

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ВОЛИНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ЛЕСІ УКРАЇНКИ
Кафедра комп'ютерних наук та кібербезпеки**

На правах рукопису

**ПАВЛЕНКО ЮЛІЯ СТЕПАНІВНА
ОРГАНІЗАЦІЯ ПОРТАТИВНОЇ СИСТЕМИ ДИСТАНЦІЙНОГО
НАВЧАННЯ НА БАЗІ LMS MOODLE**

Спеціальність: 122 Комп'ютерні науки
Освітньо-професійна програма: Комп'ютерні науки
та інформаційні технології
Робота на здобуття освітнього ступеня «магістр»

Науковий керівник:
БУЛАТЕЦЬКИЙ ВІТАЛІЙ ВІКТОРОВИЧ,
кандидат фізико-математичних наук, доцент

РЕКОМЕНДОВАНО ДО ЗАХИСТУ
Протокол № _____
засідання кафедри комп'ютерних наук
та кібербезпеки
від _____ 20__ р.
Завідувач кафедри
(_____) _____
(підпис) ПІБ

ЗМІСТ

ВСТУП	4
РОЗДІЛ 1 ПЛАТФОРМИ ДИСТАНЦІЙНОГО НАВЧАННЯ ТА ОСОБЛИВОСТІ ЇХ РОЗГОРТАННЯ.....	6
1.1 Поняття LMS, особливості.....	6
1.2 Поняття і види хостингу та серверних операційних систем	8
1.3 Взаємодія хостингу і серверних ОС.....	10
1.4 Серверні операційні системи Linux, як основа хостингу для розгортання LMS.....	11
1.4.1 Характеристика LAMP	13
1.4.2 Використання серверу NGINX	14
1.4.3 Опис стеку LEMP.....	15
1.4.4 MEAN – стек з технологій JavaScript.....	16
1.4.5 Використання платформи контейнеризації Docker	17
1.5 Принцип дії та використання платформ віртуалізації	18
1.6 Використання VPN для доступу до Інтернет.....	25
1.7 Аналіз рішень, що використовуються для розгортання LMS	27
РОЗДІЛ 2 РЕАЛІЗАЦІЯ ЗБІРКИ LMS «POCKET MOODLE».....	30
2.1 Постановка задачі, призначення та вимоги до програмного засобу «Pocket Moodle»	30
2.2 Вибір моделі розробки програмного засобу «Pocket Moodle».....	31
2.3 Загальний опис проекту.....	32
2.4 Обґрунтування вибору інструментальних засобів розробки.....	35

2.5 Особливості програмної реалізації та основні режими роботи «Rocket Moodle»	37
2.5.1 Створення віртуальної машини.....	37
2.5.2 Встановлення серверної операційної системи в якості гостьової	39
2.5.3 Моніторинг та керування.....	39
2.5.4 Забезпечення VPN-з'єднання для роботи «Rocket Moodle»	40
2.5.5 Стек LAMP	41
2.5.6 Додаткові налаштування платформи та інтерфейсу	47
2.6 Організація тестування та налагодження	51
2.7. Рекомендації по використанню та впровадженню	52
ВИСНОВКИ.....	54
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	56
ДОДАТОК А ТЕХНІЧНЕ ЗАВДАННЯ НА РОЗРОБКУ ПРОГРАМНОГО ЗАСОБУ «ROCKET MOODLE»	61
ДОДАТОК Б ВКАЗІВКИ КОРИСТУВАЧЕВІ	65

ВСТУП

Актуальність теми. Moodle – це безкоштовна та відкрита платформа для дистанційного навчання, яка завдяки своїм широким можливостям значно полегшує процес викладання та навчання не лише в закладах освіти. Вона здобула велику популярність у світі та є найбільш використовуваною системою такого типу.

Для її роботи необхідно мати вебсервер із підтримкою PHP та сервер баз даних, зазвичай MySQL. Такі умови можна забезпечити через власне серверне обладнання з публічною IP-адресою або скориставшись послугами хостинг-провайдерів. Однак це може спричинити додаткові витрати та ускладнити обслуговування платформи, особливо якщо потрібно переносити її на інший сервер чи хостинг. Ці труднощі стають особливо відчутними в умовах, коли хостинг недоступний, наприклад, через перебої в роботі глобальної мережі, коли неможливо використовувати Moodle в ізольованих локальних мережах або там, де немає доступу до Інтернет чи хостингу, на якому розгорнуто платформу.

Мета роботи – дослідити варіанти розгортання платформ дистанційного навчання та створити портативну LMS Moodle, яку можна швидко перенести з однієї мережі в іншу та просто розгорнути із забезпеченням віддаленого доступу до користувачів.

Для досягнення поставленої мети необхідно виконати **завдання**:

- розглянути поняття LMS, їх можливості та використання;
- описати види хостингу, серверних ОС та їх взаємодію;
- охарактеризувати технології розробки та розгортання вебдодатків;
- описати принцип дії та використання платформ віртуалізації;
- розглянути використання VPN;
- реалізувати, розгорнути та протестувати портативну LMS Moodle;
- вказати технічні параметри, необхідні для якісної роботи портативної LMS Moodle;

- розробити вказівки користувачу.

Об'єкт дослідження – портативна платформа дистанційного навчання.

Предмет дослідження – розгортання LMS Moodle з використанням платформи віртуалізації для створення хостингу.

Практичне застосування одержаних результатів. Портативна LMS Moodle може бути використана окремими невеликими структурними підрозділами закладів освіти різного рівня для навчання здобувачів, окремими викладачами, або вчителями, як для навчального процесу, так і для вивчення самої LMS на рівні адміністратора системи. Розробка протестована в мережі лабораторій кафедри комп'ютерних наук та кібербезпеки Волинського національного університету імені Лесі Українки та окремими викладачами кафедри у глобальній мережі.

Апробація результатів роботи. Публікації.

Булатецький, В., Булатецька, Л. і Павленко, Ю. РОЗГОРТАННЯ LMS MOODLE НА БАЗІ ORACLE VIRTUALBOX З ПІДТРИМКОЮ VPN Міжнародна науково-практична конференція "Проблеми комп'ютерних наук, програмного моделювання та безпеки цифрових систем" 13-16 червня 2024 54-55 <https://apcssm.vnu.edu.ua/index.php/conf/article/view/44/66>.

Булатецький, В., Булатецька, Л. і Павленко, Ю. 2024. ПОРТАТИВНА LMS MOODLE З ВІДДАЛЕНИМ ДОСТУПОМ. Прикладні проблеми комп'ютерних наук, безпеки та математики. 2 (Бер 2024), 70–78. URL: <https://apcssm.vnu.edu.ua/index.php/Journalone/article/view/19>.

РОЗДІЛ 1

ПЛАТФОРМИ ДИСТАНЦІЙНОГО НАВЧАННЯ ТА ОСОБЛИВОСТІ ЇХ РОЗГОРТАННЯ

1.1 Поняття LMS, особливості

LMS (Learning Management System, система управління навчанням) – це програмне забезпечення, як правило хмарне, яке дає змогу створювати, керувати і надавати освітні продукти та послуги в електронному вигляді, організувати онлайн-навчання та відстежувати прогрес учасників навчального процесу. LMS широко використовуються в освітніх установах (школи, університети) і в корпоративному секторі для навчання співробітників.

Основними функціями LMS є доставка навчальних матеріалів (тексти, відео, тести тощо), відстеження проходження навчання, тестування та оцінювання знань, звітність та аналітика, комунікація між учнями та викладачами, адміністрування користувачів і курсів.

LMS-платформа може використовуватися для просування та продажу курсів чи інших інформаційних продуктів будь-яким зацікавленим користувачам; переведення (повного або часткового) освітнього процесу в закладі освіти в онлайн-режим та управління цим процесом; навчання та підвищення кваліфікації працівників різних сфер.

Всі LMS, як правило, володіють наступними можливостями:

- створення та розміщення освітнього контенту в найбільш зручних форматах;
- централізоване та частково автоматизоване управління освітнім процесом;
- ведення єдиної бази здобувачів освіти/учнів;
- аналіз успішності здобувачів освіти/учнів, відстеження прогресу;
- тестування здобувачів освіти/учнів.

Системи управління навчанням класифікують за двома ознаками – залежно від цільової аудиторії та з погляду використовуваних технологій можна поділити на різновиди за двома основними критеріями.

Залежно від цільової аудиторії виділяють:

- корпоративні – призначені для навчання співробітників компаній, а також їх партнерів та клієнтів;
- академічні – використовують у своїй діяльності заклади освіти;
- комерційні – підходять для створення та розміщення платних курсів;
- системи змішаного типу – можна використовувати як для внутрішньокорпоративного навчання, так і для запуску комерційних інформаційних продуктів.

З погляду технології виділяють наступні типи LMS:

- серверні – у цьому випадку програмне забезпечення встановлюється на сервер, що належить компанії або взятий нею в оренду;
- хмарні – клієнти отримують доступ до готового сервісу, розташованого на серверах розробника.

Серверні LMS надають безперебійний доступ до даних та підвищену безпеку, зате вимагають високих витрат на обслуговування та є складнощі з установкою. Хмарні LMS економлять бюджет, працюють без територіальної прив'язки, але залежать від постачальника послуги та від стабільності інтернет-з'єднання. [1, 2]

Згідно з опитуваннями аналітиків, близько 83% компаній використовують LMS для онбордингу та підвищення кваліфікації своїх співробітників. Більше того, ринок систем управління навчанням зростає щороку на 15,6%. Його обсяг у 2023 році оцінювався в більш ніж \$20 мільярдів і очікується, що до 2028 року він досягне \$37 мільярдів [3].

LMS можна розгорнути двома способами: локально і з використанням хмарних середовищ.

У випадку локального (самостійного) розгортання організація, яка потребує LMS, купує ліцензію на комерційну LMS, завантажує програмне

забезпечення та встановлює його на своє апаратне забезпечення. Але в такому випадку користувачі можуть отримати доступ до системи лише на місці або за допомогою контрольованих засобів, таких як віртуальна приватна мережа (VPN).

При хмарному розгортанні немає потреби завантажувати ПЗ. Постачальник розміщує рішення в хмарі та відповідає за підтримку системи, а також за обробку всіх оновлень. Цей варіант, очевидно, кращий, бо доступ користувачів до LMS можна надати будь-коли і отримати його з будь-якого місця. Хоча при цьому треба мати якісне стабільне підключення до Інтернету. [4]

1.2 Поняття і види хостингу та серверних операційних систем

Хостинг – це послуга надання дискового простору та ресурсів на сервері, який постійно підключений до Інтернету, для розміщення вебсайтів або інших онлайн-сервісів, що дозволяє їм бути доступними в Інтернеті 24/7, забезпечуючи стабільний зв'язок між сервером і користувачами.

Виділяють наступні типи хостингу [5]:

- віртуальний хостинг (Shared Hosting);
- віртуальний приватний сервер (VPS);
- виділений сервер (Dedicated Hosting);
- хмарний хостинг (Cloud Hosting);
- хостинг для конкретних платформ (Managed Hosting). Наприклад, хостинг для WordPress, який оптимізований спеціально для цієї платформи. Хостинг-провайдер керує і підтримує сервери, а також забезпечує оптимізацію для швидкої роботи.

При використанні віртуального хостингу на одному фізичному сервері розміщуються декілька сайтів, які ділять між собою ресурси (процесор, оперативну пам'ять, дисковий простір). Він вважається найдешевшим і підходить для невеликих проєктів або сайтів із низьким рівнем трафіку.

Віртуальний приватний сервер використовує технологію віртуалізації, яка дозволяє одному фізичному серверу працювати як кілька ізольованих віртуальних серверів. При цьому кожен користувач має власні ресурси, що забезпечує більшу стабільність і контроль, ніж у віртуальному хостингу.

При використанні виділеного серверу клієнт отримує в оренду весь сервер, який надає максимальну потужність і можливість повного контролю. Підходить для великих проєктів або додатків з високим трафіком, які потребують значних ресурсів.

Хмарний хостинг використовує декілька серверів для підтримки додатків або вебсайтів, забезпечуючи масштабованість і стабільність. У випадку збою одного сервера робота продовжується на інших. Він підходить для динамічних проєктів, яким потрібні ресурси залежно від навантаження. [5, 6]

Хостинг – це послуга, яка надає ресурси для розміщення вебдодатків, а серверні ОС забезпечують керування і функціонування цих ресурсів. Хостинг-послуги базуються на інфраструктурі серверів, які працюють під управлінням серверних операційних систем. Серверні ОС забезпечують платформу для розміщення сайтів та додатків, а також керують ресурсами та безпекою, що є ключовим для ефективної роботи хостинг-сервісів.

Серверні операційні системи – це операційні системи, які забезпечують надійне, масштабоване і безпечне управління ресурсами для серверів. Вони оптимізовані для високої продуктивності при великій кількості одночасних запитів, підтримуючи стабільну роботу в мережі, і мають покращені засоби адміністрування, підтримки мережевих функцій, безпеки (контроль доступу, шифрування даних і функції захисту від атак) та роботи з різними типами навантаження, забезпечують ефективне управління базами даних і роботу з великими обсягами даних, які постійно оновлюються.

Популярними серверними ОС є Linux, Windows Server, Unix-подібні ОС (FreeBSD Solaris), контейнерні ОС (CoreOS (Container Linux) та RancherOS). Серед ОС Linux виділяють:

- Ubuntu Server – серверна версія Ubuntu, яка пропонує дружній інтерфейс і чудово підходить для вебсерверів, баз даних та додатків. Завдяки широкому ком'юніті і підтримці довготривалих версій (LTS), Ubuntu Server часто вибирають для малого та середнього бізнесу;

- CentOS – стабільна і безкоштовна ОС на основі Red Hat Enterprise Linux (RHEL), що надає більшість функцій RHEL, але без витрат на ліцензування. Ідеальна для використання в корпоративному середовищі завдяки високій стабільності;

- Red Hat Enterprise Linux (RHEL) – комерційна ОС з офіційною підтримкою від Red Hat. Вона має інструменти для адміністрування, автоматизації та підтримки систем безпеки, що робить її вибором номер один для великих підприємств.[7]

1.3 Взаємодія хостингу і серверних ОС

Взаємодія серверних операційних системи і хостингу відбувається на декількох рівнях. Серверні ОС (Linux або Windows Server) є базовою програмною платформою на фізичних або віртуальних серверах, які використовують хостинг-провайдери для надання послуг, забезпечуючи одночасну роботу багатьох вебсайтів та додатків.

В першу чергу, це управління ресурсами. Серверні ОС, такі як Linux (Ubuntu Server або CentOS) і Windows Server, забезпечують управління процесором, оперативною пам'яттю і дисковим простором. У хостинг-середовищах операційна система дозволяє ефективно розподіляти ресурси між користувачами або вебдодатками, зокрема у випадку віртуального хостингу або VPS.

Також хостинг-провайдери використовують серверні ОС для забезпечення стабільності та безпеки серверів. ОС надають інструменти для управління доступом, захисту даних і шифрування, а також регулярних оновлень і патчів для захисту від атак. Для критичних хостинг-додатків часто

обирають надійні серверні ОС, такі як Red Hat Enterprise Linux або Windows Server.

Багато хостинг-провайдерів використовують віртуалізацію та контейнеризацію, що дозволяє розміщувати на одному фізичному сервері кілька віртуальних серверів або контейнерів. Це забезпечується можливостями серверної ОС, яка підтримує технології Docker, Kubernetes, KVM та Hyper-V.

Серверні ОС надають адміністраторам доступ до інструментів і функцій для налаштування, моніторингу й управління серверами. Хостинг-провайдери використовують ці можливості для підтримки стабільної роботи серверів, керування базами даних, забезпечення безперервної роботи сайтів та обслуговування користувачів.

Слід зазначити, що різні типи хостингу (віртуальний, VPS, виділений або хмарний) залежать від можливостей серверної ОС. Наприклад, для дешевших видів хостингу використовують Linux, оскільки це безкоштовна та гнучка ОС з відкритим кодом. Для корпоративних рішень, які потребують інтеграції з іншими продуктами Microsoft, зазвичай обирають Windows Server.[8]

Серверні ОС підтримують такі вебсервіси, як Apache, Nginx, MySQL, які необхідні для функціонування вебсайтів і додатків, що розміщені на хостингу. Хостинг-провайдери вибирають серверні ОС залежно від потреб клієнтів. Linux підходить для вебсерверів із PHP і MySQL, тоді як Windows Server кращий для ASP.NET або інтеграції з Microsoft SQL Server.

1.4 Серверні операційні системи Linux, як основа хостингу для розгортання LMS

Windows-хостинг працює на базі операційної системи Windows Server і підходить для розміщення вебсайтів і додатків, що вимагають специфічних технологій та інструментів Microsoft, зокрема ASP.NET, MSSQL, Microsoft

Access і Windows PowerShell. Це дозволяє здійснювати інтеграція з продуктами Microsoft: Office 365, SharePoint, Exchange тощо. Водночас він є більш дороговартісним, оскільки ліцензії на Windows Server платні та менш стабільним для додатків, що використовують PHP або MySQL, адже ці технології краще підтримуються на Linux.

Серверні операційні системи на основі Linux є одними з найпопулярніших платформ для хостингу та розгортання LMS завдяки своїй стабільності, безкоштовній основі та можливості тонкого налаштування під специфічні потреби.

Серверні дистрибутиви Linux відомі своєю надійністю і тривалою підтримкою. Багато дистрибутивів надають LTS (Long Term Support) версії, що забезпечує стабільні оновлення та безпеку. Це дозволяє LMS безперебійно працювати, що особливо важливо для навчальних платформ із постійним доступом користувачів.

Linux надає можливість повного контролю над сервером, дозволяючи адміністраторам змінювати конфігурації сервера відповідно до потреб LMS. Наприклад, популярні LMS, такі як Moodle, можуть бути налаштовані з урахуванням продуктивності й інтеграції з іншими сервісами.

Оскільки популярні LMS (Moodle, Open edX або Canvas) потребують вебсервера (як правило, Apache або Nginx), бази даних (MySQL, PostgreSQL) та мови програмування (частіше PHP або Python), Linux забезпечує їх стабільну та швидку роботу часто краще, ніж на Windows-серверах.

Віртуалізація на базі Linux (наприклад, за допомогою Docker або Kubernetes) дозволяє легко розширювати можливості LMS при зростанні кількості користувачів. Linux підходить для розгортання в хмарних середовищах, таких як AWS, Google Cloud, Azure, де хостинг LMS можна масштабувати горизонтально. [9]

Розглянемо популярні технології та стекові рішення для розробки, розгортання і масштабування вебдодатків.

1.4.1 Характеристика LAMP

LAMP – це набір програмного забезпечення з відкритим кодом, який використовується для розробки і розгортання вебдодатків. Назва LAMP є акронімом основних компонентів стеку: операційної системи Linux, вебсервера Apache, сервера баз даних MySQL і мови програмування PHP. LAMP є гарним вибором для створення, розміщення і підтримки динамічних вебсайтів та додатків. Використовується для розробки сайтів на CMS, наприклад, WordPress, Joomla, Drupal тощо. Стек підходить для малих і середніх проєктів, вебсайтів, блогів, інформаційних порталів та інших вебзастосунків, які не потребують високого навантаження.

Принцип роботи LAMP, коли користувач відкриває вебсторінку в браузері:

- отримання запитів. Вебсервер Apache отримує вхідний запит від браузера. Якщо запит полягає у завантаженні статичного файлу, сервер Apache відповідає безпосередньо відповідним вмістом. Якщо запит відноситься до динамічного контенту, сервер Apache передає запит компоненту PHP. Компонент PHP знаходить і завантажує відповідний PHP-файл, який може обробити запит;

- обробка запитів. Файл PHP містить функції PHP. Компонент PHP обробляє функції PHP. У випадку, якщо для функцій PHP потрібна інформація з бази даних, то PHP-код видобуває збережену інформацію з бази даних і використовує її для обробки функції;

- повернення відповідей. Файл PHP передає обчислені результати вебсерверу в форматі HTML. Одночасно він зберігає нові дані в базі даних MySQL. HTTP-сервер Apache відправляє динамічні результати HTML в браузер користувача. [10]

1.4.2 Використання серверу NGINX

Nginx – це HTTP-сервер та IMAP/POP3 проксі-сервер для UNIX-подібних платформ (FreeBSD, OpenBSD, MacOS X, Solaris та GNU/Linux, а також на Windows). Це продуктивний і невимогливий до ресурсів сервер, який може одночасно обробляти багато з'єднань, гнучко розподіляючи навантаження. [11]

У Nginx особливий підхід до обробки даних, що дає змогу зменшувати навантаження на оперативну пам'ять і швидше видавати результат. Для кожного запиту створюється робоче з'єднання, всередині якого він розбивається на кілька підзадач. Частина запиту обробляються паралельно і швидше, ніж у цілому вигляді. Далі робоче з'єднання знову перетворюється на початковий запит і відправляється користувачеві. Усі ці операції проводяться у виділеному сегменті пам'яті, який автоматично розширюється в залежності від довжини запиту. Такий принцип роботи дозволяє використовувати Nginx для різних завдань [12]:

- як виділений порт або IP-адреса, щоб розподілити навантаження на сервер;
- як проксі, щоб кешувати дані і швидше повертати результат у разі повторних запитів на сторінку;
- як поштовий сервер, щоб перенаправляти авторизованого користувача на поштові сервіси;
- для асинхронної, більш швидкої обробки запитів.

Через дуже невелике споживання ресурсів системи і швидкості роботи, а також гнучкості конфігурування, вебсервер Nginx часто використовується як фронтенд до більш важких серверів, таких як Apache, в проектах з високим навантаженням. Класичним варіантом є зв'язка, Nginx – Apache – FastCGI. Працюючи в такій схемі, сервер Nginx, приймає всі запити, що надходять по HTTP, і в залежності від конфігурації і самого запиту, вирішує,

чи обробити запит самому і віддати клієнту готову відповідь або відправити запит на обробку, одному з бекендів (Apache або FastCGI).

На даний момент вебсервер Nginx є одним з найнадійніших серверних програм завдяки успішній реалізації керованої подіями та асинхронної архітектури. Його використовують такі інтернет-гіганти, як Google, WordPress, Netflix тощо. [13]

1.4.3 Опис стеку LEMP

LEMP – це стек програмного забезпечення з відкритим кодом, який використовується для розробки та розгортання вебдодатків, назва якого складається з перших літер основних компонентів: Linux, Nginx – вебсервер, який обробляє HTTP-запити та забезпечує швидку доставку контенту, MySQL або MariaDB, PHP – мова програмування, що обробляє серверну логіку та генерує динамічний контент. PHP взаємодіє з базою даних, обробляючи запити та формуючи відповіді для користувачів.

LEMP працює за наступним алгоритмом:

- користувач у браузері вводить адресу вебсайту;
- Nginx приймає запит і перевіряє, чи є запитаний файл (наприклад, CSS, зображення) статичним контентом. Якщо так – відразу повертає його. Якщо це динамічний запит (наприклад, PHP-скрипт), пересилає його на обробку PHP;
- PHP виконує код, звертається до MySQL для отримання даних, якщо це потрібно;
- MySQL повертає потрібні дані PHP;
- PHP формує HTML-відповідь із динамічним контентом і повертає її через Nginx до браузера користувача.

LEMP зазвичай використовується для створення вебсайтів, що потребують високої продуктивності та надійності. Завдяки можливостям Nginx обробляти тисячі з'єднань одночасно та ефективній взаємодії з PHP і

MySQL, цей стек підходить для великих онлайн-платформ, інтернет-магазинів та соціальних мереж.

Взагалі кажучи, основна різниця між LAMP та LEMP полягає в тому, що Nginx більш оптимізований для високонавантажених проєктів, тоді як Apache пропонує гнучкі налаштування для обробки модулів і додаткових функцій. [14]

1.4.4 MEAN – стек з технологій JavaScript

Стек MEAN це аббревіатура від MongoDB, Express.js, Angular і Node.js. Він є альтернативою LAMP і базується на однойменних технологіях JavaScript.

MongoDB – це база даних, яка добре працює з JSON – типом JavaScript для читання даних.

Node.js – це безкоштовне, кросплатформне середовище із відкритим кодом для виконання JavaScript, яке дозволяє розробникам створювати сервери, вебзастосунки, інструменти командного рядка та скрипти;

Express.js – це фреймворк для Node.js, що надає набір функцій для створення веб- і мобільних додатків. Наприклад, маршрутизація, підтримка шаблонів: інтеграція з різними шаблонізаторами (Jade, Handlebars і EJS), підтримка баз даних (MongoDB, PostgreSQL і MySQL), підтримка тестування. Express.js заснований на стандартних API Node.js, це платформа, яка надає інструменти з відкритим кодом для створення серверних додатків;

Angular – це сучасний фреймворк для розробки вебдодатків, створений Google. Він є наступником AngularJS і забезпечує значно вищу продуктивність, кращу структуру проєктів і сучасний підхід до розробки. Angular використовує мову TypeScript, що є розширенням JavaScript, для створення динамічних односторінкових додатків. [10]

1.4.5 Використання платформи контейнеризації Docker

Docker – це платформа контейнеризації, яка дозволяє розробникам створювати, запускати та масштабувати додатки в ізольованих середовищах, які називаються контейнерами. Контейнери – це легкий спосіб пакування та запуску застосунків, які можуть працювати на будь-якій машині з встановленою платформою.

Docker має низку переваг перед іншими способами розгортання застосунків (наприклад, віртуальними машинами). Контейнери набагато легші за віртуальні машини, тому вони використовують менше ресурсів і швидше запускаються. Контейнери також легше переносити.

Docker працює, створюючи уявне середовище для кожного контейнера. Це середовище включає ядро операційної системи, бібліотеки, файли та налаштування, необхідні для роботи застосунку. Контейнери ізольовані один від одного, що означає, що вони не можуть взаємодіяти безпосередньо. Це попереджає проблеми з конфігурацією та підвищує стабільність.

Платформа також забезпечує зручний спосіб керування контейнерами. Можна використовувати Docker CLI для запуску, зупинки, перезапуску та видалення контейнерів. Також є можливість використовувати Docker Compose для створення та керування групами контейнерів.

Завдяки легкій платформі контейнерної віртуалізації Docker допомагає мінімально використовувати ресурси та приховувати фонові процеси; швидше створювати, тестувати та розгортати додатки тощо. У своєму ядрі Docker дозволяє запускати практично будь-який додаток, безпечно ізольований в контейнері. Безпечна ізоляція дозволяє запускати на одному хості багато контейнерів одночасно.

Характеризується доволі простим синтаксисом, сумісний з усіма версіями операційних систем Linux і Windows, тому сфера застосування Docker практично не обмежена. [14, 15]

1.5 Принцип дії та використання платформ віртуалізації

Віртуалізація – це логічне об'єднання обчислювальних ресурсів у єдиний пул з подальшим абстрагуванням їх від фізичного обладнання. Всі платформи віртуалізації на програмному рівні імітують фізичні ресурси (процесорну потужність, пам'ять, дисковий накопичувач) і дозволяють запускати на них віртуальні машини з власними ОС, причому їх кількість може налічувати десятки в межах одного фізичного сервера.

Ядром платформи віртуалізації є гіпервізор. Він виконує роль посередника між фізичним обладнанням та віртуальним пулом ресурсів, контролюючи при цьому розподіл фізичних ресурсів між віртуальними машинами. Гіпервізор разом з додатковими утилітами, засобами розгортання та інструментами для керування віртуальними машинами утворюють платформу віртуалізації.[16]

Схематично різницю між традиційною і віртуальною архітектурою можна побачити на рис. 1.1.



Рисунок 1.1 - Різниця між традиційною і віртуальною архітектурою

Виділяють наступні види віртуалізацій:

- віртуалізація серверів;
- віртуалізація застосунків;
- віртуалізація робочих столів;
- мережева віртуалізація;

- віртуалізація даних і сховищ даних.

Віртуалізація серверів відкриває широкі можливості для оптимізації інфраструктури. Замість використання окремих фізичних серверів, компанії можуть створювати віртуальні машини, які дозволяють краще використовувати наявні ресурси. Це дозволяє знизити витрати на апаратне забезпечення та енергоспоживання. Крім того, віртуалізація серверів забезпечує легку міграцію та балансування ресурсів, що покращує доступність і забезпечує більшу надійність.

Наприклад, один фізичний сервер використовується для створення декількох віртуальних. На одному і тому ж сервері можна розгорнути пошту, сервер з базами даних і термінальний сервер. При цьому поштовий сервер буде використовувати приблизно 10% своїх потужностей, сервер бази даних – 30%, а термінальний – відповідно, приблизно 60%. Використовуючи віртуалізацію всітри сервери можна запустити на одній апаратній платформі і використовувати доступні обчислювальні ресурси з максимальною ефективністю.

Віртуалізація застосунків є цінним інструментом для розгортання та управління програмними продуктами. За допомогою віртуальних контейнерів, компанії можуть забезпечити ізольоване середовище для різних застосунків, що дозволяє запускати їх на одній фізичній машині без конфліктів. Це полегшує розгортання, управління та масштабування застосунків, що призводить до зниження витрат і сприяє ефективності розробки.

Наприклад, для вирішення завдань необхідне використати специфічне програмне забезпечення, яке працює лише на певному апаратному і програмному забезпеченнях. Тут віртуалізація допомагає, наприклад, запускати програми, написані для Windows, запускати на MacOS і навпаки.

Віртуалізація робочих столів стала популярним рішенням для організацій, що мають велику кількість користувачів, особливо, якщо вони не потребують всієї потужності повноцінної робочої станції. Цей підхід

дозволяє централізовано управляти робочими столами та забезпечити доступ до них через мережу. Користувачі можуть працювати зі своїми робочими столами з будь-якого пристрою, що забезпечує гнучкість та мобільність. Крім того, віртуалізація робочих столів дозволяє централізовано керувати безпекою, резервним копіюванням та оновленнями, що полегшує адміністрування.

Мережева віртуалізація відіграє важливу роль у підвищенні ефективності та безпеки мережевої інфраструктури компаній. Шляхом відокремлення фізичної мережі на віртуальні сегменти та використання віртуальних мережевих пристроїв, таких як комутатори та маршрутизатори, можна досягти більшої гнучкості та контролю. Мережева віртуалізація дозволяє забезпечити ізоляцію мережевих потоків, розділити трафік на основі політик безпеки та ефективно масштабувати мережеві ресурси. Застосування віртуальних пристроїв дозволяє спростити управління мережею та знизити загальну складність інфраструктури.

Також, якщо в організації є декілька філій і в них використовується різне мережеве апаратне забезпечення, то віртуалізація мережі допоможе об'єднати технічно різні сітки в одну і полегшити адміністраторам процес керування нею.

Дані сьогодні збираються з різних джерел, зберігаються в різних форматах і місцях. Інструменти віртуалізації допомагають створити зв'язок між такими даними і додатками, які працюють з цими даними. Також дані можуть зберігатися в сховищах різних типів і різних виробників. Технології віртуалізації дозволяють їх об'єднати в одне і керувати ними з допомогою одного ПЗ.

Отже, основними перевагами віртуалізації можна виділити наступні:

- оптимізація використання фізичних ресурсів та, як наслідок, витрат;
- підвищення рівня автоматизації бізнес-процесів;
- підвищення адаптивності та масштабованості інфраструктури;
- просте відновлення віртуальних машин у разі збою. [17, 18]

У випадку, якщо віртуалізація ресурсів використовується на стороні провайдера, бізнес може додатково скоротити витрати на адміністрування своєї IT-інфраструктури. Постачальники програмного забезпечення обирають віртуалізацію для розробки програмного забезпечення та подальшого тестування за її гнучкість.

Існує кілька популярних гіпервізорів, які надають різні можливості та функціонал: Microsoft Hyper-V, VMWare, Proxmox, OracleVirtualBox, QEMU.

Hyper-V є інтегрованим рішенням віртуалізації від компанії Microsoft. Цей продукт дозволяє створювати та управляти віртуальними машинами на базі операційних систем Windows Server, а також ОС сімейства Unix. За замовчуванням Hyper-V входить до складу Windows 10 Pro і Windows Server (2019 і 2022) і одночасно може працювати з Linux-системами (Debian, Ubuntu і CentOS) з деякими обмеженнями. Підходить для створення віртуального середовища: дозволяє будувати багаторівневу систему, сумісну з великим списком апаратних рішень, підтримує встановлення віртуальних робочих столів. Hyper-V забезпечує високу продуктивність та масштабованість, а також має вбудовані засоби для управління резервним копіюванням, моніторингом та забезпеченням безпеки. Використання Hyper-V дозволяє ефективно використовувати ресурси сервера та забезпечує високу доступність сервісів.[18]

Proxmox є відкритою платформою віртуалізації, яка поєднує в собі можливості віртуалізації на базі KVM (Kernel-based Virtual Machine) та контейнерів LXC (Linux Containers). Proxmox надає зручний інтерфейс для створення, управління та моніторингу віртуальних машин та контейнерів. Ця платформа також має вбудовані засоби для резервного копіювання, відновлення та масштабування, що сприяє надійності та гнучкості використання. Proxmox може бути варіантом для компаній, які шукають відкрите та економічне рішення в сфері віртуалізації.

OracleVirtualBox – кросплатформне ПЗ з відкритим вихідним кодом для віртуалізації, яке дозволяє розробникам швидше створювати код, працюючи

з кількома операційними системами на одному пристрої. Використовуючи VirtualBox, IT-команди та постачальники рішень зменшують операційні витрати й скорочують час, необхідний для безпечного розгортання програм у локальному та хмарному середовищі.

Можливості OracleVirtualBox стосуються багатьох аспектів. Розробники можуть спростити робоче середовище, використовуючи універсальне рішення, яке працює на будь-якій x86-сумісній хостовій операційній системі, зокрема Windows, Linux і macOS, і підтримує запуск різних версій ОС на віртуальних машинах.

Завдяки інтуїтивному графічному інтерфейсу та можливостям командного рядка, це рішення дозволяє ефективно працювати з кількома операційними системами на одному пристрої. Воно підтримує великі робочі навантаження, використовуючи до 32 віртуальних процесорів, і забезпечує зручний імпорт та експорт віртуальних машин у форматі OVF для локального або хмарного використання.

Команди, які займаються контролем якості програмного забезпечення, можуть оптимізувати свої процеси та зменшити витрати, використовуючи один фізичний пристрій для тестування програм на різних платформах і версіях операційних систем. Це дозволяє значно спростити середовище для роботи та підвищити ефективність використання ресурсів.

Зарахунок легкого відтворення різних середовищ замовників в одній системі, покращується підтримка замовників. Фахівці з продажу можуть легко демонструвати багаторівневі рішення потенційним клієнтам, використовуючи заздалегідь налаштовані середовища з кількома віртуальними машинами та мережевими топологіями безпосередньо на своєму ноутбучі.

VirtualBox забезпечує безпечний доступ до програм з обмеженим доступом, дозволяючи IT-менеджерам розповсюджувати десктопні образи критично важливих програм для віддалених працівників у ситуаціях, коли VPN-з'єднання недостатнє. Це рішення підвищує безпеку організацій,

впроваджуючи рольові обмеження для управління доступом до наборів даних у таких програмах.

VirtualBox забезпечує безпечний доступ [19] до програм з обмеженим доступом завдяки 256-бітному шифруванню, яке блокує можливість завантаження чи збереження даних на віддалених пристроях. Це дозволяє ІТ-менеджерам зекономити ресурси на перебудові архітектури таких програм.

VMWare надає широкий спектр функцій та можливостей для розгортання й управління віртуальними середовищами. VMWare дозволяє використовувати різні операційні системи та додатки на одній фізичній машині, що забезпечує ефективне використання апаратних ресурсів. Крім того, VMWare пропонує інструменти для автоматизації, моніторингу та управління віртуальними середовищами, що полегшує адміністрування.

Компанія розробляє засоби віртуалізації для Enterprise-сегмента, що цінує гнучкість, стабільність та функціональність. Найпопулярніші продукти: vSphere, vCenter Server, NSX Data Center та Horizon. Платформа дозволяє одночасно використовувати кілька застосунків у різних ОС, підтримує ОС Windows, Linux, Solaris та FreeBSD, інтеграцію зі стороннім ПЗ та ряд функцій, які значно підвищують зручність та продуктивність роботи – наприклад, балансування навантаження. Багато функцій, властиві платформам віртуалізації, першими з'являються саме у VMware.[20]

QEMU (Quick Emulator) – це програмний інструмент з відкритим кодом, який забезпечує емуляцію і віртуалізацію. Його можна використовувати для створення віртуальних машин, емуляції апаратного забезпечення або тестування програмного забезпечення в ізольованих середовищах.

Основними функціями QEMU є:

- емуляція процесора та інших пристроїв, наприклад, мережеві інтерфейси, накопичувачі, периферійні пристрої тощо.
- віртуалізація:

- системна емуляція, що дозволяє запускати операційні системи та їхні компоненти, створюючи ізольовані середовища для тестування або розробки;
- ізольоване середовище, яке використовується для запуску потенційно небезпечного програмного забезпечення або для тестування ОС без ризику вплинути на основну систему.

QEMU може емулювати різні архітектури процесорів, такі як x86, ARM, PowerPC, RISC-V тощо. Це дозволяє запускати програми, створені для однієї архітектури, на комп'ютерах з іншою архітектурою.

QEMU може працювати разом із KVM (Kernel-based Virtual Machine) для забезпечення апаратно-прискореної віртуалізації. Це дозволяє запускати віртуальні машини майже з нативною продуктивністю.

QEMU використовується у сфері розробки ПЗ при тестуванні ПЗ на різних платформах і ОС; в початковому процесі при вивченні роботи операційних систем або створення ізольованих навчальних середовищ; при віртуалізації серверів використовується в хмарних інфраструктурах для розгортання ізольованих серверних середовищ; при емуляції IoT-пристроїв може емулювати мікроконтролери та інші апаратні компоненти IoT.

Слід зазначити, що в режимі емуляції без апаратного прискорення QEMU працює повільніше, ніж інші рішення.

QEMU використовується в операційних системах, таких як Linux, для тестування різних середовищ; у хмарних рішеннях, зокрема OpenStack, для створення віртуальних серверів; для розробки програмного забезпечення, зокрема для мобільних пристроїв і IoT. [21]

Citrix (Xen) – рішення корпоративного рівня з відкритим кодом існує, зокрема, у безкоштовній версії. Підтримує розширені функції віртуалізації, роботу з хмарами, що гіпермасштабуються, системами віртуалізації VMware vSphere і Microsoft Hyper-V, а також з усіма популярними ОС: Windows, GNU/Linux, FreeBSD, MiniOS, NetBSD, Solaris. Володіє гарною масштабованістю, спрощеним керуванням та вбудованою перевіркою на сумісність програм.

OpenVZ – це високопродуктивна система віртуалізації лише на рівні операційної системи. Рішення створено на ядрі Linux і підтримує лише відповідні гостьові дистрибутиви. OpenVZ притаманна точна емуляція фізичного сервера, динамічне управління ресурсами, продуктивність, простота адміністрування.

Virtuozzo – це рішення для віртуалізації ресурсів фізичних серверів частково реалізовано на відкритому вихідному коді попередньої платформи. Підтримує такі ОС як Fedora, RHEL, Centos, Debian, OpenSUSE, Ubuntu, FreeBSD та всі популярні модулі ядра контейнера. Система переважно орієнтована на малий та середній бізнес, відрізняється простотою та доступністю використання. Забезпечує контроль VPS у реальному часі без перезавантажень, має продуману систему кешування, розгортання великої кількості віртуальних машин на фізичному обладнанні.

KVM – це продукт з відкритим вихідним кодом для віртуалізації в середовищі Linux, одне з найвідоміших рішень як альтернатива комерційним продуктам. Ця платформа постійно розвивається та пропонує великі можливості для кастомізації. Функціональність рішення робить його популярним серед розробників ПЗ. Володіє високим рівнем приватності, встановлення власної ОС, незалежністю ресурсів один від одного.

1.6 Використання VPN для доступу до Інтернет

VPN (Virtual Private Network) – це технологія, яка створює захищене з'єднання між пристроєм користувача і віддаленим сервером, забезпечуючи безпеку, конфіденційність і доступ до ресурсів в Інтернеті. Основний принцип роботи VPN полягає в шифруванні переданих даних і перенаправленні трафіку через сервер VPN.

З VPN можна безпечно здійснювати Інтернет-серфінг, не хвилюючись що особисті дані користувача (метадані, паролі, cookie, токени, сесії) будуть перехоплені або історія інтернет-перегляду буде кимось прочитана. [22]

Поетапно робота VPN виглядає наступним чином. Під час підключення користувача до VPN, всі його дані шифруються за допомогою криптографічних протоколів, таких як OpenVPN, L2TP/IPsec, WireGuard тощо. Далі VPN створює захищений "тунель" між пристроєм користувача і сервером VPN, через який проходять всі дані, що запобігає їхньому перехопленню або зміні під час передачі. Інтернет-трафік спочатку надходить до VPN-сервера, а потім перенаправляється до кінцевого пункту (вебсайту, додатку), завдяки чому фактична IP-адреса користувача замінюється IP-адресою сервера VPN. Оскільки користувач отримує IP-адресу сервера VPN, то він «може перебувати» у будь-якій країні, де розташовані сервери провайдера. Це дає змогу отримувати доступ до гео заблокованого контенту (наприклад, стрімінгових сервісів). Для забезпечення безпечного доступу до мережі перед встановленням з'єднання VPN-клієнт і сервер взаємно автентифікують один одного.

Одночасно з цим слід зважати на те, що VPN може знижувати швидкість Інтернет-з'єднання через шифрування і перенаправлення трафіку і надійні VPN-сервіси зазвичай платні.

Робота VPN забезпечується протоколами VPN. Протокол VPN – це набір інструкцій щодо шифрування й спрямування даних і трафіку між комп'ютером користувача і VPN-сервером. Наведемо приклади поширених VPN-протоколів

OpenVPN – це VPN-протокол із відкритим вихідним кодом: швидкий, безпечний і сумісний з багатьма маршрутизаторами. Завдяки відкритості його вихідний код наразі вже ретельно вивчений експертами з кібербезпеки усього світу. Це надійний робочий інструмент, але порівняно з IKEV2 та WireGuard він працює трохи гірше. Може використовуватися у Windows, macOS, Android, iOS, Linux і BSD.

WireGuard® – це відносно новий протокол, який перевершує OpenVPN і IPsec за характеристиками енергоспоживання й продуктивності, маючи всього 4000 рядків коду. На сьогодні WireGuard є найшвидшим із усіх

доступних VPN-протоколів. Може використовуватися у Windows, macOS, Android, iOS, FreeBSD, NetBSD, OpenBSD тощо.

Internet Key Exchange версія 2 (IKEv2) за якістю не поступається протоколу WireGuard. Він може перевершувати інші протоколи за швидкістю в разі зміни мереж і в ближніх з'єднаннях, завдяки чому є популярним серед мобільних користувачів. Може використовуватися у macOS, Android, iOS, Linux, Blackberry, Unix тощо.

PPTP і L2TP/IPSec – це доволі застарілі протоколи тунелювання, і користуватися ними не рекомендується. Працюючи відносно швидко, вони поступаються OpenVPN, IKEv2 та WireGuard за безпекою та надійністю.

SSTP – це протокол тунелювання, а не VPN-протокол. Тому йому не вистачає функціональності OpenVPN, IKEv2 й WireGuard. Хоча його код вважається відносно безпечним і простим у використанні, він ніколи не піддавався незалежній перевірці й погано працює в операційних системах, окрім Windows, Linux, BSD.

Shadowsocks – це не VPN-протокол, а проксі для тунелювання, однак він може працювати як протокол на VPN-клієнтах. Його створили з метою — обійти Великий китайський брандмауер. Він швидкий, його важко виявити й можна використовувати для обходу державної цензури. Використовується у Windows, macOS, Android, iOS, Linux. [23-24]

1.7 Аналіз рішень, що використовуються для розгортання LMS

Розглянемо можливі рішення для розгортання LMS. Одним із них є використання ноутбука із хостингом. Це можна вважати зручним й гнучким рішенням для освітніх закладів, компаній, які прагнуть створити власну платформу для онлайн-навчання. Такий підхід поєднує мобільність локального пристрою та надійність хмарних технологій.

В такому випадку можна почати налаштування LMS локально на ноутбуці, тестуючи платформу без підключення до мережі Інтернет. Це дає

можливість уникнути початкових витрат на хмарні ресурси. Налаштування середовища розробки на ноутбучі передбачає встановлення та наявність вебсерверу (Apache, Nginx), бази даних (MySQL, PostgreSQL), мови програмування (PHP, Python, JavaScript). Тоді ноутбук використовується як локальний сервер для налагодження.

Після налаштування середовища, перевірки функціональності, продуктивності, зручності використання та тестування LMS, її можна завантажити на хостинг, що забезпечує стабільний доступ до системи для користувачів з різним місцезнаходженням.

Надалі доречно використовувати ноутбук для тестування оновлень і нових модулів перед додаванням їх на основну платформу.

Іншим популярним рішенням є використання XAMPP – це локальний кросплатформний вебсервер з відкритим кодом та підтримкою усіх популярних операційних систем (Windows, Mac, Linux). Назва XAMPP є акронімом і походить від перших літер: X (усі операційні системи), A (вебсервер Apache), M (MySQL база даних), P (PHP) і P (Perl).

XAMPP дозволяє створити повністю функціональний вебсайт локально, на комп'ютері або ноутбучі. Розробники можуть перевіряти роботу своїх сайтів і скриптів локально перед їх розгортанням на реальному сервері.

XAMPP популярний серед студентів, новачків, які вивчають веброзробку, веброзробників, зокрема, завдяки простоті встановлення і налаштування. Встановлення передбачає завантаження інсталятора та його запуску, після чого весь стек буде готовий до використання. [25-26]

Слід зауважити, що XAMPP не підходить для використання в продакшені, оскільки його початкові налаштування орієнтовані на простоту, а не на безпеку. Тому доречним є його використання для невеликих проєктів.

Аналогами XAMPP можна вважати WAMP (Windows, Apache, MySQL, PHP), MAMP (Mac, Apache, MySQL, PHP), Laragon, AMPPS тощо.

WAMP є інструментом для створення локального вебсервера спеціально для Windows. WAMP має зручний інтерфейс для управління

налаштуваннями серверу, простий у використанні, підтримує різні версії PHP та включає phpMyAdmin для управління базами даних. [27]

MAMP пропонує легкий спосіб запуску локального сервера для користувачів macOS, але має версію і для Windows. MAMP включає в собі Nginx як альтернативу Apache та пропонує Pro-версію з додатковими функціями, такими як мультисайт. На жаль, безкоштовна версія має обмежену функціональність. [28]

Laragon доступний тільки для Windows, є потужною альтернативою XAMPP з оптимізацією для високої продуктивності. Підходить для створення як PHP-проектів, так і проектів на інших мовах (Node.js, Python). Характеризується легкою установкою, підтримує різні стек-технології та має вбудовану підтримку SSL. [29]

AMPPS – це комплексний локальний сервер з підтримкою Apache, MySQL, MongoDB і багатьох мов програмування, що підтримується Windows, macOS, Linux. Підтримує встановлення популярних CMS (WordPress, Joomla) в один клік, підходить для тестування різних стеків та має гнучкі налаштування PHP. Але важчий у налаштуванні для новачків. [30]

РОЗДІЛ 2

РЕАЛІЗАЦІЯ ЗБІРКИ LMS «POCKET MOODLE»

2.1 Постановка задачі, призначення та вимоги до програмного засобу «Pocket Moodle»

Основне призначення розробки – надати вчителів (викладачів, учнів, здобувачів освіти, адміністратору), який не є спеціалістом у мережах, операційних системах та вебпрограмуванні, універсальний мобільний та простий інструмент для створення та використання електронних курсів для очного або дистанційного навчання.

Необхідно створити систему, яка була б проста у розгортанні, користуванні, перенесенні та копіюванні. Вона повинна працювати як в локальній, так і у глобальній мережі. Система повинна бути безкоштовною та універсальною щодо платформ розгортання.

Тому розроблюваний програмний продукт повинен:

- виконувати основні функції LMS;
- надавати користувачеві повні адміністративні права всередині LMS;
- містити в собі мінімальну кількість файлів;
- не вимагати, або вимагати мінімум встановлення додаткового програмного забезпечення;
- не вимагати, або вимагати мінімум попереднього налаштування;
- не вимагати спеціалізованого апаратного забезпечення;
- бути зрозумілим та зручним для типового користувача;
- не залежати від операційної системи, тобто бути кросплатформним;
- бути відкритим та безкоштовним;
- містити типові поширені компоненти та бути сумісним з аналогічним програмним забезпеченням;
- працювати у локальній та глобальній мережах;
- при потребі надавати можливість кастомізації;
- містити засоби захисту для роботи у мережі;

- містити у собі лаконічні та, водночас, вичерпні коментарі, зрозумілі користувачеві.

Такий програмний продукт також повинен містити зрозумілу інструкцію для користувача.

2.2 Вибір моделі розробки програмного засобу «Pocket Moodle»

Розглянувши існуючі моделі розробки програмного забезпечення та врахувавши особливості призначення та функціонування майбутнього програмного продукту, за основу було обрано каскадну модель. У цій моделі кожна фаза проєкту виконується один раз, послідовно змінюючи одна одну. Щоб розпочати наступний етап, необхідно повністю завершити попередній.

Каскадна або "водоспадна" модель розробки, є послідовним підходом до створення програмного забезпечення. У ній проєкт розбивається на окремі етапи, які виконуються один за одним у визначеному порядку. Початкова фаза включає збір вимог, після чого слідує проєктування, реалізація, тестування, впровадження та обслуговування (рис. 2.1).

Кожен етап починається лише після завершення попереднього, що забезпечує структурованість і контроль над процесом. Важливою особливістю є те, що повернення до попередньої стадії обмежене, тому виправлення помилок на пізніх етапах може бути складним і дорогим. Каскадна модель підходить для проєктів із чіткими вимогами та обмеженими змінами, але менш ефективна для проєктів з динамічними потребами.

На початку роботи визначаються всі основні вимоги до системи, які поділяються на більш і менш важливі. Після чого виконується розробка системи за принципом послідовного виконання окремих завдань так, що виконання кожного наступного завдання можливе лише при виконанні попереднього. Основні важливі для нас переваги це:

- простота та зрозумілість – послідовний підхід із чітко визначеними етапами робить модель легкою для розуміння;

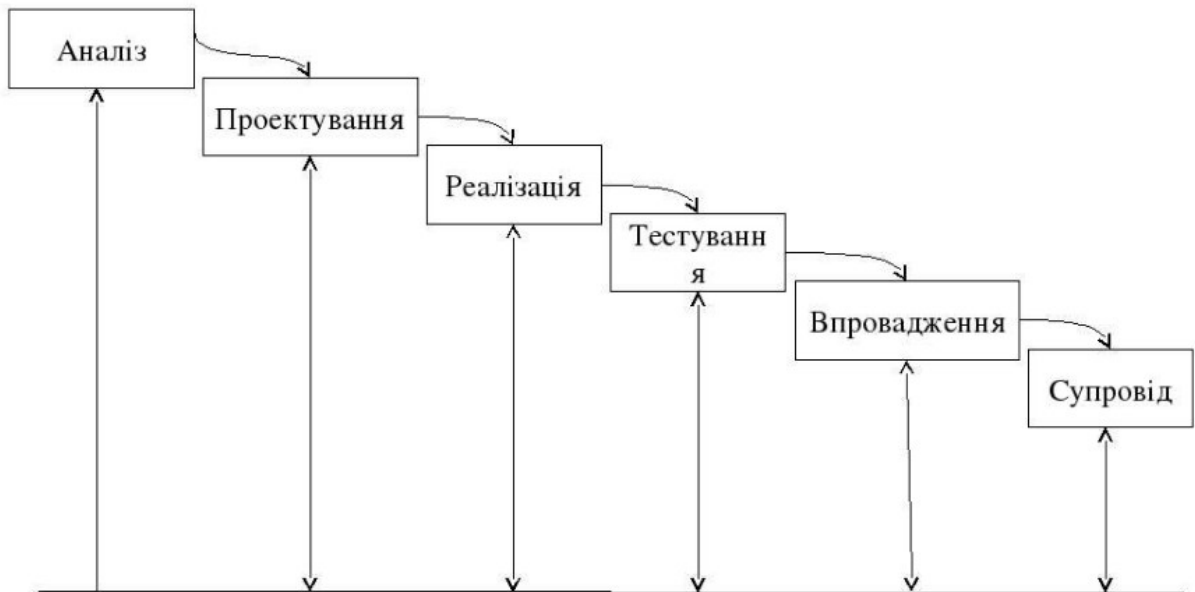


Рисунок 2.1 - Каскадна модель життєвого циклу розробки

- чітка структура та контроль обумовлені тим, що кожна стадія проєкту завершується перед початком наступної, що дозволяє легко контролювати процес розробки;
- усі вимоги, технічні рішення та результати кожної стадії фіксуються, що спрощує обслуговування;
- оскільки вимоги визначаються на початку, зменшується ймовірність змін, що позитивно впливає на стабільність розробки.

2.3 Загальний опис проєкту

Спочатку необхідно визначитись з типом LMS, яка б задовольняла наші вимоги. Очевидно, що найкраще на цю роль підходить Moodle – безкоштовна та відкрита система, яка завдяки широкому набору інструментів та середовищ, здатна розширити можливості викладання та навчання. Вона набула великої популярності і є найпоширенішою системою подібного типу у світі. [31-33]

Наступним кроком є підбір платформи для хостингу, який необхідний для розгортання на ньому Moodle, адже це веборієнтоване програмне забезпечення. Оскільки реальні апаратні рішення є досить вартісними, що не

відповідає нашим вимогам, було вирішено скористатись однією із безкоштовних платформ віртуалізації. На такій платформі можна розгорнути в якості гостьової операційної системи одну із безкоштовних та відкритих серверних операційних систем, що може слугувати основою для побудови хостингу і, відповідно, встановлення на ньому LMS.

Для забезпечення роботи в локальній мережі достатньо підібрати режим роботи мережевого інтерфейсу платформи віртуалізації (рис. 2.2).

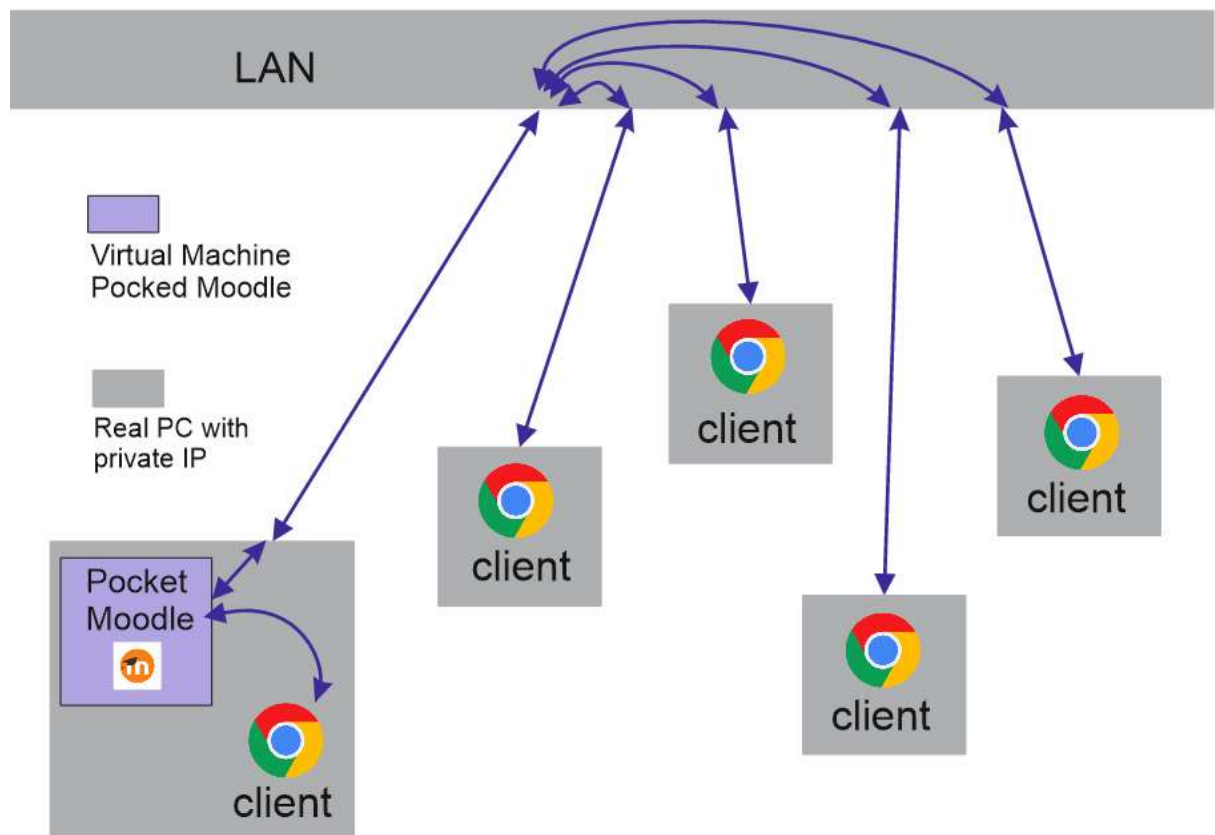


Рисунок. 2.2 - Функціонування Pocket Moodle у локальній мережі

Забезпечення роботи у глобальній мережі є дещо складнішим завданням, оскільки створований нами переносний хостинг не може використовувати публічні IP-адреси, адже наперед невідомо в якій мережі він працюватиме. Тому для забезпечення гнучкості та універсальності найкращим рішенням може бути використанні VPN-з'єднань, що дозволить працювати хостингу в глобальній мережі, так ніби він знаходиться у одній

локальній мережі із вузлами-клієнтами (рис. 2.3). Проте, для реалізації такого рішення необхідна наявність доступу до реального VPN-сервера, який ми вирішили створити самостійно, проте при потребі можна скористатися і сторонніми серверами. VPN-сервер єдиний у цій системі має публічну IP-адресу, проте його розгорнути набагато простіше ніж платформу Moodle, або можна використати існуючі безкоштовні VPN-сервери. Використання VPN принесе також елемент безпеки з'єднання в нашу систему (рис. 2.3).

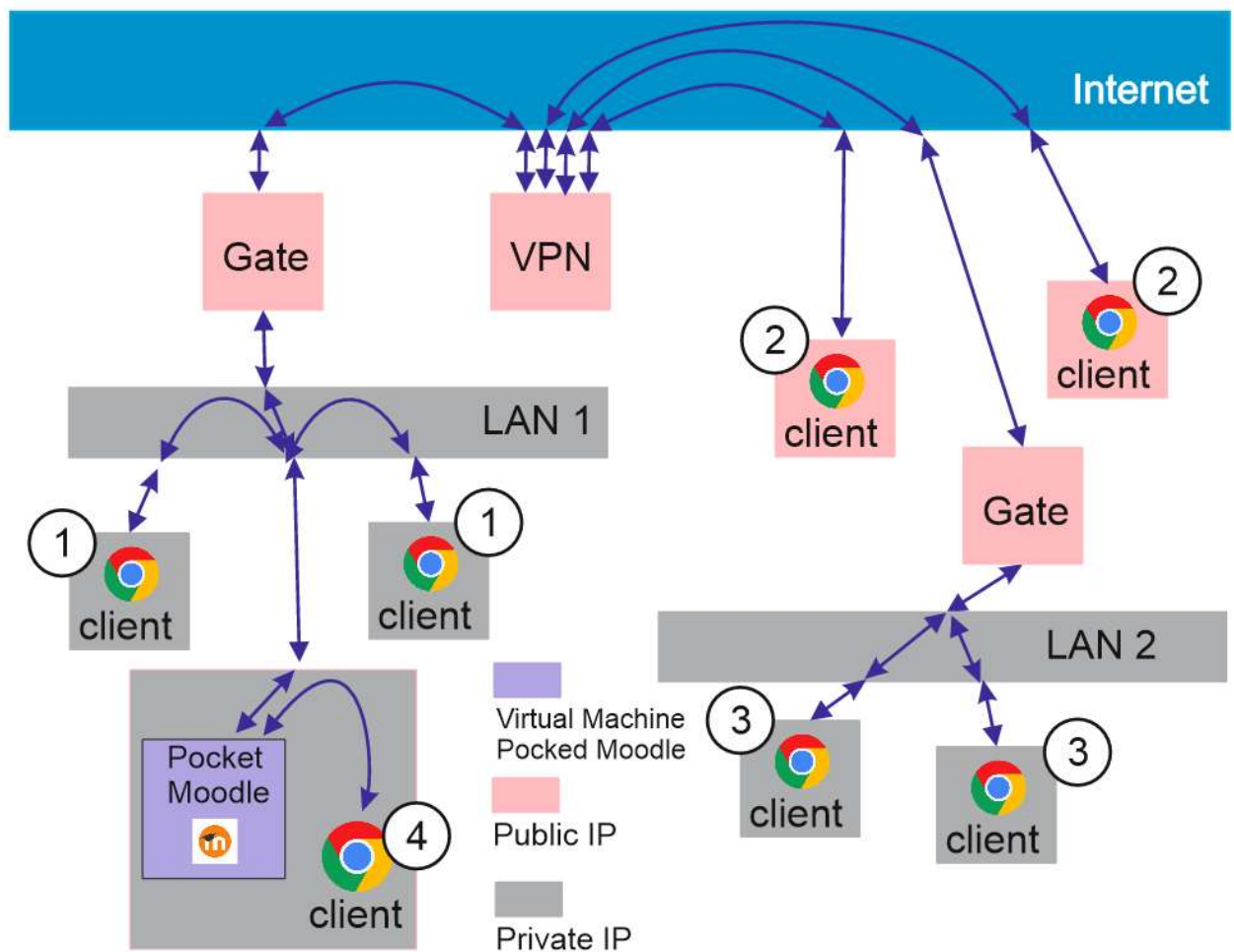


Рисунок 2.3 - Функціонування Pocket Moodle у глобальній мережі, де:

- 1 – клієнти локальної мережі у яку включено Pocket Moodle без з'єднання VPN;
- 2 – клієнти глобальної мережі, в яких активоване з'єднання VPN;
- 3 – клієнти іншої локальної мережі, в яких активоване з'єднання VPN;
- 4 – клієнт без з'єднання VPN, на якому встановлено віртуальну машину з Pocket Moodle, в якій активоване з'єднання VPN.

В результаті, після запуску віртуальної машини з хостингом, на якому розгорнуто Moodle, в конкретній локальній мережі, її вузли можуть отримати доступ до цієї LMS, в тому числі і вузол, на якому встановлено віртуальну машину. А якщо операційна система хостингу активує разом із клієнтськими вузлами VPN-з'єднання до одного VPN-сервера, то такі клієнти зможуть скористатись Pocket Moodle з будь-якої точки планети.

Оскільки така система розгорнута на віртуальній платформі, то являє собою невелику кількість файлів (один при експорті). Це робить дуже простою задачу перенесення, дублювання та резервного копіювання системи і не вимагає особливих знань від користувачів. [34-35]

2.4 Обґрунтування вибору інструментальних засобів розробки

Якщо необхідно організувати розгортання платформи Moodle для роботи в різних локальних мережах, то найочевиднішим рішенням може бути використання портативних пристроїв (наприклад, ноутбуків чи нетбуків) в якості своєрідного хостингу з повноцінною серверною операційною системою та розгорнутою на ній платформою LMS Moodle. Такі пристрої можна переносити, залучати до різних локальних мереж, таким чином надавати можливість клієнтам цих мереж взаємодіяти з цією LMS.

Проте, такий підхід передбачає використання апаратних засобів, виділених спеціально лише для розгортання цієї платформи Moodle, що є відносно вартісним рішенням, а перенесення платформи на інший портативний пристрій, або резервне копіювання вимагає спеціальних знань в галузях адміністрування апаратних засобів, адміністрування баз даних, серверних операційних систем та мережі і може зайняти досить тривалий час.

Іншим способом вирішення проблеми може бути використання пакету ХАМРР [25, 36]. Він вимагає наявності в системі однієї із клієнтських операційних систем, як хостової (Windows, Linux або Mac OSX), налагодження вебсервера та сервера баз даних, і повторних таких операцій

при перенесенні платформи Moodle на інший пристрій, що, знову ж, вимагає спеціальних знань та часу. Такий спосіб більше підходить для тестування, а не для постійного використання, він принципово не вирішує проблему складності переносу платформи на інший пристрій або резервного копіювання, а, відповідно, розповсюдження платформи та використання її пересічними користувачами.

Очевидно, щоб забезпечити виконання поставлених нами цілей варто отримати незалежну від апаратного забезпечення систему, організовану у вигляді одного або кількох файлів, які не вимагають безпосереднього редагування перед використанням і можуть бути запуснені (розгорнуті) на будь-якій наявній операційній системі з мінімумом стороннього (або на базі типового) програмного забезпечення. Така система повинна легко копіюватись, її екземпляри без попередніх налаштувань (або з їх мінімумом) залучатись у використання та безперешкодно працювати в будь-якій доступній локальній мережі й самоналаштовуватись для роботи у ній. Додатковою вимогою є можливість віддаленого доступу до цієї системи у глобальній мережі, особливо, в межах її ізольованого сегменту.

Очевидним рішенням в цьому випадку може бути використання однієї із платформ віртуалізації. Проаналізувавши найпопулярніші [18-21, 37-39], зазначимо їх переваги та недоліки.

1. Microsoft Hyper-V. Має глибоку інтеграцію з Windows та підтримує хмарні технології (Azure), але має обмеження функціоналу та підтримки гостьових операційних систем.

2. OracleVirtualBox. Підтримує широкий спектр гостьових операційних систем, має велику кількість функцій та налаштувань, безкоштовний та відкритий код. Недоліком є дещо нижча ефективність для важких завдань у порівнянні з конкурентами.

3. VMware. Відома своєю високою продуктивністю та швидкодією. Надає розширені функціональні можливості, зокрема, широкий спектр

функцій для управління та моніторингу віртуальними машинами. Проте безкоштовна версія обмежена у функціоналі.

4. QEMU. Вільне та відкрите програмне забезпечення, підтримує різні архітектури, але не має досконалого графічного інтерфейсу, є складним для новачків та має порівняно низьку продуктивність.

5. Parallels. Оптимізована для роботи на платформі MacOS, має високу продуктивність на Mac-платформі. Повний функціонал та підтримка деяких гостьових операційних систем доступні лише у комерційних версіях.

Детально ознайомившись із документацією та протестувавши ці платформи, було обрано Oracle VirtualBox з погляду на її сучасність, доступність, відкритість, популярність, функціонал та зручність і простоту у користуванні. І, хоча вона не позбавлена недоліків, в нашому випадку вони не відіграють суттєвої ролі і можуть бути нівельовані особливістю роботи гостьової операційної системи, яку ми будемо використовувати.

Наступним кроком при формуванні цільової розробки став вибір гостьової операційної системи, на якій необхідно розгорнути LMS Moodle. Очевидно, вона повинна бути безкоштовною, сучасною, стабільною, продуктивною, поширеною, з великою кількістю документації, програмного забезпечення, розроблених для неї, і максимально сумісною з LMS Moodle.

Найкраще, на нашу думку, враховуючи вимоги [40] та вище вказані критерії, підходить Ubuntu Server 20.04 [41]. Для порівняння та аналізу нами було також створено іншу версію системи на базі Ubuntu Server 23.04 та останньої на момент розробки версії Moodle 4.2.3+.

2.5 Особливості програмної реалізації та основні режими роботи

«Pocket Moodle»

2.5.1 Створення віртуальної машини

При створенні віртуальної машини було змінено ряд параметрів за замовчуванням для Ubuntu (Linux x64), а саме:

- обсяг оперативної пам'яті встановлений розміром 2 GB (як показали неодноразові експерименти, цього більше, ніж достатньо для подальшої комфортної роботи);

- розмір віртуального жорсткого диску збільшено до 80 GB (проте файл .vdi, який відповідає за нього, є суттєво меншого розміру та збільшується у міру свого заповнення);

- мережевий інтерфейс було змінено на «Проміжний адаптер (BridgedAdapter)», щоб віртуальна машина могла отримати власну IP-адресу в мережі, ніби вона є окремою фізичною машиною, і могла взаємодіяти з іншими пристроями в мережі безпосередньо.

Параметри налагодження наведено на рис. 2.4.

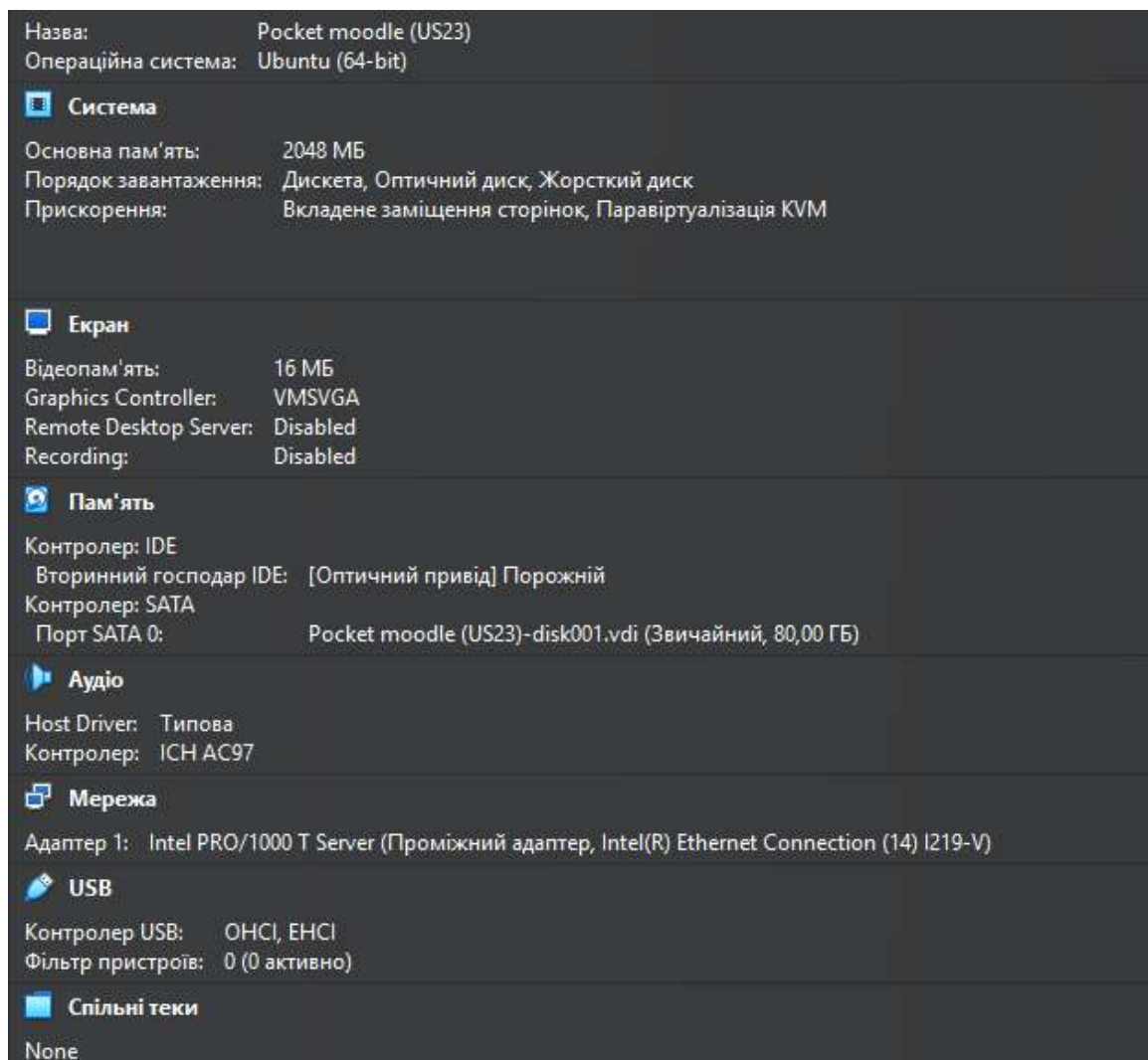
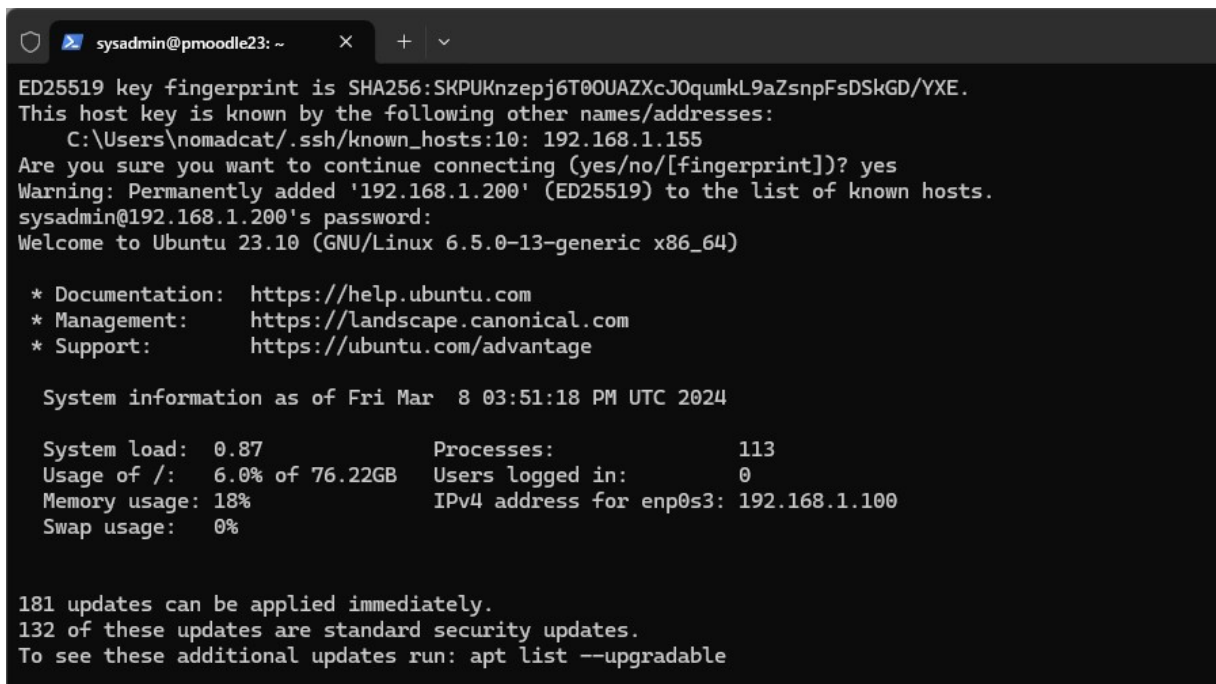


Рисунок 2.4 - Параметри створеної віртуальної машини для Pocket Moodle

2.5.2 Встановлення серверної операційної системи в якості гостьової

Для заощадження ресурсів віртуальної машини було встановлено серверну операційну систему Ubuntu Server лише з інтерфейсом командного рядка (CLI), без використання графічного інтерфейсу користувача (GUI). В процесі встановлення було обрано типову конфігурацію програмних пакетів та «піднятий» сервер SSH для подальшої роботи з системою не безпосередньо через інтерфейс VirtualBox, а з можливістю працювати віддалено, наприклад, через Putty [42] або Powershell (рис. 2.5).

Віртуальний жорсткий диск було розбито лише на два розділи, один із яких – SWAP, розміром 2 GB.



```

sysadmin@pmoodle23: ~
ED25519 key fingerprint is SHA256:SKPUKnzepj6T00UAZxcJ0qumkL9aZsnpFsDSkGD/YXE.
This host key is known by the following other names/addresses:
  C:\Users\nomadcat/.ssh/known_hosts:10: 192.168.1.155
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.1.200' (ED25519) to the list of known hosts.
sysadmin@192.168.1.200's password:
Welcome to Ubuntu 23.10 (GNU/Linux 6.5.0-13-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Fri Mar  8 03:51:18 PM UTC 2024

System load:  0.87          Processes:            113
Usage of /:   6.0% of 76.22GB Users logged in:     0
Memory usage: 18%          IPv4 address for enp0s3: 192.168.1.100
Swap usage:   0%

181 updates can be applied immediately.
132 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable
  
```

Рисунок 2.5 - Вікно терміналу PowerShell з інтерфейсом CLI Ubuntu Server 23.04

2.5.3 Моніторинг та керування

Для зручного віддаленого моніторингу та керування було встановлено та сконфігуровано Webmin (рис. 2.6) [43].

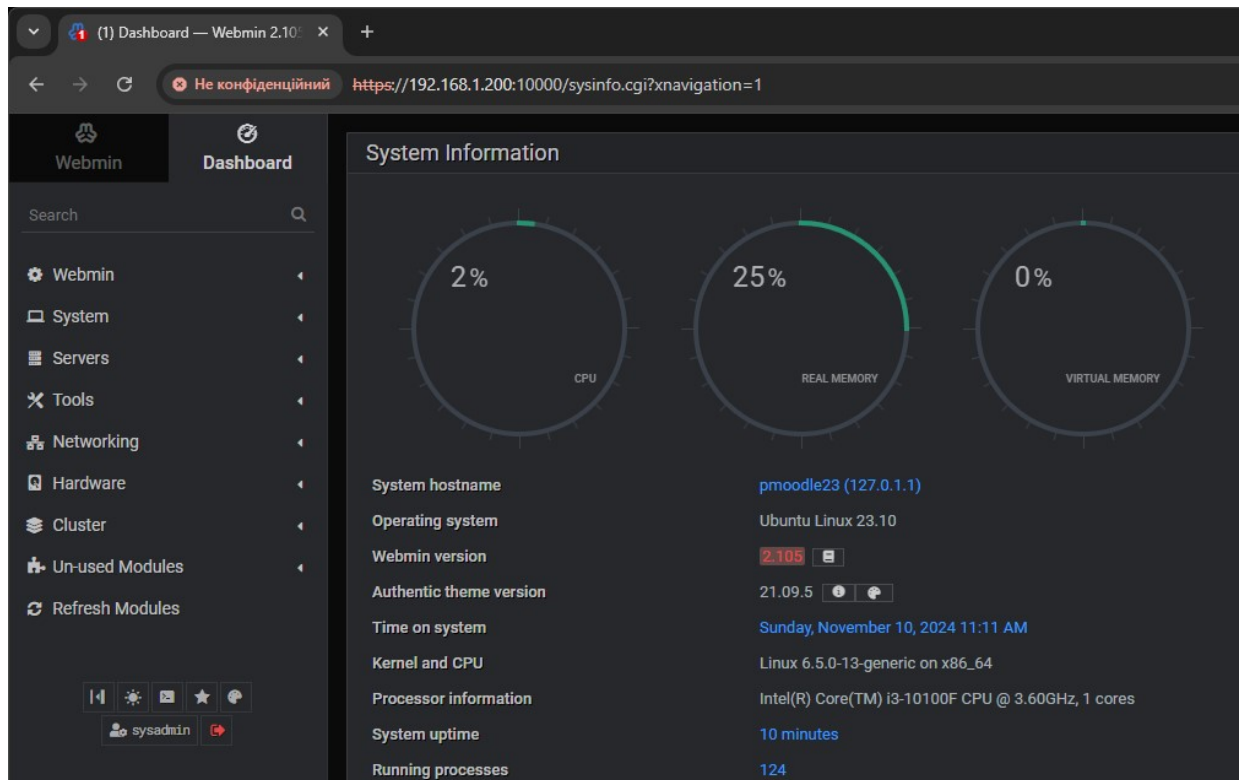


Рисунок 2.6 - Вебінтерфейс системи моніторингу та керування Webmin

Webmin – це вебінтерфейс для адміністрування Unix-подібних операційних систем, що дозволяє керувати системою через браузер. З його допомогою можна змінювати налаштування операційної системи, керувати службами та процесами, конфігурувати мережу, додавати або видаляти користувачів, редагувати файли конфігурації та інше. Webmin спрощує управління сервером для користувачів, які не мають глибоких знань командного рядка, завдяки інтуїтивно зрозумілому інтерфейсу. Це рішення підтримує модульну структуру, що дозволяє додавати додаткові функції через модулі. Webmin поширюється як безкоштовне програмне забезпечення з відкритим кодом, що робить його популярним для адміністрування як приватних серверів, так і корпоративних мереж.

2.5.4 Забезпечення VPN-з'єднання для роботи «Pocket Moodle»

Для того, щоб реалізувати роботу Pocket Moodle на рівні глобальної мережі було встановлено клієнта PPTP, та налагоджено параметри цього

з'єднання через відповідний конфігураційний файл (рис.2.7). З'єднання буде активуватись автоматично при завантаженні системи (прописано у файлі /etc/rs.local).

```

/etc/ppp/peers/scs
pty "pptp 194.44.187.57 --nolaunchpppd"
name rstudent
password 1Qaaaaaa$
remotename PPTP
require-mppe-128
require-mschap-v2
refuse-eap
refuse-pap
refuse-chap
refuse-mschap
noauth
debug
persist
maxfail 0
defaultroute
replacedefaultroute
usepeerdns

```

Рисунок. 2.7 - Конфігураційний файл VPN-з'єднання хостингової операційної системи для Pocket Moodle

Такі параметри також надаються клієнтам системи Pocket Moodle для створення VPN-з'єднань, щоб клієнти могли скористатись системою у глобальній мережі. Приклад наведено у Додатку Б(4).

2.5.5 Стэк LAMP

Після початкового тестування системи на предмет коректної роботи в мережі та встановленого програмного забезпечення, на віртуальній машині було продовжено розгортання необхідних компонентів, а саме стеку LAMP. Стэк LAMP – це набір програмного забезпечення з відкритим вихідним кодом, який використовується для розробки й розгортання вебдодатків. LAMP є популярним рішенням завдяки своїй надійності, продуктивності та можливості адаптуватися до багатьох типів вебпроектів, від простих вебсайтів до складних корпоративних додатків.

Перед встановленням стека LAMP варто оновити репозиторій та доступні пакети програмного забезпечення:

```
sudo apt update; sudo apt upgrade
```

Встановлення вказаного стеку проходить в кілька етапів: встановлення APACHE, встановлення сервера баз даних, встановлення PHP.

Для встановлення APACHE скористаємось командою:

```
sudo apt install apache2
```

Після встановлення він повинен запуснитись автоматично. Якщо ж цього, раптом не відбудеться, можна його запустити наступним чином:

```
sudo systemctl status apache2
```

Для перевірки його роботи необхідно визначити IP-адресу нашої віртуальної машини командою `ifconfig` (рис. 2.8) та ввести її у рядок інтернет-браузера (рис. 2.9).

```
sysadmin@pmoodle23:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.200 netmask 255.255.255.0 broadcast 192.168.1.255
    inet6 fe80::a08:8ff:fe08:808 prefixlen 64 scopeid 0x20<link>
    ether 08:08:08:08:08:08 txqueuelen 1000 (Ethernet)
    RX packets 249799 bytes 366007600 (366.0 MB)
    RX errors 0 dropped 3418 overruns 0 frame 0
    TX packets 95768 bytes 11651578 (11.6 MB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Рисунок 2.8 - Визначення IP-адреси PocketMoodle в локальній мережі

В якості сервера баз даних, як правило, використовується MySQL-сервер, проте ми встановимо MariaDB. MariaDB – це система керування базами даних з відкритим кодом, яка є розгалуженням MySQL. Вона була створена у відповідь на придбання MySQL компанією Oracle. MariaDB зберігає сумісність із MySQL, що дозволяє користувачам легко перейти з MySQL на MariaDB без значних змін у коді або конфігурації. Основні її переваги це: сумісність із MySQL, покращена продуктивність, розширені можливості та активна підтримка спільноти, що забезпечує постійні оновлення і виправлення.



Рисунок 2.9 - Перевірка роботи вебсервера АРАСНЕ

Встановлення БД проведемо за допомогою команди:

```
sudo apt install mariadb-server
```

і перевіримо її статус (рис. 2.10):

```
systemctl status mariadb
```

```

sysadmin@pmoodle23:~$ systemctl status mariadb
● mariadb.service - MariaDB 10.11.8 database server
   Loaded: loaded (/lib/systemd/system/mariadb.service; enabled; preset: enabled)
   Active: active (running) since Sun 2024-11-10 11:15:43 UTC; 42min ago
     Docs: man:mariadb(8)
           https://mariadb.com/kb/en/library/systemd/
  Main PID: 24602 (mariabdb)
    Status: "Taking your SQL requests now..."
     Tasks: 8 (limit: 14696)
    Memory: 100.2M
       CPU: 559ms
    CGroup: /system.slice/mariadb.service
            └─24602 /usr/sbin/mariabdb

```

Рисунок 2.10 - Перевірка статусу сервера бази даних

Після запуску сценарію безпеки після установки командою:

```
sudo mysql_secure_installation,
```

необхідно створити базу даних та користувача для майбутнього Moodle, хоча це можна зробити і безпосередньо перед установкою Moodle:

```
sudo mysql -u root -p
```

Створимо порожню базу даних, користувача з паролем, та надамо йому повний доступ до цієї бази даних:

```
CREATE DATABASE moodledb;
```

```
CREATE USER 'moodle'@'localhost' IDENTIFIED BY 'new_password_here';
```

```
GRANT ALL ON moodledb.* TO 'moodle@localhost' IDENTIFIED BY 'user_password_here' WITH GRANT OPTION;
```

Сервер PHP – це вебсервер, який підтримує виконання коду, написаного мовою програмування PHP, для створення динамічного вебконтенту. Сервер PHP інтерпретує PHP-код на серверній стороні, генерує HTML-сторінки та надсилає їх користувачеві через браузер. Для подальшої підтримки виконання php-сценаріїв нам необхідний PHP-сервер з набором відповідних компонентів.

Тому встановлюємо:

```
sysadmin@pmoodle23:~$ sudo apt install graphviz aspell ghostscript clamav php8.2-pspell php8.2-curl php8.2-gd php8.2-intl php8.2-mysql php8.2-xml php8.2-xmlrpc php8.2-ldap php8.2-zip php8.2-soap php8.2-mbstring
```

Далі необхідно активувати модуль APASHP:

```
sudo a2enmod php8.2
```

та перезавантажити APACHE:

```
sudo systemctl restart apache2.
```

Щоб протестувати PHP-скрипти за допомогою сервера Apache, потрібно створити файл info.php у кореневому вебкаталозі:

```
sudo mcedit /var/www/html/info.php
```

заповнивши його вмістом:

```
/var/www/html/info.php
<?php phpinfo(); ?>
```

Після того, як файл буде збережений, для перевірки роботи PHP-сервера в адресний рядок браузера потрібно ввести: **<IP-адреса>/info.php**. Результат цього відображено на рис. 2.11.

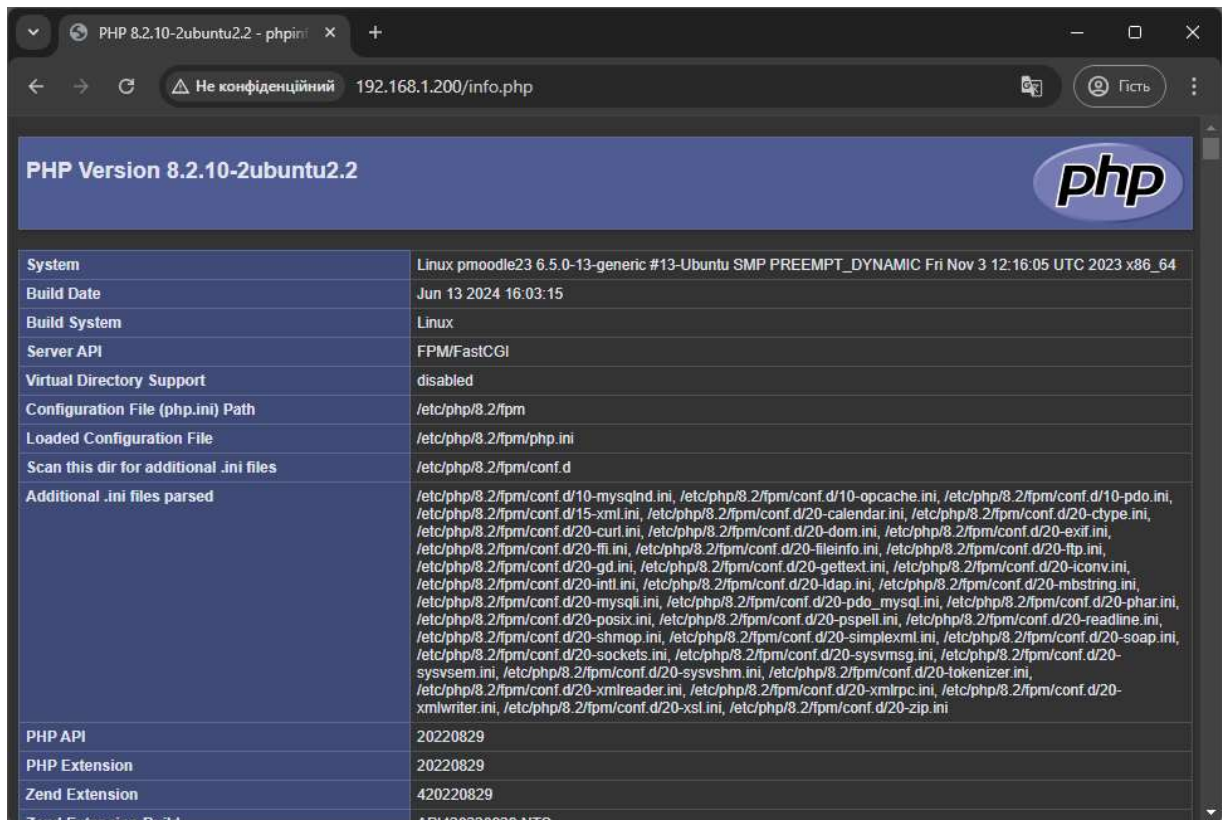


Рисунок 2.11 - Перевірка роботи PHP-сервера

2.5.6 Встановлення LMS Moodle

Завантажуємо останню стабільну версію LMS Moodle:

```
wget https://download.moodle.org/download.php/stable421/moodle-latest-421.tgz
```

після чого розпаковуємо завантажений архів

```
tar -zxvf moodle-latest-421.tgz,
```

переміщуємо Moodle у вебкаталог хостингу

```
sudo mv moodle /var/www/html/moodle,
```

створюємо новий каталог даних для LMS Moodle

```
sudo mkdir /var/www/html/moodledata
```

та налаштуємо права доступу

```
sudo chown -R www-data:www-data /var/www/html/moodle/
```

```
sudo chmod -R 755 /var/www/html/moodle/
```

```
sudo chown www-data /var/www/html/moodledata
```

Надалі, щоб запустити вебресурс, необхідно буде лише додати до IP-адреси в браузері «/moodle».

Встановлення LMS Moodle розпочнемо, ввівши в адресному рядку браузера:

```
192.168.1.200/moodle/install.php
```

де 192.168.200 – це IP-адреса нашого хостингу в локальній мережі, після чого розпочнеться процес встановлення (див. рис. 2.12).

В процесі встановлення, щоб правильно налаштувати середовище та забезпечити повноцінну роботу системи необхідно вказати:

- мову, яка буде використовуватись у процесі інсталяції та в інтерфейсі платформи;
- тип бази даних (наприклад, MySQL, MariaDB, PostgreSQL), хост бази даних, ім'я користувача, пароль і назву бази даних;
- каталог на сервері для зберігання файлів, що завантажуються користувачами, він повинен мати відповідні права доступу;
- URL-адресу, за якою буде доступна LMS Moodle;
- під час інсталяції створюється обліковий запис адміністратора платформи: ім'я користувача, пароль та електронна адреса для керування системою;
- назва навчальної платформи, опис сайту, вибирається початкова тема оформлення та початкові налаштування інтерфейсу;
- налаштування для SMTP-сервера, щоб LMS Moodle могла надсилати електронні листи для реєстрацій, відновлення паролів та інших сповіщень.

Встановлення

Мова інтерфейсу

Виберіть мову

Виберіть мову ТІЛЬКИ для процесу встановлення. Ви зможете вибрати мову для сайту та користувачів пізніше.

Мова інтерфейсу

[Далі >](#)




Рисунок 2.12 - Початок встановлення Moodle

Ці параметри забезпечують базове налаштування LMS Moodle для роботи. Після завершення інсталяції можна додатково конфігурувати модулі, плагіни та інші розширення для гнучкого налаштування платформи під потреби навчального процесу.

2.5.6 Додаткові налаштування платформи та інтерфейсу

Маючи найвищі привілеї в LMS Moodle після встановлення варто (рекомендовано) задати ряд додаткових параметрів, які б дозволили реєструватись іншим користувачам (що заборонено за замовчуванням) та відключити журналювання окремих подій, що можуть негативно вплинути на швидкість роботи з банком питань. Такі налаштування можна змінити в адміністративній панелі «Керування сайтом», скориставшись можливістю пошуку за ключовим словом «статистика».

За шляхом вебкаталогу Moodle знаходиться файл `config.php`, в якому необхідно замінити параметр `$CFG->wwwroot`, який варто змінити таким чином, щоб не вказувати конкретну IP-адресу ресурсу, а вона вказувалась автоматично, при включенні PocketMoodle в ту або іншу мережу, адже в кожній мережі і навіть при кожному включенні системи, та й в тій самій мережі вона може змінюватись. Цю корекцію варто зробити так, як вказано на рис. 2.13.

```

/var/www/html/moodle/config.php [-M--] 4 L:[ 1+ 9 10/ 31] *(179
<?php // Moodle configuration file

unset($CFG);
global $CFG;
$CFG = new stdClass();

$CFG->dbtype      = 'mariadb';
$CFG->dblibrary   = 'native';
$CFG->dbhost      = 'localhost';
$CFG->dbname      = 'moodle';
$CFG->dbuser      = 'moodle';
$CFG->dbpass      = '1Qaaaaaa';
$CFG->prefix      = '';
$CFG->dboptions   = array (
    'dbpersist' => 0,
    'dbport'    => '',
    'dbsocket'  => '',
    'dbcollation' => 'utf8mb4_unicode_ci',
);

$CFG->wwwroot     = 'http://' . $_SERVER['HTTP_HOST'] . '/moodle';
$CFG->dataroot    = '/var/www/moodledata';
$CFG->admin       = 'admin';

```

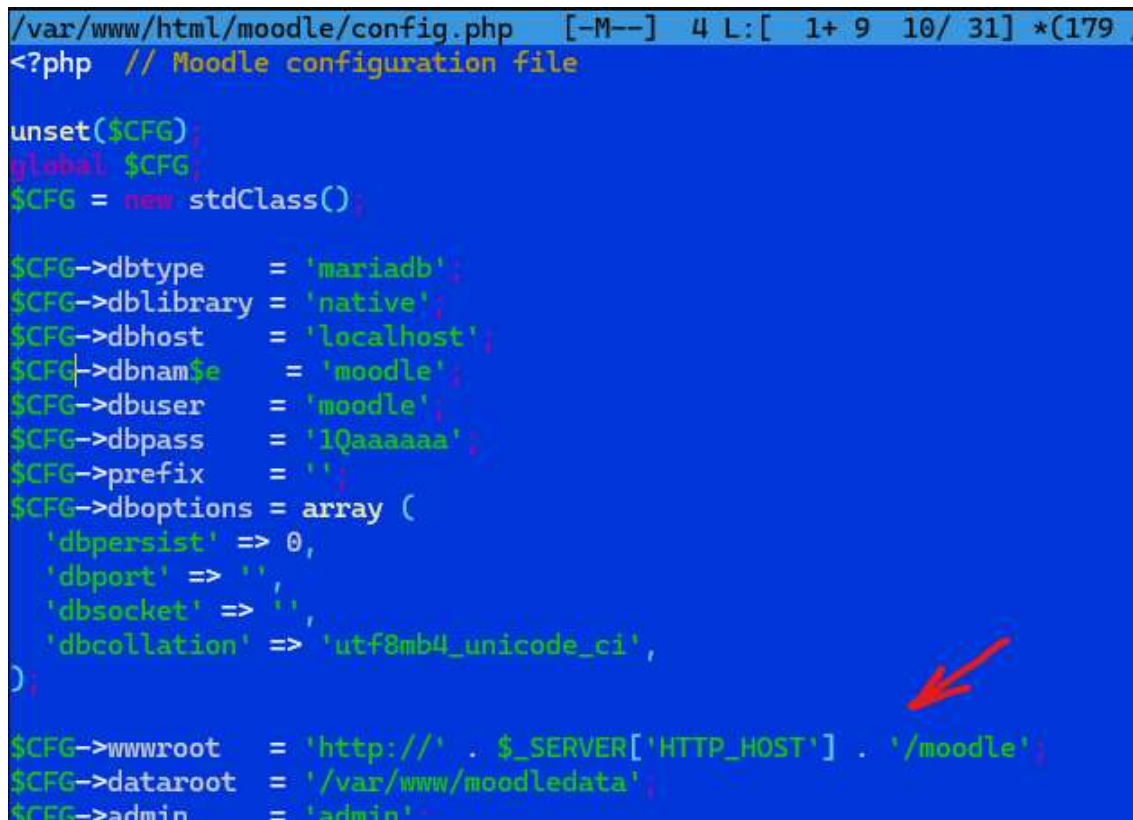


Рисунок 2.13 - Зміна параметра \$CFG->wwwroot

Після запуску системи у типового користувача можуть виникати труднощі при визначенні ним IP-адреси системи у локальній мережі, та у VPN-мережі. Необхідно також змінити паролі адміністратора за замовчуванням для посилення безпеки, та знати облікові дані, які необхідно передати клієнтам для віддаленого під'єднання до системи.

Таку інформацію можна вивести на стартовому екрані (CLI), модифікувавши файл /etc/issue. (див. рис. 2.14) і отримати стартовий екран для різних версій нашої розробки (див. рис. 2.15, рис. 2.16).

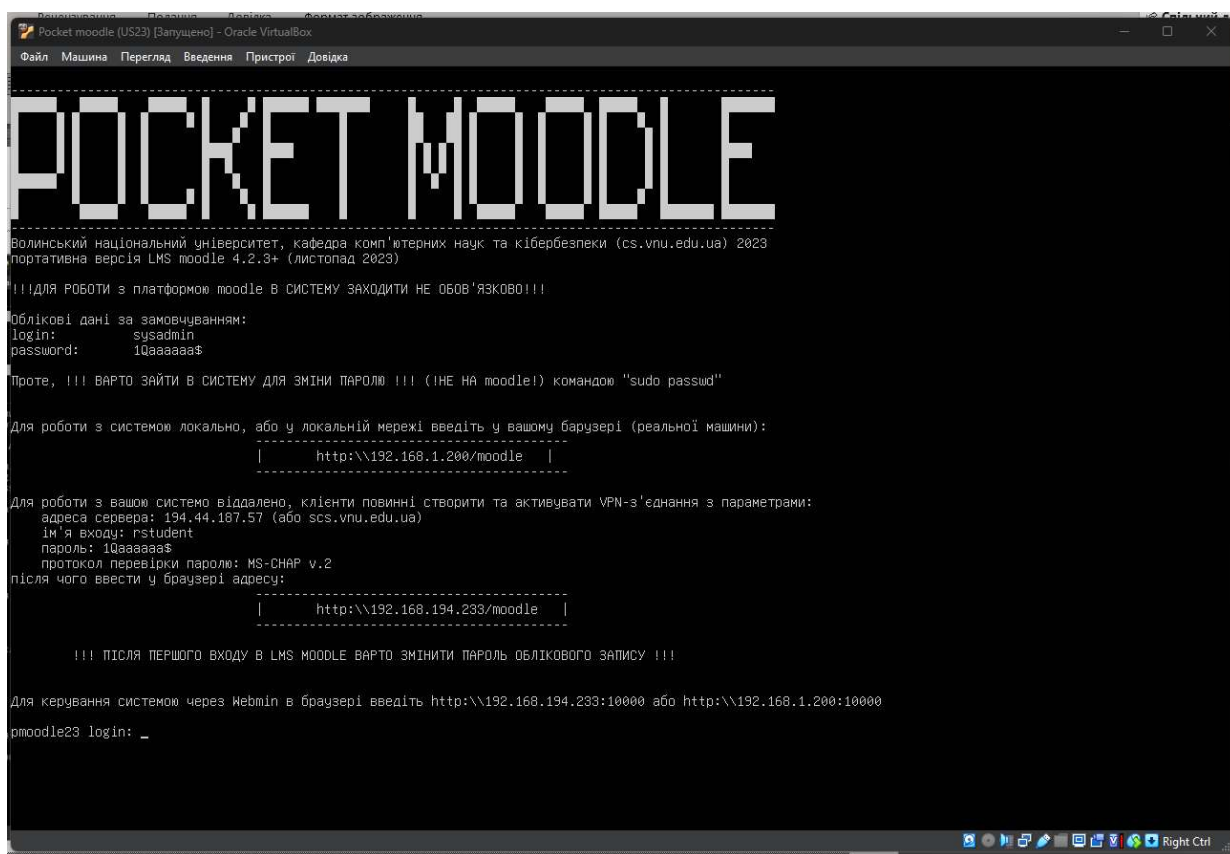


Рисунок 2.15 - Стартовий екран віртуальної машини PocketMoodle на Ubuntu server 23.04

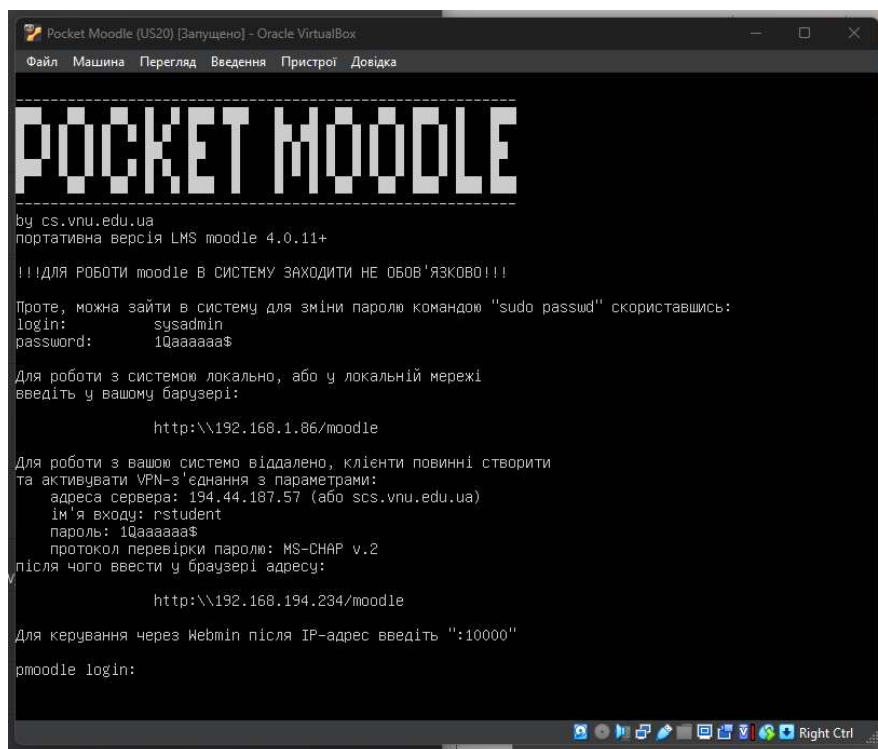


Рисунок 2.15 Стартовий екран віртуальної машини PocketMoodle на Ubuntu server 20.04

2.6 Організація тестування та налагодження

Для тестування використовувались декілька наборів апаратних засобів, як носії платформи віртуалізації OracleVirtualBox:

- CPU: i3-10100, MB:GIGABYTE H410M S2H V3, RAM: 8GB DDR4, SSD: Goodram SSDPR-CL100-480-G3 (480 Гб, SATA-III),Windows 10 Pro x64;

- CPU: i3-10100F, MB: ASUS Prime H510, RAM: 16GB DDR4, SSD: Samsung970EVOPlus250GB2B2QEXM7 (250 Гб, NVMe)Windows 11 Pro x64;

- CPU: i5-6300U, MB: HP 8079, RAM: 16GB DDR4, SSD: Samsung 970 EVO Plus 500 GB (MZ-V7S500BW) (250 Гб, NVMe), Windows 11 Pro x64.

Тестування відбувалось в провідній (100Mbps) та безпроводній (54-450 Mbps) мережах з використанням виділених каналів Internet 40 Mbps та 100 Mbps.

В якості платформи віртуалізації використовувалась Oracle VirtualBox 7.0.2 та 7.0.16 + ExtentionPack.

Процес тестування передбачав наступні кроки.

1. Перевірка на стабільність та функціонал гостьової операційної системи UbuntuServer здійснювалась покроково в процесі її розгортання та налагодження. Зокрема, включала в себе: стабільну роботу ядра операційної системи, безпроблемний процес оновлення, робота в різних локальних мережах, доступ до Інтернету, функціонування SSH, ApacheHTTP, PHP-сервера, MySQL (MariaDB).

2. Перевірка системи моніторингу та керування Webmin.

3. Встановлення та налагодження VPN-сервера PPTPна базі Windows 2003 ServerSESP2 штатними засобами ОС з використанням публічної IP-адреси.

4. Перевірка клієнта VPN.

5. Перевірка на відповідність системних вимог та функціонал LMS Moodle.

6. Перевірка роботи у локальній мережі з реальними клієнтами LMS Moodle.

7. Перевірка роботи у глобальній мережі з реальними клієнтами LMS Moodle з використанням VPN.

Збірка протестована в мережі лабораторій кафедри комп'ютерних наук та кібербезпеки Волинського національного університету імені Лесі Українки та окремими викладачами кафедри у глобальній мережі. Середня кількість підключених клієнтів варіювалась від 5 до 25.

За допомогою Webmin було проаналізовано вплив кількості користувачів на витрату оперативної пам'яті, що склало в середньому 50-80 MB на кожного нового користувача, залежно від його активності. Враховуючи пам'ять, зайняту ядром операційної системи (приблизно 300 MB) та розмір SWAP-розділу, віртуальна машина з 2 GB здатна забезпечити комфортну роботу не менше 30 клієнтів. А якщо оператор збільшить розмір оперативної пам'яті віртуальної машини за рахунок реальної, кількість клієнтів можна збільшити.

2.7. Рекомендації по використанню та впровадженню

Враховуючи досвід тестування та задачі, які ставились в процесі розробки, можна сформулювати ряд рекомендацій для повнофункціонального використання даного програмного засобу, що включають в себе апаратні засоби та вид ОС.

Наявність апаратних засобів, що відповідають мінімальним системним вимогам для роботи платформи віртуалізації Oracle VirtualBox:

- CPU: 2 Core,1GHz;
- Graphic Card: integrated;
- RAM: 4 GB DDR2 800MHz;
- HDD (SSD): 20GB;
- LAN: 100Mbps;

- WLAN: 54Mbps.

Встановлена одна із операційних систем, що підтримує Oracle VirtualBox7.x (як материнська):

- Windows: Windows 8/10/11, Windows Server 2012 / 2016 / 2019 / 2022;
- macOS: macOS 10.13 (High Sierra) або новіші;
- Linux: Більшість сучасних дистрибутивів (Ubuntu, Fedora, CentOS, Debian тощо);
- Solaris: Solaris 11.

Перенесення PocketMoodle з однієї системи на іншу зводиться до переносу папки віртуальної машини і в подальшому лише встановлення в якості проміжного адаптера наявного у новій системі через графічний інтерфейс налагодження віртуальної машини. Така папка в початковому стані має розмір приблизно 6-7 GB і містить менше десяти файлів. Іншим способом переносу може слугувати експортування (одночасно із стисненням) у вигляді єдиного файлу .ova і подальшим імпортом на новій системі. Такий файл в початковому стані має розмір приблизно 2-3 GB.

PocketMoodle може бути використана як окремими невеликими структурними підрозділами закладів освіти різного рівня для навчання здобувачів, так і окремими викладачами, або вчителями, як для навчального процесу, так і для вивчення самої LMS на рівні адміністратора системи. Слід враховувати при використанні кількість клієнтів та можливості апаратних засобів, які використовуються для роботи PocketMoodle.

Більш детальні інструкції для роботи з PocketMoodle подані у вказівках користувачеві (Додаток Б).

ВИСНОВКИ

В процесі дослідження було створено збірку портативної LMS Moodle – «PocketMoodle», яку можна використовувати як в локальній, так і в глобальній мережі, або в ізольованих сегментах глобальної мережі. Для цього було проаналізовано та використано особливості функціонування однієї із платформ віртуалізації Oracle VirtualBox, зокрема, розглянуті мережеві можливості та підтримка засобів роботи хостингів. Детально розглянуто процес розгортання та налагодження хостингу на основі серверної операційної системи UbuntuServerLTS кількох різних версій. Реалізовано підтримку VPN для розширення функціоналу кінцевої розробки.

Слід зазначити, що, оскільки розроблена система не використовує безпосередньо публічної IP-адреси, то неможливо скористатись протоколом https, проте, враховуючи особливості VPN, мобільність системи, періодичне її включення та невелику кількість користувачів, загрози практично відсутні. Файл-архів віртуальної машини є порівняно великим файлом, а після імпортування він збільшується в декілька разів, збільшується його розмір і під час роботи системи. Але на даному етапі розвитку апаратних засобів це не є критично, оскільки для користувачів тепер доступні швидкі та ємкі носії інформації. При експортуванні (архівуванні) віртуальної машини розмір її в декілька разів зменшується.

Зміна VPN-сервера для роботи у глобальній мережі вимагає втручання в гостьову операційну систему віртуальної машини, що є недоліком такої портативної збірки.

До вдосконалення «PocketMoodle» можна віднести використання VirtualBox Main API для розробки спеціалізованого графічного інтерфейсу, що полегшить роботу некваліфікованих користувачів з системою. Також варто звернути увагу на розробку інструменту автоматичної оптимізації файлу віртуального носія з метою зменшення його розміру.

Скористатись системою можуть викладачі, або вчителі, які не бажають бути залежним від розгорнутих на хостингах платформ дистанційного навчання. Користувачі можуть вільно переносити таку систему з одних комп'ютерів на інші комп'ютери, з однієї мережі в іншу мережу, мати адміністративні права до неї, вільно клонувати її, вивчати, а, при потребі, запускати її вдома або віддалено, завчасно повідомивши параметри з'єднання здобувачам, щоб ті могли нею скористатись.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Що таке LMS та як підібрати собі LMS-систему - Блог про email та інтернет-маркетинг. *Блог про email та інтернет-маркетинг*. URL: <https://sendpulse.ua/blog/learning-management-system> (дата звернення: 25.11.2024).
2. 10 найкращих LMS для освітніх закладів - Oksim. *Oksim - IT допомога*. URL: <https://www.oksim.ua/2023/12/08/10-najkrashhih-lms-dlya-osvitnih-zakladiv/> (дата звернення: 25.11.2024).
3. Learning Management System Market Size, Share | Growth 2032. *Fortune Business Insights™ | Global Market Research Reports & Consulting*. URL: <https://www.fortunebusinessinsights.com/industry-reports/learning-management-system-market-101376> (date of access: 25.11.2024).
4. Що таке система управління навчанням (LMS)? - Oksim. *Oksim - IT допомога*. URL: https://www.oksim.ua/2023/12/08/shho-take-sistema-upravlinnya-navchannjam-lms/#Параметри_розгортання_LMS (дата звернення: 25.11.2024).
5. Що таке хостинг та як вибрати його тип залежно від сайту - Блог про email та інтернет-маркетинг. *Блог про email та інтернет-маркетинг*. URL: <https://sendpulse.ua/blog/how-to-choose-hosting> (дата звернення: 25.11.2024).
6. HOSTiQ.ua. *Хостинг в Україні від HOSTiQ – найкращий український хостинг*. URL: <https://hostiq.ua/ukr/info/what-is-hosting/> (date of access: 25.11.2024).
7. Серверні операційні системи. Види та призначення *Корисна інформація та матеріали у сфері інформаційних технологій* URL: <https://lvivservice.com.ua/sysadmin/serverni-operatsiyni-systemy-vydy-i-pryznachennya/> (date of access: 25.11.2024).

8. Linux Хостинг VS Windows Хостинг. Що обрати для розміщення сайту? • Блог Hostpro. *Блог Hostpro*. URL: <https://hostpro.ua/blog/ua/linux-and-windows-website-hosting/> (дата звернення: 09.12.2024).

9. Використання віртуалізації у Linux - налаштування та використання віртуальних машин - ефективний спосіб оптимізації роботи в сучасному світі інформаційних технологій. *Mediacom*. URL: <https://mediacom.com.ua/vikoristannya-virtualizatsii-u-linux-nalashtuvannya-ta-vikoristannya-virtualnix-mashin-efektivnij-sposib-optimizatsii-roboti/> (дата звернення: 26.11.2024).

10. What is steak LAMP? – LAMP Stack Explained – AWS. *Amazon Web Services, Inc*. URL: <https://aws.amazon.com/ru/what-is/lamp-stack/> (дата звернення: 26.11.2024).

11. Що таке NGINX і навіщо він потрібний. *Вікіпедія серверов и хостинга*. URL: <https://vps.ua/wiki/ukr/nginx/> (дата звернення: 26.11.2024).

12. NGINX: продуктивний і популярний веб-сервер | студія Brander. *Створення і розробка інтернет-магазинів від Brander*. URL: <https://brander.ua/technologies/nginx> (дата звернення: 26.11.2024).

13. Що таке NGINX? Чим nginx кращий за інші веб-сервери? – *Хостинг-компанія FREEhost.UA*. URL: <https://freehost.com.ua/ukr/faq/wiki/chto-takoe-nginx/> (date of access: 26.11.2024).

14. Docker: новий підхід до розробки та впровадження програмного забезпечення – *Хостинг-компанія FREEhost.UA*. URL: <https://freehost.com.ua/ukr/faq/wiki/docker-novij-podhod-k-razrobotke-i-vnedreniju-programmnogo-obespechenija/> (date of access: 26.11.2024).

15. Що таке Docker і навіщо він? *QualityAssuranceGroup*. URL: <https://qagroup.com.ua/publications/shcho-take-docker-i-navishcho-vin/> (дата звернення: 26.11.2024).

16. Віртуалізація: що це, як працює, огляд та порівняння платформ | Блог Colobridge. *blog.colobridge.net*. URL:

<https://blog.colobridge.net/uk/2023/06/virtualization-platforms-ua/> (дата звернення: 26.11.2024).

17. Системи віртуалізації на основі Hyper-V, VMWare, Proxmox ⇒ 20 років на ринку ІТ послуг для бізнесу - Techexpert.ua. *TechExpert*. URL: <https://techexpert.ua/it-services/building-of-virtualization-systems/> (дата звернення: 26.11.2024).

18. Windows 10 Hyper-V System Requirements. Microsoft Learn: Build skills that open doors in your career. URL: <https://learn.microsoft.com/en-us/virtualization/hyper-v-on-windows/reference/hyper-v-requirements>.

19. VirtualBox | Віртуалізація | Oracle Україна *Oracle Україна* URL: <https://www.oracle.com/ua/virtualization/virtualbox/> (дата звернення: 26.11.2024).

20. VMware by Broadcom - Cloud Computing for the Enterprise. *VMware by Broadcom - Cloud Computing for the Enterprise*. URL: <https://www.vmware.com/> (date of access: 09.12.2024).

21. Welcome to QEMU's documentation! – QEMU documentation. *QEMU*. URL: <https://www.qemu.org/docs/master/> (date of access: 09.12.2024).

22. Що таке VPN: як працює і який обрати? | KR. Laboratories. *KR. Laboratories*. URL: <https://kr-labs.com.ua/blog/shho-take-vpn-i-chym-vin-korysnyj> (дата звернення: 09.12.2024).

23. Як обрати найкращий протокол VPN у 2024 - Surfshark. *Surfshark*. URL: <https://surfshark.com/uk/blog/vpn-protocols> (дата звернення: 09.12.2024).

24. Що таке VPN – основні поняття і принцип роботи мережі - Planet VPN. *Planet VPN*. URL: <https://freevpnplanet.com/ua/blog/shho-take-vpn-osnovni-ponyattya-i-printsip-roboti-merezhi/> (дата звернення: 09.12.2024).

25. XAMPP сервер: встановлення і налаштування в ОС Linux | KR. Laboratories. *KR. Laboratories*. URL: <https://kr-labs.com.ua/blog/xampp-setup-and-install-in-linux> (дата звернення: 09.12.2024).

26. Download XAMPP. *XAMPP Installers and Downloads for Apache Friends*. URL: <https://www.apachefriends.org/download.html> (date of access: 09.12.2024).

27. WampServer. *WampServer*. URL: <https://www.wampserver.com/> (date of access: 09.12.2024).

28. MAMP & MAMP PRO - your local web development solution for PHP and WordPress development. *MAMP & MAMP PRO - your local web development solution for PHP and WordPress development*. URL: <https://www.mamp.info/en/windows/> (date of access: 09.12.2024).

29. Laragon - portable, isolated, fast & powerful universal development environment for PHP, Node.js, Python, Java, Go, Ruby. *Laragon - portable, isolated, fast & powerful universal development environment for PHP, Node.js, Python, Java, Go, Ruby*. URL: <https://laragon.org/> (date of access: 10.12.2024).

30. AMPPS – WAMP, MAMP and LAMP Stack. *AMPPS – WAMP, MAMP and LAMP Stack*. URL: <https://www.ampps.com/> (date of access: 10.12.2024).

31. MoodleDocs. *MoodleDocs*. URL: https://docs.moodle.org/400/en/Main_page (date of access: 10.12.2024).

32. Moodle в Україні: Що таке Moodle | Moodle.org. Moodle challenge. URL: <https://moodle.org/mod/page/view.php?id=8174> (дата звернення: 09.12.2024).

33. Булатецький, В., Булатецька, Л. і Павленко, Ю. 2024. ПОРТАТИВНА LMS MOODLE З ВІДДАЛЕНИМ ДОСТУПОМ. Прикладні проблеми комп'ютерних наук, безпеки та математики. 2 (Бер 2024), 70–78. URL: <https://apcssm.vnu.edu.ua/index.php/Journalone/article/view/19>.

34. Булатецький В., Булатецька Л., Книш В. Особливості комунікації у мережах з порушенням зовнішніх каналів зв'язку. Міжнародна благодійна конференція together united: науковці проти війни: матеріали конф., м. Луцьк, 20 трав. 2022 р. URI: <https://evnuir.vnu.edu.ua/handle/123456789/20551>.

35. Windows installation using XAMPP – MoodleDocs. *MoodleDocs*. URL: https://docs.moodle.org/2x/pl/Windows_installation_using_XAMPP (date of access: 09.12.2024).

36. Pickavance M., Turner B. Best virtual machine software of 2024. *TechRadar*. URL: <https://www.techradar.com/best/best-virtual-machine-software> (date of access: 09.12.2024).

37. The best virtual machine tools for Windows: Run a PC inside your PC. *PCWorld*. URL: <https://www.pcworld.com/article/1936055/the-best-virtualization-tools-for-windows.html> (date of access: 09.12.2024).

38. Documentation – Oracle VirtualBox. *Oracle VirtualBox*. URL: <https://www.virtualbox.org/wiki/Documentation> (date of access: 09.12.2024).

39. Resources & System Requirements | Parallels Desktop for Mac. *Parallels: Mac & Windows Virtualization, Remote Application Server, Mac Management Solutions*. URL: <https://www.parallels.com/products/desktop/resources/> (date of access: 09.12.2024).

40. Installing Moodle - MoodleDocs. *MoodleDocs*. URL: https://docs.moodle.org/20/en/Installing_Moodle#Software (date of access: 09.12.2024).

41. Ubuntu server system requirements. Linux Tutorials – Learn Linux Configuration. URL: <https://linuxconfig.org/ubuntu-server-system-requirements> (date of access: 09.12.2024).

42. Download PuTTY - a free SSH and telnet client for Windows. *Download PuTTY - a free SSH and telnet client for Windows*. URL: <https://www.putty.org/> (date of access: 09.12.2024).

43. Webmin. *Webmin*. URL: <https://webmin.com/> (date of access: 09.12.2024).

ДОДАТОК А

ТЕХНІЧНЕ ЗАВДАННЯ НА РОЗРОБКУ ПРОГРАМНОГО ЗАСОБУ «ROCKET MOODLE»

1. ЗАГАЛЬНІ ВІДОМОСТІ

1.1. Назва системи

Повне найменування системи: переносима платформа дистанційного навчання «RocketMoodle».

1.2. Найменування організацій-учасників робіт

Виконавець: здобувач вищої освіти факультету інформаційних технологій і математики Волинського національного університету імені Лесі Українки – Павленко Юлія Степанівна.

1.3. Передумови для проведення робіт

Платформа дистанційного навчання LMS Moodle – це безкоштовна та відкрита система, яка завдяки широкому набору інструментів та середовищ, здатна розширити можливості викладання та навчання, набула великої популярності і є найпоширенішою системою подібного типу у світі. До переліку базових вимог для її роботи входять вебсервер з підтримкою PHP та сервер баз даних (як правило MySQL). Їх функціонування може забезпечити відповідне обладнання власного сервера з публічною IP-адресою або послуги сторонніх хостингів. Проте це пов'язано з певними вартісними затратами та складністю обслуговування кожного екземпляру платформи Moodle. Зокрема, складнощі виникають у випадку необхідності переносу платформи на інше обладнання або хостинг, особливо, коли хостинг недоступний в екстремальних умовах або у разі виникнення перебоїв зв'язку з глобальною мережею при працездатній регіональній мережі. Недоліком такого підходу є неможливість роботи у випадку відсутності інтернет з'єднання в ізольованих локальних мережах або у мережах, де недоступний хостинг, на якому розгорнуто LMS Moodle.

Серед очевидних вирішень описаної проблеми є розгортання платформи Moodle таким чином, щоб була можливість її оперативного перенесення (в тому числі фізичного) в стиснені часові терміни з метою її тимчасового або періодичного включення в склад окремих локальних або ізольованих мереж, та забезпечення взаємодії з нею клієнтів цих мереж, в тому числі і віддалено. Причому таке перенесення, резервне копіювання не повинне вимагати відкористувачів особливих знань в розгортанні та налагодженні платформи Moodle.

Тому необхідна система, проста в розгортанні та запуску, швидкому і простому перенесенні, резервному копіюванні, з можливістю віддаленого доступу (роботі у глобальній мережі), яка не вимагає від користувачів спеціальних вмінь, окрім користування платформою Moodle, кросплатформна та без вартісних вкладень. Метою дослідження є створення цілісного продукту, завантаживши який, користувач швидко і без сторонньої допомоги може ним скористатись в локальній мережі, а при потребі – віддалено, отримати всі адміністративні права в платформі Moodle і перемістити платформу без особливих труднощів.

1.4. Порядок оформлення передачі замовнику результатів робіт по розробці програмного забезпечення

Програмне забезпечення оформлене у вигляді єдиного файлу для імпорту з розширенням .ova, який може бути записаний на носій, або розміщений на загальнодоступній платформі у мережі. Комплектується короткою вмонтованою інструкцією для користувача.

2. ПРИЗНАЧЕННЯ СИСТЕМИ

Програмна система призначена для використання у якості портативної платформи дистанційного навчання з легким копіюванням, розгортанням та перенесенням.

3. ВИМОГИ ДО СИСТЕМИ

3.1. Вимоги до функціональних характеристик

Програмний продукт повинен володіти наступними функціональними характеристиками:

- містити повнофункціональну платформу Moodle;
- працювати в локальній та глобальній мережах;
- підтримувати роботу клієнтів, кількість яких співрозмірна з кількістю здобувачів в одній, або кількох академічних групах типового закладу вищої освіти;
- допускати легке копіювання, перенесення та розгортання;
- працювати в переважній більшості операційних систем персональних комп'ютерів (повинна бути кросплатформною);
- не вимагати спеціальних знань та навичок для роботи з мережею та серверною операційною системою, або хостингом;
- вимагати мінімум стороннього програмного забезпечення без складних налагоджень;
- містити у собі (або у вказівках користувачеві) лаконічні та водночас вичерпні коментарі, зрозумілі користувачеві під час використання.

3.2. Технічні вимоги

Програмна розробка повинна являти собою мінімальний набір файлів, доступного для копіювання та перенесення розміру, що легко розгортається у будь-якій поширеній сучасній операційній системі і не вимагає специфічного та високопродуктивного апаратного забезпечення.

3.3. Вимоги до експлуатаційних характеристик

В розробці повинна бути можливість створення електронних курсів, реєстрації клієнтів з розмежуванням прав та інший базовий функціонал, притаманний типовій LMS Moodle, яка розміщена на стаціонарному хостингу. Така система повинна працювати як в локальній так і в глобальній мережі.

3.4. Вимоги до технічного забезпечення

Програмна розробка розрахована на функціонування при наступному наборі технічних засобів:

Мінімальна апаратна конфігурація комп'ютера:

CPU: 2 Core,1GHz;

Graphic Card: integrated;

RAM: 4 GB DDR2 800MHz;

HDD (SSD): 20GB;

LAN: 100Mbps;

WLAN: 54Mbps.

4. ОСНОВНІ РЕЖИМИ ФУНКЦІОНУВАННЯ ПРОГРАМНОГО ЗАСОБУ

Функціонування реалізованої програмної системи з точки зору користувача можна описати наступним чином:

1. Користувач встановлює платформу віртуалізації (одноразово).
2. Користувач імпортує файл системи в платформу віртуалізації (одноразово).
3. Користувач обирає проміжний адаптер (одноразово).
4. Користувач запускає програмний продукт.
5. Користувач входить в систему через-інтернет браузер.
6. Користувач передає клієнтам параметри під'єднання.
7. Якщо необхідно, користувач здійснює експорт системи після її завершення роботи з нею та її вимкнення.

5. ПОРЯДОК ПРИЙОМУ СИСТЕМИ В ЕКСПЛУАТАЦІЮ

Робота приймається у випадку відсутності помилок у роботі та її ефективності. Система перевіряється на відповідність усім вимогам. При знаходженні недоліків у роботі розробка допрацьовується.

ДОДАТОК Б

ВКАЗІВКИ КОРИСТУВАЧЕВІ

Використовується наперед створений файл-образ (.ova) віртуальної машини для імпортування з базовими необхідними налаштуваннями.

<https://drive.google.com/drive/folders/1CV-8N6yfHhXrhn5Pn9rfZAxMGovLd9z7?usp=sharing>

1. ВСТАНОВЛЕННЯ І ПОЧАТКОВЕ НАЛАШТУВАННЯ (ПОТРІБНЕ ЛИШЕ НА ПОЧАТКОВОМУ ЕТАПІ)

1.1. Імпорт образу віртуальної машини

Необхідно встановити Oracle VirtualBox та Extention Pack до неї (<https://www.virtualbox.org/wiki/Downloads>).

Запустити Oracle VirtualBox Manager (рис. Б.1).

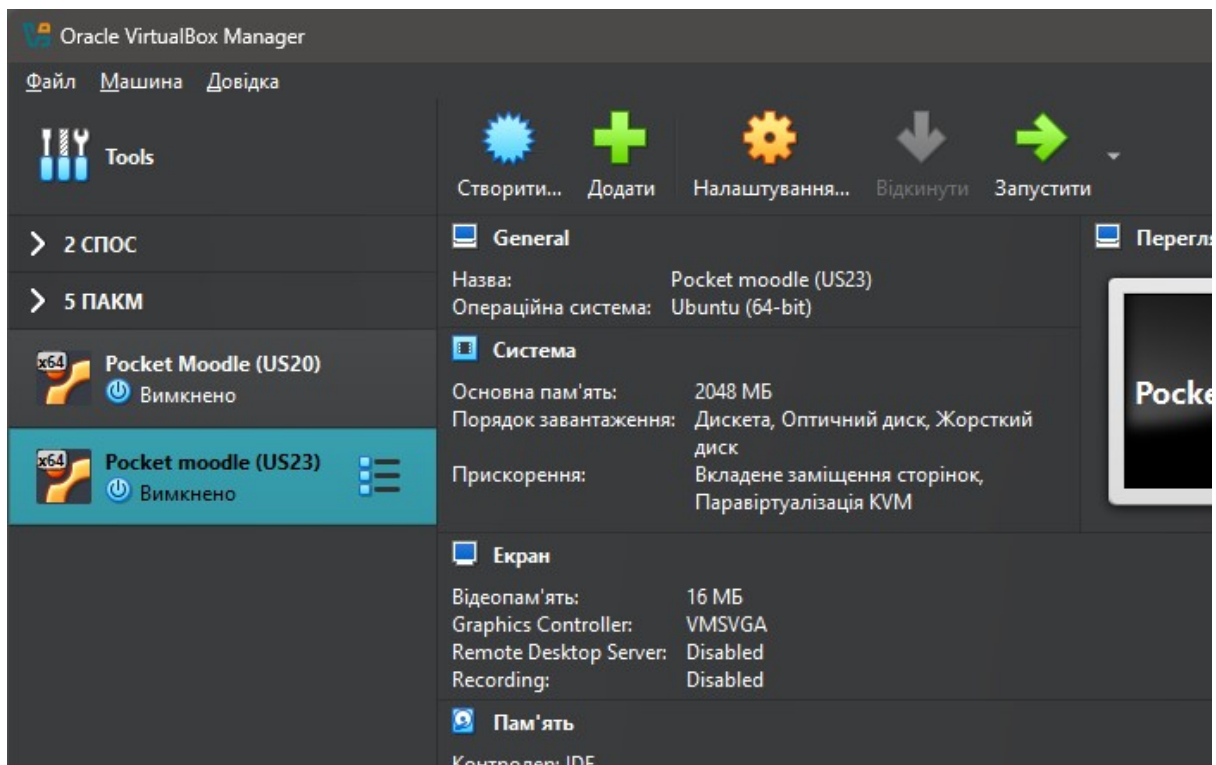


Рисунок Б.1 - Знімок екрану вікна Oracle VirtualBox Manager

Викликати вікно імпорту через меню, або за допомогою [Ctrl+i], де обрати файл-образ віртуальної машини .ova (рис. Б.2). Здійснити імпорт.

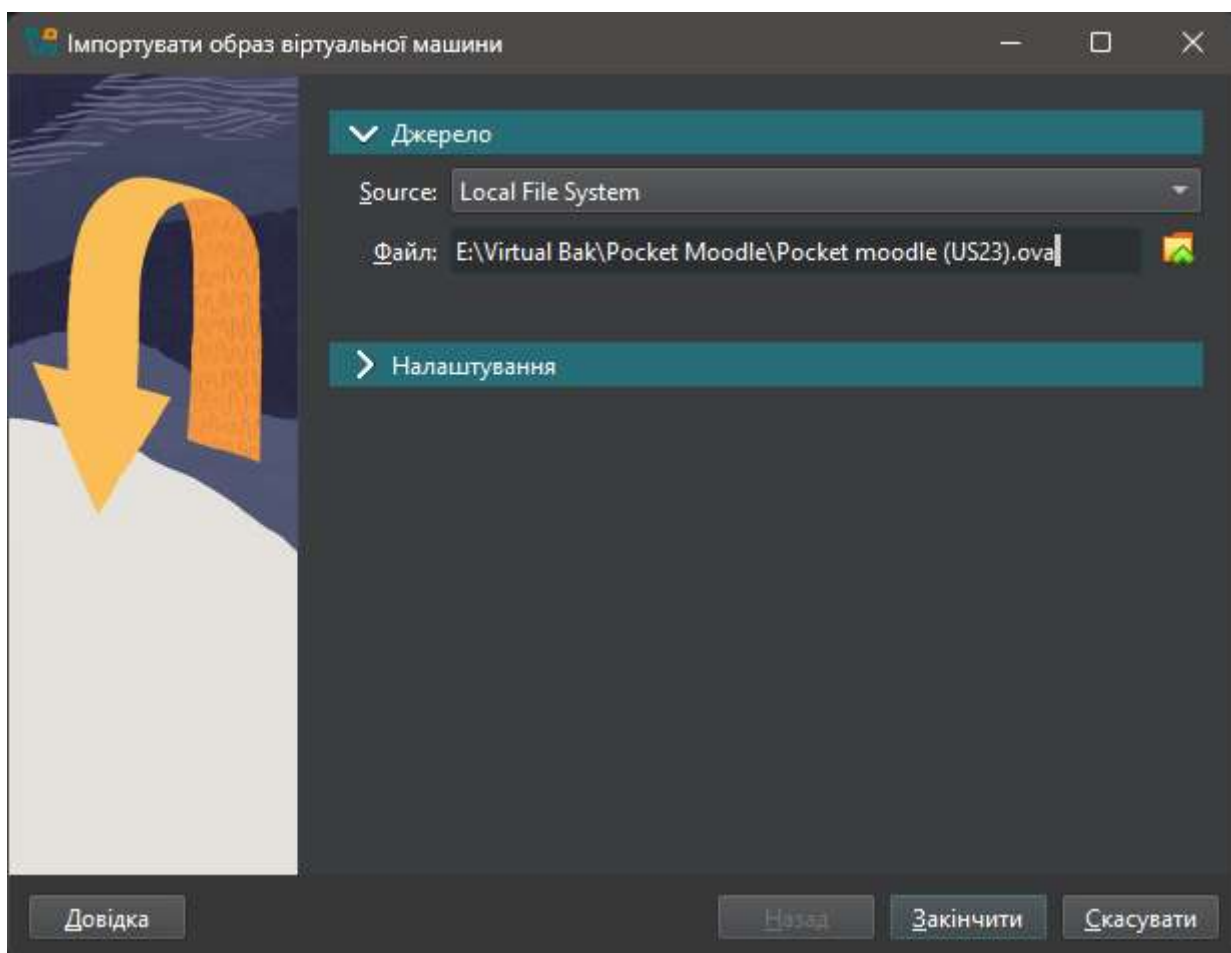


Рисунок Б.2 - Знімок екрану вікна імпорту образу віртуальної машини

1.2 Перевірка (корекція) налагоджень віртуальної машини

Відкрити налаштування імпортованої віртуальної машини (рис. Б.3).

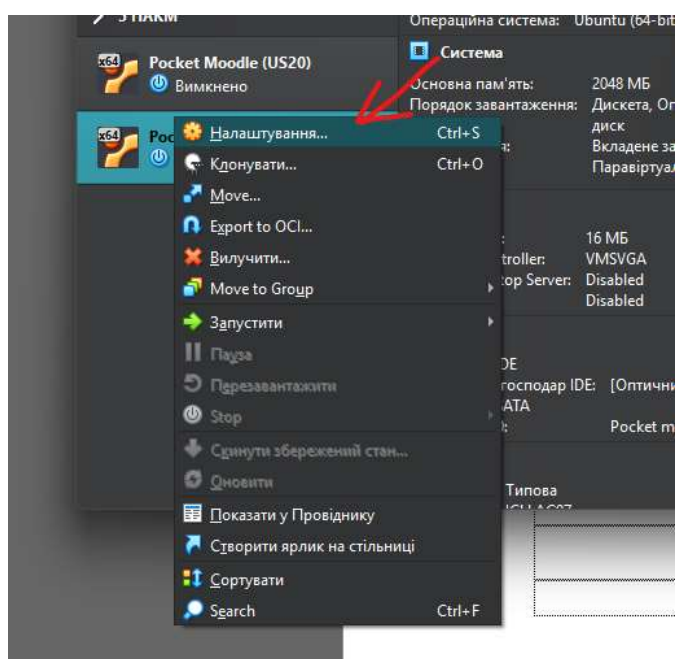


Рисунок Б.3 - Знімок екрану вікна налаштувань віртуальної машини

Перевірити (змінити) важливі параметри: **1 – Мережа, 2 – Під'єднання до: (Проміжний адаптер), 3 – <Назва>** (обирається адаптер, через, який підключена реальна система до мережі). **3** – найважливіший з них, оскільки в різних системах використовуються різні типи з'єднань, і саме він вимагає втручання і виправлення (рис. Б.4).

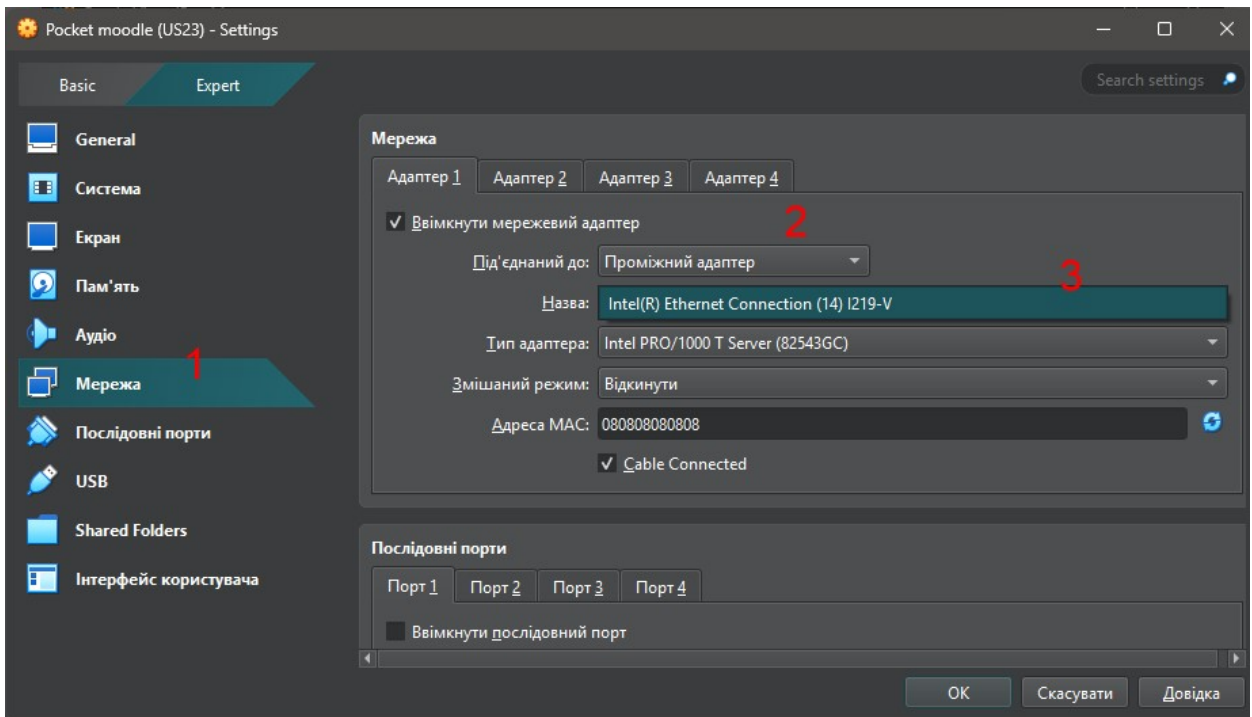


Рисунок Б.4 – Вікно налаштувань мережевих параметрів віртуальної машини

1.3 Запуск віртуальної машини

Запустити віртуальну машину, як показано на рис. Б.5.

1.4 Визначення облікових даних та IP-адрес віртуальної машини

Одразу після завершення процесу завантаження на екрані віртуальної машини можна побачити:

1. Облікові дані для входу в гостьову операційну систему (або через SSH) – пароль варто змінити на власний при першому вході (рис. Б.6).

2. Адресу, за якою можна звернутись до PocketMoodle у локальній мережі.

3. Адресу, за якою можна звернутись до PocketMoodle у глобальній мережі, за умови активації у клієнта VPN з'єднання, яке буде описано нижче (рис. Б.6).

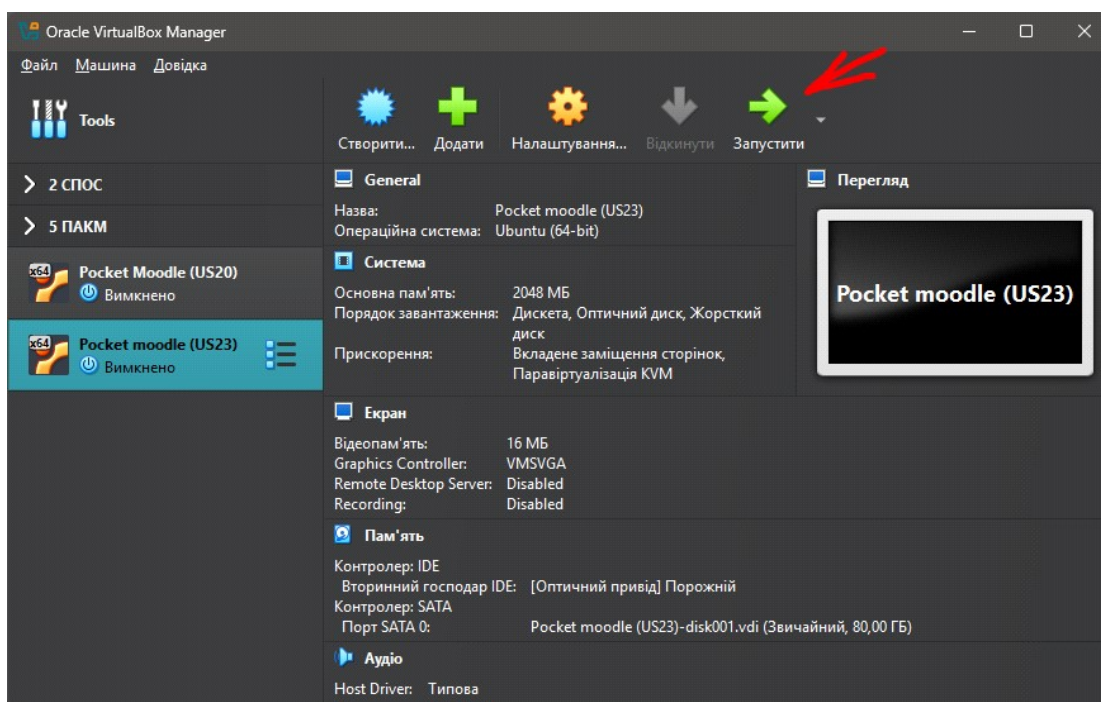


Рисунок Б.5 - Запуск віртуальної машини

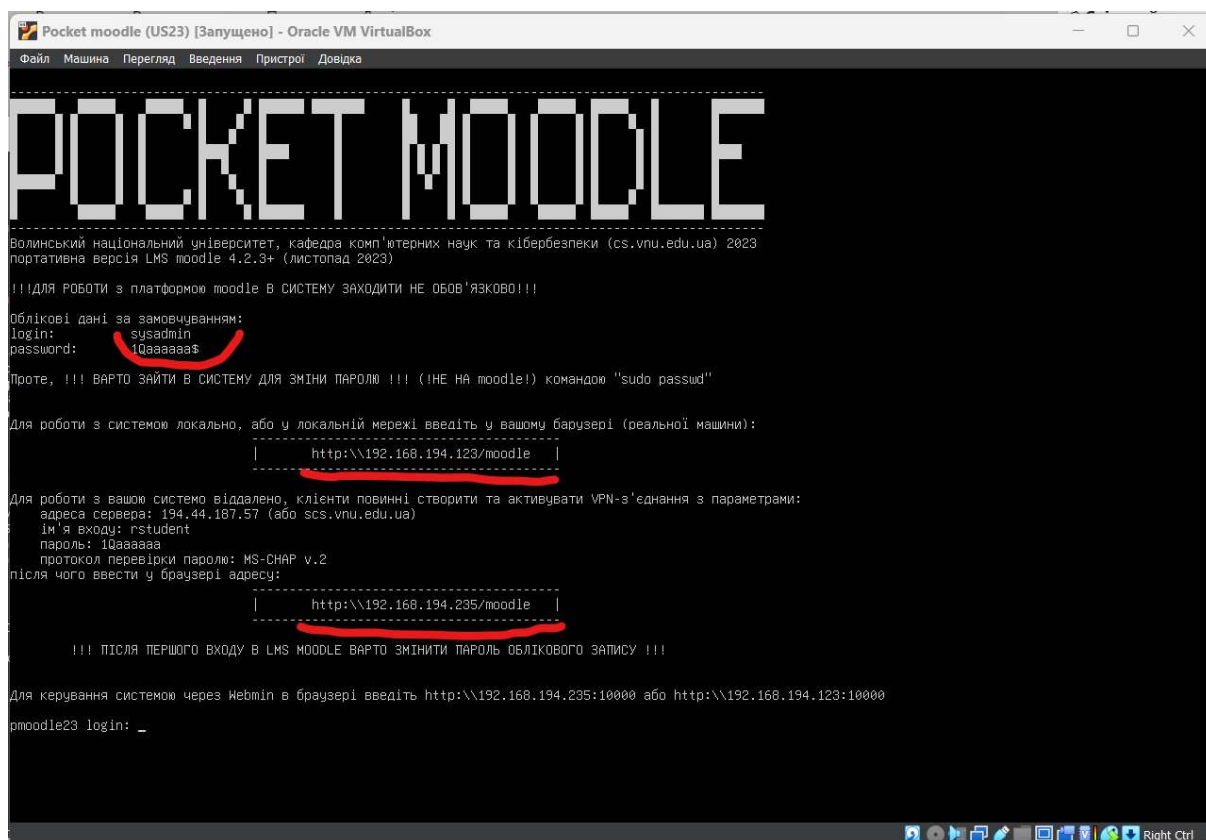


Рисунок Б.6- Визначення облікових даних та IP-адрес віртуальної машини

На екрані можемо побачити IP-адресу (рис. Б.6), яку було призначено віртуальній машині і яку надалі можна використовувати для звертання до неї локально через браузер, в тому числі і клієнтам у локальній мережі (рис. Б.7).

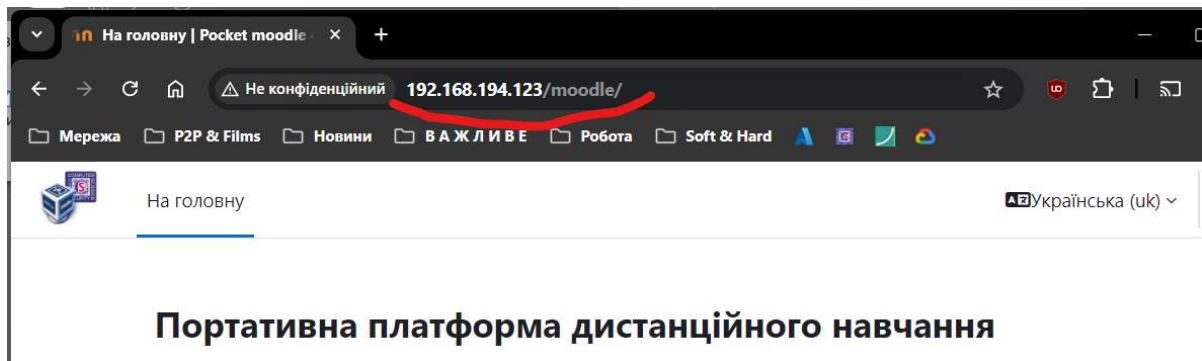


Рисунок Б.7 - Знімок екрана запуску PocketMoodle

Другу вказану адресу можна повідомити клієнтам, які хочуть отримати доступ до системи віддалено через VPN. Також їм необхідно повідомити параметри з'єднання, які теж вказано на стартовому екрані.

1.5 Вхід під користувачем на віртуальну машину за допомогою PowerShell на реальній машині де встановлено віртуальну, або по локальній мережі

Іноді виникає необхідність віддалено працювати з віртуальною машиною. Для цього можна використати засіб PowerShell, що входить до складу ОС Windows (рис. Б.8).

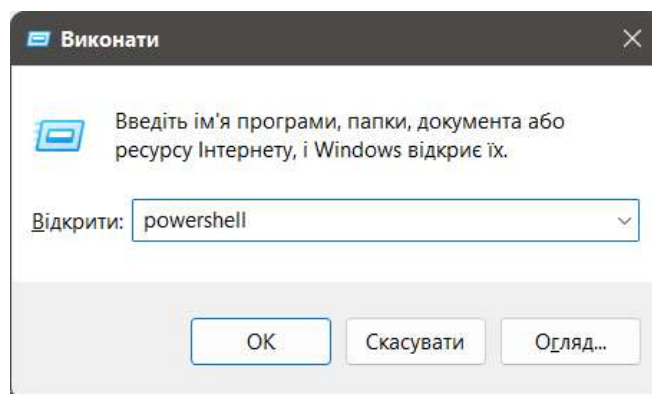


Рисунок Б.8 - Вікно запуску PowerShell

Для цього використовують: `ssh користувач@адреса` (рис. Б.9).



```

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

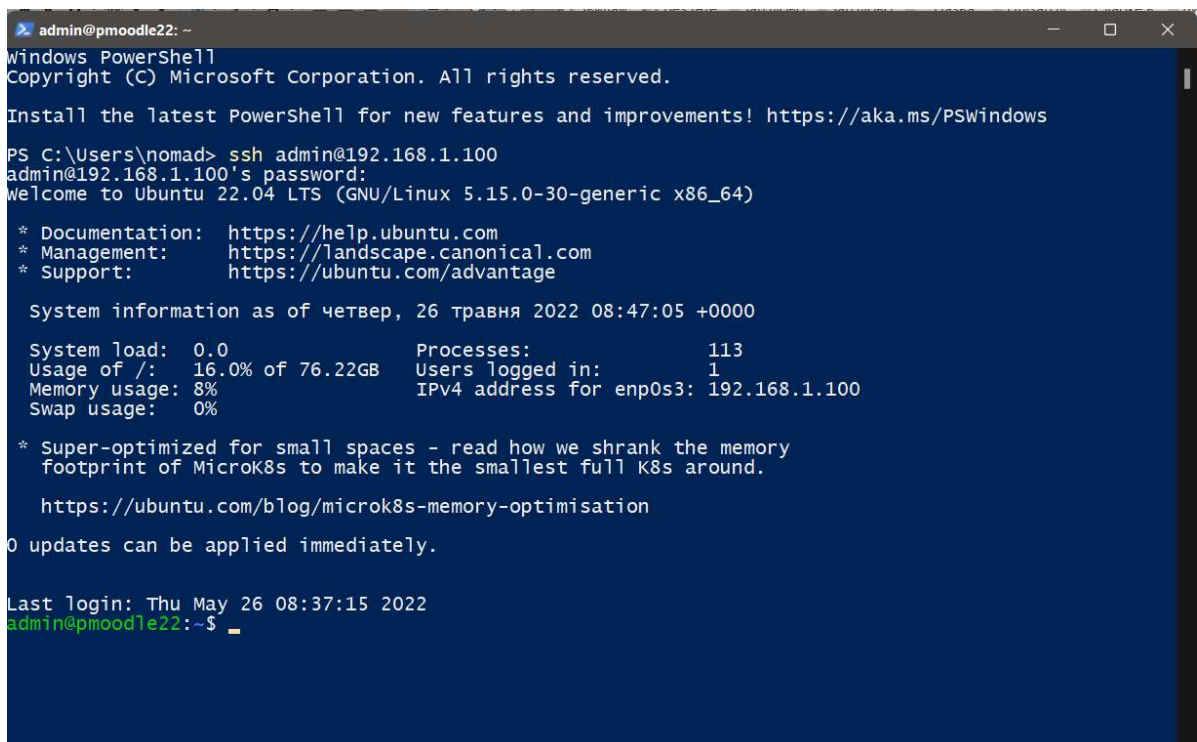
Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\nomad> ssh admin@192.168.1.100

```

Рисунок Б.9 - Вікно запуску PowerShell

Після введення пароля отримаємо середовище для введення команд операційної системи віртуальної машини (рис. Б.10).



```

admin@pmoodle22: ~
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\nomad> ssh admin@192.168.1.100
admin@192.168.1.100's password:
Welcome to Ubuntu 22.04 LTS (GNU/Linux 5.15.0-30-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:   https://landscape.canonical.com
 * Support:      https://ubuntu.com/advantage

System information as of четверг, 26 травня 2022 08:47:05 +0000

System load:  0.0               Processes:    113
Usage of /:   16.0% of 76.22GB   Users logged in: 1
Memory usage: 8%               IPv4 address for enp0s3: 192.168.1.100
Swap usage:  0%

 * Super-optimized for small spaces - read how we shrank the memory
   footprint of MicroK8s to make it the smallest full K8s around.
   https://ubuntu.com/blog/microk8s-memory-optimisation

0 updates can be applied immediately.

Last login: Thu May 26 08:37:15 2022
admin@pmoodle22:~$

```

Рисунок Б.10 – Середовище для введення команд операційної системи

Такий спосіб роботи вважають зручнішим, оскільки він дозволяє простіше працювати з буфером обміну за допомогою правої клавіші миші.

1.6 Зміна пароля для користувача-адміністратора операційної системи

Оскільки кожен викладач повинен мати свій пароль, його варто змінити: *sudo passwd:*

```

Last login: Thu May 26 08:57:15 2022
admin@pmoodle22:~$ sudo passwd ←
[sudo] password for admin:
New password:
Retype new password:
passwd: password updated successfully
admin@pmoodle22:~$

```

1.7 Активація VPN-з'єднання

В запропонованій віртуальній машині вже є створене VPN-з'єднання, яке активується автоматично. Але для віддалених клієнтів його необхідно створити та активувати вручну, щоб отримати доступ до системи у глобальній мережі. Іншими словами потрапити у ту саму мережу, де знаходиться і система PocketMoodle.

2. ВИКОРИСТАННЯ ВИКЛАДАЧЕМ

Вхід у систему, якщо викладач працює локально, можна здійснювати через браузер ввівши адреси:

192.168.1.100/moodle,

(де **192.168.1.100** – визначена IP-адреса віртуальної машини у локальній мережі, вона може змінитись при її перезапуску).

Або, якщо викладач працює віддалено:

192.168.194.236/moodle,

де **192.168.194.236** – визначена IP-адреса у VPN, вона змінюється кожного разу після активації VPN-з'єднання у віртуальній машині і, тим більше при її перезапуску, коли необхідно повторювати активацію VPN.

3.ВИКОРИСТАННЯ КЛІЄНТАМИ (СТУДЕНТАМИ)

3.1 Створення та активація VPN-з'єднання на клієнтських машинах

Наступні параметрами надаються викладачем, або вони наперед відомі:

Адреса VPN сервера: *194.44.187.57*

Ім'я з'єднання: *<будь-яке>*

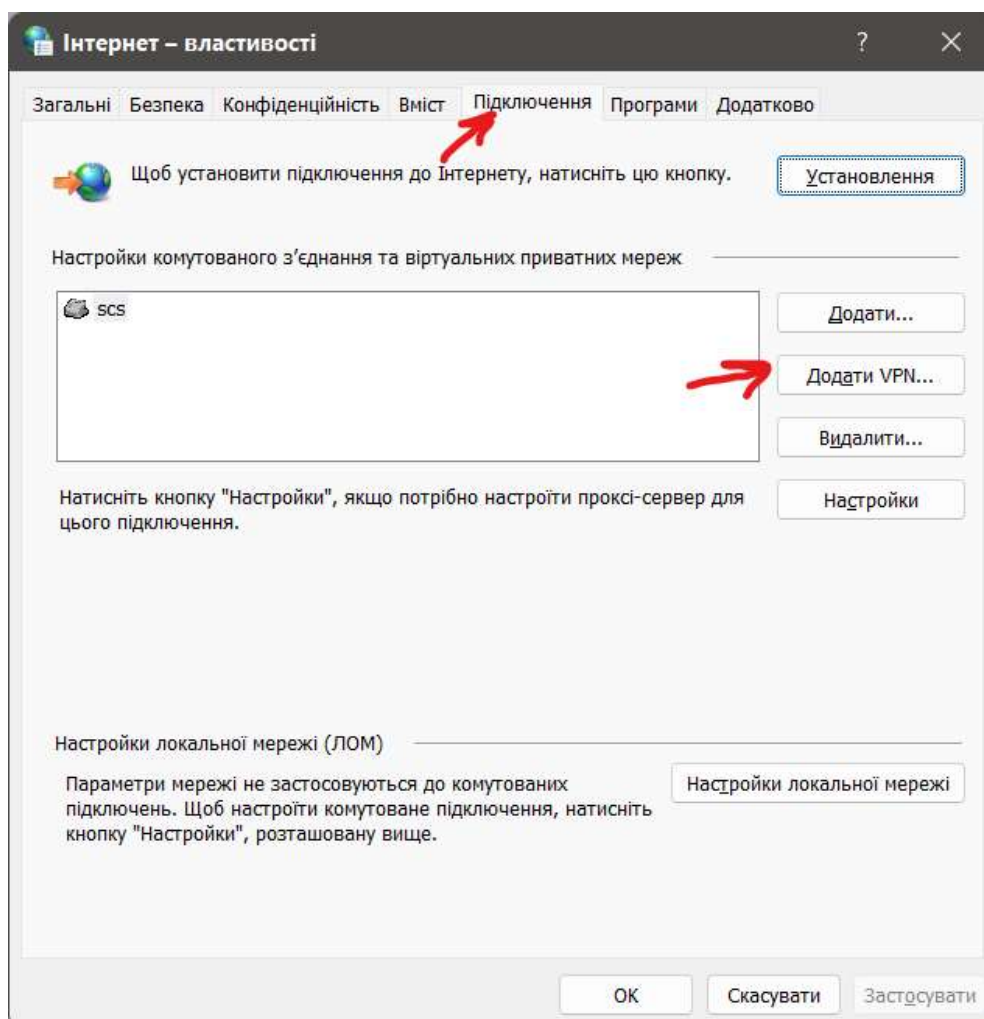
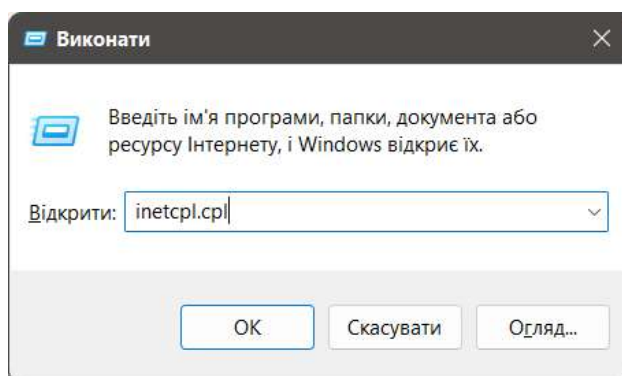
Ім'я користувача: *rstudent*

Пароль: *1Qaaaaaa\$* (якщо необхідно, то вказати перевірку пароля MS-CHAP v.2)

Далі студент у браузері вводить:

<адресу повідомлену викладачем>/moodle

1. СТВОРЕННЯ ТА АКТИВАЦІЯ VPN-З'ЄДНАННЯ (здійснюється по кроках згідно рисунків, наведених нижче)



← Створити підключення до віртуальної приватної мережі (VPN)

Введіть потрібну адресу в Інтернеті


Ця адреса надається адміністратором мережі.

Адреса в Інтернеті:

Ім'я місця призначення:


Використовувати смарт-картку

Запам'ятати мої облікові дані



 Дозволити іншим користувачам використовувати це підключення
Цей параметр дозволяє використовувати це підключення будь-якому користувачу, який має доступ до комп'ютера.


Інтернет – властивості


Загальні | Безпека | Конфіденційність | Вміст | Підключення | Програми | Додатково

 Щоб установити підключення до Інтернету, натисніть цю кнопку.

Налаштування комутованого з'єднання та віртуальних приватних мереж

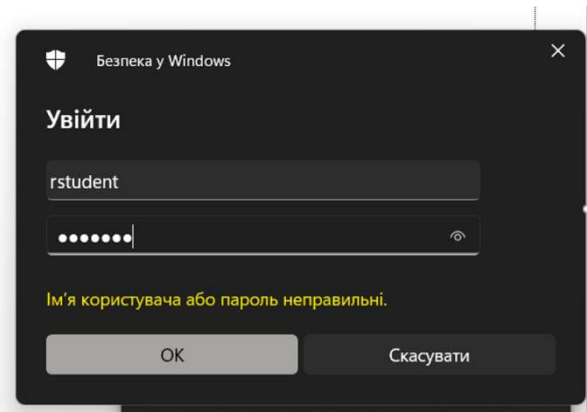
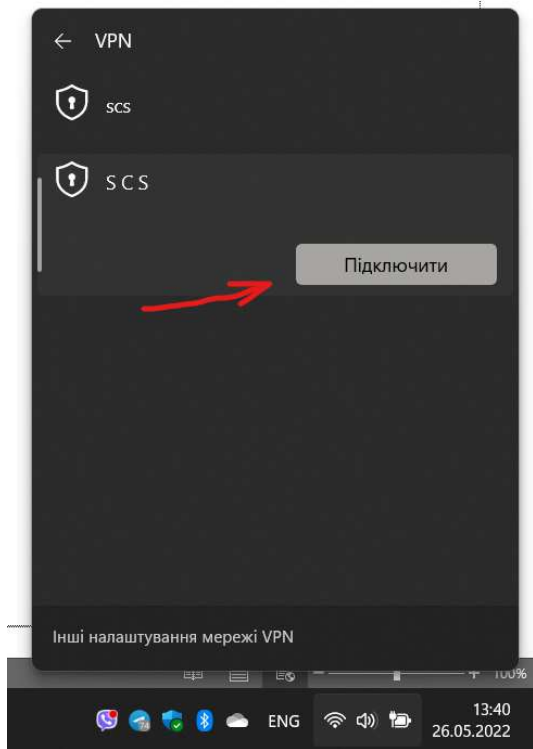
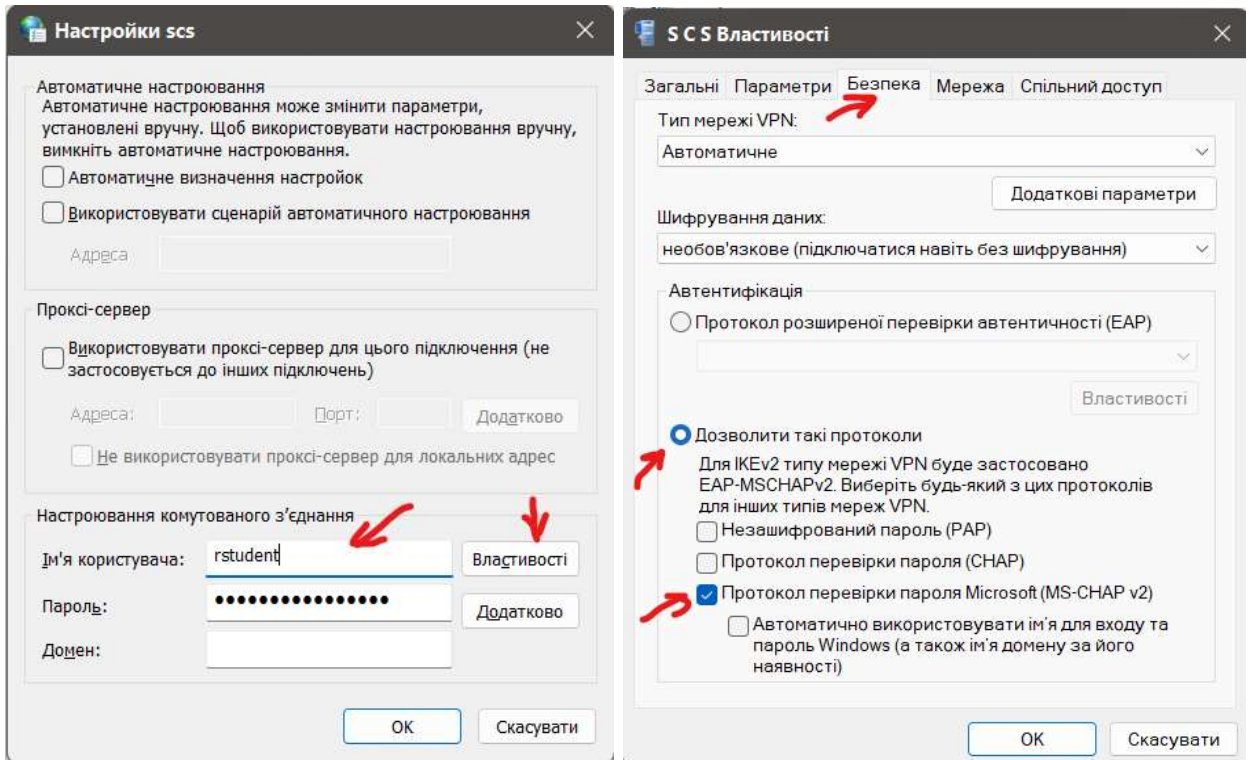
 SCS 

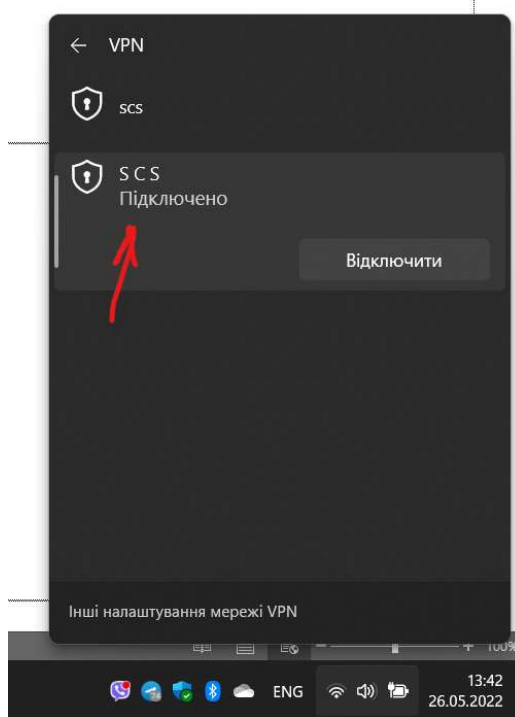
 SCS

Натисніть кнопку "Налаштування", якщо потрібно налаштувати проксі-сервер для цього підключення. 

Налаштування локальної мережі (ЛОМ)

Параметри мережі не застосовуються до комутованих підключень. Щоб налаштувати комутоване підключення, натисніть кнопку "Налаштування", розташовану вище.





Анотація

Павленко Ю.С. – Реалізація переносимої LMS платформи PocketMoodle на базі UbuntuServer засобами OracleVirtualBox – Рукопис.

Магістерська робота за спеціальністю 122 – Комп’ютерні науки – Волинський національний університет імені Лесі Українки, Луцьк. – 2024 р.

В роботі досліджено питання використання платформи віртуалізації для побудови хостингу для розгортання на ньому LMS Moodle. Розглянуті мережеві можливості та підтримка засобів роботи хостингів. Детально розглянуто процес розгортання та налагодження хостингу на основі серверної операційної системи UbuntuServerLTS кількох різних версій. Реалізовано підтримку VPN для розширення функціоналу кінцевої розробки.

Створено збірку портативної LMS Moodle – «PocketMoodle», яку можна використовувати як в локальній, так і в глобальній мережі, або в ізольованих сегментах глобальної мережі.

Реалізовано декілька екземплярів таких систем та адаптовано їх для роботи у локальній і глобальній мережах. Такі збірки можуть бути використані некваліфікованими користувачами як у навчальному процесі, так і для отримання навиків роботи з LMS Moodle на рівні адміністратора. Система легко копіюється, переноситься та розгортається.

Перенесення PocketMoodle з однієї системи на іншу зводиться до переносу папки віртуальної машини і в подальшому лише встановлення в якості проміжного адаптера наявного у новій системі через графічний інтерфейс налагодження віртуальної машини.

Ключові слова: LMS, LMS Moodle, віртуальна машина, хостинг, сервер, серверна операційна система, VPN.

Abstract

Pavlenko Y.S. – Implementation of a Portable LMS Platform Pocket Moodle Based on Ubuntu Server Using Oracle VirtualBox – Manuscript.

Master's thesis in the specialty 122 – Computer Science – Lesya Ukrainka Volyn National University, Lutsk. – 2024.

The study explores the use of a virtualization platform to create hosting for deploying the LMS Moodle. Network capabilities and support tools for hosting operation are considered. The process of deploying and configuring hosting based on the Ubuntu Server LTS operating system of several different versions is examined in detail. VPN support was implemented to expand the functionality of the final development.

A portable LMS Moodle build, named "PocketMoodle," was created. It can be used in both local and global networks or in isolated segments of a global network. Several instances of such systems were implemented and adapted for operation in local and global networks. These builds are user-friendly for unskilled users and can be utilized both in the educational process and to acquire skills in managing LMS Moodle at the administrator level. The system is easily copied, transferred, and deployed.

Transferring PocketMoodle from one system to another is simplified to moving the virtual machine folder and subsequently setting up the existing adapter in the new system through the graphical interface of the virtual machine settings.

Keywords: LMS, LMS Moodle, virtual machine, hosting, server, server operating system, VPN.