

Кубік М. А.
здобувач третього освітньо-наукового рівня вищої освіти
Тернопільський національний педагогічний університет ім. В. Гнатюка
kub@tnpu.edu.ua
Науковий керівник:
Мартинюк С. В.
кандидат фізико-математичних наук,
доцент кафедри інформатики та методики її навчання,
Тернопільський національний педагогічний університет ім. В. Гнатюка
sergmart65@tnpu.edu.ua

ВИКОРИСТАННЯ ТЕХНОЛОГІЇ DOCKER В ОСВІТНЬОМУ ПРОЦЕСІ

Швидкий розвиток сучасних технологій вимагає від освітніх закладів постійного адаптування своїх методів навчання до вимог часу. Враховуючи необхідність підготовки студентів до викликів сучасного інформаційного суспільства, освітні заклади активно розвивають нові підходи до навчання, в тому числі використання інноваційних технологій. Однією з таких інноваційних технологій є Docker – платформа для розробки, доставки та запуску програмного забезпечення з використанням контейнеризації. Вона дозволяє створювати і використовувати ізольовані віртуальні середовища, що спрощує налаштування та розгортання програм на будь-якій платформі. Ми розглянемо, як використання технології Docker може підвищити ефективність та зручність налаштування середовищ для навчання та розробки програмного забезпечення. Docker забезпечує швидке створення та використання контейнерів з програмним забезпеченням та його залежностями, що дозволяє уникнути складнощів з налаштуванням середовища на кожному окремому пристрої [1]. Такий підхід сприяє зниженню часу, необхідного для підготовки до занять та проведення практичних вправ, та забезпечує єдність середовища для всіх учасників навчального процесу.

Docker – це платформа для розробки, доставки та запуску програмного забезпечення за допомогою контейнеризації. Вона дозволяє упаковувати програми та їх залежності в контейнери, які можна легко переносити та запускати на будь-якому пристрої, де встановлений Docker.

Використання технології Docker суттєво спрощує процес налаштування середовища для навчання та розробки програмного забезпечення. Завдяки Docker, користувачі можуть швидко створювати та встановлювати необхідне середовище без необхідності вручну налаштовувати кожен складову окремо. Це особливо корисно для студентів та викладачів, які зосереджені на розвитку програм та концентрації на навчанні, а не на рутинних завданнях налаштування інфраструктури.

За допомогою Docker можна легко створювати контейнери з встановленими необхідними програмами, бібліотеками та середовищами виконання. Це означає, що користувачам не потрібно витрачати час на пошук, встановлення та налаштування кожного компонента окремо. Docker забезпечує єдність та стандартизацію середовищ, що дозволяє уникнути супутніх проблем з несумісністю або відмовами через неправильні налаштування.

Ця можливість створювати готові до використання контейнери значно прискорює процес підготовки до навчальних занять або розробки проектів [2]. Крім того, вона забезпечує консистентність середовища між різними користувачами та пристроями, що сприяє спільній роботі та спільному навчанню. Такий підхід до налаштування середовища забезпечує ефективність навчання та розробки програмного забезпечення, оскільки дозволяє зосередитися на важливих завданнях та максимально використовувати час.

Стандартизація середовища є ключовою складовою ефективного навчання та розробки програмного забезпечення. Docker дозволяє створювати стандартні образи контейнерів, які можна використовувати у різних групах або на різних курсах. Це забезпечує єдність середовища для всіх учасників навчального процесу, що сприяє зручності у вивченні та

спільній роботі. Використання стандартизованих образів контейнерів дозволяє уникнути різниці у середовищах між різними групами студентів або курсами, забезпечуючи однакові умови навчання для всіх.

Крім того, мобільність Docker є надзвичайно корисною функцією для учнів та викладачів. Контейнери Docker легко переносяться між різними платформами та пристроями, що дозволяє працювати зі своїми проектами у будь-якому місці та на будь-якому пристрої, де є підтримка Docker. Це дає можливість учням та викладачам мати доступ до своїх ресурсів та проектів навіть у віддалених місцях або під час подорожей, забезпечуючи постійність та неперервність навчального процесу.

Також важливою перевагою Docker є ефективне використання ресурсів. Docker дозволяє створювати та запускати контейнери на віртуальних машинах або фізичних серверах, що дозволяє ефективно використовувати обчислювальні ресурси. Це дозволяє оптимізувати використання обладнання та зменшує витрати на обслуговування та підтримку інфраструктури. Використання Docker сприяє оптимальному розподілу ресурсів та забезпечує швидке розгортання та масштабування навчальних та розробних середовищ. Таким чином, Docker відіграє важливу роль у створенні ефективних та зручних середовищ для навчання та розробки програмного забезпечення.

Приклади використання Docker в освітньому процесі:

– **Навчання програмуванню:** Docker може бути використаний для створення стандартних середовищ для навчання різним мовам програмування. Учні можуть отримати доступ до вже налаштованих контейнерів з необхідними інструментами та ресурсами для вивчення конкретних мов програмування.

– **Розробка веб-додатків:** Docker дозволяє розробникам легко налаштувати локальне середовище для розробки веб-додатків, включаючи веб-сервери, бази даних та інші компоненти. Це допомагає уникнути проблем з різницею у середовищах між розробкою та продакшеном.

– **Лабораторні роботи з комп'ютерних наук:** Docker може бути використаний для створення віртуальних лабораторних середовищ, де учні можуть виконувати практичні завдання та експерименти з різними технологіями та програмним забезпеченням [3].

Використання технології Docker в освітньому процесі виявляється дуже корисним інструментом для підвищення ефективності та зручності налаштування середовищ для навчання та розробки програмного забезпечення. Вона дозволяє створювати стандартизовані та переносні середовища, що полегшує навчання та сприяє зростанню компетентностей учасників [4].

Список використаних джерел:

1. DIXON, Bryan. Simplifying teaching continuous integration and continuous deployment with hands-on application in a web development course. *Journal of Computing Sciences in Colleges*, 2020, 35.10: 15-20.
2. MALAN, David J. Standardizing Students' Programming Environments with Docker Containers: Using Visual Studio Code in the Cloud with GitHub Codespaces. In: *Proceedings of the 27th ACM Conference on on Innovation and Technology in Computer Science Education Vol. 2*. 2022. p. 599-600.
3. CHAE, Minsu; HAN, Sangwook; LEE, HwaMin. Docker-based cloud system for computer programming labs. In: *2019 14th International Conference on Computer Science & Education (ICCSE)*. IEEE, 2019. p. 622-626.
4. ZHANG, Yanyan; RAN, Xiangjin. Research on Experimental Teaching Platform of Computer Courses based on Docker. In: *2020 International Conference on Robots & Intelligent System (ICRIS)*. IEEE, 2020. p. 529-532.