

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БІОРЕСУРСІВ
І ПРИРОДОКОРИСТУВАННЯ УКРАЇНИ
Факультет інформаційних технологій

НУБІП України

УДК 004.8:378.091.2

«ПОГОДЖЕНО»

«ДОПУСКАЄТЬСЯ ДО

НУБІП України

Декан факультету
інформаційних технологій
Глазнова О.І., д.п.н., професор

ЗАХИСТУ»
Завідувач кафедри комп'ютерних наук
Голуб Б.Л., к.т.н., доцент

НУБІП України

2022 р.

2022 р.

МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА

на тему Інтелектуальна система підбору навчальних електронних курсів

Спеціальність 121 Інженерія програмного забезпечення

НУБІП України

Світлова програма Програмне забезпечення інформаційних систем
(код і назва)
(назва)
Орієнтація освітньої програми освітньо-професійна
(освітньо-професійна або освітньо-наукова)

НУБІП України

Гарант освітньої програми «Програмне забезпечення інформаційних систем»

К.т.н., доцент
(науковий ступінь та вчене звання)

(підпис)

Голуб Б.Л.
(ПІБ)

Керівник магістерської кваліфікаційної роботи

К.т.н., доцент
(науковий ступінь та вчене звання)

(підпис)

Голуб Б.Л.
(ПІБ)

НУБІП України

Виконав

(підпис)

Таран А.О.
(ПІБ студента)

НУБІП України

КИЇВ-2022

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БІОРЕСУРСІВ
І ПРИРОДОКОРИСТУВАННЯ УКРАЇНИ
Факультет інформаційних технологій

НУБІП України

ЗАТВЕРДЖУЮ

Завідувач кафедри комп'ютерних наук

К.Т.Н., доцент (науковий ступінь, вчене звання) (підпис) Голуб Б.Л. (ПІБ)

НУБІП України

"01" листопада 2021 року

ЗАВДАННЯ

ДО ВИКОНАННЯ МАГІСТЕРСЬКОЇ КВАЛІФІКАЦІЙНОЇ РОБОТИ СТУДЕНТУ

Тарану Анатолію Олександровичу

(прізвище, ім'я, по батькові)

Спеціальність 121 Інженерія програмного забезпечення

(код і назва)

Освітня програма Програмне забезпечення інформаційних систем

(назва)

Орієнтація освітньої програми освітньо-професійна

(освітньо-професійна або освітньо-наукова)

Тема магістерської кваліфікаційної роботи Інтелектуальна система підбору навчальних електронних курсів

затверджена наказом ректора НУБІП України від "01" листопада 2022 р. №1861 «С»

Термін подання завершеної роботи на кафедру 27 жовтня 2022 р.

(три, місяць, число)

Вихідні дані до магістерської кваліфікаційної роботи:

- дані про популярність типів курсів
- дані про екладність курсів та подання матеріалу

Перелік питань, що підлягають дослідженню:

№ з/п	Питання, що підлягає дослідженню	Строк виконання	Примітка
1.	Аналіз предметної області.	01.04.2022	
2.	Моделювання системи	15.07.2022	
3.	Проектування системи	16.09.2022	
4.	Розробка алгоритмів аналізу даних	05.10.2022	
5.	Дослідження отриманих результатів	21.10.2022	

Дата видачі завдання "01" листопада 2021 р.

Керівник магістерської кваліфікаційної роботи

Голуб Б.Л.

(підпис)

(прізвище та ініціали)

Завдання прийняв до виконання

(підпис)

Таран А.О.
(прізвище та ініціали студента)

НУБІП України

	Зміст
Вступ.....	3
1 Системний аналіз предметної області.....	5
1.1 Загальні характеристики онлайн-курсів.....	5
1.2 Сфера онлайн-курсів у мережі інтернет.....	9
1.3 Технічне завдання дипломного проекту.....	16
2 Моделювання системи.....	17
2.1 Моделювання предметної області.....	17
2.2 Моделювання архітектури.....	22
3 Розробка системи.....	26
3.1 Архітектура системи.....	26
3.2 Вибір засобів для реалізації інформаційного забезпечення системи.....	28
3.3 Вибір засобів для реалізації прикладного програмного забезпечення.....	35
4 Результати дослідження.....	44
4.1 Вимоги до апаратного та програмного забезпечення.....	44
4.2 Опис програмного продукту.....	46
4.3 Тестування програмного продукту.....	49
4.4 Аналіз дослідження.....	51
Висновки.....	56
Список використаних джерел.....	58
ДОДАТОК А.....	Ошибка! Закладка не определена.
ДОДАТОК Б.....	Ошибка! Закладка не определена.

ВСТУП

НУБІП України

Актуальність теми. У наш час самоосвіта та вміння отримувати знання самостійно є однією із складових успішної особистості. Люди все частіше

відходять від традиційних методів отримання знань – чи то університет, чи

НУБІП України

заклад підвищення кваліфікації. У зв'язку з пандемією COVID-19 та військовими діями на території України суспільство починає працювати та освоювати нові для себе навички, користуючись онлайн засобами. Саме сфера ІТ є провідною у

цьому напрямку. Розуміючи це, люди часто шукають сервіси неформальної

НУБІП України

освіти, проте більшість із цих сервісів – або не дають змоги в повній мірі опанувати ту чи іншу галузь, або не надають необхідний функціонал для успішного навчання.

Об'єктом дослідження є управління навчальною системою.

Предметом дослідження є система аналізу даних навчальної платформи,

спрямована на оптимізацію підбору курсів.

НУБІП України

Проте керувати настільки великою системою варто пам'ятати про певне узагальнене управління інформацією, що накопичується в ході її роботи. Аналіз

цих даних та пропозиції щодо курсів на основі цього аналізу даних і є *метою*

роботи.

НУБІП України

Практичне завдання дослідження полягає у:

- проведенні аналізу предметної області;
- дослідженні алгоритми аналізу та технології роботи з даними;
- проектуванні сховища даних та архітектури системи;
- аналізу роботи обраних технологій та значень розробленої системи;
- реалізації елементів системи та застосуванні технологій OLAP.

НУБІП України

Методи дослідження. Для проведення аналізу використовуються OLAP-

технології. Саме вони якнайкраще підходять для даного аналізу, завдяки

НУБІП України

функціям, які вони пропонують. Куби OLAP являють собою фактично мета-звіти. Розрізаючи мета-звіти за вимірюваннями, аналітик отримує двовимірні звіти.

Оскільки аналітики, як правило, не працюють з інформацією, яка доповнюється і змінюється в процесі користування системою, сформований куб є актуальним протягом досить тривалого часу. Завдяки цьому, не тільки виключаються перебої в роботі сервера реляційної СКБД, але і різко підвищується швидкість доступу до даних для самого аналітика, оскільки тепер йому не потрібно постійно звертатись до СКБД для отримання даних, він це може зробити напряму, звертаючись до СД.

Наукова новизна даної роботи полягає у запропонованому удосконаленню архітектури системи, алгоритмів обробки інформації та застосування новітніх інформаційних технологій для подальшого використання аналітиками.

Апробація результатів дослідження. Основні результати розробки були висвітлені у тезах «Інтелектуальна система підбору навчальних електронних» на XIII Міжнародну науково-практичну конференцію молодих учених «Інформаційні технології: економіка, техніка, освіта» (м. Київ, 26-27 жовтня 2022 р.).

Дана робота містить 60 сторінок та має таку *структуру*: зміст, вступ, 4 розділи, висновок, список використаних джерел. Основні розділи описані в такій послідовності:

- системний аналіз предметної області;
- моделювання системи;
- розробка системи;
- результати дослідження.

1 СИСТЕМНИЙ АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Загальні характеристики онлайн-курсів

Право кожного на освіту впродовж життя шляхом формальної, неформальної та інформальної освіти забезпечує Закон України «Про освіту» [1].

Сучасна молодь, студенти, а також усі, хто потребує професійного зросту чи змін у кар'єрі, все частіше користуються можливостями навчання з використанням відкритих освітніх ресурсів та онлайн-курсів.

Інтерес та участь у онлайн-навчанні продовжує зростати: у 2020 році було зафіксовано рекордну кількість студентів. Університети запустили нові онлайн-програми, щоб задовольнити цей підвищений попит.

Від докторантів до тих, хто навчається протягом усього життя, люди все частіше використовують онлайн-інструменти для навчання та отримання нових навичок.

Онлайн-навчання не тільки означає, що студенти можуть отримати доступ до інформації, де б вони не були, але й гарантує, що ресурси завжди доступні. Наприклад, замість того, щоб відвідувати бібліотеку, студенти можуть отримати доступ до книг онлайн. Незалежно від того, чи вони в аеропорту, чи в довгій подорожі протягом, все доступно.

Наразі, Європейська Комісія підтримала проєкт Virtual Mobility Pass щодо визнання навчання з використанням відкритих освітніх ресурсів та масових онлайн-курсів. В інноваційному «Паспорті навчання» «оцінки можуть бути задокументовані таким чином, що він може вважатися кваліфікацією вищої освіти» [2]. Це дозволить не втратити перспективних студентів, адже вони останнім часом все більше віддають перевагу західній освіті. Поєднання традиційного, самостійного та онлайн навчання називають змішаним навчанням.

При змішаному навчанні у навчальному процесі та наукових дослідженнях на опрацювання освітнього матеріалу онлайн припадає від 30% до 79 % часу, визначеного на засвоєння курсу [3].

Говорячи про онлайн-класи – це місце, де учень і викладач ніколи не бачать один одного – окрім випадків, коли вони пишуть та обмінюються фотографіями.

Завдяки функціоналу платформи, викладач конфігурує курси, які допомагають вивчати матеріал онлайн. Студенти зазвичай відвідують онлайн-заняття, тому що це зручніше для них і їх розкладу. Навчання в Інтернеті виявилось таким самим, якщо не більш ефективним, ніж традиційне навчання в класі.

Характеристики онлайн-класу:

- студенти входять до системи, щоб виконувати заняття, коли у них є час – 24 години на добу 7 днів на тиждень;

- студенти виконують завдання як онлайн, так і офлайн (залежно від того, як це налаштував викладач);

- студенти складають тести/іспити в Інтернеті або через контрольований екзаменаційний сайт;

- діяльність орієнтована на студента, а не на викладача;

- групові проекти допомагають колективно вирішити завдання;

- дошки обговорень додають можливості зворотного зв'язку з колегами та викладачами;

- матеріал часто викладається інтенсивно.

- викладач більше не в центрі уваги – часто він виступає в ролі фасилітатора навчання [4].

Онлайн-навчання пропонує широкий спектр переваг:

- комфорт роботи вдома;

- самостійне навчання;

- поліпшення технічних, комунікаційних навичок і навичок критичного

мислення:

- менші витрати;

- формування навичок управління часом.

Очікується, що до 2022 року світовий ринок електронного навчання перевищить 243 мільярди доларів США. У Сполучених Штатах загальна

кількість студентів у коледжах зменшувалася за останнє десятиліття, але онлайн-реєстрація студентів зростає вже 14-й рік поспіль. Через безпрецедентні зриви

пандемії COVID-19 освітні системи в усіх куточках світу похитнулися та були змушені перейти на певний тип дистанційного навчання.

Незалежно від того, чи є бажання вивчити нову мову чи отримати сертифікати для кар'єрного зростання, запис на онлайн-курс може стати

ідеальним рішенням. Онлайн-заняття зазвичай проводяться через навчальні онлайн-платформи.

Як і будь-яка інша онлайн-платформа, онлайн-курси, призначені для навчання і являють собою веб-простір або портал, багатий освітнім контентом і

ресурсами. Зміст цього курсу можна розділити на різні теми або предмети. Зазвичай його завантажує на платформу інструктор курсу або вчитель, залежно

від закладу. Платформа електронного навчання може бути обмежена членством (лише

ті, хто зареєстрований на ній, можуть отримати доступ до неї), але вона також може бути безкоштовною для всіх, хто хоче навчатися.

Переваги платформи онлайн-курсів полягають у тому, що як студенти, так і викладачі можуть відстежувати прогрес студентів у матеріалі курсу, і все

акуратно організовано в одному місці. Крім того, студенти також можуть мати можливість взаємодіяти один з одним на платформі.

Існують різні типи онлайн-платформ для навчання. **LDS** – це сайт, на якому багато різних розробників курсів пропонують

свої навчальні матеріали. Це щось на зразок Amazon, тільки для онлайн-освіти. Там є можливість вибрати теми та викладачів, і записатись на індивідуальний

курс. Програми, які пропонуються через LDS, зазвичай є самоетичними. **Системи управління навчанням (LMS)** - це програмне забезпечення,

яке використовується всередині конкретної організації чи установи. Кожна

школа чи університет може мати різну платформу LMS залежно від типу курсів, які вони пропонують (найпопулярнішим, ймовірно, є Blackboard). До нього можна отримати доступ онлайн, як LMS, або програму, яку потрібно завантажити та встановити на своєму комп'ютері.

Створення курсу в цьому випадку знаходиться в руках професорів або будь-кого іншого, хто викладає в цій організації чи закладі.

Екосистеми управління навчанням (LME) – це розширений навчальний онлайн-інструмент, який охоплює кілька програмних рішень, таких як механізми адаптивного навчання, функції керування навчальним вмістом, програмне забезпечення для розробки курсів, інструменти для оцінювання тощо.

LME зазвичай зарезервовані для великих організацій, шкіл або університетів, які пропонують багато онлайн-дипломних програм і навчальних ресурсів.

Краса віртуального класу полягає в тому, що кожен може навчитися практично всьому! Деякі програми вимагають практичних навичок, таких як навчання, щоб стати медичним працівником (медсестрою, лікарем тощо) або механіком. Однак більшість теоретичних знань можна легко отримати завдяки онлайн-навчанням.

Деякі з найпопулярніших курсів – це курси для IT-індустрії, наприклад курси з інформатики. Цифровий маркетинг також є популярною темою в наші дні, як і всі інші види менеджменту. Кожен може знайти курси для вдосконалення своїх навичок спілкування, лідерства, командної роботи, організації та подібних навичок, які не обов'язково пов'язані з конкретною роботою чи кар'єрою.

Крім того, можна зареєструватися для онлайн-навчання, щоб вибрати нове хобі, як-от цифровий живопис, кераміка, вивчення нової мови, кулінарія тощо.

Не кожна програма пропонує в кінці професійний сертифікат. Якщо користувач хоче використати свої нові знання чи навички в професійному середовищі, то він може записатись на курс, який пропонує сертифікацію після його завершення. Зазвичай це не безкоштовний курс.

Отже, платформи дистанційного навчання – це майбутнє освіти. Навіть після того, як глобальна криза охорони здоров'я закінчиться, ймовірно, буде сплеск студентів, які прагнуть покращити свої знання та кар'єру за допомогою самостійного онлайн-навчання.

Значною перевагою онлайн-навчання є те, що кожен може навчитися всьому, що завгодно! Люди можуть просунути свою академічну кар'єру або вибрати щось зовсім інше. Можуть відшліфувати свої навички спілкування або взяти ті уроки мистецтва чи ораторського мистецтва, які вони завжди хотіли відвідувати [5].

1.2 Сфера онлайн-курсів у мережі інтернет

Thinkific. Thinkific довіряють понад 50 000 розробників курсів з усього світу. Платформа, яка налічує понад 500 співробітників, складається з 48% жінок у її керівній команді. Програмне забезпечення пропонує сертифікацію SSL, безпечні варіанти оплати, відкритий API та інструменти автоматизації. За допомогою цієї платформи користувачі можуть відстежувати прогрес свого студента, проводити заняття в прямому ефірі або на замовлення, продавати та продавати свої курси, інтегрувати платформи та багато іншого.



Рис. 1 Платформа Thinkific

Особливості даної платформи:

- прості у використанні інструменти для побудови курсу з функцією перетягування;

- попередньо створені шаблони та теми;

- інструменти для проведення живих уроків, проведення вікторин, видачі сертифікатів тощо;

- інтеграція з багатьма платформами;

- спеціальний домен з усіма платними планами.

Недоліки:

- інструменти інтеграції досить повільно працюють.

LearnWorlds. LearnWorlds – це платформа для створення курсів. Програмне забезпечення містить інструменти для створення, продажу та маркетингових курсів, створення веб-сайтів, інструменти звітності та аналітики тощо.

LearnWorlds довіряють такі компанії, як Sendinblue, Adidas тощо, завдяки зручності використання, доступній ціні та високій задоволеності клієнтів.

Платформа відповідає GDPR і пропонує кілька дуже корисних інтеграцій, зокрема Google Analytics, Active Campaign, Facebook Pixel, ZenDesk, Fomo, Mailchimp та багато іншого.



Рис. 2 Платформа LearnWorlds

Особливості даної платформи:

• цілодобове обслуговування клієнтів і менеджера з успіху клієнтів;

• мобільні програми для користувачів iOS і Android доступні за додаткову плату;

- інтеграція з API;

• інструменти для створення сайтів, спливаючих вікон, сторінок оформлення замовлення, сертифікатів тощо.

Недоліки:

- деякі дуже корисні функції доступні лише з найбільш

високооплачуваним планом.

• відсутність аналітики для власників курсів в безкоштовній версії.

Teachable. Це універсальна платформа, яка пропонує доступні тарифні плани та інтегрується про ваші податки від вашого імені. Також пропонується безкоштовна версія.

Програмне забезпечення можна розгорнути в хмарі, SaaS, Інтернеті або робочому столі Mac/Windows. Простота використання, яку пропонує програмне забезпечення, є йогою найкращою рисою, за якою слід розумна ціна.



Рис.3 Платформа Teachable

Особливості даної платформи:

- інструменти, які дозволяють створити курс і налаштувати його;

- дозволяє підключити ваш існуючий веб-сайт до Teachable за

допомогою спеціального домену;

- інтеграція з великою кількістю інструментів;
- комплексні інформаційні панелі для отримання інформації про продажі курсів і студентів;

- усі плани дають змогу завантажувати необмежену кількість курсів, пропонують необмежену пропускну здатність відео та дозволяють зараховувати необмежену кількість студентів;

- інструменти для створення привабливих сторінок для продажу та пропозиції студентам знижок, купонів тощо.

Недоліки:

- відсутність мобільного додатку.

Podia. Ця хмарна платформа є потужною та доступною в однечас. Вони пропонують вам цілодобове обслуговування клієнтів та інтеграцію з багатьма платформами, щоб викладачі могли надати своїм студентам найкращий досвід навчання.

Вони також пропонують маркетингові інструменти, включаючи маркетинг електронною поштою, обмін повідомленнями, створення спеціального веб-сайту тощо. Платформа підходить для творців будь-якого віку з усього світу.

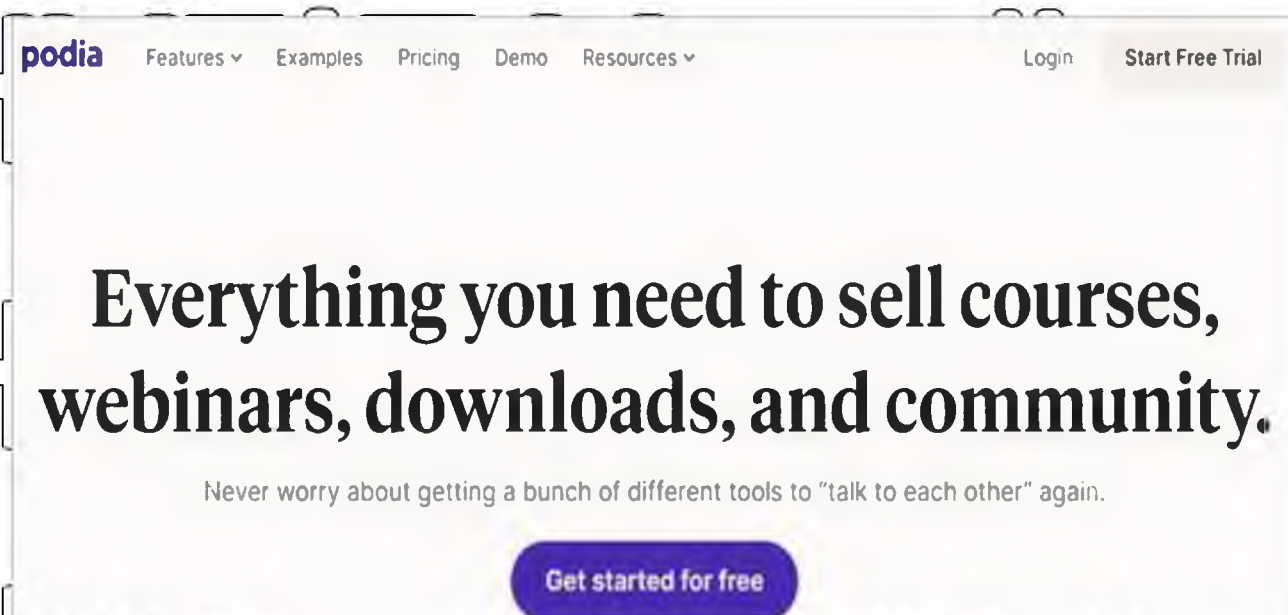


Рис.4 Платформа Podia

Особливості даної платформи:

• кожен план дозволяє створювати необмежену кількість курсів і вебінарів, а також отримувати інструменти електронного маркетингу;

• субдомен Podia, який входить до кожного плану;

• інструменти для налаштування тестів, пропозиції купонів, чату на сторінці тощо;

• інструменти для відстеження прогресу ваших студентів і надсилання їм електронних листів/сповіщень;

• налаштований веб-сайт, зручний для мобільних пристроїв, або

субдомен Podia

Недоліки:

• відсутність мобільного додатку.

• відсутність аналітики для власників курсів.

Kajabi. Kajabi створив понад 300 000 продуктів, допоміг творцям заробити понад 3,5 мільярда доларів США та допоміг створити понад 100 000 компаній.

Вони також пропонують безкоштовні ресурси, включаючи понад 350 навчальних відео, ресурси для завантаження та інструкції експертів, щоб навчити, як зробити свій бізнес успішним.

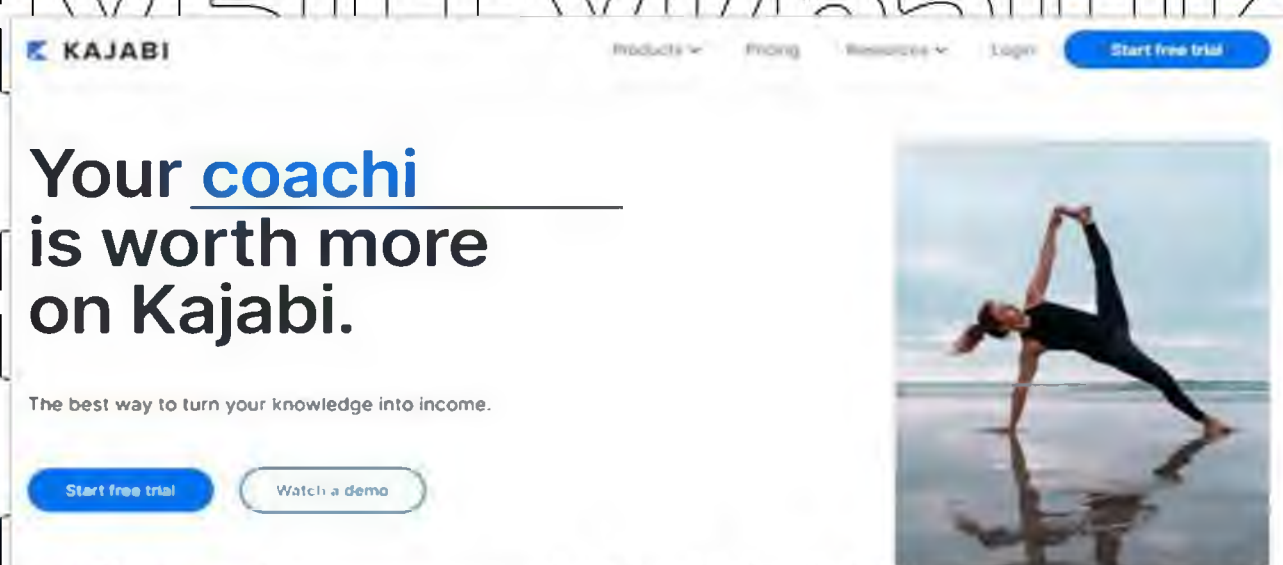


Рис.5 Платформа Kajabi

Особливості даної платформи:

- пропонує готові до використання шаблони та дозволяє створювати необмежену кількість цільових сторінок.

- інтеграція зі Stripe і PayPal для швидкої обробки платежів.

- інструменти автоматизації для налаштування маркетингових кампаній.

- повністю інтегровані інструменти створення веб-сайтів, включаючи готові теми та власні домени.

Недоліки:

- відсутність мобільного додатку.

- відсутність аналітики для власників курсів.

LearnDash. LearnDash – це одна з найкращих платформ онлайн-курсів, яка дозволяє створювати красиві онлайн-курси, отримувати автоматичні сповіщення, створювати сертифікати та бейджі та ділитися ними, інтегрувати їх із улюбленими програмами та створювати докладні звіти про зарахування студентів, прогресі продуктивність і багато іншого.

Це хмарне програмне забезпечення відповідає GDPR, тож користувачі можуть бути впевнені щодо конфіденційності їх особистих даних.



Рис.6 Платформа LearnDash

Особливості даної платформи:

- прості інструменти для створення курсів.
- дозволяє планувати заняття та встановлювати дати початку
- інтеграція з PayPal, Stripe, WooCommerce, 2checkout, Mailchimp, Slack, Zapier та багатьма іншими надзвичайно корисними платформами.

- інструменти звітування для відстеження прогресу, ефективності та реєстрації.
- підтримує зображення, відео, аудіо, файли, тощо

Недоліки:

- відсутність мобільного додатку.
- відсутність аналітики для власників курсів.

Ruzuku. Ruzuku – це глобальне програмне забезпечення для онлайн-курсів, яке просте у використанні та якому довіряють сотні й тисячі авторів, спікерів, тренерів і блогерів.

Платформа стверджує, що допомогла понад 500 000 користувачів пройти онлайн-курси, які пропонуються через Ruzuku. Їх обслуговування клієнтів неймовірне, і програмне забезпечення можна розгорнути в хмарі, SaaS або Інтернеті.



Рис. 7 Платформа Ruzuku

Особливості даної платформи:

- платежі через PayPal або Stripe;
- повна підтримка та спеціальний менеджер з роботи з обліковими записами;

- власний домен;
- опитування, вікторини та необмежені вебінари.

Недоліки:

- відсутність мобільного додатку.
- відсутність аналітики для власників курсів [6].

Технічне завдання дипломного проекту

Постановка завдання. Для аналізу роботи навчальної платформи в системі необхідно забезпечити зберігання таких даних:

- інформація про країни учасників курсів
- інформація про курси

тип курсу

ціна курсу

середня кількість переглядів

- середня оцінка по курсу

- інформація про рівень освіти учасників курсу

Саме внесення вище переліченої інформації в систему, цілісне збереження цих даних в оперативній базі та подальше транспортування даних до СД є завданням.

З метою забезпечення управління процесом продажів курсів і їх складових також необхідно виконати наступні завдання:

- розробити систему для перегляду аналітичних даних;
- інтегрувати систему для перегляду аналітичних даних;
- дослідити вплив отриманої освіти на зацікавленість в курсах;
- дослідити вплив країни користувача на складність виконання завдань курсів та зацікавленість у нових курсах.

2 МОДЕЛЮВАННЯ СИСТЕМИ

Моделювання предметної області

2.1.1. Діаграма прецедентів. Це один зі способів узагальнення деталей

системи та користувачів у цій системі. Дана діаграма визначає події в системі та те, як ці події протікають, однак вона не описує, яким чином дані події реалізовані. Вона складається з набору можливих послідовностей взаємодій між системами та користувачами, що визначає функції, які мають бути реалізовані, і вирішення будь-яких помилок, які можуть виникнути. [7]

Діаграма прецедентів містить чотири компоненти:

- межа системи, в якій діють актори та варіанти використання;
- актори – особи, які беруть участь в системі та визначаються відповідно до їх ролей;
- варіанти використання – описують, як актори використовують систему для досягнення певної мети. Вони зазвичай ініціюються користувачем для виконання цілей, що описують дії та варіанти, які стосуються досягнення мети;
- взаємозв'язки між акторами та варіантами використання.

У табл. 1, табл. 2 наведений опис акторів та прецедентів системи відповідно.

На рис.8 відображена діаграма використання для системи.

Таблиця 1

Актор	Короткий опис
Користувач	Особа, котра зареєстрована в системі та має власний акаунт.
Оператор	Користувач системи, який має права щодо внесення даних.
Аналітик	Аналітик системи, який має права до СД
Топ-менеджер	Адміністратор системи, який має всі права та привілеї

Прецедент	Короткий опис
Перегляд усіх наявних курсів	Прецедент, який позначає можливість перегляду усіх можливих навчальних курсів.
Перегляд куплених курсів	Прецедент, який позначає можливість перегляду курсів, що були оплачені.
Проходження курсу	Прецедент, який позначає прослуховування лекцій, проходження тестів та отримання сертифікату за курс.
Подання заявки на верифікацію	Прецедент, який позначає можливість подання заявки на верифікацію в системі.
Внесення особистих даних	Прецедент, який позначає можливість будь-якої модифікації особистих даних (email, password, інше).
Одобрення нового курсу в системі	Прецедент, який позначає можливість перегляду заявки на створення курсу та схвалення чи відмова, виходячи з політики системи.
Внесення даних щодо існуючих курсів	Прецедент, який позначає можливість модифікації існуючих курсів згідно з політикою системи.
Внесення даних щодо користувачів	Прецедент, який позначає можливість модифікації даних користувачів згідно з політикою системи.
Внесення даних щодо новин	Прецедент, який позначає можливість будь-якої модифікації новин в системі.
Аналіз отриманих даних та перевірка їх достовірності	Прецедент, який позначає можливість аналізу отриманих даних (у тому числі доступ до сховища даних) та перевірка їх достовірності.
Формування на основі отриманих даних гіпотез та тез	Прецедент, який позначає формування на основі отриманих даних певних гіпотез та тез стосовно них.
Розробка стратегій покращення ситуації	Прецедент, який позначає розробку певних стратегій, що будуть спрямовані на покращення ситуації в системі.

Прецедент	Короткий опис
Визначення політики схвалення нових курсів користувачів	Прецедент, що позначає визначення певної політики, за якою буде відбуватися схвалення нових курсів користувачів
Визначення політики схвалення верифікації користувача	Прецедент, що позначає визначення певної політики, за якою буде відбуватися схвалення верифікації користувача й даних щодо себе
Перегляд ефективності підбору курсів	Прецедент, який позначає можливість перегляду ефективності підбору курсів
Перегляд ефективності курсів	Прецедент, який позначає можливість перегляду ефективності курсів (у тому числі аналітики з сховища даних)
Перегляд щоденних звітів	Прецедент, який позначає можливість перегляду щоденних звітів щодо ситуації в системі
Перегляд щомісячних звітів	Прецедент, який позначає можливість перегляду щомісячних звітів щодо ситуації в системі
Перегляд річних звітів	Прецедент, який позначає можливість перегляду річних звітів щодо ситуації в системі

Розглянемо прецедент використання «Перегляд щомісячних звітів».

Назва прецеденту використання: «Перегляд щомісячних звітів».

1. Короткий опис. Прецедент використання дозволяє переглянути щомісячні звіти.

2. Потіки подій.

2.1 Головний потік подій.

Цей прецедент розпочне виконуватися, коли аналітик хоче переглянути звітність.

- 1) система запитує про наявність аналітичних даних
- 2) аналітик обирає часовий період
- 3) система відображає необхідні аналітичні дані у вигляді графіків

2.2 Підлеглі потоки подій.

2.2.1 Вибірка аналітичних даних:

- 1) Система виконує пошук доступних даних та формує їх; якщо система каталогу курсів недоступна, то виконується потік 2.3.1; якщо користувач не має необхідних доступів, то виконується потік 2.3.2;

2.3 Альтернативні потоки

2.3.1 Система, яка керує аналітичними даними недоступна. Видається повідомлення про помилку. Після того, як користувач підтвердить це повідомлення, прецедент завершиться.

2.3.2 Аналітик не має необхідних прав доступу. Видається повідомлення про недоступність даних. Після того, як користувач підтвердить це повідомлення, прецедент завершиться.

3. Спеціальні вимоги. Відсутні.

4. **Передумова.** Перед виконанням прецеденту аналітик має увійти в систему.

5. **Постумова.** Якщо прецедент завершиться успішно, буде зроблено вибірку даних. Інакше стан системи не зміниться.

2.1.2. **Діаграма активності.** Діаграми активності дозволяють моделювати складний життєвий цикл об'єкта, з переходами з одного стану в інший [8].

Але цей вид діаграм може бути використаний і для опису динаміки сукупності об'єктів. Діаграми активності описують перехід від однієї діяльності до іншої, на відміну від діаграм взаємодії, де акцент робиться на переходах потоку управління від об'єкта до об'єкта [9].

На рис.9 відображена діаграма активності для системи.

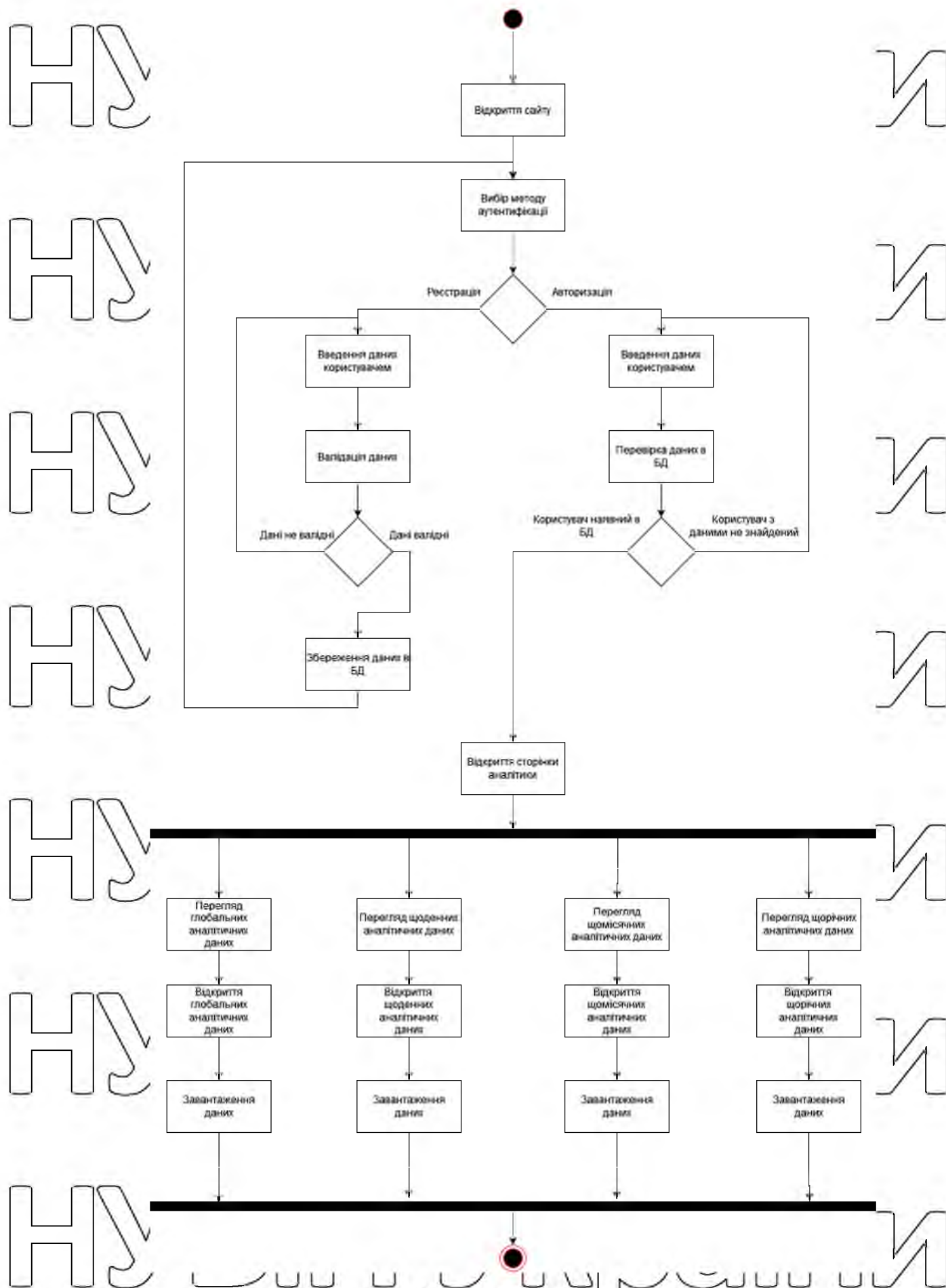


Рис. 9 Діаграма активності

Grafana, який не тільки спідкується з ClickHouse, а й має можливість графічного відображення аналітичних даних.

2.2.2. Діаграма розгортання. Діаграма розгортання – це діаграма, яка показує конфігурацію вузлів обробки часу виконання та компонентів, які на них живуть. Діаграми розгортання – це різновид структурної діаграми, яка використовується для моделювання фізичних аспектів об'єктно-орієнтованої системи. Вони часто використовуються для моделювання статичного вигляду розгортання системи (топології апаратного забезпечення).

Діаграми розгортання важливі для візуалізації, специфікації та документування вбудованих, клієнт/серверних і розподілених систем, а також для керування виконуваними системами за допомогою прямого та зворотного проєктування.

Діаграма розгортання – це лише особливий вид діаграми класів, яка фокусується на вузлах системи. Графічно діаграма розгортання являє собою сукупність вершин і дуг.

Призначення діаграм розгортання:

- вони фіксують апаратне забезпечення, яке використовуватиметься для реалізації системи, і зв'язки між різними елементами апаратного забезпечення;
- вони моделюють фізичні апаратні елементи та шляхи зв'язку між ними;
- їх можна використовувати для планування архітектури системи;
- вони також корисні для документування розгортання програмних компонентів або вузлів БД [11];

Формально можна говорити про такі складові системи:

- робоча станція користувача, на якій є доступ до модуля введення та виведення даних;
- робоча станція аналітика, на якій є доступ до модуля введення та виведення даних та модуля аналітики;
- сервер БД, на якому розгорнута СКБД та реалізована БД;
- сервер СД, на якому реалізована СД система;

• модуль введення та виведення даних, який здійснює свою роботу завдяки фреймворку Vue;

• модуль аналітики, а саме оперативний аналіз OLAP.

• сервер системи, на якому знаходяться БД та СД та який здійснює обробку запитів за допомогою модуля обробки запитів;

• модуль обробки запитів здійснює обробку запитів які надходять використовуючи архітектуру мережевих протоколів REST.

На рис. 11 зображена діаграма розгортання для системи, що описується.

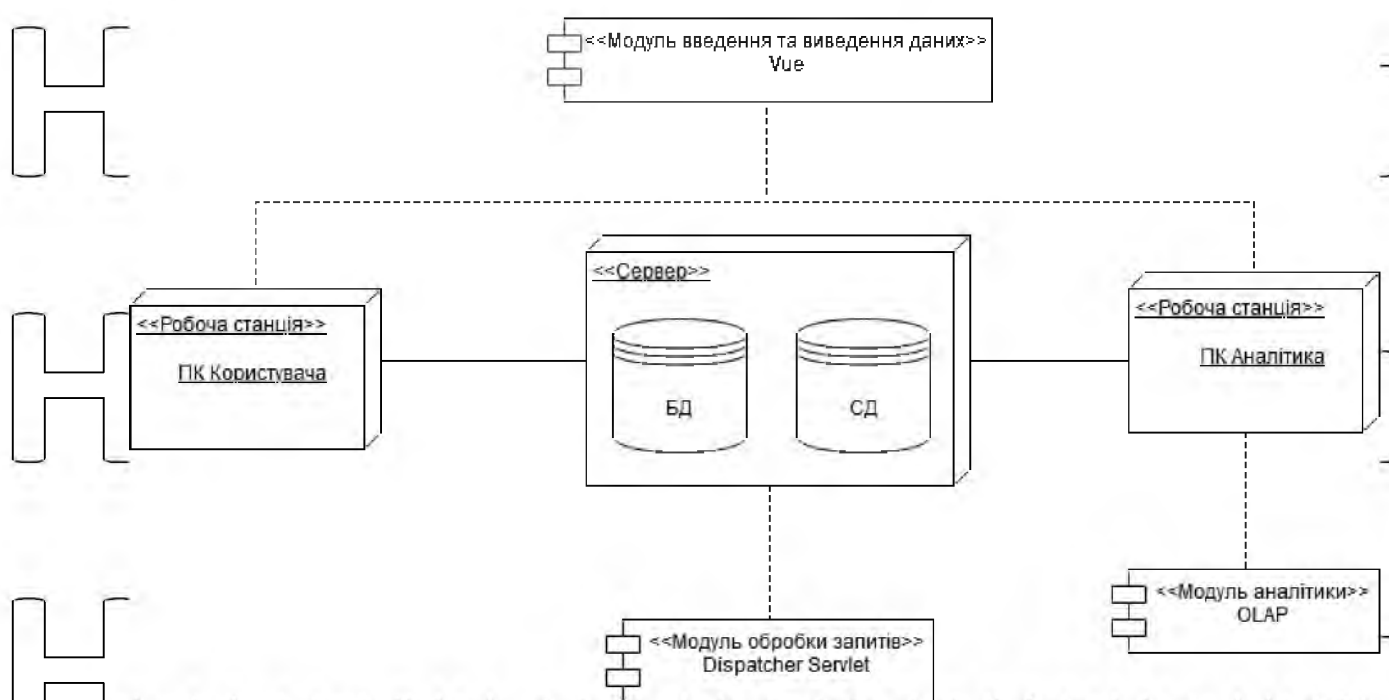


Рис. 11 Діаграма розгортання

3 ПРОЄКТУВАННЯ СИСТЕМИ

3.1 Архітектура системи

Архітектура клієнт-сервер відноситься до архітектур систем, які розміщують, надають та керують більшістю ресурсів і послуг, які запитує клієнт.

У цій моделі всі запити та послуги доставляються через мережу, і її також називають моделлю мережових обчислень або мережею клієнт-сервер.

Архітектура клієнт-сервер, яка також називається моделлю клієнт-сервер, – це мережева архітектура, при якій програма розподіляє завдання та навантаження між клієнтами та серверами, які знаходяться в одній системі або об'єднані комп'ютерною мережею.

Клієнт-серверна архітектура зазвичай складається з декількох робочих станцій користувачів, ПК або інших пристроїв, підключених до центрального сервера через Інтернет або іншу мережу. Клієнт надсилає запит на дані, а сервер приймає та влаштовує запит, надсилаючи пакети даних назад користувачеві, якому вони потрібні [12].

3.1.1. Опис обраної архітектури. Для проєкту було вирішено обрати клієнт-серверну архітектуру проєктування ПЗ. Вона складається з двох типів компонентів: клієнтів та серверів. Серверний компонент постійно слухає запити від клієнтських компонентів. Коли запит отримано, сервер обробляє запит, а потім відправляє відповідь назад клієнту.

3.1.2. Розмежування між серверною частиною та клієнтською. Серверна частина – це програми, які працюють на сервері та зосереджені на генерації динамічного контенту. Вони використовуються для запитів та взаємодії з базою даних, доступу до файлів на сервері, взаємодії з іншими серверами, обробки вводу користувача та структурування веб-додатків. Популярні мови програмування серверної частини: C++, Java (Spring та Java EE) та JSP/Thymeleaf, PHP (Laravel), Python (Django) та Ruby on Rails.

Клієнтська частина – це програми, які працюють на клієнтській машині та зосереджені на користувацькому інтерфейсі та інших процесах, таких як

читання або запис файлів cookie. Вони використовуються для надсилання запитів на сервер, взаємодії з локальним сховищем, взаємодії з тимчасовим сховищем, створення інтерактивних веб-сторінок та інтерфейсу між клієнтом та сервером.

Популярні засоби для створення клієнтської частини включають AJAX (Axios), CSS (SASS/SCSS), HTML, Javascript (Vue/React/Angular).

3.1.3. Переваги клієнт-серверної архітектури.

- цей архітектурний стиль надзвичайно гнучкий і може бути адаптований до користувача та вирішення його проблем;

- цю архітектуру також можна поєднувати з іншими типами архітектури на стороні клієнта або сервера;

- у міру зміни функціональних та нефункціональних вимог, модулі можуть оновлюватися, не змінюючи архітектуру клієнт-сервер або не порушуючи роботу служби;

- оскільки дані, що передаються між клієнтом і сервером, і послуги, які сервер надає, залежать лише від розробників, існує безліч способів використовувати цей стиль архітектури для вирішення проблем, які можуть виникнути в майбутньому.

3.1.4. Недоліки клієнт-серверної архітектури.

- має більший відсоток для відмови через централізований контроль;
- особливо вразлива до DOS атак, оскільки кількість серверів значно менша за кількість клієнтів,

досить дорого розгортати мережу та адмініструвати.

3.1.5. Опис взаємодії архітектури в системі.

Серверна частина використовується для обробки запитів і взаємодії з базою даних, доступу до файлів на сервері, взаємодії з іншими серверами, обробки вводу користувача та структурування веб-додатків.

Клієнтська частина використовується для надсилання запитів на сервер, взаємодії з локальним сховищем та тимчасовим сховищем, створення інтерактивних веб-сторінок й інтерфейсу між клієнтом та сервером.

Клієнт-серверна архітектура реалізована окремими серверами, перший з яких, має в собі задеплований App.vue файл, на другому знаходиться задеплований SpringBootApplication файл з всіма необхідними компонентами та встановлений сервер баз даних на окремому порті.

Будь-який користувач робить запит користуючись пристроями вводу, при цьому клієнтська частина системи – Vue, обробляє, перетворює наданий користувачем запит та, використовуючи REST API, звертається до SpringBootApplication. Вона відповідає за бізнес-логіку, тому, відфільтрувавши наданий запит, здійснює попередні модифікації з даними та, отримавши доступ до бази даних, відкриває з'єднання й отримує (або додає) необхідні дані в базі даних MySQL.

3.2 Вибір засобів для реалізації інформаційного забезпечення системи

3.2.1. Загальні поняття з напрямку OLAP-технології. OLAP - це програмне забезпечення для виконання багатовимірного аналізу на високій швидкості великих обсягів даних зі сховища даних, вітрини даних або іншого уніфікованого централізованого сховища даних.

Більшість бізнес-даних мають кілька вимірів - кілька категорій, на які дані розбиваються для представлення, відстеження чи аналізу. Наприклад, показники продажів можуть мати кілька параметрів, пов'язаних із місцезнаходженням (регіон, країна, штат/провінція, магазин), часом (рік, місяць, тиждень, день), продуктом (одяг, чоловіки/жінки/діти, марка, тип), тощо.

В сховищі даних набори даних зберігаються в таблицях, кожна з яких може організувати дані лише за двома з цих вимірів одночасно. OLAP виягує дані з кількох реляційних наборів даних і реорганізує їх у багатовимірний формат, що забезпечує дуже швидку обробку та дуже глибокий аналіз.

OLAP дозволяє зберігати дані в трьох або більше вимірах замість лише двох, що призводить до ряду переваг. Це робить OLAP потужним інструментом

для виявлення даних і прогнозування, що буде, і тому його використовують як базову технологію для багатьох програм бізнес-аналітики (BI).

OLAP-система надає користувачам можливість проводити складний аналіз даних, що дозволяє краще зрозуміти принципи функціонування компанії і знайти способи поліпшення результатів її діяльності. Аналітичні додатки, побудовані на основі OLAP-технологій, дозволяють вирішувати цілий спектр аналітичних задач, таких як аналіз виробництва, фінансовий аналіз, маркетингові дослідження, аналіз електронного бізнесу, CRM-аналіз, аналіз трудових ресурсів (HRM-аналіз).

OLAP дає можливість створення аналітичних додатків, що охоплюють аналіз усього виробничого та фінансового циклу підприємства, при цьому не виникає проблем з наступністю і співставленістю даних [13].

3.2.2. Структура БД навчальної платформи. Система управління базами даних (СКБД) - це комплекс програмних і мовних засобів, необхідних для створення баз даних, підтримання їх в актуальному стані та організації пошуку в них необхідної інформації.

Можливості СКБД:

- дозволяється створювати БД, це здійснюється за допомогою мови визначення даних DDL;
- дозволяється додавання, оновлення, видалення та читання інформації з БД за допомогою мови маніпулювання даними DML, яку часто називають мовою запитів;
- можна надавати контрольований доступ до БД [14].

СКБД була обрана MySQL, проте для виконання аналізу стала необхідною ClickHouse.

Проектуючи структуру бази даних, основою були сутності з логічної моделі. Таким чином структура таблиці повністю відповідала атрибутам сутностей. Схема бази даних, яка була створена зображена на рис. 12.

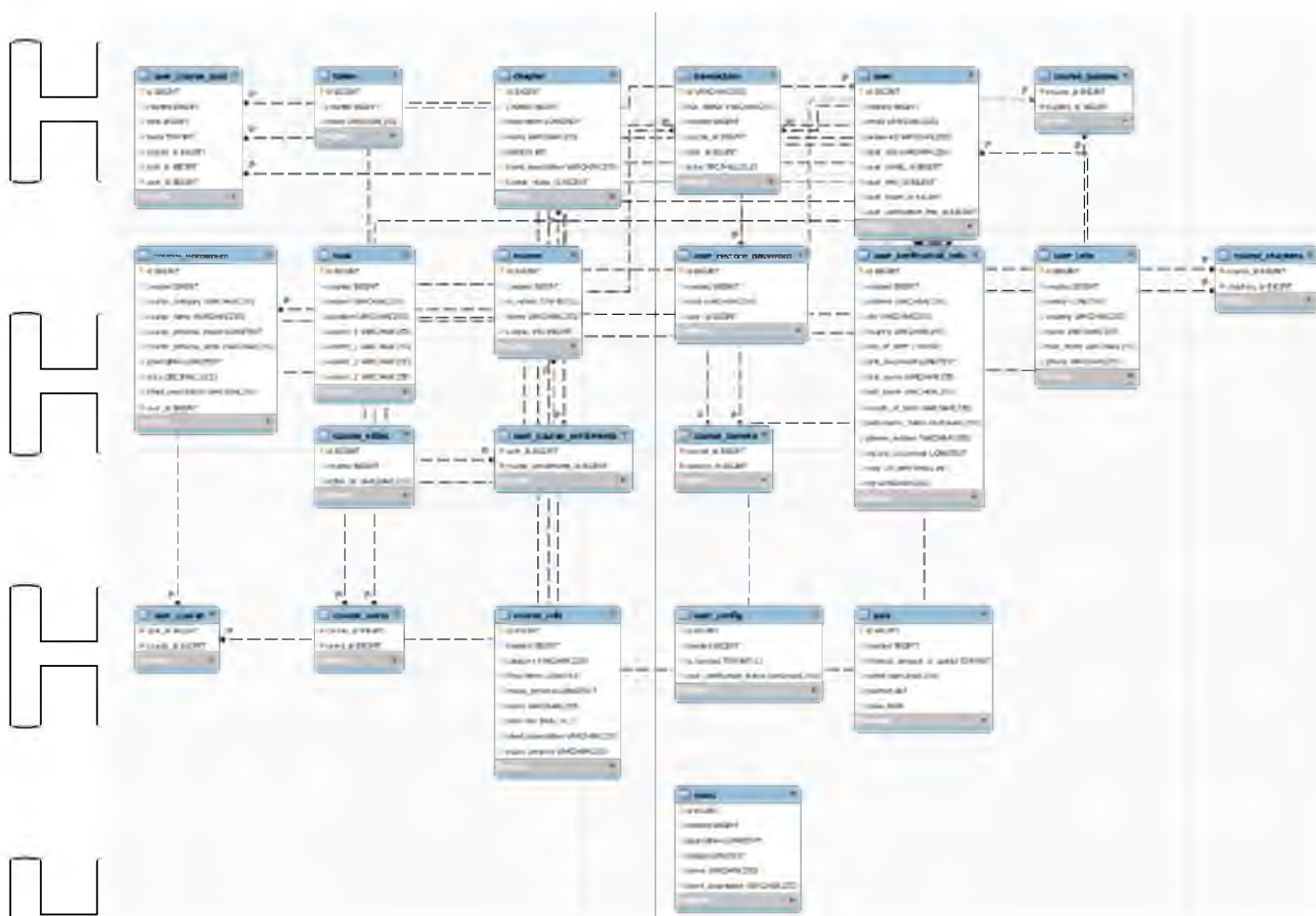


Рис. 12 Діаграма оперативної БД

3.2.3. Вибір СУБД для OLAP. Бази даних онлайн-аналітичної обробки (OLAP) створені спеціально для обробки аналітичних запитів. Аналітичні запити, що виконуються в базах даних онлайн-обробки транзакцій (OLTP), часто потребують багато часу, щоб отримати відповідь. На це є декілька причин.

По-перше, бази даних OLTP зазвичай знаходяться в третій нормальній формі, тому аналітичні запити повинні виконувати складні операції JOIN для багатьох таблиць, що може бути дорогим з точки зору обчислень. По-друге, бази даних OLTP, як правило, мають відносно мало індексів, щоб оптимізувати швидкість запису, тоді як аналітичні запити, які потребують читання, часто виграють від додаткових індексів. По-третє, бази даних OLTP, як правило, постійно зайняті невеликими транзакціями, що може спричинити суперечку (переважно за індексів) під час виконання довгих аналітичних запитів, уповільнюючи як транзакції, так і запити.

Бази даних OLAP вирішують ці проблеми, надаючи окрему оптимізовану базу даних для аналітичних запитів. Вони призначені для прискорення багатовимірного аналізу великих обсягів даних із сховища даних або вітрини даних. Високошвидкісний аналіз можна здійснити шляхом вилучення реляційних даних у багатовимірний формат, який називається кубом OLAP; шляхом завантаження даних для аналізу в пам'ять; шляхом зберігання даних у порядку стовпців; та/або за допомогою паралельного використання багатьох процесорів для виконання аналізу [14].

Порівняльна характеристика OLAP серверів

Таблиця 3

Характеристика	Microsoft Analysis Services	ClickHouse	IBM Cognos TMI	Oracle Database OLAP Option
Ціна	\$899	\$0	\$480	\$460
MOlap	+	+	+	+
ROlap	+	+	-	+
HOLAP	+	+	-	-
Аутентифікація	NTLM, Kerberos	Built-in, LDAP, Microsoft Active Directory, Kerberos	Built-in, LDAP, Microsoft Active Directory, NTLM, IBM Cognos BI authentication	Oracle Database authentication
Шифрування	SSL / SSP	SSL	SSL	SSL

Закінчення таблиці 3

Характеристика	Microsoft Analysis Services	ClickHouse	IBM Cognos TM1	Oracle Database OLAP Option
HOLAP	+	+	-	-
Аутентифікація	NTLM, Kerberos	Built-in, LDAP, Microsoft Active Directory, Kerberos	Built-in, LDAP, Microsoft Active Directory, NTLM, IBM Cognos BI authentication	Oracle Database authentication
Шифрування	SSL / SSP	SSL	SSL	SSL
З відкритим кодом SQL	-	+	-	-
Візуалізація	Microsoft Excel, SharePoint, Microsoft Power BI, інші	Superset, Zeppelin, Tableau, Qlik, Redash, DataLens	IBM Performance Modeler, IBM Cognos Insight, TM1 Web/TM1 Contributor	-

Отже, порівнюючи різні фактори та характеристики, було обрано ClickHouse.

ClickHouse — це система керування базами даних із відкритим вихідним кодом, орієнтована на колонки, яка дозволяє генерувати звіти з аналітичними даними в режимі реального часу [15].

ClickHouse може повертати оброблені результати в реальному часі за частки секунди. Це робить його ідеальним для додатків, що працюють із масивними структурованими наборами даних: аналітика даних, звіти про складні дані, наукові обчислення даних.

ClickHouse обирають за винятково високу продуктивність. Ця ефективність залежить від суми багатьох факторів:

- зберігання даних, орієнтоване на стовпці;
- стиснення даних;
- механізм векторних обчислень;
- приблізні розрахунки;
- використання фізичних розріджених індексів

Проте продуктивність - не єдина перевага ClickHouse. ClickHouse - це більше, ніж база даних, це складна система керування базами даних, яка підтримує розподілену обробку запитів, розділення, реплікацію даних і сегментування. Це масштабована та надійна система, здатна обробляти терабайти даних. На рис.13 зображено графік порівняння швидкості виконання запитів.

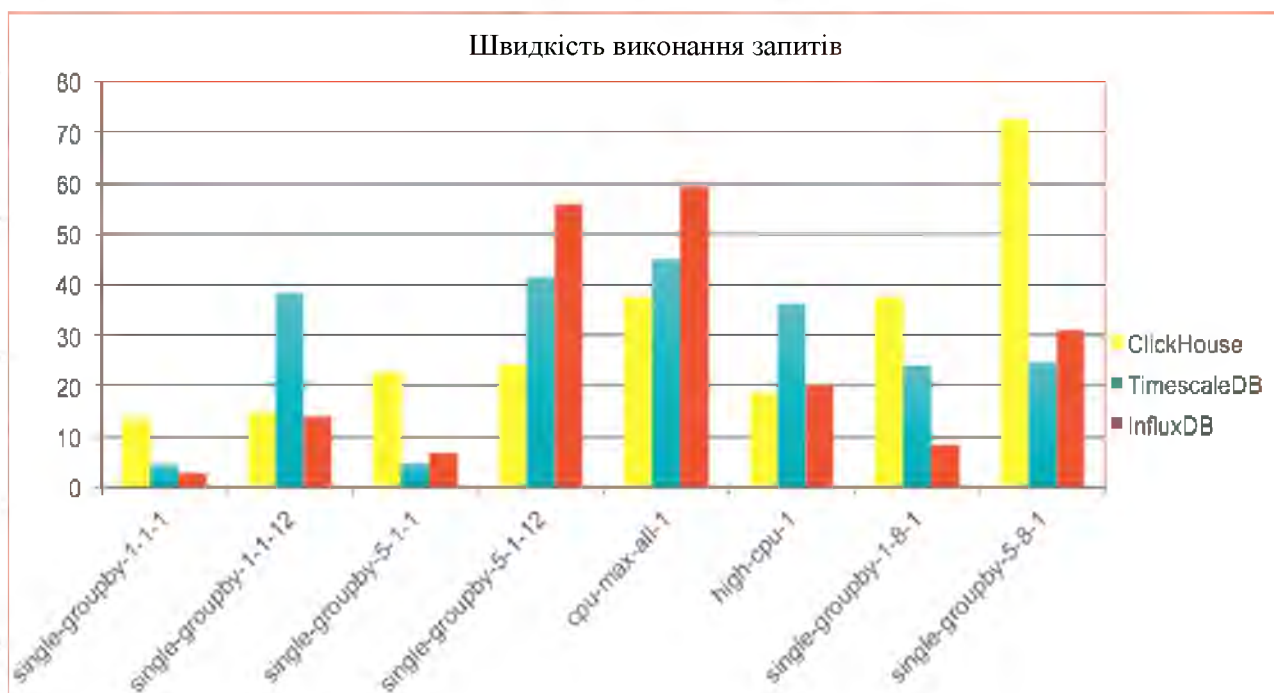


Рис. 13 Графік порівняння швидкості виконання запитів

Насправді ClickHouse призначений для запису величезних обсягів даних і одночасної обробки великої кількості запитів на читання. При цьому в межах ClickHouse досить зручно використовувати декларативну SQL-подібну мову запитів [16].

3.2.4 Структура сховища даних. Сховище даних - це систематизована інформація з різних джерел, яка є необхідною для обробки з метою ухвалення стратегічно важливих рішень [17].

Збереження аналітичних даних для системи аналізу навчальної платформи було забезпечено завдяки СД, фізична модель якого надана на рис. 14.

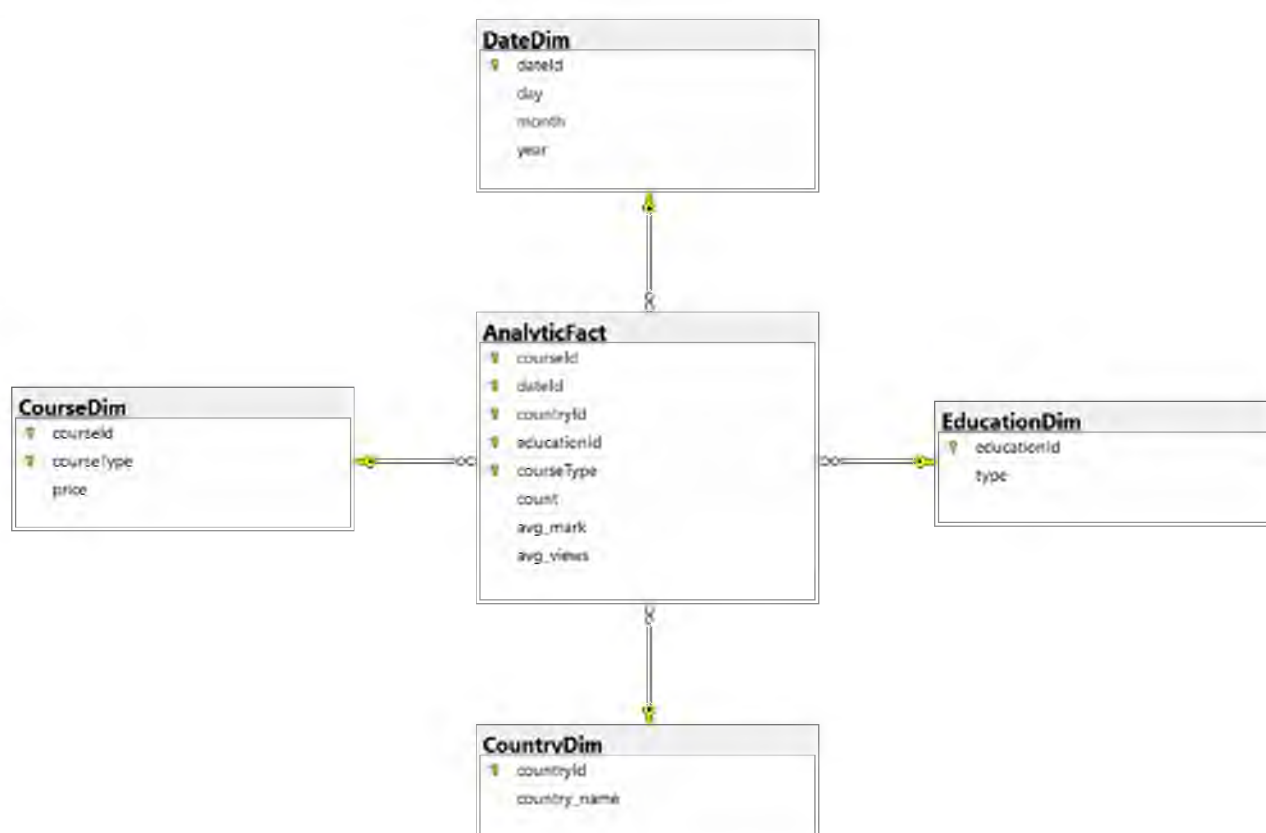


Рис. 14 Фізична модель СД

Вимір (Dimension) - це множина однотипних даних, що утворюють одну з осей куба і характеризують якусь ознаку бізнес-процесу, інформація по якому зберігається у сховищі. Для збереження даних було створено такі таблиці

Вимірів:

- DateDim - часовий вимір, який містить інформацію про рік, місяць та день;

- CourseDim – містить дані про курс, а саме про його ціну та тип курсу;
- EducationDim – містить інформацію про тип здобутої освіти користувачем (середня освіта, вища, тощо);

- CountryDim – містить інформацію про країну, де знаходиться користувач.

Факт – шаблон або подія, які необхідно проаналізувати. У більшості багатовимірних моделей даних факти однозначно визначаються комбінацією значень вимірів; факт існує тільки тоді, коли комірка для конкретної комбінації значень не порожня. Проте деякі моделі трактують факти як «об’єкти першого класу» з особливими властивостями.

AnalyticsFact містить інформацію у розрізі курсу, країни, типу курсу, типу здобутої освіти, дати про:

сумарна кількість придбаних курсів;

середнє значення переглядів;

середнє значення проходження тестів.

Оскільки інформація в таблицях вимірів є відносно постійною, то дані для цих таблиць заповнювались SQL запитами [17].

3.3 Вибір засобів для реалізації прикладного програмного забезпечення

Оскільки у дипломному проєкті розробляється веб-орієнтована система, тому для того, щоб система працювала, необхідно обрати потрібні інструменти для бек-енд та фронт-енд частини додатку та платформу для хостингу.

3.3.1. Вибір засобів для реалізації бек-енд частини програмного забезпечення. Бек-енд, або серверна сторона, складається з сервера, який надає дані на запит, програми, яка їх передає, і бази даних, які організовує інформацію. Основним завданням розробників бек-енду полягає у створенні додатків, які можуть знаходити та передавати дані на інтерфейс, у нашому випадку – фронт-енд [18].

Саме для реалізації бек-енд частини було обрано мову програмування Java разом зі фреймворком Spring.

Spring Framework забезпечує комплексну модель програмування та конфігурації для сучасних корпоративних програм на основі Java на будь-якій платформі розгортання.

Ключовим елементом Spring є інфраструктурна підтримка на рівні додатків: Spring зосереджується на підтримці корпоративних додатків, щоб команди могли зосередитися на бізнес-логіці рівня додатків без непотрібних зв'язків із певними середовищами розгортання [19]. На рис. 15 зображена екосистема даного фреймворку.

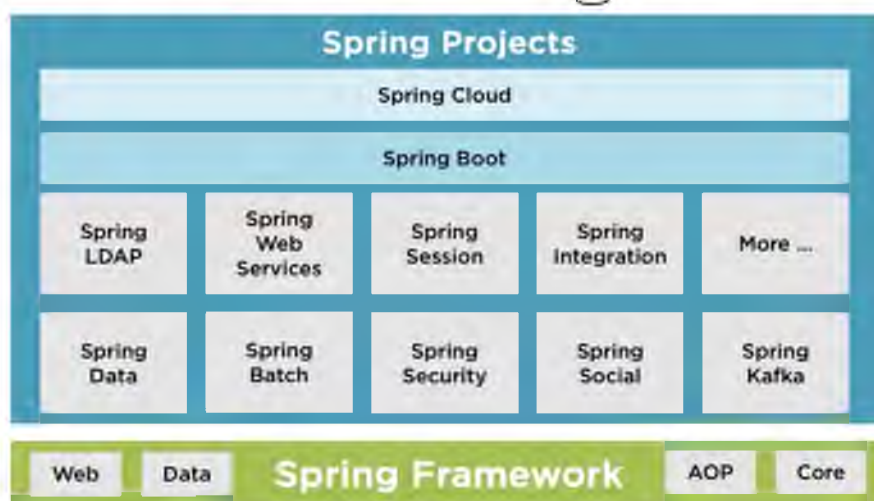


Рис. 15 Екосистема фреймворку Spring

3.3.2. Вибір засобів для реалізації фронтенд частини програмного забезпечення.

Фронт-енд додатки створені з використанням комбінації таких технологій, як мова гіпертекстової розмітки (HTML), JavaScript і каскадні таблиці стилів (CSS).

Фронт-енд розробники проектують і створюють елементи взаємодії з користувачем на веб-сторінці або в програмі, включаючи кнопки, меню, сторінки, посилання, графіку тощо [20].

Для реалізації фронт-енд частини було обрано технології: HTML, SCSS, Vue, Grafana. Вони якнайкраще підходять для реалізації поставлених цілей.

Vue - це фреймворк JavaScript для створення інтерфейсів користувача. Він створений на основі стандартних HTML, CSS і JavaScript і надає декларативну та компонентну модель програмування, яка допомагає ефективно розробляти

інтерфейси користувача, прості чи складні. На рис.16 зображено принцип роботи реактивного зв'язування даних.

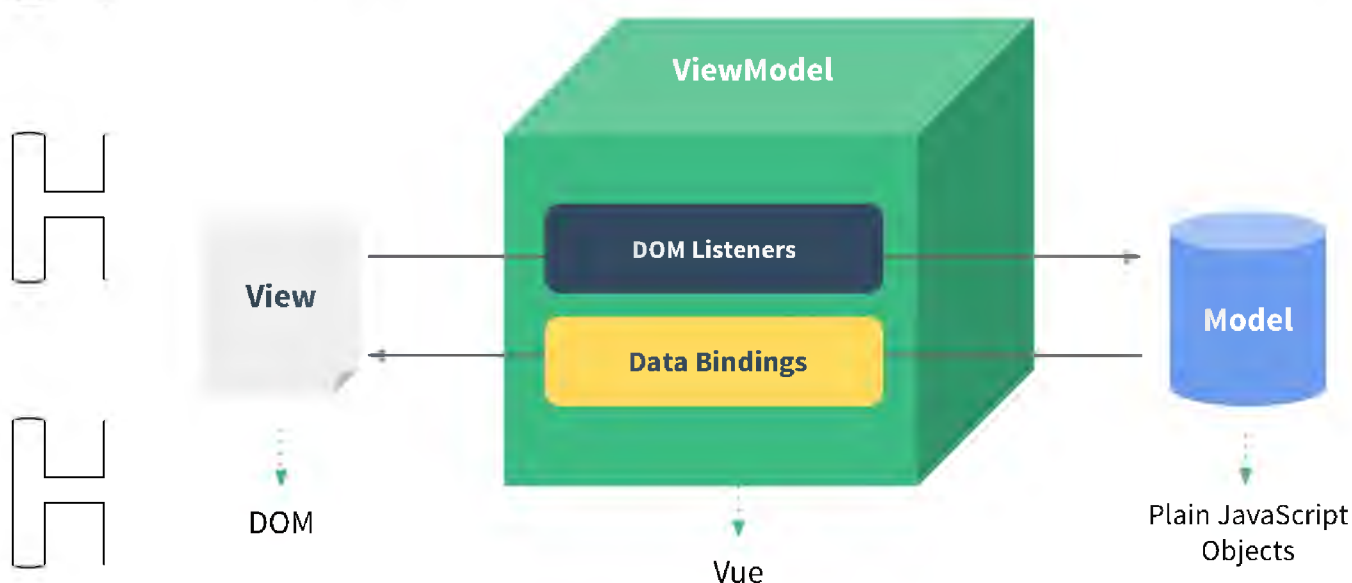


Рис. 16 Реактивне зв'язування даних

Він охоплює більшість загальних функцій, необхідних для розробки інтерфейсу. Але Інтернет надзвичайно різноманітний – речі, які ми створюємо в Інтернеті, можуть кардинально відрізнятись за формою та масштабом. Зважаючи на це, Vue розроблено таким чином, щоб бути гнучким і поступово адаптуватися.

Залежно від вашого випадку використання Vue можна використовувати різними способами:

- покращення статичного HTML без етапу створення;
- вбудовування веб-компонентів на будь-якій сторінці;
- односторінкова програма (SPA);
- fullstack / server-side rendering (SSR);
- jamstack / генерація статичного сайту (SSG);
- орієнтація на комп'ютер, мобільний пристрій, WebGL і навіть

термінал [21].

Серед поставлених завдань є можливість створення графіків та графічних елементів на основі даних саме з сховища даних. Фреймворку Vue для цього завдання буде недостатньо, оскільки, використовуючи його неможливо

повноцінно створити візуальний інтерфейс, який зможе динамічно формувати графічні елементи. Тому для виконання цього завдання було обрано технологію Grafana.

Grafana - це рішення з відкритим вихідним кодом для аналізу даних, отримання показників, які дають сенс величезній кількості даних, і для моніторингу наших додатків за допомогою чудових інформаційних панелей, які можна налаштувати.

Grafana з'єднується з усіма можливими джерелами даних, такими як Graphite, Prometheus, Influx DB, Elasticsearch, MySQL, PostgreSQL ClickHouse тощо.

Оскільки Grafana є рішенням з відкритим вихідним кодом є можливість створювати плагіни з нуля для інтеграції з різними джерелами даних.

Інструмент допомагає вивчати, аналізувати та контролювати дані за певний період часу, який технічно називається аналітикою часових рядів.

Це допомагає відстежувати поведінку користувачів, поведінку додатків, частоту помилок, що виникають у робочому чи попередньому середовищі, типи помилок, що виникають, і контекстні сценарії, надаючи відносні дані [22].

На рис. 17 зображено взаємодію графічного рішення Grafana та бази даних онлайн-аналітичної обробки ClickHouse.

Інформаційні панелі отримують дані з підключених джерел даних. Вони містять набір опцій візуалізації, таких як геокарти, теплові карти, гістограми, усі різноманітні діаграми та графіки, які зазвичай потрібні для вивчення даних.

Інформаційна панель містить кілька різних окремих панелей у сітці. Кожна панель має різні функції [22]. На рис. 18 зображено інформаційну панель аналітики.

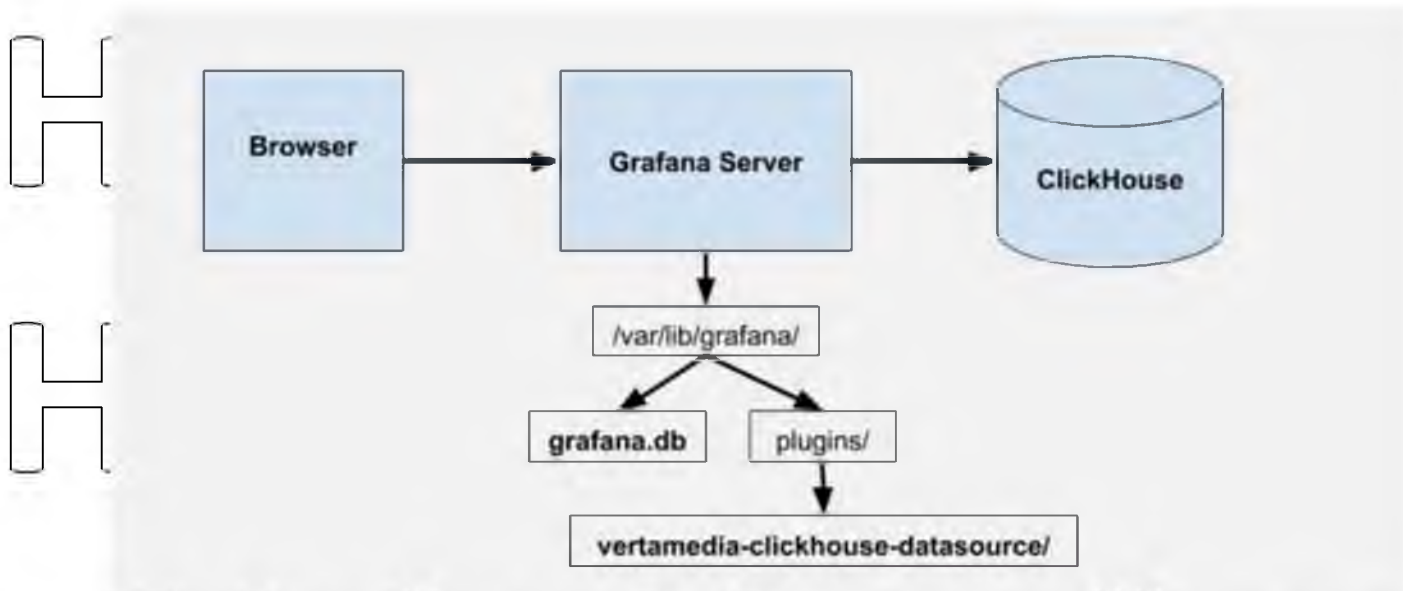


Рис. 17 Взаємодія Grafana та ClickHouse

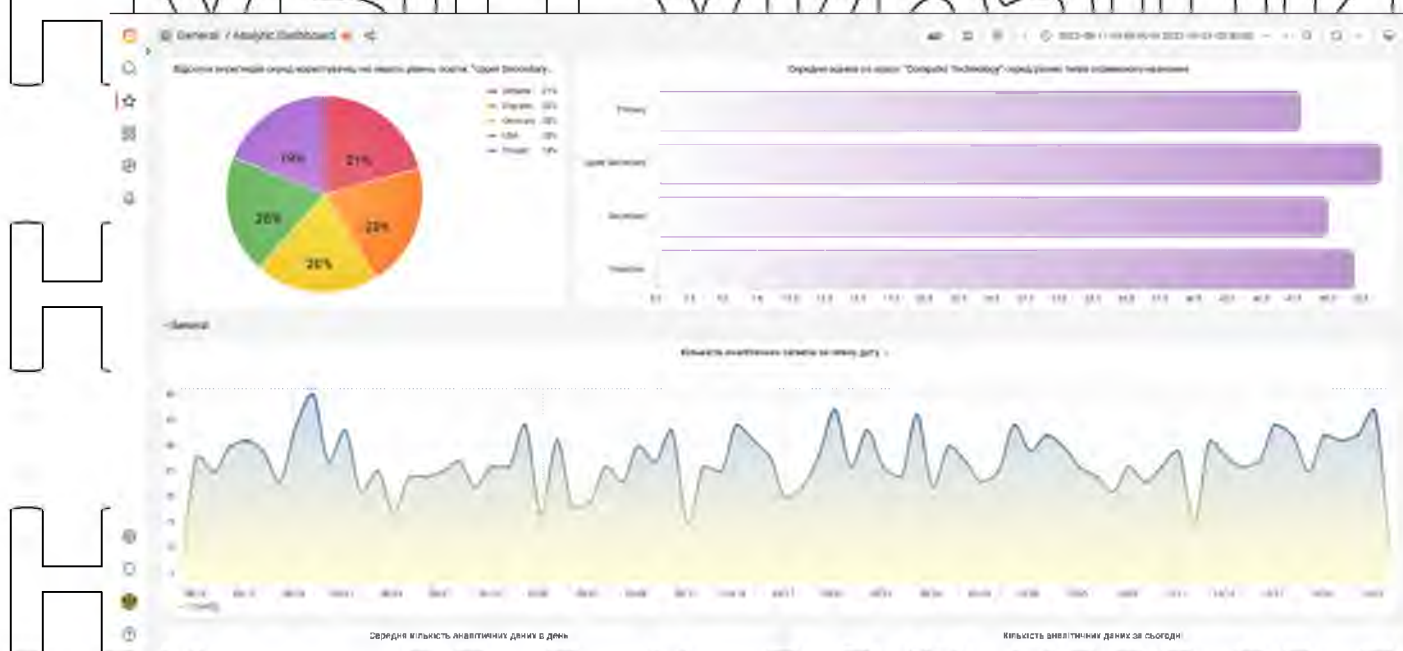


Рис. 18 Інформаційна панель аналітики

3.3.3. Вибір платформи для хостингу. Для хостингу було обрано DigitalOcean.

Це унікальний провайдер хмарного хостингу, який пропонує послуги хмарних обчислень бізнес-структурам, щоб вони могли масштабувати себе.

Програми, розгорнуті на DigitalOcean працюють паралельно на кількох хмарних серверах, що значно збільшує їх продуктивність!

Переваги DigitalOcean серед інших хмарних провайдерів.

Естетичний і простий інтерфейс користувача. Його інтерфейс користувача естетичний, функціональний і позбавлений різних нелогічних сторінок, які ускладнюють роботу нового користувача.

Кількість посилань, і кнопок, є оптимальними для забезпечення доступу до доступних функцій.

Чудова продуктивність. Це один із перших постачальників віртуальних машин на базі SSD, і він використовував IPv6 ще до того, як інші постачальники послуг навіть розглянули це.

Дроплети, які пропонує DigitalOcean, мають блискавичний час завантаження, який становить приблизно 55 секунд.

Доступність. Це те, що відрізняє його від інших компаній хмарних обчислень, які пропонують подібні послуги хостингу. Їх базовий план становить 5 доларів на місяць, а вибір погодинної та щомісячної оплати робить невеликі стартапи та індивідуальні розробки програмного забезпечення дуже доступними для прийняття їхньої платформи.

Незважаючи на зручні для розробників програмного забезпечення механізми, він забезпечує високу продуктивність, що справді стало причиною їхнього успіху.

Активна спільнота. Однією з найкращих речей цієї хмарної платформи є активна цифрова спільнота, яка допомагає, відповідаючи на запити та проводячи обговорення відгуків, тож кожен може отримати від них користь.

Інфраструктура DigitalOcean завжди включала нові передові технології, навіть якщо вони ще не повністю зрілі. Їхні експерти спільноти готові допомогти вам зі своїми пакетами знань, щоб користувачі могли насолоджуватися свободою мобільності практично без будь-яких обмежень [22].

На рис.19 зображено створення проекту EdCourser в межах платформи DigitalOcean.

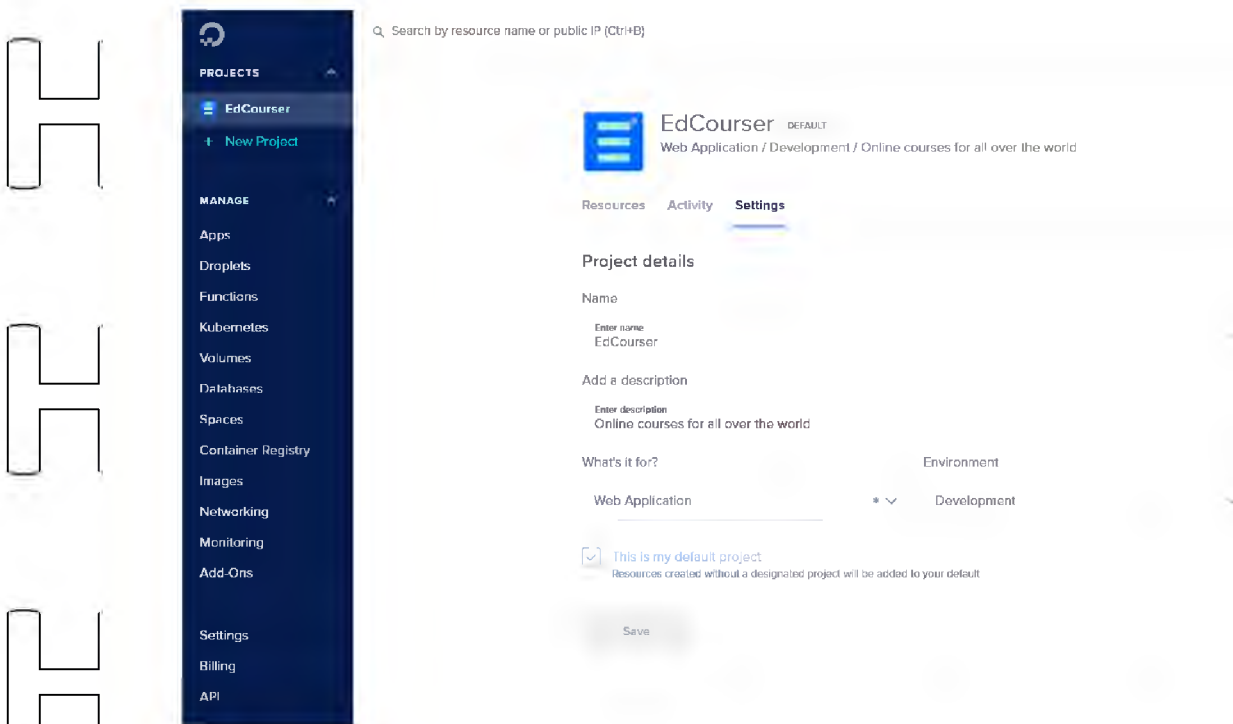


Рис. 19 Платформа DigitalOcean та проєкт EdCoursер

3.3.4. Вибір інструментів для контейнеризації. Контейнеризація - це

форма віртуалізації, коли програми працюють в ізольованих просторах користувача, які називаються контейнерами, використовуючи ту саму спільну операційну систему (ОС). Однією з переваг контейнеризації є те, що контейнер, по суті, є повністю упакованим і портативним обчислювальним середовищем.

Усе, що потрібно програмі для роботи - файли, бібліотеки, конфігураційні файли та залежності - інкапсульовано та ізольовано в її контейнері. Сам контейнер абстрагований від основної ОС, лише з обмеженим доступом до базових ресурсів, подібно до легкої віртуальної машини (VM). Як наслідок, контейнерну програму можна запускати на різних типах інфраструктури, у віртуальних машинах і в хмарі — без необхідності її рефакторити для кожного середовища [23].

Для контейнеризації було використано Docker та Kubernetes для оркестрації контейнерів.

Docker - це відкрита платформа для розробки, доставки та запуску програм. Docker дозволяє відокремити програми від інфраструктури, щоб була можливість швидко доставляти програмне забезпечення. За допомогою Docker можна керувати своєю інфраструктурою так само, як і програмами.

Скориставшись методологіями Docker для швидкої доставки, тестування та розгортання коду, можна значно зменшити затримку між написанням коду та його запуском [24]. На рис.20 зображено створені контейнери створених додатків.

Kubernetes - це потужна система з відкритим кодом, спочатку розроблена Google, для керування контейнерними програмами в кластерному середовищі. Він спрямований на надання кращих способів керування розподіленими компонентами та службами в різноманітній інфраструктурі. Це платформа, розроблена для повного керування життєвим циклом контейнерних програм і служб за допомогою методів, які забезпечують передбачуваність, масштабованість і високу доступність [25]



Рис. 20 Контейнери додатків

Переваги у використанні Kubernetes:

- має здатність до самовідновлення
- залучає основні обчислювальні ресурси та дозволяє розробникам розгортати робочі навантаження на весь кластер, а не лише на окремий сервер;
- адміністратор може самостійно керувати та контролювати декілька одночасно запущених контейнерів,

• забезпечує перенесення робочого навантаження без обмеження типів програм, які він підтримує. Kubernetes також може запускати будь-яку програму, яку може запускати контейнер;

• для швидкого балансування навантаження K8s надає індивідуальні IP-адреси для кожного модуля та єдине ім'я DNS для набору модулів;

• портативна та економічно ефективна платформа (для роботи потрібне менше обчислювальних ресурсів) [26].

НУБІП України

НУБІП України

НУБІП України

НУБІП України

НУБІП України

4 РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

4.1 Вимоги до апаратного та програмного забезпечення

4.1.1. Апаратні вимоги для розгортання системи. Для розгортання системи необхідна будь-яка операційна система. Мінімальні вимоги до системи:

- процесор – Intel Core i3;
- оперативна пам'ять – 4 ГБ;
- жорсткий диск – 64 ГБ.

4.1.2. Програмні вимоги для розгортання системи. Spring Boot вимагає

від системи наявність Java 17 і вище.

Повна підтримка надається для даних інструментів (мається на увазі дані інструменти вже йдуть з «коробки»):

Характеристика інструментів та вимог до версійності

Таблиця 4

Назва	Інструменти	Версія
Gradle	Збірка проєкту	7.4.1
Tomcat	Контейнер для сервлетів	9.0
Spring	Фреймворк	2.6.7

Встановлений та запущений **MySQL** та **ClickHouse** сервер на окремому порті. Встановлений та сконфігурований **nginx** сервер. Встановлене **JS** оточення **Node.js** та менеджер пакунків **npm**.

Для перегляду аналітичних даних має бути запущений **Grafana** сервер та проведена конфігурація Data Source.

4.1.3. Апаратні вимоги для користування системою. Для користування системою необхідна будь-яка операційна система, яка підтримує графічні браузері.

Мінімальні вимоги для успішного користування системою під Windows:

- процесор – Pentium 4 або новіший процесор, який підтримує SSE2;
- оперативна пам'ять – 512МБ ОЗУ / 2ГБ ОЗУ для 64-бітної версії;

- жорсткий диск – 200МБ вільного місця на диску.

Мінімальні вимоги для успішного користування системою під Mac:

- процесор – Macintosh комп'ютер з Intel x86 процесором;
- оперативна пам'ять – 512МБ ОЗУ;
- жорсткий диск – 200МБ вільного місця на диску.

4.1.4. Програмні вимоги для користування системою. Наявність одного з браузерів:

- Safari 5.1-14.1 (для Mac систем);
- Chrome 13-90.0 (для Windows/Mac/Linux);
- Firefox 6.0-88 (для Windows/Mac/Linux);
- Opera 11.50-75.0 (для Windows/Mac/Linux);
- Microsoft Edge 90.0 (для Windows/Mac);
- Pale Moon 28.17.0 (для Windows/Linux);
- SeaMonkey (для Windows/Mac/Linux);
- Vivaldi 3.0-3.8 (для Windows/Mac/Linux).

Для запуску браузерів на Windows/Mac платформах не потрібно додаткових пакетів, проте для запуску під Linux платформи, потрібно:

- GTK+ 3.4-4.0 (інструменти для створення графічних інтерфейсів);
- Glib 2.22-2.68 (низькорівнева бібліотека);
- Pango 1.48 (бібліотека для відображення тексту на різних мовах);
- Libstdc++ (стандартна бібліотека C++ для компіляції)
- X.Org 1.0-1.7 (графічна підсистема Windows).

Опис програмного продукту

Відкривши сайт системи та не авторизувавшись в ній, на рис.21 бачимо сторінку входу до облікового запису



Рис. 21 Сторінка входу до облікового запису

За відсутністю даних для входу, будь-який користувач може створити власний аккаунт, користуючись сторінкою для створення аккаунту, яка зображена на рис.22.

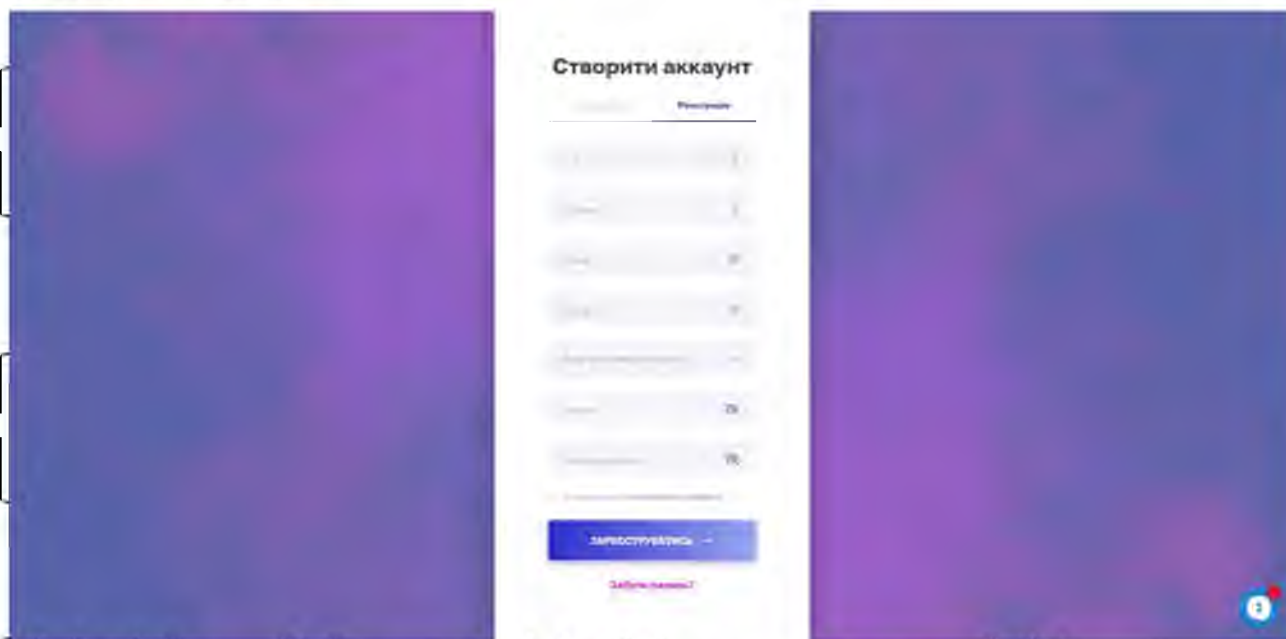


Рис. 22 Сторінка створення аккаунту

Якщо користувач системи не пам'ятає пароль або цей пароль було втрачено, є можливість відновлення паролю для цього облікового запису. На сторінці відновлення паролю, яка зображена на рис.23, це можливо зробити.

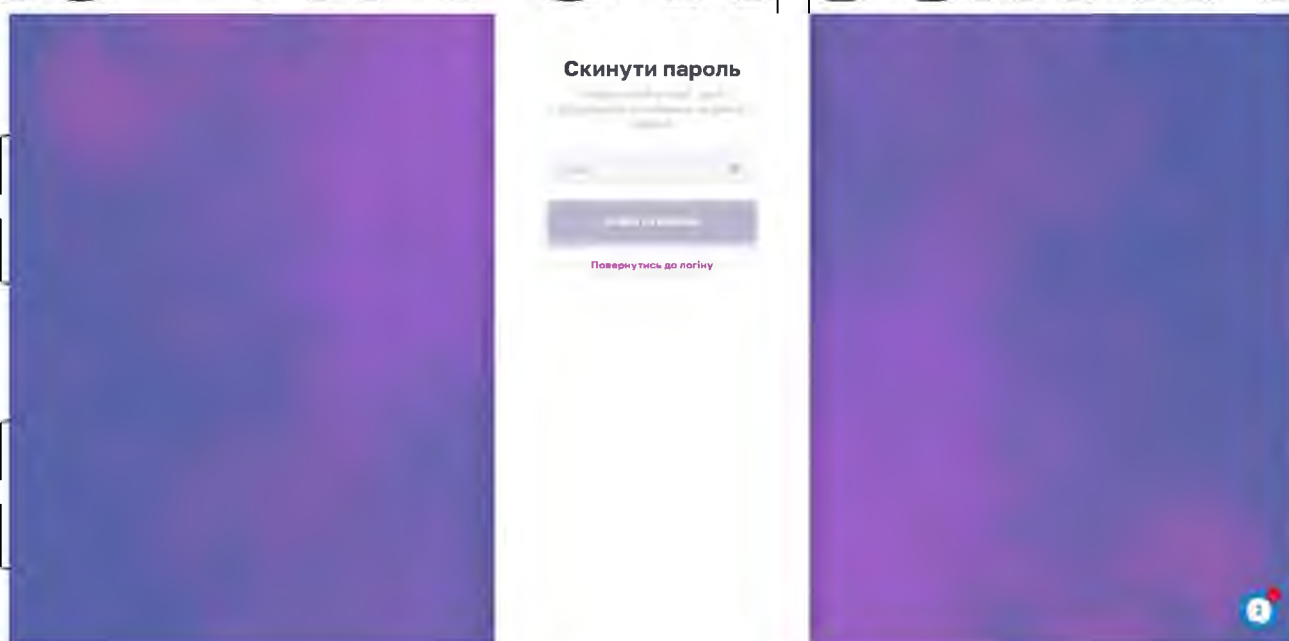


Рис. 23 Сторінка відновлення паролю

Після подання заявки на відновлення паролю, користувачеві прийде лист на пошту з подальшими кроками. Перейшовши за посиланням, користувач потрапить на сторінку зміни паролю, яка зображена на рис.24.



Рис. 24 Сторінка зміни паролю

Успішно авторизувавшись в системі, користувач потрапляє на головну сторінку сайту, яка зображена на рис.25, де розміщені нові курси, новини,

куплені активні курси. Назвне меню зліва та можливий доступ до зміни налаштувань у верхній частині сайту.

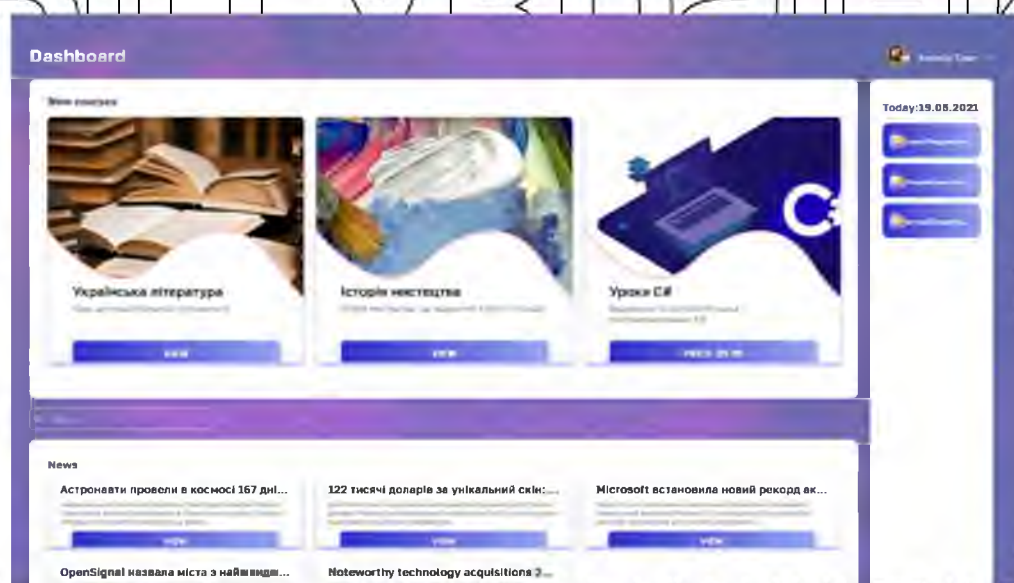


Рис. 25 Головна сторінка сайту

Натиснувши на курси, отримуємо доступ до всіх можливих курсів, зображених на рис.26. На даній сторінці наявна можливість фільтрації, сортування, розбиття на сторінки, зміна вигляду самих курсів та можлива зміна кількості елементів на одній сторінці.

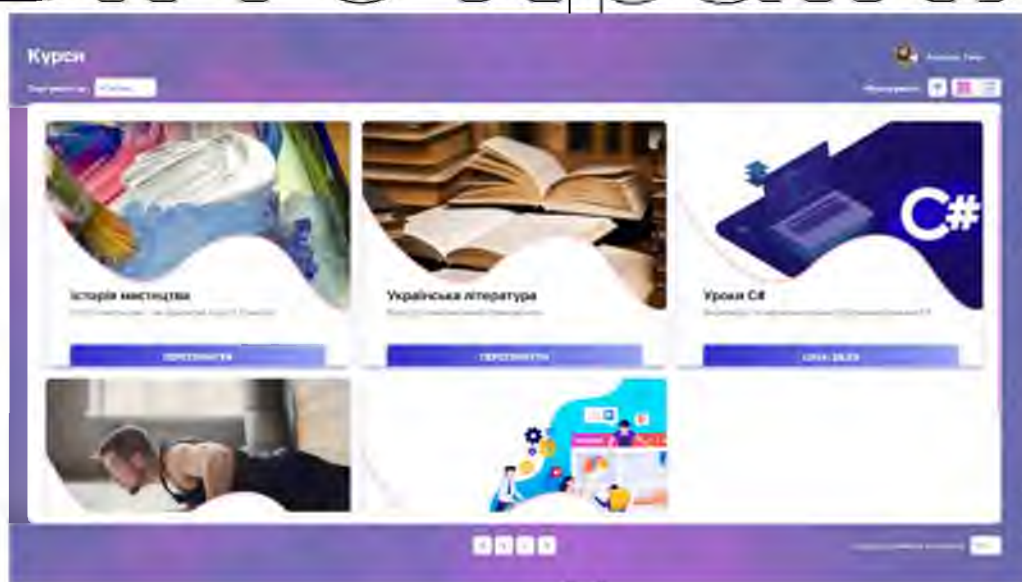


Рис. 26 Сторінка курсів

Перейшовши на сторінку куплених курсів, користувач отримує інформацію про куплені курси, лекції та завдання, які наявні. На рис.27 зображено сторінку куплених курсів.



Рис. 27 Сторінка куплених курсів

Якщо користувач верифікований в системі, то йому відкривається доступ до власних створених курсів на рис.28, де він може переглянути їх.

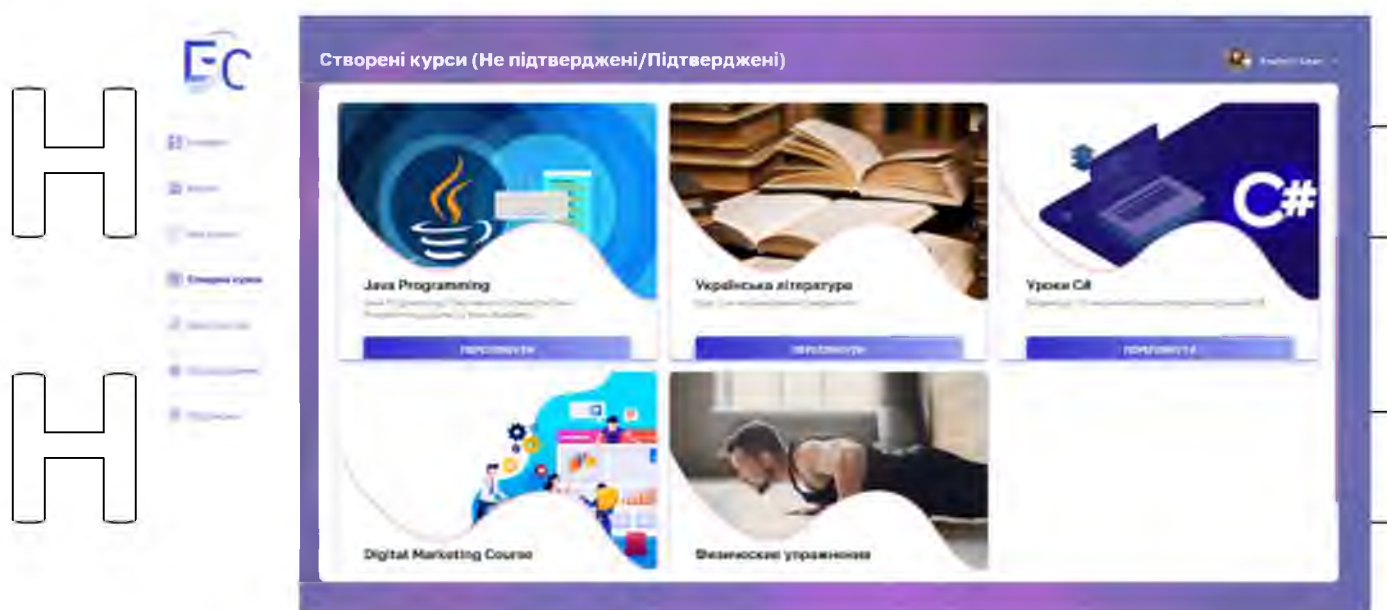


Рис. 28 Сторінка створених курсів

Тестування програмного продукту

4.3.1. Модульне тестування. Модульне тестування визначається як тип

тестування програмного забезпечення, який перевіряє окремі підпрограми, класи або процедури в програмі. Замість того, щоб тестувати всю програму одночасно, модульне тестування рекомендує тестувати менші оудівельні блоки програми.

Модульне тестування здебільшого орієнтоване на «білий ящик». Мета тестування модуля полягає не в тому, щоб продемонструвати належне

функціонування модуля, а в тому, щоб продемонструвати наявність помилки в модулі.

Тестування на рівні модуля дозволяє реалізувати паралелізм у процесі тестування, надаючи можливість тестувати декілька модулів одночасно [27].

На рис.29 зображено приклад одного з модульних тестів в системі.

```

class ValidatorTest {
    @Test
    void isValidEmail_test() {
        assertTrue(Validator.isValidEmail("ataran@gmail.com"));
    }
}

```

Рис. 29 Приклад модульних тестів в системі

4.3.2. Інтеграційне тестування. Це тип тестування програмного забезпечення, у якому різні одиниці, модулі чи компоненти програмного додатку перевіряються як об'єднана сутність. Однак ці модулі можуть бути закодовані різними програмістами.

Метою інтеграційного тестування є тестування інтерфейсів між модулями та виявлення будь-яких дефектів, які можуть виникнути, коли ці компоненти інтегровані та повинні взаємодіяти один з одним.

Як правило, звичайний програмний проект складається з безлічі програмних модулів, багато з яких створені різними програмістами. Інтеграційне тестування показує команді, наскільки добре ці різні елементи працюють разом. Зрештою, кожна одиниця може ідеально функціонувати сама по собі, але гостро постає питання: «Чи можна їх об'єднати і працювати безперебійно?».

Типи інтеграційних тестів:

- метод великого вибуку;
- метод «знизу вгору»;
- гібридний метод тестування;

НУБІП України

На рис. 30 зображено приклад одного з інтеграційних тестів в системі.

```

- Anatoliy
class ChapterServiceImplementationTest extends BaseIntegrationTest {

    1 usage
    @Autowired
    private ChapterService chapterService;

    - Anatoliy
    @Test
    void test() {
        assertThrows(ChapterNotFoundException.class, () -> chapterService.findById(1L));
    }

```

Рис. 30 Приклад інтеграційних тестів в системі

Аналіз дослідження

Аналіз – метод, що полягає в уявному або практичному розчленуванні цілого на складові частини, кожна з яких аналізується окремо у межах єдиного цілого.

В ході виконання дипломної роботи було створено графічні об'єкти, використовуючи наскричені дані про курси. Доступ до певних графіків та даних має звичайний користувач, в даному випадку, власник курсу. Щодо глобальних даних з всієї системи, то доступ до такої інформації можливий тільки з точки зору аналітика та власника системи.

Розглянемо рис.31. На ньому зображена сторінка статистики, доступ до якої має лише власник курсу. В ній наявна інформація про кількість завершених завдань, кількість куплених курсів в розрізі часу та популярність курсу. Серед інших даних – наявна інформація про користувачів та країни користувачів.

НУБІП України



Рис. 31 Сторінка статистики

На рис. 32. зображено стовпчикову діаграму з інформацією про кількість куплених курсів категорії “Комп’ютерні технології” в розрізі країн.

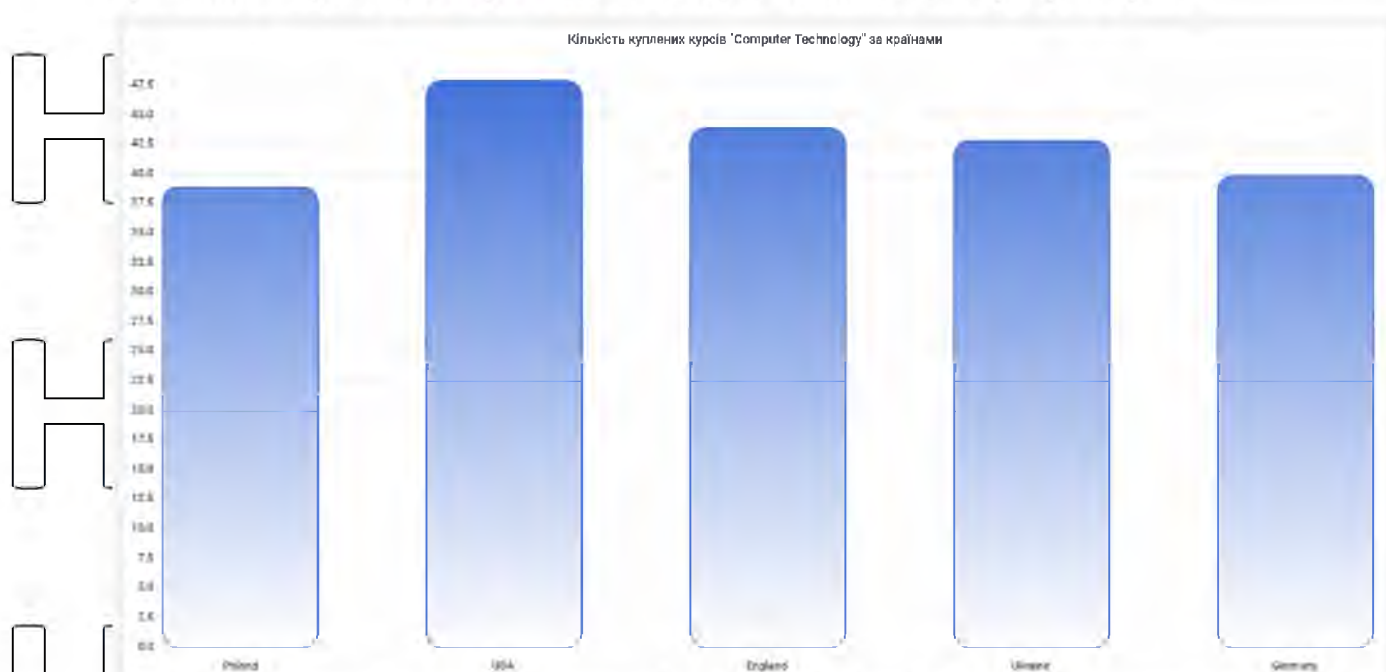


Рис. 32 Кількість куплених курсів “Комп’ютерні технології” в розрізі країн

Висновки

Як бачимо, найбільша кількість – США, найменша – Польща. Аналізуючи ці показники, можна сформуванати комплексне уявлення про зацікавленість користувачів та створюючи нові курси, орієнтуватись на ці показники.

На рис.33 зображено графік середньої оцінки користувачів категорії курсу “Комп’ютерні технології” в розрізі різних рівнів отриманого навчання.

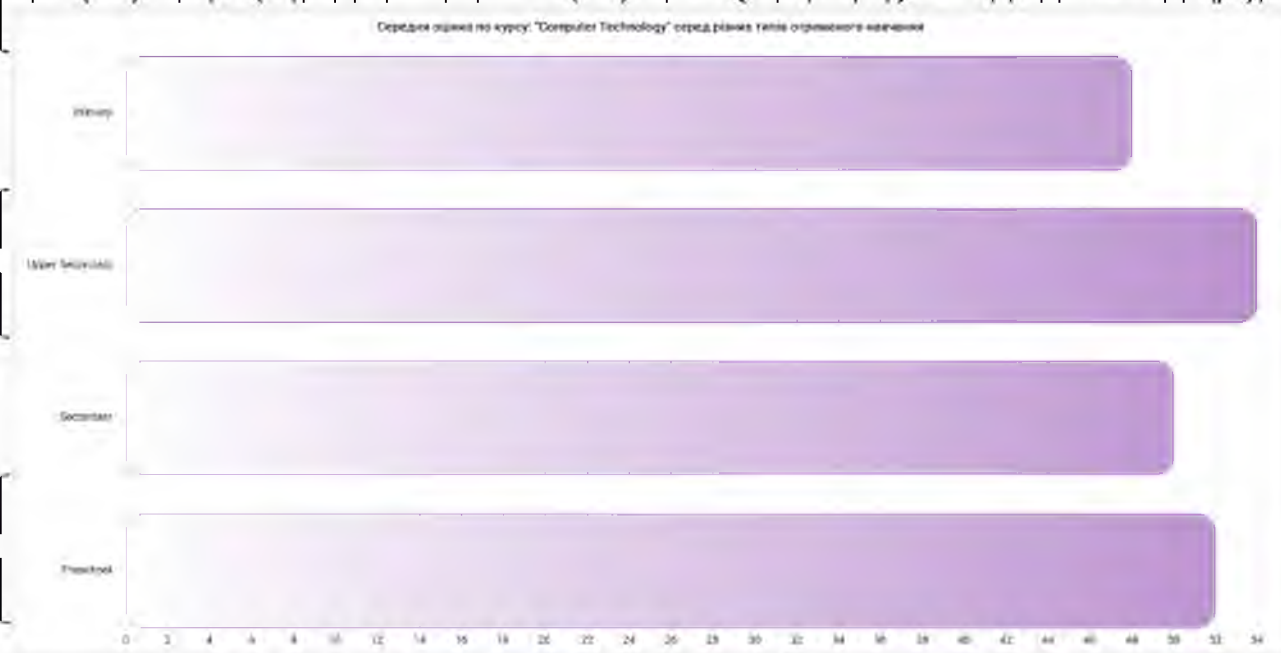


Рис. 33 Середня оцінка по курсу “Комп’ютерні технології” серед різних типів отриманого навчання

Висновки

Аналізуючи ці дані, можна зробити висновок про перш за все складність курсів та подання матеріалу в розрізі різних рівнів отриманого навчання. Як бачимо, найбільшу оцінку мають користувачі з рівнем “Вища середня”, та найменшу – “Початкова”. Отже, варто звернути увагу на курси даної категорії та створити спеціальні матеріали саме для користувачів з рівнем “Початкова освіта”.

Рис.34 зображає графік типу time series з інформацією про кількість куплених курсів категорії “Мистецтво” в розрізі часу.

Висновки

В результаті, аналітик, проаналізувавши дані зможе зробити чіткі висновки щодо популярності категорії курсу та, використовуючи механізми спрогнозувати майбутню ситуацію.

Стосовно саме цього графіка, то бачимо, що зацікавленість користувачів не зменшується з плином часу, та навіть на деяких ділянках більше за середню

Таким чином, можна зробити висновок, що варто просувати цю категорію курсів в подальшому.

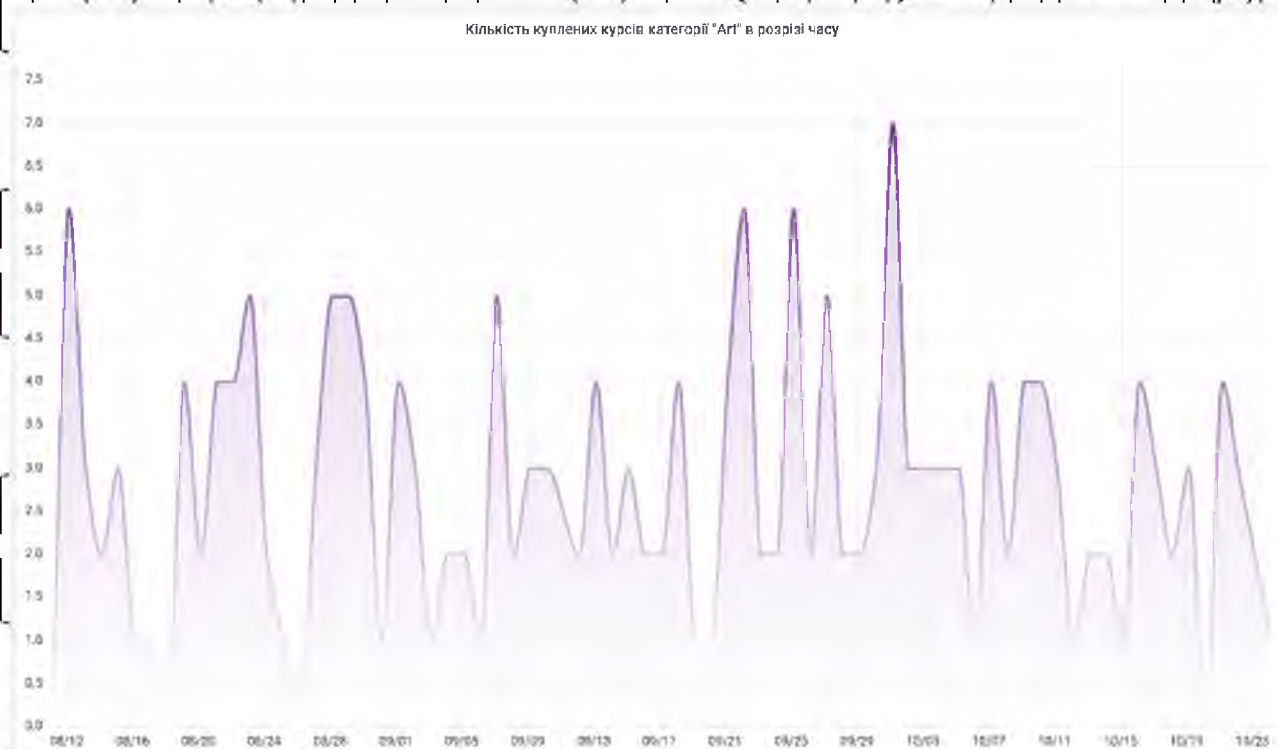


Рис. 34 Кількість куплених курсів категорії "Мистецтво" в розрізі часу

На рис.32 зображено кругову діаграма, яка відображає відсоток переглядів серед користувачів, що мають рівень освіти "Вища середня" в розрізі різних країн.

Висновки

Дані досить схожі, проте опираючись на них, можна зробити висновок про популярність курсів в межах різних країн. В даному випадку, користувачі з різних країн рівномірно зацікавлені до курсів. Відсоток користувачів з України найбільший, з Польщі – найменший, тому варто прийняти рішення щодо створення нових курсів саме на польській мові.

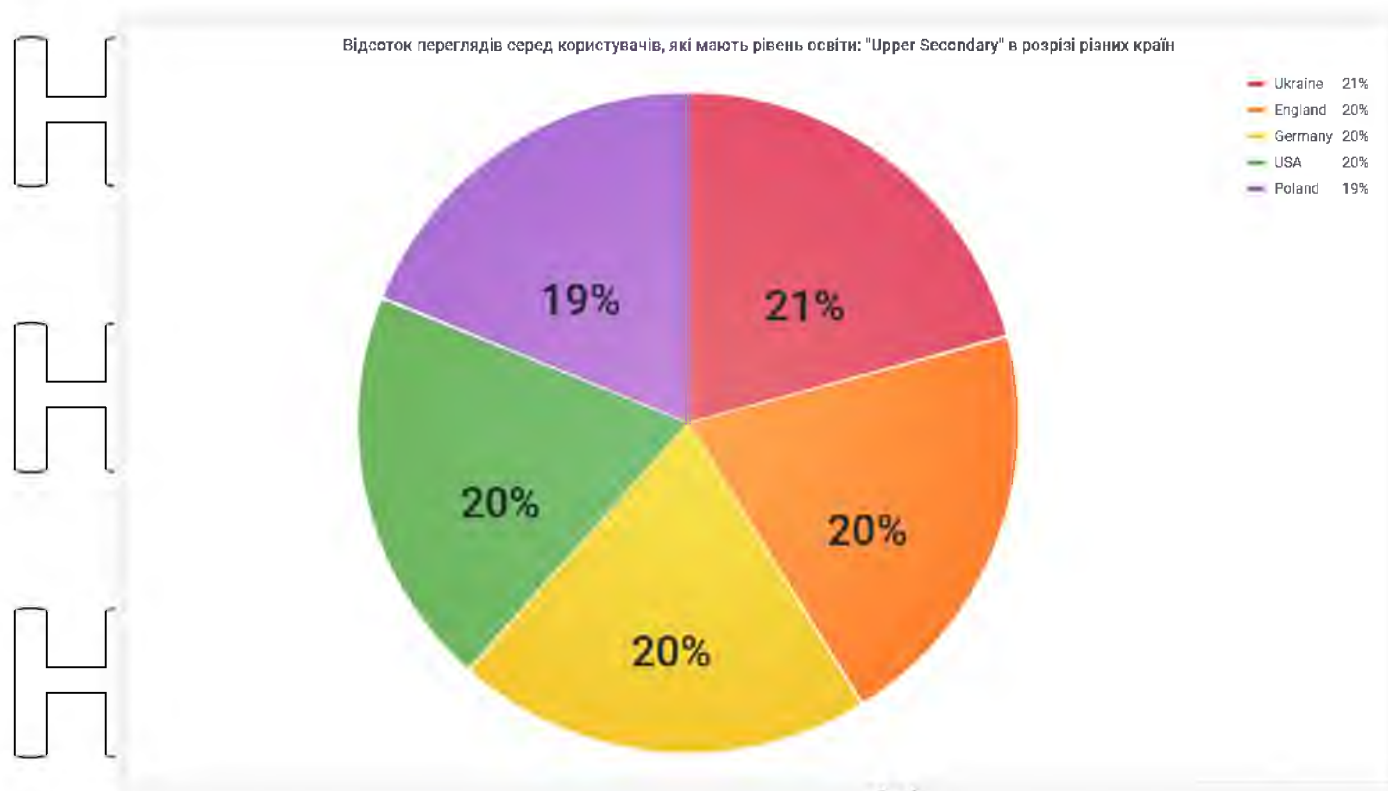


Рис. 32 Відсоток переглядів серед користувачів, які мають рівень освіти "Вища середня" в розрізі різних країн

НУБІП України

НУБІП України

НУБІП України

НУБІП України

ВИСНОВКИ

У ході роботи над дипломним проектом була розроблена інтелектуальна система для аналізу електронних навчальних курсів, впроваджена до існуючої системи онлайн-курсів та досліджено ключові показники для системи.

Практична значущість дослідження полягає в запропонованому удосконаленню існуючої архітектури системи та алгоритмів обробки інформації.

Під час виконання завдання було проведено системний аналіз предметної області розроблюваної системи, досліджено та змодельовано предметну область та архітектуру системи з використанням UML діаграм. Було розроблено архітектуру системи, обрано засоби для реалізації інформаційного та прикладного програмного забезпечення. Сформовано вимоги до апаратного та програмного забезпечення, описано програмний продукт та проведено тестування.

В результаті аналізу даних, було сформовано графіки та досліджено ключові показники в системі.

Найбільше було куплено курсів категорії “Комп’ютерні технології” саме серед користувачів США, найменше – Польща. Аналізуючи ці показники, можна сформуванати комплексне уявлення про зацікавленість користувачів та створюючи нові курси, орієнтуватись на ці показники.

Найбільшу оцінку в розрізі курсів категорії “Комп’ютерні технології” мають користувачі з рівнем “Вища середня”, та найменшу – “Початкова”. Отже, варто звернути увагу на курси даної категорії та створити спеціальні матеріали саме для користувачів з рівнем “Початкова”.

Зацікавленість користувачів щодо курсів категорії “Мистецтво” не зменшується з плином часу, та навіть на деяких останніх часових ділянках більше за середню. Таким чином, можна зробити висновок, що варто просувати цю категорію курсів в подальшому.

Найбільший відсоток переглядів користувачів, які мають рівень освіти “Вища середня” – саме з України, з Польщі – найменший, тому варто прийняти рішення щодо створення нових курсів саме на польській мові.

Результати дослідження будуть використані в подальшому для розвитку системи, підготовці нових та модернізації старих електронних навчальних курсів.

НУБІП УКРАЇНИ

НУБІП УКРАЇНИ

НУБІП УКРАЇНИ

НУБІП УКРАЇНИ

НУБІП УКРАЇНИ

НУБІП УКРАЇНИ

НУБІП УКРАЇНИ

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Верховна рада України [Електронний ресурс]: Закон України про освіту – Режим доступу: <https://cutt.ly/1NfzaD1> (дата звернення 23.10.2022)
2. Освіта.Ua [Електронний ресурс]: Навчання онлайн – Режим доступу: <https://cutt.ly/zNfzE3t> (дата звернення 23.10.2022)
3. Baoviewanalytics [Електронний ресурс]: Blending In – Режим доступу: <https://cutt.ly/ANfzAJN> (дата звернення: 23.10.2022)
4. LucidWay [Електронний ресурс]: Characteristics of an Online Course – Режим доступу: <https://cutt.ly/KNH1g49> (дата звернення: 23.10.2022)
5. MyComputerCareer [Електронний ресурс]: What are Online Learning Platforms? – Режим доступу: <https://cutt.ly/jNH1W10> (дата звернення: 23.10.2022)
6. Software Testing Help [Електронний ресурс]: 15 Best Online Course Platforms & Websites In 2022 – Режим доступу: <https://cutt.ly/1NH1KjV> (дата звернення: 23.10.2022)
7. Visual Paradigm [Електронний ресурс]: Use Case Diagram Tutorial – Режим доступу: <https://cutt.ly/GhnBMk2> (дата звернення: 23.10.2022)
8. ИНТУИТ [Електронний ресурс]: Диаграмма активностей: крупным планом – Режим доступу: <https://cutt.ly/PhnNesP> (дата звернення 23.10.2022)
9. Guru99 [Електронний ресурс]: Activity Diagram in UML – Режим доступу: <https://cutt.ly/XhnnLKE> (дата звернення 23.10.2022)
10. Visual Paradigm [Електронний ресурс]: What is Component Diagram? Режим доступу: <https://cutt.ly/HNoQFCq> (дата звернення 24.10.2022)
11. Visual Paradigm [Електронний ресурс]: What is Deployment Diagram? Режим доступу: <https://cutt.ly/wNoEc97> (дата звернення 24.10.2022)
12. StudFiles [Електронний ресурс]: Подання архітектурних стилів – Режим доступу: <https://cutt.ly/JntfrSw> (дата звернення 24.10.2022)
13. InfoWorld – [Електронний ресурс]: What is OLAP? Analytical databases – Режим доступу: <https://cutt.ly/hNs7B11> (дата звернення 25.10.2022)
14. ЦДПУ – [Електронний ресурс]: Поняття та функції СКБД. Компоненти СКБД – Режим доступу: <https://cutt.ly/tNYRr32> (дата звернення 25.10.2022)

15. GitHub – [Електронний ресурс]: ClickHouse – Режим доступу: <https://cutt.ly/7NdqWll> (дата звернення 25.10.2022)

16. Aiven – [Електронний ресурс]: What is ClickHouse? – Режим доступу: <https://cutt.ly/BNUI5OZ> (дата звернення 25.10.2022)

17. Pidru4niki – [Електронний ресурс]: Створення сховищ даних. Технології OLAP та Data Mining – Режим доступу: <https://cutt.ly/INleopG> (дата звернення 25.10.2022)

18. Concepta – [Електронний ресурс]: What Is the Difference Between Front-End and Back-End Development? – Режим доступу: <https://cutt.ly/fNY62Vf> (дата звернення 26.10.2022)

19. Spring – [Електронний ресурс]: Spring Framework – Режим доступу: <https://cutt.ly/GNURx79> (дата звернення 26.10.2022)

20. Vue.js – [Електронний ресурс]: Introduction – Режим доступу: <https://cutt.ly/CNUusJu> (дата звернення 26.10.2022)

21. Scaleyourapp [Електронний ресурс]: What is Grafana? Why Use It? Everything You Should Know About It – Режим доступу: <https://cutt.ly/ONUG6hD> (дата звернення 26.10.2022)

22. CloudWays – [Електронний ресурс]: What Is DigitalOcean and Why Should You Select It for Your Web Hosting? – Режим доступу: <https://cutt.ly/RNY5e0f> (дата звернення 26.10.2022)

23. Citrix – [Електронний ресурс]: What is containerization? – Режим доступу: <https://cutt.ly/DNUR7zC> (дата звернення 26.10.2022)

24. Docker Docs – [Електронний ресурс]: Docker overview – Режим доступу: <https://cutt.ly/9NUP4u9> (дата звернення 26.10.2022)

25. DigitalOcean – [Електронний ресурс]: Kubernetes – Режим доступу: <https://cutt.ly/9NUSuzH> (дата звернення 26.10.2022)

26. HiTechNectar – [Електронний ресурс]: Pros and Cons of Kubernetes Explained – Режим доступу: <https://cutt.ly/aNN7GsC> (дата звернення 26.10.2022)

27. QALight – [Електронний ресурс]: Модульне тестування – Режим доступу: <https://cutt.ly/OnwhR7t> (дата звернення 25.10.2022)

28. QALight – [Електронний ресурс]: Інтеграційне тестування. – Режим доступу: <https://cutt.ly/eNpJCuE> (дата звернення 26.10.2022).

НУБІП України

НУБІП України

НУБІП України

НУБІП України

НУБІП України

НУБІП України

НУБІП України