

РОЗРОБКА WEB-ДОДАТКУ ДЛЯ КЕРУВАННЯ СЕРВЕРНИМ СЕРЕДОВИЩЕМ КОРПОРАТИВНОЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ

Вінницький національний технічний університет

Анотація

Виконано розробку вебдодатку для керування серверним середовищем корпоративної інформаційної системи. Система реалізує повноцінний функціонал для автоматизованого розгортання, моніторингу, адміністрування, а також резервного копіювання та відновлення інстанцій. У додатку передбачено інтерактивний дашборд, який відображає ключові технічні метрики серверів у режимі реального часу, включаючи навантаження на процесор, обсяг використаної пам'яті, кількість активних процесів і параметри мережевого трафіку.

Ключові слова: web-додаток, автоматизація розгортання, серверна інфраструктура, моніторинг, Docker, Python.

Abstract

A web application for managing the server environment of a corporate information system has been developed. The system provides full functionality for automated deployment, monitoring, administration, as well as backup and recovery of instances. The application includes an interactive dashboard that displays key technical metrics of the servers in real time, including CPU load, memory usage, the number of active processes, and network traffic parameters.

Keywords: web application, deployment automation, server infrastructure, monitoring, Docker, Python.

Вступ

Швидкий розвиток ІТ та зростаюча залежність бізнесу від цифрових рішень ускладнюють структуру корпоративних інформаційних систем. Для забезпечення ефективної роботи таких систем критично важливою є стабільність, керованість та масштабованість серверної інфраструктури. У відповідь на ці виклики все більшого поширення набувають технології автоматизації, які зменшують трудові витрати, знижують ризик людських помилок і підвищують надійність систем. Одним із найефективніших підходів є контейнеризація, зокрема використання Docker і Docker Compose. Ці інструменти дозволяють створювати ізольовані середовища, які легко налаштовуються, масштабуються та переносяться між платформами. Водночас важливою складовою сучасного управління інфраструктурою є постійний моніторинг системних метрик — таких як навантаження, активність контейнерів та мережевий трафік — що дає змогу вчасно виявляти проблеми та забезпечувати безперервну роботу системи.

Постановка задачі дослідження

У межах дослідження необхідно виконати низку завдань, що охоплюють повний цикл створення програмного продукту. Спершу необхідно обґрунтувати доцільність розробки спеціалізованого веб-додатку для керування серверним середовищем корпоративної інформаційної системи, враховуючи потреби в автоматизації, масштабованості та моніторингу. Далі виконується порівняльний аналіз існуючих рішень, з метою виявлення їхніх сильних і слабких сторін та визначення вимог до власного рішення. На основі цього проєктується архітектура системи, обираються та обґрунтовуються технології, зокрема Python, Django, і супутні інструменти. Після розробки веб-додатку з функціями розгортання, моніторингу, резервного копіювання та керування інстанціями, проводиться тестування для перевірки його функціональності, надійності та відповідності технічним вимогам.

Результати дослідження

Результатом дослідження стало створення веб-додатку для керування серверним середовищем корпоративної інформаційної системи з урахуванням вимог до автоматизації, гнучкості,

масштабованості та моніторингу. У межах реалізації було спроектовано логічну структуру програмного забезпечення, що складається з окремих взаємопов'язаних компонентів, кожен з яких виконує чітко визначену функцію. Відповідно до функціональних вимог, архітектура системи побудована за модульним принципом, що дозволяє легко масштабувати додаток та розширювати його можливості.

Інтерфейс користувача реалізовано як багатокомпонентну клієнтську частину, побудовану з використанням HTML, CSS, JavaScript та бібліотек візуалізації Chart.js і D3.js [1]. Головна сторінка виконує роль інформаційного дашборду, де в реальному часі відображаються ключові метрики продуктивності системи. Окрім дашборду, інтерфейс включає таблицю з переліком інстанцій, а також сторінку розширеної інформації про кожну з них, що забезпечує зручну навігацію та доступ до основних дій.

Ключовим компонентом є сервіс взаємодії з Docker, побудований на основі бібліотеки python_on_whales [2]. Він відповідає за створення, запуск і зупинку інстанцій, а також виконання резервного копіювання та відновлення середовищ. Кожна інстанція розгортається на основі автоматично згенерованого файлу docker-compose.yml, з допомогою якого можна враховувати змінні оточення та специфіку конфігурацій.

Система моніторингу реалізована на основі бібліотеки psutil та низки допоміжних модулів (os, cpuinfo, platform), що дозволяють здійснювати збір системних метрик, таких як навантаження на процесор, використання оперативної пам'яті, дисковий простір, кількість активних процесів та швидкість мережевого трафіку[3]. Отримані дані передаються у форматі JSON до клієнтської частини, де візуалізуються у вигляді інтерактивних графіків.

Модуль REST API, реалізований засобами Django, дозволяє забезпечити обробку запитів від користувача, взаємодію з Docker API, запуск скриптів резервного копіювання та повернення актуальних статусів інстанцій [4]. Уся логіка взаємодії з серверною частиною зведена до роботи з HTTP-запитами, що дає змогу динамічно оновлювати інтерфейс без перезавантаження сторінки.

Додаток також містить внутрішню базу даних, де зберігається інформація про всі інстанції, їхній статус, конфігурації, історію дій користувача та метадані про останні резервні копії. Такий підхід забезпечує цілісність даних та можливість подальшої аналітики. Структурна модель взаємодії всіх компонентів представлена у вигляді UML-діаграми, що відображає логічні зв'язки між модулями системи та дозволяє наочно оцінити її архітектуру.

Висновки

У результаті проведеного дослідження було розроблено функціональний веб-додаток для керування серверним середовищем корпоративної інформаційної системи. Реалізоване рішення об'єднує інструменти автоматизованого розгортання, централізованого моніторингу, резервного копіювання та адміністрування інстанцій у зручному, інтуїтивно зрозумілому інтерфейсі. Завдяки використанню сучасного технологічного стеку, вдалося досягти високого рівня інтерактивності, гнучкості та стабільності системи. Створений програмний продукт може бути впроваджений у середовищах IT-відділів підприємств, центрів обробки даних або освітніх установ для навчання адміністрування сучасної інфраструктури.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. JavaScript – MDN Web Docs – Mozilla [Електронний ресурс] – Режим доступу: <https://developer.mozilla.org/en-US/docs/Web/JavaScript> (дата звернення: 15.05.2025).
2. Python on whales [Електронний ресурс] – Режим доступу: <https://gabriellemarmiesse.github.io/python-on-whales/> (дата звернення: 15.05.2025).
3. psutil documentation — psutil 6.0.0 documentation [Електронний ресурс] – Режим доступу: <https://psutil.readthedocs.io/en/latest/> (дата звернення: 15.05.2025).
4. Develop with Docker Engine API [Електронний ресурс] – Режим доступу: <https://docs.docker.com/engine/api/> (дата звернення: 15.05.2025).

Ткаченко Дар'я Олексіївна – студентка групи 2КН-216, факультет інтелектуальних інформаційних технологій та автоматизації, Вінницький національний технічний університет, м. Вінниця, e-mail: tkachenko.dash@gmail.com

Богач Ілона Віталіївна – к.т.н., доцент, факультет інтелектуальних інформаційних технологій та автоматизації, Вінницький національний технічний університет, м.Вінниця, e-mail: ilona.bogach@gmail.com

Tkachenko Daria Oleksiivna – student of group 2KN-21b, Faculty of Intelligent Information Technologies and Automation, Vinnytsia National Technical University, Vinnytsia, e-mail: tkachenko.dash@gmail.com

Bogach Ilona Vitaliivna – Associate Professor of Automation and Intelligent Information Technologies Department, Faculty of Intelligent Information Technology and Automation, Vinnytsia National Technical University, Vinnytsia, e-mail: ilona.bogach@gmail.com