



”

Лучкевич М. DevOps як обов'язковий компонент підготовки IT-фахівців: аналіз актуальності та перспектив. *Освіта. Інноватика. Практика*, 2025. Том 13, № 9. С. 91-97. <https://doi.org/10.31110/2616-650X-vol13i9-012>.

Luchkevych M. DevOps yak obov'iazkovyi komponent pidhotovky IT-fakhivtsiv: analiz aktualnosti ta perspektiv [DevOps as a mandatory component of IT specialist training: an analysis of its relevance and prospects]. *Osvita. Innovatyka. Praktyka – Education. Innovation. Practice*, 2025. Vol. 13, No 9. S. 91-97. <https://doi.org/10.31110/2616-650X-vol13i9-012>.

УДК 378.147:004.42:004.9

DOI: 10.31110/2616-650X-vol13i9-012

Михайло ЛУЧКЕВИЧ

Національний університет «Львівська політехніка», Україна

<https://orcid.org/0000-0002-2196-252X>

luchkevychmm@gmail.com

DEVOPS ЯК ОБОВ'ЯЗКОВИЙ КОМПОНЕНТ ПІДГОТОВКИ IT-ФАХІВЦІВ: АНАЛІЗ АКТУАЛЬНОСТІ ТА ПЕРСПЕКТИВ

Анотація. У статті досліджено актуальність інтеграції методології DevOps у систему професійної підготовки майбутніх фахівців з інформаційних технологій. Обґрунтовано, що стрімкий розвиток цифрової економіки зумовлює необхідність формування у студентів компетентностей, які поєднують знання з програмування, адміністрування, автоматизації та управління життєвим циклом програмного забезпечення. Визначено, що традиційні освітні програми з IT-спеціальностей не забезпечують належного рівня практичної готовності випускників до роботи в умовах сучасних DevOps-процесів, що створює розрив між академічною освітою та потребами IT-індустрії. На основі аналізу наукових джерел і практичного досвіду узагальнено основні принципи DevOps – автоматизацію процесів, безперервну інтеграцію та розгортання, інфраструктуру як код, моніторинг і командну взаємодію. Показано, що ці принципи мають значний дидактичний потенціал для формування у студентів системного мислення, навичок співпраці та рефлексії. Виявлено основні проблеми інтеграції DevOps у навчальний процес: дефіцит кваліфікованих викладачів-практиків, відсутність стандартизованих програм, обмеженість матеріально-технічної бази та швидку змінюваність інструментів. Запропоновано напрями подолання виявлених труднощів, зокрема залучення фахівців із галузі до викладання, розробку єдиних навчальних стандартів DevOps, використання відкритих освітніх платформ і хмарних середовищ для практичної підготовки студентів. Наголошено, що впровадження DevOps у систему вищої освіти сприятиме підвищенню конкурентоспроможності випускників, посиленню зв'язків між університетами та IT-індустрією, а також формуванню нового типу інженера – фахівця, здатного працювати в умовах безперервного вдосконалення та високої технологічної динаміки. Отримані результати мають практичне значення для розробки освітніх стандартів і програм підготовки IT-спеціалістів, орієнтованих на реальні потреби цифрової економіки.

Ключові слова: DevOps; професійна підготовка; IT-освіта; автоматизація; безперервна інтеграція.

Mykhailo LUCHKEVYCH

Lviv Polytechnic National University, Ukraine

<https://orcid.org/0000-0002-2196-252X>

luchkevychmm@gmail.com

DEVOPS AS A MANDATORY COMPONENT OF IT SPECIALIST TRAINING: AN ANALYSIS OF ITS RELEVANCE AND PROSPECTS

Abstract. This article examines the relevance of integrating the DevOps methodology into professional training for future IT specialists. It argues that the rapid growth of the digital economy requires students to develop competencies combining programming, administration, automation, and software lifecycle management knowledge. It is concluded that traditional IT degree programmes do not prepare graduates for work in modern DevOps processes, creating a gap between academic education and the needs of the IT industry. An analysis of scientific sources and practical experience has summarised the basic principles of DevOps: process automation, continuous integration and deployment, infrastructure as code, monitoring, and team interaction. These principles have significant didactic potential for developing students' systematic thinking, collaboration, and reflection skills. The main problems of integrating DevOps into the educational process have been identified as follows: a shortage of qualified teaching practitioners, a lack of standardized programs, limited material and technical resources, and the rapid changeability of tools. Proposed solutions to the identified difficulties included involving industry experts in teaching, developing uniform DevOps training standards, and using open educational platforms and cloud environments to train students. The introduction of DevOps into the higher education system was emphasised as a means of increasing the competitiveness of graduates, strengthening the relationship between universities and the IT industry, and creating a new type of engineer – a specialist capable of working in an environment of continuous improvement and high technological dynamics. The results obtained are of practical importance for developing educational standards and training programmes for IT specialists that focus on the real needs of the digital economy.

Keywords: DevOps; professional training; IT education; automation; continuous integration.

Постановка проблеми. Сучасний ринок інформаційних технологій зазнає стрімких змін, які диктують нові вимоги до компетентності фахівців. Традиційні підходи до розробки програмного забезпечення, такі як каскадна модель (Waterfall) або навіть Agile-методології без автоматизації, більше не задовольняють потреби швидкого розгортання та оновлення програмних продуктів.

DevOps (Development & Operations) – це методологія, яка інтегрує процеси розробки та експлуатації ПЗ, дозволяючи командам працювати більш злагоджено, мінімізуючи час між внесенням

змін у код та їх впровадженням у продакшн. Основні принципи DevOps, такі як автоматизація процесів, безперервна інтеграція (CI/CD), інфраструктура як код (IaC) та моніторинг, стали стандартами сучасного інженерного підходу.

Однак, незважаючи на очевидну важливість DevOps у реальній практиці, багато університетських навчальних програм все ще зосереджені переважно на традиційних дисциплінах, таких як алгоритми, структури даних, об'єктно-орієнтоване програмування, мережі та бази даних. DevOps же або викладається фрагментарно в межах курсів з хмарних обчислень чи системного адміністрування, або взагалі відсутній у навчальному процесі.

Це створює значний розрив між академічною підготовкою випускників і реальними потребами роботодавців. Компанії очікують, що молоді спеціалісти володітимуть хоча б базовими навичками DevOps, такими як робота з системами контролю версій (Git), автоматизація CI/CD-процесів (Jenkins, GitLab CI, GitHub Actions), управління контейнерами (Docker, Kubernetes), хмарні технології (AWS, Azure, GCP) та інші практики. Відсутність цих знань значно ускладнює адаптацію випускників до робочого середовища та зменшує їхню конкурентоспроможність на ринку праці.

З іншого боку, інтеграція DevOps у навчальний процес стикається з рядом викликів. По-перше, це швидка еволюція технологій та інструментів, що вимагає постійного оновлення навчальних матеріалів та підготовки викладачів. По-друге, DevOps вимагає міждисциплінарного підходу, який поєднує програмування, адміністрування, кібербезпеку та менеджмент, що ускладнює його впровадження в традиційну модульну структуру навчальних програм.

Аналіз останніх досліджень і публікацій. Останні роки DevOps привертає дедалі більше уваги як у науковій, так і у професійній літературі, що відображає його зростаюче значення у сфері інформаційних технологій [1]. Дослідники та практики вказують на те, що DevOps не лише підвищує швидкість та ефективність розробки програмного забезпечення, а й сприяє покращенню якості продуктів за рахунок автоматизації, моніторингу та безперервної інтеграції [2].

У роботах [3, 4] підкреслюють, що DevOps значно скорочує цикл розробки та впровадження програмного забезпечення завдяки використанню CI/CD, що дозволяє командам швидше доставляти оновлення без втрати якості. Дослідження впливу DevOps на процеси розробки показало, що використання автоматизованих засобів тестування, моніторингу та управління інфраструктурою зменшує кількість помилок та прискорює релізи [5]. Впровадження DevOps-підходу значно підвищує продуктивність розробницьких команд [6]. Дослідження [7] показало, що організації, які активно застосовують DevOps, мають швидший випуск нових функцій, вищу надійність програмного забезпечення та меншу ймовірність відмови систем після оновлення.

У сфері освіти питання інтеграції DevOps у навчальні програми досліджував Jones C. Стаття [8] спрямована на дослідження інтеграції концепції DevOps у навчальні програми з інженерії програмного забезпечення. Автор аргументує необхідність змін у традиційних підходах до викладання, оскільки сучасні методи розробки програмного забезпечення вимагають безперервної інтеграції, автоматизації та співпраці між командами розробки та експлуатації.

Стаття [9] досліджує впровадження підходу системного навчання (System-Based Learning, SBL) у викладання DevOps інженерії. Автори підкреслюють необхідність інтеграції системного мислення та інженерних принципів для глибшого розуміння процесу розробки програмного забезпечення, що є актуальним у сучасному швидкозмінному технологічному середовищі.

Стаття [10] представляє систематичний огляд літератури щодо сучасних методик викладання DevOps. Метою дослідження було виявлення та аналіз технік, які використовувалися для навчання DevOps протягом останніх п'яти років. Дослідження підкреслює важливість використання активних методик навчання у викладанні DevOps для забезпечення глибокого розуміння та практичних навичок студентів. Водночас, воно вказує на необхідність подолання існуючих викликів, щоб забезпечити ефективне та актуальне навчання у цій галузі.

Таким чином, огляд літератури підтверджує, що DevOps є важливим фактором підвищення ефективності розробки ПЗ, проте його викладання в університетах ще недостатньо розвинене. Це вказує на необхідність подальших досліджень щодо інтеграції DevOps у навчальні програми для IT-фахівців.

Мета дослідження: обґрунтування необхідності інтеграції методології DevOps у систему професійної підготовки фахівців з інформаційних технологій, виявлення основних проблем і суперечностей цього процесу та розроблення рекомендацій щодо ефективного впровадження DevOps-підходів у навчальні програми закладів вищої освіти з урахуванням вимог сучасного IT-ринку.

Методи дослідження. У дослідженні застосовано комплекс взаємодоповнювальних методів: аналіз і синтез наукових джерел для визначення теоретичних засад DevOps-підходу; узагальнення й систематизація для формулювання педагогічних умов інтеграції DevOps; а також елементи контент-аналізу та експертного оцінювання для визначення практичних тенденцій і проблем реалізації DevOps у підготовці IT-фахівців.

Виклад основного матеріалу дослідження. DevOps є однією з ключових методологій, що визначає сучасні підходи до розробки та експлуатації програмного забезпечення. Його основною метою є забезпечення швидкої, безперервної та стабільної розробки, тестування, доставки та експлуатації програмних продуктів. Завдяки інтеграції розробки (Development) та операційної діяльності (Operations), DevOps сприяє ефективному управлінню життєвим циклом програмного забезпечення, зменшенню ризиків та прискоренню впровадження інновацій. Рис. 1 ілюструє ключові аспекти DevOps, які допомагають організаціям досягати ефективності та стабільності у своїй IT-інфраструктурі.

Основні компоненти DevOps охоплюють автоматизацію процесів, культуру співпраці та моніторинг систем. Автоматизація є ключовим елементом, що мінімізує рутинні завдання, знижує ризик людських помилок і прискорює випуск оновлень. Безперервна інтеграція та розгортання забезпечує автоматичне тестування, перевірку коду та його впровадження у середовище розробки, тестування або продуктивне середовище. Використання таких інструментів, як Jenkins, GitHub Actions, GitLab CI/CD і CircleCI, дозволяє швидко виявляти та усувати помилки, покращуючи якість програмного забезпечення.

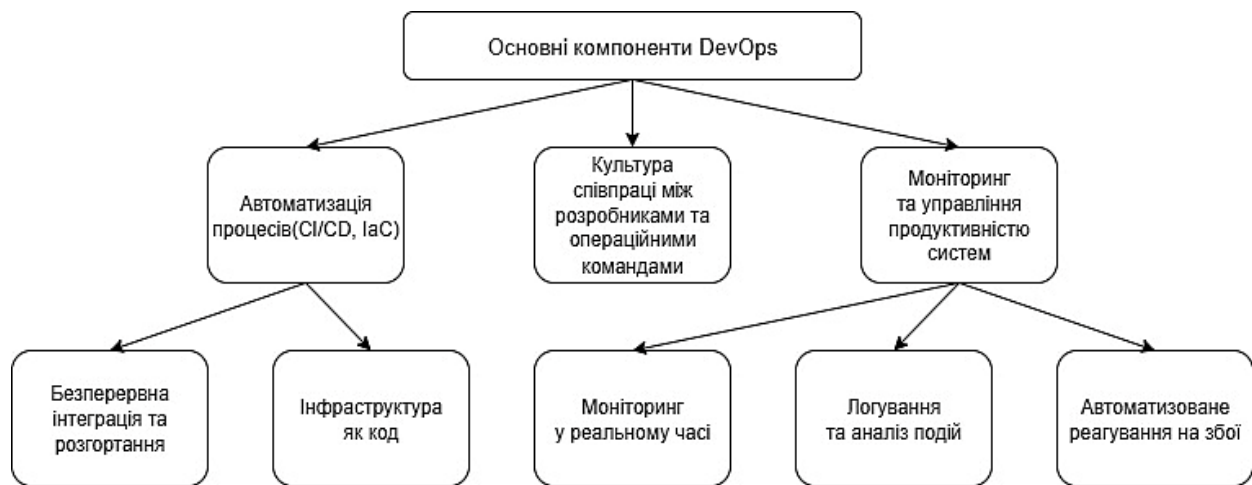


Рис. 1. Основні компоненти DevOps

Інфраструктура як код забезпечує керування інфраструктурою через конфігураційні файли, усуваючи потребу в ручному налаштуванні серверів. Інструменти на кшталт Terraform, Ansible і CloudFormation автоматизують розгортання серверів і хмарних сервісів, підвищуючи ефективність управління IT-інфраструктурою. DevOps також сприяє налагодженню тісної співпраці між розробниками та операційними командами, що допомагає швидше впроваджувати зміни та адаптуватися до вимог ринку. Постійний моніторинг і аналіз продуктивності дозволяє підтримувати стабільність і надійність програмних рішень.

DevOps передбачає зміну традиційного підходу до взаємодії між командами розробки та експлуатації. В умовах класичних підходів розробники зосереджувалися на створенні коду, а операційні команди займалися його розгортанням та підтримкою. Це часто призводило до конфліктів та затримок у випуску нових версій.

Завдяки DevOps ці команди працюють спільно, використовуючи принципи спільної відповідальності, безперервного зворотного зв'язку та автоматизованих процесів. Такі підходи, як ChatOps (використання месенджерів та автоматичних ботів для комунікації) та GitOps (керування інфраструктурою через систему контролю версій Git), дозволяють командам ефективніше взаємодіяти та швидко вирішувати проблеми.

Моніторинг та управління продуктивністю систем є невід'ємною частиною DevOps, забезпечуючи стабільну роботу програмного забезпечення та його інфраструктури. Постійний збір метрик, аналіз журналів подій і налаштування систем сповіщень дозволяють оперативно виявляти та усувати проблеми.

Моніторинг у реальному часі за допомогою інструментів, таких як Prometheus, Grafana та Datadog, забезпечує детальний аналіз продуктивності застосунків, використання ресурсів і можливих відмов. Логування та аналіз подій за допомогою ELK Stack (Elasticsearch, Logstash, Kibana) або Splunk дозволяє виявляти аномалії, аналізувати поведінку користувачів і запобігати потенційним збоєм.

Сучасні підходи включають автоматизоване реагування на збої, зокрема використання AIOps (Artificial Intelligence for IT Operations), що аналізує великі обсяги даних та прогнозує можливі відмови.

Завдяки впровадженню таких рішень DevOps-команди можуть не лише швидко усувати проблеми, а й запобігати їм ще до виникнення.

Таким чином, DevOps є не просто набором технологій, а комплексним підходом, який змінює принципи роботи IT-команд, підвищуючи швидкість, гнучкість та надійність програмних рішень. Включення DevOps до навчальних програм підготовки IT-фахівців є необхідним кроком для формування компетентних та затребуваних на ринку праці спеціалістів.

Попри зростаючу потребу в DevOps-фахівцях на ринку праці, впровадження цієї методології у навчальний процес залишається складним завданням для більшості університетів. Це пов'язано з низкою викликів, які стосуються як кадрових ресурсів, так і структурних аспектів освітніх програм. Основні проблеми інтеграції DevOps у вищу освіту показано на рис. 2.

Основною проблемою інтеграції DevOps у вищу освіту є нестача кваліфікованих викладачів із практичним досвідом у цій сфері. Викладання DevOps потребує знань не лише у програмуванні, а й у системному адмініструванні, хмарних обчисленнях, автоматизації інфраструктури та безперервній інтеграції. Багато викладачів мають переважно академічний досвід і недостатньо працювали з сучасними DevOps-інструментами, що обмежує можливості студентів у набутті практичних навичок.

Серед додаткових викликів – відсутність програм підвищення кваліфікації для викладачів, які б дозволили їм опанувати сучасні методики DevOps, а також обмежена співпраця університетів із комерційними компаніями, які могли б залучати практикуючих спеціалістів до навчального процесу.

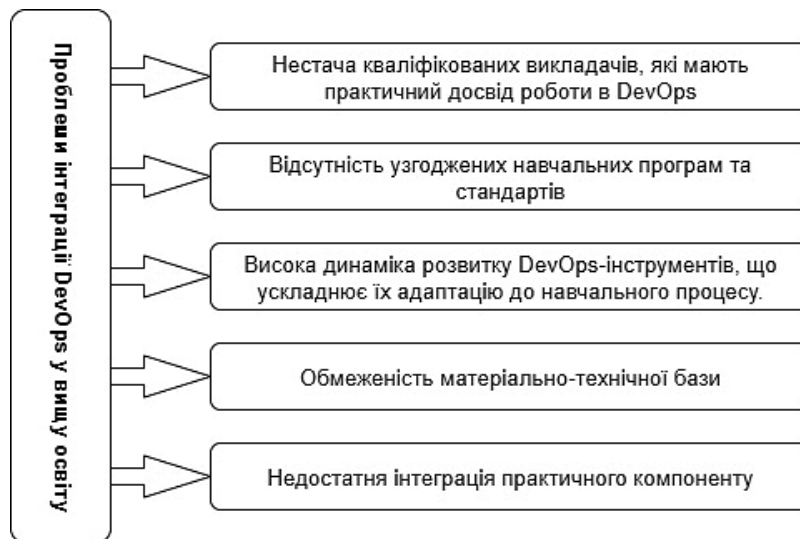


Рис. 2. Проблеми інтеграції DevOps у вищу освіту

Відсутність узгоджених навчальних програм та стандартів є однією з ключових проблем викладання DevOps у вищій освіті. На відміну від традиційних дисциплін, таких як алгоритми, бази даних чи операційні системи, DevOps не має чітко визначеного стандарту викладання, що призводить до його фрагментарного представлення у різних курсах.

Через відсутність єдиних навчальних стандартів розробка спеціалізованих курсів стає складнішою, а освітні установи змушені самостійно формувати програми, які мають відповідати як академічним вимогам, так і потребам ринку праці. Це створює труднощі в узгодженні навчального процесу з актуальними тенденціями у сфері DevOps.

Висока динаміка розвитку DevOps-інструментів створює значні виклики для їх адаптації в освітньому процесі. Нові технології, методології та інструменти для автоматизації CI/CD, контейнеризації та моніторингу з'являються з великою швидкістю, а оновлення можуть відбуватися кожні кілька місяців.

Через це університети часто використовують застарілі навчальні матеріали, оскільки процес оновлення програм потребує часу. Традиційна система навчальних курсів не дозволяє оперативно змінювати їх зміст, що ускладнює підготовку студентів відповідно до актуальних вимог індустрії.

Обмеженість матеріально-технічної бази є суттєвою перешкодою для впровадження DevOps у навчальний процес. Ця методологія вимагає використання сучасних інструментів, таких як Docker, Kubernetes, Terraform, Jenkins, GitLab CI/CD, Prometheus і Grafana, а також доступу до хмарних платформ, зокрема AWS, Azure та Google Cloud.

Не всі університети мають достатні серверні ресурси для забезпечення студентів необхідними навчальними середовищами. Використання комерційних хмарних сервісів потребує додаткового фінансування, яке не завжди є доступним для закладів освіти.

Недостатня інтеграція практичного компонента є значним викликом у викладанні DevOps. Ця методологія передбачає активне використання CI/CD-процесів, автоматизованого розгортання інфраструктури та управління контейнеризованими середовищами.

У багатьох університетах бракує лабораторних занять і проектів, що відтворюють реальні DevOps-процеси. Відсутність доступу до практичних кейсів та робочих сценаріїв обмежує можливість студентів отримати необхідні навички, що знижує їхню конкурентоспроможність на ринку праці.

Інтеграція DevOps у навчальні програми університетів є необхідною для підготовки конкурентоспроможних IT-фахівців, здатних працювати в умовах сучасного програмного забезпечення. Для успішного впровадження цієї методології у вищу освіту варто врахувати низку ключових аспектів (рис. 3).

Залучення практикуючих DevOps-інженерів до викладання може значно підвищити якість підготовки студентів. Співпраця з IT-компаніями сприятиме впровадженню гостьових лекцій і воркшопів від спеціалістів-практиків, організації менторських програм, у межах яких досвідчені фахівці допомагатимуть студентам у виконанні реальних завдань.

Крім того, спільні дослідницькі проекти між університетами та IT-компаніями, а також стажування студентів у реальних DevOps-процесах дадуть можливість отримати практичний досвід. Такий підхід забезпечить студентам актуальні знання та навички, які відповідатимуть сучасним вимогам ринку праці.

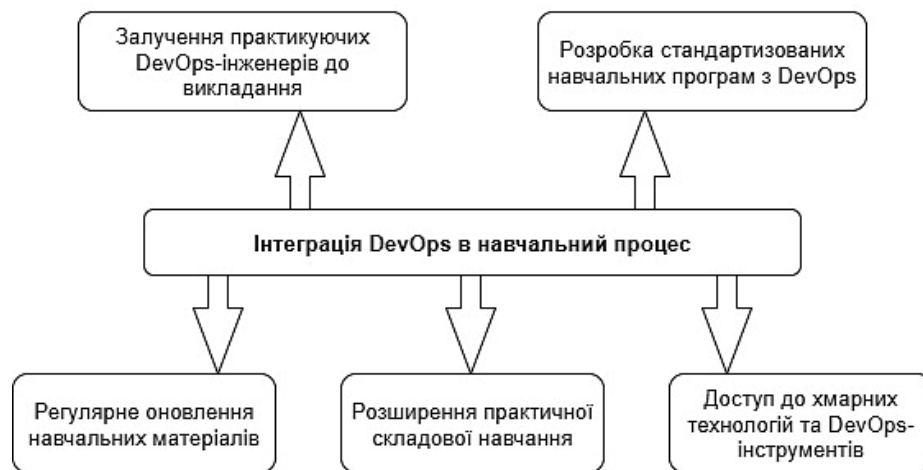


Рис. 3. Інтеграція DevOps у навчальний процес

Стандартизовані навчальні програми з DevOps повинні охоплювати ключові компоненти, що забезпечують комплексну підготовку студентів. Основи DevOps мають включати концепцію цього підходу, взаємодію між командами розробки та операційної підтримки, а також принципи безперервної інтеграції та розгортання.

Автоматизація процесів передбачає роботу з такими інструментами, як Jenkins, GitHub Actions і GitLab CI/CD, тоді як Infrastructure as Code базується на використанні Terraform, Ansible та AWS CloudFormation для керування інфраструктурою. Важливим аспектом є контейнеризація та оркестрація, де застосовуються Docker, Kubernetes і Helm, а також моніторинг та логування за допомогою Prometheus, Grafana та ELK Stack.

Практичні кейси повинні включати роботу з реальними інфраструктурними задачами та розгортання додатків у хмарних середовищах (AWS, Azure, GCP). Впровадження таких навчальних стандартів дозволить зробити DevOps обов'язковою складовою освітніх програм для IT-спеціальностей.

Регулярне оновлення навчальних матеріалів є критично важливим через стрімкий розвиток DevOps-інструментів. Для забезпечення актуальності курсів необхідно періодично переглядати навчальні програми та адаптувати їх відповідно до змін у галузі.

Включення матеріалів від авторитетних DevOps-спільнот, таких як CNCF, Linux Foundation та DevOps Institute, сприятиме підвищенню якості викладання. Додатково варто використовувати відкриті освітні ресурси, зокрема онлайн-курси від AWS, Google Cloud та Microsoft Azure, що допоможуть студентам освоїти найсучасніші підходи та технології.

Розширення практичної складової навчання є ключовим фактором ефективного засвоєння DevOps. Теоретичні знання мають доповнюватися реальним досвідом роботи з актуальними інструментами та технологіями.

Для цього необхідно впровадити лабораторні роботи на базі сучасних DevOps-платформ, що дозволить студентам освоювати процеси автоматизації та управління інфраструктурою. Розробка курсових і дипломних проєктів, спрямованих на оптимізацію процесів розробки та експлуатації програмного забезпечення, сприятиме закріпленню отриманих навичок. Використання підходу "навчання через практику" (learning by doing) допоможе студентам виконувати реальні завдання, наближені до умов сучасного IT-середовища.

Доступ до хмарних технологій та DevOps-інструментів є важливою умовою для ефективного навчання студентів. Для цього необхідно забезпечити можливість роботи з реальними інфраструктурними рішеннями через безкоштовні освітні програми, такі як AWS Educate, Google Cloud for Education та Microsoft Learn for Students, які надають ресурси для навчання.

Співпраця університетів із провайдерами хмарних сервісів дозволить отримувати гранти або безкоштовні акаунти для студентів, що значно розширить можливості практичного навчання. Також доцільно використовувати віртуалізовані середовища, створюючи внутрішні навчальні сервери для роботи з CI/CD-конвеєрами, Docker-контейнерами та Kubernetes-кластерами. Це дасть студентам змогу набувати практичних навичок у максимально наближених до реальних умовах.

Очікувані результати впровадження DevOps у навчальні програми включають підвищення рівня підготовки студентів до реальних умов роботи в IT-компаніях, скорочення розриву між академічними знаннями та вимогами ринку праці, а також покращення співпраці університетів із бізнесом у сфері IT.

Крім того, забезпечення студентів доступом до сучасних інструментів DevOps сприятиме підвищенню їх конкурентоспроможності на ринку праці. Інтеграція DevOps у навчальний процес є стратегічно важливим кроком для підготовки IT-фахівців нового покоління, здатних ефективно працювати у високодинамічному середовищі розробки програмного забезпечення.

Висновки і перспективи подальших досліджень. Проведене дослідження підтвердило, що впровадження DevOps у систему професійної підготовки IT-фахівців є стратегічно важливим напрямом модернізації вищої освіти. DevOps виступає не лише як технологічна концепція, а як педагогічна рамка, що формує у студентів цілісне розуміння життєвого циклу програмного забезпечення, розвиває міждисциплінарні компетентності, сприяє командній роботі та критичному мисленню. Визначено основні бар'єри інтеграції DevOps у навчальний процес – кадрові, організаційні та технологічні, а також окреслено шляхи їх подолання через співпрацю з IT-компаніями, розробку єдиних навчальних стандартів і створення практикоорієнтованих DevOps-лабораторій.

Подальші наукові розвідки доцільно зосередити на розробленні моделі DevOps-орієнтованого освітнього середовища, оцінюванні рівня сформованості DevOps-компетентностей у студентів, створенні методичних рекомендацій для викладачів та впровадженні експериментальних курсів, що поєднують елементи автоматизації, CI/CD, контейнеризації й хмарних технологій у навчальному процесі. Це сприятиме формуванню нового покоління IT-фахівців, здатних ефективно працювати в умовах цифрової трансформації.

Конфлікт інтересів. Автори заявляють про відсутність конфлікту інтересів.

Джерела фінансування. Дослідження не отримувало зовнішнього фінансування.

Доступність даних. Це дослідження не передбачало використання окремих наборів даних.

Використання засобів штучного інтелекту (ШІ). Під час підготовки цієї роботи автори не використовували інструменти штучного інтелекту.

Список використаних джерел / References

1. Amaro, R., Pereira, R., & Mira da Silva, M. (2024). DevOps metrics and KPIs: a multivocal literature review. *ACM Computing Surveys*, 56(9), 1-41. <https://doi.org/10.1145/365250>
2. Kumar, A., Nadeem, M., & Shameem, M. (2025). A Systematic Literature Review for Investigating DevOps Metrics to Implement in Software Development Organizations. *Journal of Software: Evolution and Process*, 37(1), e2733. <https://doi.org/10.1002/smr.2733>
3. Nareesh, E., Murthy, S. V. N., Pareek, P. K., Reddy, K. T., Darshan, S. S., & Kumar, B. P. (2024, April). DevOps Life Cycle Implementation on Real Life Scenarios. In *2024 International Conference on Knowledge Engineering and Communication Systems (ICKECS)* (Vol. 1, pp. 1269-1271). IEEE. <https://doi.org/10.1109/ICKECS61492.2024.10617200>
4. Saini, L. M. (2024, July). The Impact of Devops on Software Quality. In *2024 Third International Conference on Smart Technologies and Systems for Next Generation Computing (ICSTSN)* (pp. 1-6). IEEE. <https://doi.org/10.1109/ICSTSN61422.2024.10670849>
5. Azad, N., & Hyrnsalmi, S. (2023). DevOps critical success factors—A systematic literature review. *Information and Software Technology*, 157, 107150. <https://doi.org/10.1016/j.infsof.2023.107150>
6. Silva, M. A., Faustino, J. P., Pereira, R., & Mira da Silva, M. (2018). Productivity gains of DevOps adoption in an IT team: a case study. <http://aisel.aisnet.org/isd2014/proceedings2018/ISDevelopment/8>

7. Muñoz, M., & Rodríguez, M. N. (2024). A guidance to implement or reinforce a DevOps approach in organizations: A case study. *Journal of Software: Evolution and Process*, 36(3), e2342. <https://doi.org/10.1002/smr.2342>
8. Jones, C. (2018, March). A proposal for integrating DevOps into software engineering curricula. In *International Workshop on Software Engineering Aspects of Continuous Development and New Paradigms of Software Production and Deployment* (pp. 33-47). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-030-06019-0_3
9. Zarour, M., Akour, M., & Alenezi, M. (2024). Enhancing DevOps Engineering Education Through System-Based Learning Approach. *Open Education Studies*, 6(1), 20240012. <https://doi.org/10.1515/edu-2024-0012>
10. Garcia, P. S., Ferreira, J., Gonçalves, M., Carneiro, T., Figueiredo, E., & Pereira, I. M. (2024). Current DevOps Teaching Techniques: A Systematic Literature Review. *Simpósio Brasileiro de Engenharia de Software (SBES)*, 389-398. <https://doi.org/10.5753/sbes.2024.3503>

| Матеріал надійшов до редакції: 08.09.2025 р. | Прийнято до друку: 25.10.2025 р. | Опубліковано: 28.11.2025 р. |

