МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»

# 

Звіт до практичної роботи №2

**«Збір та візуалізація даних на сервері»**

з дисципліни:

«Інтернет речей»

на тему **«Автоматична поїлка для кота»**

Виконав: асп.гр. КІа -11 Цюник Б.С.

Прийняв: к.т.н., доцент,

Морозов Ю.В.

Львів – 2024

### Структури бази даних

Для виконання даної практичної роботи було обрано використовувати реляційну базу даних було обрано SQLite для зберігання даних.

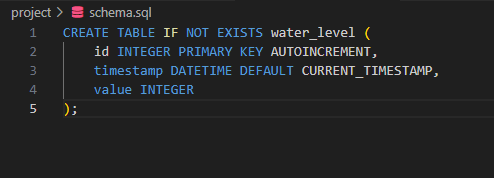


Рис.1 Схема water\_level таблиці

На рисунку 1 зображена таблиця збереження інформації переданої з сервера.

### Серверна частина системи моніторингу

Для виконання даної практичної роботи було обрано використовувати мову JS та бібліотеку express. Для підняття сервера на локалхості було використано даний функціонал:

const express = require('express');

const sqlite3 = require('sqlite3').verbose();

const app = express();

const PORT = process.env.PORT || 3000;

// Initialize SQLite database

const db = new sqlite3.Database(':memory:');

// Create table to store sensor data

db.serialize(() => {

    db.run("CREATE TABLE IF NOT EXISTS water\_level (timestamp DATETIME DEFAULT CURRENT\_TIMESTAMP, value INTEGER)");

});

// Serve static files

app.use(express.static('public'));

// API endpoint to fetch sensor data

app.get('/api/water\_level', (req, res) => {

    db.all("SELECT \* FROM water\_level", (err, rows) => {

        if (err) {

            console.error(err.message);

            res.status(500).json({ error: 'Internal Server Error' });

        } else {

            res.json(rows);

        }

    });

});

// Start server

app.listen(PORT, () => {

    console.log(`Server is running on port ${PORT}`);

});

*PUT /api/water\_level* - приймає дані вимірювань з пристрою. Використовується метод PUT. Приклад даних:

|  |
| --- |
| {  "water\_level": 2010, } |

### Візуалізація даних та інтерфейс користувача системи

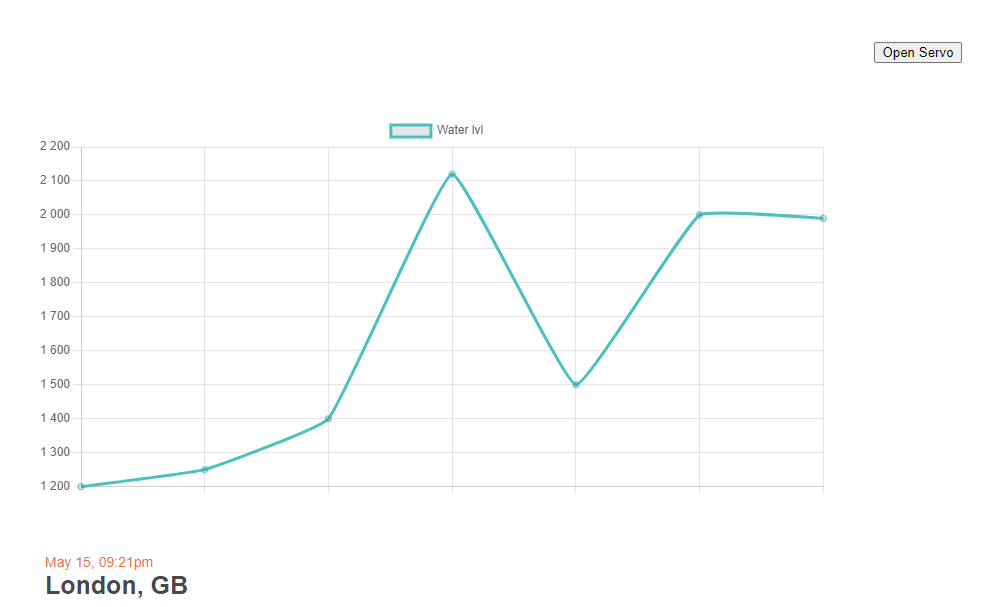


Рис.2. Візуалізація даних

На рисунку 2 зображений графік передачі даних від сенсора до бази даних і відображення у графіку.Для візуалізації даних було використано бібліотеку *ApexChart*.

function renderSensorDataChart(data) {

    // Extract timestamps and values from sensor data

    const timestamps = data.map(entry => entry.timestamp);

    const values = data.map(entry => entry.value);

    // Render chart using ApexCharts

    const chart = new ApexCharts(document.getElementById('water\_level-chart'), {

        chart: {

            type: 'line',

            height: 400,

            animations: {

                enabled: false

            }

        },

        series: [{

            name: 'Sensor Value',

            data: values

        }],

        xaxis: {

            categories: timestamps,

            type: 'datetime'

        },

        yaxis: {

            title: {

                text: 'Sensor Value'

            }

        }

    });

    chart.render();

}

**Список використаних джерел:**

1. <https://apexcharts.com/docs/installation/>
2. 2.https://expressjs.com/en/guide/overriding-express-api.html