

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ  
ІМЕНІ ТАРАСА ШЕВЧЕНКА  
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ  
КАФЕДРА ІНФОРМАЦІЙНИХ СИСТЕМ ТА ТЕХНОЛОГІЙ**

**До захисту допущено**

**Завідувач кафедри ІСТ**

\_\_\_\_\_ Олександр КУЧАНСЬКИЙ  
(підпис) (ім'я, ПРІЗВИЩЕ)

“ \_\_\_ ” \_\_\_\_\_ 2022р.


**КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА**  
спеціальності 126 «Інформаційні системи та технології»  
освітньої програми «Програмні технології інтернет речей»  
на тему: « Розумний Інкубатор »

Виконав: студент 4 курсу, групи ІР-41

(шифр групи)

Микита ШЕСТОПАЛОВ \_\_\_\_\_

(Ім'я, ПРІЗВИЩЕ)

(підпис) 

Керівник кандидат технічних наук, доцент Ольга КРАВЧЕНКО 

(посада, науковий ступінь, вчене звання, Ім'я, ПРІЗВИЩЕ)

(підпис)

Консультант нормо контроль к.т.н., доцент Ростислав ЛІСНЕВСЬКИЙ \_\_\_\_\_

(назва розділу)

(посада, вчене звання, науковий ступінь, Ім'я, ПРІЗВИЩЕ)

(підпис)

Рецензент доцент кафедри ІСТ ЧДТУ, к.т.н., Анаїт КАРАПЕТЯН 

(посада, науковий ступінь, вчене звання, науковий ступінь, Ім'я, ПРІЗВИЩЕ)

(підпис)

Засвідчую, що у пояснювальна записка не має запозичень з праць інших авторів без відповідних посилань.

Здобувач освіти \_\_\_\_\_

(підпис)

Київ – 2022 року

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ  
ІМЕНІ ТАРАСА ШЕВЧЕНКА**

**Факультет інформаційних технологій**

Кафедра Інформаційні системи та технології

Освітній рівень Бакалавр

Спеціальність 126 Інформаційні системи та технології

Освітня програма Програмні технології інтернет речей

**ЗАТВЕРДЖУЮ  
Завідувач кафедри ІСТ**

\_\_\_\_\_ Олександр \_\_\_\_\_ КУЧАНСЬКИЙ  
(підпис) (ім'я, ПРІЗВИЩЕ)

“ \_\_\_\_\_ ” \_\_\_\_\_ 2022р.

**ЗАВДАННЯ**

**НА ВИКОНАННЯ КВАЛІФІКАЦІЙНОЇ РОБОТИ БАКАЛАВРА**

Здобувач освіти: Микита ШЕСТОПАЛОВ

Група: IP-41

**1. Тема кваліфікаційної роботи бакалавра:** « Розробка IoT системи забезпечення життєдіяльності інкубатора ».

Затверджена протоколом засідання кафедри ІСТ №18/20 від 01.12.2021 року

**2. Строк подання студентом готової роботи** – «26» червня 2021 р.

**3. Вихідні дані до роботи:** дослідження в області управління та автоматизації розумного інкубатора. Програмні рішення для системи автоматизованого керування розумним інкубатором. Дані про режим роботи датчиків, температура, вологість, обсяг води, перехід на запасний енергоносії та збереження у базі даних.

**4. Зміст роботи:** робота складається з трьох розділів: аналіз сфери застосування розумного інкубатора, проектування схеми та бази даних для розумного інкубатора, програмна реалізація проекту розумний інкубатор.

Також робота включає висновок і два додатки.

5. **Перелік графічного матеріалу:** план схема проекту, структурні схеми системи, приклади аналогів елементів системи.

6. **Календарний план виконання роботи:**

<b>Етапи виконання кваліфікаційної роботи бакалавра</b>	<b>Термін виконання</b>	<b>Результат виконання</b>
1. Вибір тематики кваліфікаційної роботи бакалавра	до 01.12.2021	виконано
2. Наказ про затвердження тем кваліфікаційної роботи бакалавра та призначення керівників	01.12.2021	виконано
3. Розробка плану кваліфікаційної роботи бакалавра і його погодження з керівником	25.12.2021	виконано
4. Написання I розділу кваліфікаційної роботи	20.01.2022	виконано
5. Написання II розділу кваліфікаційної роботи	19.02.2022	виконано
6. Написання III розділу кваліфікаційної роботи	05.03.2022	виконано
7. Написання IV розділу кваліфікаційної роботи		виконано
8. Підготовка висновків і пропозицій	05.04.2022	виконано
9. Попередній захист кваліфікаційної роботи	20.04.2022	виконано
10. Перевірка на плагіат	до 15.06.2022	виконано
11. Нормоконтроль	до 17.06.22	виконано
12. Рецензування кваліфікаційної роботи бакалавра і представлення роботи на кафедру в друкованому вигляді	до 21.06.2022	виконано
13. Захист кваліфікаційної роботи бакалавра	23.06.2022	

Дата видачі завдання «\_\_» \_\_\_\_\_ 2022 р.

Керівник роботи: к. т. н., доцент Ольга КРАВЧЕНКО \_\_\_\_\_ (підпис)

Завдання прийняв до виконання:

Здобувач освіти на освітньому рівні «бакалавр» 4-го курсу групи ІР-41

Микита\_ШЕСТОПАЛОВ

(Власне Ім'я, ПРІЗВИЩЕ)

(підпис)

## **АНОТАЦІЯ**

КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ТАРАСА

ШЕВЧЕНКА

Факультет інформаційних технологій

Кафедра Інформаційних систем та технологій

Освітня програма «Програмні технології інтернет речей»

Кваліфікаційна робота бакалавра Микити ШЕСТОПАЛОВА

**Тема роботи:** « Розробка IoT системи забезпечення життєдіяльності інкубатора ».

**Мета кваліфікаційної роботи бакалавра** – Розробка IoT системи забезпечення життєдіяльності інкубатора

**Об'єкт дослідження** – інкубатор для птахів.

**Предмет дослідження** – процес проектування та програмної реалізації IoT- рішення забезпечення життєдіяльності інкубатора

### **Апробація результатів.**

Кваліфікаційна робота бакалавра складається зі змісту, вступу, основної частини, яка включає три розділи, висновків та списку використаних джерел. Всього 60 сторінок.

**КЛЮЧОВІ СЛОВА:** IoT, Wi-Fi, ПРОЕКТ, ІНКУБАТОР, ПЛАТФОРМА, АВТОМАТИЗОВАНА СИСТЕИМА, РОЗУМНІ ТЕХНОЛОГІЇ.

# NOTATION

TARAS SHEVCHENKO KYIV NATIONAL UNIVERSITY

Faculty of Information Technology

Department of Information Systems and Technologies

Educational program "Software technologies of the Internet of Things"

Qualification work of bachelor Yaroslav NIZOV

**R&D:** " Development of IoT system to ensure the viability of the incubator  
".

**The purpose of the bachelor's** Development of IoT system to ensure the  
viability of the incubator.

**The object of research** is bird incubator.

**The subject of research** is the creation of database for the results of the  
analysis of the elements of the system and the work of the incubator for the  
implementation of the software addendum.

**The bachelor's thesis** consists of a table of contents, introduction, main  
part, which includes three sections, conclusions and a list of sources used. Total  
pages 60.

**KEY WORDS:** IoT, Wi-Fi, PROJECT, INCUBATOR, PLATFORM,  
AUTOMATED SYSTEM, REASONABLE TECHNOLOGIES.

## ЗМІСТ

ВСТУП .....	8
1. АНАЛІЗ СФЕРИ ЗАСТОСУВАННЯ РОЗУМНОГО ІНКУБАТОРА.....	9
1.1 Розгляд існуючих моделей інкубаторів .....	10
1.2 IoT платформи та їх використання .....	13
1.3. Система обігріву та зволоження .....	18
1.4. Мікроконтролер.....	22
1.5 Система провітрювання та корпус .....	23
1.6. Постановка задачі майбутньої ІОТ системи.....	28
Висновок до розділу 1: .....	29
2. ПРОЕКТУВАННЯ СХЕМИ СИСТЕМИ ТА БАЗИ ДАНИХ ДЛЯ РОЗУМНОГО ІНКУБАТОРА .....	30
2.1. Процес роботи розумного інкубатора.....	30
2.2. Процес роботи інкубатора в мережі.....	33
2.3. Моделі для створення бази даних.....	36
Висновок до розділу 2: .....	37
3. ПРОГРАМНА РЕАЛІЗАЦІЯ ПРОЕКТУ РОЗУМНИЙ ІНКУБАТОР .....	39
3.1 Створення симуляції процесу роботи датчиків інкубатора .....	39
3.2 Візуалізація роботи інкубатора з базою даних та системою оповіщень .....	44
Висновок до розділу 3: .....	49
ВИСНОВОК.....	50
Додаток А.....	55
Додаток Б .....	58
Додаток В.....	67

## ВСТУП

На сьогоднішній день дуже популярним є вирощування птахів в інкубаторах, тому що вони можуть надати максимальну відсоткову складову виведення живих пташенят. Для того щоб мінімізувати втрати та детально врахувати всі складові, в процесі вилуплювання пташенят доцільно у нас час вдатися до розумних технологій. Одним з варіантів може бути створення IoT рішення для автоматичної роботи самого пристрою та створення мобільного додатку, що працює з інкубатором, оснащеним системою оповіщення.

Освоєння знань у сфері оптимізації роботи системи інкубатора надають можливість спроектувати схеми автоматизації процесу підтримання життєдіяльності в ізольованому середовищі. також вирішити питання взаємодії різних елементів системи між собою.

Метою кваліфікаційної роботи бакалавра є “Розробка IoT системи забезпечення життєдіяльності інкубатора”. Для досягнення мети необхідно виконати наступні задачі:

- створити діаграми для процесу роботи інкубатора;
- спроектувати схему структурного пристрою, на основі існуючих рішень та аналогів;
- спроектувати мобільний додаток та програмно його реалізувати як зручний користувацький інтерфейс для IoT рішення.

Серед обраних елементів для проекту треба зазначити розумну систему зволоження, та температури, коли яйце цього потребує.

Створення схеми є основним етапом початку проекту, адже завдяки їй можна детальніше зрозуміти які варіанти з представлених аналогів будуть найкраще підходити.

## 1. АНАЛІЗ СФЕРИ ЗАСТОСУВАННЯ РОЗУМНОГО ІНКУБАТОРА

З кожним роком все більше набирає популярності технологія ІОТ (Internet Of Things), і це не дивно. Адже основним фактором інтернету речей є побудова автоматизації процесу, а автоматизація у свою чергу означає час та зменшення його витрат.

Проте ІОТ це не тільки про автоматизацію. Ця технологія так само дає змогу поліпшити і спростити життя людини пов'язану із сільським господарством. Розведення птахів є ретельним заняттям і нерідко трапляється, що відсоткова складова доброго потомства становить хоча б 70%.

Цей проект спрямований саме на те, щоб привести втрати при народженні птиці до мінімуму, а також автоматизувати процес, щоб він був не тільки безпечним, а й менше залежав від постійного догляду людини.

Отже основними перевагами цього проекту є:

- *Економія часу.* Зменшення часу на догляд за рослинами надає людині більше часу займатися своїми речами, та не переживати що з рослиною щось трапиться.
- *Автоматизація та оптимізація процесів.* Коли усі процеси можуть робитися без доглядачів, це зменшує людську працю, а також при правильно налаштованій системі рослина виросте здоровою та якісною без усяких добавок.
- *Можливість масштабування.* Цей проект розробляється таким чином щоб його можна було трансформувати. Якщо наприклад людину все влаштовує, і вона хоче більше, тому що на всіх членів сім'ї не вистачає. Можна докупити таку ж саму модель та об'єднати в більшу.

Ключовою ідеєю проекту - це удосконалити всі недоліки існуючих інкубаторів (Рис. 1.1). І автоматизувати весь процес. В даному проекті розглядаються такі ідеї як: автоматичне доливання води, бесперебійна робота при відключенні світла, автоматичний контроль температури та вологості.



*Рисунок 1.1 - Візуальний приклад розумного інкубатора*

Всі ці умови дотримуються виключно для того, щоб розробити інкубатор не вимагає багаторазових зусиль людина, а також щоб виведення здорових пташенят прагнуло максимуму.

### **1.1 Розгляд існуючих моделей інкубаторів**

Інкубатори давно використовуються в сільському господарстві і не є новинкою у приватному використанні. Подібні за принципом роботи пристрої, відрізняються габаритами, кількістю пташенят і потужністю.

Перші автоматизовані птахоферми, можливості яких зводилися до підтримки певної температури та вологості, не відрізнялися надійністю та потребували постійного контролю.

Щоб зрозуміти, як працює автоматичний інкубатор, необхідно розглянути тонкощі та послідовність процесів, що відбуваються в ньому.

Для обігріву використовуються закриті або відкриті електронагрівальні прилади, які керуються за допомогою датчика та регулятора температури. Після того, як пташенята, що знаходяться в яйцях, самостійно стають джерелом тепла, починає працювати система охолодження. Зволоження навколишнього

середовища, яке досягається за допомогою випаровування води, не допускає пересушування яєць. Для забезпечення свіжого повітря передбачені вентилятори. Лотки змінюють положення за допомогою поворотних механізмів, періодичність яких контролюється таймером.

Інформація про всі процеси збирається за допомогою датчиків і концентрується на мікроконтролері, в якому відбувається корекція режимів з подальшою видачею команд на відповідні виконавчі пристрої.

“Крім вбудованого пульта, розумний інкубатор може контролюватись на відстані. Електроніка відкриває можливість поєднання птахоферми з різними системами дистанційного керування, які можуть бути провідними та бездротовими. В останньому випадку управління здійснюється за допомогою радіо- та ІЧ-пристроїв, а також за допомогою WIFI, GSM та інтернету. Виступаючи як пульти дистанційного керування гаджети дозволяють отримувати інформацію про стан яєць і керувати автоматичною «засідкою» на будь-якому видаленні”[8].

При виборі інкубатора необхідно звернути увагу на функціоналу складову. Хоча деякі додаткові можливості можуть підвищувати вартість пристрою, їх наявність спрощує процес виведення пташенят і дозволяє уникнути помилок. Приклад цього служить вбудований овоскоп, за допомогою якого визначаються яйця з дефектами.

Крім цього, для домашньої птахоферми необхідно передбачити резервне харчування. Акумулятори не допустять загибелі ембріонів у разі перебоїв із електрикою. Інкубатор створить всі необхідні умови для безпеченого розвитку пташенят після вилуплювання.

King Suro20 - корейський інкубатор (Рис. 1.2), призначений для виведення курей та інших видів птахів. При дотриманні всіх умов використання, відсоток продуктивності інкубатора може становити 100%.

Цей пристрій є унікальним завдяки своїм корпусом і високою якістю використовуваного для нього матеріалу. Інкубатор був розроблений з урахуванням всіх необхідних пропорцій і збереженням компактності, завдяки чому місткість даного інкубатора є максимальною для свого розміру і можна не переживати про процес перевертання яєць. .

Стосовно особливостей представленого інкубатора, то вагомим фактором є велике вікно для спостереження за процесом інкубації, автоматичний переверт яєць, відповідає за повну автономність у питанні підтримання температури та вологості всередині приладу, також має міцний корпус, який робить цей варіант найкращим з усіх існуючих аналогів для домашнього використання. Приклад даного інкубатора представлений у (рис.1.2).



*Рисунок 1.2 - Візуальний приклад розумного інкубатора King Suro20*

Проте ця модель не є абсолютно автономною так як має проблеми з подачею води в інкубатор, ця дія розрахована на допомогу людини і кілька разів за період вирощування людині доведеться міняти піддон з водою. Метою даної роботи є продумування всіх аспектів, за яких інкубатор зможе самостійно вирощувати птахів без втручання людини.

Для більш детального порівняння інших моделей інкубаторів та розроблюваного представлена таблиця порівнянь основних функцій інкубатора, для повної автономної роботи інкубатора, були додані автоматичний набір води, та автоматичний перехід на запасний акумулятор.

Таблиця 1.1 Таблиця порівняння характеристик аналогів з власною розробкою

Переваги	Інші інкубатори	Власна розробка
Система обігріву/охолодження	+	+
Система зволоження/осушення	+	+
Автоматична система набору води		+
Автоматичний перехід на запасний енергоносії		+

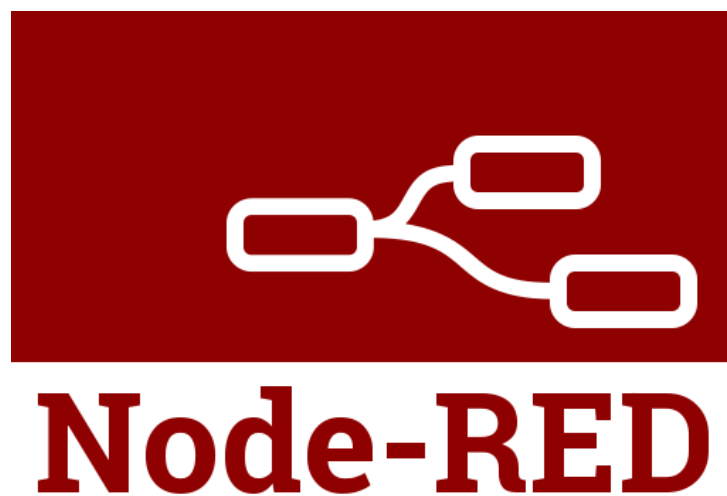
За результатами таблиці видно, що навіть сучасні розумні інкубатори є повністю ідеальними. У цій роботі були враховані всі можливі вади, такі як автоматичний набір води, а також перехід на запасний енергоносії у разі відсутності електроенергії.

## 1.2 IoT платформи та їх використання

Для відтворення цілого ресурсу за стеженням роботи інкубатора можна використовувати різні платформи або сервіси. Саме ІОТ-платформи є основною частиною проекту при автоматизації та оптимізації чогось, які використовуються все частіше в нинішньому світі.

Одним з прикладів таких технологій є Node-RED (Рис. 1.3). “Node-RED дає змогу працювати з браузерним редактором потоків даних як окремими вузлами з різним функціоналом (Рис. 1.4), що уможлиблюють створення JavaScript-

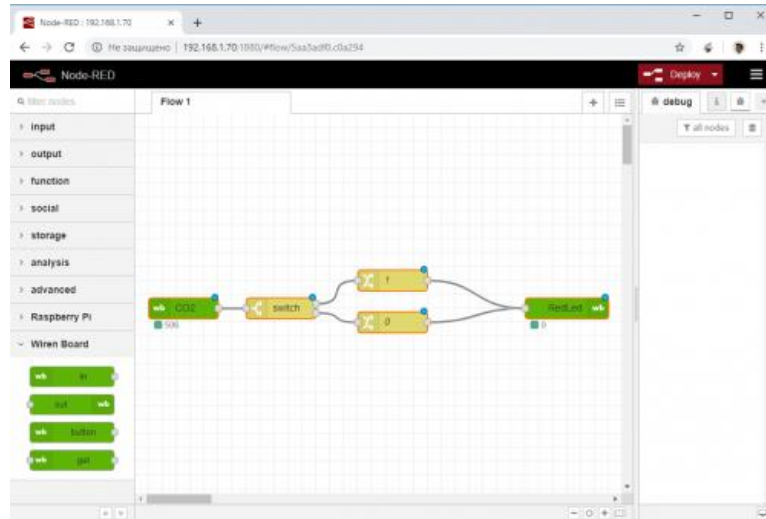
функцій. Причому, можна використовувати як базові вузли, якими одразу забезпечений Node-RED, так і встановлювати вузли з додатковим функціоналом з репозиторію NPM або ж навіть створити свій власний вузол з унікальним функціоналом. Програми або ж їхні частини, розроблені за допомогою Node-RED, можуть бути збережені та поширені для вільного використання. Саме середовище побудовано на основі Node.js. Потоки, створені за допомогою Node-RED, зберігаються у вигляді JSON. Починаючи з MQTT версії 0.14, вузли можна налаштовувати для TLS-з'єднання”[9].



*Рисунок 1.3 - Логотип Node-RED*

Програмування на основі потоку винайдене Дж. Полом Моррісоном у 1970-х роках програмування на основі потоків — “це спосіб опису поведінки програми як мережі чорних скриньок, або вузлів, як їх називають у Node-RED. Кожен вузол має чітко визначене призначення, йому надаються деякі дані, він щось робить з цими даними, а потім передає ці дані далі. Мережа відповідає за потік даних між вузлами”[9].

Це модель, яка дуже добре піддається візуальному відображенню та робить її більш доступною для більш широкого кола користувачів. Якщо хтось може розбити проблему на окремі кроки, він може подивитися на потік і зрозуміти, що він робить, без необхідності розуміти окремі рядки коду в кожному вузлі.



*Рисунок 1.4 - Приклад сервісу Node-RED*

Час виконання редактора Node-RED складається з середовища виконання на основі Node.js, на яке вказується веб-браузер, щоб отримати доступ до редактора потоків. У браузері потрібно створити свою програму, перетягуючи вузли з розділу з інструментами в робочу область і починати з'єднувати їх разом. Пізніше програма повертається до середовища виконання, де вона запускається.

Палітру вузлів можна легко розширити, встановивши нові вузли, створені спільнотою, а створені вами потоки можна легко поділитися як файли JSON.

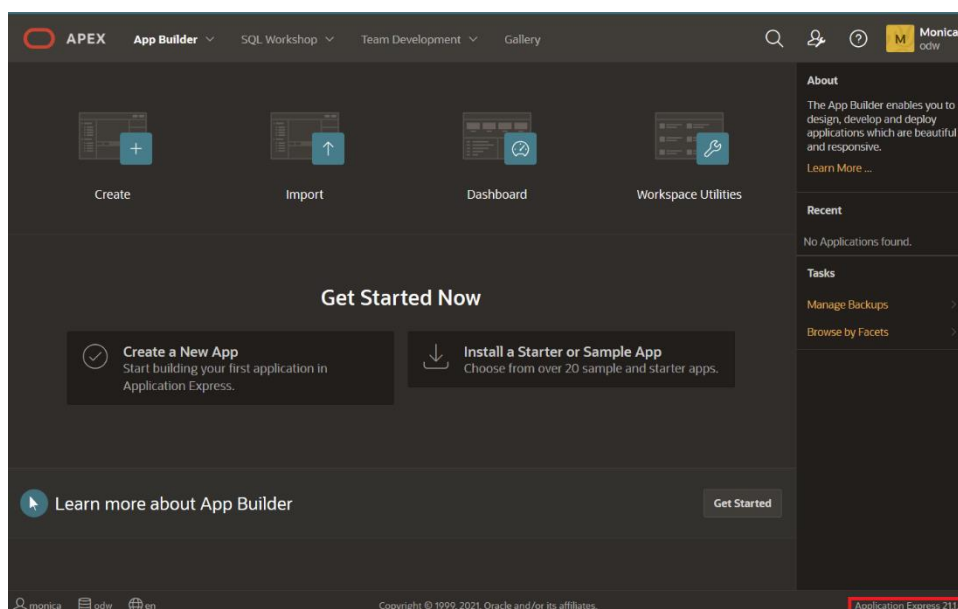
Так само ще одним хорошим аналогом є Apex Oracle.

Oracle Application Express – це платформа малокодової розробки (Рис. 1.5), що дозволяє створювати масштабовані, безпечні корпоративні веб-програми з великим функціоналом. Створені програми можуть бути розгорнуті на різних майданчиках.



*Рисунок 1.5 - Логотип Apex Oracle*

Програмний продукт Oracle APEX від компанії Oracle Corporation “при використанні виключно веб-браузера призначений для проектування (Рис. 1.6), розробки та розгортання функціональних, зручних, чуйних додатків, що взаємодіють із базою даних. Oracle Application Express є безкоштовним доповненням Oracle Database. За допомогою програмного забезпечення APEX можна створювати складні веб-програми, які можуть застосовуватися в більшості сучасних веб-браузерів”[10].



*Рисунок 1.6 - Приклад інтерфейсу Apex Oracle*

Серед основних можливостей програми Oracle APEX такі:

- Візуалізація даних. Платформа дозволяє швидко шукати і фільтрувати дані з фасетним пошуком, вбудовувати в додаток можливості створення налаштованих звітів користувача і використовувати при розробці великі інструменти побудови інтерактивних графіків.

- Перетворення електронних таблиць програми. Система дозволяє створювати робочі програми з електронних таблиць, що використовуються в бізнесі. На відміну від електронних таблиць APEX дає доступ до єдиного централізованого додатку, де всі зберігають і бачать завжди ті самі дані.

- Готові до використання програми. У поставку включені повністю функціональні та готові до використання компоненти, призначені для задоволення найпопулярніших вимог та покриття робочих процесів.

- Додатки, що розширюються. Готові програми, побудовані на Oracle Application Express, можуть бути розширені та додатково налаштовані у відповідність до індивідуальних потреб бізнесу.

- Швидке складання REST API. Платформа має зручний інструментарій створення веб-служб на основі об'єктів БД Oracle.

- Підвищена безпека. Платформа дозволяє вибрати спосіб захисту програм зі списку SSO, LDAP або авторизацію через соціальні мережі. APEX-програми можуть включати відстеження безпеки та аудит.

Також система Oracle APEX включає набір готових додатків, прикладів та інструментів розробки, таких як конструктор опитувань, інструмент відстеження помилок, засіб управління проектами.

Ці дві платформи є одними з найзручніших інструментів для роботи з базами даних. Вони мають велику кількість інструментів, що спрощує роботу і прискорює процес роботи.

### 1.3. Система обігріву та зволоження

Основним способом підтримання життєздатності яєць, і для їх повноцінного перетворення на пташенят, є підтримка протягом усього інкубаційного періоду, підтримання стабільної, потрібної температури.

Для обігріву використовуються закриті або відкриті електронагрівальні прилади, які керуються за допомогою датчика та регулятора температури. Після того як пташенята самостійно стають джерелом тепла, починає працювати система охолодження.

Для економії місця, відповідним цифровим датчиком температури та вологості Sonoff Sensor AM2301 (Рис. 1.7), вологість теж є невід'ємним і необхідним фактором правильного змісту. Зволоження навколишнього середовища, яке досягається за допомогою випаровування води, не допускає пересушування яєць.



*Рисунок 1.7 - Цифровий датчик температури та вологості Sonoff Sensor AM2301*

Цей датчик працює із Wi-Fi реле (вимикача) SonoffTH10. Дозволяє використовувати при керуванні вимикачем функції, що залежать від температури та вологості вимірюваних датчиком.

Серед інших аналогів, найбільш підходящим є вище представлений екземпляр, що підходить нам за трьома основними параметрами:

- Точність виміру: від 0,3 до 1. Точність значення дуже важлива для правильного догляду за яйцями
- Діапазон вимірювання вологості: 0-100%. Діапазон вимірювання температури: -40 ~ +80°C. В діапазоні є потрібне ідеальне значення, це 38°C і 60% вологості.
- Підключення до реле яке буде передавати інформацію в реальному часі та задіяти функції необхідні для підтримання життєздатності пташенят.

Реле Sonoff TH16 (Рис. 1.8) призначене для контролю температури та вологості в реальному часі через програму eWeLink. Завантажити програму eWeLink можна в магазині програм для iOS версії або Google play для Android версії. Інтелектуальне реле Sonoff TH16 керує електричними пристроями для підтримки температури та вологості у інкубаторі. Для здійснення своїх функцій до реле підключаються датчики температури та вологості. Залежно від завдань, можна підключити один із датчиків AM2301 або DS18B20. Дане реле має можливість сценарного управління, що є основним параметром при виборі пристрою. Так при певних змінах у вологості та температурі, з реле надходять дані на мікроконтролер для належних функцій. Також даному реле властива мінімальна похибка при використанні датчиків заявлених виробником, яким і є Sonoff Sensor AM2301.



*Рисунок 1.8 - Одноканальне Wi-Fi реле з датчиком температури та вологості Sonoff TH16*

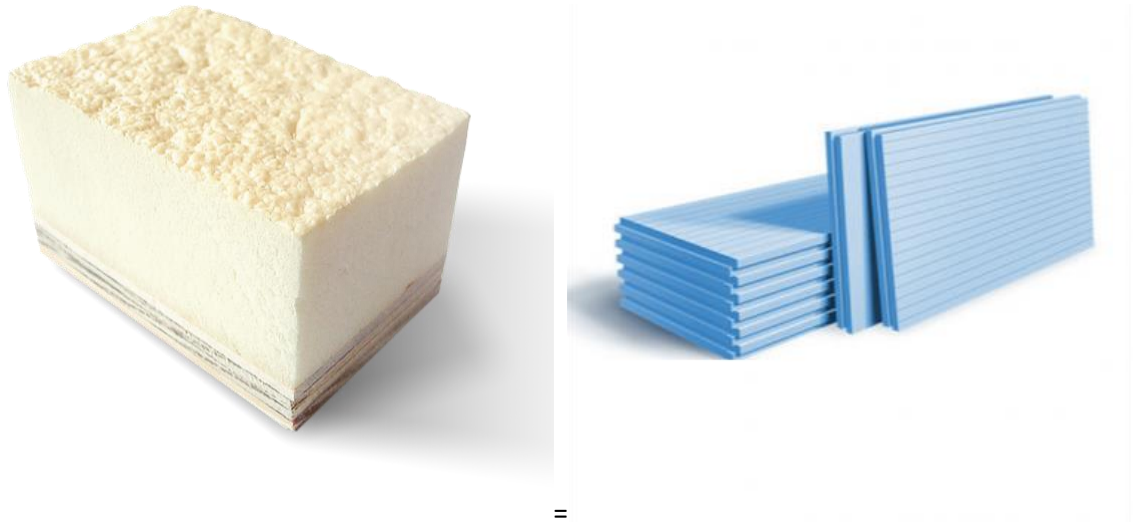
При виборі пристрою, яке буде нагрівати яйця для необхідної температури, довелося зіткнутися з великим об'ємом фактів на основі яких буде і вибрано потрібний пристрій.

Декілька проблем при виборі нагрівача:

- Матеріал з якого буде виготовлена конструкція
- Об'єм інкубатора
- Потужність пристрою

Отже, при виборі матеріалу, було виділено 2 основні матеріали для виготовлення інкубатора: пінополістирол і пінополіуретан.

Пінополіуретан, в порівнянні з такими популярними утеплювачами, як мінеральна вата і пінопласт, має найнижчий коефіцієнт теплопровідності - 0,025 Вт/мк. Теплопровідність екструзійного пінополістиролу становить 0,028 Вт/мк, теплопровідність пінопласту - 0,039 Вт/мк. Тому було обрано використовувати пінополіуретан (Рис. 1.9) для того, щоб уникнути перегріву надалі.



*Рис. 1.9 - Зліва приклад пінополіуретан, а з права приклад пінополістиролу*

Так як інкубатор невеликого розміру було прийнято рішення використовувати тепловий електронагрівальний шнур (Рис. 1.10). Вибір для даного нагрівача стало те, що він дуже гнучкий і можна розташувати по всьому інкубатору. Також провід можна протягнути поруч із контейнером з водою для підтримки вологості в інкубаторі. Шнуровий електронагрівник являє собою гнучкий електронагрівальний елемент у кабельній поліхлорвінілової оболонці з питомим навантаженням 7,5 Вт/м та температурою на поверхні не більше 50°C, потужністю 30 Вт.



*Рисунок 1.10 - Тепловой шнур*

Цей шнур буде підключений до одноканального Wi-Fi реле Sonoff TH16, яке і буде контролювати нагрівання даного дроту до потрібних температур. Усі параметри можна встановити в мобільному додатку, який підтримується на пристроях ОС таких як IOS і Android.

Завдяки даним пристроям та датчикам інкубатор зможе підтримувати температуру та вологість потрібну для вирощування здорових курчат із яєць.

#### **1.4. Мікроконтролер**

Основною деталлю інкубатора є ХМ-18SW (Рис. 1.11) це свого роду мозок інкубатора, який відповідає за всі процеси в інкубаторі, починаючи з нагрівання, закінчуючи частотою обороту вентиляторів. Цей мікроконтролер розроблений компанією JiLong, Китай, яка вже багато років удосконалює ринок своїми технологіями.

WIFI інтелектуальний інкубаційний контролер - це мікрокомп'ютерна система управління нового покоління для дистанційного керування та моніторингу мобільного телефону в режимі реального часу. Гуманізований повнокольоровий дизайн екрану гарний і елегантний з яскравими зображеннями. При цьому існує п'ять типів режимів інкубації: кастомний, курячий, качиний, гусак, голуб, і користувач може вибрати вільно, використовуючи нові мікроелектронні технології та нові компоненти. Мікрокомп'ютерний чіп, який має сильну антивтручальну здатність і високу стабільність роботи. Датчик температури використовує високоточні датчики збору температури. Для високої точності використовується високопродуктивний датчик вологості, який ефективно забезпечує стабільність та надійність виконання продукту.



*Рисунок 1.11 - Інтелектуальний інкубаційний контролер XM-18SW*

Користувач задає такі параметри, як еталонна температура, еталонна вологість, поворот яєць і вентиляція через мобільний клієнт. Параметри завантажуються на хмарний сервер, а сервер хмари відправляє керуючі параметри основному контролеру. Основний контролер регулює різні параметри інкубатора, потім повертає результати керування на сервер хмар, а хмарний сервер надсилає результати управління мобільному клієнту, і таким чином завершує дистанційне управління.

### **1.5 Система провітрювання та корпус**

Основним місцем для розміщення всіх датчиків і яєць стане корпус від інкубатора Nest 200 (Рис. 1.12). Він оснащений тим самим утепленим матеріалом, про який розповілося вище. Перевага цього корпусу в тому, що він вже оснащений окремим місцем для доливання води. У ньому є контейнер з водою і місце для насоса.



*Рисунок 1.12 - Корпус Nest200*

Підтримка води в інкубаторі завжди була величезною проблемою і саме це не дозволяло зробити абсолютно незалежний від людини інкубатор, але завдяки насосу та розумному датчику рівня рідини така проблема буде вирішена.

Для вимірювання рівня води буде вибрано рівнів Alisonic Delphi FX (flexible) (Рис. 1.13). Він передаватиме дані на мікроконтролер, як тільки вода в контейнері випарується до певного мінімуму, рівнемір передасть інформацію мікроконтролеру, а той у свою чергу дасть вказівку насосу набрати певний обсяг води.



*Рисунок 1.13 - Уровнемер Alisonic Delphi FX (flexible)*

У ролі насоса вибраний компресор aPUMP (Рис. 1.14). компресор aPUMP є на даний момент найменшим і найтихішим компресором у світі. Тиск повітряного потоку, що створюється компресором aPUMP, дозволяє використовувати його для ємностей висотою водного стовпа до 80см. Так само одне з переваг у низькому рівні шуму - менше 35дБ.



*Рисунок 1.14 - Компресор aPUMP*

Для провітрювання інкубатора буде використовуватися два 12Вт вентилятори, внутрішній (Рис. 1.15) та зовнішній (Рис. 1.16). Один з них вкачуватиме повітря всередину, інший видуватиме. Зроблено це для циркуляції повітря та зміни температури в інкубаторі.



*Рисунок 1.15 - Вентилятор на вдув Sunon (DC) HA40101V4-A99 12 В 40x40x10 мм*



*Рисунок 1.16 - Вентилятор на видув Sunon (DC) HAC0251S4-C99 12 В 120x120x25 мм*

Заключна частина пристроїв для повної роботи інкубатора потрібен блок живлення (Рис. 1.17). У цьому випадку до буде додано ще й акумулятор (Рис. 1.18) для безперебійної роботи системи, у разі відключення світла. Так як без акумулятора доведеться щоразу включати всю систему заново, що руйнує уявлення про повністю автономний інкубатор.



*Рисунок 1.17 - ADD-55A Mean Well Блок живлення з функцією UPS 52,58Вт*

Основною функцією UPS є забезпечення безперервності електроживлення за допомогою використання альтернативного джерела енергії. Крім того, UPS підвищує якість електроживлення, стабілізуючи його параметри у встановлених межах. В UPS як накопичувач енергії зазвичай використовуються хімічні джерела струму.



*Рисунок 1.18 - Акумулятор LOGICPOWER LP 12V 12AH (LP 12 - 12 AH)*

При несправності роботи вся мережа переходить на забезпечення живлення за рахунок акумулятора. Ця функція дуже важлива для забезпечення життєздатності пташеня.

## 1.6. Постановка задачі майбутньої ІОТ системи

Існує багато аналогів різних пристроїв, які необхідні для повноцінного вирощеного у штучних умовах пташенят. Ключовим фактором є те, що вся система має працювати автоматично без втручання людини. Низька температура? система повинна автоматично підвищити температура, потрібно збільшити вологість або осушити інкубатор? система повинна зібрати цю інформацію та передати її контролерам, ті у свою чергу повинні виконати необхідні функції.

Насамперед метою кваліфікаційної роботи бакалавра є розробка моделі ІоТ рішень для вирощування птахів в інкубаторі, що має задовольняти наступні вимоги:

- високоефективна система, яка буде підтримувати високий відсоток підтримки життєздатності птахів на протязі всього терміну знаходження в інкубаторі;
- система передачі даних з датчик користувачеві для контролю стану роботи інкубатора;
- модель інкубатора, яка буде оснащена всім необхідним для повноцінної роботи;
- пристосованість до експлуатації у відповідних кліматичних умовах;
- використання сучасних приладів для точного стеження за процесом вирощування;

Для досягнення кінцевої мети необхідно виконати наступні завдання:

- аналіз базових принципів роботи ІоТ систем;
- аналіз аналогів для побудови моделі потрібного інкубатора;
- аналіз процесу роботи інкубатора;
- описати базову діаграми процесу роботи інкубатора;
- описати діаграму передачі даних з апаратних засобів в програмний з відображенням мережі;
- розробити блок-схему роботи інкубатора;

- розробити симуляцію інкубатора;
- розробити програмний продукт та графічний інтерфейс для користувача;

Таким чином, сформовано початкові вимоги для розробки моделі системи IoT рішень для розумного інкубатора та завдання для кваліфікаційної роботи бакалавра.

**Висновок до розділу 1:** Розроблено принцип роботи розумного інкубатора. Як приклад використовувався корейський інкубатор King Suro20, що є одним з популярних розумних інкубаторів на даний момент. Розглянуті платформи використання програмної реалізації бази даних для ведення точних значень у процесі роботи, а також виявлення помилок. Обрані аналоги пристроїв, які будуть використовуватися в інкубаторі. Особливу увагу було приділено корпусу, який повинен мати певний поріг пропуску тепла, для точного вибору наступних аналогів обігріву та охолодження.

## 2. ПРОЕКТУВАННЯ СХЕМИ СИСТЕМИ ТА БАЗИ ДАНИХ ДЛЯ РОЗУМНОГО ІНКУБАТОРА

### 2.1. Процес роботи розумного інкубатора

Переглянувши всі аналоги, і укомплектовуючи з певних пристроїв повноцінний розумний інкубатора, настав час перейти до процесу роботи самої ідеї проекту.

Спочатку потрібно розглянути діаграму роботи розумного інкубатора (Рис. 2.1).

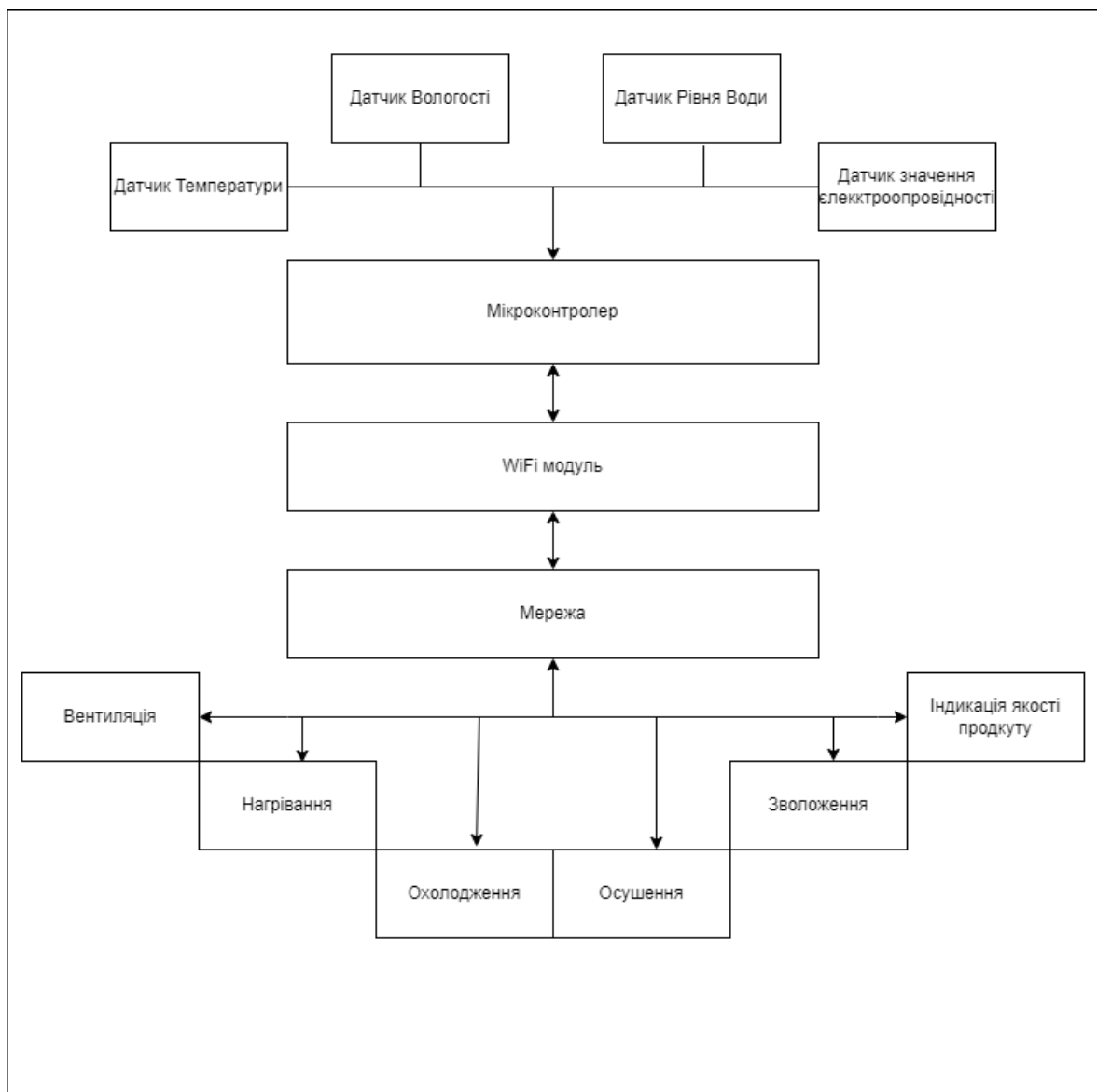


Рисунок 2.1 - Діаграма роботи розумного інкубатора

Інформація з датчик передається на мікроконтролер, той у свою чергу порівнює отриману інформацію з датчиків. Потім надсилає цю інформацію за допомогою WiFi модуля на сервер. Там відбувається перевірка значень, і надсилається відповідь із вказівками на мікроконтролер. Той роздає передані йому вказівки пристроїв, які вказані в даних.

Щоб розглянути процес передачі вказівок на пристрої, була розроблена блок схема (Рис. 2.2), з детальним описом роботи перевірки інкубатора.

Отже, спочатку йде перевірка на температуру, дані беруться з датчика і розглядаються. Якщо температура не в проміжку від 37.5-37.8°C, що є ідеальною температурою для вирощування курячих яєць, то йде розгляд, чи вище температура з датчика або нижче. Якщо температура вище, спрацьовує пристрій охолодження, в даному випадку це великий вентилятор, який охолоджує інкубатор до потрібної температури. Якщо ж температура нижче, спрацьовує пристрій нагрівання, це тепловий шнур. Після цих процесів йде перехід до перевірки вологості. Якщо температура була стабільна, то перевірка наступного датчика проходить відразу.

З датчиком вологості все приблизно схоже. Якщо Вологість вище вказаного у схемі проміжку значень йде процес перевірки вище дані значення чи нижче. Як наслідок і вибирається осушувати інкубатор або зволожувати. За потрібних показників перевірка не відбувається.

Далі йде індикатор рівня вологості. Тут відсутня як така умова. Якщо води в контейнері нижче за потрібне, відбувається команда наповнити контейнер, об'єм 2л. Вода нам потрібна для підтримки вологості в інкубаторі.

Фінальна та задоволена важлива частина роботи інкубатора, перевірка електроживлення інкубатора. Якщо інкубатор перестає отримувати енергію від мережі 220В, він сповіщає власника про відсутність світла, і переходить на резервне забезпечення з допомогою акумулятора.

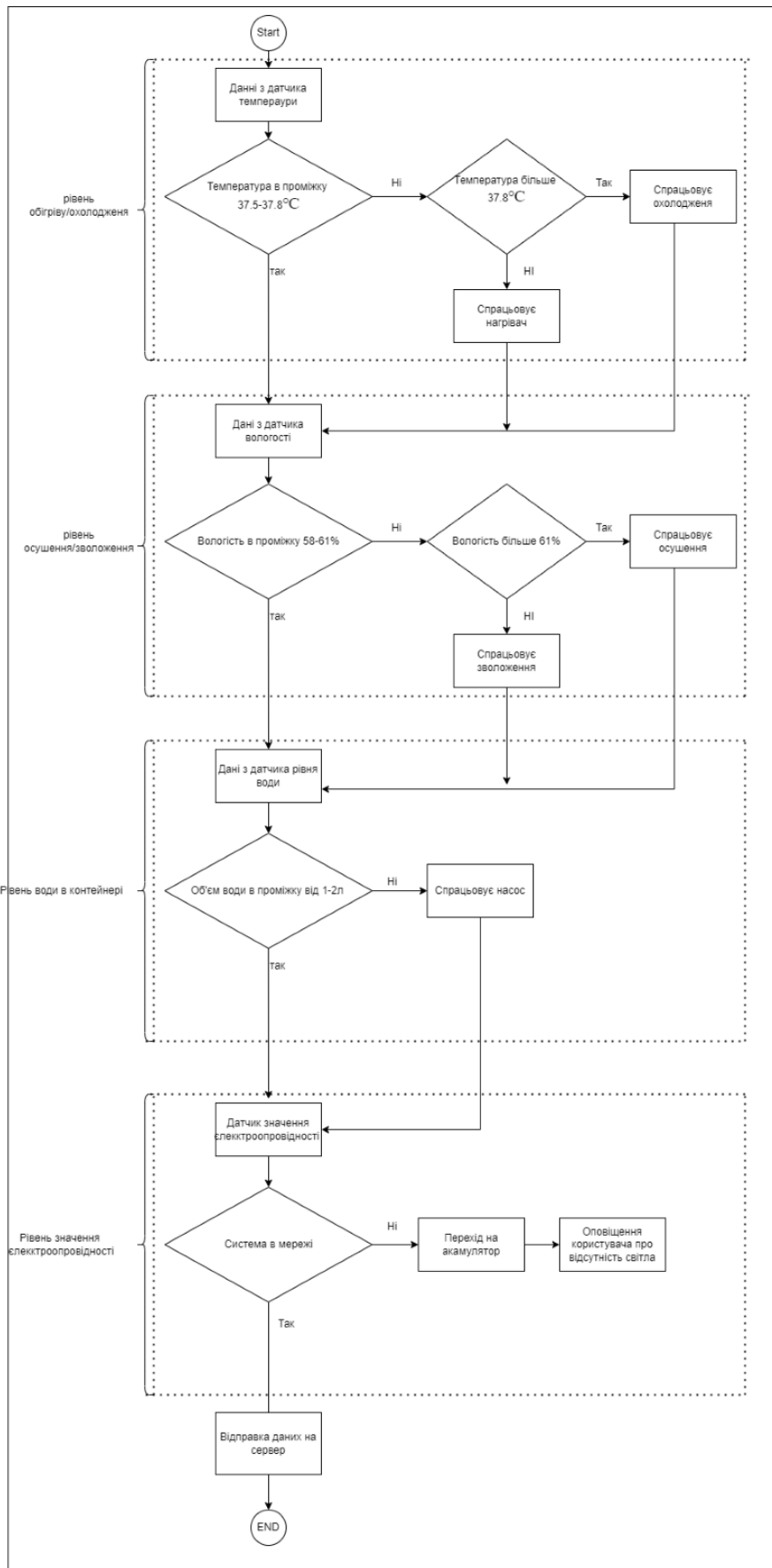
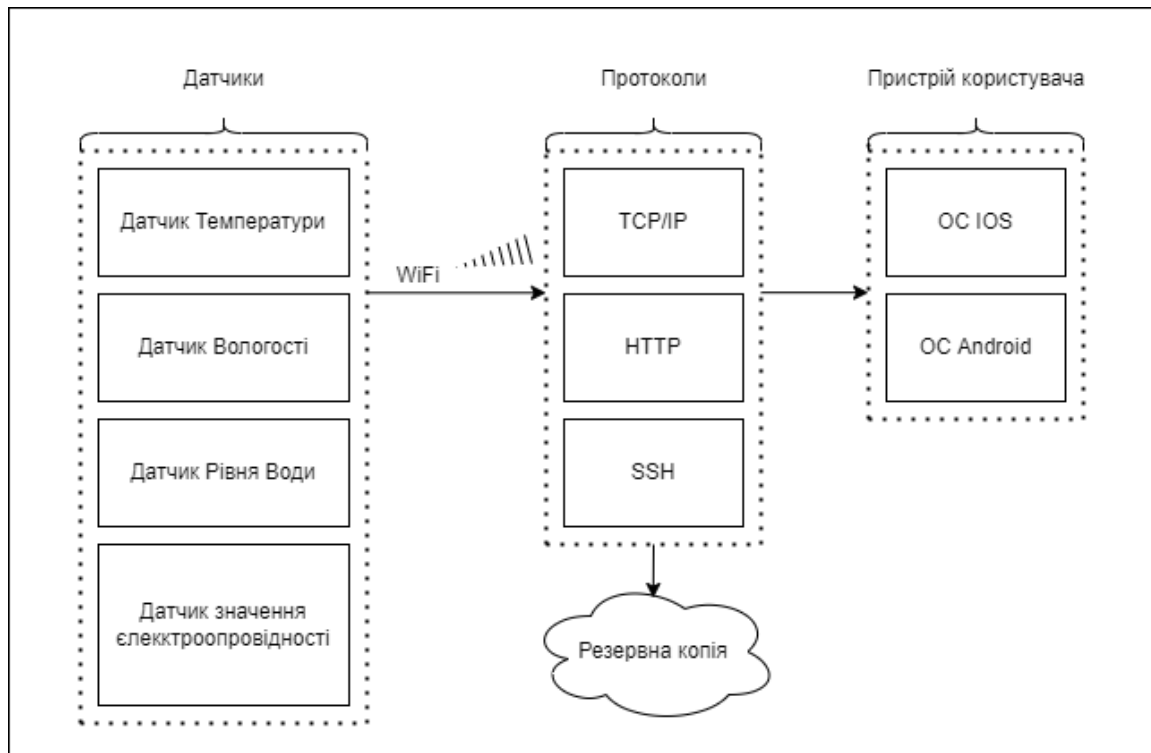


Рисунок 2.2 - Блок-схема роботи інкубатора

## 2.2. Процес роботи інкубатора в мережі

Важливою функцією розумного інкубатора є можливість постійного моніторингу із системою повідомлень користувачеві. Таким чином була розроблена діаграма, що показує процес отримання даних з датчиків користувачем (Рис. 2.3).

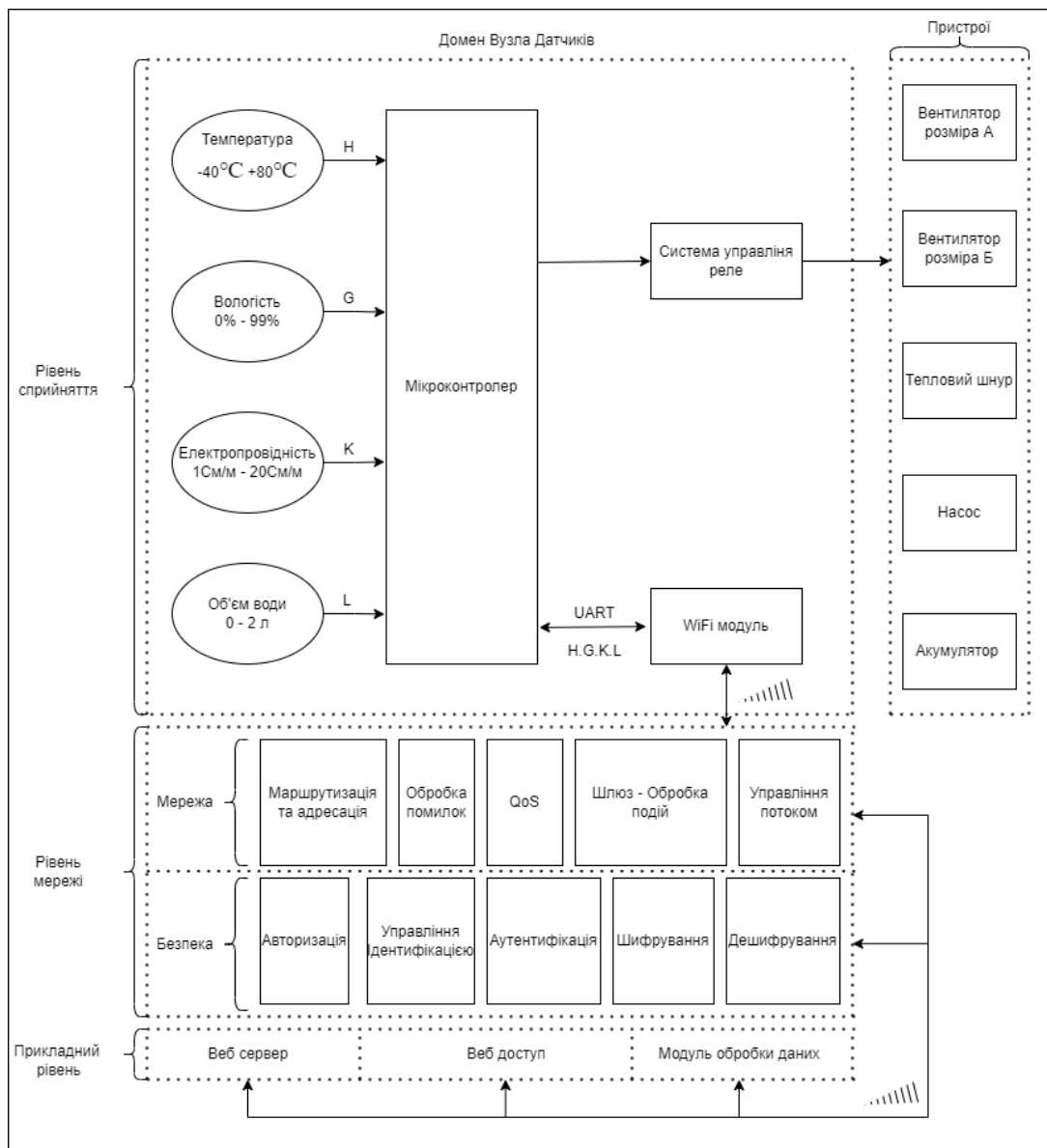


*Рисунок 2.3 - Діаграма передачі даних з апаратних засобів в програмний з відображенням мережі*

Ця діаграма показує що за допомогою WiFi технології, дані з датчиків передаються мікроконтролером в мережу за допомогою протоків передачі даних, де вони створюється резервна копія даних, щоб надалі мати значення про кожен день роботи інкубатора, а також на пристрій користувача в додаток, який підтримується ОС IOS та Android.

Також було розроблено більш детальна діаграма, процесу роботи розумного інкубатора в мережі, із забезпеченням необхідними даними користувача, а також автоматичним управлінням інкубатора (Рис. 2.4).

Процес починається з скидання даних з датчиків. Далі дані за допомогою технології UART передаються на мікроконтролер, а з мікроконтролера на WiFi модуль, який у свою чергу за допомогою технології WiFi передає інформацію з датчиків на рівень мережі. Рівень мережі містить два підпункти - мережу та безпеку. У розділі мережа перевіряється така інформація про дані, які передаються, як Маршрутизація та адресація, обробка помилок, QoS, шлюз та управління потоком. Розділ Безпека включає перевірку авторизації, щоб інформація не потрапила до рук зловмисника, управління ідентифікацією, автентифікація, шифрування та дешифрування.



*Рисунок 2.4 - Більш детальна схема роботи розумного інкубатора в мережі*

Після цих двох етапів дані передаються на прикладний рівень. Там вони розподіляються на веб-сервер, веб доступ і на модуль обробки даних. Після того як дані потрапили на веб-сервер, йде порівняння даних і значень, які повинні бути. На мікроконтролер посилається завдання, для кожного пристрій залежно від потреб, і за допомогою системи керування роздаються команди на пристрої.

### 2.3. Моделі для створення бази даних

Для створення бази даних необхідно описати саму структуру, а саме концептуальну, логічну та фізичні моделі.

Концептуальна модель більше створюється для більшого розуміння самого процесу передавання даних (Рис. 2.5). У цьому випадку мікроконтролер за допомогою технології UART отримує значення з датчиків. Потім передає дані за допомогою модуля WiFi на веб-сервер. Інформація з веб-сервера передається користувачам разом з отриманими даними. Така же інформація поступає в базу даних. Після цього, якщо значення не виявляються допустимими на етапі обробки даних, що дають запити на мікроконтролер, із задачами для пристроїв, для приведення роботи інкубатора в норму.

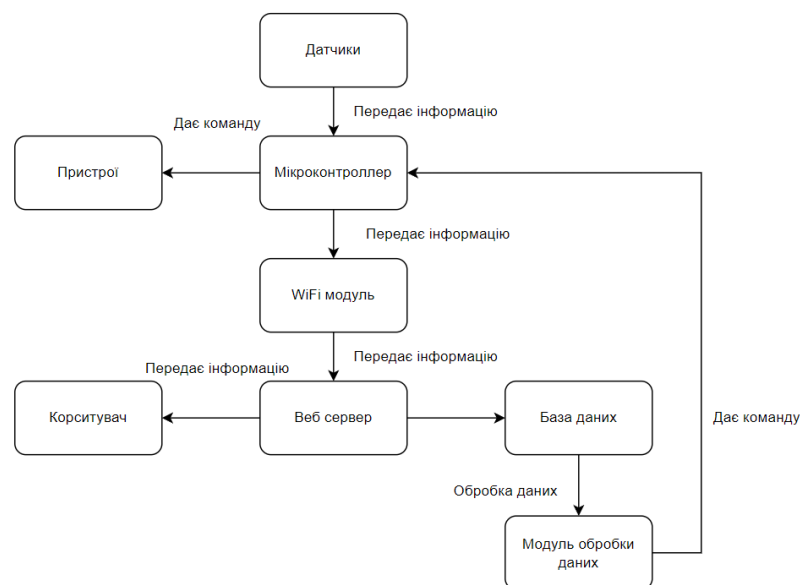


Рисунок 2.5 - Концептуальна модель "Розумного інкубатора"

Наступним кроком проектування було створення логічної моделі (Рис. 2.6) та налагодження зв'язків у ній. Логічна модель відображає сутність та відносини між таблицями. На даному прикладі інкубатор включає в себе сенсори, котрі містять інформацію стосовно логів з сенсорів та дані з датчиків.

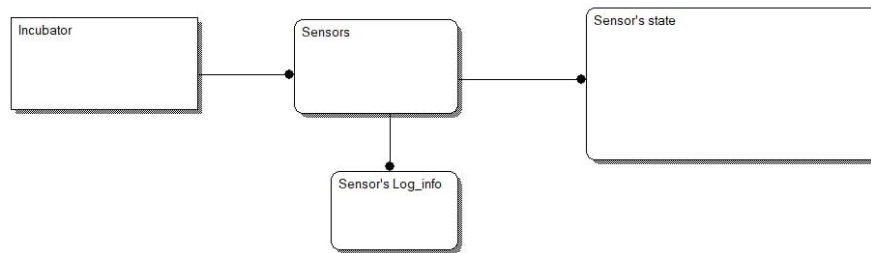


Рисунок 2.6 - Логічна модель схема "Розумного інкубатора"

Після створення логічної моделі треба переходити до фізичної. Фізична модель (Рис. 2.7) описує які саме дані з яких пристроїв будуть передані до бази даних. Специфікація таблиць включає в себе назву таблиць, тип даних. Вона також містить первинні ключі з кожної таблиці, а також показує зв'язок між таблицями.

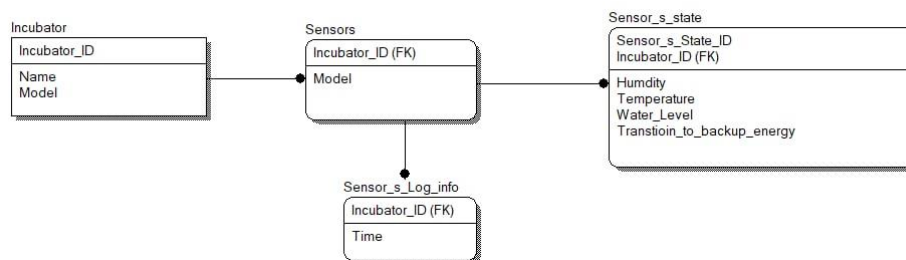


Рисунок 2.7 - Фізична модель схема "Розумного інкубатора"

У даному випадку таблиця інкубатор включає в себе назву та модель інкубатора. Таблиця з сенсорами модель сенсора. Від сенсорів відходять дві таблиці логи з сенсора та стан сенсорів, тобто дані з датчиків. В таблиці дані з датчиків знаходиться інформація стосовно температури, вологості, рівня води, а також стан інкубатора в мережі, саме на цих даних і буде працювати база даних. Також з таблиці логи датчиків, братиметься інформація, коли саме дані були передані з датчиків

**Висновок до розділу 2:** Розроблено кілька діаграм та блок-схем роботи інкубатора. Перша діаграма включала сам приклад роботи розумного інкубатора. Блок-схема розумного інкубатора показує процес аналізу інкубатором датчиків

та процесів при порушеннях. Так само була створена діаграма передачі даних на апаратні пристрої користувачів. Фінальною діаграмою був докладний процес роботи розумного інкубатора, що включає абсолютно всі процеси, від передачі даних, до функцій роботи пристроїв.

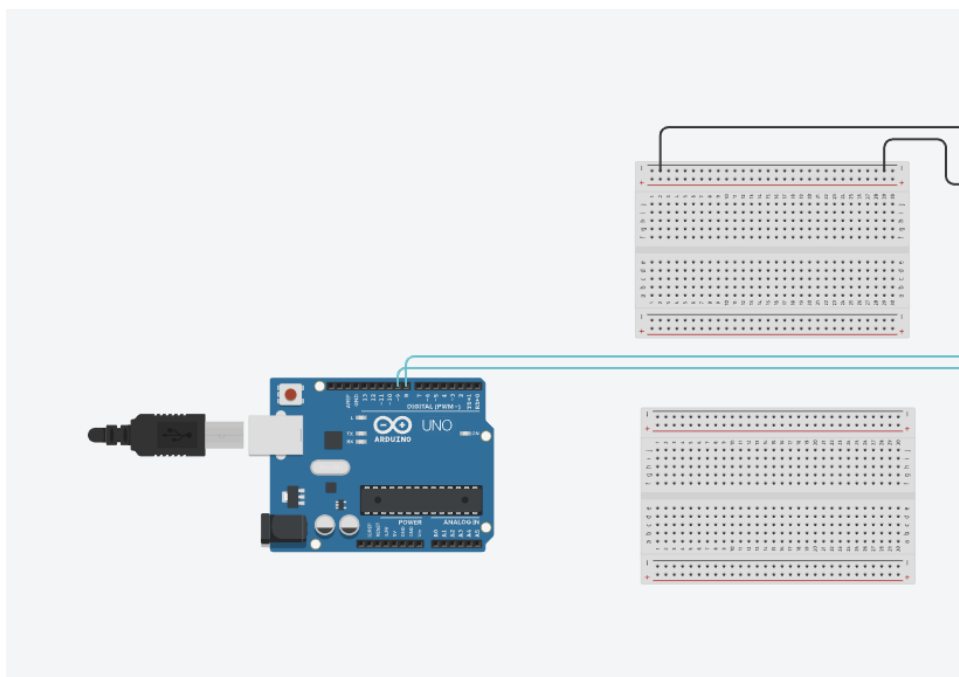
### 3. ПРОГРАМНА РЕАЛІЗАЦІЯ ПРОЕКТУ РОЗУМНИЙ ІНКУБАТОР

#### 3.1 Створення симуляції процесу роботи датчиків інкубатора

Розглянувши аналоги всіх елементів системи потрібних для повноцінної роботи інкубатора, а також створивши діаграми процесу роботи потрібно переходити до створення програмного засобу.

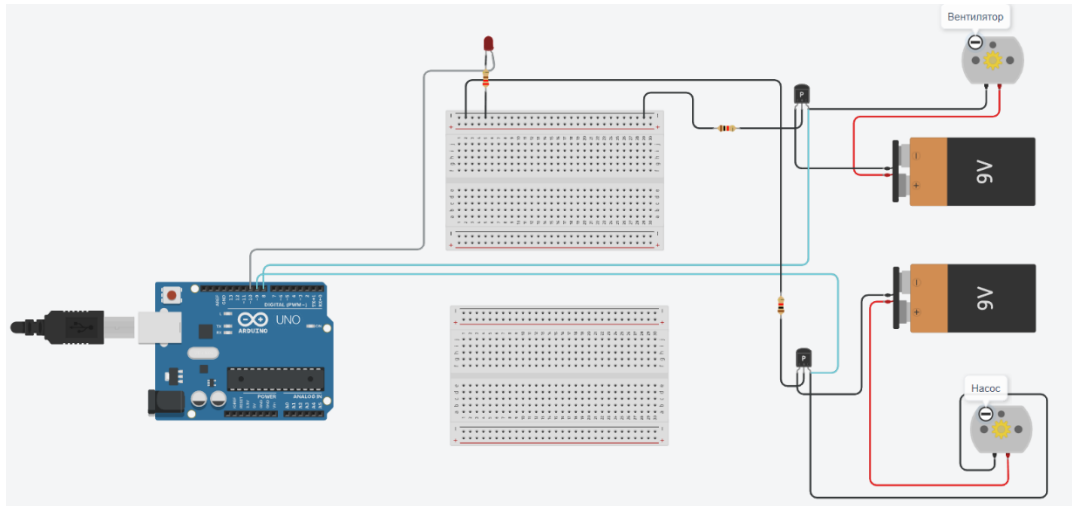
Для симуляції процесу роботи інкубатора було вирішено використати Tinkercad. Tinkercad - це онлайн сервіс, який отримав можливість створення електронних схем та підключення їх до віртуальної плати Arduino.

Для початку було вирішено почати з двох маленьких плат та однієї ардуїни, на першій платі будуть задіяні самі процеси, а друга відповідатиме за візуалізацію на екрані інкубатора (Рис. 3.1). У цьому випадку на LCD дисплеї.



*Рисунок 3.1 - Приклад базової схеми інкубатора*

Потім потрібно було підключити до плат необхідні пристрої. Вентилятор для охолодження інкубатора. Насос, для накачки води, і лампочка, яке буде відповідати за показання, чи підключений до мережі інкубатор (Рис. 3.2).

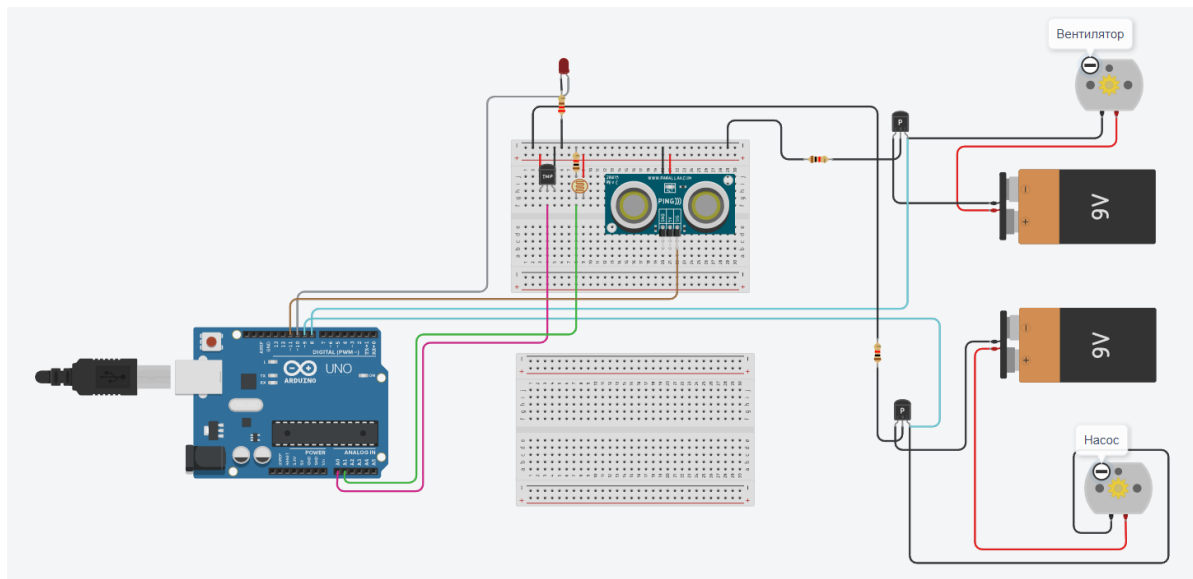


*Рисунок 3.2 - Приклад схеми інкубатора, з підключенням пристроїв*

Після того як підключені пристрої, тепер потрібно щоб були датчики, отримуючи значення з яких, пристрої будуть приводитися в дії. Першим буде свого роду датчик температури, він необхідний, щоб отримувати інформацію про температуру в інкубаторі, і коли вона досягатиме значень вище потрібних, в даному випадку більше 38, буде включатися вентилятор, який охолоджуватиме систему.

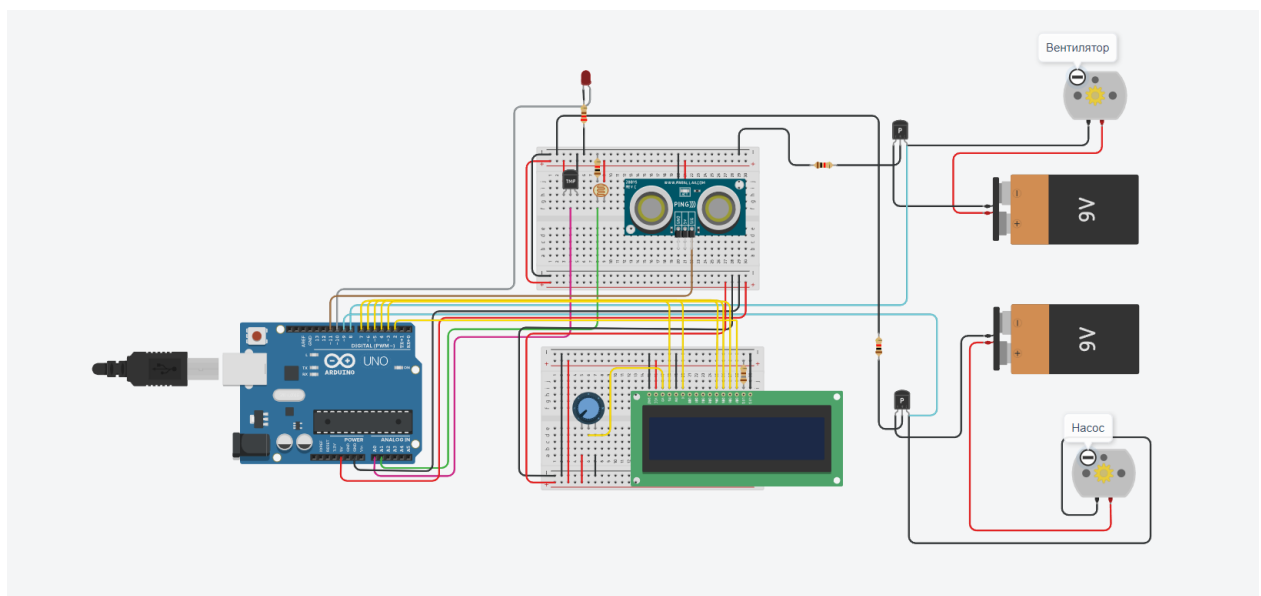
Наступним буде пристрій, за допомогою якого ми контролюватимемо провідність струму в інкубаторі. Використовуватиметься фоторезистор, він буде підключений до лампочки. Якщо лампочка світиться, фоторезистор приймає світло, і нічого не відбувається, але якщо лампочка тухне, фоторезистор перестає отримувати світло, а значить сам інкубатор не буде підключений до мережі. Це свого роду показник чи підключений інкубатор до мережі чи ні.

Останнім датчиком став ультразвуковий датчик. З його допомогою можна вимірювати відстань до певного об'єкта (Рис. 3.3). Він буде поміщений у контейнер з водою і показуватиме певну відстань до води. Якщо ця відстань збільшується, води стає менше. Коли значення доходять до певних, спрацьовує насос, який накачує воду до певного рівня.



*Рисунок 3.3 - Приклад схеми інкубатора, з підключеними датчиками*

Як і в будь-якому інкубаторі, необхідний екран, який виводитиме нам значення, які існують на даний момент (Рис. 3.4). Для використання екрана було вибрано використовувати дисплей LCD. На ньому буде виводитися температура в інкубаторі, і коли значення будуть доходити до певних, пристрої які будуть відображатися на ньому.



*Рисунок 3.4 - Приклад схеми інкубатора, з підключеним LCD дисплеєм*

Після того як всі пристрої підключені, перше, що потрібно зробити, перетворити ультразвуковий датчик так, щоб з нього можна було отримувати

відстань до води (Рис. 3.5), тим самим даючи насосу команду, коли йому потрібно накачувати воду.

```
    cm = microsecondsToCentimeters(duration);

    Serial.print("Distance: ");
    Serial.print(cm);
    Serial.print("cm");
    Serial.println();

    if(cm > 100) {

        digitalWrite(motor, HIGH);
        lcd.setCursor(0,0);
        lcd.print("pump 1");
    }

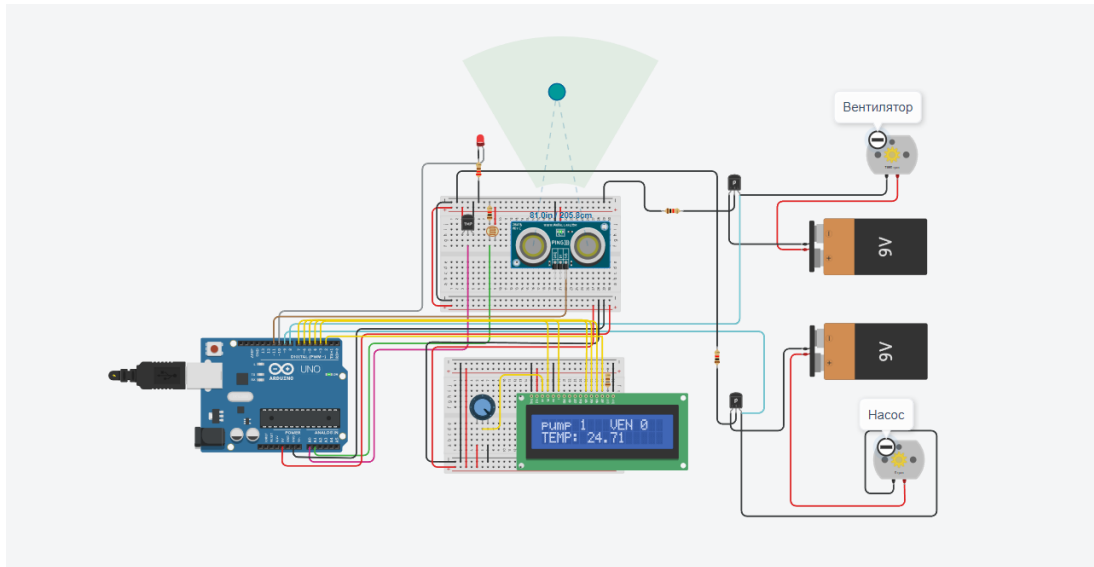
    if(cm < 20)
    {
        digitalWrite(motor, LOW);
        lcd.setCursor(0,0);
        lcd.print("pump 0");
    }

    delay(100);
}
```

*Рисунок 3.5 - Перетворення часу на відстань*

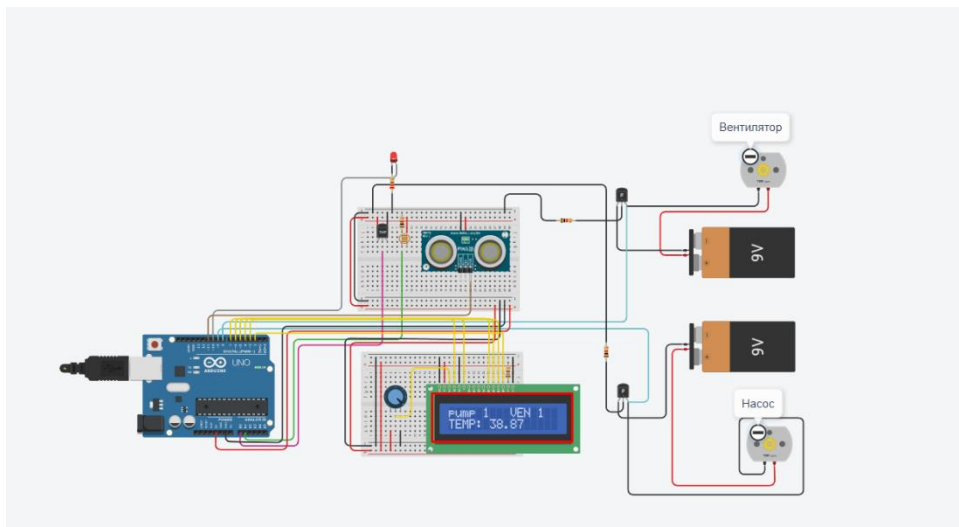
За допомогою першої сточки коду на скріншоті час конвертується в відстань, тим самим даючи можливість визначати коли рівень води досяг мінімум, а після того як рівень води досягає певної позначки, насос починає свою роботу.

На (Рис. 3.6) можна побачити що коли повзунок датчик доходить до певних значення на LCD дисплеї виводиться значення pump1, що означає що насос почав свою роботу, коли ж значення буде pump0 це означає, що насос перестав працювати.



*Рисунок 3.6 - Работа насоса*

Також працює і датчик температури з вентилятором. Коли температура досягає певних значень, LCD дисплей показує Vent1, це означає, що температура перевищила допустимі значення і вентилятор почав свою роботу (Рис. 3.7).



*Рисунок 3.7 - Приклад контролю температури*

Симуляція інкубатора на платформі Tinkercad була зроблена, були враховані певні умови роботи інкубатора, а також візуалізація у вигляді дисплея LCD. Код програми знаходиться у додатку А.

### 3.2 Візуалізація роботи інкубатора з базою даних та системою оповіщень

Для роботи бази даних та оповіщення певних значень у додатку, перш за все потрібно створити інтерфейс (Рис. 3.8). Інтерфейс буде розроблено на базі html та css, буде створено сайт із певними таблицями. Таблиці будуть вміщувати в собі інформацію про стабільну роботу інкубатора, назви датчиків і пристроїв. Також будуть зроблені дві кнопки: введення інкубатора в робочий стан, і симуляція помилок. За допомогою кнопки симуляції помилок буде створюватися збої в роботі інкубатора, а також виводитиметься повідомлення, які саме значення змінилися. Під час того як значення змінюються, біля кожного пристрою, яке відповідає за певний параметр, світитиме зелена кнопка "On", що означає роботу пристрою, і буде світити поки що значення на інтерфейсі (Рис. 3.2.) не прийде в норму.

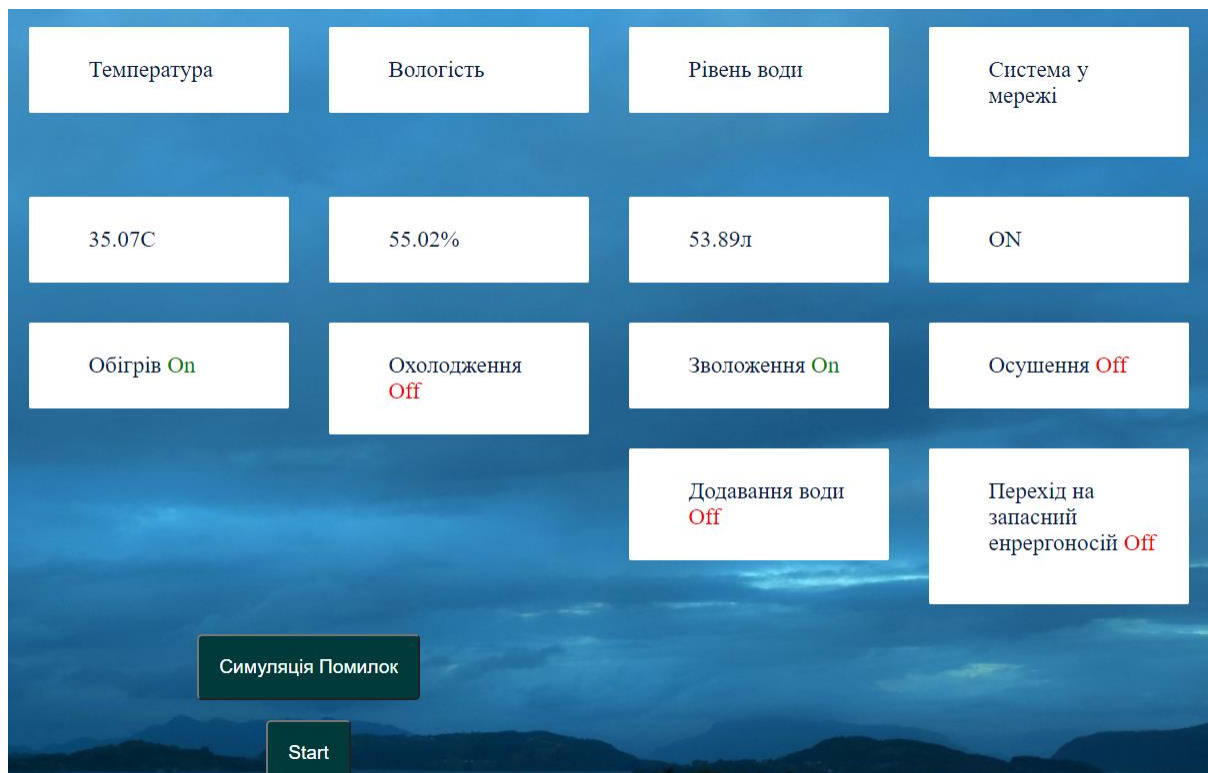


Рисунок 3.8 - Приклад роботи інтерфейсу

Таким чином, будуть створювати значення, необхідні для подальшої роботи з ними. Дані отримані з інтерфейсу будуть перенаправлятися до бази даних та у додаток користувача. Код інтерфейсу додано у Додаток Б.

Додатком користувача було вирішено використовувати телеграм бота. По-перше телеграм можна встановити на різні версії ОС. По-друге це досить популярна платформа, якою користуються мільйони людей щодня. Для зручності користувача йому надходитиме повідомлення, якщо значення порушені, прямо в телеграм бота. Все що потрібно користувачеві, це підключити свій інкубатор до чат бота, щоб отримувати повідомлення безпосередньо від свого інкубатора (Рис. 3.9).

```
var temperature = getRandomIntInclusive(30, 50);
var humidity = getRandomIntInclusive(50, 70);
var water = getRandomIntInclusive(0, 2);
var energy = getRandomIntInclusive(1, 2);

var energyinf = (energy === 1) ? "Перехід на запасний енергоносіЙ": "Без переходу на запасний енергоносіЙ";

var emergency = "Нові показники системи: \n Температура: " + temperature.toString() +
"\n Вологість: " + humidity.toString() +
"\n Рівень води: " + water.toString() +
"\n " + energyinf;

alert(emergency);

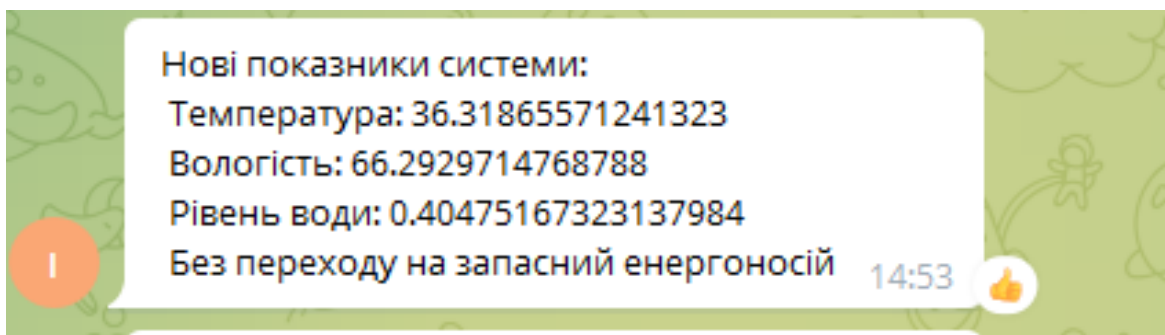
encoded_text = encodeURIComponent(emergency);

var url = "https://api.telegram.org/bot5316359945:AAEqu_4XMi_jR11-2wIqxcw716Xv3rHX4H7k/sendMessage?chat_id=346642670&text=" + encoded_text + "&parse_mode=HTML";
//var url = "https://api.telegram.org/bot5316359945:AAEqu_4XMi_jR11-2wIqxcw716Xv3rHX4H7k/getUpdates";

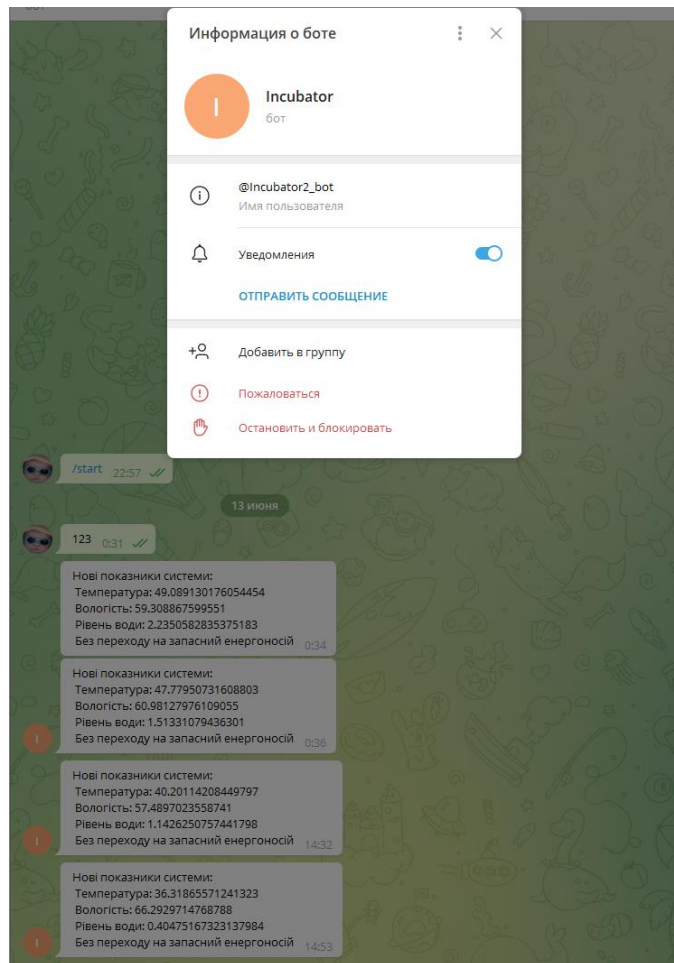
$.ajax({
  type: "GET",
  url: url,
  success: function(data)
  {
    alert(JSON.stringify(data));
  }
});
```

*Рисунок 3.9 - Приклад коду для відправки значень до бота*

Під час створення бота в телеграм, потрібно було взяти його API ключ, і написати певний код (Рис. 3.10), який буде відправляти вихідні значення в телеграм боту (Рис. 3.11), який у свою чергу буде повідомляти користувача про поломки або критичні зміни.



*Рисунок 3.10 - Приклад переданих даних до чат бота*



*Рисунок 3.11 - Приклад чат бота*

Після створення повідомлень користувача. Потрібно переходити до створення бази даних для зберігання інформації та фільтрації даних з помилками від звичайних даних роботи інкубатора.

Спочатку потрібно було створити саму модель, як саме дані взаємодіятимуть одна з одною. Для розробки моделі був використаний Erwin Data Modeler, досить проста програма, яка дозволяє не тільки візуалізувати саму суть моделі бази даних, але й зробити базову модель (Рис. 3.12), яку можна буде занести на вже готову платформу, та підключити базу даних.

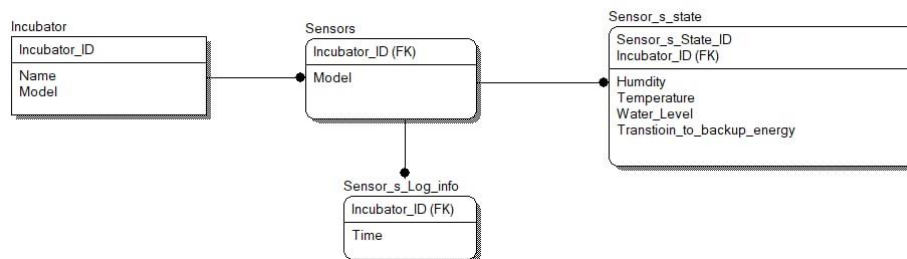


Рисунок 3.12 - Приклад фізичної моделі бази даних

Коли вже є базова структура, час переходити до самого додатку, де ця база даних буде розташована. Основою програми буде PostgreSQL. PostgreSQL – “вільна об’єктно-реляційна система управління базами даних. Існує в реалізаціях для багатьох UNIX-подібних платформ, включаючи AIX, різні BSD-системи, HP-UX, IRIX, Linux, macOS, Solaris/OpenSolaris, Tru64, QNX, а також для Microsoft Windows. PostgreSQL базується на мові SQL і підтримує багато можливостей стандарту SQL 2011”[11].

База даних буде розміщена на PostgreSQL (Рис. 3.13), а саме на зручному інструменті управління базами даних з відкритим вихідним кодом PgAdmin 4. З його допомогою можна буде налаштовувати всю роботу бази даних, редагувати, додавати нові рядки, стовпці.

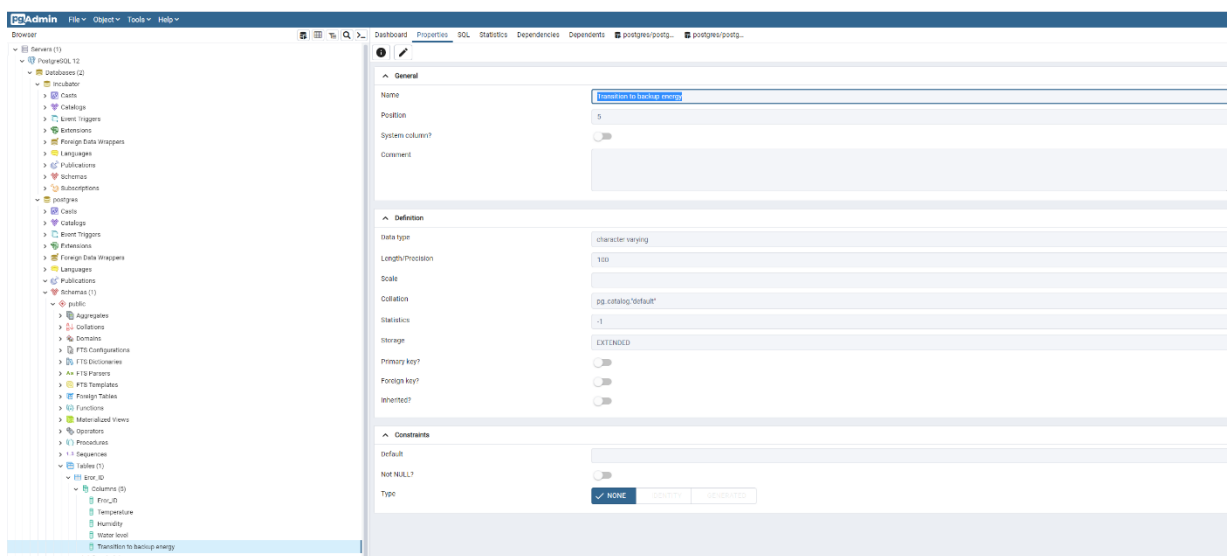


Рисунок 3.13 - Приклад програми PgAdmin 4

Для того, щоб користувач міг самостійно заходити і більш детально оглядати інформацію щодо даних отриманих з інкубатора, сам дашборд програми буде розміщений на портативному локальному сервері Openserver (Рис. 3.14). Openserver - це багатофункціональна програма, що має в наявності великий вибір компонентів, що підключаються, це повноцінний професійний інструмент, який був розроблений спеціально для веб-розробників.

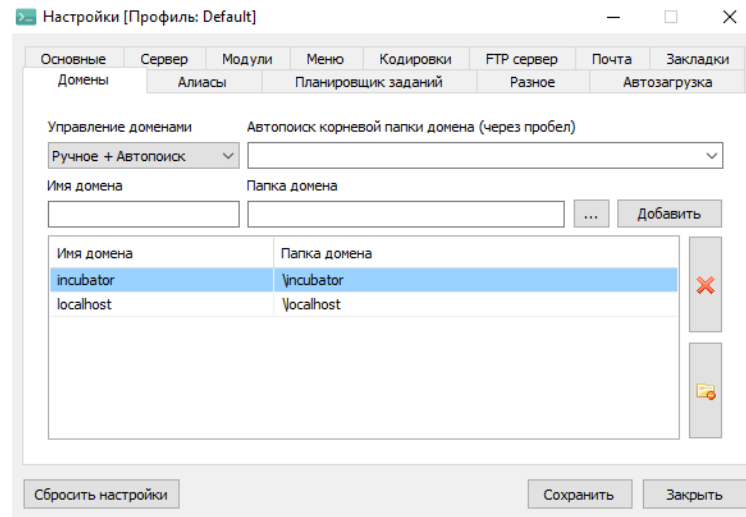


Рисунок 3.14 - Пример программы Openserver

Відтак PgAdmin 4 та Openserver будуть використовуватись разом для розміщення бази даних на сайті (Рис. 3.15).

▼ Действия ▼ Группировка 🔍 Поиск 📁 Импорт 📄 Экспорт + Добавить новую запись

<input type="checkbox"/>	ID ↕	Temperature Σ ↕	Humidity Σ ↕	Water Level Σ ↕	Transition to backup energy Σ ↕	time Σ ↕
<input checked="" type="checkbox"/>	5	37.800	67.000	1.800	0	10:00
<input type="checkbox"/>	1					
<input checked="" type="checkbox"/>	4	37.500	56.000	2.000	0	09:45
<input type="checkbox"/>	2					
<input checked="" type="checkbox"/>	3	38.000	61.000	2.000	0	09:30
<input type="checkbox"/>	3					
<input checked="" type="checkbox"/>	2	37.800	61.000	2.000	0	09:15
<input type="checkbox"/>	4					
<input checked="" type="checkbox"/>	1	37.700	61.000	2.000	0	09:00
<input type="checkbox"/>	5					

Рисунок 3.15 - Пример размещенной базы данных на сайте

**Висновок до розділу 3:** Розроблена симуляція інкубатора в додатку Tinkercad для кращої візуалізації на рівні мікросхем, всього процесу роботи. Створено зручний інтерфейс для симуляції помилок під час роботи інкубатора. Дані помилки перенаправляються прямо в додаток користувача, а саме в телеграм бота. Також була створена база даних для моніторингу роботи інкубатора, і виявлення проблем, коли вони відбувалися і з чим саме.

## ВИСНОВОК

У даній кваліфікаційній роботі бакалавра запропоновані методи покращення процесу роботи інкубатора засобами IoT. На основі цих методів спроектовано та програмно реалізовано ефективну систему IoT рішень для процесу роботи інкубатора з підтримкою автоматизованого управління процесами обігріву, охолодження, зволоження, осушування, додавання води та переходу на запасний енергоносіє.

Виконано:

1. Аналіз базових принципів роботи IoT систем.
2. Аналіз аналогів для побудови моделі потрібного інкубатора.
3. Аналіз процесу роботи інкубатора.
4. Описати базову діаграму процесу роботи інкубатора.
5. Описати діаграму передачі даних з апаратних засобів в програмний з відображенням мережі.
6. Розробити блок-схему роботи інкубатора.
7. Розробити симуляцію інкубатора.
8. Розроблено програмний продукт та графічний інтерфейс для системи IoT рішень для роботи інкубатора з підтримкою автоматизованого управління процесами обігріву, охолодження, зволоження, осушування, додачі води та автоматичним переходом на запасний енергоносіє.
9. Проведення симуляцію роботи датчиків засобами прототипу на основі Arduino.

Під час створення даної системи отримано наступні наукові та практичні результати:

1. Проаналізовано базові принципи роботи технології Інтернету речей. Дана технологія все більше набирає обертів в повсякденному житті. Популярними галузями впровадження IoT рішень є медицина, освіта, торгівля, економіка, агрокультура та промисловість. В основі архітектури IoT рішень є 4

вузли: датчики та пристрої, мережа, методи обробки даних та інтерфейс користувача. Задачі, які вирішує впровадження IoT автоматизація багатьох процесів та покращення життя людей. З кожним днем надається перевага широкому впровадженню IoT технологій.

2. Аналіз останніх наукових досліджень підкреслює актуальність розвитку Промислового Інтернету речей. Дана технологія використовується в різних сферах виробництва: обладнання для розумних будинків, транспорту, побутової техніки. При цьому відсутній варіант реалізації системи IoT рішень з інкубаторами для вирощування яєць. Що підкреслює новизну та актуальність обраної теми дослідження.

3. Проаналізовано готові інкубатори на ринку, серед яких King Suro20. Оснащений автоматичною системою нагрівання та охолодження, а також контролем вологості. З характеристик була зроблена таблиця порівняльних характеристик між даним інкубатором і проектом бакалаврської роботи. Визначено вхідні дані для проектування моделі систему IoT рішень для процесу сироваріння.

4. Систематизуючи результати проведених аналізів, відсутність автоматичного додавання води та автоматичного переходу на запасний енергоносії, а також засобів реалізації системи IoT рішень для інкубатора, свідчить про те що навіть передові моделі інкубаторів не є ідеальними. Тому вважаємо за доцільне розгляд нової технологічної пропозиції за темою бакалаврської роботи. Прийнято рішення розробити систему, яка є простою в реалізації, підтримує автоматичну нормалізацію необхідної для роботи інкубатора, з урахуванням вище перерахованих недоліків сучасних інкубаторів.

5. Реалізовано проектну частину системи: побудовані діаграми процесу роботи інкубатора, що показують порядок дій, взаємозв'язки, розроблено алгоритм роботи системи та блок-схему. Проведено вибір компонентів необхідних для відповідності поставленої задачі. Мікроконтролер XM-18SW зчитує необхідні дані з датчиків, та відповідно результату порівняння отриманих

даних. МК подає сигнал на відповідне реле для виконання операції відносно нормалізації показників. Також побудовано схему мережі зв'язку системи.

6. Розроблено модель для симуляції роботи датчиків інкубатора на базі Arduino, за допомогою платформи Tinkercad. За допомогою пристроїв та спеціально написаного коду у додатку А, було реалізовано процес роботи інкубаторів на рівні мікроконтролерів.

7. Розроблено систему передачі даних з інкубатора користувачеві, та зберігання показників в базу даних. Для цього було створено інтерфейс, який симулює помилки в системі. Отримані помилки відправляються у додаток створений на базі телеграм бота з автоматичною системою оповіщень. Також дані з інтерфейсу записуються в таблиці бази даних. База даних створена за допомогою панелі редактора Erwin Data Base, PgAdmin4, та Openserver.

8. Проведено порівняння розробленої системи IoT рішень для роботи інкубатора з аналогами, розглянутими в розділі 1. Власна розробка врахувала всі недоліки аналогів, має покращений функціонал, а також відповідає закладеним на початку проектування вимогам.

Мета кваліфікаційної роботи бакалавра досягнута.

## ПЕРЕЛІК ВИКОРИСТАНИХ ІНФОРМАЦІЙНИХ ДЖЕРЕЛ

1. Дані про охорону праці в Україні, 2020 [Електронний ресурс] / Режим доступу: <https://zakon.rada.gov.ua/laws/show/z0035-93> (Дата звернення 2.05.2022)
2. Пошук аналогів для інкубатора, 2022 [Електронний ресурс] Режим доступу: <http://prom.ua> (Дата звернення 7.05.2022)
3. Дані про побудову інкубатора, 2022 [Електронний ресурс] Режим доступу: <https://www.youtube.com/c/minifermeros> (Дата звернення 10.05.2022)
4. Дані про побудову інкубатора, 2022 [Електронний ресурс] Режим доступу: <https://avanguardco.ua> (Дата звернення 10.05.2022)
5. Дані про побудову інкубатора, 2022 [Електронний ресурс] Режим доступу: <https://latifundist.com> (Дата звернення 10.05.2022)
6. Дані про системи захисту IP 2019 [Електронний ресурс] / Режим доступу: <https://asoft.by/> (Дата звернення 3.05.2022)
7. Дані про сайт для створення схем Draw.io, 2022 [Електронний ресурс] / Режим доступу: <https://app.diagrams.net/> (Дата звернення 3.05.2022)
8. Дані про інкубатор для яєць R-Com King Suro20, 2022 [Електронний ресурс] / Режим доступу: <https://taya.com.ua/ogljad-avtomatichnogo-inkubatora-dlja-jaec-r-com/> (Дата звернення 3.05.2022)
9. Node-Red 2022 [Електронний ресурс] / Режим доступу: <https://uk.wikipedia.org/wiki/Node-RED> (Дата звернення 3.05.2022)
10. Apex Oracle 2022 [Електронний ресурс] / Режим доступу: <https://soware.ru/products/oracle-apex> (Дата звернення 3.05.2022)
11. Дані про PostgreSQL 2022 [Електронний ресурс] / Режим доступу: <https://ru.wikipedia.org/wiki/PostgreSQL> (Дата звернення 3.05.2022)
12. Дані про PostgreSQL 2022 [Електронний ресурс] / Режим доступу: <https://ospanel.io> (Дата звернення 3.05.2022)

13. Інформація про інкубатори 2022 [Електронний ресурс] / Режим доступу: <https://mlunok.com.ua/uk/blog/stati/avtomatichnij-inkubator-instrukciya-vikoristannya> (Дата звернення 3.05.2022)
14. Інформація про інкубатори 2022 [Електронний ресурс] / Режим доступу: <https://jeou.donnu.edu.ua> (Дата звернення 3.05.2022)
15. Інформація про інкубатори 2022 [Електронний ресурс] / Режим доступу: <https://mirinkub.ru/poleznaja-informacija/Kak-sdelat-inkubator-samostojatelno> (Дата звернення 3.05.2022)
16. Tinkercad 2022 [Електронний ресурс] / Режим доступу: <https://www.tinkercad.com> (Дата звернення 3.05.2022)
17. Наукова стаття про інкубатори 2022 [Електронний ресурс] / Режим доступу: [http://www.economy.nayka.com.ua/pdf/12\\_2019/7.pdf](http://www.economy.nayka.com.ua/pdf/12_2019/7.pdf) (Дата звернення 3.05.2022)
18. Мікроконтролер 2022 [Електронний ресурс] / Режим доступу: <https://fresh.co.ua/product-11624269737> (Дата звернення 3.05.2022)

## Додаток А

Код від програми для роботи інкубатора в Tinkercad

Листів 3

Розробник

Шестопалов М.І.

Керівник

Квращенко О.В.

Київ – 2022

## Код від програми для роботи інкубатора в Tinkercad

```
#include <LiquidCrystal.h>
LiquidCrystal lcd(7,6,5,4,3,2);

const int pingPin = 11;
int tempsensor= A0;
int photoresistor= A1;

int sensorValue = 0;
char degree = 176;

int motor=8;
int ventilator=9;
int ledPin=10;
void setup()
{

  Serial.begin(9600);

  lcd.begin(16,2);

  pinMode(motor, OUTPUT);
  pinMode(ventilator, OUTPUT);
  pinMode(ledPin, OUTPUT);

  pinMode(photoresistor, INPUT);
  pinMode(tempsensor,INPUT);
}

void loop() {

  int tmp = analogRead(tempsensor);
  float voltage = (tmp * 5.0)/1024;
  float milliVolt = voltage * 1000;
  float tmpCel = (milliVolt-500)/10 ;

  lcd.setCursor(0,1);
  lcd.print("TEMP:");
  lcd.setCursor(6,1);
  lcd.print(tmpCel);

  if(tmpCel>38)
  {
    digitalWrite(ventilator,HIGH);
    lcd.setCursor(8,0);
    lcd.print(" VEN 1");
  }
  else{
    digitalWrite(ventilator,LOW);
    lcd.setCursor(8,0);
    lcd.print(" VEN 0");
  }

  sensorValue = analogRead(photoresistor);
  if(sensorValue<=511)
  {
    digitalWrite(ledPin,HIGH);
```

```

}
else {
    digitalWrite(ledPin, LOW);

}

long duration, cm;

pinMode(pingPin, OUTPUT);
digitalWrite(pingPin, LOW);
delayMicroseconds(2);
digitalWrite(pingPin, HIGH);
delayMicroseconds(5);
digitalWrite(pingPin, LOW);

pinMode(pingPin, INPUT);
duration = pulseIn(pingPin, HIGH);

cm = microsecondsToCentimeters(duration);

Serial.print("Distance: ");
Serial.print(cm);
Serial.print("cm");

Serial.println();

if(cm > 100) {

    digitalWrite(motor, HIGH);
    lcd.setCursor(0,0);
    lcd.print("pump 1");
}

if(cm < 20)
{
    digitalWrite(motor, LOW);
    lcd.setCursor(0,0);
    lcd.print("pump 0");
}

delay(100);
}

long microsecondsToCentimeters(long
microseconds) {
    return microseconds / 29 / 2;
}

```

## Додаток Б

Код від програми для роботи інтерфесу інкубатора, оповіщення та зберігання даних в таблиці

Листів 8

Розробник

Шестопалов М.І.

Керівник

Квращенко О.В.

Київ – 2022

**Код від програми для  
роботи інтерфесу інкубатора,  
оповіщення та зберігання даних  
в таблиці**

```

<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport"
content="width=device-width, initial-
scale=1.0" />
    <title>Weather app</title>
    <!--Подключаем библиотеку--
>
    <script
src="https://ajax.googleapis.com/ajax/libs/
jquery/2.2.0/jquery.min.js"></script>
    <link
href="https://fonts.googleapis.com/css2?fa
mily=Indie+Flower&display=swap"
rel="stylesheet"
/>
    <!--=====
Design by foolishdeveloper.com
=====-->
    <style media="screen">
      * {
        margin: 0;
        padding: 0;
        box-sizing: border-box;

```

```

}
body {
  background: url("https://w-
dog.ru/wallpapers/9/3/525335754506241/t
eplicity-noch-pejzazh.jpg") fixed;
  background-position: center;
  background-repeat: no-repeat;
  background-size: cover;
}
/* -----background
*/
.container-fluid {
  width: 100%;
  padding: 10px;
}
.main {
  max-width: 1200px;
  margin-right: auto;
  margin-left: auto;
}
/* -----
display */
.display {
  text-align: center;
  width: 300px;
  color: #16a864;
  float: left;
}
.display2 {
  text-align: center;

```

```

width: 600px;
color: #16a864;
float: left;
}

.wrapper {
  margin: 0 9rem;
  background-color: white;
  height: 10vh;
  margin: 20px;
  border-radius: 2px;
  padding: 10px;
  height: auto;
}

.button {
  margin: 0 9rem;
  background-color:
#003A3A;
  margin: 10px auto;
  border-radius: 5px;
  padding: 20px;
  color: white;
  font-size: 1.2em;
}

.button:hover {
  margin: 0 9rem;
  background-color:
#004E70;
  margin: 10px auto;
  border-radius: 5px;
  padding: 20px;
  color: white;
  font-size: 1.2em;
}

font-size: 1.2em;
}

.start {
  margin: 0 9rem;
  background-color: grey;
  height: 5vh;
  margin: 10px auto;
  border-radius: 2px;
  padding: 10px;
}

.wrapper h2 {
  padding: 5px 0;
  text-align: center;
  background: #0548b5;
  color: white;
  font-family: sans-serif;
}

.wrapper p {
  margin: 20px 50px;
  margin-right: 20px;
  text-align: left;
  color: #04214c;
  font-size: 23px;
}

input {
  cursor: pointer;
}

```

</style>	</section>
</head>	
<body>	<section class="display">
<div class="container-fluid">	<div class="wrapper">
<section class="main">	<p
	id="temperature2">38.5C</p>
	</div>
<section class="display">	</section>
<div class="wrapper">	
<p	<section class="display">
id="temperature">Температура</p>	<div class="wrapper">
</div>	<p
</section>	id="humidity2">61%</p>
	</div>
<section class="display">	</section>
<div class="wrapper">	
<p	
id="humidity">Вологість</p>	<section class="display">
</div>	<div class="wrapper">
</section>	<p
	id="water2">2л</p>
<section class="display">	</div>
<div class="wrapper">	</section>
<p id="water">Рівень	<section class="display">
води</p>	<div class="wrapper">
</div>	<p
</section>	id="energy2">ON</p>
	</div>
<section class="display">	</section>
<div class="wrapper">	
<p	<section class="display">
id="energy">Система у мережі</p> 	<div class="wrapper">
</div>	

```
        <p id="heat">Обігрів
<span id="heat_status" style="color:
red">Off</span> </p>
```

```
        </div>
</section>
```

```
<section class="display">
```

```
  <div class="wrapper">
```

```
    <p
id="cooling">Охолодження <span
id="cooling_status" style="color:
red">Off</span></p>
```

```
    </div>
</section>
```

```
<section class="display">
```

```
  <div class="wrapper">
```

```
    <p
id="hydration">Зволоження <span
id="hydration_status" style="color:
red">Off</span></p>
```

```
    </div>
</section>
```

```
<section class="display">
```

```
  <div class="wrapper">
```

```
    <p
id="drainage">Осушення <span
id="drainage_status" style="color:
red">Off</span></p>
```

```
    </div>
</section>
```

```
<section class="display">
```

```
  <div class="wrapper">
    <p
id="waterup">Додавання води <span
id="waterup_status" style="color:
red">Off</span></p>
```

```
  </div>
</section>
```

```
<section class="display">
```

```
  <div class="wrapper">
```

```
    <p
id="energyinf">Перехід на запасний
енергоносій <span id="energyinf_status"
style="color: red">Off</span></p><br>
```

```
  </div>
</section>
```

```
<section class="display2">
```

```
  <form name="search">
```

```
    <input type="button"
class="button" value="Симуляція
Помилки" id="error"/>
```

```
  </form>
```

```
<form name="search">
```

```
  <input type="button"
class="button" value="Start" />
```

```
</form>
```

```
</section>
```

```
</section>
```

```
</div>
```

```

<script>

    // Получение случайного
    целого числа в заданном интервале,
    включительно

    function
    getRandomIntInclusive(min, max) {
        min = Math.ceil(min);
        max = Math.floor(max);
        return
        Math.floor(Math.random() * (max - min +
        1)) + min + Math.random(); //Максимум и
        минимум включаются
    }

    var rounded =
    function(number){
        return +number.toFixed(2);
    }

    $("#error").click(function() {

        var tempature =
        getRandomIntInclusive(30, 50);
        var humidity =
        getRandomIntInclusive(50, 70);
        var water =
        getRandomIntInclusive(0, 2);
        var energy =
        getRandomIntInclusive(1, 2);

```

```

        var energyinf = (energy ===
        1) ? "Перехід на запасний
        энергоносій":"Без переходу на запасний
        энергоносій";

        var emergency = "Нові
        показники системи: \n Температура: " +
        tempature.toString() +
        "\n Вологість: " +
        humidity.toString() +
        "\n Рівень води: " +
        water.toString() +
        "\n " + energyinf;

        alert(emergency);

        encoded_text =
        encodeURIComponent(emergency);

        var url =
        "https://api.telegram.org/bot5316359945:A
        AEqu_4XMijRI1-
        2wIqxcw716Xv3rHX4H7k/sendMessage?
        chat_id=346642670&text=" +
        encoded_text + "&parse_mode=HTML";

        //var url =
        "https://api.telegram.org/bot5316359945:A
        AEqu_4XMijRI1-
        2wIqxcw716Xv3rHX4H7k/getUpdates";

        $.ajax({
            type: "GET",

```

```

        url: url,
        success: function(data)
        {
alert(JSON.stringify(data));
        }
    });

    function regulation() {
        setTimeout(function () {

            if(
                (tempature < 37.5 ||
tempature > 38.5) ||
                (humidity != 61) ||
                (water < 1) ||
                (energy != 2)
            ) {

                if(tempature < 37.5)
                {
                    $("#colling_status").html("Off");

                    $("#colling_status").css("color", "red");

                    $("#heat_status").html("On");

                    $("#heat_status").css("color", "green");

                    tempature +=
                    Math.random();

                    tempature =
                    rounded(tempature);

                    $("#tempature2").html(tempature.toString(
                    ) + "C");

                }
                else {

                    tempature +=
                    Math.random();

                    $("#heat_status").html("Off");

                    $("#heat_status").css("color", "red");
                }
            }
        }, 1000);
    }
}

```

```

$("#colling_status").html("Off");
$("#colling_status").css("color", "red");
    }
    if(humidity > 61) {
        $("#hydration_status").html("Off");
        $("#hydration_status").css("color", "red");
        $("#drainage_status").html("On");
        $("#drainage_status").css("color",
"green");
        humidity -=
Math.random();
        humidity =
rounded(humidity);
        $("#humidity2").html(humidity.toString()
+ "%");
    }
    else if(humidity <
61) {
        $("#hydration_status").html("On");
        $("#hydration_status").css("color",
"green");
        $("#drainage_status").html("Off");
        $("#drainage_status").css("color", "red");
        humidity +=
Math.random();
        humidity =
rounded(humidity);
        $("#humidity2").html(humidity.toString()
+ "%");
        $("#drainage_status").html("Off");
        $("#drainage_status").css("color", "red");
        if(water < 1) {

```

```

$("#waterup_status").html("On");
$("#energyinf_status").html("On");

$("#waterup_status").css("color", "red");
$("#energyinf_status").css("color",
"green");

    water    +=
Math.random();
    }
    water    =
    else {
rounded(humidity);
$("#energyinf_status").html("Off");

$("#water2").html(water.toString() + "л");
$("#energyinf_status").css("color", "red");
    }
    }
    else {
    }

$("#waterup_status").html("Off");
    regulation();

$("#waterup_status").css("color", "red");
    }, 1000);
    }

    if(energy != 2) {
    energy = 2;
    regulation();

```

## Додаток В

Основні слайди з презентації для захисту дипломної роботи

Листів 2

Розробник

Шестопалов М.І.

Керівник

Квращенко О.В.

Київ – 2022

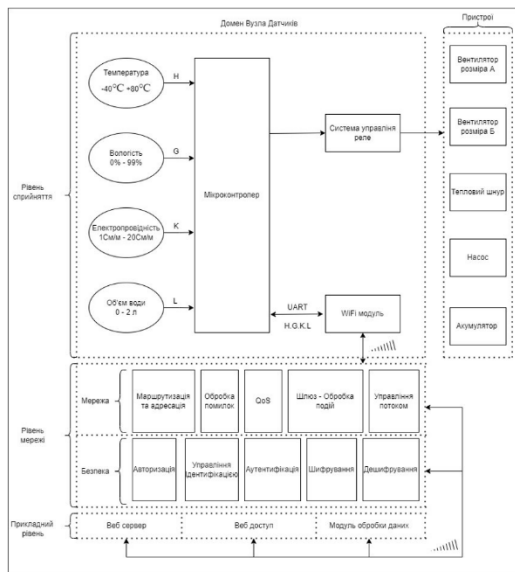
## Основні слайди з презентації для захисту дипломної роботи

### Об'єкт дослідження:

- Розумний Інкубатор
- Ключовою ідеєю роботи - це удосконалити усі недоліки минулих інкубаторів. І автоматизувати весь процес, щоб інкубатор міг спокійно працювати без взаємодії людини, та передавати необхідну інформацію користувачу в додаток.



Активация Windows  
Чтобы активировать Windows, перейдите в раздел "Параметры"



### Повна діаграма роботи розумного інкубатора

Активация Windows  
Чтобы активировать Windows, перейдите в раздел "Параметры"

