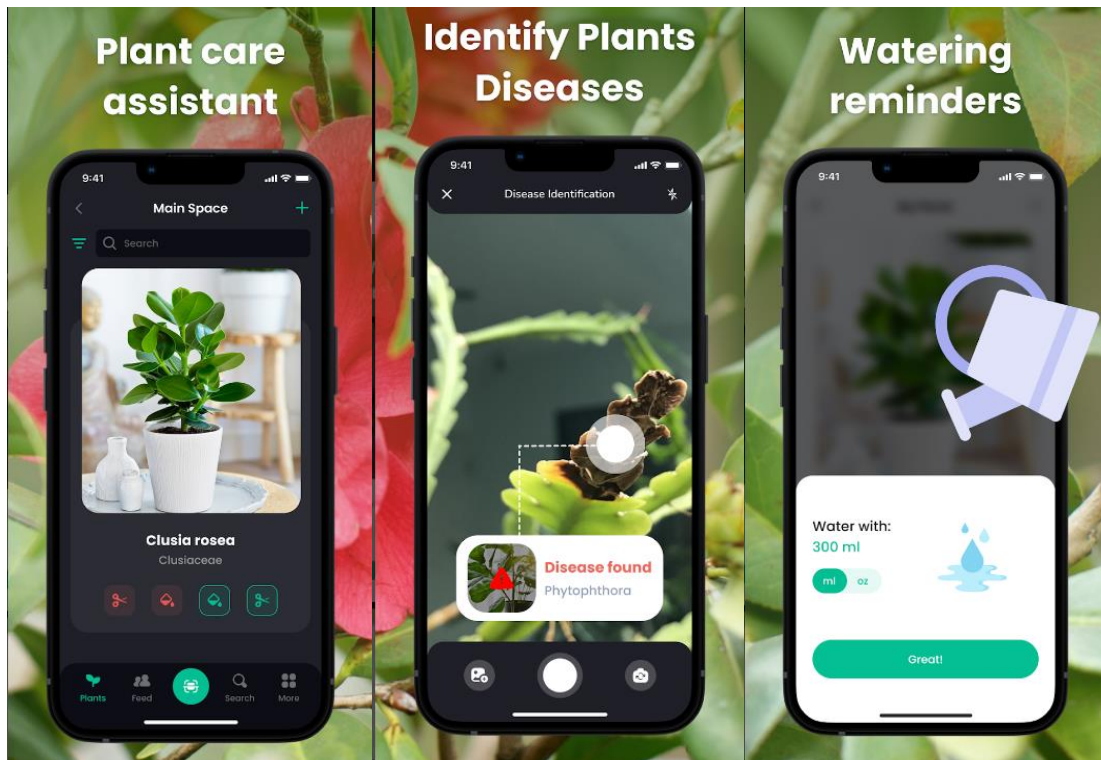


Рисунок 1.6 – Додаток PlantIn: Визначник Рослин

Визначник рослин PlantIn(рис. 6) – розпізнавач рослин українською. Надає детальні поради з догляду для кожної з них, нагадує про полив та пересадку, а також пропонує поради професійних ботаніків, які вирішують будь-які проблеми рослин.

За фото миттєво дізнаєшся, яка це рослина разом з експертною інформацією про догляд. Цей ідентифікатор використовує передові технології розпізнавання зображення та визначає більше ніж 17000 видів рослин. Розпізнавач на базі штучного інтелекту робить свою справу. Визначені рослини додаються до віртуальної колекції та можна отримати професійні поради по догляду за кожною.

Розпізнавач хвороб PlantIn дозволяє миттєво діагностувати будь-яку проблему та пропонує поради щодо лікування.

Більше не потрібно вгадувати, яке місце у домі буде підходити для рослини. Вимірювач світла допоможе знайти саме той куточок, де вона отримуватиме ідеальне світло для росту.

Блог надає необмежений доступ до сотень професійно написаних статей про садівництво та догляд за кімнатними рослинами [6].

На 2024 рік додаток безкоштовний та має рейтинг 4.5 із 5 на більш ніж 5 мільйонів кількості завантажень.

Висновки

Аналіз готових рішень наявних систем догляду за кімнатними рослинами: розумний вазон, гідропонна установка та краплинний полив для кімнатних рослин, а також відомих мобільних додатків: Plant Care Reminder, Flora Incognita та PlantIn: Визначник Рослин, свідчить про можливість постійного удосконалення та спрощення їх використання у сучасних умовах. Різна цінова політика, певні габарити та відсутність дистанційного керування. Постійна автоматизація умов життєдіяльності вимагає більш мобільні системи з можливістю підв'язати під смарт будинок.

Під час дослідження було розроблено основну концепцію власної моделі розумного горщика з дистанційним керуванням на прикладі порівняння готових на ринку рішень.

					КНУ.РБ.123.24.06.01. АПСГР	Арк.
Арк.	№ документа	Підпис	Дата			

2 ПРОЄКТУВАННЯ ДИСТАНЦІЙНОЇ СИСТЕМИ ДОГЛЯДУ ЗА РОСЛИНАМИ

2.1 Реалізація виконавчого модулю системи

Створення розумної системи догляду за рослинами базується на використанні датчиків, мікроконтролерів, зв'язку між пристроями і різноманітних алгоритмів та сценаріїв роботи для забезпечення потрібних функцій. Використання МК, таких як Arduino, ESP8266, ESP32 та інші, відкриває безмежні можливості для творчості та інженерних рішень. Однією з переваг є те, що можна налаштувати їх під власні потреби. Створити систему, що вирішує конкретну проблему або забезпечує певну функціональність, яка потрібна. На прикладі створення розумної системи догляду за рослинами(рис. 2.1).

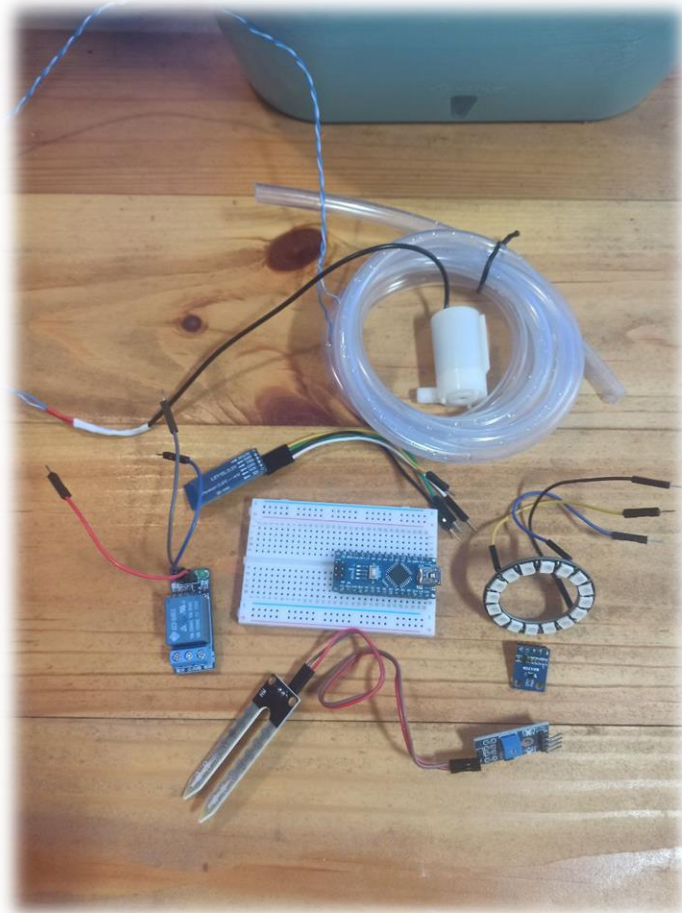


Рисунок 2.1 – Основні елементи системи

					КНУ.РБ.123.24.06.02.ПДСДР					
Змн.	Арк.	№ документа	Підпис	Дата	ПРОЄКТУВАННЯ ДИСТАНЦІЙНОЇ СИСТЕМИ ДОГЛЯДУ ЗА РОСЛИНАМИ					
Розробив	Духов							Літера	Аркуш	Аркушів
Перевірив	Кузнецов									
Н.контроль	Кузнецов							КІ-20		
Затвердив	Купін									

2.1.1 Вибір обладнання для реалізації системи

Після детального аналізу ринку готових рішень у категорії інтернет речей наявність готових рішень систем контролю догляду за рослинами та різноманітних варіантів вирішення наявних проблем, було визначено ключові елементи готових систем та складено список потрібного обладнання для реалізації власної системи.

Реалізація автономного поливу рослини.

Обираємо елементи для транспортування води з цистерни до рослини за сигналом від МК, що працює за певним алгоритмом та припиняє роботу у разі завершення процедури поливу. Для моделі було обрано занурюваний насос 5В з урахуванням його потужності для виконання завдання подачі води з резервуару до рослини.



Рисунок 2.2 – Занурюваний насос

Дана модель занурюваного насосу(рис. 2.2) може качати від 80 до 120 літрів рідини на годину, споживаючи при цьому від 2,5 до 6 В. Рідина, що перекачується насосом, може бути подібною до фізичного та хімічного складу з водою. Сам насос виконаний з пластику [7].

Характеристика:

- Напруга постійного струму: 5V;
- Робочий струм: 130-220mA;
- Потужність: 0,4-1.5W;
- Швидкість потоку: 80-120 л/год;
- Зовнішній діаметр вихід води: приблизно 7,5 мм;

- Внутрішній діаметр отвору води: приблизно 4,7 мм;
- Діаметр: близько 24 мм;
- Довжина: близько 45 мм;
- Висота: близько 33 мм;
- Довжина дроту: близько 15-20 см;
- Матеріал: пластик;
- Ціна: 83,60 грн.

Також потрібен шланг для води(рис. 2.3) до насосу з необхідним діаметром під'єднання у залежності від потрібної відстані. Для моделі обрано ПВХ трубку з внутрішнім діаметром 8 мм.



Рисунок 2.3 – Шланг для води

ПВХ трубка (полівінілхлоридна трубка) — це синтетичний термопластичний виріб, що відзначається високою хімічною стійкістю, довговічністю та механічною міцністю. Вона широко використовується в системах водопостачання. Легко монтується і має доступну ціну, що робить її економічно вигідним рішенням [8].

Характеристика:

- Внутрішній діаметр - 8 мм;
- Товщина стінки - 1 мм;
- Маса 1 метра - 40 грам;
- Зроблена із прозорого пластику;
- Робочий тиск/критичний тиск: 3/6 бар;
- Діапазон робочих температур – від -5°C до +60°C;
- Матеріал: гнучкий, еластичний.
- Ціна: 12,20 грн. за 1 м.

Реле. Для під'єднання насосу до мікроконтролера потрібне реле.

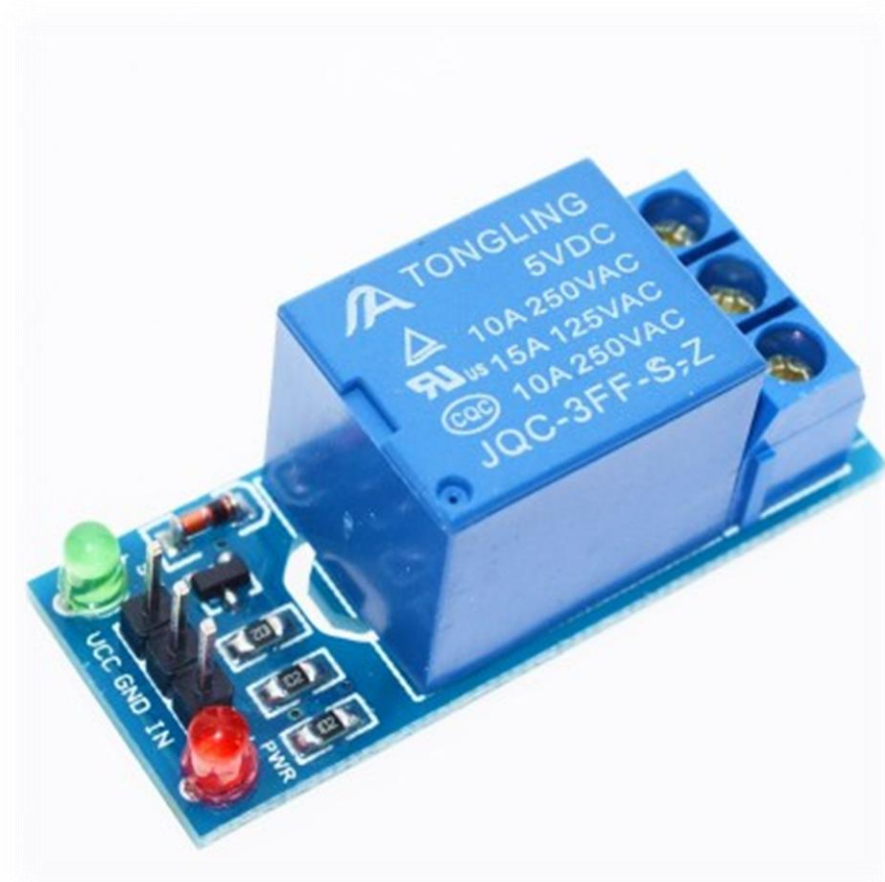


Рисунок 2.4 – Модуль реле

Реле 5В 1-канальний модуль(рис 2.4) вимагає 15-20мА, що дає можливість керувати ним безпосередньо з виходів мікроконтролера Arduino або подібних. Вмикається логічним нулем, вимикається логічною одиницею. На модулі є два світлодіоди: червоний сигналізує про наявність напруги живлення, зелений – про роботу реле [9].

Характеристики:

- Максимальний струм комутації реле 10А за 250В;
- На платі є світлодіод, що сигналізує про поточний статус реле;
- Розміри: 4.4см x 1.7см;
- Ціна: 46,20 грн.

Щоб дізнатися рівень вологості землі у горщику, можна використовувати модульний гігрометр або, простими словами, датчик вологості ґрунту. Завдяки даним, щодо вологості землі в якій знаходиться рослина, можна запобігти надмірного поливу, що може призвести до погіршення стану рослини чи навпаки повідомити, що рослина давно не поливалась та бажано підняти рівень вологості ґрунту до необхідних показників.



Рисунок 2.5 – Датчик вологості ґрунту

Датчик вологості ґрунту(рис 2.5) складається з двох частин: контактного щупа YL-69 і датчика YL-38, у комплекті йдуть дроти для підключення. Між двома електродами щупа YL-69 створюється невелика напруга. Якщо ґрунт сухий, опір великий і струм буде меншим. Якщо земля волога - опір менший, струм - більший. За підсумковим аналоговим сигналом можна судити про рівень вологості. Щуп YL-69 з'єднаний з датчиком YL-38 двома дротами. Крім контактів з'єднання з щупом, датчик YL-38 має чотири контакти для підключення до контролера.

Датчик YL-38 побудований на основі компаратора LM393, що видає напругу на вихід D0 за принципом: вологий ґрунт – низький логічний рівень, сухий ґрунт – високий логічний рівень. Рівень визначається граничним значенням, що можна регулювати за допомогою регулятора напруги. На контакт A0 подається аналогове значення, що можна передавати у МК для подальшої обробки, аналізу та прийняття рішень. Датчик YL-38 має два світлодіоди, що сигналізують про наявність напруги на датчик живлення та рівня цифрового сигналу на виході D0. Наявність цифрового виведення D0 та світлодіод рівня D0 дозволяє використовувати модуль без підключення до контролера [10].

Характеристики:

- Напруга живлення DC 3.3 - 5V;
- Струм споживання до 20mA;
- Розмір блоку компаратора 14x31мм;
- розмір датчика 20x60mm;
- Комплектація: плата в зборі, датчик, проводи;
- Ціна: 41,20 грн.

					КНУ.РБ.123.24.06.02.ПДСДР	Арк.
Арк.	№ документа	Підпис	Дата			

Реалізація додаткового освітлення для рослини.

Крім автономного поливу рослини, було б добре мати додаткове освітлення, тому що є періоди з меншою кількістю сонячних годин, що є досить суттєвим. Зазвичай використовують фітоосвітлюючі прибори такі як: фітолампи чи фітострічки, що мають спеціальний спектр світла. Для реалізації цього було обрано світлодіодну RGB матрицю де можна налаштувати потрібний спектр світла та робити експерименти з різними спектрами світла щодо впливу на рослину.

Було обрано матрицю на 16 елементів. Це оптимальний варіант для освітлення рослин в одному горщику середнього розміру.



Рисунок 2.6 – Led RGB матриця

Led RGB матриця(рис. 2.6) з 16 яскравих розумних світлодіодів NeoPixel, що розташовані у формі кільця зовнішнім діаметром 44,5 мм. Є можливість послідовного підключення.

Кожен світлодіод має свою адресу, оскільки кожен містить чіп управління. У кожному ~18мА постійного струму, тому колір світлодіодів буде стабільним навіть за перепадів напруги. Кожен піксель з трьох кольорів може змінювати яскравість у 256 значеннях та отримати 16777216 кольорів. Для роботи кільця необхідно 5В постійного струму (може працювати у діапазоні 4-7В).

У пристрої лише одна шина даних, протокол якої дуже чутливий. Для шини даних потрібен МК з обчисленнями у реальному часі, наприклад AVR, Arduino, PIC, mbed та подібні до них [11].

Характеристики:

- Чіп: WS2812B;
- Світлодіод: 5050 RGB;
- Напруга: 4,6-6В (5В рекомендовано);

- Діаметр: 45мм;
- Ціна: 104,60 грн.

Для вимірювання потрібного рослині світла використовується датчик освітленості приміщення(рис 2.7).

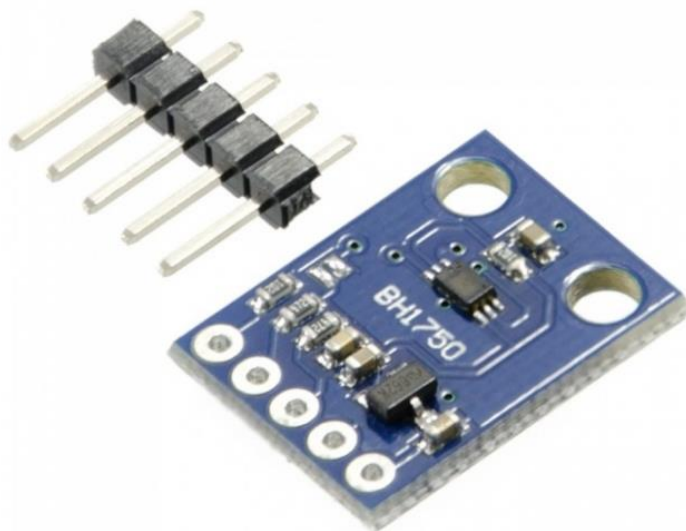


Рисунок 2.7 – Датчик освітленості

Цифровий датчик освітленості GY-302 на чіпі BH1750 призначений для вимірювання фонового освітлення. Має високу чутливість і поширений послідовний інтерфейс I2C. Спектр чутливості збігається з кривою чутливості людського ока. Усередині модуля розташований сам фотодатчик у вигляді фотодіода, підсилювач сигналу фотодіода, АЦП, що обробляє дані, перекладає все в одиниці виміру Люкс і передає до керуючого пристрою [12].

Характеристики:

- Тип: GY-302;
- Оригінальний чіп BH1750FVI ROHM;
- Вбудований сенсор і цифровий перетворювач;
- Нечутливий до фонового світла;
- Спектральна характеристика близька до візуальної чутливості;
- Для широкого діапазону, точність вимірювання - 1 люкс;
- Напруга живлення: 3 - 5 В;
- Діапазон даних: 0-65535 лк;
- Інтерфейс I2C;
- Розміри: ширина 13.9мм та довжина 18.5мм;
- Ціна: 61,50 грн.

Реалізація МК та способу зв'язку з ним.

Для зв'язку з МК було обрано Bluetooth з'єднання, що забезпечує надійний обмін даними на короткій відстані без дротів. Це дозволяє пристроям з'єднуватися між собою, передавати інформацію та взаємодіяти без необхідності фізичного підключення. Вибір Bluetooth з'єднання також обумовлено його невеликою потужністю споживання, що дозволяє працювати протягом тривалого часу без перерви. Таке рішення виявляється дуже зручним та ефективним, особливо у випадках, коли необхідно забезпечити бездротове з'єднання між різними пристроями або в умовах, де дротове підключення не є практичним.

HC-05/HC-06 - дуже поширені Bluetooth-модулі, які працюють на основі стандарту Bluetooth 2.0. Вони легко підключаються до Arduino через звичайні порти RX/TX і дозволяють здійснювати зв'язок між Arduino і смартфонами, планшетами та іншими пристроями.

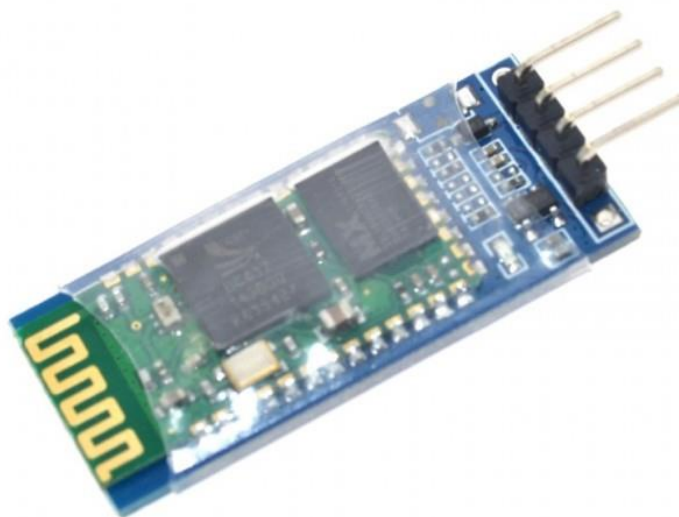


Рисунок 2.8 – Bluetooth модуль

Bluetooth-модуль HC-06(рис. 2.8) – простий спосіб бездротового дистанційного керування пристроєм за допомогою BT. З боку керованого пристрою, такого як Arduino, цей модуль виглядає як звичайний послідовний інтерфейс. Тому можна налагодити комунікацію з пристроєм на комп'ютері, а потім просто підключити цей BT-модуль.

З HC-06 можна керувати пристроєм безпосередньо зі свого смартфона. Установив одну з численних програм керування через BT. Можливо використовувати і для пересилання показань різноманітних детекторів. Робоча напруга цього BT-модуля - 3,3 В, але його входи толерантні до 5 В, тому він сумісний з усіма платами Arduino. BT-модуль HC-06 може виступати лише у slave-

режимі. Це означає, що він не може самостійно підключатися до інших пристроїв ВТ [13].

Характеристики:

- Напруга живлення: 3,3-6 В;
- Максимальний струм споживання: 45 мА;
- Швидкість передачі даних: 1200-1382400 бод;
- Дальність зв'язку при прямій видимості: 10 м;
- Ціна: 187,80 грн.

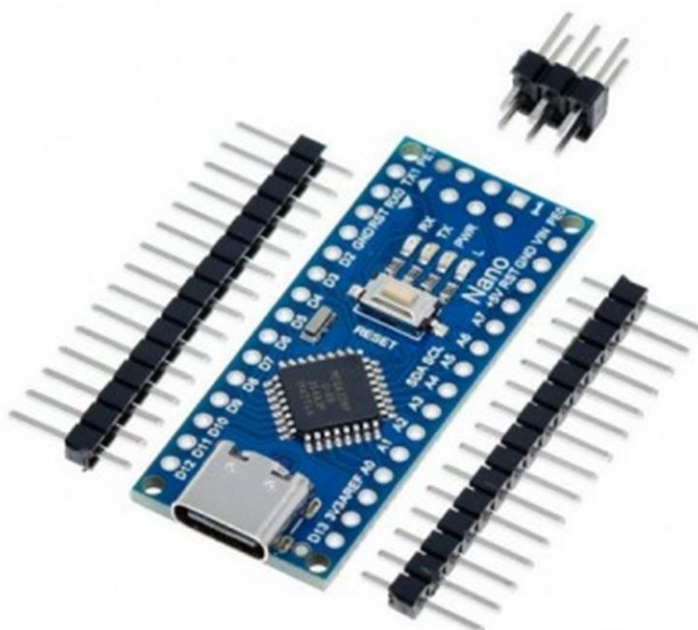


Рисунок 2.9 – Arduino Nano

Arduino Nano(рис 2.9)— це повнофункціональний мініатюрний пристрій на базі МК АТmega328 (Arduino Nano 3.0) або АТmega168 (Arduino Nano 2.x), адаптований для використання з макетної плати. За функціональністю пристрій схожий на Arduino Duemilanove, і відрізняється від нього розмірами, відсутністю роз'єму живлення, а також іншим типом (Mini-B) USB-кабелю. Arduino Nano розроблено і випускається фірмою Gravitech [14].

Arduino легко програмується за допомогою Arduino IDE, що надає широкий вибір бібліотек та прикладів для різноманітних проєктів. Підтримує різні мови програмування, включаючи С та С++, що робить доступною для широкого кола розробників, навіть початківців. Завдяки своїм компактним розмірам, потужності та легкості використання, Arduino Nano - популярний вибір для багатьох електронних проєктів, від автоматизації домашніх пристроїв до роботи з сенсорами та IoT (інтернет речей) проєктів.

Характеристики:

- Мікроконтролер: ATmega328;
- Робоча напруга (логічний рівень): 5В;
- Напруга живлення (рекомендована): 7-12В;
- Цифрові входи/виходи: 14;
- Аналогові входи: 8;
- Flash-пам'ять: 32 КБ;
- EEPROM: 1 КБ;
- Тактова частота: 16 МГц;
- Розміри плати: 1.85 см x 4.3 см;
- Ціна: 252,60 грн.

Реалізація системи живлення.

Для роботи обраного модулю потрібне 3,3-5В джерело живлення. Воно може бути автономним: на акумуляторі чи батарейках типу АА. Також можливо використання адаптеру живлення на 5В, наприклад, блок живлення для смартфона. Під'єднати дротами або на МК Arduino вже є вбудований інтерфейс для живлення.

Після вибору всіх основних елементів створення модулю системи контролю догляду за рослинами, з метою економічного обґрунтування реалізації проекту, було прораховану собівартість на 2024 рік(таблиця 2.1).

Таблиця 2.1 – Таблиця цін основних компонентів системи

№	Назва елемента	Ціна, грн.
1	Занурюваний насос для води (3-6В)	83,60
2	ПВХ трубка діаметром 8 мм 1 метр	12,20
3	Модуль реле 1 канал 5В	47,40
4	Гігрометр, датчик вологості ґрунту	41,20
5	Модуль світлодіодний кільцевий WS2812B 5050 RGB LED 16 bit	104,60
6	Цифровий датчик освітленості GY-302 на чіпі BH1750	61,50
7	Bluetooth-модуль HC-06	187,80
8	Arduino Nano	252,60
Усього:		790,90

2.1.2 Алгоритм роботи виконавчої частини системи

Написання алгоритму роботи системи автономного контролю є важливим етапом у створенні надійного модулю. Алгоритм визначає, як система буде реагувати на різні події та умови, забезпечуючи її належне функціонування. Є декілька ключових аспектів, що слід врахувати під час написання алгоритму роботи для системи автономного контролю.

Визначення цілей. Спочатку необхідно чітко визначити, які завдання повинна виконувати система. Це можуть бути вимірювання параметрів середовища, керування обладнанням, реагування на аварійні ситуації тощо.

Збір даних. Алгоритм повинен включати процеси збору даних з датчиків та інших джерел. Важливо забезпечити надійне та точне зчитування показників, що є основою для прийняття рішень.

Обробка даних. Зібрані дані повинні бути оброблені стосовно необхідної інформації. Це може включати фільтрацію шумів, нормалізацію значень, перевірку на аномалії та інші методи обробки [15].

Прийняття рішень. На основі оброблених даних система повинна приймати рішення. Алгоритм має передбачати різні сценарії та умови, визначаючи, які дії потрібно виконати в кожному конкретному випадку.

Виконання дій. Після прийняття рішення система повинна виконати відповідні дії. Це може бути керування пристроями, зміна режимів роботи, надсилання сповіщень тощо.

Моніторинг і корекція. Алгоритм повинен включати механізми моніторингу стану системи та її компонентів. У разі виявлення помилок або відхилень від нормальної роботи система повинна мати можливість корекції або сповіщення оператора.

Безпека. Автономні системи повинні мати вбудовані механізми безпеки для захисту від збоїв, неправильних дій або зовнішніх атак. Це включає резервні процедури, аварійні алгоритми та захист даних.

Тестування і вдосконалення. Написаний алгоритм необхідно ретельно протестувати в різних умовах, щоб переконатися в його надійності та ефективності. На основі результатів тестування алгоритм може бути вдосконалений і оптимізований.

Загалом написання алгоритму для системи автономного контролю вимагає детального планування, знань у галузі програмування та розуміння специфіки завдань, які система повинна виконувати. Цей процес є ключовим для створення надійної та ефективної автономної системи.

Основна ціль системи контролю догляду за рослинами - це допомога та полегшення рутинного процесу догляду за хатніми рослинами. А саме: своєчасний та збалансований полив рослин, а також компенсація нестачі світла.

Збір даних з датчиків вологості ґрунту та освітленості передаються по ВТ до смартфона через мобільний додаток, що відображає цю інформацію або сигналом для завершення роботи виконавчої частини модулю.

Основні показники для запуску виконавчих модулів системи - це період часу при настанні якого запускається робота виконавчого елемента системи та час

					КНУ.РБ.123.24.06.02.ПДСДР	Арк.
Арк.	№ документа	Підпис	Дата			

роботи виконавчого модулю, а у випадку коли є світлодіодна матриця ще й налаштування кольору та яскравості освітлення.

При проходженні заданого інтервалу реле буде замикати контакти для подачі води. Датчик вологості ґрунту буде зчитувати дані під час поливу та при отриманні потрібних показників подавати сигнал за яким реле буде від'єднувати контур до якого підключено насос. Детальний алгоритм роботи зображено на рисунку 2.10

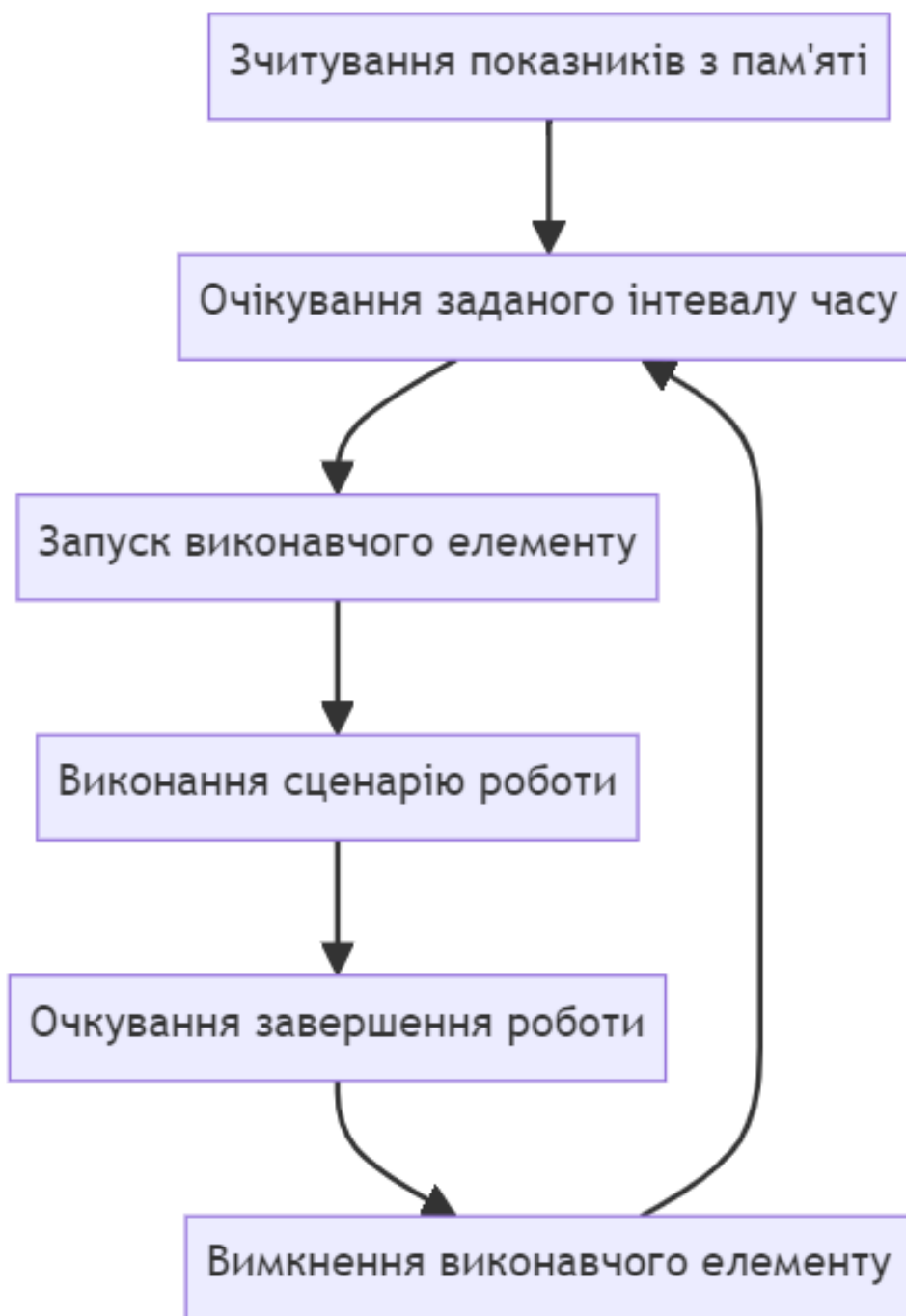


Рисунок 2.10 – Схема роботи алгоритму

2.1.3 Електрична схема модулю та поетапна збірка модулю

Важливим етапом в процесі збирання пристрою за схемою є уважне й акуратне виконання кожного кроку. Навіть найдрібніша помилка може призвести до неправильної роботи пристрою або навіть до його пошкодження. Тому перед тим, як почати збирання, слід ретельно ознайомитися зі схемою і матеріалами, а також переконатися, що всі необхідні деталі та інструменти доступні.

Перший крок у збиранні пристрою - це підготовка робочого місця(рис. 2.). Воно повинно бути чистим і добре освітленим, щоб уникнути помилок через недостатню видимість. Також важливо мати під рукою всі необхідні інструменти, які зазначені у схемі. Далі слід уважно розглянути схему і зрозуміти послідовність дій. Починають з найбільш простих деталей і поступово переходять до складніших. При збиранні слід дотримуватися всіх вказівок і рекомендацій, щоб уникнути помилок. Після завершення збирання важливо перевірити правильність підключення всіх деталей і дротів. Потім можна перейти до перевірки пристрою на працездатність. Це може включати в себе запуск тестових програм або проведення спеціальних вимірювань.

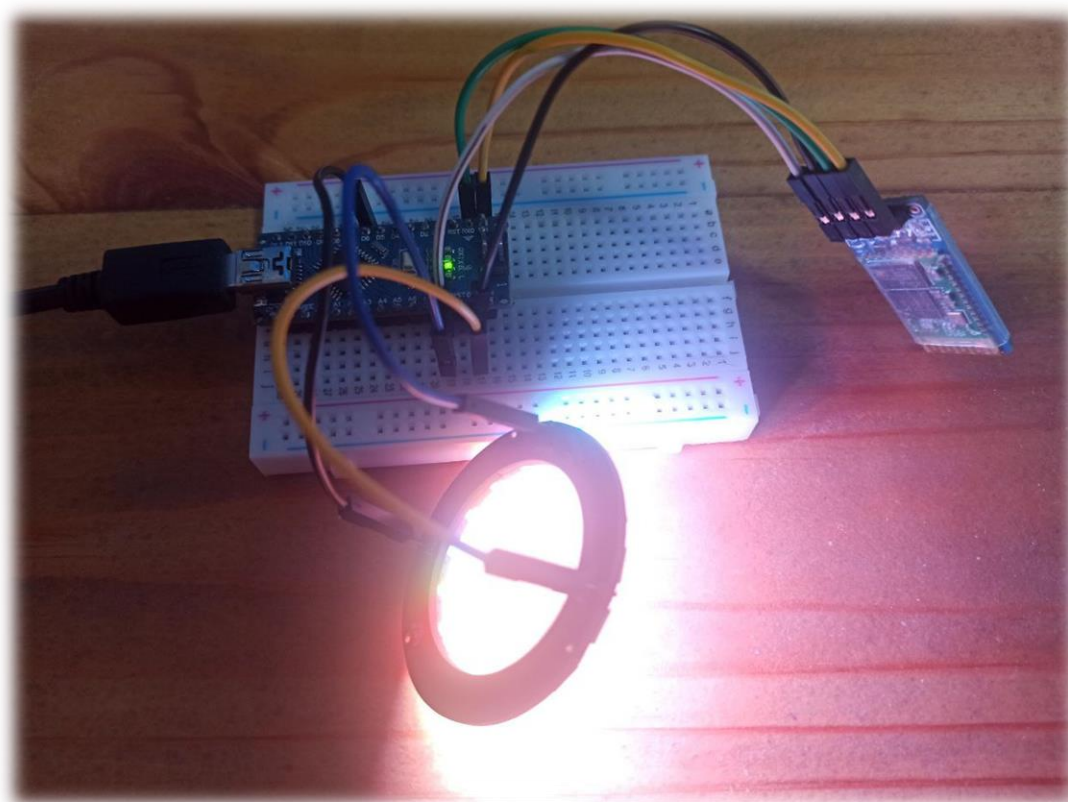


Рисунок 2.11 – Вигляд робочого місця

Підключення насосу та реле до мікроконтролера.

Для підключення реле до Arduino Nano треба підключити DC+ до 5V піну та DC- до GRD(земля). До піну IN буде йти сингал для керування реле до будь якого

					КНУ.РБ.123.24.06.02.ПДСДР	Арк.
Арк.	№ документа	Підпис	Дата			

цифрового піну Arduino на прикладі D2. Мінус насоса підключаємо до джерела живлення та плюс до СОМ контакту. Плюс джерела живлення NC чи NO в залежності від типу керування. Під'єднаємо ПВХ шланг до насоса [16].

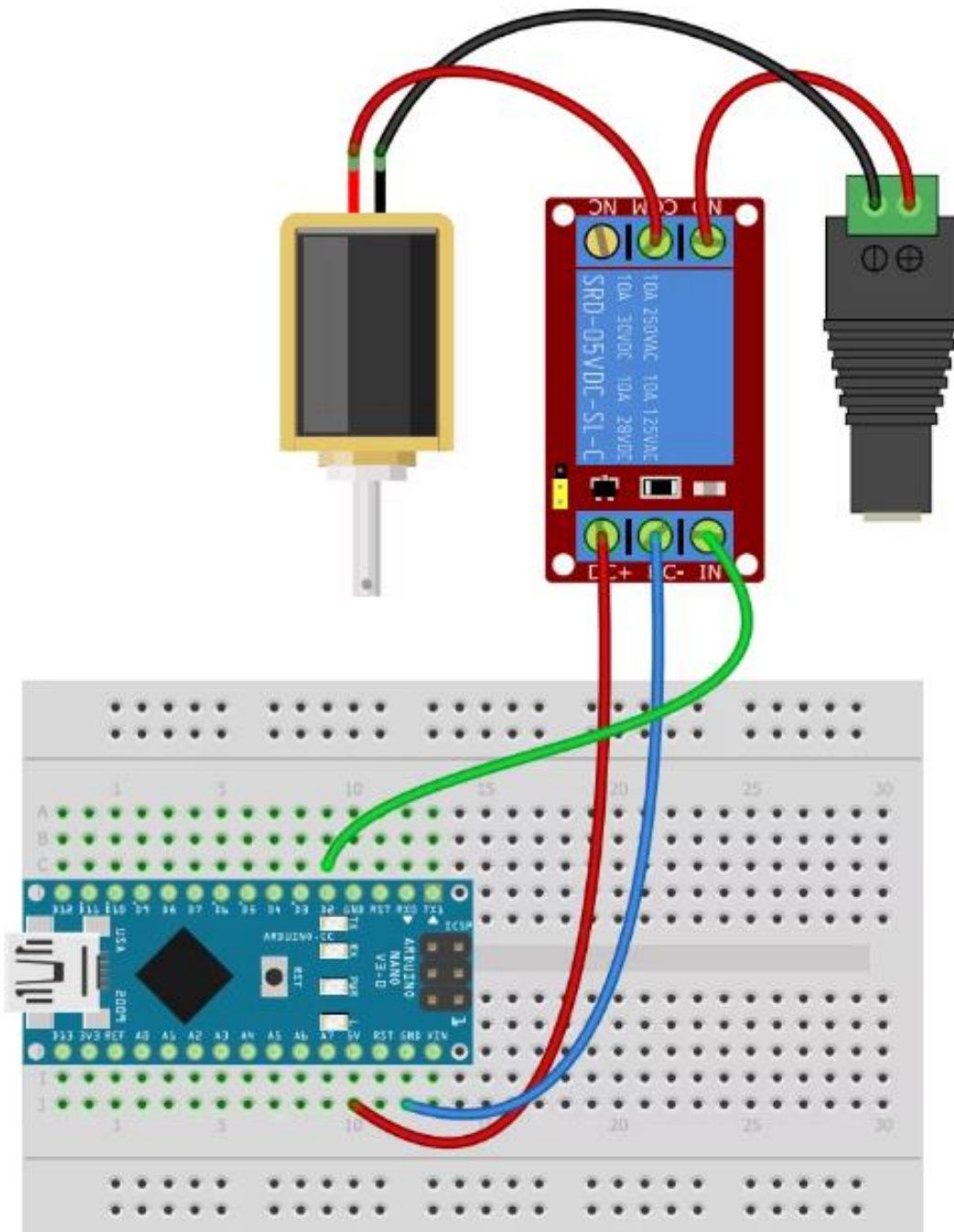


Рисунок 2.12 – Схема підключення реле та насосу до мікроконтролера

Після підключення елементів за схемою(рис. 2.12) треба перевірити її на дієздатність. Через Arduino IDE завантажуюємо код для тестування зібраної схеми насоса та реле.

Код тестування схеми:

```
const int relayPin = 7; // Пін до якого підключене реле
```

```
void setup() {
  pinMode(relayPin, OUTPUT); // Встановлюємо пін як вихід
  digitalWrite(relayPin, LOW); // Вимикаємо реле на початку
}
```

```
void loop() {
  // Приклад коду для керування насосом
  digitalWrite(relayPin, HIGH); // Увімкнути насос
  delay(10000); // Насос працює 10 секунд
  digitalWrite(relayPin, LOW); // Вимкнути насос
  delay(10000); // Насос вимкнений 10 секунд
}
```

Підключення датчика вологості ґрунту.

Для підключення датчика вологості ґрунту(рис. 2.13) треба позитивний (червоний) дріт датчика до позитивного (+) виходу на Arduino до 5V. А негативний (чорний) дріт датчика до GND виходу на Arduino. Інформація з датчика буде іти по "A0" до аналогового входу на Arduino (наприклад, до A0) [17].

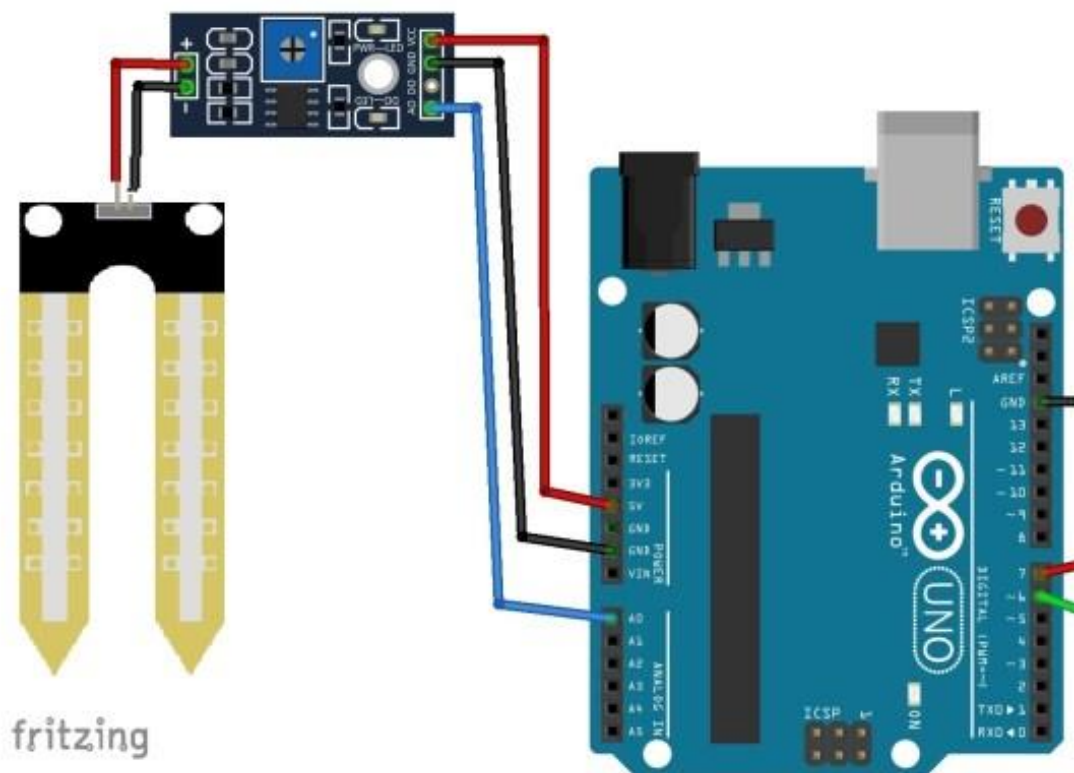


Рисунок 2.13 – Схема підключення датчика вологості

Код тестування схеми:

// Оголошення піну для зчитування вологості ґрунту

					КНУ.РБ.123.24.06.02.ПДСДР	Арк.
Арк.	№ документа	Підпис	Дата			

```

const int soilMoisturePin = A0;

void setup() {
  Serial.begin(9600); // Ініціалізація з'єднання з монітором
}

void loop() {
  // Зчитування значення вологості з датчика
  int soilMoistureValue = analogRead(soilMoisturePin);

  // Конвертація значення вологості у відсотки (при потребі)
  // Наприклад, якщо датчик має діапазон від 0 до 1023:
  float moisturePercentage = map(soilMoistureValue, 1023, 0, 0, 100);

  // Вивід значення вологості в консоль
  Serial.print("Soil Moisture: ");
  Serial.print(soilMoistureValue);
  Serial.print(" ");
  Serial.print(moisturePercentage);
  Serial.println("%");

  delay(1000); // Затримка для стабілізації відображення даних
}

```

Підключення адресної світлодіодної матриці.

Адресний світлодіод (LED) - це світлодіод, який має власну унікальну адресу, яка дозволяє індивідуально керувати його світінням. Це відкриває безліч можливостей в освітленні. Адресні світлодіоди можуть бути програмовані для відображення різних кольорів, виконувати складні анімації, імітувати ефекти пульсації, створювати змінювання яскравості та інші вражаючі ефекти.

WS2812B - це програмований адресний світлодіод, що має вбудований контролер управління. Він складається з червоного, зеленого та синього світлодіодів, які можуть бути керовані окремо. Цей чіп дозволяє створювати широкий спектр кольорів за допомогою програмного керування. WS2812B має простий протокол передачі даних, що дозволяє легко інтегрувати його в різноманітні проекти. Висока яскравість і енергоефективність роблять його популярним в застосуваннях освітлення, декорування та відображення інформації.

Цей чіп широко використовується в сучасних світлодіодних індикаторах, LED стрічках, підсвічуванні екранів та інших схожих пристроях. Він має компактний розмір і низьку вартість, що робить його ефективним рішенням для різних застосувань у різних галузях, включаючи IoT (інтернет речей), рекламу та розваги.

Схема підключення світлодіодної стрічки(рис. 2.14) чи матриці досить проста. Зазвичай світлодіоди у матриці організовані у вигляді рядків і стовпців. Підключіть живлення до відповідних контактів на світлодіодній матриці. Зазвичай це контакти VCC і GND. Підключіть сигнальні лінії даних (Data In) із вільного

цифрового піна контролера до вхідного світлодіоду у матриці. Це зазвичай є перший світлодіод у матриці [18].

Головне при підключенні адресної світлодіодної стрічки чи матриці - це обрати відповідний блок живлення. Чим більше світлодіодів буде підключено, тим більше ампер вони будуть споживати. Для вирішення такої проблеми існують спеціальні калькулятори, що розраховують кількість ампер для роботи матриць чи стрічок заданого розміру за кількістю світлодіодів та за типом чіпу.

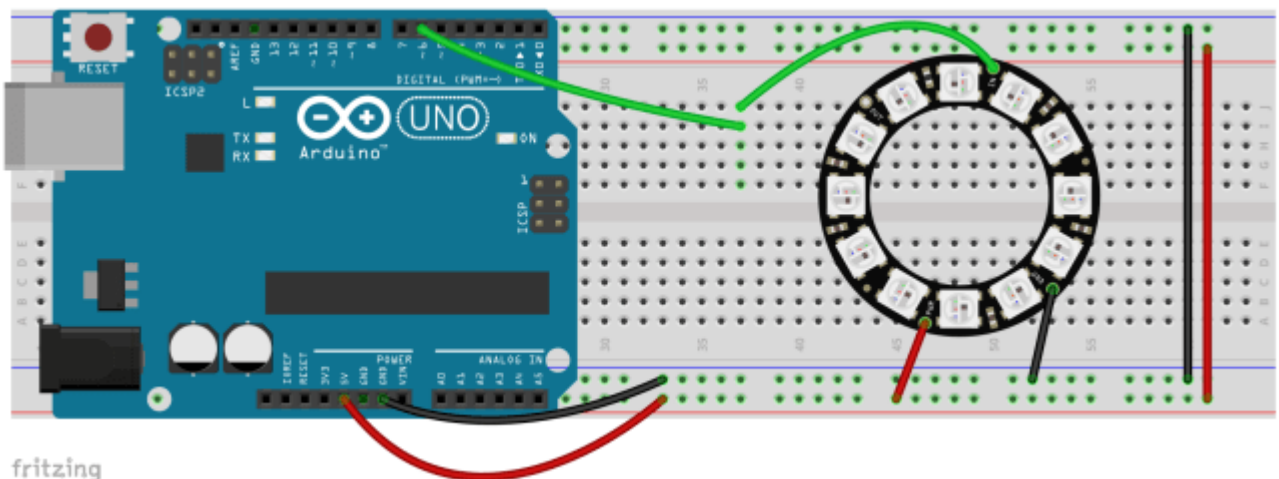


Рисунок 2.14 – Схема підключення адресних світлодіодів

Код тестування схеми:

```
#include <Adafruit_NeoPixel.h> // Бібліотека для роботи з світлодіодами

#define PIN      6 // Пін, до якого підключена світлодіодна матриця
#define NUM_PIXELS  16 // Загальна кількість світлодіодів у матриці

// Створення об'єкта для роботи з світлодіодами
Adafruit_NeoPixel pixels = Adafruit_NeoPixel(NUM_PIXELS, PIN, NEO_GRB
+ NEO_KHZ800);

void setup() {
  pixels.begin(); // Ініціалізація світлодіодної матриці
}

void loop() {
  // Масив з різними кольорами для всієї матриці
  uint32_t colors[] = {
    pixels.Color(255, 0, 0), // Червоний
```

```

pixels.Color(0, 255, 0), // Зелений
pixels.Color(0, 0, 255), // Синій
pixels.Color(255, 255, 0), // Жовтий
pixels.Color(255, 0, 255), // Пурпурний
pixels.Color(0, 255, 255), // Бірюзовий
pixels.Color(255, 255, 255), // Білий
pixels.Color(0, 0, 0) // Вимкнутий (чорний)
};

```

```

// Вмикаємо всі світлодіоди з визначеним кольором
for (int j = 0; j < sizeof(colors)/sizeof(colors[0]); j++) {
  for (int i = 0; i < NUM_PIXELS; i++) {
    pixels.setPixelColor(i, colors[j]); // Встановлення кольору для всіх
світлодіодів
  }
  pixels.show(); // Показати зміни на матриці
  delay(3000); // Затримка 3 секунда
}
}

```

Підключення датчику освітленості.

Модуль датчика освітленості GY-302, також відомий як модуль TSL2561, є компонентом, що використовується для вимірювання рівня освітленості в навколишньому середовищі. Модуль включає два фотодіоди: один для вимірювання світла у видимому спектрі, а інший - для вимірювання інфрачервоного світла. Цей модуль також має вбудований оптичний фільтр, що допомагає відокремлювати видиме світло від інфрачервоного. Для перетворення аналогових сигналів фотодіодів в цифровий формат, модуль GY-302 має вбудований ADC. Модуль може бути підключений до мікроконтролера або іншого пристрою через інтерфейс зв'язку, такий як I2C (Inter-Integrated Circuit) або аналоговий вихід.

I2C (Inter-Integrated Circuit) - це протокол зв'язку, що дозволяє різним пристроям комунікувати між собою за допомогою двох проводів: лінія даних (SDA) та лінія тактового сигналу (SCL). Цей протокол зв'язку був розроблений компанією Philips (тепер NXP Semiconductors) і знаходить широке застосування в багатьох електронних пристроях. Протокол I2C широко використовується в різних пристроях. Його простота та ефективність роблять його популярним в електроніці.

Підключення модуля датчика освітленості GY-302(рис. 2.15) до Arduino можна здійснити за допомогою протоколу зв'язку I2C. VCC підключається до живлення Arduino Nano (+5V). GND підключається до землі Arduino Nano (GND). SCL підключається до піна A5 на Arduino Nano. Це лінія з тактовим сигналом для протоколу I2C. SDA підключається до піна A4 на Arduino Nano. Це лінія передачі даних для протоколу I2C [19].

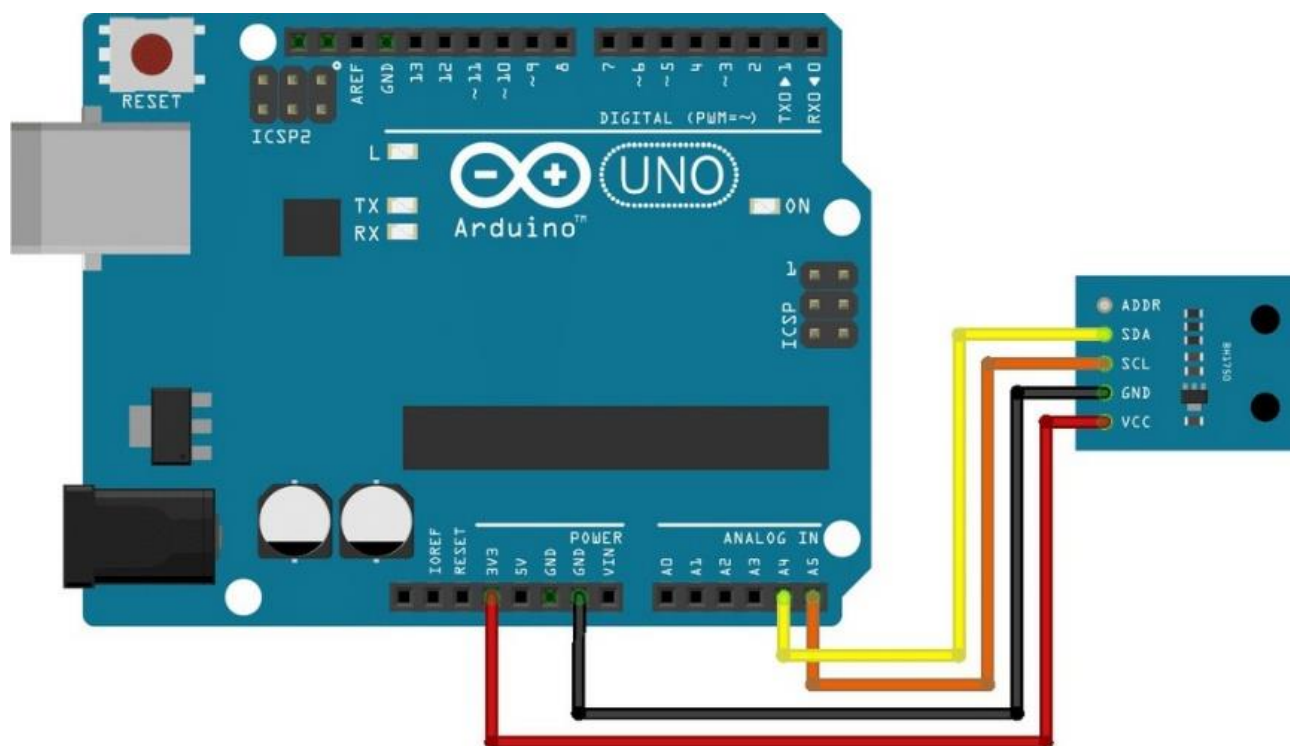


Рисунок 2.15 – Схема підключення адресних світлодіодів

Після підключення ви можете використовувати бібліотеку Wire.h для зчитування даних з датчика через інтерфейс I2C.

Код тестування схеми:

```
#include <Wire.h> // Включення бібліотеки для роботи з шиною I2C
```

```
#include <BH1750.h> // Бібліотеки для роботи з датчиком освітленості BH1750
```

```
BH1750 lightMeter; // Створення об'єкта для датчика освітленості
```

```
void setup() {
  Serial.begin(9600); // Ініціалізація зв'язку з монітором швидкістю 9600 біт/с
  lightMeter.begin(); // Ініціалізація датчика освітленості
  Serial.println("Running...");
}
```

```
void loop() {
  uint16_t lux = lightMeter.readLightLevel(); // Зчитування рівня освітленості з датчика
```

```
  Serial.print("Light: ");
```

```
  Serial.print(lux); // Вивід значення освітленості
```

```
  Serial.println(" lx");
```

```
  delay(500); // Затримка 500 мілісекунд перед наступною ітерацією циклу
```

```
}
```

Підключення Bluetooth-модулю.

Bluetooth-модуль HC-06 можна підключити до Arduino за допомогою інтерфейсу UART (Universal Asynchronous Receiver-Transmitter).

UART (Universal Asynchronous Receiver-Transmitter) - це апаратний пристрій або інтерфейс, який використовується для передачі та отримання даних між двома пристроями через два дроти. UART використовується для забезпечення зв'язку між МК, комп'ютерами та іншими електронними пристроями.

У режимі UART передача даних відбувається асинхронно, тобто без синхронізації тактовим сигналом між передавачем і приймачем. Замість цього, передавач і приймач узгоджуються заздалегідь на швидкість передачі даних, біти старту та стопу, інтервали передачі, і т.д.

У стандартному підключенні UART використовує два проводи: TX (Transmit) для передачі даних від передавача до приймача та RX (Receive) для отримання даних від приймача до передавача.

Передача даних починається з біта старту, який вказує на початок передачі, і завершується бітом стопу, що вказує на кінець передачі. Кількість бітів старту та стопу може варіюватися у залежності від налаштувань UART.

Швидкість передачі даних (бодов) визначається частотою тактового сигналу, який використовується для передачі та отримання даних. Швидкість може бути налаштована для різних значень, таких як 9600, 19200, 38400 бод і т.д.

Бод — це одиниця вимірювання швидкості передачі сигналів в телекомунікаціях та електроніці. Він визначає кількість змін сигналу (електричних станів або символів), які можуть бути передані за одну секунду. Названий на честь французького інженера Émile Baudot.

У загальному випадку, швидкість в бодах (baud rate) не завжди дорівнює кількості переданих бітів за секунду (bps, bits per second). Це пов'язано з тим, що одна зміна стану сигналу (один бод) може передавати більше ніж один біт інформації, залежно від використовуваного методу модуляції.

UART широко використовується в електроніці для забезпечення зв'язку між пристроями, і він є стандартним інтерфейсом для зв'язку між МК, сенсорами, модулями Bluetooth, GPS-приймачами та іншими пристроями.

Отже, для підключення Bluetooth треба:

- VCC підключається до живлення Arduino (+5V);
- GND підключається до землі Arduino (GND);
- TX Підключається до пін RX (приймач) Arduino (D0);
- RX Підключається до пін TX (передавач) Arduino (D1).

Після підключення можна використовувати бібліотеку SoftwareSerial або вбудований UART-інтерфейс (наприклад, Serial) для зчитування та відправлення даних через Bluetooth-модуль HC-06. Для цього можна використовувати стандартні функції зчитування і запису даних з інтерфейсу Serial, такі як Serial.read() та Serial.write().

При такому підключенні (рис. 2.16) модуль заважатиме завантаженню прошивки (вона завантажується по RX TX), на момент завантаження потрібно відключити дроти від пінів RX та TX Arduino. З модулем можна комунікувати за допомогою штатного Serial, фактично він дублюватиме монітор COM порту [20].

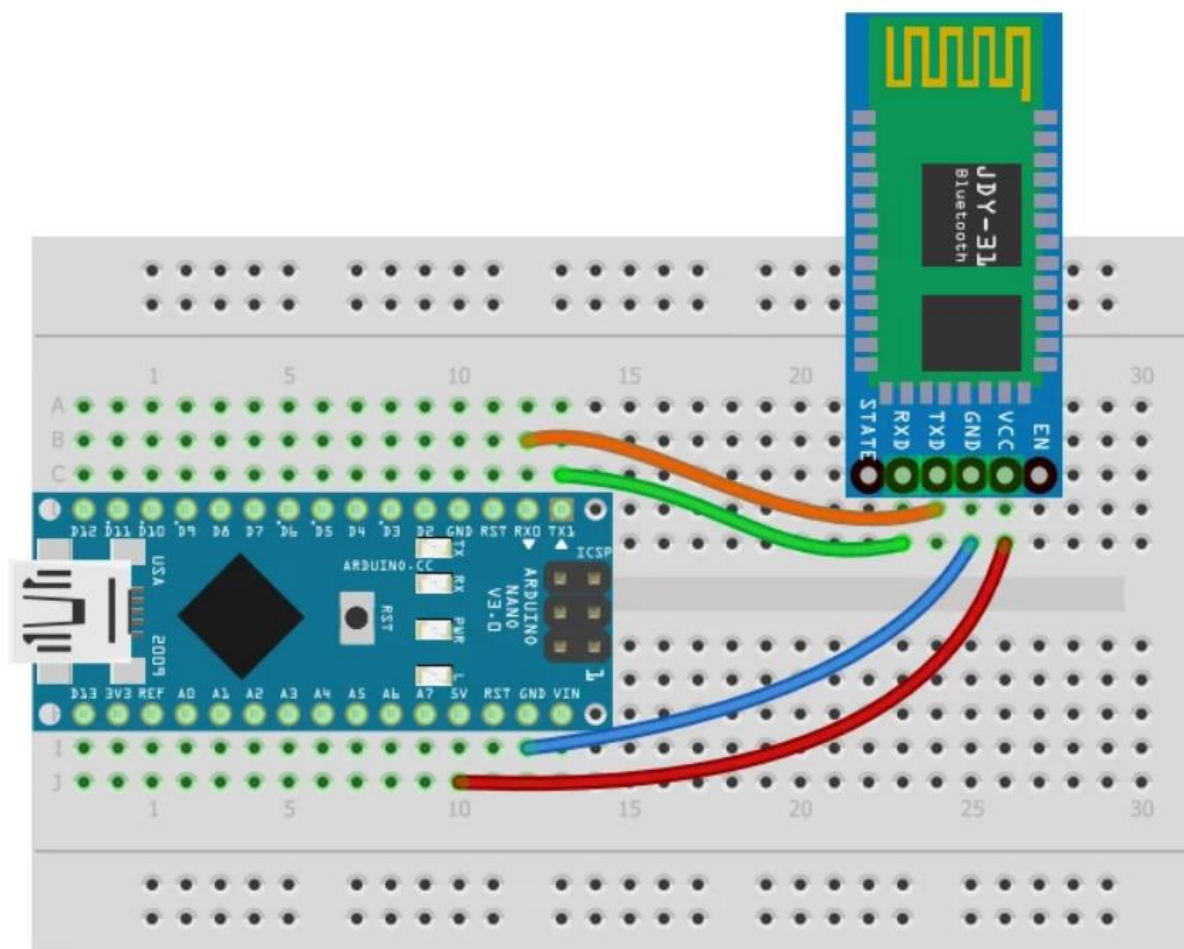


Рисунок 2.16 – Схема підключення Bluetooth-модулю по RX TX

Для тесту на Android можна використовувати програму Serial Bluetooth Terminal. Заходимо в програму, відкриваємо налаштування зліва, Devices, вибираємо Bluetooth-модуль і підключаємося. Пишемо до терміналу – модуль нам відповідає.

Код тестування схеми:

```
void setup() {
  Serial.begin(9600);
  Serial.setTimeout(100);
}

void loop() {
  if (Serial.available()) {
    Serial.write(Serial.read());
  }
}
```

Наприклад, підключимо модуль за другою схемою (на пini D2 та D3) та завантажимо просту програму, яка буде пересилати дані з програмного Serial (на

пінах D2 та D3, Bluetooth) на апаратний (який ми можемо дивитися через монітор порту в Arduino IDE).

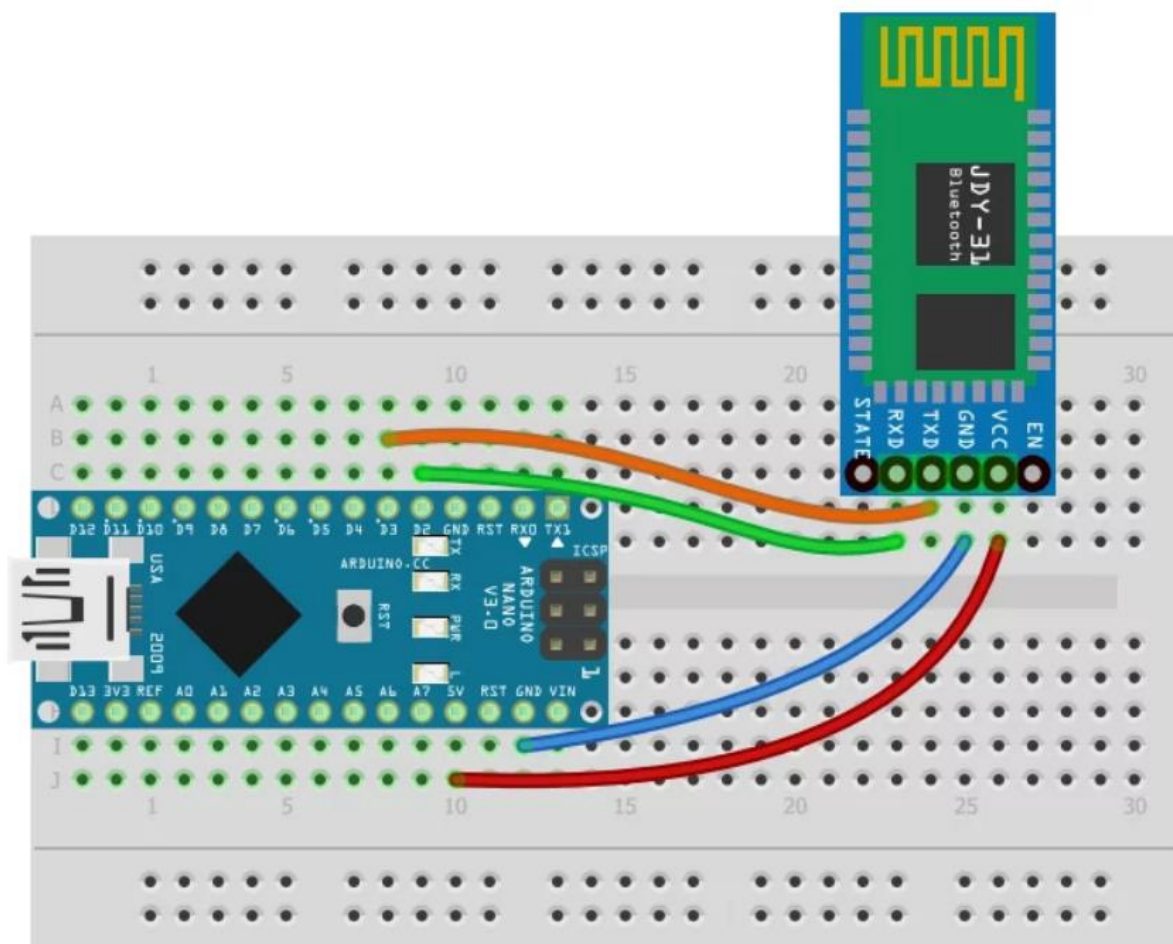


Рисунок 2.17 – Схема підключення а Bluetooth-модуль по цифровим пінам

При такому підключенні(рис. 2.17) модулю нічого не заважає але для роботи з ним доведеться використовувати стандартну бібліотеку програмного Serial - SoftwareSerial. Бібліотека SoftwareSerial для Arduino дозволяє створювати додаткові послідовні порти (UART), використовуючи будь-які цифрові виводи. Це корисно, коли потрібні додаткові послідовні порти але у Arduino є лише один апаратний послідовний порт.

Код тестування схеми:

```
#include <SoftwareSerial.h>
```

```
SoftwareSerial mySerial(3, 2); // RX, TX
```

```
void setup() {
  Serial.begin(9600);
  mySerial.begin(9600);
}
```

```
void loop() {
```

```

if (mySerial.available()) {
  Serial.write(mySerial.read());
}
if (Serial.available()) {
  mySerial.write(Serial.read());
}
}

```

Збірка всього модулю.

Отже, для підключення всіх компонентів до МК потрібно такі порти:

- 3 аналогових піна – А0 для датчика вологості ґрунту, А4 та А5 для датчику освітленості;
- 2 цифрові піна – D2 для реле та D3 для дату піна адресної матриці;
- RX та TX – Bluetooth-модуль;
- VCC(5V) та GRN – блок живлення.

Вся схема модулю зображена на додатку А.

2.2 Програмування керуючого елемента системи

Для керування Bluetooth-пристроєм потрібно врахувати кілька ключових етапів: вибір платформи та інструментів розробки та написання коду для підключення та взаємодії з пристроєм. Створено додаток на Android з використанням Expo та React Native.

Expo та React Native є потужними інструментами для створення мобільних додатків, які можуть керувати Bluetooth-пристроями. Expo спрощує розробку та тестування додатків, надаючи набір бібліотек і сервісів, що спрощують створення і публікацію додатків.

JavaScript є основною мовою програмування для розробки додатків з використанням React Native. Замість HTML тегів та CSS використовуються React Native компоненти та StyleSheet для розмітки додатку. Для уникнення проблем типізації даних у JS використовують TypeScript.

А для прошивки Arduino у Arduino IDE використовується мова Arduino Sketch яка в свою чергу схожа на C++.

2.2.1 Технологія з'єднання Bluetooth та його використання

Bluetooth є широко поширеною технологією бездротового зв'язку, що дозволяє обмінюватися даними на коротких відстанях між різними пристроями. Розроблена компанією Ericsson в 1990-х роках, вона стала стандартом у галузі бездротових технологій. Bluetooth використовує радіохвилі в діапазоні частот 2,4 ГГц, що забезпечує достатню пропускну здатність для більшості застосувань [21].

Завдяки низькому енергоспоживанню, пристрої з Bluetooth можуть працювати тривалий час без необхідності частого підзаряджання.

У побуті Bluetooth(рис. 2.18) часто використовується для підключення бездротових навушників, колонок та інших аудіопристроїв до смартфонів та

					КНУ.РБ.123.24.06.02.ПДСДР	Арк.
Арк.	№ документа	Підпис	Дата			

інших джерел звуку. Це забезпечує свободу руху без обмеження кабелями, що підвищує комфорт користувача.



Рисунок 2.18 – Bluetooth у побуті

Bluetooth також знайшов застосування в Інтернеті речей (IoT), де він використовується для з'єднання різних пристроїв та датчиків в єдину мережу. Це дозволяє здійснювати автоматизований контроль та моніторинг різних аспектів повсякденного життя, таких як освітлення, опалення та безпека в розумному будинку.

Bluetooth продовжує вдосконалюватися. Нові версії стандарту пропонують вищу швидкість передачі даних, більшу надійність з'єднання та покращену безпеку. Зокрема, версії Bluetooth 5.0 і 5.1 пропонують розширений діапазон та покращені можливості для підключення пристроїв.

Загалом, Bluetooth є ключовою технологією, яка зіграла важливу роль у розвитку сучасного цифрового світу. Його багатогранність та універсальність дозволяють застосовувати його у різних галузях, покращуючи повсякденне життя та сприяючи розвитку нових технологій.

Для тестування своїх ВТ пристроїв можна використовувати різні програми для відправки команди наприклад Bluetooth Serial Terminal який має термінал для відправки команд.

Також є Bluetooth Electronics - це додаток для мобільних пристроїв, який дозволяє взаємодіяти з електронними пристроями через Bluetooth з використанням Arduino або інших мікроконтролерів. Цей додаток постачається з бібліотекою, що містить 11 прикладів Bluetooth для Arduino. Його також можна використовувати з Raspberry Pi. Великий вибір елементів керування, включаючи кнопки, вимикачі, повзунки, колодки, світильники, датчики, клеми, акселерометри та графіки

2.2.2 Програмування мікроконтролера Arduino для виконавчого модулю

Мова програмування, яка використовується для розробки програмного забезпечення для платформи Arduino, називається мовою Arduino (або Arduino Sketch). Фактично це спеціалізована версія мови C++, оптимізована для спрощення розробки проектів, що використовують МК Arduino.

Мова Arduino дуже схожа на звичайну мову програмування C++, але має деякі специфічні особливості, що полегшують роботу з МК та апаратними компонентами.

В мові Arduino не потрібно задавати відправну точку, як в мові C++. Дійсно, під час запуску програми, вона завантажує те, що називається об'єкт в об'єктно-орієнтованому програмуванні ООП в пам'ять. Об'єкти в ООП належать класам. Для кожного об'єкта того ж класу стан визначається у вигляді набору атрибутів або елементів, котрі можна розглядати як змінні, представлені як набір даних різних типів в пам'яті. Стан об'єкта може управлятися методами: набір станів, призначених для виконання заданої операції по зміні або забезпеченню стану об'єкта [22].

Скетч Arduino складається з самостійного файлу, в якому, на відміну від мови C++, потрібно визначити принаймні, дві секції: перша називається `setup()`, а друга `loop()`. Змінні, доступні з обох секцій програми, повинні бути оголошені за їх межами, як глобальні змінні.

Як тільки програма запуститься, виконуються операнди, розміщені в блоці `setup()`: вони призначені для ініціалізації значень змінних на початку запуску, а також для налаштування портів периферії Arduino. Після закінчення обробки `setup()` Arduino починає циклічне виконання інструкцій в блоці `loop()`. Після виконання всіх операндів, цикл повторюється знову і знову.

2.2.3 Програмування мобільного додатку для керування системою в середовищі програмування EXPO

Expo - це фреймворк і платформа для розробки мобільних додатків, побудованих на React Native. Спрощує створення кроссплатформових додатків для iOS та Android, дозволяючи розробникам використовувати знайомі інструменти та мову програмування JavaScript [23].

Проект Expo був запущений в 2014 році з метою полегшити розробку мобільних додатків. Він об'єднав у собі різні інструменти, спрямовані на полегшення процесу розробки. Протягом наступних років Expo здобував популярність завдяки своїм зручним інструментам та багатофункціональній платформі. Розробники відзначали його простоту використання та широкий функціонал. Вирішальним моментом стало впровадження Expo SDK - набору інструментів та API, що дозволяють розробникам використовувати різноманітні функціональні можливості для створення мобільних додатків.

Основні особливості Expo включають:

					КНУ.РБ.123.24.06.02.ПДСДР	Арк.
Арк.	№ документа	Підпис	Дата			

- Простота початку роботи: Ехро надає набір інструментів та бібліотек, які дозволяють розпочати роботу безпосередньо, оминаючи складні налаштування та конфігурації;
- Кроссплатформовість: Додатки, розроблені з використанням Ехро, можуть працювати як на пристроях iOS, так і на пристроях Android, використовуючи спільний код на JavaScript;
- Багато готових компонентів та API: Ехро надає широкий набір готових компонентів інтерфейсу та API для доступу до пристрійних функцій, таких як камера, геолокація, сповіщення та інше;
- Зручна розробка та відлагодження: Ехро забезпечує зручне середовище розробки з можливістю живого оновлення, швидкої зворотньої зв'язку та інструментами для відлагодження;
- Хмарні сервіси: Ехро пропонує хмарні сервіси для хостингу, збирання та публікації додатків, що спрощує процес розгортання додатків;
- Розширюваність: В Ехро також можна інтегрувати власні компоненти React Native та розширювати його функціональність за потреби.

У цілому, Ехро є потужним інструментом для розробки мобільних додатків на базі React Native, особливо для початківців розробників та тих, хто прагне швидко створити прототип або мінімальний життєздатний продукт додатка.

React Native - фреймворк інтерфейсу користувача з відкритим кодом, створений Meta Platforms що базується на JavaScript, і дозволяє розробникам створювати мобільні додатки для платформ Android та iOS з використанням одного і того ж коду.

Основною перевагою React Native є можливість використання React-підходу для розробки мобільних додатків, що дозволяє розробникам ефективно використовувати компоненти для побудови інтерфейсу користувача. React Native має велику та активну спільноту розробників, що забезпечує наявність численних бібліотек, пакетів та інструментів, які полегшують процес розробки та дозволяють додавати нові функціональні можливості [24].

Основна концепція React полягає в тому, що інтерфейс користувача складається з компонентів, що відображають стан даних і автоматично оновлюються при його зміні. React використовує віртуальний DOM для ефективного оновлення лише тих частин інтерфейсу, які змінилися [25].

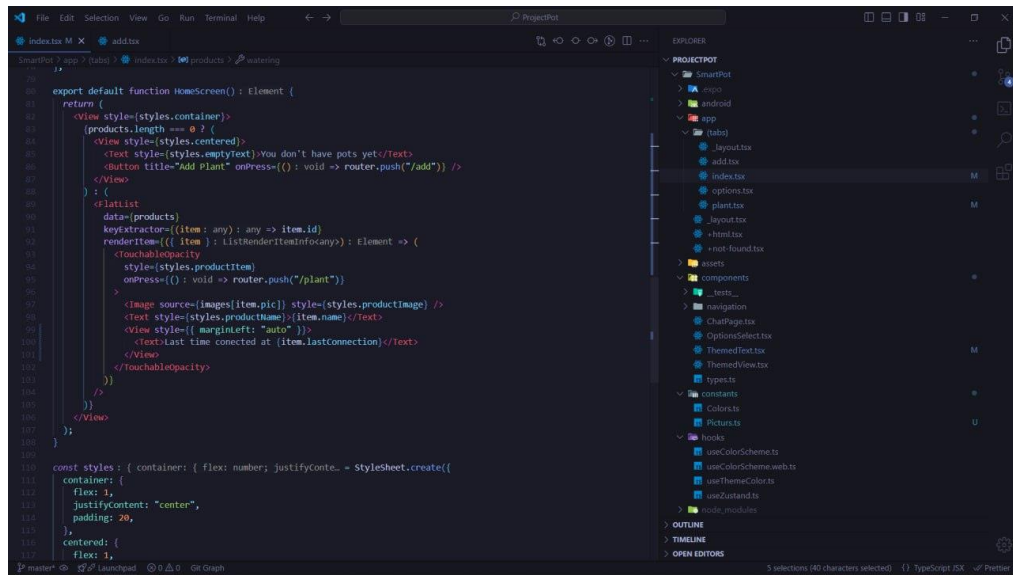
DOM - це скорочення від "Document Object Model" (Модель Об'єктів Документу). Це стандарт, який використовується для представлення веб-сторінок у вигляді дерева об'єктів. Веб-браузер використовує DOM для представлення структури і вмісту веб-сторінки, що дозволяє JavaScript і CSS маніпулювати цим вмістом.

DOM представляє кожен елемент сторінки (такі як заголовки, абзаци, зображення тощо) як об'єкт з властивостями, які можна змінювати за допомогою JavaScript. Взаємодія з DOM дозволяє JavaScript змінювати структуру, вигляд і поведінку веб-сторінки без необхідності перезавантаження сторінки.

Основні переваги React включають простоту використання, швидкодію розробки, зручність у відлагодженні і можливість повторного використання

					КНУ.РБ.123.24.06.02.ПДСДР	Арк.
Арк.	№ документа	Підпис	Дата			

компонентів. Він також має велику спільноту розробників та багато розширень, що полегшують його використання.



```

export default function HomeScreen(): Element {
  return (
    <View style={styles.container}>
      {products.length === 0 ? (
        <View style={styles.centered}>
          <Text style={styles.emptyText}>You don't have pots yet.</Text>
          <Button title="Add plant" onPress={() => router.push("/add")} />
        </View>
      ) : (
        <FlatList
          data={products}
          keyExtractor={(item: any) => item.id}
          renderItem={({ item }: ListRenderItemInfo<any>): Element => (
            <TouchableOpacity
              style={styles.productItem}
              onPress={() => router.push("/plant")}
            >
              <Image source={images[item.pic]} style={styles.productImage} />
              <Text style={styles.productName}>{item.name}</Text>
              <View style={{ margin: 10, align: "center" }}>
                <Text>Last time connected at {item.lastConnection}</Text>
              </View>
            </TouchableOpacity>
          )
        </FlatList>
      )}
    </View>
  );
}

const styles = {
  container: { flex: 1, justify: "center", align: "center" },
  centered: { flex: 1, align: "center" },
  emptyText: { text: "You don't have pots yet." },
  productImage: { width: 50, height: 50, margin: 10 },
  productName: { text: "Product Name" },
  lastConnection: { text: "Last time connected at" }
};

```

Рисунок 2.19 – Написання інтерфейсу через JSX

JSX (JavaScript XML) (рис. 2.19) - це синтаксичне розширення для JavaScript, що дозволяє використовувати HTML-подібний синтаксис для створення елементів React. Це робить код більш зрозумілим і зручним для читання, оскільки він виглядає дуже схожим на звичайний HTML, але з можливістю використовувати всю міць JavaScript. JSX виглядає як HTML, що спрощує його сприйняття для розробників з досвідом роботи з веб-технологіями.

У JSX можна використовувати JavaScript вирази, обгорнуті у фігурні дужки. Це дозволяє динамічно змінювати вміст компонентів на основі стану чи пропсів. Перед виконанням в браузері JSX компілюється в JavaScript функції.

JSX дозволяє описувати, як повинен виглядати інтерфейс у будь-який момент часу, а React займається оновленням реального DOM для відповідності цьому опису.

React Hooks - це функціональність, що була представлена в React 16.8. Вони дозволяють реакт компонентам використовувати стан та інші можливості React без необхідності використання класових компонентів.

Деякі з основних вбудованих хуків включають useState, useEffect, useContext, useReducer, useCallback, useMemo та useRef. Кожен з них дозволяє розробникам реагувати на різні аспекти компонентів, такі як збереження стану, реагування на зміни життєвого циклу, спільне використання стану між компонентами, роботу зі змінними, які зберігаються поза рендером тощо. Використання хуків дозволяє писати більш простий та зрозумілий код, а також полегшує перехід на функціональний підхід до написання компонентів в React.

TypeScript - це мова програмування, що є надбудовою над JavaScript, що додає статичну типізацію. Вона була розроблена компанією Microsoft і вперше випущена у 2012 році. TypeScript зберігає всі можливості JavaScript і додає

додаткові функції, такі як типізація та підтримка сучасних можливостей ECMAScript.

TypeScript додає можливість визначати типи для змінних, функцій, об'єктів тощо. Це дозволяє виявляти помилки на етапі компіляції, що робить код більш надійним і зрозумілим. TypeScript підтримує інтерфейси та класи, що дозволяє використовувати об'єктно-орієнтоване програмування. Це полегшує створення масштабованих і підтримуваних додатків. TypeScript є надбудовою JavaScript, тому будь-який валідний код JavaScript також є валідним кодом TypeScript. Це дозволяє поступово переходити від JavaScript до TypeScript. Полегшує розробку великих додатків завдяки своїм об'єктно-орієнтованим можливостям [26].

TypeScript став популярним вибором серед розробників, які хочуть додати строгість та структуру JavaScript-додаткам, зберігаючи при цьому сумісність із існуючим JavaScript-кодом.

Щоб почати створювати мобільний додаток треба завантажити Node.js.

Node.js - це серверне середовище виконання JavaScript, що дозволяє запускати JavaScript-код поза веб-браузером. Створено Раяном Далем у 2009 році і побудована на рушії V8, який розроблений компанією Google для виконання JavaScript у веб-браузері Google Chrome [27].

Node.js використовує неблокуючий ввід/вивід, що дозволяє обробляти велику кількість одночасних запитів без блокування потоку виконання. Це робить його дуже ефективним для створення мережеских додатків. Незважаючи на асинхронність, Node.js працює в одному потоці. Це означає, що він не створює нових потоків для кожного запиту, що зменшує накладні витрати на створення і управління потоками.

Node.js поставляється з менеджером пакетів (npm), який дозволяє легко встановлювати та управляти бібліотеками і залежностями. NPM є одним з найбільших репозиторіїв програмного забезпечення у світі. Він підтримує модульну архітектуру, де кожен файл є окремим модулем. Це сприяє організації коду і повторне використання компонентів.

Для виконання команд Node.js рекомендується використовувати Git Bash, особливо на операційній системі Windows. Git Bash - це середовище командного рядка, що забезпечує емуляцію командного рядка Unix і Git, дозволяючи користувачам виконувати команди, подібні до тих, що використовуються в Unix-системах [28]. Git Bash поставляється з вбудованою підтримкою Git, що полегшує роботу з системами контролю версій. Ви можете виконувати всі Git-команди безпосередньо з Git Bash. Використання Git Bash може спростити налаштування та управління проектами, особливо коли потрібно працювати з багатьма залежностями та пакетами, встановленими через npm.

Після встановлення середовища розробки можна створити новий проєкт за допомогою команди `npm create-express-app`. Ця команда генерує базову структуру проєкту, яка включає всі необхідні файли та налаштування для початку роботи з Express.

Code smell - це термін, що використовується для опису явних ознак того, що код може бути погано спроектований або написаний, і, можливо, вимагає рефакторингу. Хоча код smell не обов'язково є помилкою в коді, він може бути

					КНУ.РБ.123.24.06.02.ПДСДР	Арк.
Арк.	№ документа	Підпис	Дата			

індикатором потенційних проблем, які можуть впливати на якість, продуктивність або підтримку коду.

Основні ознаки code smell:

- Дублювання коду: повторення одного й того ж коду в різних місцях програми. Це може ускладнити підтримку та призвести до непослідовності;
- Довгі методи: методи, які занадто довгі та складні для розуміння. Це може призвести до того, що код буде важко підтримувати й розуміти;
- Великий клас: клас, який виконує надто багато функцій або містить занадто багато властивостей та методів;
- Невідповідні імена: імена змінних, функцій або класів, які не відображають їхнє призначення, що ускладнює розуміння коду;
- Бездоглядний стан: стан, який є невизначеним або може призвести до неправильного поведінки програми;
- Надмірна складність: код, який використовує складні конструкції або алгоритми без необхідності, що ускладнює розуміння та підтримку;
- Залежність від глобальних змінних: використання глобальних змінних, які можуть створити побічні ефекти та порушити передбачуваність коду;
- Занадто багато параметрів: методи або функції, які приймають занадто багато параметрів, можуть бути складними для розуміння та використання;
- Приховані побічні ефекти: функції або методи, які не явно вказують, що вони можуть змінювати зовнішній стан або викликати побічні ефекти;
- Незвичні методи організації коду: використання незвичних підходів до організації коду, які відрізняються від стандартних практик, може ускладнити розуміння коду іншими розробниками;

Code smell — це індикатор того, що код може бути оптимізований або рефакторингований для покращення його якості, продуктивності та підтримки. Виявлення та виправлення code smell може допомогти зробити код більш чистим, надійним та легшим для підтримки [29].

2.2.4 UML діаграма програмної частини системи

Уніфікована мова моделювання (UML, англ. Unified Modeling Language) є стандартною мовою для візуалізації, специфікації, конструювання та документування артефактів програмних систем. UML діаграми дозволяють моделювати як структуру, так і поведінку програмного забезпечення, систем та бізнес-процесів. Вони допомагають розробникам, аналітикам та іншим учасникам проекту зрозуміти та обговорювати структуру системи або процесу.

Діаграма послідовностей (Sequence Diagram) є одним з типів поведінкових UML діаграм. Вона використовується для моделювання взаємодій між об'єктами у системі контексту виконання конкретного процесу або функції. Діаграма послідовностей ілюструє, як об'єкти обмінюються повідомленнями у визначеній послідовності часу [30].

					КНУ.РБ.123.24.06.02.ПДСДР	Арк.
Арк.	№ документа	Підпис	Дата			

Порядок дій роботи системи залежить від дій користувача(рис. 2.21). Якщо це перший запуск, то треба налаштувати горщик. Для цього треба зайти у додаток та увімкнути Bluetooth на мобільному пристрої. Після цього із списку доступних BT підключення треба вибрати сам горщик та додати його до системи. Після цього він з'явиться у списку на головній сторінці додатку з базовими налаштуваннями.

За бажанням користувач може змінити налаштування системи обравши його у списку та змінивши показники, чи просто зчитати показники з датчиків.

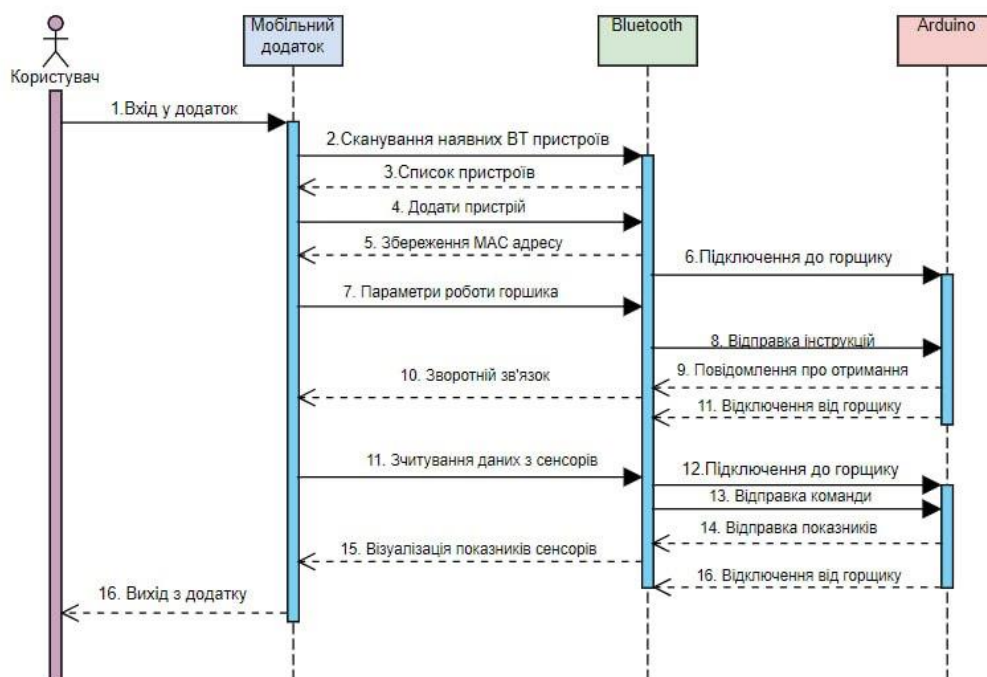


Рисунок 2.20 – Діаграма послідовностей системи

Висновки

З метою реалізації виконавчого модулю системи догляду за кімнатними рослинами зроблено аналіз ринку доступного обладнання та вивчення їх цін за основними характеристиками. Визначені основні компоненти для створення базової моделі з подальшим проведенням тестування. Для економічного обґрунтування доцільності проекту прорахована собівартість основних компонентів моделі системи автоматичного догляду за кімнатними рослинами. Створено алгоритм роботи виконавчої частини системи. Визначена електрична схема модулю з поетапною збіркою.

Програмування керуючого елементу системи здійснено за допомогою використання технології з'єднання Bluetooth з використанням МК Arduino. Створено мобільний додаток для керування системою в середовищі програмування EXPO. Зображено UML діаграму програмної частини, де визначено принципи роботи системи.

3 РЕЗУЛЬТАТИ ТЕСТУВАННЯ МОДЕЛІ ТА ПРАВИЛА ЕКСПЛУАТАЦІЇ СИСТЕМИ

3.1 Тестування схеми на Arduino

Тестування схеми на Arduino є важливим етапом розробки будь-якого проекту, що включає апаратні та програмні компоненти. Цей процес дозволяє виявити та усунути помилки, забезпечити правильну роботу схеми та програмного коду, а також гарантувати, що система відповідає заданим вимогам.

Перед початком тестування необхідно переконатися, що всі компоненти підключені відповідно до схеми. Слід перевірити з'єднання за допомогою схеми проекту або діаграми з'єднань. Використання макетної плати полегшує швидке прототипування і внесення змін.

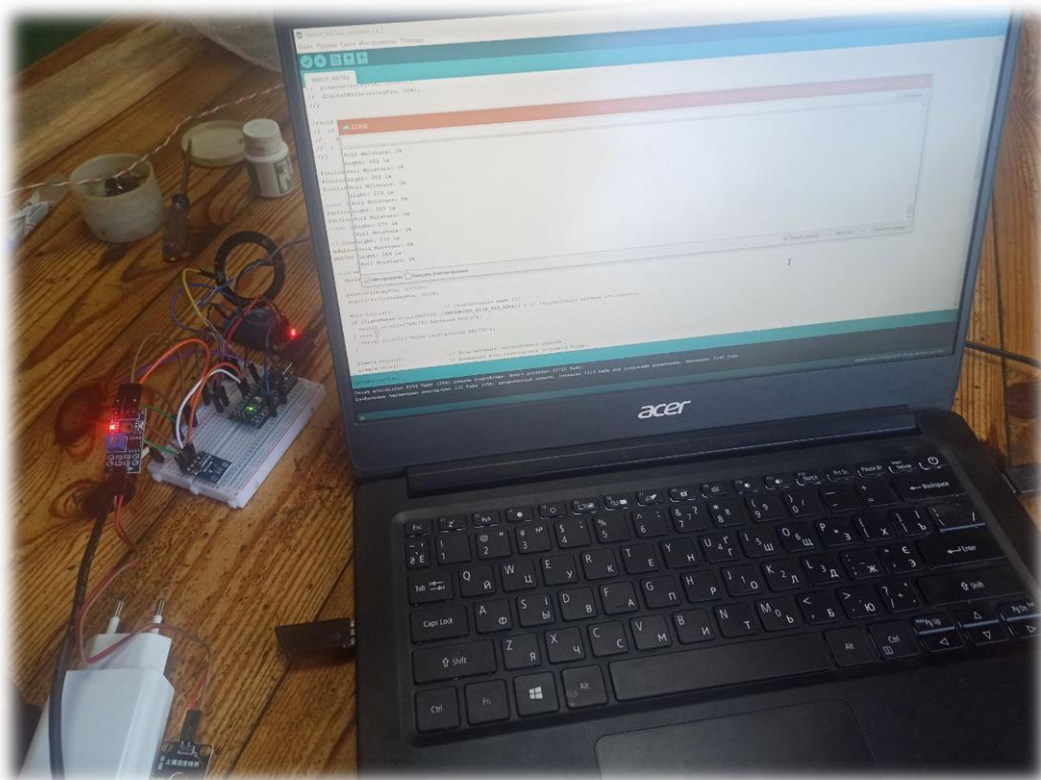


Рисунок 3.1– Використання серійного монітора Arduino IDE

Правильна робота коду визначається шляхом використання серійного монітора Arduino IDE(рис. 3.1) для виводу даних та налагодження. Додавання команд Serial.print() у ключові точки коду для відстеження значень змінних і поведінки програми.

					КНУ.РБ.123.24.06.02.РТМПЕС					
Змн.	Арк.	№ документа	Підпис	Дата	РЕЗУЛЬТАТИ ТЕСТУВАННЯ МОДЕЛІ ТА ПРАВИЛА ЕКСПЛУАТАЦІЇ СИСТЕМИ					
Розробив		Духов						Літера	Аркуш	Аркушів
Перевірив		Кузнецов								
Н.контроль		Кузнецов						КІ-20		
Затвердив		Купін								

Перевірка всіх з'єднань між компонентами є важливою для виявлення можливих помилок або непередбачуваної поведінки. Щоб уникнути проблем з підключенням треба зробити міцне підключення всіх провідників та компонентів.

Також необхідно переконатися, що Arduino і всі підключені компоненти отримують правильну напругу живлення. Неправильна напруга може призвести до нестабільної роботи або навіть пошкодження компонентів. Для цього можна використовувати мультиметр для перевірки напруги та опору в різних точках схеми.

Запуск проєкту в умовах, максимально наближених до реальних, для перевірки стабільності його роботи. Тестування схеми в умовах, в яких вона буде експлуатуватися. Запуск тривалих тестів для впевненості в стабільній роботі системи протягом тривалого часу.

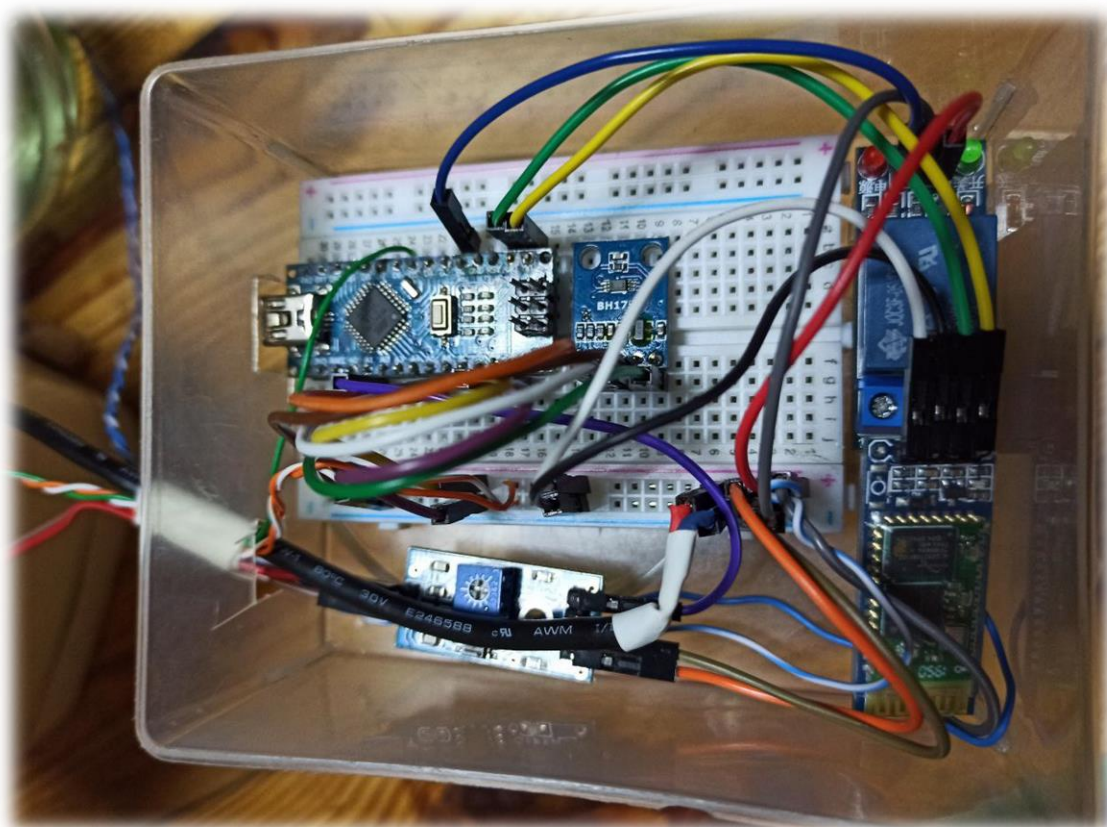


Рисунок 3.2 – Кабель-менеджмент на макетній платі

Окрім технічних аспектів, важливим є також кабель-менеджмент (рис. 3.2). Добре організовані кабелі допомагають уникнути плутанини, зменшують ризик з'єднання неправильно та полегшують налагодження та технічне обслуговування. Рекомендується використовувати маркування кабелів та тримачі для кабелів, щоб підтримувати порядок. Це особливо важливо для складних схем, де велика кількість проводів може заплутатися. Використання спеціальних трубок або стрічок для збирання проводів у пучки допомагає підтримувати чистоту та організованість робочого простору, що полегшує тестування та діагностику проблем.

Кабель-менеджмент є важливою частиною розробки проектів на Arduino, яка значно впливає на зручність роботи, безпеку та естетичний вигляд кінцевого продукту. Грамотне управління кабелями допомагає уникнути помилок у з'єднаннях, зменшити електромагнітні завади та забезпечити легкий доступ до компонентів для тестування та обслуговування.

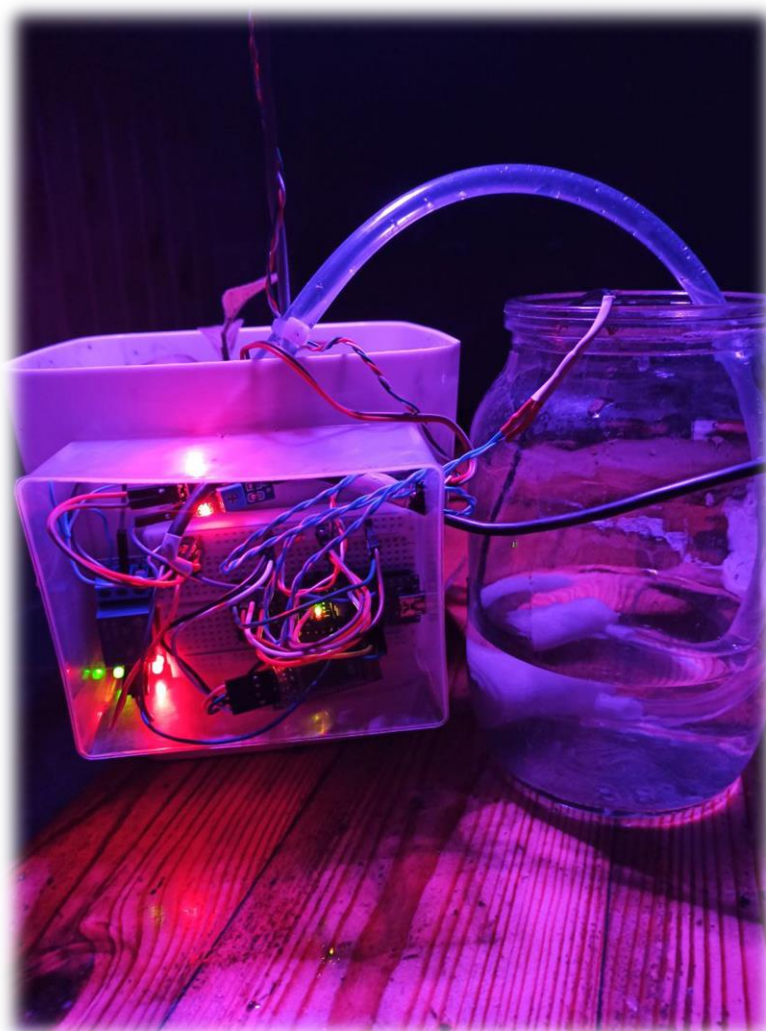


Рисунок 3.3 – Місце розташування модулю

Вибір місця розташування(рис. 3.3) Arduino є важливим аспектом при створенні електронного проекту. Від правильного розташування може залежати як зручність роботи з платою, так і стабільність її функціонування.

Arduino має бути розташоване в місці, захищеному від пилу, вологи та інших агресивних середовищ. Вологість може призвести до корозії контактів, а пил може викликати короткі замикання.

Плата повинна мати достатню вентиляцію для уникнення перегріву, особливо якщо проект включає компоненти, які генерують значну кількість тепла. Навколо плати потрібно достатньо простору для циркуляції повітря.

Плата Arduino повинна бути легкодоступною для налагодження, програмування та обслуговування. Це особливо важливо на етапі розробки, коли часто потрібно вносити зміни у підключення або програмний код.

Arduino слід закріпити на стабільній поверхні, щоб уникнути пошкоджень від випадкових ударів або переміщень. Можна використовувати монтажні отвори на платі для кріплення до корпусу або основи.

Уникнути розташування Arduino поблизу сильних електромагнітних полів або електричних приладів, які можуть викликати перешкоди в роботі плати. Такі пристрої можуть включати трансформатори, великі електродвигуни або потужні бездротові передавачі.

При розробці складних проєктів варто врахувати можливість швидкої заміни або оновлення окремих модулів. Це дозволяє легко вносити зміни у конструкцію без необхідності повного демонтажу.

Правильне розташування плати Arduino допоможе забезпечити стабільність і надійність проєкту, а також спростить процес його розробки та обслуговування. Урахування всіх зазначених факторів дозволяє уникнути багатьох потенційних проблем та оптимізувати робочий процес.

Калібрування датчиків(рис. 3.4) є важливим етапом у налаштуванні та точному вимірюванні даних у проєктах з Arduino. Воно дозволяє підвищити точність і надійність показників, що збираються датчиками, та забезпечити коректну роботу системи.

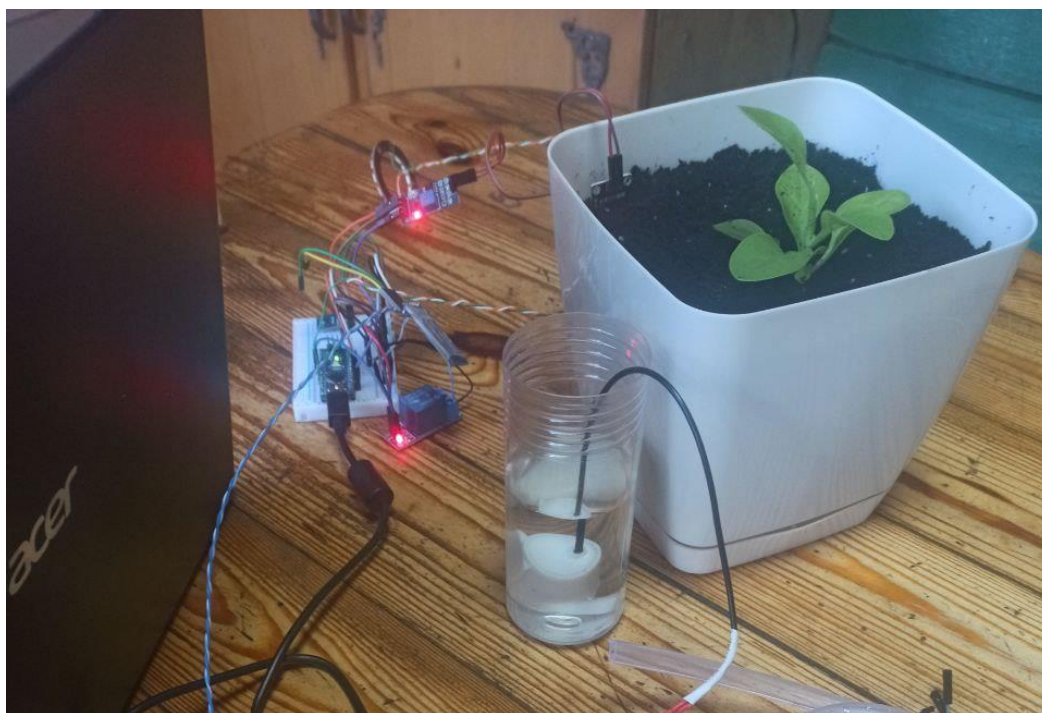


Рисунок 3.4 – Калібрування датчику вологості ґрунту

Калібрування датчиків — це процес налаштування і коригування показань датчика для забезпечення їх точності відповідно до еталонних значень або

стандартів. Це дозволяє мінімізувати похибки вимірювань та забезпечити узгодженість даних.

Перед початком калібрування треба переконатись, що всі компоненти правильно підключені, а датчики знаходяться в умовах, які відповідають їхньому призначенню. Важливо мінімізувати вплив сторонніх факторів, які можуть спотворити результати калібрування. Для точного калібрування необхідно знати еталонні значення, з якими будуть порівнюватися показання датчика. Це можуть бути значення, отримані з точного приладу або визначені теоретично.

Виходячи з різниці між вихідними даними датчика та еталонними значеннями, треба обчислити коефіцієнт поправки. Ці коефіцієнти допоможуть скоригувати показання датчика для досягнення максимальної точності.

Можна скорегувати програмний код на Arduino, додавши обчислені коефіцієнтів поправки. Це може бути реалізовано шляхом простого додавання або множення показань датчика на відповідний коефіцієнт. Після внесення змін у код, знову запусити скетч і перевірити нові показання датчика та порівняти їх з еталонними значеннями. Якщо похибка все ще є значною, треба повторити процес калібрування до досягнення прийняттого рівня точності.

Калібрування датчиків є необхідним процесом для забезпечення точності та надійності вимірювань у проектах на Arduino. Слід ретельно підходити до кожного кроку калібрування, починаючи з підготовки середовища і закінчуючи внесенням коефіцієнтів у код. Це дозволить отримати максимально точні дані та забезпечити стабільну роботу системи.

3.2 Тестування мобільного додатка

Тестування мобільного додатка є важливим етапом розробки, оскільки воно допомагає забезпечити якість та стабільність програми, а також її відповідність вимогам користувачів. Тестування включає різні види тестів(рис. 3.5), кожен з яких має свою мету та підхід.

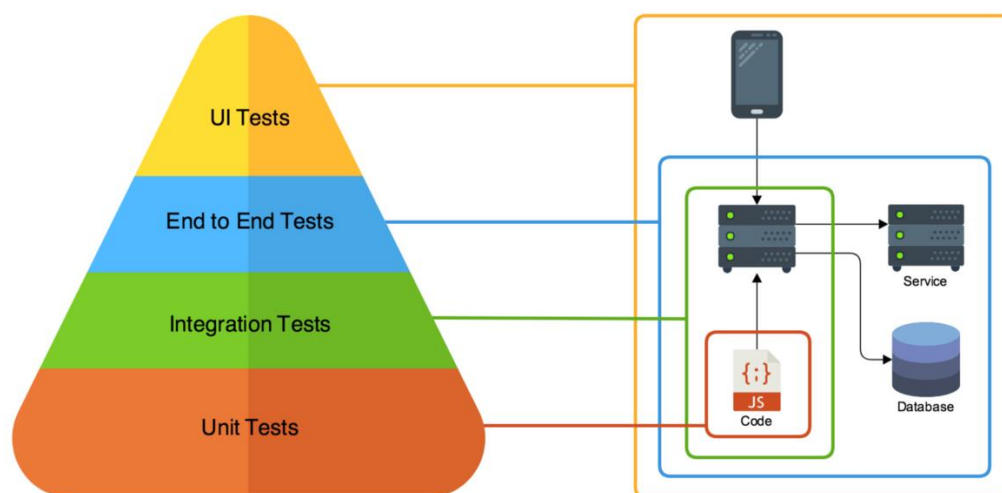


Рисунок 3.5 – Види тестів

Юніт-тести(unit tests) перевіряють окремі компоненти або функції програми. Вони зосереджуються на малих одиницях коду, таких як функції, методи, модулі тощо. Мета юніт-тестів - виявити помилки у логіці коду на ранніх стадіях розробки. Юніт-тести зазвичай швидкі у виконанні та забезпечують верифікацію індивідуальних частин програми. Розробники пишуть юніт-тести, щоб гарантувати, що окремі частини додатка працюють згідно з очікуваннями [31].

Інтеграційні тести перевіряють взаємодію між різними частинами програми. Вони зосереджуються на тому, щоб переконатися, що різні модулі працюють разом коректно. Ці тести часто перевіряють роботу додатка з зовнішніми службами, АРІ тощо.

Енд-ту-енд(end-to-end tests) тести перевіряють весь додаток від початку до кінця, імітуючи дії користувача. Вони охоплюють всі частини програми, включаючи інтерфейс користувача, бази даних, сервери та інші компоненти. Мета енд-ту-енд тестів - забезпечити, що додаток працює коректно, і користувач може виконати всі необхідні завдання. Вони часто є найдовшими та найскладнішими у виконанні, але дають загальний огляд працездатності програми.

Функціональні тести перевіряють виконання конкретних функцій програми відповідно до вимог. Вони перевіряють, чи працює програма так, як очікується, коли користувач виконує певні дії.

Регресійні тести перевіряють, що зміни в програмі не вплинули негативно на раніше працюючі частини. Ці тести забезпечують стабільність програми після внесення змін або виправлення помилок.

Тести продуктивності перевіряють продуктивність додатка, включаючи швидкість завантаження, швидкість обробки даних, продуктивність мережевих запитів тощо.

Тести безпеки перевіряють додаток на наявність вразливостей, які можуть призвести до несанкціонованого доступу або втрати даних.

Тести зручності використання перевіряють зручність використання додатка для користувача, включаючи легкість навігації, зрозумілість інтерфейсу та взаємодії з додатком.

Загалом, поєднання різних видів тестів забезпечує комплексну перевірку мобільного додатка, що дозволяє виявити та виправити проблеми на різних рівнях розробки та забезпечити якісний продукт для кінцевих користувачів.

Використання інструментів для контролю версій додатку може бути надзвичайно корисним для вирішення проблем з кодом та керування версіями вашого програмного продукту. Контроль версій - це система, що дозволяє відстежувати зміни у файлах та документах, щоб зберігати їхні різні версії у часі. Це надзвичайно корисний інструмент для розробників програмного забезпечення, письменників, дизайнерів та будь-кого, хто працює з файлами та потребує зберігання різних версій [32].

Основні концепції контролю версій:

- Історія змін. Кожна зміна в файлі або документі реєструється в системі контролю версій, зберігаючи повну історію змін;

- Гілки розробки. Розробники можуть створювати окремі гілки розробки, де вони можуть вносити зміни до файлів незалежно від основної гілки. Це дозволяє експериментувати та розвивати функціональність без впливу на основний код;
- Злиття змін. Після внесення змін у окремі гілки розробки, їх можна злити з основною гілкою, щоб впровадити їх у основний код;
- Відгалуження та маркування версій. Використання тегів або маркувань для позначення конкретних версій програми або документа.

У кожного коміта є свій унікальний ідентифікатор, який відомий як хеш або хеш-код. Хеш-код генерується за допомогою алгоритму хешування, який обробляє всі дані, що входять до коміта, включаючи зміст файлів, метадані коміта (такі як автор, дата, повідомлення) та інші параметри.

Цей унікальний хеш-код служить ідентифікатором для кожного коміта в системі контролю версій, такої як Git. Він дозволяє легко відстежувати, ідентифікувати та посилатися на певні коміти в історії репозиторію(рис. 3.6).

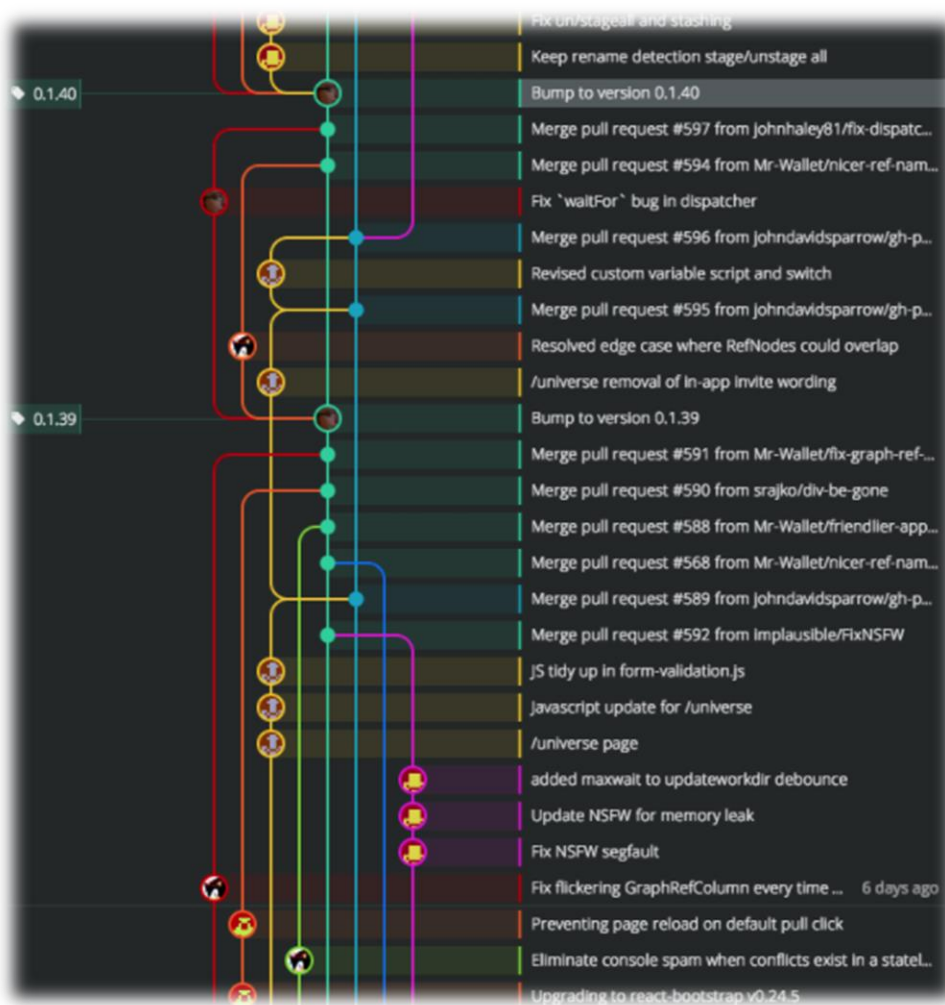


Рисунок 3.6 – Історія репозиторію

3.3 Інструкція з використання системи

Після ретельного аналізу та тестування зібраної системи можна використовувати її у повсякденному житті. Для виконавчого модулю на Arduino потрібно лише завантажити останню версію прошивки зі GitHub та прошити МК через Arduino IDE чи інший компілятор з подібним функціонуванням. При цьому не забувши завантажити необхідні бібліотеки через менеджер бібліотек у Arduino IDE (рис. 3.7) чи завантажити з інтернет джерел які використовуються у коді для функціонування елементів системи такі як: датчик освітленості та адресна світлодіодна матриця. Також на GitHub буде сторінка з командами для відправки по Bluetooth по serial terminal для налаштування системи без спеціального додатку.

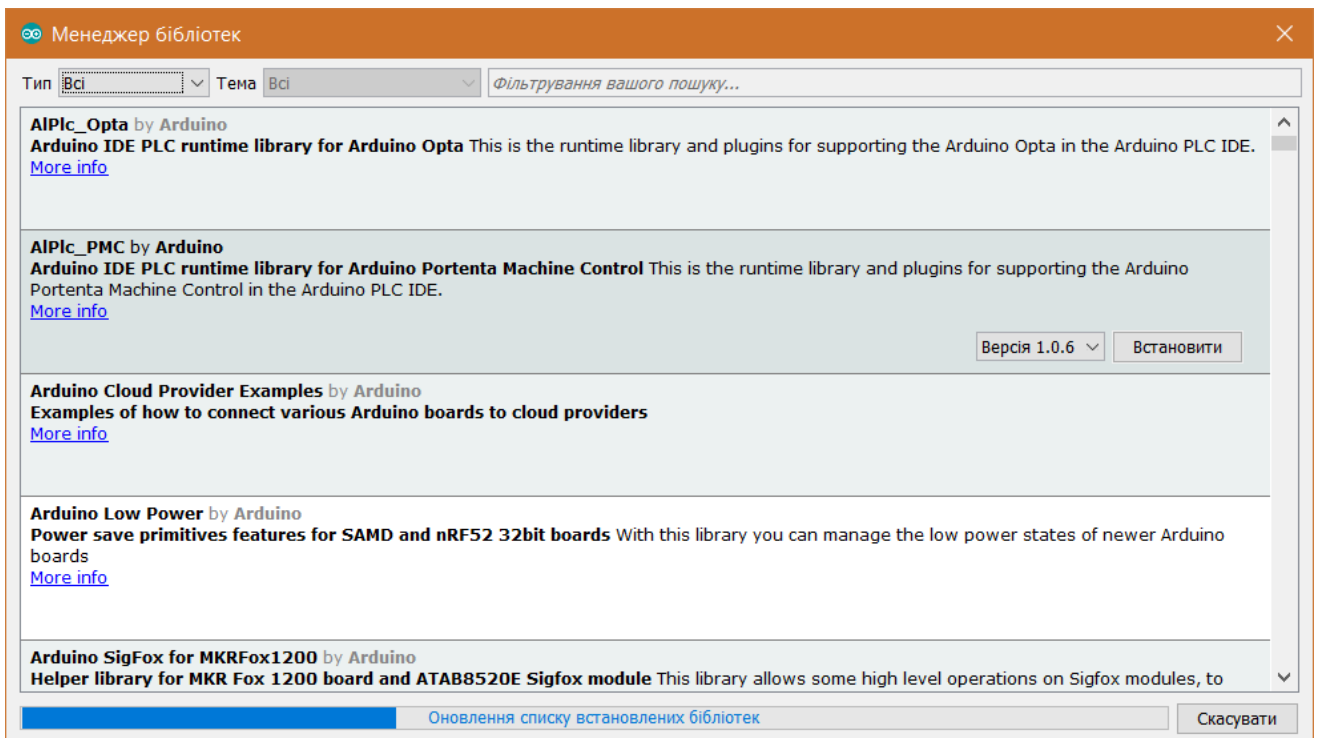


Рисунок 3.7 – Менеджер бібліотек Arduino IDE

Після того, як зібрали виконавчу частину системи та прошили її, треба перетворити код мобільного додатку у функціональну програму яку можна встановити на свій смартфон і через інтерфейс користувача керувати системою.

Нажаль, якщо у вас операційна не Mac OS, то Expo не дає можливості створювати збірки під продукцію Apple. Тому даний спосіб компіляції додатка підходить тільки для користувачів смартфонів на платформі Android. Для цього треба зробити клон коду за посиланням <https://github.com/DeduxOnline/SmartPot> та перейти в директорію проекту. Щоб створити програму, потрібно встановити EAS CLI виконавши команду у gitBash терміналі `npm install -g eas-cli`.

Щоб створити власну програму, потрібно створити обліковий запис Expo та увійти в EAS CLI. Після чого виконати команду у `eas login` терміналі, щоб увійти в EAS CLI та зробити файл конфігурацій EAS через команду `eas build:configure`.

Фінальний крок - це ввести команду `eas build --platform android --profile development`, щоб запустити збірку додатку у середовищі та серверах EAS. Але замість створення повноцінного додатку він створить Expo Go додаток який буде спеціально для тестування додатку. Тому потрібно зайти у файл `eas.json` і додати новий профіль для збірки додатку вказавши при цьому `buildType` як файл `apk`. Після цього ввести команду `eas build --platform android --profile preview`.

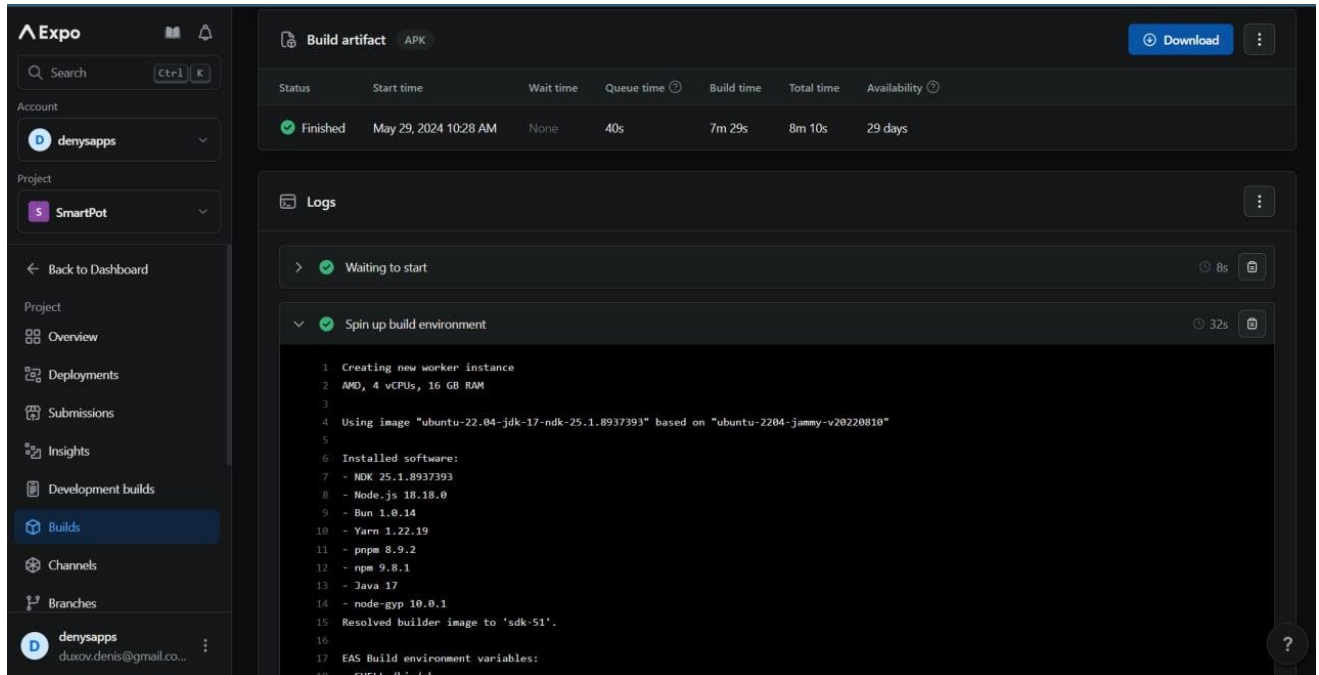


Рисунок 3.8 – Процес збірки мобільного додатку

Після завершення збірки додатку(рис. 3.8) можна буде зробити скан QR-код у своєму терміналі або відкрити посилання на своєму пристрої через браузер. На сайті буде файл для встановлення зробленого додатку. У нових смартфонах треба увімкнути дозвіл у браузері для встановлення додатків із джерел за Google Play Store. Даємо дозвіл на встановлення мобільного додатку та запускаємо файл установки. Після цих всіх кроків отримаємо повноцінний додаток із написаного коду на TypeScript з використанням React Native.

Для розміщення додатка на Google Play Store та Apple App Store необхідно сплатити відповідні збори. Для розміщення додатків на Google Play Store, потрібно створити обліковий запис розробника Google Play. Це вимагає одноразової плати за реєстрацію у розмірі 25 доларів США. Для розміщення додатків на Apple App Store, потрібно створити обліковий запис розробника Apple Developer. Це вимагає щорічної плати за підписку у розмірі 99 доларів США.

Отже, після створення всього необхідного для реалізації системи контролю догляду за рослинами які в неї плюси та мінуси. Почнемо з плюсів.

- Низька ціна: реалізація всієї системи коштувала всього 800 гривень що дешевше у декілька разів ніж готові рішення на ринку з подібним функціонуванням. І це ще без врахування грошей на хостинг мобільного

додатку на різних платформах та заробітну платню розробнику додатку, що є автором цієї роботи.

- Модульність системи: дана система реалізована так, що її можна використовувати з горщиками та цистернами різного розміру та об'єму, а готові збірки з таким функціоналом зазвичай прив'язані до свого горщика що є їх частиною або з використанням спеціальних картиджів рослин, що не дає можливість використовувати свій ґрунт та рослину.
- Створення сценаріїв роботи: завдяки використанню МК Arduino та датчиків освітленості і вологості землі можна реалізувати різну поведінку системи на різні показники сенсорів. Наприклад, при зниженні вологості землі у горщику до певних значень, запускати екстрений полив рослини, чи, при рівні освітленості нижче денних, вмикати додаткове освітлення на пару годин.
- Датчики освітленості: більшість готових систем з датчиками мають лише датчик вологості, а, замість датчика освітленості, рекомендують використовувати мобільний додаток через камеру смартфона, що не дає таких чітких показників.
- Легкість ремонту системи та можливість змінювати схему під себе: завдяки модульності та використанню макетної плати для підключення елементів система легко піддається ремонту та заміні окремих компонентів. Також можна взяти більш потужний блок живлення для виконавчих елементів системи та більш потужні насоси чи світлодіодні стрічки, що може вистачити на домашню теплицю. Також можна додати RTC модуль та додати інші види датчиків.

Також у даної системи є деякі мінуси.

- Відсутність елементу для аналізу якості ґрунту: на жаль не вийшло знайти датчик для МК, щоб робити аналіз ґрунту у горщику на якість, як це є у деяких готових рішень.
- Робота від розетки: дана система працює від блоку живлення через використання великої кількості елементів споживання тому, у випадку відсутності світла, система не буде працювати. Але за бажанням можна зробити автоматичний полив на МК ATtiny85, що відома своїм низьким споживанням та може довго працювати від декількох АА елементів, за принципом реле з таймером, що буде запускати насос.

Висновки

Здійснено тестування схеми на Arduino на виявлення та усунення помилок, забезпечення правильної роботи схеми та програмного коду. Встановлено, що система відповідає заданим вимогам.

Тестування мобільного додатка відповідає якості та стабільності програми.

Визначено плюси та мінуси розробки та функціонування даної системи контролю за кімнатними рослинами. Після ретельного аналізу та тестування модулю можна його використовувати у повсякденному житті.

					КНУ.РБ.123.24.06.02. РТМПЕС	Арк.
Арк.	№ документа	Підпис	Дата			

ВИСНОВКИ

Аналіз готових рішень наявних систем догляду за кімнатними рослинами: розумний вазон, гідропонна установка та краплинний полив для кімнатних рослин, а також відомих мобільних додатків: Plant Care Reminder, Flora Incognita та PlantIn: Визначник Рослин, свідчить про можливість постійного удосконалення та спрощення їх використання у сучасних умовах. Різна цінова політика, певні габарити та відсутність дистанційного керування. Постійна автоматизація умов життєдіяльності вимагає більш мобільні системи з можливістю підв'язати під смарт будинок.

Під час дослідження було розроблено основну концепцію власної моделі розумного горщика з дистанційним керуванням на прикладі порівняння готових на ринку рішень.

З метою реалізації виконавчого модулю системи догляду за кімнатними рослинами зроблено аналіз ринку доступного обладнання та вивчення їх цін за основними характеристиками. Визначені основні компоненти для створення базової моделі з подальшим проведенням тестування. Для економічного обґрунтування доцільності проекту прорахована собівартість основних компонентів моделі системи автоматичного догляду за кімнатними рослинами. Створено алгоритм роботи виконавчої частини системи. Визначена електрична схема модулю з поетапною збіркою.

Програмування керуючого елемента системи здійснено за допомогою використання технології з'єднання Bluetooth з використанням МК Arduino. Створено мобільний додаток для керування системою в середовищі програмування EXPO. Зображено UML діаграму програмної частини, де визначено принципи роботи системи.

Здійснено тестування схеми на Arduino на виявлення та усунення помилок, забезпечення правильної роботи схеми та програмного коду. Встановлено, що система відповідає заданим вимогам.

Тестування мобільного додатка відповідає якості та стабільності програми.

Визначено плюси та мінуси розробки та функціонування даної системи контролю за кімнатними рослинами. Після ретельного аналізу та тестування модулю можна його використовувати у повсякденному житті.

Даний проєкт створено з метою подальшого удосконалення системи контролю догляду за рослинами у сфері інтернет речей та має відкритий код викладений на платформі GitHUB для залучення однодумців та бажаючих реалізувати дану систему у себе.

					КНУ.РБ.123.24.06.В			
Змн.	Арк.	№ документа	Підпис	Дата				
Розробив		Духов			ВИСНОВКИ	Літера	Аркуш	Аркушів
Перевірив		Кузнецов						
Н.контроль		Кузнецов				KI-20		
Затвердив		Купін						

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Розумний вазон Lechuza Maxi Cubi Premium 14 Білий (18050). *Rozetka*. URL: https://rozetka.com.ua/ua/lechuza_maxi_cubi_18050/p638980/ (дата звернення: 29.05.2024).
2. Гідропонна установка Click and Grow 8875 Smart Garden 9 Бежевий. *Rozetka*. URL: <https://rozetka.com.ua/ua/285682333/p285682333/> (дата звернення: 29.05.2024).
3. Крапельний полив для кімнатних рослин Wi Drop M12. Автоматичне крапельне зрошення на 12 рослин. *Rozetka*. URL: https://rozetka.com.ua/ua/347976288/p347976288/?gclid=CjwKCAjwhNWZBhB_EiwAPzlhNucvWFgd3EzSB5asfPs7_eFpLH7CeKHZzZKiBM7_l8eVT2DUx_aMtqhoCjbEQAvD_BwE (дата звернення: 29.05.2024).
4. Turbin A. Plant care reminder - apps on google play. *Android Apps on Google Play*. URL: <https://play.google.com/store/apps/details?id=com.atcorapps.plantcarereminder> (дата звернення: 29.05.2024).
5. Plmenau T. U. Flora incognita - apps on google play. *Android Apps on Google Play*. URL: https://play.google.com/store/apps/details?id=com.floraincognita.app.flora_incognita (дата звернення: 29.05.2024).
6. PlantIn. PlantIn: plant identifier hero - apps on google play. *Android Apps on Google Play*. URL: <https://play.google.com/store/apps/details?id=com.myplantin.app> (дата звернення: 29.05.2024).
7. Занурювальний насос для води (мікронасос 3-6В) 70-120 Л/год. *Diy Shop*. URL: <https://diyshop.com.ua/ua/pogruzhnoj-nasos-dlya-vody-mikronasos-3-6v-70-120-l-ch> (дата звернення: 29.05.2024).
8. Трубка ПВХ 8 мм. Фермерський магазин Ukrferma: купити обладнання для птахівництва та тваринництва Київ, Україна. URL: https://ukrferma.com.ua/trubka-pvkh-8-mm/?gclid=Cj0KCQiAyMKbBhD1ARIsANs7rEEo0cE0m94mxuDhvW8GquB_HACKrSOtaKmlMv7b8RwXz_mo7Fc1GdvEaAqB8EALw_wcB (дата звернення: 29.05.2024).
9. Модуль реле 1 канал 5V для Arduino, Pic, ARM, esp8266 та інших мікроконтролерів купити з доставкою по Україні в магазині DIY Shop. *Diy Shop*. URL: <https://diyshop.com.ua/ua/modul-rele-1-kanal-5v-dlya-arduino-pic-arm> (дата звернення: 29.05.2024).

					КНУ.РБ.123.24.06.СВД					
Змн.	Арк.	№ документа	Підпис	Дата	СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ					
Розробив		Духов						Літера	Аркуш	Аркушів
Перевірив		Кузнєцов								
Н.контроль		Кузнєцов						КІ-20		
Затвердив		Купін								

10. Купити гігрометр, датчик вологості ґрунту, модуль Arduino з доставкою по Україні або у Кривому Розі. *Diy Shop*. URL: <https://diyshop.com.ua/ua/gigrometr-datchik-vlazhnosti-pochvy-modul-arduino> (дата звернення: 29.05.2024).
11. Модуль світлодіодний кільцевий WS2812B 5050 RGB LED 16 bit, веселка світлодіодів Neopixel led адресований pixel ring rainbow, купити в магазині DIY Shop у Кривому Розі, з доставкою по Україні. *Diy Shop*. URL: <https://diyshop.com.ua/ua/modul-svetodiodnyj-kolcevoj-WS2812B-5050-rgb-led-16-bit-raduga-svetodiodov-neopixel-led-adresuemyj-pixel-ring-rainbow> (дата звернення: 29.05.2024).
12. Купити датчик освітленості GY-302 на чіпі BH1750 у Кривому Розі. *Diy Shop*. URL: <https://diyshop.com.ua/ua/datchik-osveschennosti-gy-302-na-chipe-bh1750> (дата звернення: 29.05.2024).
13. HC-06 bluetooth модуль для arduino, купити у магазині DIY shop у кривому розі, з доставкою по Україні. *Diy Shop*. URL: <https://diyshop.com.ua/ua/hc-06-bluetooth-modul> (дата звернення: 29.05.2024).
14. Купити мікроконтролер Arduino Nano V3.0 ATmega328P CH340 Mini USB з HE розпаяними ніжками з доставкою по всій Україні. *Diy Shop*. URL: <https://diyshop.com.ua/ua/arduino-nano-v3-0-atmega328p-ch340-mini-usb-s-ne-raspayannymi-nozhkami> (дата звернення: 29.05.2024).
15. Що таке алгоритми: кроки, приклади, конструкції, мислення. *Онлайн навчання IT спеціальностям | IT курси онлайн | IT STEP*. URL: https://cloud.itstep.org/blog_3/building-and-understanding-algorithms-a-step-by-step-guide-for-beginners (дата звернення: 29.05.2024).
16. Control a water pump by Arduino. *Le Guide de la Robotique - Ouvrage réalisé par Med Ali*. URL: <https://www.robotique.tech/robotics/control-a-water-pump-by-arduino/> (дата звернення: 29.05.2024).
17. Guide for soil moisture sensor YL-69 or HL-69 with the arduino | random nerd tutorials. *Random Nerd Tutorials*. URL: <https://randomnerdtutorials.com/guide-for-soil-moisture-sensor-yl-69-or-hl-69-with-the-arduino/> (дата звернення: 29.05.2024).
18. Neo pixels ring with arduino nano. *Hackster.io*. URL: <https://www.hackster.io/pradeeplogu0/neo-pixels-ring-with-arduino-nano-d5d09c> (дата звернення: 29.05.2024).
19. Using GY-302 digital light intensity sensor module with arduino - phipps electronics. *Phipps Electronics*. URL: <https://www.phippselectronics.com/using-gy-302-digital-light-intensity-sensor-module-with-arduino/> (дата звернення: 29.05.2024).
20. Arduino and Bluetooth module HC-06 • AranaCorp. *AranaCorp*. URL: <https://www.aranacorp.com/en/arduino-and-bluetooth-module-hc-06/> (дата звернення: 29.05.2024).
21. Все що потрібно знати про технології Bluetooth - Motorvision. *Motorvision*. URL: <http://motorvision.com.ua/vse-shho-potribno-znaty-pro-tehnologiyi-bluetooth/> (дата звернення: 29.05.2024).

22. Arduino основи програмування. *geekmatic*.
URL: https://geekmatic.in.ua/ua/arduino_osnovyi_programmirovaniya (дата звернення: 29.05.2024).
23. Faq. *Expo Documentation*. URL: <https://docs.expo.dev/faq/> (дата звернення: 29.05.2024).
24. What is react native? Complex guide for 2024. *Digital Acceleration Company / Netguru*. URL: <https://www.netguru.com/glossary/react-native> (дата звернення: 29.05.2024).
25. Посібник: знайомство з react – react. *React – JavaScript-бібліотека для створення користувацьких інтерфейсів*.
URL: <https://uk.legacy.reactjs.org/tutorial/tutorial.html> (дата звернення: 29.05.2024).
26. Typescript: плюси та мінуси використання у розробці ПЗ. *FoxmindEd*.
URL: <https://foxminded.ua/typescript/> (дата звернення: 29.05.2024).
27. Що таке Node JS простими словами - Андрій Козюля. *Dan.IT*.
URL: <https://dan-it.com.ua/uk/blog/chto-jeto-takoe-node-js-prostymi-slovami/> (дата звернення: 29.05.2024).
28. Git bash: Definition, commands, & getting started | Atlassian. *Atlassian*.
URL: <https://www.atlassian.com/git/tutorials/git-bash> (дата звернення: 29.05.2024).
29. Запахи коду. *www.wikidata.uk-ua.nina.az*. URL: https://www.wikidata.uk-ua.nina.az/Запахи_коду.html (дата звернення: 29.05.2024).
30. Everything you need to know about sequence diagrams - Visual Paradigm Blog. *Visual Paradigm Blog*. URL: <https://blog.visual-paradigm.com/everything-you-need-to-know-about-sequence-diagrams/> (дата звернення: 29.05.2024).
31. The different types of testing in software | Atlassian. *Atlassian*.
URL: <https://www.atlassian.com/continuous-delivery/software-testing/types-of-software-testing> (дата звернення: 29.05.2024).
32. What is version control | Atlassian Git Tutorial. *Atlassian*.
URL: <https://www.atlassian.com/git/tutorials/what-is-version-control> (дата звернення: 29.05.2024).

Додаток А

Схема модулю системи контролю догляду за рослинами

