



ПЕДАГОГІЧНА АКАДЕМІЯ:
НАУКОВІ ЗАПИСКИ

Інформаційно-комунікаційні технології в освіті

УДК 004.4'2:378.147:004.272.26:004.032.26

DOI <https://doi.org/10.5281/zenodo.15269510>

**Впровадження віртуалізованого середовища для лабораторних робіт з
паралельного програмування**

Сіциліцин Юрій Олександрович

PhD, старший викладач кафедри інформатики і кібернетики
Мелітопольського державного педагогічного університету імені Богдана
Хмельницького, 69000, м. Запоріжжя, вул. Наукового містечка, 59, Україна,
Sicylicyn_Yurij@mspu.edu.ua, <https://orcid.org/0000-0002-3888-5575>

Лубко Дмитро Вікторович

Кандидат технічних наук, доцент кафедри комп'ютерних наук
Таврійського державного агротехнологічного університету імені Дмитра
Моторного, 69600, м. Запоріжжя, вул. Жуковського, 66, Україна,
di75ma@gmail.com, <https://orcid.org/0000-0002-2506-4145>

Прийнято: 10.04.2025 | Опубліковано: 22.04.2025

Анотація. Метою даного дослідження є розробка та впровадження віртуалізованого середовища для виконання лабораторних робіт з паралельного програмування в умовах дистанційного навчання. Аналіз літератури підтвердив ефективність розробки віртуальних лабораторій для навчання здобувачів. Також із аналізу літератури можна зробити висновок про недостатність



існуючих рішень віртуальних лабораторій для використання їх у паралельному програмуванні. Актуальність дослідження зумовлена потребою забезпечити студентів рівним доступом до обчислювальних ресурсів без прив'язки до фізичного обладнання. Методологія дослідження включає аналіз сучасних технологій віртуалізації, порівняння п'яти альтернативних рішень, розробку критеріїв оцінки ефективності та обґрунтований вибір оптимального варіанту. У статті докладно описано переваги обраного рішення — віртуальних машин з доступом через SSH та JupyterHub — з точки зору доступності, продуктивності, адміністрування та масштабованості. Представлено архітектуру системи, що базується на платформі Proxmo, а також детально описано можливості використання бібліотек OpenMP та MPI. Визначено ключові переваги впровадженого підходу, зокрема уніфікацію навчального середовища, спрощення технічної підтримки та зменшення вірогідності помилок під час виконання завдань. Проведено попередню оцінку результативності, заплановано експериментальне впровадження за участю студентів. Очікується, що впроваджене середовище забезпечить не лише технічну сумісність і продуктивність, але й позитивно вплине на мотивацію та успішність студентів, що є важливим чинником у розвитку дистанційної освіти. У висновках акцентовано на перспективах масштабування розробленого рішення, можливості його використання в інших технічних дисциплінах, а також на необхідності подальших досліджень у напрямку автоматизації, розподілу ресурсів і моніторингу навчального прогресу.

Ключові слова: *віртуалізація, паралельне програмування, дистанційне навчання, JupyterHub, Proxmo, SSH-доступ, лабораторні роботи, OpenMP, MPI, ІКТ в освіті.*



ПЕДАГОГІЧНА АКАДЕМІЯ:
НАУКОВІ ЗАПИСКИ

Implementation of a Virtualized Environment for Laboratory Work in Parallel Programming

Yurii Sitsylitsyn

PhD, Senior Lecturer at the Department of Informatics and Cybernetics
Bogdan Khmelnytsky Melitopol State Pedagogical University, 69000, Zaporizhzhia,
Naukovogo Mistechka St., 59, Ukraine, Sicylicyn_Yurij@msspu.edu.ua,
<https://orcid.org/0000-0002-3888-5575>

Lubko Dmytro

Candidate of Technical sciences, Associate professor Department of Computer
Science, Dmytro Motorny Tavria State Agrotechnological University, 69600,
Zaporizhzhia, str. Zhukovsky, 66, Ukraine, di75ma@gmail.com,
<https://orcid.org/0000-0002-2506-4145>

***Abstract.** The aim of this study is to design and implement a virtualized environment for conducting laboratory work in parallel programming under conditions of distance learning. A review of the literature confirms the effectiveness of developing virtual laboratories for student education. Furthermore, the analysis reveals a lack of existing virtual lab solutions suitable for use in parallel programming education. The relevance of the research is driven by the need to provide students with equal access to computational resources without reliance on personal hardware. The research methodology includes an analysis of current virtualization technologies, comparison of five alternative solutions, development of evaluation criteria, and a reasoned selection of the most appropriate option.*

This paper describes in detail the advantages of the selected solution — virtual machines accessed via SSH and JupyterHub — in terms of accessibility, performance,



manageability, and scalability. The architecture of the system, based on the Proxmox platform, is presented along with detailed discussion of its support for OpenMP and MPI libraries. Key benefits of the proposed approach are identified, including the unification of the learning environment, simplified technical support, and reduced probability of user-side errors. A preliminary performance assessment has been conducted, and experimental implementation involving students is planned. It is expected that the virtual environment will ensure not only technical compatibility and high performance but also have a positive impact on student motivation and academic success — factors that are crucial in the context of remote education.

The conclusions highlight the scalability prospects of the developed solution, its applicability to other technical disciplines, and the need for further research in the areas of deployment automation, resource distribution, and learning progress monitoring.

Keywords: *virtualization, parallel programming, distance learning, JupyterHub, Proxmox, SSH access, laboratory work, OpenMP, MPI, ICT in education.*

Постановка проблеми у загальному вигляді та її зв'язок з важливими науковими чи практичними завданнями. Розвиток інформаційних технологій суттєво змінив підходи до організації навчального процесу. Дистанційна освіта, яка стала особливо актуальною в останні роки, створює нові виклики для забезпечення якісного навчання. Особливо це стосується курсів, пов'язаних з паралельним програмуванням, оскільки такі дисципліни потребують спеціалізованого апаратного та програмного забезпечення.

Основна проблема полягає в тому, що більшість студентів не мають доступу до високопродуктивних обчислювальних ресурсів. Виконання лабораторних робіт з паралельного програмування потребує багатоядерних процесорів, підтримки багатопотокової обробки та можливості працювати з



бібліотеками, такими як OpenMP, MPI та CUDA. Проте не всі студенти мають персональні комп'ютери з достатньою обчислювальною потужністю, що ускладнює виконання завдань та обмежує якість навчального процесу.

В умовах дистанційного навчання виникає додаткова проблема – відсутність єдиного середовища для виконання лабораторних робіт. Використання різних операційних систем та конфігурацій обладнання ускладнює налаштування та тестування програм студентами. Відсутність стандартного середовища призводить до несумісності коду, проблем із запуском програм та складності у виправленні помилок під час навчального процесу.

З огляду на ці виклики, було запропоновано створення централізованого віртуалізованого середовища для навчання. Сервер на базі Proxmox дозволяє розгорнути віртуальні машини для кожного студента або групи студентів, забезпечуючи їм рівні можливості для виконання лабораторних завдань. Віртуальні машини надають доступ до необхідного програмного забезпечення, підтримують необхідні бібліотеки та забезпечують єдине середовище для роботи з паралельними алгоритмами.

Наукове та практичне значення цього підходу полягає у можливості масштабування навчального процесу без прив'язки до фізичного обладнання студентів. Таке рішення сприяє підвищенню якості освіти, дозволяючи студентам отримати практичні навички роботи з паралельними обчисленнями без необхідності встановлення та налаштування складного програмного забезпечення на власних пристроях. Крім того, централізоване керування віртуальним середовищем дозволяє викладачам контролювати процес навчання, оперативно надавати технічну підтримку та аналізувати ефективність виконання лабораторних завдань.

Аналіз останніх досліджень і публікацій. Питання використання віртуалізованих середовищ та хмарних обчислень для навчання паралельного



програмування активно досліджується в сучасній науковій літературі. В останні роки спостерігається підвищений інтерес до застосування таких технологій у навчальному процесі, особливо в умовах дистанційного навчання. Дослідники Pastor та інші [1] у статті показують свій досвід використання хмарних платформ для організації навчальних лабораторій. Автори відзначають, що використання віртуальних машин дозволяє ефективно ізолювати середовище кожного студента та забезпечити стабільність обчислювальних ресурсів. У роботі [2] представлена віртуальна лабораторія для виконання здобувачами робіт з кібербезпеки, яка заснована на технології віртуалізації Linux Docker, та дозволяє засновникам створювати послідовні реалістичні сценарії з меншими вимогами до конфігурації обладнання. Систематичний огляд літератури, який провели дослідники Jürgen Anders та Guerra Ramos [3] виявили, що на даний момент існує великий обсяг роботи над технологією лабораторії як послуги (LaaS), яку можна використовувати для розробки віддалених або віртуальних лабораторій у різних сферах інженерної освіти. Дослідники з політехнічного університету Гонконгу розробили Віртуальну Лабораторну Платформу [4] як безпечну та масштабовану платформу віртуальної лабораторії з використанням контейнерної технології та хмарної технології з відкритим кодом. Викладачі можуть налаштовувати та розгортати персоналізовані лабораторні середовища в хмарі, а студенти можуть використовувати веб-браузери, щоб отримати доступ до ізольованого програмного середовища. Лабораторія працює з такими технологіями як OpenVSCode, Docker, JupyterLab, Wekas and Orange Data mining tools. У статті [5] розглядаються сформовані дослідниками принципи проектування хмарного освітнього середовища для університетів із ефектом занурення. Ці принципи включають відкритість і доступність, персоніфікацію і адаптивність, інноваційність і практичну спрямованість технологій, а також інтеграцію традиційних і хмарних інструментів навчання. У статті [6] автори роблять огляд



декількох міжнародних досліджень у області лабораторій на основі Інтернету речей та їх вплив на академічну успішність студентів у вищій освіті. Автори вважають, що інтеграція Інтернету речей у вищих навчальних закладах покращує успішність студентів, оскільки дозволяє їм виконувати автентичні завдання та досвід практичного й активного навчання. Нове контейнеризоване паралельне обчислювальне середовище з використанням Docker показали автори статті «HPC-in-Containers: A Containerized Parallel Environment for Parallel Programming Learning Using Docker» [7]. Воно створене для полегшення вивчення концепцій паралельного програмування, де користувачам не потрібно розгорнути багатокомп'ютерну інфраструктуру. Ще одне віртуальне рішення для розробки паралельних програм пропонують науковці Kaler T. et al. [8]. Вони пропонують розроблену ними платформу онлайн-програмування, яка спрямована на покращення доступності освіти з розробки програмного забезпечення. Платформа Speedcode підтримує паралельне програмування за допомогою OpenCilk, платформи для паралельних обчислень із відкритим вихідним кодом.

Загалом, сучасні дослідження підтверджують ефективність використання віртуалізованих та хмарних середовищ для навчання. Але з огляду можна побачити невелику кількість віртуальних середовищ саме для паралельного програмування. А ті що існують орієнтовані на Jupyter Notebook у сполученні з контейнерним середовищем Docker. Використання Proxmox у поєднанні з віртуальними машинами, Jupyter Notebook та JupyterHub узгоджується з сучасними тенденціями у сфері освітніх технологій, дозволяючи студентам отримати практичний досвід без необхідності мати потужне власне обладнання.

Виділення невирішених раніше частин загальної проблеми. Попри значний прогрес у впровадженні віртуалізованих середовищ для навчання паралельного програмування, залишаються невирішені аспекти, що потребують подальших досліджень. Однією з таких проблем є ефективність використання



серверних ресурсів при великій кількості студентів. Хоча віртуальні машини дозволяють створити ізольовані середовища, їх масштабованість потребує подальшого аналізу та оптимізації.

Ще однією невирішеною проблемою є забезпечення високої продуктивності під час виконання обчислювальних задач. Використання JupyterHub та віддаленого доступу через SSH значно спрощує роботу студентів, однак питання ефективного розподілу ресурсів між паралельними процесами та мінімізації затримок залишається відкритим. Подальші дослідження повинні бути спрямовані на оцінку ефективності навчання у порівнянні з традиційними методами.

Формулювання цілей статті. У даному дослідженні основна увага зосереджена на розробці та впровадженні ефективного віртуалізованого середовища для організації лабораторних занять з паралельного програмування в умовах дистанційного навчання. Очікується, що запропоноване рішення забезпечить не лише технічну уніфікацію та рівний доступ до обчислювальних ресурсів, а й підвищить якість навчального процесу за рахунок спрощення технічної підтримки, покращення продуктивності виконання завдань і зменшення кількості помилок з боку студентів. Також передбачається, що впровадження цієї інфраструктури сприятиме зростанню мотивації студентів та дозволить викладачам більш ефективно контролювати та оцінювати навчальний прогрес.

Для досягнення цієї мети поставлено такі завдання:

- Провести аналіз сучасних технологій віртуалізації та їхнього застосування у навчальному процесі.
- Розробити архітектуру віртуалізованого середовища на базі Proxmox із використанням віртуальних машин, Jupyter Notebook та JupyterHub.



- Визначити оптимальні параметри конфігурації віртуальних машин для забезпечення ефективного навчання паралельному програмуванню.
- Розглянути перспективи подальшого розвитку та оптимізації даного підходу.

Очікувані результати цього дослідження спрямовані на підвищення якості дистанційного навчання та створення доступного середовища для студентів, які вивчають паралельне програмування.

Виклад основного матеріалу дослідження з повним обґрунтуванням здобутих наукових результатів. Для визначення оптимального підходу до створення віртуалізованого середовища для лабораторних робіт з паралельного програмування було розглянуто кілька варіантів [9]:

Варіант 0. Використання власного обладнання студентів. У цьому варіанті кожен студент виконує лабораторні роботи на власному комп'ютері. Це підхід, що не потребує додаткових ресурсів з боку навчального закладу, але має значні недоліки. Насамперед, у студентів можуть бути різні конфігурації комп'ютерів, що призводить до несумісності програмного забезпечення, складності у налаштуванні середовища та відсутності можливості для тестування реальних паралельних обчислень. Крім того, продуктивність сильно варіюється, що ускладнює навчальний процес.

Варіант 1. Виділений сервер для НРС. Використання окремого високопродуктивного сервера для навчальних лабораторій є класичним підходом. В даному випадку створюється одна потужна віртуальна машина або контейнер LXC, до якого студенти підключаються віддалено. Переваги цього підходу полягають у високій продуктивності, можливості централізованого адміністрування та налаштування єдиного середовища. Проте цей варіант має обмежену масштабованість – при великій кількості студентів продуктивність сервера може значно знижуватися.



Варіант 2. Хмарне середовище для студентів. В цьому підході кожному студенту або групі студентів виділяється окрема віртуальна машина у хмарному середовищі. Це рішення дозволяє забезпечити ізоляцію робочого середовища та спростити контроль за навчальним процесом. Основні переваги включають масштабованість, зручність налаштування та рівний доступ до обчислювальних ресурсів. Водночас, використання хмарних платформ може спричинити додаткові фінансові витрати та залежність від стабільного інтернет-з'єднання.

Варіант 3. Контейнеризація з Docker/LXC. Використання контейнерів дозволяє швидко розгорнути стандартизоване середовище з необхідним набором бібліотек і програмного забезпечення. Контейнери споживають менше ресурсів, ніж віртуальні машини, що робить цей підхід ефективним для масштабованих рішень. Проте контейнеризація не завжди дозволяє ізолювати ресурси на рівні ядра операційної системи, що може викликати проблеми з безпекою та продуктивністю при виконанні ресурсомістких завдань.

Варіант 4. Kubernetes + JupyterHub. Використання Kubernetes у поєднанні з JupyterHub є одним із найбільш масштабованих підходів для організації дистанційного навчання. Kubernetes дозволяє автоматично розгорнути, масштабувати та управляти контейнерами, а JupyterHub забезпечує інтерактивне середовище для роботи студентів. Це рішення ідеально підходить для великих груп студентів, проте вимагає складного адміністрування та високих ресурсів для розгортання.

Таким чином, кожен варіант має свої переваги та недоліки, що враховуються при виборі оптимального рішення.

Вибір оптимального середовища для виконання лабораторних робіт з паралельного програмування повинен базуватися на об'єктивних критеріях, які визначають його ефективність, доступність та придатність для використання у навчальному процесі. Основні критерії оцінки ґрунтуються на сучасних



дослідженнях у галузі комп'ютерних наук та освітніх технологій [10]. Розглянемо їх докладніше. По-перше це простота доступу. Важливо, щоб студенти могли легко підключатися до середовища та працювати з ним без додаткових технічних налаштувань. Дослідження Mathur et al. [11] підтверджують, що простота інтерфейсу позитивно впливає на залученість студентів до лабораторних робіт. По-друге це вимоги до обладнання. Оскільки не всі студенти мають доступ до потужного апаратного забезпечення, рішення повинно бути оптимізоване для роботи на мінімальних ресурсах. Mathur et al. [11] наголошують, що низькі вимоги до обладнання підвищують рівні можливості для студентів. Далі можна визначити гнучкість та масштабованість. Вибране середовище має легко адаптуватися до зміни кількості користувачів і підтримувати зростання обчислювального навантаження. Биков В. та інші [12] демонструють, що масштабовані рішення значно покращують стабільність обчислювальних середовищ у навчальному процесі. Важливим фактором є зручність централізованого управління середовищем, оскільки складність адміністрування може значно впливати на доступність ресурсу для студентів. Segrelles J. D. [13] відзначають, що спрощене адміністрування сприяє ефективнішому використанню обчислювальних ресурсів у навчальних закладах. Виконання паралельних обчислень вимагає високої швидкодії та мінімізації затримок у роботі програм. Valigodugula V. V. et al. [14] доводять, що продуктивність віртуалізованих середовищ безпосередньо впливає на якість навчального процесу. Простота розгортання на сервері. Час та складність початкового налаштування системи є критичними факторами для її ефективного впровадження. Дослідження Heldens, S.J. [15] зазначає, що автоматизовані рішення для розгортання значно скорочують навантаження на адміністраторів. Довгострокова підтримка системи повинна бути простою та вимагати мінімального втручання з боку адміністраторів. Wang et al. (2024) наголошують,

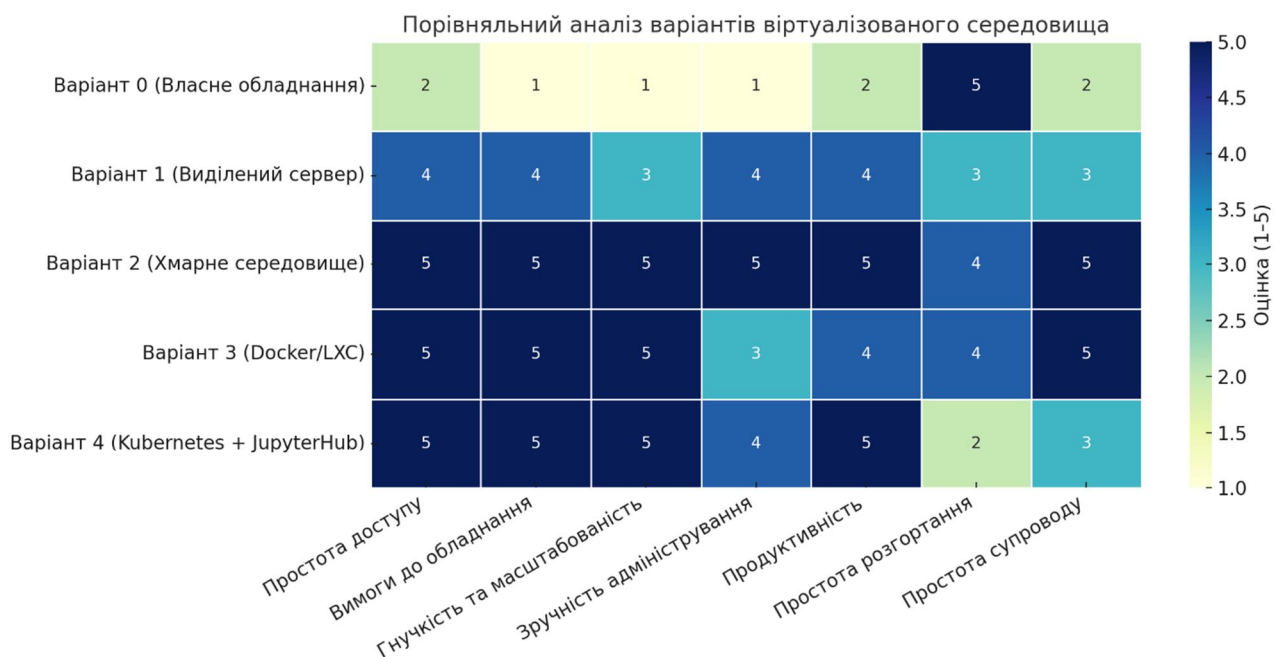


що рішення з низькими вимогами до обслуговування є більш стабільними та ефективними у довгостроковій перспективі.

Для оцінки кожного з варіантів було проведено аналіз на основі визначених критеріїв. Результати наведені на рисунку 1.

Рисунок 1

Аналіз варіантів впровадження віртуалізованого середовища



Аналіз показав, що найбільшу сумарну оцінку (34 з можливих 35 балів) отримав Варіант 2 — хмарне середовище з використанням віртуальних машин, SSH-доступу та JupyterHub. Це рішення забезпечує:

- максимальну доступність для студентів завдяки простому інтерфейсу;
- низькі вимоги до їхнього обладнання, оскільки всі обчислення виконуються на сервері;
- високу гнучкість і масштабованість, що дозволяє ефективно працювати з великими групами;
- зручне централізоване адміністрування та підтримку, що суттєво знижує навантаження на викладачів і технічний персонал;



- продуктивність, достатню для виконання реальних паралельних обчислень;
- швидке розгортання і централізоване оновлення програмного забезпечення.
- У поєднанні ці характеристики роблять обране рішення найбільш збалансованим за критеріями ефективності, доступності та підтримки в умовах дистанційного навчання.

Використання SSH та JupyterHub забезпечує зручний доступ до середовища без необхідності встановлення додаткового програмного забезпечення. Студенти можуть працювати через стандартні SSH-клієнти та веб-інтерфейс JupyterHub, що значно спрощує навчальний процес. Низькі вимоги до обладнання. Всі обчислення виконуються на сервері, тому студентам не потрібно мати потужні персональні комп'ютери. Це робить навчання доступним для всіх, незалежно від технічних можливостей їхніх пристроїв. Віртуальні машини дозволяють легко масштабувати середовище відповідно до кількості студентів. За потреби можна швидко створювати нові віртуальні середовища або змінювати конфігурації існуючих. Централізоване управління середовищем дозволяє викладачам та адміністраторам налаштовувати середовище один раз і використовувати його для всіх студентів. Це значно зменшує витрати часу на технічне обслуговування. Серверні ресурси можуть бути налаштовані так, щоб забезпечити ефективне виконання паралельних обчислень, використовуючи OpenMP, MPI та інші бібліотеки. Це дозволяє студентам тестувати реальні паралельні алгоритми без обмежень продуктивності їхніх персональних комп'ютерів. Proxmox дозволяє швидко розгорнути віртуальні машини з необхідним середовищем, що значно спрощує налаштування системи. Одного разу налаштована конфігурація може бути легко клонована для нових користувачів. Оновлення програмного забезпечення та бібліотек можна



ПЕДАГОГІЧНА АКАДЕМІЯ: НАУКОВІ ЗАПИСКИ

здійснювати централізовано, без необхідності втручання студентів. Це забезпечує стабільність та узгодженість робочого середовища.

Враховуючи всі ці фактори, варіант з використанням віртуальних машин з доступом по SSH + JupyterHub виявився найкращим рішенням для організації лабораторних робіт з паралельного програмування в умовах дистанційного навчання. Він поєднує високу продуктивність, простоту адміністрування, низькі вимоги до студентських пристроїв і можливість масштабування навчального середовища. Отримані результати підтверджують доцільність впровадження цього підходу в навчальний процес.

Це рішення забезпечує простоту доступу для студентів, які можуть працювати через стандартні SSH-клієнти та веб-інтерфейс JupyterHub. Низькі вимоги до обладнання студентів, оскільки всі обчислення виконуються на сервері. Високу гнучкість та масштабованість, що дозволяє швидко додавати нові середовища при зростанні кількості користувачів. Зручне адміністрування – викладачі та технічні адміністратори можуть централізовано налаштовувати середовище та моніторити процес виконання лабораторних робіт. Високу продуктивність, оскільки сервер забезпечує достатні ресурси для ефективного виконання паралельних програм. Простоту розгортання та обслуговування – разове налаштування системи дозволяє мінімізувати витрати часу на підтримку середовища. Отримані результати свідчать про доцільність впровадження цього рішення для організації лабораторних робіт з паралельного програмування.

Попри очевидні переваги віртуалізованого середовища для проведення лабораторних робіт з паралельного програмування, запропоноване рішення має також низку обмежень та потенційних ризиків, які необхідно враховувати при його впровадженні.

Один із ключових чинників — залежність від стабільного інтернет-з'єднання. В умовах дистанційного навчання нестабільний або повільний



інтернет може значно впливати на комфорт та ефективність роботи студентів. Зокрема, затримки при підключенні через SSH або переривання сесій у JupyterHub можуть спричинити втрату прогресу або ускладнення при виконанні завдань. Цей ризик особливо актуальний для студентів із віддалених або технічно обмежених регіонів.

Іншим важливим аспектом є безпека доступу до віртуального середовища. Оскільки взаємодія зі студентськими середовищами здійснюється через відкриті мережеві протоколи (SSH, HTTPS), важливо впровадити засоби автентифікації та шифрування, зокрема використання ключів SSH, сертифікатів TLS та механізмів обмеження доступу за IP-адресами. Недостатня увага до захисту може призвести до несанкціонованого доступу до навчального середовища або до витоку даних.

Також варто враховувати залежність запропонованої архітектури від адміністративної підтримки. Початкове налаштування серверної інфраструктури на базі Proxmox, створення шаблонів віртуальних машин, встановлення та конфігурування JupyterHub потребує фахових знань та часу. У разі відсутності кваліфікованих системних адміністраторів або високого рівня автоматизації цей фактор може стати вузьким місцем у масштабному або довготривалому використанні системи.

З огляду на вищезазначене, подальше вдосконалення системи повинно включати:

- автономне кешування сесій для роботи в режимі offline-first (по мірі можливості),
- запровадження систем моніторингу та журналювання активності,
- автоматизацію резервного копіювання та оновлення середовищ,
- використання платформ авторизації (наприклад, OAuth2) для централізованого контролю доступу.



Висновки. У межах цього дослідження було успішно розроблено та віртуалізоване середовище для лабораторних робіт з паралельного програмування. Аналіз існуючих підходів до організації навчального процесу в умовах дистанційного навчання дозволив обґрунтувати вибір варіанту з використанням віртуальних машин на сервері з Proxmox та доступом через SSH і JupyterHub. Це рішення забезпечило централізоване керування, рівні умови для всіх студентів та ефективне використання ресурсів сервера. Був проведений аналіз сучасних технологій віртуалізації, розроблено архітектуру середовища, здійснено його розгортання та оцінено ефективність у навчальному процесі. Планується проведення тестування в межах таких освітніх компонентів як «Паралельне програмування» та «Паралельні та розподілені обчислення», яке дозволить оцінити ефективність запропонованого рішення. Очікується, що результати дослідження покажуть зростання успішності, покращення якості виконаних робіт, підвищення зацікавленості студентів та кращу контрольованість навчального процесу з боку викладача.

Подальші дослідження передбачають розробку та впровадження методики оцінки ефективності віртуалізованого середовища у реальних умовах навчального процесу. Зокрема, планується використання кількісних і якісних показників: рівень виконання лабораторних робіт, кількість технічних збоїв, продуктивність виконання паралельних програм, а також опитування студентів щодо зручності використання середовища та їхньої загальної мотивації. Проведення такого аналізу дозволить зробити обґрунтовані висновки щодо впливу впровадженого рішення на навчальні результати та ефективність дистанційного навчання з паралельного програмування.

Таким чином, впроваджене рішення демонструє потенціал для масштабування та застосування в інших технічних дисциплінах, які потребують складного обчислювального середовища. Проте подальші дослідження можуть



бути зосереджені на вдосконаленні автоматизації розгортання, оптимізації розподілу ресурсів між студентами, а також на аналізі впливу віртуалізованого середовища на тривалі навчальні результати.

Список використаних джерел

1. Pastor R., Caminero A.C., Rama D.S., Hernández R., Ros S., Robles-Gómez A., Tobarra L. Laboratories as a Service (LaaS): Using Cloud Technologies in the Field of Education. *UCS*. 2013. Vol.19. URL: <https://pdfs.semanticscholar.org/b6b9/0d00bcd1ed53525458a2a4a12280cb13595e.pdf> (date of access: 09.04.2025)
2. Tobarra L., Robles-Gómez A., Pastor R., Hernández R., Duque A., Cano J. Students' acceptance and tracking of a New container-based virtual laboratory. *Appl. Sci*. 2020. Vol. 10. URL: <https://doi.org/10.3390/app10031091> (date of access: 09.04.2025)
3. Ramos J. A. G., Albertini B., Solis-Lastra J. A Systematic Literature Review on Laboratory as a Service (LaaS). *IEEE International Congress on Electronics, Electrical Engineering and Computing (INTERCON)*. 2022. URL: <https://doi.org/10.1109/INTERCON55795.2022.9870147> (date of access: 09.04.2025)
4. Lui R. W. C., Zhang A. W. Y., Lee P. T. Y. A Secure and Scalable Virtual Lab Platform for Computing Education. *International Journal of Information and Education Technology*. 2024. Vol. 14, No. 1, URL: <https://www.ijiet.org/vol14/IJiet-V14N1-2024.pdf> (date of access: 09.04.2025)
5. Semerikov S. O., Vakaliuk T.A., MintiiI.S., Hamaniuk V.A., Bondarenko O.V. et al. Designing an immersive cloud-based educational environment for universities: a comprehensive approach. *Proceedings of the 6th International Workshop on Augmented Reality in Education*. 2024. Vol. 3844. P. 107-116. URL: <https://ceur-ws.org/Vol-3844/paper09.pdf> (date of access: 09.04.2025)



6. Asad M.M., Naz A., Shaikh A. et al. Investigating the impact of IoT-Based smart laboratories on students' academic performance in higher education. *Univ Access Inf Soc*. 2024. Vol. 23, p.p.1135–1149. URL:<https://doi.org/10.1007/s10209-022-00944-1>(date of access: 09.04.2025)
7. Arisal A., Suhartanto H. HPC-in-Containers: A Containerized Parallel Environment for Parallel Programming Learning Using Docker. *2024 International Conference on Information Technology Systems and Innovation (ICITSI)*. 2024, pp. 101-105, URL: <https://doi.org/10.1109/ICITSI65188.2024.10929375>. (date of access: 09.04.2025)
8. Kaler T. et al. Speedcode: Software Performance Engineering Education via the Coding of Didactic Exercises. *2024 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW)*. 2024, pp. 391-394, URL: <https://doi.org/10.1109/IPDPSW63119.2024.00087>. (date of access: 09.04.2025)
9. Newhall T., Webb K. C., Chaganti V., Danner A. An introductory-level undergraduate CS course that introduces parallel computing. *Journal of Parallel and Distributed Computing*. 2025. Vol.199. URL: <https://www.sciencedirect.com/science/article/pii/S0743731525000115> (date of access: 09.04.2025)
10. Computer Science Curricula 2023. URL: <https://csed.acm.org/wp-content/uploads/2023/03/Version-Beta-v2.pdf> (date of access: 08.04.2025).
11. Mathur P., Ahmad K.A., Hamid N.A.W.A., Jawaid M., Khan T., Singh, B. Cloud Computing Infrastructure, Platforms, and Software for Scientific Research. In: (eds) High Performance Computing in Biomimetics. Series in BioEngineering. Springer, Singapore. 2025. URL: https://doi.org/10.1007/978-981-97-1017-1_4 (date of access: 08.04.2025)
12. Bykov V., Shyshkina M., Emerging technologies for personnel training for IT industry in Ukraine. *2014 International Conference on Interactive Collaborative*



Learning (ICL). 2014, pp. 945-949, URL: <https://doi.org/10.1109/ICL.2014.7017903>.
(date of access: 08.04.2025)

13. Segrelles J. D., Martinez A., Castilla N., Moltó G. Virtualized Computational Environments on the cloud to foster group skills through PBL: A case study in architecture. *Computers & Education*. 2017. Vol. 108. p.p. 131-144 URL: <https://doi.org/10.1016/j.compedu.2017.02.001>. (date of access: 09.04.2025)

14. Baligodugula V. V., Amsaad F., Jhanjhi N., Analyzing the Parallel Computing Performance of Unsupervised Machine Learning. *2024 IEEE 1st Karachi Section Humanitarian Technology Conference (KHI-HTC)*. 2024, pp. 1-6, URL: <https://doi.org/10.1109/KHI-HTC60760.2024.10482277>. (date of access: 09.04.2025)

15. Heldens S.J. Parallel programming systems for scalable scientific computing. Amsterdam: University of Amsterdam, 2024. 187 p. URL: <https://core.ac.uk/download/pdf/604736195.pdf> (date of access: 09.04.2025)