

РОЗРОБКА ЗАСТОСУНКУ ДЛЯ ВИРОЩУВАННЯ ПЛОДОВО- ВОЧЕВИХ КУЛЬТУР

¹ Вінницький національний технічний університет;

Анотація

Розглянуто питання розробки програмного застосунку для автоматизації процесів вирощування плодово-овочевих культур. Основну увагу приділено вибору архітектури програмної системи, технологічного стеку, засобів інтеграції з IoT-пристроями та методів обробки даних. Запропоновано підхід до побудови масштабованої інформаційної системи з використанням сучасних мов програмування, фреймворків та хмарних сервісів.

Ключові слова: інформаційна система, IoT, застосунок, API, мікроклімат.

Abstract

The paper considers the development of a software application for automation of fruit and vegetable cultivation processes. Special attention is paid to system architecture, technology stack, IoT integration, and data processing methods. A scalable information system approach using modern programming tools and cloud technologies is proposed.

Keywords: information system, IoT, software application, API, microclimate.

Вступ

Цифровізація аграрного сектору обумовлює перехід від локальних систем автоматизації до розподілених інформаційних систем, що поєднують апаратні та програмні компоненти. У цьому контексті програмний застосунок виступає центральним елементом, який реалізує функції агрегації даних, їх обробки та формування керуючих впливів [1].

На відміну від класичних SCADA-подібних рішень, сучасні системи орієнтовані на використання веб-технологій, мікросервісної архітектури та хмарної інфраструктури. Це дозволяє забезпечити масштабованість, відмовостійкість та доступність системи [2].

Метою роботи є розробка застосунку для моніторингу та керування процесами вирощування плодово-овочевих культур із використанням сучасних підходів до інженерії програмного забезпечення.

Результати дослідження

Розроблюваний застосунок реалізується як розподілена інформаційна система, в якій функції збору даних, їх обробки та візуалізації рознесені між окремими програмними компонентами. На рівні пристроїв використовуються мікроконтролери (зокрема ESP32), що здійснюють первинний збір даних із сенсорів та передають їх у мережу за протоколом MQTT. Такий підхід забезпечує асинхронну взаємодію та мінімальне навантаження на канал передачі [3].

Серверна частина планується реалізувати у вигляді backend-застосунку з REST API, розробленого з використанням FastAPI або Node.js (NestJS) [4]. Вибір обумовлений підтримкою асинхронної обробки запитів та високою продуктивністю при роботі з потоками даних. Дані сенсорів зберігаються у базі часових рядів (InfluxDB), що оптимізована для запису та агрегації телеметрії, тоді як конфігураційні дані системи розміщуються у реляційній СУБД (PostgreSQL).

Клієнтська частина застосунку планується реалізувати у вигляді веб-інтерфейсу на базі React або Vue.js, що забезпечує інтерактивну візуалізацію параметрів мікроклімату в реальному часі. Обмін даними між frontend і backend здійснюється через REST API або WebSocket-з'єднання для зменшення затримок. З позиції програмної інженерії система реалізується із застосуванням контейнеризації (Docker), що дозволяє ізолювати окремі сервіси та спростити розгортання. Для масштабування може бути використана оркестрація Kubernetes, що є доцільним при збільшенні кількості пристроїв та обсягу даних [5].

Структурна організація програмної системи відображена у вигляді UML-діаграми компонентів. Загальна можлива архітектура застосунку показана на рис. 1.

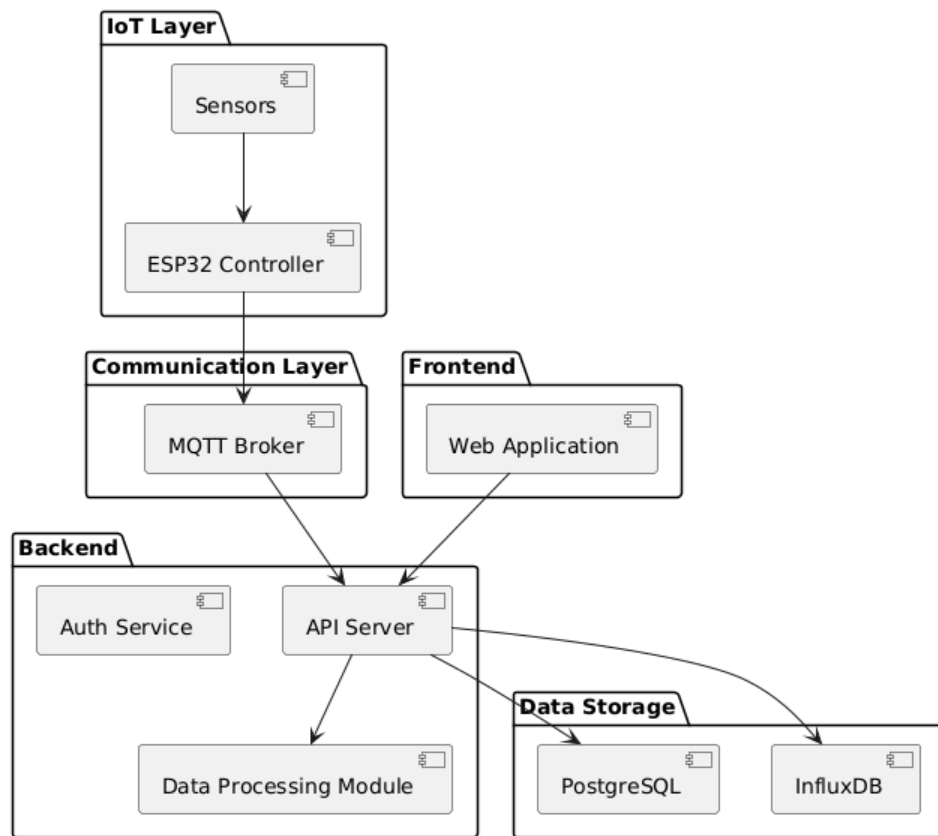


Рис. 1. UML-діаграма розгортання застосунку

Наведена модель відображає декомпозицію системи на логічні компоненти та канали взаємодії між ними. Такий підхід спрощує подальшу модифікацію системи та дозволяє незалежно масштабувати окремі підсистеми.

Висновки

Запропоновано підхід до розробки програмного застосунку як розподіленої інформаційної системи для вирощування плодовоовочевих культур. Реалізація базується на сучасних технологіях веб-розробки, IoT та обробки даних.

Отримані результати демонструють можливість побудови масштабованого рішення з чітким розподілом відповідальності між компонентами, що є характерним для сучасних IT-систем.

Подальший розвиток доцільно пов'язати з інтеграцією алгоритмів машинного навчання та адаптивного керування.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Chebudie A. B., Minerva R., Rotondi D. Towards a definition of the Internet of Things (IoT). 2014.
2. Konjeti M. Building Reliable, Scalable, and Cloud-Native Distributed Applications Using Microservices Architecture. 2025.
3. Ballamudi S. Evaluating IoT Platforms: An Approach Using the COPRAS Method. *Journal of Data Science and Information Technology*. 2025. Vol. 2. P. 55–65. URL: <https://doi.org/10.55124/jdit.v2i1.243>.
4. FastAPI. *FastAPI*. URL: <https://fastapi.tiangolo.com/> (дата звернення: 22.03.2026).
5. Bullington-McGuire R. Docker for Developers: Develop and run your application with Docker containers using DevOps tools for continuous delivery. 2020.

Лесько Олександр Владиславови - студент групи ІСТ-226, факультет інтелектуальних інформаційних технологій та автоматизації, Вінницький національний технічний університет, Вінниця, e-mail: lesko.v.o@vntu.edu.ua .

Науковий керівник – Кулик Ярослав Анатолійович – к.т.н., доцент кафедри Автоматизації та інтелектуальних інформаційних технологій, факультет інтелектуальних інформаційних технологій та автоматизації, Вінницький національний технічний університет, Вінниця, e-mail: kulyk.y.a@vntu.edu.ua.

Lesko Olsandr V. — Department of Automation and Intelligent Information Technologies, Faculty of Intelligent Information Technologies and Automation, Vinnytsia National Technical University, Vinnytsia, e-mail: lesko.v.o@vntu.edu.ua.

Supervisor: **Kulyk Yaroslav Anatoliyovych** - Ph.D., associate professor of the Department of Automation and Intelligent Information Technologies, Faculty of Intelligent Information Technologies and Automation, Vinnytsia National Technical University, Vinnytsia, e-mail: kulyk.y.a@vntu.edu.ua.