

Вінницький національний технічний університет

Факультет інформаційних технологій та комп'ютерної інженерії

Кафедра програмного забезпечення

Бакалаврська дипломна робота

на тему: «Створення порталу курсів для дистанційного навчання»

Виконав: студент 4 курсу

групи 1П-18б

спеціальності

121 – Інженерія програмного забезпечення

(шифр і назва напрямку підготовки, спеціальності)

Мартос Н. М.

(прізвище та ініціали)

Керівник: к.т.н., доц. каф. ПЗ

Коваленко О.О.

(прізвище та ініціали)

Рецензент: к.т.н., доц. каф КН

Колесницький О.К.

(прізвище та ініціали)

Допущено до захисту

Зав. кафедри _____

« » _____ 2022 р.

Вінницький національний технічний університет
Факультет інформаційних технологій та комп'ютерної інженерії
Кафедра програмного забезпечення
Рівень вищої освіти перший бакалаврський
Галузь знань 12 – Інформаційні технології
Освітньо-професійна програма – Інженерія програмного забезпечення
ЗАТВЕРДЖУЮ
Завідувач кафедри ПЗ
Романюк О. Н.
25 березня 2022 р.

З А В Д А Н Н Я НА БАКАЛАВРСЬКУ ДИПЛОМНУ РОБОТУ СТУДЕНТУ

Мартосу Назару Максимовичу

1. Тема роботи – «Створення порталу курсів для дистанційного навчання»
Керівник роботи: Коваленко Олена Олексіївна , к.т.н., доцент кафедри ПЗ,
затверджені наказом вищого навчального закладу від 24 березня 2022 р. №66.
2. Строк подання студентом роботи 13 червня 2022 р.
3. Вихідні дані до роботи: Середовище розробки – Atom, Мови розробки – html, css, php, js, Операційна система – MacOS.
4. Зміст розрахунково-пояснювальної записки: вступ; аналіз стану питання та постановка задачі; проектування програмної системи; розробка програмного модуля порталу курсів для дистанційного навчання; тестування програмної частини порталу курсів для дистанційного навчання, висновки; список використаних джерел, додатки, графічна частина.
5. Перелік графічного матеріалу: демонстрація роботи аналогів, блок-схема функціонування програмного продукту, блок-схема взаємодії модулів програмного продукту, загальний алгоритм роботи додатку та тестування додатку

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1-4	Коваленко О.О., к.т.н., доцент кафедри ПЗ	26.03.22	10.06.22

7. Дата видачі завдання 25 березня 2022 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів бакалаврської дипломної роботи	Строк виконання етапів роботи	Примітка
1	Аналіз питання, обґрунтування актуальності розробки системи та постановка задачі	04.04.2022 – 15.04.2022	Вик.
2	Розробка моделей та графів роботи інформаційно-навчальної системи	18.04.2022 – 22.04.2022	Вик.
3	Вибір середовища та мови розробки	25.04.2022 – 02.05.2022	Вик.
4	Розробка програмного продукту	03.05.2022 – 23.05.2022	Вик.
5	Тестування роботи системи	24.05.2022 – 30.05.2022	Вик.
6	Оформлення матеріалів до захисту БДР	31.05.2022 – 10.06.2022	Вик.

Студент

(підпис)

Маргос Н.М.
(прізвище та ініціали)

Керівник бакалаврської дипломної роботи

(підпис)

Коваленко О.О.
(прізвище та ініціали)

Анотація

Бакалаврська дипломна робота складається з 50 сторінок формату А4, на яких є 46 рисунків, 2 таблиці, список використаних джерел містить 20 найменувань

У бакалаврській дипломній роботі проведено аналіз порталів курсів для дистанційного навчання, що являли собою поширені сервіси по наданню послуг дистанційного навчання. Було проведено аналіз аналогів, які існують, визначено їх переваги і недоліки.

На основі проведених досліджень було запропоновано розробити власний портал курсів для дистанційного навчання, при застосуванні розробленої моделі інформаційно-навчальної системи.

Розроблено портал «GlobalUaCourses», який несе навчальний характер і допомагає не лише вивчити теоретичну частину матеріалу, але й перевірити її практичним способом за допомогою тестування власних знань.

Розроблено модель організації навчального процесу та програмного реалізованого його основні задачі.

Програмний продукт було створено з використанням мови програмування PHP. Для розробки графічних елементів гри було використано HTML та CSS. В якості середовища розробки порталу було обрано редактор вихідного коду Atom.

Ключові слова: портал курсів, дистанційне навчання, інформаційно-навчальна система, PHP.

Abstract

The bachelor's thesis consists of 50 A4 pages, which include 46 figures, 2 tables, the list of sources used contains 20 items

In the bachelor's thesis the analysis of portals of courses for distance learning which represented various services on rendering of services of distance learning is carried out. An analysis of analogues that have been identified, their advantages and disadvantages have been identified.

Based on the research, it was proposed to develop its own portal of distance learning courses, using the developed model of information and training system.

Developed web-program «GlobalUaCourses», which is educational in nature and not only to study the theoretical part of the material, but also to test its practical way by testing your own knowledge.

The model of the organization of educational process and the program realized its basic tasks is developed.

The software product was created using the PHP programming language. HTML and CSS were used to develop the graphic elements of the game. The Atom source code editor was chosen to develop the portal environment.

Key words: portal of courses, distance learning, information-educational system, PHP.

ЗМІСТ

ВСТУП.....	7
1 АНАЛІЗ СТАНУ ПИТАННЯ ТА ПОСТАНОВКА ЗАДАЧІ РОЗРОБКИ.....	10
1.1 АНАЛІЗ СТАНУ ПИТАННЯ	10
1.2 ПОРІВНЯЛЬНИЙ АНАЛІЗ АНАЛОГІВ.....	11
1.3 ПОСТАНОВКА ЗАДАЧІ	17
1.4 ВИСНОВКИ.....	17
2 ПРОЕКТУВАННЯ ІНФОРМАЦІЙНО-НАВЧАЛЬНОЇ СИСТЕМИ	18
2.1 ОСНОВНІ ВИМОГИ ЩОДО ПРОЕКТУВАННЯ ВЕБ-ДОДАТКУ КУРСІВ ДЛЯ ДИСТАНЦІЙНОГО НАВЧАННЯ	18
2.2 МОДЕЛЮВАННЯ ІНФОРМАЦІЙНО-НАВЧАЛЬНОЇ СИСТЕМИ (ІНС).....	22
2.3 МОДЕЛІ ІНФОРМАЦІЙНО-НАВЧАЛЬНОЇ СИСТЕМИ	24
2.4 ВИСНОВКИ	34
3 РОЗРОБКА ПРОГРАМНОГО МОДУЛЯ ПОРТАЛУ КУРСІВ ДЛЯ ДИСТАНЦІЙНОГО НАВЧАННЯ	35
3.1 ОБҐРУНТУВАННЯ ВИБОРУ МОВИ РОЗРОБКИ ДОДАТКУ	35
3.2 РОЗРОБКА МОДУЛЯ БАЗИ ДАНИХ.....	36
3.3 РОЗРОБКА ПРОГРАМНИХ БЛОКІВ ПОРТАЛУ.....	43
3.4 ВИСНОВКИ	48
4 ТЕСТУВАННЯ ПРОГРАМНОЇ ЧАСТИНИ ПОРТАЛУ КУРСІВ ДЛЯ ДИСТАНЦІЙНОГО НАВЧАННЯ	49
4.1 ТЕСТУВАННЯ ПРОГРАМНОГО МОДУЛЯ АВТОРИЗАЦІЇ ТА РЕЄСТРАЦІЇ	49
4.2 ТЕСТУВАННЯ ПРОГРАМНОГО МОДУЛЯ СТВОРЕННЯ ТА ПРОХОДЖЕННЯ ТЕСТІВ	52
4.3 ВИСНОВКИ	58
ВИСНОВКИ	59
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	60
ДОДАТКИ	62
Додаток А – ТЕХНІЧНЕ ЗАВДАННЯ	63
Додаток Б – ПРОТОКОЛ ПЕРЕВІРКИ НА ПЛАГІАТ	67
Додаток В – ЛІСТИНГ ПРОГРАМИ	68
Додаток Г – ГРАФІЧНА ЧАСТИНА	90

ВСТУП

Обґрунтування вибору теми дослідження. За останні роки освітня система не тільки в Україні, але й у всьому світі, стикається з проблемами, які задають тенденцію створення нових платформ для дистанційного навчання. Ця ситуація є результатом пандемії «Covid-19» чинники якої полягають в наступному [1].

По-перше, попит на платформи для створення онлайн курсів буде рости, не тільки на внутрішньому, але й на міжнародному ринках. На внутрішньому ринку, де використання платформ для дистанційного навчання буде залишатися стабільним, дані платформи будуть користуватися підвищеним попитом серед населення країни. На міжнародному рівні, ситуація простіше, оскільки ці чинники не на стільки розвинуті як в нашій країні. Одночасно з цим буде спостерігатися підвищення конкурентоспроможності. Це буде відбуватися як наслідок розробки нових платформ іншими розробниками [2].

Тому, нові платформи для дистанційного навчання повинні найефективніше реагувати на потреби користувачів. Тенденції в області надання послуг дистанційного навчання буде розширено в зв'язку з розвитком діджиталізації по всьому світу. В цьому випадку, студенти вимагатимуть більше контролю і якості наданих сервісом відео курсів. Ця тенденція на пряму торкнеться інформаційну сферу, де наявність комп'ютерних систем і технологій дозволяє користувачам істотно підвищити якість послуг, що надаються. Для цього потрібна інтеграція багатьох інформаційних систем.

Оскільки існуючі портали курсів для дистанційного навчання не відповідають стандартам використання, то актуальним є питання створення нової платформи з метою підвищення зручності використання та ефективності.

Зв'язок роботи з науковими програмами, планами, темами. Робота виконувалася згідно плану виконання наукових досліджень на кафедрі програмного забезпечення.

Мета і завдання дослідження. Метою бакалаврської дипломної роботи є підвищення сприятливості використання та ефективності порталу курсів для дистанційного навчання для популяризації та збільшення кількості учнів що навчаються.

Задачі дослідження

1. Виконати аналіз предметної області з розробки порталу курсів для дистанційного навчання.
2. Виконати порівняльний аналіз аналогів, які являють собою поширені онлайн сервіси у вигляді веб-сайтів.
3. Визначити основні завдання які необхідно вирішити для розробки порталу курсів для дистанційного навчання.
4. Виконати проектування інформаційно-навчальної системи.
5. Розробити алгоритм розробки порталу курсів для дистанційного навчання.
6. Виконати тестування порталу курсів для дистанційного навчання.

Об'єкт дослідження – процеси функціонування порталу курсів для дистанційного навчання.

Предмет дослідження – методи та засоби побудови програмного додатку, що дозволяють виконувати підтримку платформ для дистанційного навчання.

Наукова новизна отриманих результатів. Подальшого розвитку отримав метод розробки моделі інформаційно-навчальної системи для організації дистанційного навчання який на відміну від аналогів створений для більш ефективно його роботи.

Практичне значення одержаних результатів. Розроблено алгоритми та методи для інформаційно-навчальної моделі використання порталу та оптимізації його роботи.

Особистий внесок здобувача. Усі наукові результати, викладені у бакалаврській дипломній роботі, отримані автором особисто. В опублікованій науковій роботі [3], автору належать такі результати: розробка порталу курсів для дистанційного

навчання, що базується на моделі інформаційно-навчальної системи; загальний аналіз моделей системи та графів надання складності навчального контенту.

Апробація матеріалів бакалаврської дипломної роботи. Основні положення бакалаврської дипломної роботи доповідалися та обговорювалися на науково-технічній конференції підрозділів Вінницького національного технічного університету (Вінниця 2022).

Публікації. Основні результати дослідження опубліковані в науковій роботі – в тезах доповіді на науково-технічній конференції підрозділів Вінницького національного технічного університету (НТКП ВНТУ) [3].

Аналіз. У пояснювальній записці до бакалаврської дипломної роботи було розглянуто 4 розділи та було використано 20 джерел.

1 АНАЛІЗ СТАНУ ПИТАННЯ ТА ПОСТАНОВКА ЗАДАЧІ РОЗРОБКИ

1.1 Аналіз стану питання

Сьогодні в Україні стає все популярнішим використання інформаційного простору, сучасних освітніх технологій дистанційного та електронного навчання. Тому виникає проблема у наданні навчального контенту, який буде враховувати загально-світові тенденції, що спрямовані на гнучкість та доступність освіти.

У сучасних умовах виникає потреба в отриманні вищої освіти дистанційно, що дає змогу навчатися без фізичного перебування у аудиторіях навчальних закладів. Це важливо студентам що працюють, а також людям з обмеженими можливостями. Саме дистанційне навчання, яке здійснюється із застосуванням новітніх інформаційно-освітніх технологій, надає таку можливість [4].

За концепцією розвитку дистанційної освіти в Україні дистанційна освіта це форма навчання, еквівалента очній, вечірній, заочній та екстернату, що реалізується, в основному, за технологіями дистанційного навчання.

Застосування технологій дистанційного навчання підвищує та модернізує ефективність освітнього процесу, що залежить від готовності самих студентів навчатися в умовах дистанційної освіти, а також значною мірою від рівня підготовки педагогів до реалізації дистанційного навчання. Не менш важливим для ефективного освітнього процесу в умовах дистанційного навчання є науково-методичне та матеріально-технічне забезпечення.

Метою популяризації дистанційного навчання є:

- Підтримка традиційного навчального процесу, коли за допомогою системи дистанційного навчання проводяться дистанційні курси;
- Підвищення якості освіти;
- Підготовка конкурентоспроможного фахівця здатного конкурувати на ринку праці;
- Залучення здобувачів освіти, які не в змозі відвідувати очні заняття;

- Залучення великої кількості слухачів.

Серед основних завдань використання методики дистанційного навчання можна виокремити такі: створення найзручніших умов для здобувачів освіти у здобутті ними вищої освіти; підвищення кваліфікації викладацького складу та перепідготовка кадрів на основі упровадження новітніх комунікаційних та педагогічних технологій навчання.

Для здійснення процесу дистанційного навчання викладачам та здобувачам освіти потрібен інструмент. Портал є одним з таких інструментів. Це система програмного забезпечення, яка дозволяє розміщувати курси, комунікувати, контролювати успішність студентів, здійснювати управління курсом та процесом навчання.

1.2 Порівняльний аналіз аналогів

Серед переваг дистанційного навчання для студента можна виділити: можливість підлаштовувати темп навчання під себе, виявити зони розвитку здобувачів освіти, проглядати навчальний матеріал повторно, зниження психологічного та фізичного навантаження, формування навички самоосвіти.

Однак, поряд з перевагами є і недоліки в інтеграції дистанційного навчання в освітній процес. До них можна віднести: обмеження соціального спілкування, мала кількість годин, що відведена на практичну частину навчання.

Можна зробити висновок, що сучасні інформаційно-комунікаційні технології не замінять живого спілкування студентів з викладачами, але з будь якої проблеми можна знайти нові можливості: для студентів – здобути навички самостійного навчання. Але, проблема залишається актуальною, саме тому в останні часи почало з'являтися все більше порталів курсів для дистанційного навчання.

Розглянемо переваги і недоліки найбільш поширених сайтів з курсами для дистанційного навчання (табл. 1.1).

Таблиця 1.1 – Порівняльна характеристика аналогів найбільш поширених порталів курсів для дистанційного навчання

Аналог	Функціональність	Кількість контенту	Зручність використання	Захищеність
EdEra	Середня	Висока	Висока	Середня
Prometheus	Середня	Висока	Середня	Низька
WiseCow	Низька	Середня	Середня	Висока
ВУМ	Низька	Низька	Низька	Середня

До одного з найпоширеніших відносять сайт Prometheus. Це одна з найбільших українських платформ масових відкритих онлайн-курсів. На ній розміщуються курси як від українських спеціалістів, так і переклади іноземних освітніх програм [5].

Там можна знайти курси на такі теми як: програмування, англійської мови, маркетингу, побудови онлайн-бізнесу, агрономії, дизайну, економіки, управління персоналом, письменності, управління громадами, антикризових комунікацій. Також, є курси для підвищення кваліфікації медиків, вчителів і новачків на державних посадах.

PROMETHEUS Курси ▾ Prometheus+ Корпоративне навчання

ПІДТРИМАЙТЕ НАШУ АРМІЮ
БФ «ПОВЕРНИСЬ ЖИВИМ»

Розкрийте свій потенціал

Опануйте актуальні навички на онлайн-курсах від кращих викладачів на найбільшій освітній платформі України

1 800 000
слухачів

250+
онлайн-курсів

1 000 000
виданих сертифікатів

7
років у сфері освіти

Рисунок 1.1 – головна сторінка сайту Prometheus

Ще одним з представників відомих сайтів онлайн-курсів є сайт WiseCow. Це портал українських відео лекцій, де зібрані короткі ролики по 10 хвилин. Ролики розповідають про українську культуру та історію [6].

Наразі, цей сайт пропонує лекції 9-ти тематик – це література, кіно, мистецтво, історія, журналістика, мода, міста і соціум. У кожному з розділів планується по 10 курсів. Кожен курс вміщує 10 відео та 10 додаткових матеріалів. Розділ «Місто», на відміну від інших, містить не ролики, а цікаві культурні ініціативи кожної з областей України.

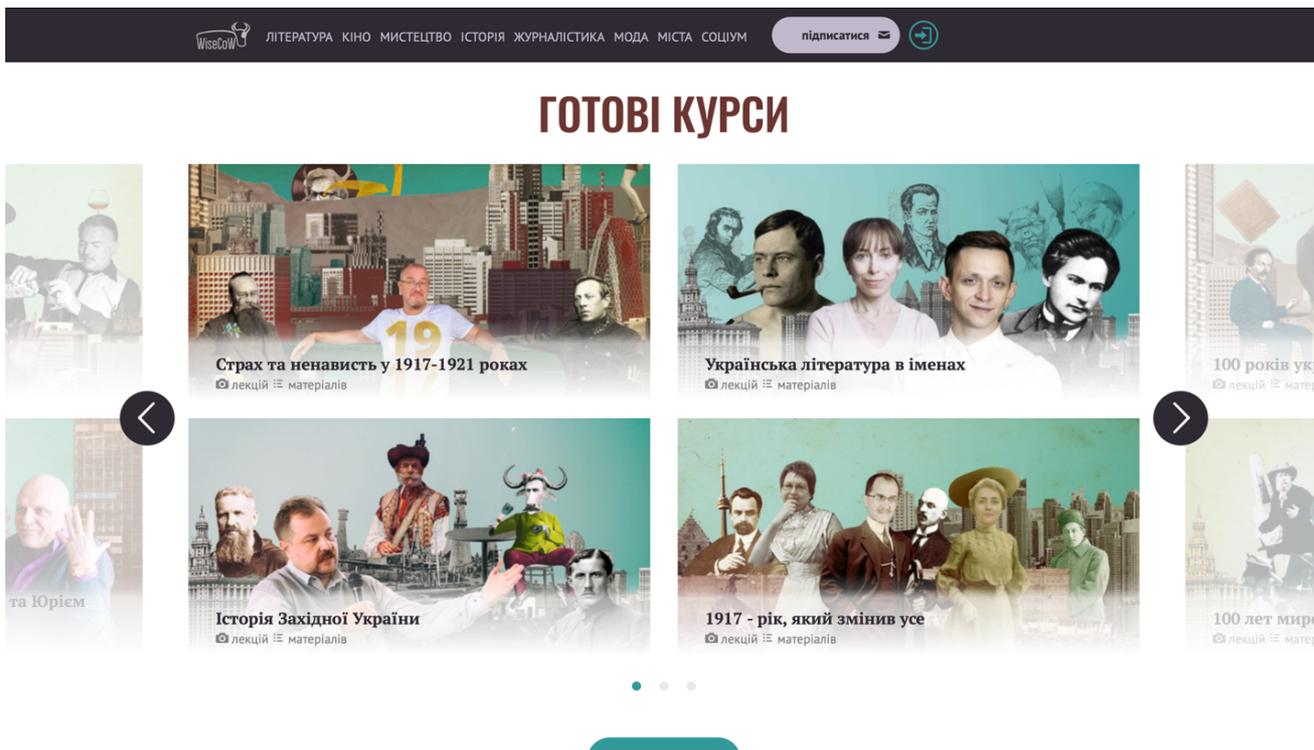


Рисунок 1.2 – головна сторінка сайту WiseCow

Також розглянемо портал для онлайн навчання ВУМ. Це освітня ініціатива, яка поширює ідеї та сприяє розвитку громадянського суспільства в Україні [7].

Більшість курсів пов'язані з персональним розвитком, підприємництвом, розумінням побудови та діяльності відкритого суспільства і його формування в Україні. На порталі є курси з правової грамотності, проектного менеджменту, креативного мислення, фінансів і інших тематик.

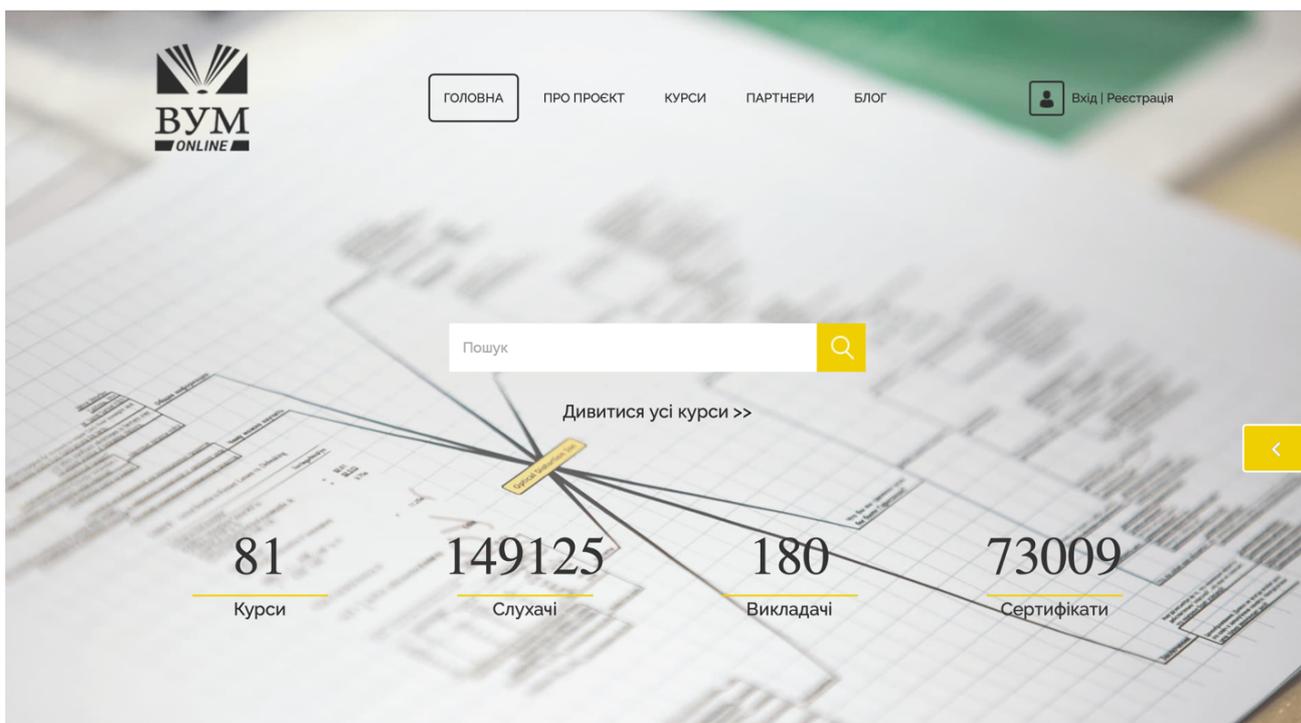


Рисунок 1.3 – головна сторінка сайту BVM Online

EdEra – це українська студія онлайн освіти. Команда платформи створює онлайн-курси, спецпроекти, інтерактивні підручники та освітні блоги.

На платформі є багато курсів для вчителів і матеріалів для учнів, які готуються до ЗНО. Також можна знайти курси по медицині, захисту прав, історії України, англійській мові, підприємництву, програмуванню, української мови та літератури, біології та математиці [8].

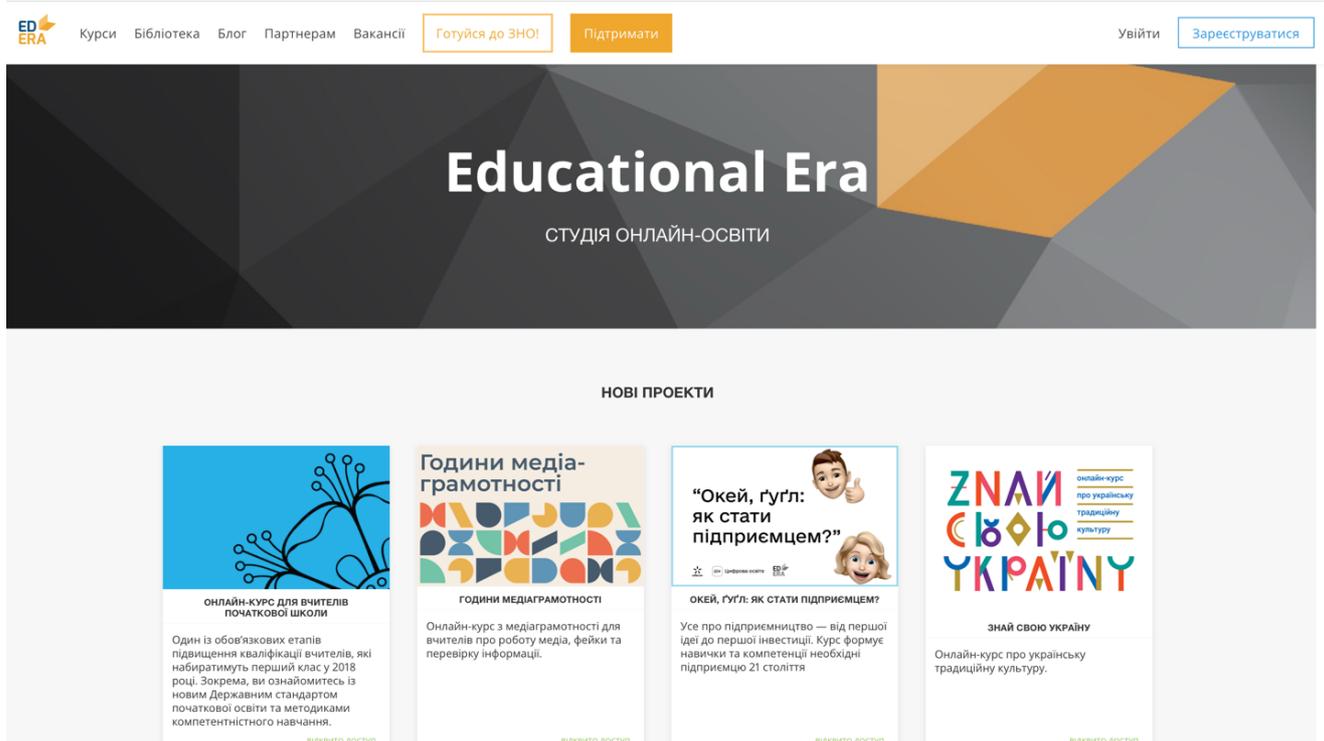


Рисунок 1.4 – головна сторінка сайту «EdEra»

Для зручності користувачам запропонована покрокова інструкція, яка надасть змоги досить швидко входити в систему, проходити тести, переглядати відео, ознайомлюватись з текстовою інформацією, яка знадобиться під час періодичного та підсумкового контролів. Інформація, яка розташована на сайті EdEra, полегшує роботу користувача та дозволяє побачити орієнтовані терміни публікації усіх навчальних матеріалів, що стосуються конкретного курсу. Додатки уміщують відповіді на часті запитання користувачів, які вивчають курс у дистанційному режимі та подають покрокову інструкцію з питань реєстрації на онлайн платформі. Користувачі, які не зможуть знайти відповіді на свої питання, можуть скористатися зворотнім зв'язком та вчасно отримати потрібну інформацію.

1.3 Постановка задачі

До вирішення загальних задач розробки порталу курсів для дистанційного навчання відносять такі кроки:

- Визначити ціль
- Вивчити цільову аудиторію
- Сформувати задачу навчання
- Підготувати матеріали

Постановка задачі, що полягає у розробці програмного забезпечення для порталу курсів для дистанційного навчання вимагає виконати наступні задачі:

- Виконати аналіз предметної області з комп'ютеризації дистанційного навчання.
- Виконати порівняльний аналіз стандартних веб-додатків, що виконують функцію сервісів для дистанційного навчання
- Визначити вимоги та критерії щодо оцінки якості веб-додатків курсів для дистанційного навчання
- Виконати проектування веб-додатку курсів для дистанційного навчання
- Розробити алгоритм та програмний код додатку курсів для дистанційного навчання на мові програмування PHP
- Виконати тестування додатку курсів для дистанційного навчання

1.4 Висновки

В першому розділі зроблено порівняльний аналіз аналогів які являли собою поширені онлайн сервіси у вигляді сайтів. В цьому випадку, були розглянуті такі важливі параметри функціонування веб-додатків з курсами для дистанційного навчання як функціональність веб-додатків, надійність у наданні навчального відео матеріалу, кількість контенту, зручність використання для студентів та для викладачів, захищеність про роботі з користувачами.

У постановці задачі визначені основні завдання які необхідно вирішити для розробки веб-додатку що впроваджується за допомогою розробки порталу курсів для дистанційного навчання.

2 ПРОЕКТУВАННЯ ІНФОРМАЦІЙНО-НАВЧАЛЬНОЇ СИСТЕМИ

2.1 Основні вимоги щодо проектування веб-додатку курсів для дистанційного навчання

Одним з найважливіших етапів створення порталу курсів є проектування розробки, це сприяє ефективній роботі розробників, а в подальшому більш налагодженій роботі самого додатку. Така процедура є невід'ємною частиною для вдалої реалізації будь-якого проекту. Адже, саме опрацювання всіх необхідних етапів розробки зможе гарантувати якість та відсутність будь-яких проблем на протязі всього процесу розробки додатку.

За останні часи, все більше компаній відкривають в себе ІТ-відділи, що займаються комп'ютеризацією усіх процесів компанії, зокрема створенням веб-платформи цієї компанії. Але далеко не всі компанії в змозі мати власних фахівців, що займаються створенням своїх платформ компанії. Тому, на допомогу їм приходять організації, в яких є висококваліфіковані фахівці, що займаються даним типом діяльності [9].

Зазвичай, таку процедуру планування створення порталу курсів потрібно впроваджувати під час проектування основних частин додатку, де її поділяють на декілька етапів, а саме (рис. 2.1):

1. Мобільна адаптивність.

Сучасний веб-додаток, не залежно від його призначення – чи це масштабний портал або інтернет-магазин повинен бути оптимізований під мобільні пристрої. Це дуже необхідно, адже велика частина людей що заходять, роблять це через смартфони і ця цифра невпинно росте з кожним роком все більше. Адаптивні веб-додатки підлаштовуються під масштаб екрану, на якому вони переглядаються, надаючи однаковий рівень користувальницької роботи на будь-якому пристрої та зберігають весь вміст. Тобто, веб-додатком можливо буде користуватися як з ноутбука, так і зі смартфона або планшета, не втрачаючи зручності використання [10].



Рисунок 2.1 – Етапи планування та реалізації розробки порталу курсів для дистанційного навчання

2. Проста навігація

Досвід користувача базується на тому, чи зможе він знайти те, що шукає. Зайшовши на портал, користувач повинен точно знати, де саме він перебуває, як він туди потрапив і як дістатися до того місця, де він хоче бути. Навігація на порталі повинна бути чіткою, зрозумілою та простою. Така навігація складається з деяких правил:

- Кількість розділів у меню має складатись лише з семи пунктів, щоб не створювати плутанину;
- Навігація повинна бути на видному місці, щоб її можна було легко знайти;
- Логотип повинен завжди повертатись на головну сторінку. Це дає змогу відвідувачам легко та швидко розпочати пошук;
- Оптимізований та зрозумілий дизайн. Використовувати простий шрифт для більшої зрозумілості. А також, належний інтервал, це особливо важливо для мобільних пристроїв.

3. Зрозумілий дизайн

Сучасний дизайн сайту повинен бути безпосередньо пов'язаний з особливостями платформи. Варто виділити найбільш важливі елементи дизайну за допомогою правильної візуальної ідентифікації. Такими елементами є:

- Головна картинка;
- Анімація і відео;
- Розширені описи продуктів і послуг;
- Добре підібрані шрифти;
- Правильно підібрані кольори;
- Мінімалізм;
- Адаптація під мобільні пристрої;
- Ілюстрації.

4. Цінний контент

Ще однією важливою складовою, на яку слід звернути увагу, є вміст, який використовується у веб-додатках. Текст веб-додатку повинен бути інформативним, унікальним та легким для читання.

Щоб створити змістовний контент, потрібно буде опрацювати вподобання цільової аудиторії та створити вміст, що базується на її потребі. Ще одна причина для адаптації вмісту до аудиторії – це SEO. Тобто, ключові слова, що будуть використовуватись, допоможуть порталу розміститись у потрібних місцях у пошуковій системі. Основна причина, по якій люди відвідують освітні веб-сервіси – це інформація, тому потрібно переконатись, що вона доступна, розбірлива та зрозуміла.

5. Швидкість завантаження сторінок сайту

Одним з найважливіших показників якісної та ефективної роботи порталу є швидкість з якою завантажуються його сторінки [11].

Якщо для завантаження сторінки потрібно більше 5 секунд, це може спричинити розчарування у відвідувачів, і вони, ймовірно, вийдуть. Основними чинниками оптимізації веб-додатку є:

- Видалення будь-якого «непотребу»;
- Використання картинок або ілюстрацій з невеликим розміром;
- Використання кешування браузера, щоб зберігати кешовані версії статичних ресурсів;
- Оптимізація веб-додатку під смартфони та планшети;

2.2 Моделювання Інформаційно-навчальної системи (ІНС)

Інформаційно-навчальна система (ІНС) – це людино-машинний комплекс, що реалізує сценарії навчальної діяльності, та підготовлених знань (структурованої інформації та системи вправ для її осмислення та закріплення), що призначений для управління навчальною діяльністю, метою якої є оволодіння знань, умінь та навичок.



Рисунок 2.1 – Схема процесу навчання в інформаційно-навчальній системі

Навчальна система повинна навчати теорії з використанням практичних навичок. Вивчення лише теоретичного матеріалу ще не є навчанням. Отже, навчальна система – ширше поняття, ніж електронний підручник. Вона повинна включати теоретичний матеріал з прикладами (тобто електронний підручник), а також засоби для вироблення практичних навичок у учнів і засоби контролю набутих знань, умінь і навичок (контролюючу систему і програму, що проводить контроль).

Основне призначення навчання, отже, і навчальної системи - оволодіння вміннями, а не тільки знаннями. Механізм здійснення діяльності є вирішення завдань. Отже, переважна більшість навчальної системи – тренуюча.

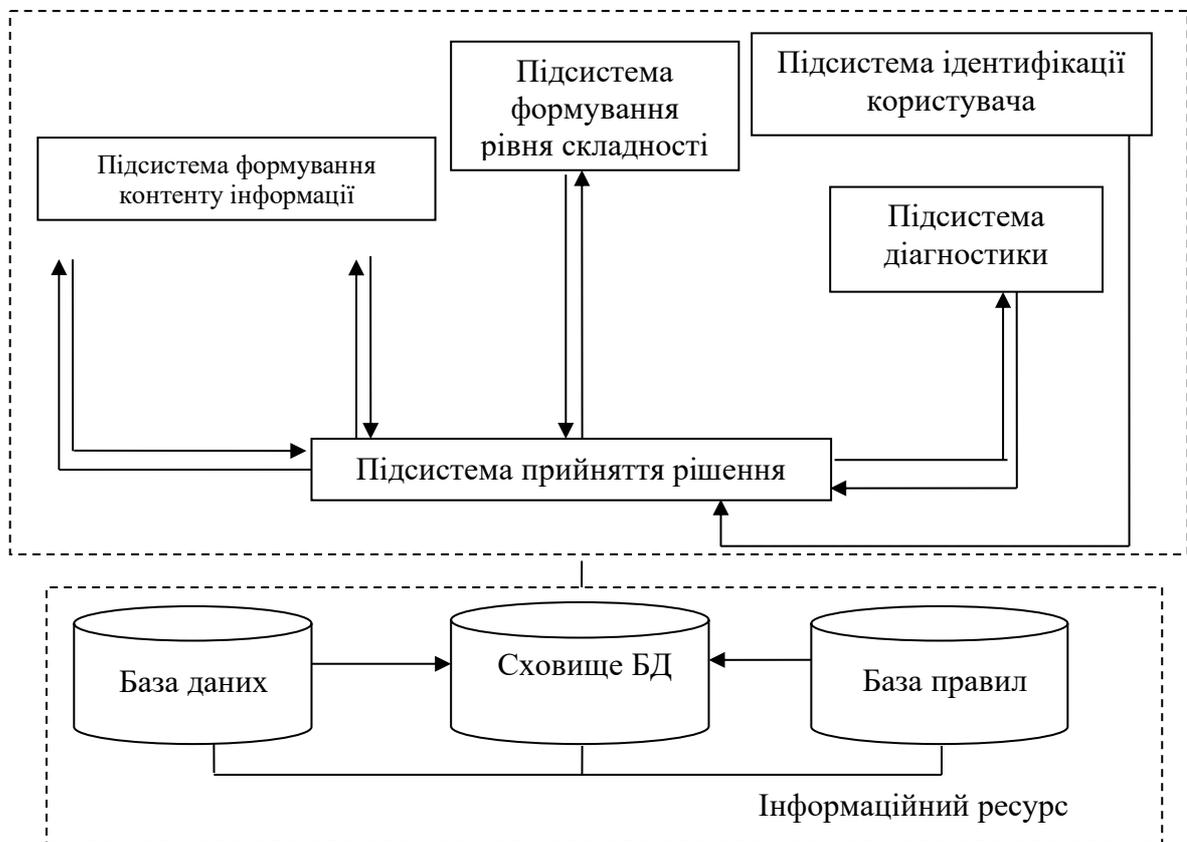


Рисунок 2.2 – Узагальнена структурна схема інформаційно-навчальної системи

Сценарій навчання в ІНС формується динамічно у співвідношенні з текучою ситуацією. Реалізація здійснюється на підставі протоколу процесу навчання на кожному навчальному елементі.

Розглянемо коротко призначення всіх складових:

- підсистема ідентифікації користувача призначена для персоналізації учня;
- підсистема формування контенту інформації призначена для визначення та формування «порцій інформації» теоретичного, практичного та контролюючого модулів;
- підсистема формування рівня складності визначає рівень складності матеріалу, що вивчається;
- підсистема діагностики призначена для контролю знань учня, здійснює розрахунок рівня знань учня за суперкритерієм багатокритеріальної моделі знань, враховуючи рівень реакції, сумніву, впевненості та ін. критерії;
- підсистема прийняття рішень призначена для прийняття рішень про формування послідовності навчання, кількість завдань, вибір рівня складності і т.д., завдяки багатокритеріальній моделі прийняття рішень.

2.3 Моделі Інформаційно-навчальної системи

1. Модель організації контенту навчального матеріалу

Модель формування контенту інформаційно-навчальної системи показує, у якій послідовності повинні вивчатися теми, які логічні зв'язки між ними, як можна вивчити даний матеріал за найменшу кількість кроків, який контент необхідно формувати при проведенні тестування, який вплив складності контенту навчальних елементів на формування складності наступного елемента при паралельному та послідовному вивченні матеріалу. З яких чинників складається адаптація складності контенту, скільки часу необхідно витратити для вивчення навчального елемента та всього контенту навчального матеріалу.

На початковому етапі проектування моделі контенту системи, навчальний матеріал розбивають на окремі навчальні елементи x_i або, як їх називають у міжнародних специфікаціях електронного навчання SCORM та на об'єктах змісту, що використовуються спільно (Sharable Content Object-SCO).

Вид моделі засвоєння контенту навчального матеріалу значною мірою визначається змістом і формою представлення навчального матеріалу, а ці фактори, у свою чергу, залежать від суб'єктивних дидактичних поглядів авторів змісту ІНС. Тому процедури формування моделі освоєння є інтерактивними за своєю суттю та передбачають участь авторів змісту.

У моделі розглядаються математичні аспекти цих процедур, що дозволяють автоматизувати побудову моделей освоєння ІНС, що не менш важливо, здійснювати їх повноцінний дидактичний аналіз.

Для математичного опису моделі освоєння будемо використовувати елементи теорії графів і фрагменти теорії відносин відповідно до визначень і символіки.

- Побудову моделі виробляють у шість етапів:
 - формування матриці відносин черговості навчальних елементів;
- обробка матриці відносин черговості;
- побудова послідовності вивчення навчального матеріалу як списку навчальних елементів;
 - формування матриці логічних зв'язків навчальних елементів;
 - побудова графа логічних зв'язків навчальних елементів;
 - побудова графа складності контенту при послідовному та паралельному вивченні навчальних елементів;

2. Модель відношення черговості навчальних елементів

Модель відносин черговості навчальних елементів контенту показує, якою послідовності повинні вивчатися теми навчального матеріалу [12].

Модель будуватися у три етапи:

- формування матриці відносин черговості навчальних елементів;

- обробка матриці відносин черговості;
- побудова послідовності вивчення навчального матеріалу як списку навчальних елементів.

Матриці відносин черговості навчальних елементів є квадратними. Розмір її дорівнює кількості навчальних елементів.

Алгоритм:

1. Будуємо матрицю суміжності A орграфа відношення черговості (V, R) . Відповідна інтерактивна процедура передбачає аналіз експертом (автором змісту навчального матеріалу) лише попарних відносин черговості навчальних елементів. Цей експертний аналіз можна скоротити вдвічі та аналізувати відносини черговості тільки для верхнього, або нижнього трикутника матриці. Інший трикутник матриці може бути заповнений автоматично на основі властивостей асиметричності та негативної асиметричності (наприклад, якщо в заповненій експертом комірці матриці (a, b) стоїть одиниця, то в комірці (b, a) ставиться нуль, і, навпаки, якщо у заповненій експертом комірці матриці (a, b) стоїть нуль, то в комірці (b, a) ставиться одиниця). Таким чином, виключаються потенційні помилки експерта, що порушують властивості асиметричності та негативної асиметричності.

Далі послідовність освоєння навчальних елементів визначається формальним чином. Уявимо безліч навчальних елементів у вигляді вектора $V = (v_1, v_2, v_3, \dots, v_n)$, де v_1, v_2, v_3 - *номера* навчальних елементів у моделі змісту навчального матеріалу. Введемо вектор $F = (g(v_1), g(v_2), g(v_3), \dots, g(v_n))$ — вектор порядкових номерів навчальних елементів у послідовності їх вивчення. Відповідно до формули 1.

$$F = E^t (A + I), \quad (1)$$

де A – матриця суміжності орграфа бінарного відношення черговості (V, R) ; I - одинична матриця розміром n ; E - вектор-стовпець з n одиниць.

Вище вже зазначалося, як уникнути помилок матриці A експерта, що порушують властивості асиметричності або негативної асиметричності. Помилки,

що порушують властивості транзитивності або негативну транзитивність, призводять до появи у векторі F однакових значень. Таким чином, діагностика подібних помилок є досить простою і може здійснюватися автоматично, але їх виправлення вимагає залучення експерта (автора змісту навчального матеріалу) для виявлення допущених ним порушень транзитивності або негативної транзитивності.

Будуються порожні матриці та нумеруються їх рядки та стовпці відповідно до зростання навчальних елементів.

$$A = \begin{bmatrix} 1,1 & 1,2 & 1,3 & \dots & 1,n \\ 2,1 & 2,2 & 2,3 & \dots & 2,n \\ 3,1 & 3,2 & 3,3 & \dots & 3,n \\ \dots & \dots & \dots & \dots & \dots \\ n,1 & n,2 & n,3 & \dots & n,n \end{bmatrix}, \quad (2)$$

де N – кількість навчальних елементів

2. Виконується строкове заповнення осередків матриць нулями та одиницями відповідно до рисунку 3.

	1	2	3	4	5	Σ
1	1	0	0	1	1	3
2	1	1	1	1	1	5
3	0	0	0	0	1	1
4	1	1	1	1	0	4
5	1	1	0	0	0	2

Рисунок 3 – Матриця відношень черговості начальних елементів

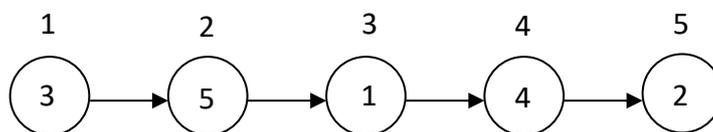


Рисунок 4 - Порядок вивчення навчальних елементів

Послідовність вивчення навчальних елементів у покроковій процедурі навчання визначають у процесі обробки матриці відносин черговості, підсумовуючи коефіцієнти кожного рядка матриці. Отримані суми записують у колонці праворуч від матриці. Величини сум вказують порядкові номери відповідних навчальних елементів у списку послідовності вивчення навчального матеріалу.

3. Заповнюють матрицю відносин черговості між двома навчальними елементами. Одиниці ставлять у комірку, якщо навчальний елемент, вказаний у номері рядка, повинен вивчатися після навчального елемента, вказаного у номері стовпця. Протилежне відношення черговості позначають банкрутом або залишають відповідну комірку матриці порожньою. Усі осередки головної діагоналі матриці відносин черговості заповнюють одиницями. Осередки матриці, симетричні щодо головної діагоналі, повинні мати протилежні значення. Тому аналіз парних відносин черговості можна проводити лише для лівого нижнього або для правого верхнього трикутника матриці, заповнюючи її частину, що залишилася на основі властивості антисиметрії [13].

Процес заповнення матриць доцільно вести, маючи тексти з навчальним матеріалом за всіма навчальними елементами. Аналіз змісту навчального матеріалу дозволяє об'єктивніше виявляти парні відносини черговості та логічні зв'язки між навчальними елементами.

3. Модель логічних зв'язків контенту системи

Нехай V — кінцева кількість номерів навчальних елементів розміром n . На цій множині визначимо бінарне відношення (V, L) , сенс якого для деяких $a, b \in V$ і aLb означає, що навчальний елемент b логічно пов'язаний з навчальним елементом a ("опирається" на нього), тобто при викладі змісту навчального елемента b використовуються поняття з a .

При практичній підготовці моделі освоєння після заповнення матриці суміжності відношення черговості та побудови послідовності вивчення навчального елементу заповнюють матрицю суміжності (V, L) . При цьому експерт – автор змісту облікового матеріалу – проводить аналіз попарних відносин логічної зв'язності (спірності) навчального елементу. У комірку (a, b) матриці стає одиниця, якщо навчальний елемент a є опорним для навчального елементу b . Інакше комірка матриці ставиться нуль, але зменшити трудомісткість цього експертного аналізу, якщо формально використовувати властивість асиметричності відношення логічної зв'язності. Наприклад, якщо в заповненій експертом комірці матриці (a, b) стоїть одиниця, то в комірку (b, a) автоматично ставиться нульовим чином, виключаються потенційні помилки експерта, що порушують властивості асиметричності [14].

При заповненні матриці логічних зв'язків навчальних елементів ставлять одиницю в комірку, якщо навчальний матеріал навчального елемента, вказаного в рядку, логічно пов'язаний з навчальним матеріалом навчального елемента, вказаного в номері стовпця. Упорядкування матриці логічних зв'язків зручно вести з урахуванням матриці відносин черговості шляхом виключення одиниць із тих осередків, котрим відсутні логічні, опорні зв'язок між елементами .

Модель логічних зв'язків навчального матеріалу комплексу визначає послідовність його викладу в навчальній системі, варіанти траєкторій вивчення, логічні зв'язки при побудові гіпертексту.

На вигляд матриць відносин черговості і логічних зв'язків впливають як об'єктивні, так і суб'єктивні чинники: смаки розробника, його звички, інтуїтивні уявлення, склад мислення тощо.

Алгоритм:

1. Формування матриці логічних зв'язків із темами навчального матеріалу відповідно до малюнку 2.5:

	1	2	3	4	5	6	7	8	9	10
1	1									
2	1	1								
3			1	1						
4			1		1					
5				1		1				
6			1				1			
7	1			1				1		
8					1		1		1	
9						1		1		1
10	1			1	1	1		1	1	1

Рисунок 2.5 – Матриця логічних зв'язків з темами навчального матеріала

2. Побудова графа логічних зв'язків контенту системи відповідно до рисунка 2.6:

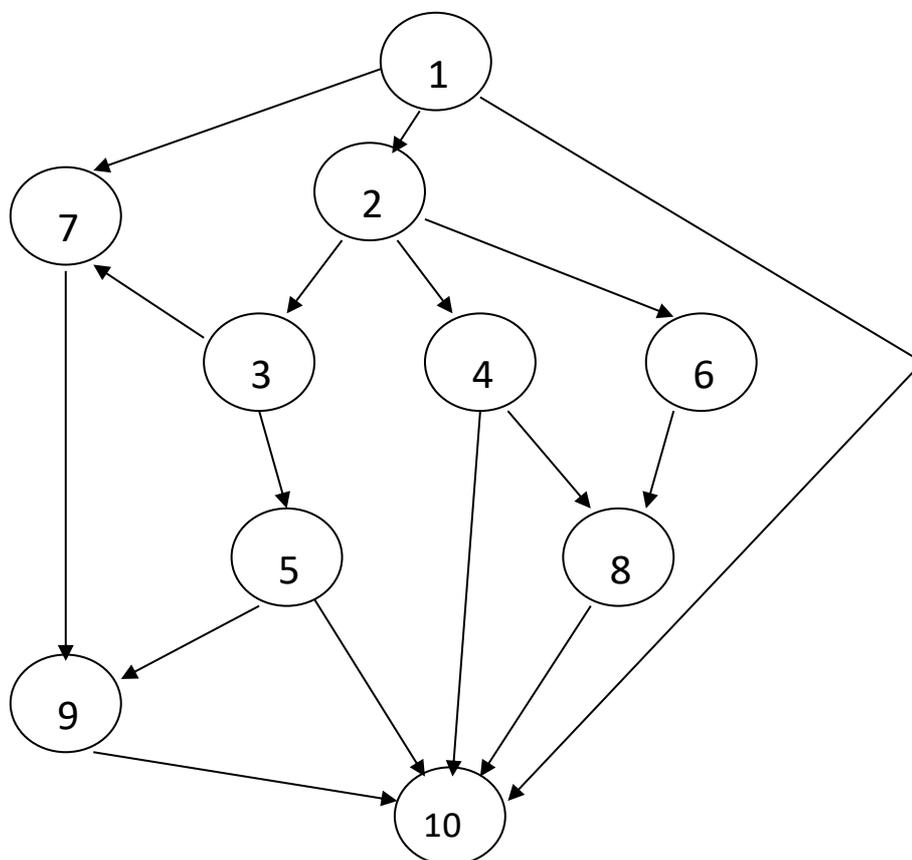


Рисунок 6 – Граф логічних зв'язків

3. З орієнтованого графа логічних зв'язків контенту системи можна вивести наступну залежність (за умови, що теми вивчаються в порядку від 1 до 9):

Для вивчення 2 теми необхідно обов'язкове вивчення 1 теми.

Для вивчення 3 теми необхідно вивчення 1,2 тем.

Для вивчення 4 теми – вивчення лише 1,2 тем.

Для вивчення 5 тем – вивчення лише 1,2,3 тем.

Для вивчення 6 тем – вивчення 1,2 тем.

Для вивчення 7 тем – вивчення лише 1 теми.

Для вивчення 8 тем – вивчення 1,2,4,6 тем.

Для вивчення 9 тем – вивчення 1,2,3,5,7 тем.

На графі видно, що до найважливіших тем навчального матеріалу відносять навчальні елементи, яких відходить більше одного ребра. Це учбові елементи 2,5,4.

Елемент, куди входить і виходить лише одне ребро є малозв'язним елементом. Його виявлення може означати його низьку важливість у системі і, можливо, прийняття рішення про його видалення та переосмислення послідовності та логічного зв'язку навчальних елементів.

Ставлення логічної зв'язності моделі освоєння є ефективним механізмом допомоги учням у формуванні контенту навчального матеріалу ІНС. Зокрема, під час перегляду ІНС, використовуючи фрагменти графа логічних зв'язків навчальних елементів, можна цілеспрямовано повертатися до потрібних раніше пройдених навчальних елементів.

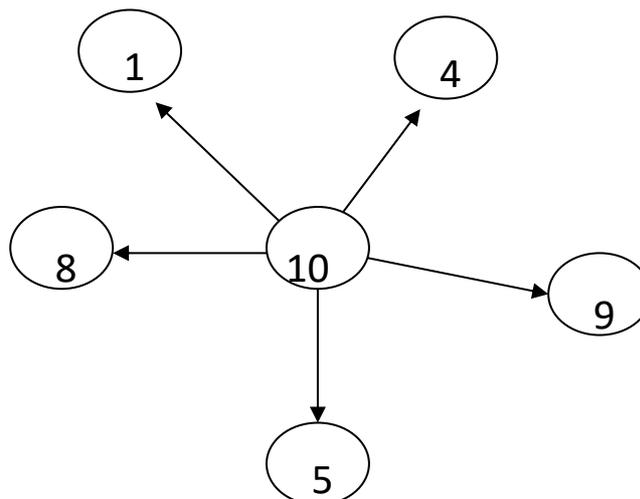


Рисунок 2.7 – Ключові навчальні елементи всього контенту навчального матеріалу

Відповідно до рисунка 2.7 граф показує, що для адаптивного тестування (тестування з найменшою кількістю кроків навчальних елементів) всього контенту навчального матеріалу, достатньо проходження 1,4,5,8, та 9 навчальних елементів, тому що всі інші навчальні елементи пов'язані з цими ключовими.

4. Граф складності контенту при послідовному вивченні навчальних елементів

Граф складності контенту при послідовному вивченні навчальних елементів формує наступну складність відповідно до рисунка 2.8, залежно від освоєння попередньої складності за наступним алгоритмом:

Послідовне вивчення – це процес, закладений викладачем за умовчанням у навчальній системі.

Рівень А – матеріал із основного підручника.

Рівень – матеріал, що вимагає логічного мислення з основного матеріалу підручника.

Рівень З – глибокий матеріал із проміжними етапами розв'язання (завдання).

Спочатку формується перший елемент (тобто, вивчається перша тема) рівня А, досить легка. У разі її освоєння здійснюється перехід на рівень елемента даної теми, інакше – перехід на наступну тему рівня А.

У разі переходу на рівень формується тема складніше. При повному освоєнні цього елемента здійснюється перехід на рівень цієї теми, інакше – перехід на наступну тему рівня А.

У разі освоєння рівня перехід на рівень З цієї теми. При освоєнні рівня З перехід елемент наступної теми рівня А.

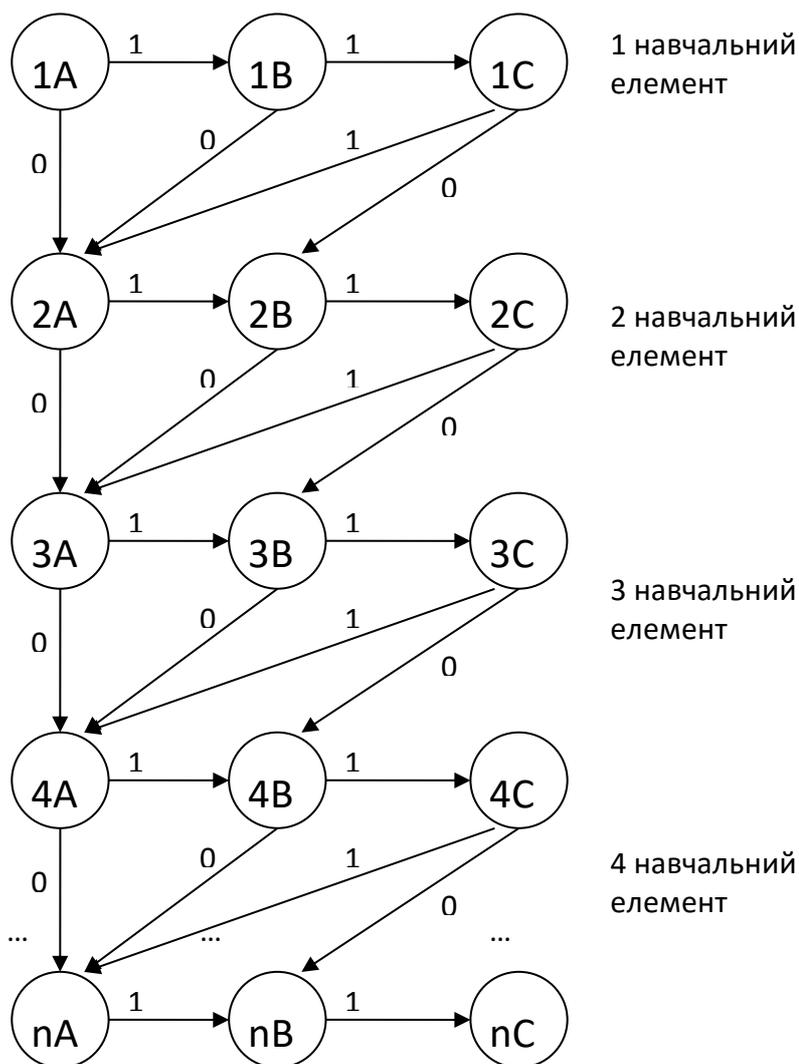


Рисунок 2.8 – Граф складності навчальних елементів контенту

(1 – освоєння навчального елемента, 0 – не освоєння навчального елемента)

В результаті цього алгоритму результат освоєння залежить від рівня, на якому користувач опиниться в кінці освоєння матеріалу.

Освоєння лише рівня А вважається задовільну оцінку. Освоєння рівня на добре. Освоєння на рівні С – відмінно.

2.4 Висновки

У другому розділі було проаналізовано основні вимоги до проектування порталу курсів для дистанційного навчання. Було виділено основні етапи на яких потрібно зробити основний наголос. Такими етапами є: мобільна адаптація, проста навігація зрозумілий дизайн, цінний контент та швидкість завантаження.

У другій та третій частинах другого розділу було проектування моделі інформаційно-навчальної системи та її підмоделей. Були розроблені схеми процесу навчання в інформаційно-навчальній системі, а також узагальнена схема інформаційно-навчальної системи, яка допомагає наглядно розглянути основні принципи роботи додатку.

Було розглянуто принципи створення та заповнення матриць методом зростання навчальних елементів. Був розроблений алгоритм порядку вивчення навчальних елементів. Також, у роботі було розроблено граф логічних зв'язків інформаційно-навчальної системи.

Основною роботою представленої інформаційно-навчальної системи є надання користувачу ключових навчальних елементів всього контенту навчального матеріалу. Одним зі складових чинників принципу роботи є модель складності навчальних елементів контенту. Тому при роботі з розробленим додатком, система буде визначати вподобання та рівень складності курсів які користувач буде потребувати.

3 РОЗРОБКА ПРОГРАМНОГО МОДУЛЯ ПОРТАЛУ КУРСІВ ДЛЯ ДИСТАНЦІЙНОГО НАВЧАННЯ

3.1 Обґрунтування вибору мови розробки додатку

C++ - універсальна ООП мова загального призначення, що застосовується у щонайширшій галузі розробки ПЗ, в тому числі в розробці операційних систем та високонавантажених систем. C++ підтримує декілька парадигм програмування: об'єктно-орієнтованої, узагальненої та процедурної. Компілятори C++ створені для якнайширшого діапазону архітектур ЕОМ та операційних систем [15].

Сучасний JavaScript - це "безпечна" мова програмування. Він не надає низькорівневий доступ до пам'яті або процесора, тому що спочатку був створений для браузерів, які цього не вимагають [16].

Можливості JavaScript залежать від оточення, в якому він працює. Наприклад, Node.JS підтримує функції читання/запису довільних файлів, виконання мережевих запитів тощо.

У браузері для JavaScript доступне все, що пов'язане з маніпулюванням веб-сторінками, взаємодією з користувачем та веб-сервером.

Наприклад, у браузері JavaScript може:

- Додавати новий HTML-код на сторінку, змінювати існуючий вміст, модифікувати стилі.
- Реагувати на дії користувача, клацання миші, переміщення вказівника, натискання клавіш.
- Відправляти мережеві запити на віддалені сервери, завантажувати та завантажувати файли (технології AJAX та COMET).
- Отримувати та встановлювати куки, ставити запитання відвідувачу, показувати повідомлення.
- Запам'ятовувати дані на стороні клієнта ("local storage").

PHP – це поширена мова програмування загального призначення з відкритим вихідним кодом. PHP спеціально сконструйований для веб-розробок, і його код може впроваджуватися безпосередньо в HTML [17].

PHP відрізняється від JavaScript тим, що PHP-скрипти виконуються на сервері та генерують HTML, який надсилається клієнту. Якби на сервері був розміщений скрипт, який би видавав текст, клієнт отримав би тільки результат його виконання, але не зміг би з'ясувати, який саме код його зробив. Ви навіть можете налаштувати свій сервер таким чином, щоб звичайні HTML-файли оброблялися процесором PHP, тому клієнти навіть не зможуть дізнатися, чи отримують вони звичайний HTML-файл або результат виконання скрипта.

Порівняння наведених мов програмування представлено таблицею 3.1.

Таблиця 3.1 – Порівняння мови програмування

	C++	JavaScript	PHP
Швидкодія	0	1	1
Використання апаратних можливостей	0	1	1
Зручність використання	1	0	1
Оптимізація коду	0	1	1
Загалом	1	3	4

На основі таблиці порівняння мов програмування для розробки веб-додатку була обрана мова PHP: вона дає змогу користуватись повними можливостями апаратного забезпечення та створювати швидкодіючі додатки.

3.2 Розробка модуля бази даних

Для створення та адміністрування бази були використано PhPMyAdmin.

phpMyAdmin — це програмне забезпечення з відкритим вихідним кодом, представлене 9 вересня 1998 року, написане на PHP. По суті, це сторонній інструмент для керування таблицями та даними всередині бази даних. phpMyAdmin підтримує різні типи операцій на MariaDB і MySQL. Основною метою phpMyAdmin є керування адмініструванням MySQL через Інтернет [18].

Це найпопулярніший додаток для керування базами даних MySQL. За допомогою цього програмного забезпечення ми можемо створювати, оновлювати, скидати, змінювати, видаляти, імпортувати та експортувати таблиці бази даних MySQL. phpMyAdmin також підтримує широкий спектр операцій, таких як керування базами даних, відносинами, таблицями, стовпцями, індексами, дозволами, користувачами тощо, на MySQL та MariaDB. Ці операції можна виконувати через користувальницький інтерфейс, при цьому ми все ще маємо можливість виконувати будь-який оператор SQL.

phpMyAdmin перекладено 72 мовами, а також підтримує мови RTL і LTR, щоб широкі коло людей могло легко використовувати це програмне забезпечення. Ми можемо виконувати запити MySQL, ремонтувати, оптимізувати, перевіряти таблиці, а також виконувати інші команди керування базою даних. phpMyAdmin також можна використовувати для виконання адміністративних завдань, таких як створення бази даних, виконання запитів.

phpMyAdmin — це програма на основі графічного інтерфейсу, яка використовується для керування базою даних MySQL. Ми можемо вручну створити базу даних і таблицю і виконати запит до них. Він забезпечує веб-інтерфейс і може працювати на будь-якому сервері. Оскільки він є веб-основою, ми можемо отримати до нього доступ з будь-якого комп'ютера [19].

phpMyAdmin підтримує кілька функцій, наведених нижче:

- phpMyAdmin може створювати, змінювати, переглядати та скидати бази даних, представлення даних, таблиці, стовпці та індекси.
- Він може відображати кілька наборів результатів за допомогою запитів і збережених процедур.
- phpMyAdmin використовує збережені процедури та запити для відображення кількох наборів результатів.
- Він підтримує зовнішні ключі та таблиці InnoDB.
- phpMyAdmin може відстежувати зміни, внесені в базах даних, представленнях і таблицях.
- Ми також можемо створити PDF-графіку макета нашої бази даних.
- phpMyAdmin можна експортувати в різні формати, такі як XML, CSV, PDF, ISO/IEC 26300 - OpenDocument Text and Spreadsheet.
- Він підтримує mysqli, який є покращеним розширенням MySQL.
- phpMyAdmin може взаємодіяти з 80 різними мовами.
- phpMyAdmin може редагувати, виконувати та створювати закладки будь-які оператори SQL і навіть пакетні запити.
- Використовуючи набір попередньо визначених функцій, він може перетворювати збережені дані в будь-який формат. Наприклад - BLOB-дані як зображення або посилання для завантаження.
- Він забезпечує можливість резервного копіювання бази даних у різних формах.

Перевагами phpMyAdmin є:

- phpMyAdmin може працювати на будь-якому сервері або будь-якій ОС, оскільки він має веб-браузер.
- Можливість легко створювати, видаляти та редагувати базу даних і керування всіма елементами за допомогою графічного інтерфейсу

phpMyAdmin, який набагато простіше, ніж редактор командного рядка MySQL.

- phpMyAdmin допомагає контролювати дозвіл користувача та працювати з кількома серверами одночасно.
- Можливість створювати резервну копію наявної бази даних та експортувати дані в різні формати, як XML, CSV, SQL, PDF, OpenDocument Text, Excel, Word та електронні таблиці тощо.
- Можливість виконувати складні оператори та запити SQL, створювати та редагувати функції, тригери та події за допомогою графічного інтерфейсу phpMyAdmin.

Під час розробки порталу курсів для дистанційного навчання було створено бази даних для: учнів, викладачів та курсів. В кожного елемента системи є свій унікальний ідентифікатор (ID) що допомагає системі не здійснювати помилок.

Приклад створення бази даних для процесу реєстрації продемонстровано на рисунку 3.1.

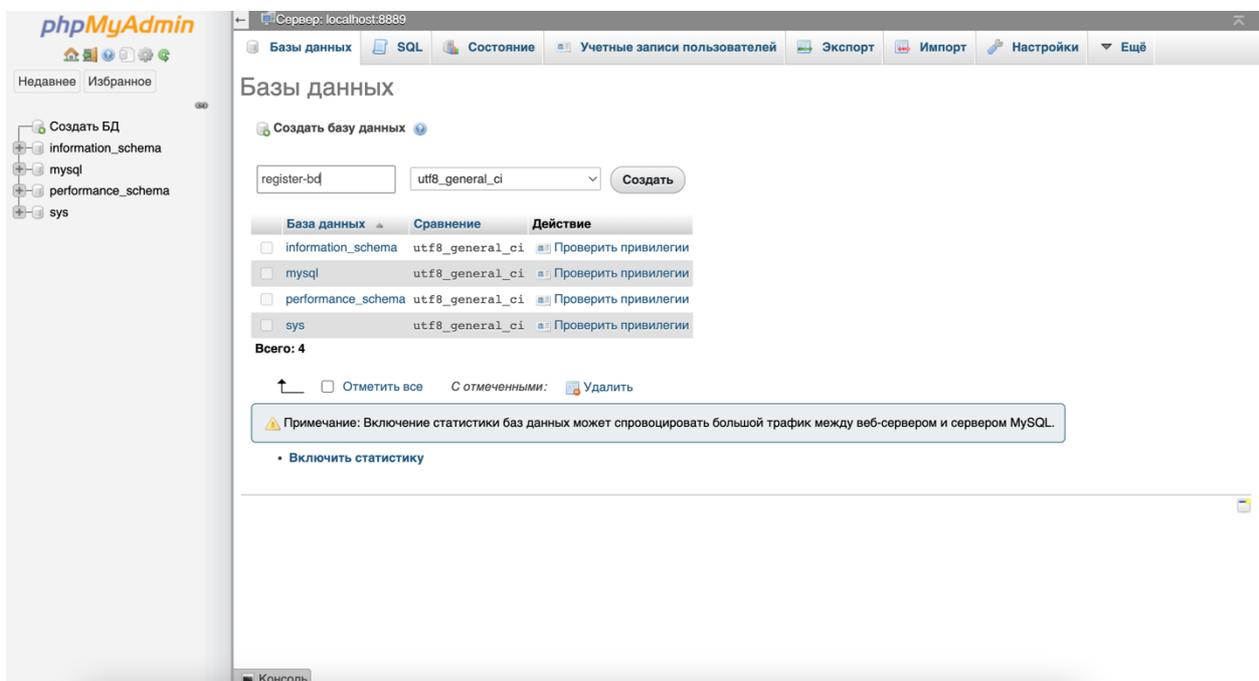


Рисунок 3.1 – Процес створення бази даних

Спочатку йде процес створення таблиці бази даних. Де вказується назва таблиці, кількість стовпців та тип кодування. Було обрано тип кодування `utf8_general_ci`, оскільки він є стандартом для баз даних, та на відміну від `utf8_bin` використовує небінарне порівняння. Саме це допомагає даному кодуванню використовувати підтримку всіх мов. Тому, при бажанні перекладу додатка на іншу мову не виникне ніяких проблем з перекладом баз даних та таблиць.

Далі йде етап створення стовпців бази даних, де вказуються поля самих ідентифікаторів користувачів, їх логіни, ім'я та паролі. Це допомагає більш детально проводити операції з користувачами, якому саме користувачеві надати той чи інший курс. Процес ідентифікації користувачів продемонстровано на рисунку 3.2.

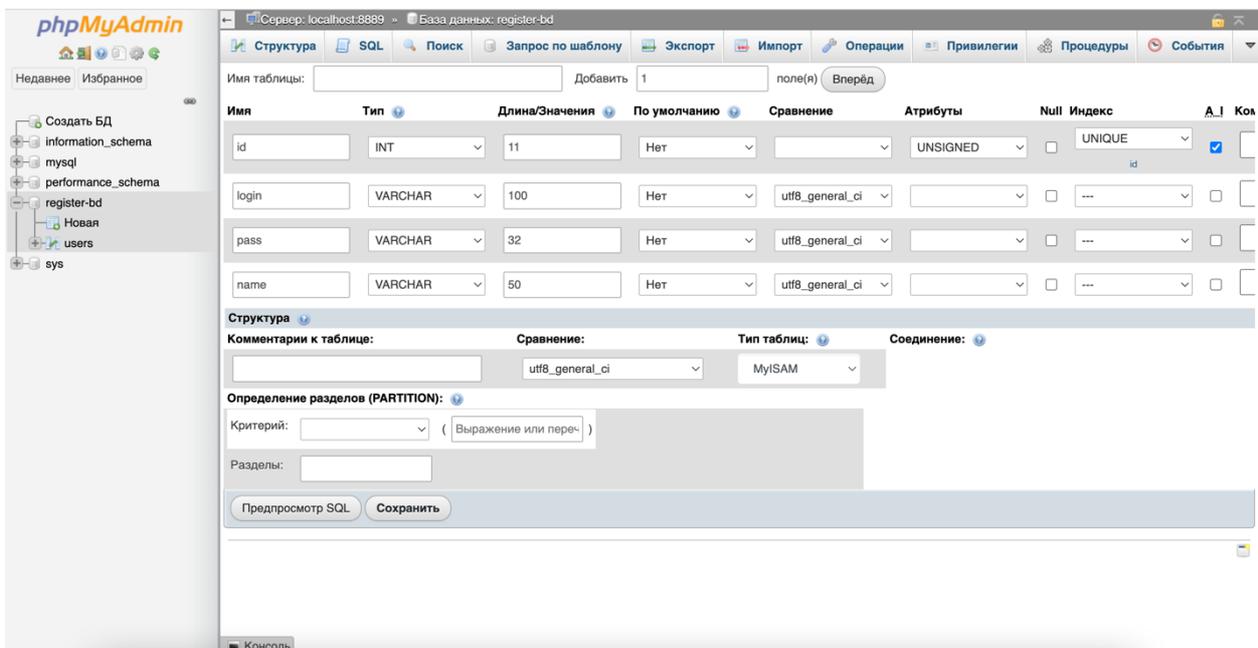


Рисунок 3.2 – Процес ідентифікації стовпців таблиці бази даних

На даному етапі було створено стовпці за допомогою яких в подальшому буде використовуватись ідентифікація. У полі ім'я потрібно вказати назву стовпця. Назву потрібно вказувати щоб вона була пов'язана з типом її використання. Наприклад назва стовпця `pass` вказує на те що в ній буде зберігатися паролі користувачів.

Далі потрібно вказати тип поля, для поля `id` було вказано тип `int`, оскільки там буде лише числа. Для інших було вказано тип `varchar`, тому як в них будуть

зберігатися не лише числа, але й букви. Це допоможе користувачам використовувати логіни та паролі з будь якими символами.

Після цього потрібно вказати максимально допустиму кількість символів для поля. Це потрібно для оптимізації додатку, оскільки якби користувачі могли б створювати логіни або паролі з нескінченною кількістю символів, це б сприяло погіршенню роботи порталу. Тому, було обрано найоптимальніші числа для коректної роботи додатку. Для поля `pass` було використано число 32, оскільки додаток використовує функцію хешування `md5`, розміром якої є саме 32 символа.

Кодування полів використовувалось так само `utf8_general_ci`, оскільки воно є найбільш коректним при роботі з даним проектом. Для поля `id` кодування не використовувалось, оскільки в ньому зберігаються лише числа.

Також для поля `id` було прописано атрибут `UNSIGNED` за допомогою якого в таблиці не будуть створюватись числа з від'ємним показником. Індекс був прописаний `UNIQUE`, він дозволяє записати лише унікальне значення користувача, завдяки якому буде можливість проводити операції лише з потрібними учнями. Стопець `A_I` (`Auto Increment`) допомагає автоматизувати додання користувачів до таблиці зберігаючи їх унікальність, тобто значення кожного наступного користувача буде збільшуватись на одиницю.

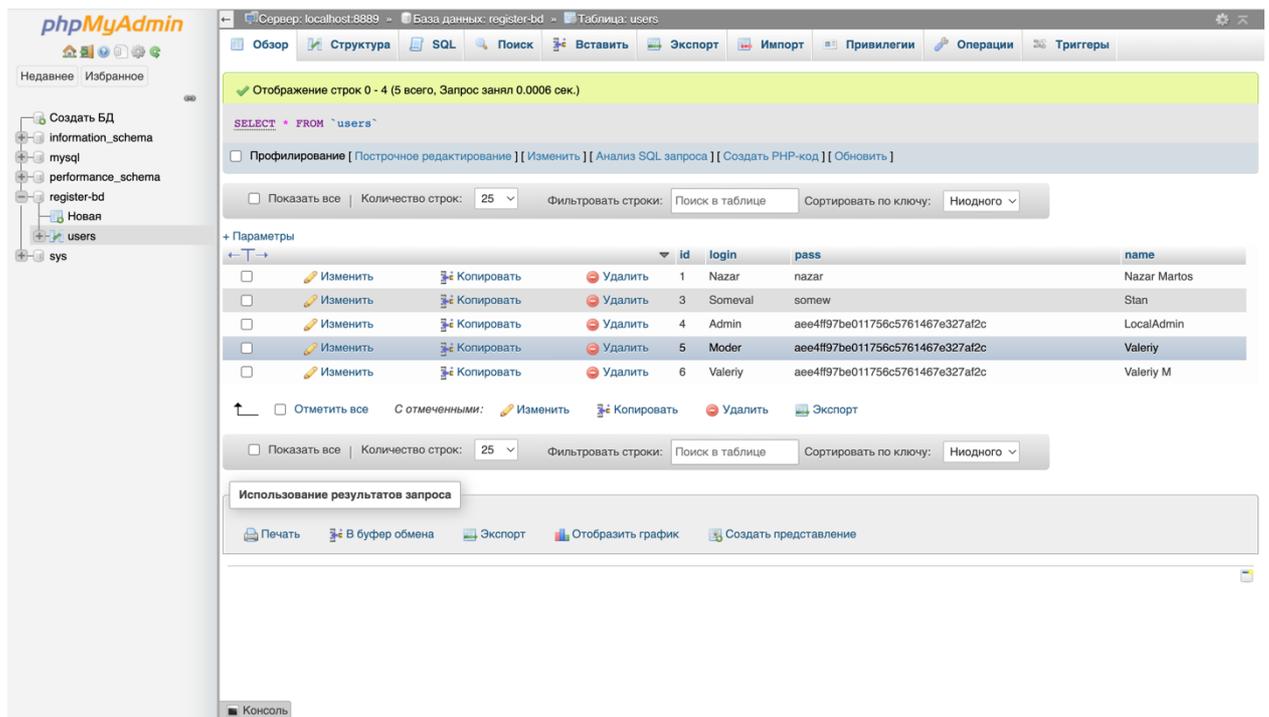


Рисунок 3.3 – Створена таблиця бази даних користувачів

На рисунку 3.3 показана реалізована таблиця бази даних користувачів, де вказані усі їх параметри. Також прослідковується результат додавання хешування до додатку у полі pass. До додання таблиця вказувала пароль користувача таким яким його вводив учень, після додання спостерігається хешування паролів.

Основна взаємодія з базами даних йде через запити мови програмування MySQL. Це є найоптимальнішим варіантом взаємодії з базами даних. На прикладі прослідковується програмування з використанням мови програмування MySQL:

```
$result = $mysql->query("SELECT * FROM `users` WHERE `login` = '$login' AND `pass` = '$pass'");
```

На даній частині коду виконано запит мовою MySQL до бази даних де відбувається вибірка з таблиці users за допомогою оператора WHERE для вибору користувачів за критеріями. При завершенні даної функції користувач зможе авторизуватись на сайті, після того як система виконає перевірку змінних login та pass, тобто логіну та паролю.

3.3 Розробка програмних блоків порталу

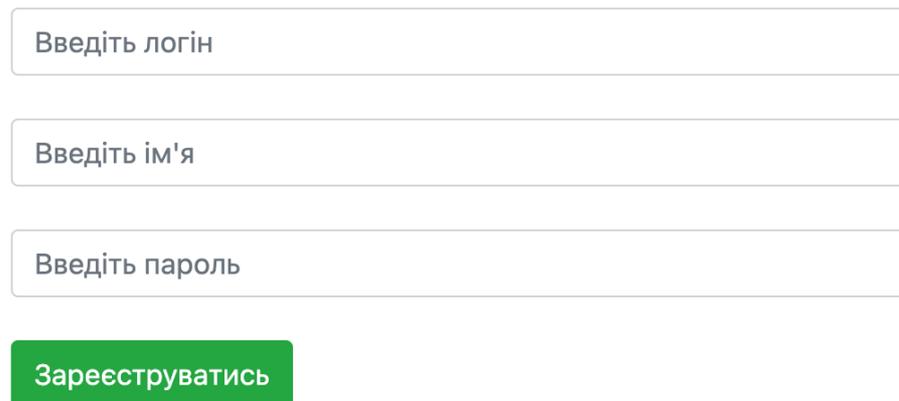
Для розробки порталу курсів для дистанційного навчання використовувався редактор коду Atom.

Atom — це безкоштовний редактор тексту та вихідного коду з відкритим вихідним кодом для macOS, Linux і Microsoft Windows з підтримкою плагінів, написаних на JavaScript, і вбудованим Git Control. Розроблений від авторів GitHub, Atom — це настільна програма, створена за допомогою веб-технологій. Більшість розширюваних пакетів мають ліцензії на безкоштовне програмне забезпечення, створені та підтримуються спільнотою. Atom заснований на Electron, фреймворку, який дозволяє використовувати міжплатформні настільні додатки з використанням Chromium і Node.js. Спочатку Atom був написаний на CoffeeScript і Less, але значна частина була перетворена на JavaScript [20].

При розробці додатку були використані такі мови програмування як: PHP, HTML, CSS та MySQL. HTML та CSS використовувались для розмітки та дизайну додатку. MySQL використовувався для роботи з базами даних.

Значна частина додатку була розроблена мовою програмування PHP. Зокрема, блок реєстрації та авторизації розроблений даною мовою програмування. Спочатку була створена лише модель форми реєстрації мовою розмітки HTML, після чого була оформлена графічна частина додатку технологією CSS, а весь функціонал програми був виконаний мовою PHP. Графічна частина блоку реєстрації без функціоналу продемонстрована на рисунку 3.4.

Реєстрація



Введіть логін

Введіть ім'я

Введіть пароль

Зареєструватись

Рисунок 3.4 – Графічна частина блоку реєстрації

Після чого був створений файл `check.php` у якому розташовувався код написаний лише мовою програмування PHP. Був застосований глобальний масив `$_POST` для отримання даних з форм. В порівнянні з масивом `$_GET`, цей масив є більш ефективним у даному випадку, але це залежить від методу який використовується. Після цього потрібно створити змінну в яку необхідно помістити даний глобальний масив, вказавши назву поля з якого будуть братись дані. Для фільтрації значень була використана функція `filter_var`, вона дозволяє відфільтрувати певні значення, в нашому випадку це `login`, `name` та `pass`. Видалити з нього певні HTML символи та символи які краще не добавляти в базу даних. Типом фільтрації було використано `FILTER_SANITIZE_STRING` для фільтрації цього поля як звичайного рядка. Для видалення лишніх пробілів з рядка була використана функція `trim`. Дана функція використовувалась для змінних `login`, `name` та `pass`. Тобто робила вибірку даних полів та елементів бази даних.

Після отримання даних полів було використано оператор `if, else` та команди `mb_strlen` для перевірки введених даних для коректної кількості введених символів. Таким чином логін користувача не може перевищувати 20 символів, пароль повинен складатись від 3 до 7 символів, а ім'я користувача не може перевищувати 50 символів. Це зроблено для оптимізації процесу введення даних у таблицю бази даних. Таким чином при вводі недопустимої кількості символів у полі додаток оповіщає про невірне введене поле (рис. 3.5).



Рис 3.5 – Оповіщення системою про недопустиму кількість символів у полі паролю

При введенні допустимої кількості символів у поля, система переводить користувача на сторінку з авторизацією. Блок авторизації працює з такими самими операторами та функціями як блок реєстрації, тобто після перевірки поля у таблиці бази даних відбувається процес входу на портал курсів для дистанційного навчання. Загальний алгоритм роботи блоку авторизації проілюстрований на рисунку 3.6.



Рисунок 3.6 – Блок-схема алгоритму роботи блоку авторизації

Також у додатку є блок додавання та проходження тестів. Тестування є невід’ємною частиною будь-якого порталу курсів для дистанційного навчання. Але не у всіх представлених аналогів є дана функція. Тим більше, що дана система базується на проходженні основного опитування, результатом якого буде представлена вибірка курсів за вподобаннями користувача. Тобто, якщо користувач при проходженні тесту обирав варіанти відповіді зв’язані з ІТ-тематикою, а також в подальшому проходженні, з веб-розробкою, то йому після проходження усього тесту будуть запропоновані декілька курсів які його можуть зацікавити. Це зроблено для користувачів які не до кінця розуміють що саме вони хочуть вивчати. З іншої сторони, якщо людина знає вектор напрямку в якому вона хоче розвиватись, вона може не проходити даний тест, а курси шукати власноруч. Графічна частина блоку додавання та проходження тестів показана на рисунку 3.7

The screenshot shows a web browser window with the URL `localhost:8080/admin.php?do=add`. The page title is "Система тестування". The main content area is titled "Додавання тесту" and contains the following form elements:

- Назва тесту:** A text input field with the value "1".
- Додавання питань:**
 - Питання #1:** A text input field.
 - Відповідь #1:** A text input field.
 - Кількість балів за відповідь #1:** A text input field.
 - Додати варіант відповіді:** A button.
 - Додати питання:** A blue button.
- Додавання результатів:**
 - Результат #1:** A text area.
 - Бал (від) #1:** A text input field.
 - Бал (до) #1:** A text input field.
 - Додати результат:** A blue button.
- Зберегти:** A green button at the bottom of the form.

Рисунок 3.7 – Графічна частина блоку додавання тестів

3.4 Висновки

У третьому розділі було розглянуто обґрунтування мови розробки додатку, роботи з базами даних та розробка програмних модулів порталу курсів для дистанційного навчання.

У підрозділі про обґрунтування мови розробки було проведено аналіз можливих мов програмування для даного додатку. Після порівняння було обрано мову програмування PHP, так як вона є найефективнішою для даного виду додатків, також в ній є можливість оптимізації коду та адаптації самого додатку для різних типів пристроїв.

У другому підрозділі розглядалась робота з базами даних, а саме з додатком PHPMyAdmin. Після аналізу переваг та можливостей даного додатку було вирішено продовжувати працювати саме з ним. Також, розглядалась робота з базами даних та їх таблицями. Створення та обґрунтування стовпців таблиці для роботи самого додатку було основною задачею роботи з цим сервісом.

Третій підрозділ вміщує в себе інформацію про розробку програмних блоків порталу курсів для дистанційного навчання. Для прикладу розробки були обрані такі модулі як: модуль реєстрації та авторизації, а також модуль створення та проходження тестування.

У даному розділі були продемонстровані зображення баз даних, графічна частина форм авторизації, реєстрації та тестування, а також блок-схема роботи алгоритму авторизації.

4 ТЕСТУВАННЯ ПРОГРАМНОЇ ЧАСТИНИ ПОРТАЛУ КУРСІВ ДЛЯ ДИСТАНЦІЙНОГО НАВЧАННЯ

4.1 Тестування програмного модуля авторизації та реєстрації

Тестування програмної частини додатку є невід’ємним чинником під час впровадження в експлуатацію та при повсякденній роботі з даним додатком.

Розглянемо випадок неправильно введених даних у поля форми реєстрації (рис 4.1).

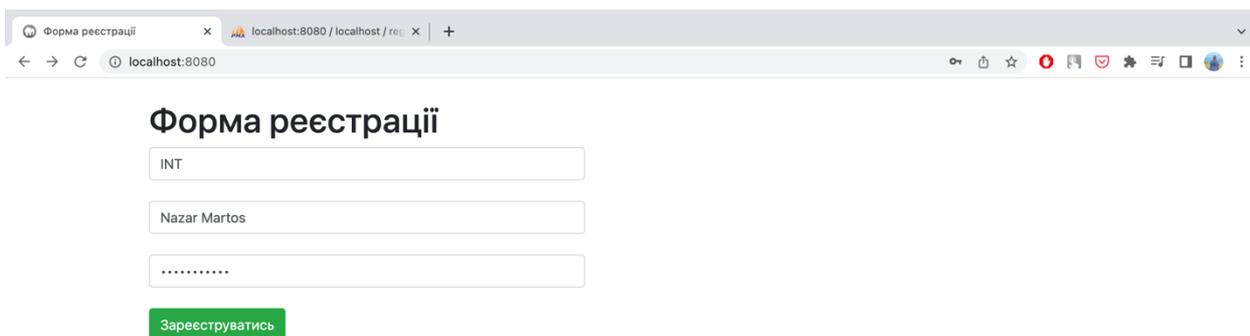


Рисунок 4.1 – Тестування роботи програмного блоку реєстрації у випадку невірно введених даних для поля логіну

У поле «Логін» була введена невірна кількість символів для реєстрації користувача. Після чого на екрані з’явилось сповіщення про помилку введення (рис 4.2)

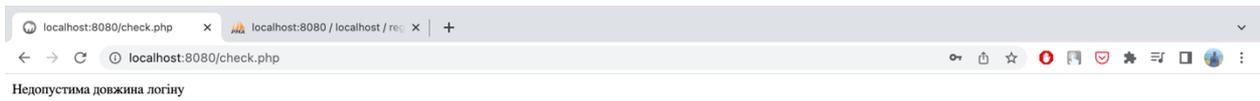


Рисунок 4.2 – Сповідження про помилку процесу реєстрації

Розглянемо програмний код програмної частини, що відповідає за тестування полів даної форми. Цей програмний код написаний на мові PHP та представляє собою наступну частину коду:

```
<?php
$login = filter_var(trim($_POST ['login']), FILTER_SANITIZE_STRING);
$name = filter_var(trim($_POST ['name']), FILTER_SANITIZE_STRING);
$pass = filter_var(trim($_POST ['pass']), FILTER_SANITIZE_STRING);

if(mb_strlen($login) < 5 || mb_strlen($login) > 90) {
    echo "Недопустима кількість символів логіну";
    exit();
} else if(mb_strlen($name) < 3 || mb_strlen($name) > 50) {
    echo "Недопустима кількість символів ім'я";
    exit();
}
```

```

} else if(mb_strlen($pass) < 2 || mb_strlen($pass) > 6) {
    echo "Недопустима кількість символів паролю";
    exit();
}

$pass = md5($pass."asdfghj12795");

$mysql = new mysqli('localhost', 'root', 'root', 'register-bd');
$mysql ->query("INSERT INTO `users` (`login`, `pass`, `name`) VALUES ('$login',
'$pass', '$name')");

$mysql ->close();

header('Location: /');
?>

```

В ній ми написали три змінних, які являють собою логін, пароль та ім'я користувача. Після цього ми використали оператор `if` і при невиконанні умов даного оператора наш додаток видає помилку. При цьому, програмний код дає запит на базу даних та якщо усі умови виконані, то програма додає інформацію про користувача до таблиці бази даних.

Після реєстрації на сайті, йде процес авторизації. Він представлений у додатку наступний чином:

```

<?php
$login = filter_var(trim($_POST ['login']), FILTER_SANITIZE_STRING);
$pass = filter_var(trim($_POST ['pass']), FILTER_SANITIZE_STRING);

$pass = md5($pass."asdfghj12795");

```

```

$mysql = new mysqli('localhost', 'root', 'root', 'register-bd');

$result = $mysql->query("SELECT * FROM `users` WHERE `login` = '$login' AND
`pass` = '$pass'");

$user = $result->fetch_assoc();
if(count($user) == 0) {
    echo "Такого користувача не знайдено";
    exit();
}

setcookie('user', $user['name'], time() + 3600, "/");
$mysql ->close();

header('Location: /');
?>

```

У даній частині коду так само йде запит на таблицю бази даних, тільки в даному випадку ми не додаємо інформацію про користувача, а перевіряємо чи присутня в таблиці інформація про того чи іншого користувача. Якщо такої інформації не знайдено, то програма буде видавати сторінку з наступним текстом: «Такого користувача не знайдено».

4.2 Тестування програмного модуля створення та проходження тестів

Тестування модуля проходження тестів призначено для перевірки результатів знань користувачів та визначення вектору розвитку певного користувача після проходження тесту під назвою «Кваліфікаційний тест». У ньому буде визначено яку саме вибірку курсів призначати тому чи іншому користувачеві. Форма додання тестів показана на рисунку 4.3.

Система тестування

localhost:8080 / localhost / te

localhost:8080/admin.php?do=add

Додавання тесту

Назва тесту

Додавання питань

Питання #1

Відповідь #1

Кількість балів за відповідь #1

Додати варіант відповіді

Додати питання

Додавання результатів

Результат #1

Бал (від) #1

Бал (до) #1

Додати результат

Зберегти

Рисунок 4.3 – Форма додавання тестів у додаток

При заповненні полів необхідно вказати декілька варіантів відповідей, а також кількість балів яка буде нараховуватись при правильній відповіді. І на цій самій сторінці потрібно вказати який результат буде виводитись користувачам при набраних балах.

Після заповнення усіх необхідних полів, потрібно натисну на кнопку «Зберегти», після чого додаток відкриває список тестів. (рис 4.4)

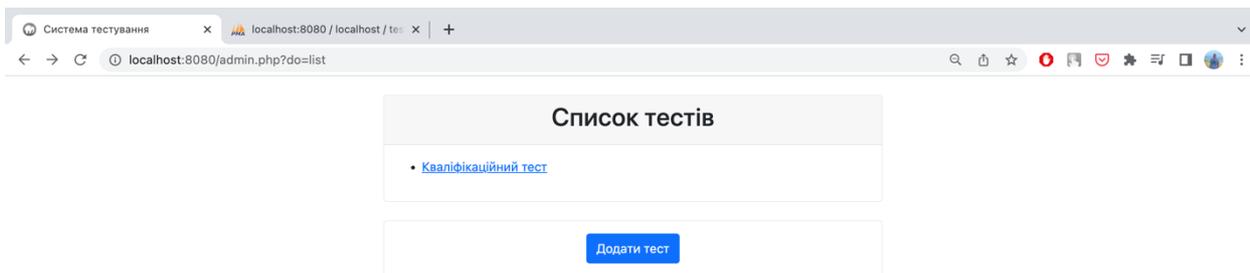


Рисунок 4.5 – Список тестів у додатку

Після створення та додання тесту дані про нього заносяться до таблиці бази даних під назвою «tests». Завершивши процес створення тесту ми можемо його пройти. Форма проходження тестування представлена на рисунку 4.6.

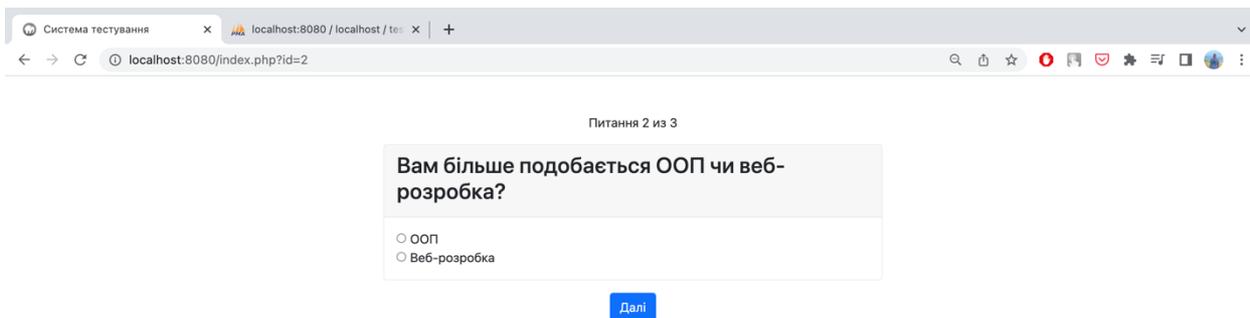


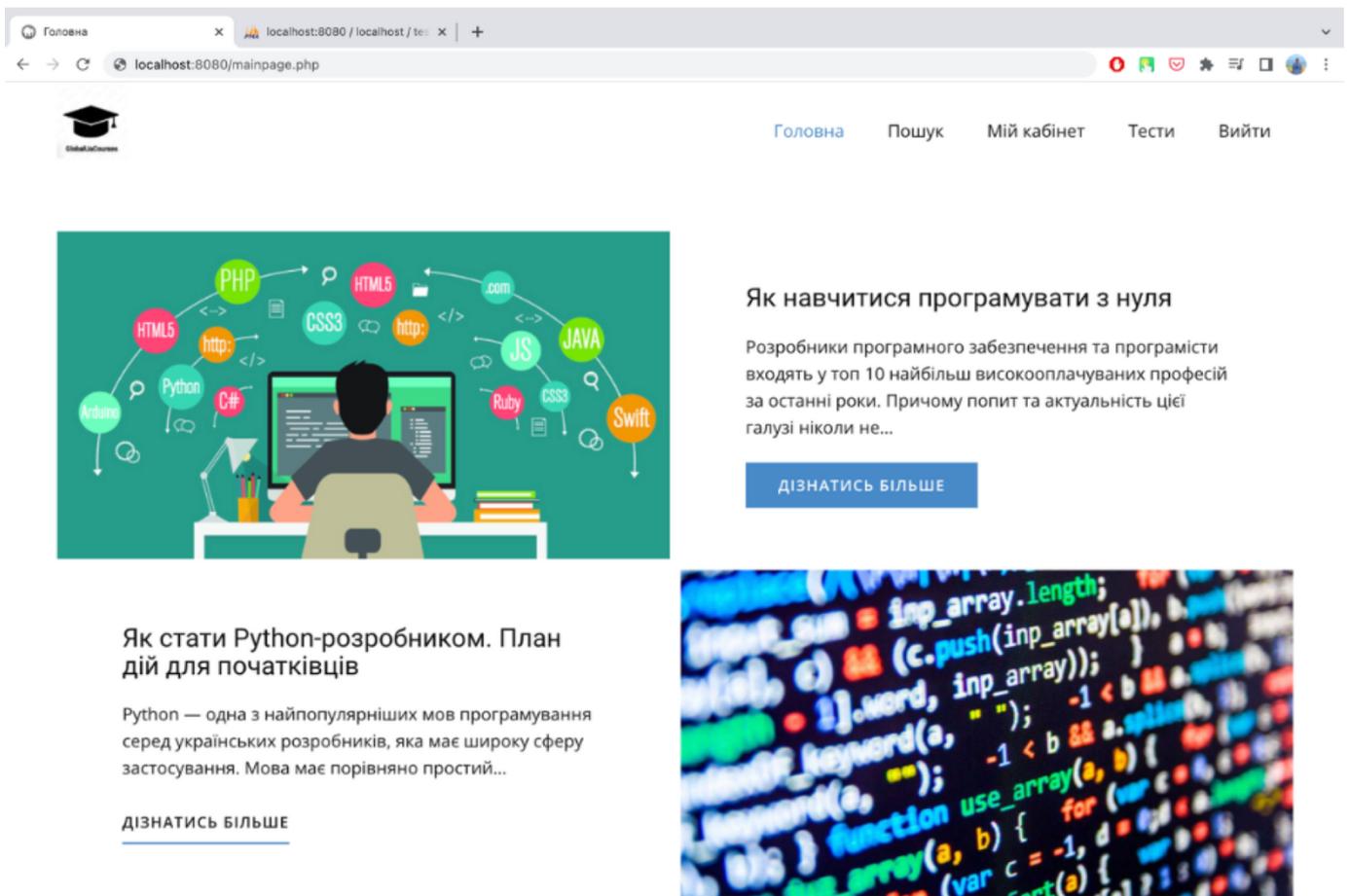
Рисунок 4.6 – Форма проходження тестування

Завершивши проходження тесту, додаток повідомляє користувачів про результати тестування. Залежно від кількості балів, системо розподілить користувачів за векторами їх подальшого розвитку.



Рисунок 4.6 – Результати тестування

Таким чином, проходить процес тестування модуля створення та проходження тестів. Після завершення проходження тестування системи, користувач може перейти на головну сторінку додатку, що представлена на рисунку 4.7.



Головна | Пошук | Мій кабінет | Тести | Вийти

Як навчитися програмувати з нуля

Розробники програмного забезпечення та програмісти входять у топ 10 найбільш високооплачуваних професій за останні роки. Причому попит та актуальність цієї галузі ніколи не...

[ДІЗНАТИСЬ БІЛЬШЕ](#)

Як стати Python-розробником. План дій для початківців

Python — одна з найпопулярніших мов програмування серед українських розробників, яка має широкую сферу застосування. Мова має порівняно простий...

[ДІЗНАТИСЬ БІЛЬШЕ](#)

```

inp_array = inp_array.length;
inp_array.push(inp_array[a]);
}
keyword(a, " ");
keyword(a, " ");
function use_array(a, b) {
  for (var c = -1; c < a; c++) {
    var d = inp_array[c];
    var e = inp_array[c];
  }
}

```

Рисунок 4.7 – Головна сторінка додатку

Також користувач може перейти на сторінку свого кабінету, де йому будуть запропоновані курси, що були підібрані в залежності від проходження кваліфікаційного тесту. Сторінка кабінету користувача продемонстрована на рисунку 4.8.

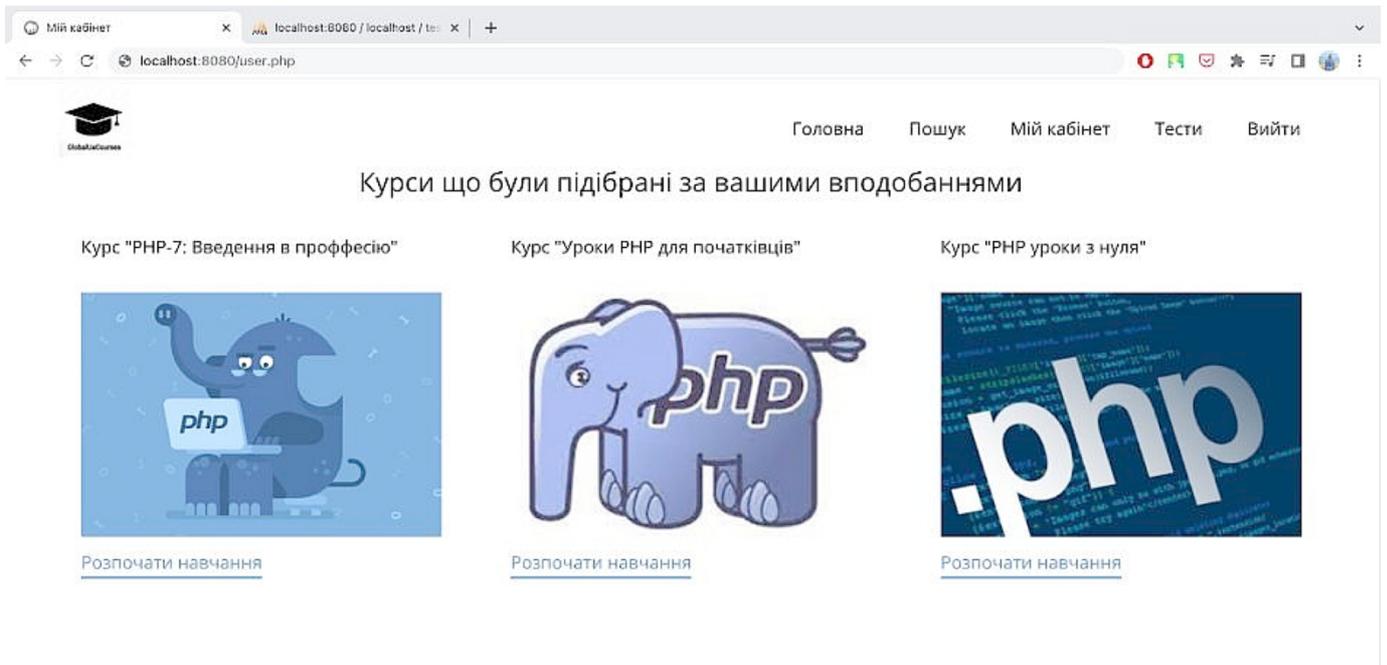


Рисунок 4.8 – Сторінка кабінету користувача

Після завершення усіх результатів користувач може вийти з додатку, натиснувши кнопку «Вихід». Вихід з системи реалізовано файлом `exit.php`.

4.3 Висновки

Цей розділ призначений для тестування програмного блоку порталу курсів для дистанційного навчання. У першому модулі було розглянуто можливість авторизації та реєстрації користувачів. Була розписана послідовна розробка даного блоку, описані оператори та функції, що були застосовані. Також було розказано про взаємодію з базами даних.

У другому модулі тестування полягало у створенні та проходженні тестів, що були розроблені. В цьому випадку, було ретельно описано та розказано про сам процес створення та проходження тестів. Також, було продемонстровано вигляд головної сторінки додатку та сторінки кабінету користувача.

ВИСНОВКИ

В роботі виконано аналіз стану задачі, де основним завданням є розробка порталу курсів для дистанційного навчання. Підкреслена важливість щодо здатності ефективної праці самого веб-додатку та створення інформаційно-навчальної системи для збільшення продуктивності при роботі з ним.

В першому розділі було зроблено порівняльний аналіз аналогів які являли собою поширені онлайн сервіси. В цьому випадку, були розглянуті такі важливі параметри функціонування порталів курсів для дистанційного навчання як функціональність сайтів, надійність у наданні тестування для перевірки знань, кількість контенту, зручність використання для студентів, захищеність про роботі з користувачами.

У постановці задачі було визначено основні завдання які необхідно вирішити для розробки порталу що впроваджується за допомогою розробки інформаційно-навчальної системи.

В другому розділі визначені основні вимоги щодо проектування порталу курсів для дистанційного навчання. Була розроблена модель інформаційно-навчальної системи, а також підмоделі самої системи.

В третьому розділі було обґрунтовано вибір мови розробки додатку, проведено аналіз можливих мов програмування, а також вибір середовища редактору коду. Було розроблено основні програмні модулі додатку.

В четвертому розділі було проведено тестування програмних модулів порталу курсів та представлено головну сторінку додатку та сторінку кабінету користувача.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Коронавірус та освіта: аналіз проблем та наслідків пандемії: веб-сайт. URL: <https://cedos.org.ua/researches/koronavirus-ta-osvita-analiz-problem-i-naslidkiv-pandemii/> (дата звернення: 28.05.2022).
2. Дистанційне навчання: веб-сайт. URL: <https://mon.gov.ua/ua/osvita/pozashkilna-osvita/distancijne-navchannya>.
3. Мартос Н.М. Особливості веб-програмування у сфері дистанційного навчання. Матеріали Всеукраїнської конференції: Молодь в науці: дослідження, проблеми, перспективи: веб-сайт. URL: <https://conferences.vntu.edu.ua/index.php/all-fitki/all-fitki-2022/paper/view/15540>.
4. Особливості впровадження дистанційного навчання в освітній процес закладу вищої освіти: веб-сайт. URL: http://pedagogy-journal.kpu.zp.ua/archive/2021/75/part_3/20.pdf/.
5. Портал з курсами Prometheus: веб-сайт. URL: <https://prometheus.org.ua/>.
6. Портал з курсами WiseCow: веб-сайт. URL: <https://wisecow.com.ua/>.
7. Відкритий Університет Майдану (ВУМ): веб-сайт. URL: <https://vumonline.ua/>.
8. Портал з курсами EducationalEra: веб-сайт. URL: <https://www.ed-era.com/>.
9. ІТ-відділи в компаніях: веб-сайт. URL: <https://biz.nv.ua/ukr/experts/popit-na-programistiv-yak-vtrimati-specialista-v-kompaniji-ostanni-novini-50206416.html>.
10. How to book: Get your website noticed: навч. посіб. / Motaus F.– Bluebird, 2020. – 180 ст.
11. Оптимізація для пошукових систем: веб-сайт. URL: <https://lanet.click/zovnishnya-optimizaciya-saitiv/>.
12. Дистанційне та змішане навчання у закладах освіти. Путівник.: веб-сайт. – URL: https://www.researchgate.net/ Distancijne_ta_zmisane_navcanna_v_skoli_Putivnik.

13. 25 моделей MBA NEED-TO-KNOW. / Джуліан Біркіншоу – Олімп-Бізнес, 2020. – 147 ст.
14. Моделі та методи прийняття рішень: веб-сайт. URL: [https://sau.nmu.org.ua/ua/osvita/metod/Models_and_decision-making_techniques\(Us_Koryashkina\)_NMU_SAU.pdf](https://sau.nmu.org.ua/ua/osvita/metod/Models_and_decision-making_techniques(Us_Koryashkina)_NMU_SAU.pdf).
15. Посібник з мови програмування C++: веб-сайт. URL: <https://metanit.com/cpp/tutorial/>.
16. JavaScript: The Definitive Guide/ David Flanagan, 2021. – 270 ст.
17. Programming PHP: Creating Dynamic Web Pages 4th Edition / Kevin Tatroe, Peter MacIntyre, O'Reilly Media, 2020. – 256 ст.
18. Що таке PhPMyAdmin?: веб-сайт. URL: <https://freehost.com.ua/faq/wiki/chto-takoe-phpmyadmin/>.
19. MySQL та PhPMyAdmin: включення серверу та доступ до його адміністрування: веб-сайт. URL: <https://www.aitishnik.com/web/ampinst/ustanovka-phpmyadmin.html>.
20. Редактор коду Atom: веб-сайт. URL: <https://atom.io/>.

ДОДАТКИ

Додаток А – Технічне завдання

Міністерство освіти і науки України
Вінницький національний технічний університет
Факультет інформаційних технологій та комп'ютерної інженерії

ЗАТВЕРДЖУЮ
д.т.н., проф. О. Н. Романюк
31 березня 2022 р.

Технічне завдання
на бакалаврську дипломну роботу «Створення порталу курсів для
дистанційного навчання»»
за спеціальністю
121 – Інженерія програмного забезпечення

Керівник бакалаврської дипломної роботи:

_____ к.т.н., доц. О.О. Коваленко

" 31 " березня _____ 2022 р.

Виконав:

_____ студент гр. 1ПІ-186 Н.М. Мартос

" 31 " березня _____ 2022 р.

Вінниця – 2022 року

1. Найменування та галузь застосування

Бакалаврська дипломна робота: «Створення порталу курсів для дистанційного навчання».

Галузь застосування – дистанційне навчання.

2. Підстава для розробки.

Завдання на роботу, яке затверджене на засіданні кафедри програмного забезпечення – протокол № 12 від «07» лютого 2022 р.

3. Мета та призначення розробки.

Метою бакалаврської дипломної роботи є підвищення тренувальних можливостей логіки прийняття рішень та реакції користувача за допомогою розробки і використання 2d-гри у жанрі екшн платформера, що дозволяє проводити тренування в ігровій формі.

Призначення роботи – створення порталу курсів для дистанційного навчання.

4. Вихідні дані для проведення НДР

1. Programming PHP: Creating Dynamic Web Pages 4th Edition / Kevin Tatroe, Peter MacIntyre, O'Reilly Media, 2020. – 256 ст
2. JavaScript: The Definitive Guide/ David Flanagan, 2021. – 270 ст.
3. How to book: Get your website noticed: навч. посіб. / Motaus F.– Bluebird, 2020. – 180 ст.
4. 25 моделей MBA NEED-TO-KNOW. / Джуліан Біркіншоу – Олімп-Бізнес, 2020. – 147 ст.

5. Технічні вимоги

Вхідні дані – особисті дані користувача, проходження курсу та тестування;
Вихідні дані – пройдений курс та тестування.

6. Конструктивні вимоги.

Конструкція пристрою повинна відповідати естетичним та ергономічним вимогам, повинна бути зручною в обслуговуванні та керуванні.

Графічна та текстова документація повинна відповідати діючим стандартам України.

7. Перелік технічної документації, що пред'являється по закінченню робіт:

- пояснювальна записка до бакалаврської дипломної роботи;
- технічне завдання;
- лістинги програми.

8. Вимоги до рівня уніфікації та стандартизації

При розробці програмних засобів слід дотримуватися уніфікації і ДСТУ.

Додаток Б – Протокол перевірки на плагіат

ПРОТОКОЛ ПЕРЕВІРКИ БАКАЛАВРСЬКОЇ ДИПЛОМНОЇ РОБОТИ НА НАЯВНІСТЬ ТЕКСТОВИХ ЗАПОЗИЧЕНЬ

Назва роботи: Створення порталу курсів для дистанційного навчання

Тип роботи: БДР

Підрозділ : кафедра програмного забезпечення, ФІТКІ

Науковий керівник: к.т.н., доц. каф. ПЗ Коваленко О.О.

Оригінальність	87,2%
Схожість	12,8%

Аналіз звіту подібності

■ **Запозичення, виявлені у роботі, оформлені коректно і не містять ознак плагіату.**

- Виявлені у роботі запозичення не мають ознак плагіату, але їх надмірна кількість викликає сумніви щодо цінності роботи і відсутності самостійності її автора. Роботу направити на доопрацювання.
- Виявлені у роботі запозичення є недобросовісними і мають ознаки плагіату та/або в ній містяться навмисні спотворення тексту, що вказують на спроби приховування недобросовісних запозичень.

Особа, відповідальна за перевірку _____ Черноволик Г. О.

Ознайомлені з повним звітом подібності, який був згенерований системою Unichesk

Автор роботи _____ Мартос Назар Максимович

Керівник роботи _____ Коваленко Олена Олексіївна

Додаток В – Лістинг програми

validationForm.php

```
<!DOCTYPE html>

<html lang="ua" dir="ltr">

  <head>

    <meta charset="utf-8">

    <meta name="viewport" content="width=device-width, initial-
scale=1.0">

    <meta http-equiv="X-UA-Compatible" content="ie=edge">

    <title>Форма реєстрації</title>

    <link rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/bootstrap@4.6.1/dist/css/bootstrap.min.cs
s" integrity="sha384-
zCbKRCUGaJDkqS1kPbPd7TveP5iyJE0EjAuZQTgFLD2ylzuqKfdKlfG/e
SrtxUkn" crossorigin="anonymous">

    <link rel="stylesheet" href="/css/style.css">

  </head>

  <body>

    <div class="container mt-4">

      <?php

        if($_COOKIE ['user'] == ""):

          ?>
```

```

<div class="row">
  <div class="col">
    <h1>Реєстрація</h1>
    <form action="validation-form/check.php" method="post">
      <input type="text" class="form-control" name="login" id="login"
placeholder="Введіть логін"><br>
      <input type="text" class="form-control" name="name"
id="name" placeholder="Введіть ім'я"><br>
      <input type="password" class="form-control" name="pass"
id="pass" placeholder="Введіть пароль"><br>
      <button class="btn btn-success"
type="submit">Зареєструватись</button>
    </form>
  </div>
  <div class="col">
    <h1>Авторизація</h1>
    <form action="validation-form/auth.php" method="post">
      <input type="text" class="form-control" name="login" id="login"
placeholder="Введіть логін"><br>
      <input type="password" class="form-control" name="pass"
id="pass" placeholder="Введіть пароль"><br>
      <button class="btn btn-success"

```

```

type="submit">Авторизуватись</button>
    </form>
</div>
</div>
<?php else:?>
    <p>Привіт, <?=$_COOKIE['user']?>, виконується вхід на портал.
Щоб скасувати вхід нажміть <a href="/exit.php">тут</a>.</p>
<?php endif;?>

</div>
</body>
</html>

```

Check.php

```

<?php
    $login = filter_var(trim($_POST ['login']),
FILTER_SANITIZE_STRING);
    $name = filter_var(trim($_POST ['name']),
FILTER_SANITIZE_STRING);
    $pass = filter_var(trim($_POST ['pass']),
FILTER_SANITIZE_STRING);

    if(mb_strlen($login) < 5 || mb_strlen($login) > 90) {

```

```
echo "Недопустима кількість символів логіну";  
exit();  
} else if(mb_strlen($name) < 3 || mb_strlen($name) > 50) {  
    echo "Недопустима кількість символів ім'я";  
    exit();  
} else if(mb_strlen($pass) < 2 || mb_strlen($pass) > 6) {  
    echo "Недопустима кількість символів паролю";  
    exit();  
}  
  
$pass = md5($pass."asdfghj12795");  
  
$mysql = new mysqli('localhost', 'root', 'root', 'register-bd');  
$mysql ->query("INSERT INTO `users` (`login`, `pass`, `name`)  
VALUES ('$login', '$pass', '$name')");  
  
$mysql ->close();  
  
header('Location: /');  
  
?>
```

auth.php

```
<?php

    $login = filter_var(trim($_POST ['login']),
FILTER_SANITIZE_STRING);

    $pass = filter_var(trim($_POST ['pass']),
FILTER_SANITIZE_STRING);

    $pass = md5($pass."asdfghj12795");

    $mysql = new mysqli('localhost', 'root', 'root', 'register-bd');

    $result = $mysql->query("SELECT * FROM `users` WHERE `login` =
'$login' AND `pass` = '$pass'");

    $user = $result->fetch_assoc();

    if(count($user) == 0) {
        echo "Такого користувача не знайдено";
        exit();
    }

    setcookie('user', $user['name'], time() + 3600, "/");

    $mysql ->close();
```



```
header('Location: /');
```

```
?>
```

```
testSystemForm.php
```

```
<?php
```

```
include_once 'db.php';
```

```
session_start();
```

```
$id = (int) $_GET['id'];
```

```
if ($id < 1) {
```

```
    header ('location: admin.php');
```

```
}
```

```
$testId = $id;
```

```
if (!isset($_SESSION['test_id']) || $_SESSION['test_id'] != $testId) {
```

```
    $_SESSION['test_id'] = $testId;
```

```
    $_SESSION['test_score'] = 0;
```

```
}
```

```
$res = $db->query("SELECT * FROM tests WHERE id = {$testId}");
```

```
$row = $res->fetch();
```

```
$testTitle = $row['title'];

$questionNum = (int) $_POST['q'];
if (empty($questionNum)) {
    $questionNum = 0;
}
$questionNum++;
$questionStart = $questionNum - 1;

$res = $db->query("SELECT count(*) AS count FROM questions
WHERE test_id = {$testId}");
$row = $res->fetch();
$questionCount = $row['count'];

$answerId = (int) $_POST['answer_id'];
if (!empty($answerId)) {
    $res = $db->query("SELECT * FROM answers WHERE id =
{$answerId}");
    $row = $res->fetch();
    $score = $row['score'];
    $_SESSION['test_score'] += $score;
}
```

```
$showForm = 0;

if ($questionCount >= $questionNum) {

    $showForm = 1;

    $res = $db->query("SELECT * FROM questions WHERE test_id =
{$testId} LIMIT {$questionStart}, 1");

    $row = $res->fetch();

    $question = $row['question'];

    $questionId = $row['id'];

    $res = $db->query("SELECT * FROM answers WHERE
question_id = {$questionId}");

    $answers = $res->fetchAll();

} else {

    $score = $_SESSION['test_score'];

    $res = $db->query("SELECT * FROM results WHERE test_id =
{$testId} AND score_min <= {$score} AND score_max >= {$score}");

    $row = $res->fetch();

    $result = $row['result'];

}
```

```
?>
<!doctype html>
<html lang="ru">
<head>
  <meta charset="UTF-8">
  <meta name="viewport"
    content="width=device-width, user-scalable=no, initial-scale=1.0,
maximum-scale=1.0, minimum-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Система тестування</title>
  <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.cs
s" rel="stylesheet" integrity="sha384-
EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65Vohhpuu
COmLASjC" crossorigin="anonymous">
  <link rel="stylesheet" href="css/app.css">
</head>
<body>
  <div class="container">
    <?php if ($showForm) { ?>
      <form action="index.php?id=<?php echo $testId; ?>"
```

```

method="post">
    <input type="hidden" name="q" value="<?php echo
    $questionNum; ?>">

    <div class="row justify-content-center">
        <div class="col-md-6">
            <div class="text-center mt-5">
                <p>Питання <?php echo $questionNum . ' из ' .
                $questionCount; ?></p>
            </div>
            <div class="card mt-3">
                <div class="card-header">
                    <h3><?php echo $question; ?></h3>
                </div>
                <div class="card-body">
                    <?php foreach ($answers AS $answer) { ?>
                        <div>
                            <input type="radio" name="answer_id" required
                            value="<?php echo $answer['id']; ?>" <?php echo $answer['answer']; ?>
                        </div>
                    <?php } ?>
                </div>
            </div>
        </div>
    </div>

```

```

</div>

<div class="text-center mt-3">
    <?php if ($questionCount == $questionNum) { ?>
        <button type="submit" class="btn btn-
success">Отримати результат</button>
    <?php } else { ?>
        <button type="submit" class="btn btn-
primary">Далі</button>
    <?php } ?>
</div>

</div>

</div>

</div>

</form>

<?php } else { ?>
<div class="row justify-content-center">
    <div class="col-md-6">
        <div class="card mt-3">
            <div class="card-header">
                <h3>Ваш результат</h3>
            </div>
            <div class="card-body">
                <div class="result-print">

```

```

        <?php echo $result; ?>
    </div>
</div>
</div>
</div>
</div>
<?php } ?>
</div>

```

```

<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.
min.js" integrity="sha384-
MrcW6ZMFY1zcLA8Nl+NtUVF0sA7MsXsP1UyJoMp4YLEuNSfAP+JcX
n/tWtIaxVXM" crossorigin="anonymous"></script>
</body>
</html>

```

db.php

```
<?php
```

```
$config = [
```

```

'host' => 'localhost',
'name' => 'testsystem',
'user' => 'root',
'password' => 'root',
'charset' => 'utf-8',
];

```

```

$db = new PDO ('mysql:host=' . $config['host'] . ';dbname=' .
$config['name'], $config['user'], $config['password']);
$db->query('SET character_set_connection = ' . $config['charset'] . ');');
$db->query('SET character_set_client = ' . $config['charset'] . ');');
$db->query('SET character_set_results = ' . $config['charset'] . ');');

```

App.js

```

let questionNum = 1;
let resultNum = 1;

$(document).on('click', '.addAnswer', function() {
    let question = $(this).data('question');
    let answer = $(this).data('answer');
    let answerBlock = $(this).parents('.answers').find('.answer-items');

```



```

answer++;

$(this).data('answer', answer);

answerBlock.append(`<div class="divider">
    <label for="answer_text_${question}_${answer}"
class="form-label">Відповідь #${answer}</label>
    <input type="text"
name="answer_text_${question}_${answer}"
id="answer_text_${question}_${answer}" class="form-control">
</div>
<div class="mt-2">
    <label for="answer_score_${question}_${answer}"
class="form-label">Кількість балів за відповідь #${answer}</label>
    <input type="text"
name="answer_score_${question}_${answer}"
id="answer_score_${question}_${answer}" class="form-control">
</div>`);
});

$('.addQuestion').on('click', function() {
    questionNum++;
    let questionBlock = $('.question-items');

```

```

questionBlock.append(`
    <div class="mt-4">
        <label for="question_${questionNum}" class="form-
label">Питання #${questionNum}</label>
        <input type="text" name="question_${questionNum}"
id="question_${questionNum}" class="form-control">
        <div class="answers">
            <div class="answer-items">
            </div>
            <div class="text-center mt-4">
                <button type="button" class="btn btn-light border
addAnswer" data-question="${questionNum}" data-answer="0">Додати
варіант відповіді</button>
            </div>
        </div>
    </div>`);
});
$(document).on('click', '.addResult', function() {
    resultNum++;
    let resultBlock = $('<div class="result-items">');

    resultBlock.append(`

```

```

<div class="mt-4 divider">
  <div class="">
    <label for="result_${resultNum}" class="form-
label">Результат #${resultNum}</label>
    <textarea name="result_${resultNum}" id="result_1"
class="form-control"></textarea>
  </div>
  <div class="mt-2">
    <label for="result_score_min_${resultNum}"
class="form-label">Бал (від) #${resultNum}</label>
    <input type="text"
name="result_score_min_${resultNum}"
id="result_score_min_${resultNum}" class="form-control">
  </div>
  <div class="mt-2">
    <label for="result_score_max_${resultNum}"
class="form-label">Бал (до) #${resultNum}</label>
    <input type="text"
name="result_score_max_${resultNum}"
id="result_score_max_${resultNum}" class="form-control">
  </div>
</div>`);

```

```
});
```

Admin.php

```
<?php
    include_once 'db.php';

    $do = trim(strip_tags($_GET['do']));
    if ($do == 'save') {
        $title = trim($_POST['title']);

        $res = $db->prepare("INSERT IGNORE INTO tests (`title`)
VALUES (:title)");
        $res->execute([
            ':title' => $title,
        ]);
        $testId = $db->lastInsertId();

        $questionNum = 1;
        while (isset($_POST['question_' . $questionNum])) {
            $question = trim($_POST['question_' . $questionNum]);
            if (empty($question)) {
```

```

        continue;
    }

    $res = $db->prepare("INSERT IGNORE INTO questions
(`test_id`, `question`) VALUES (:test_id, :question)");
    $res->execute([
        ':test_id' => $testId,
        ':question' => $question,
    ]);
    $questionId = $db->lastInsertId();

    $answerNum = 1;
    while (isset($_POST['answer_text_' . $questionNum . '_' .
$answerNum])) {
        $answer = trim($_POST['answer_text_' . $questionNum . '_' .
$answerNum]);
        $score = trim($_POST['answer_score_' . $questionNum . '_' .
$answerNum]);
        if (empty($answer)) {
            continue;
        }
    }

```

```

        $res = $db->prepare("INSERT IGNORE INTO answers
(`question_id`, `answer`, `score`)
        VALUES (:question_id, :answer, :score)");
        $res->execute([
            ':question_id' => $questionId,
            ':answer' => $answer,
            ':score' => $score,
        ]);

        $answerNum++;
    }
    $questionNum++;
}

$resultNum = 1;
while (isset($_POST['result_' . $resultNum])) {
    $result = trim($_POST['result_' . $resultNum]);
    $scoreMin = trim($_POST['result_score_min_' . $resultNum]);
    $scoreMax = trim($_POST['result_score_max_' . $resultNum]);

    $res = $db->prepare("INSERT IGNORE INTO results (`test_id`,
`score_min`, `score_max`, `result`)

```

```
VALUES (:test_id, :score_min, :score_max,
:result)");
    $res->execute([
        ':test_id' => $testId,
        ':score_min' => $scoreMin,
        ':score_max' => $scoreMax,
        ':result' => $result,
    ]);

    $resultNum++;
}

header ('Location: admin.php?do=list');
}

if ($do != 'add') {
    $do = 'list';
}
?>

<!doctype html>
```

```
<html lang="ru">
<head>
  <meta charset="UTF-8">
  <meta name="viewport"
    content="width=device-width, user-scalable=no, initial-scale=1.0,
maximum-scale=1.0, minimum-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Система тестування</title>
  <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.cs
s" rel="stylesheet" integrity="sha384-
EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65Vohhpuu
COmLASjC" crossorigin="anonymous">
  <link rel="stylesheet" href="css/app.css">
</head>
<body>

  <div class="container">
    <div class="row justify-content-center">

      <?php include _once 'inc/' . $do . '.php'; ?>
```



```
</div>
```

```
</div>
```

```
<script
```

```
src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js" integrity="sha384-
```

```
MrcW6ZMFY1zcLA8Nl+NtUVF0sA7MsXsP1UyJoMp4YLEuNSfAP+JcXn/tWtIaxVXM" crossorigin="anonymous"></script>
```

```
<script
```

```
src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.4/jquery.min.js"></script>
```

```
<script src="js/app.js"></script>
```

```
</body>
```

```
</html>
```

Додаток Г – Графічна частина

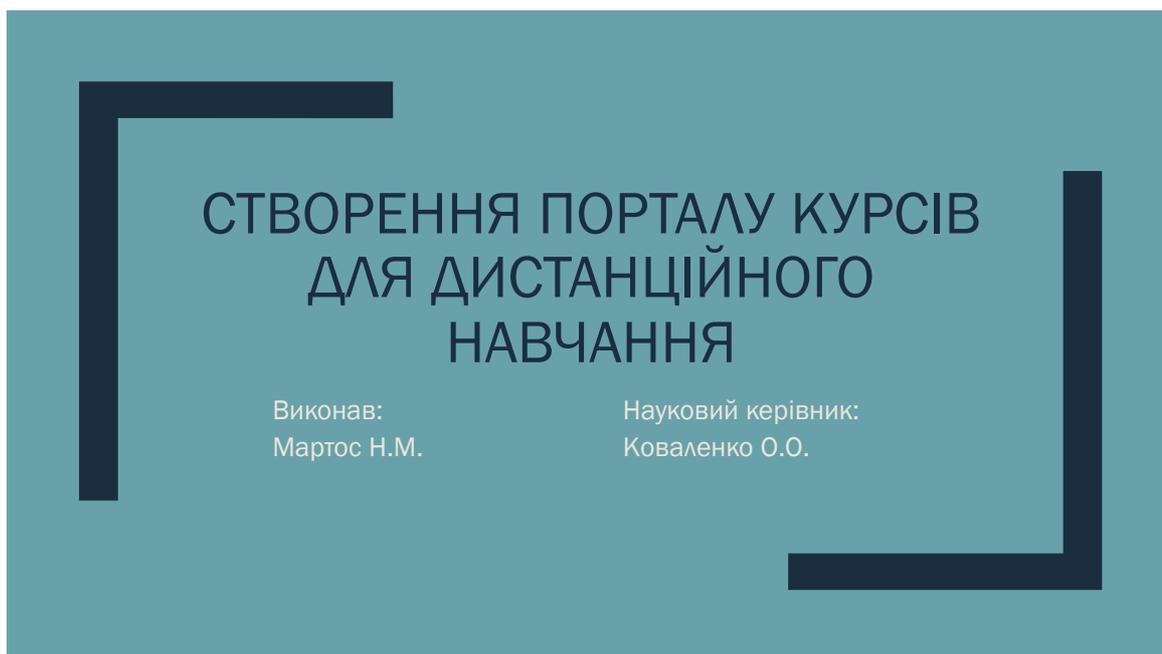


Рисунок Г.1 – Назва роботи



Рисунок Г.2 – Мета, об'єкт і предмет дослідження

Задачі дослідження

- Виконати аналіз предметної області з розробки порталу курсів для дистанційного навчання.
- Виконати порівняльний аналіз аналогів, які являють собою поширені онлайн сервіси у вигляді веб-сайтів.
- Визначити основні завдання які необхідно вирішити для розробки порталу курсів для дистанційного навчання.
- Виконати проектування інформаційно-навчальної системи.
- Розробити алгоритм розробки порталу курсів для дистанційного навчання.
- Виконати тестування порталу курсів для дистанційного навчання

Рисунок Г.3 – Задачі бакалаврської дипломної роботи

Наукова новизна полягає:

- Подальшого розвитку отримав метод розробки моделі інформаційно-навчальної системи для організації дистанційного навчання який на відміну від аналогів створений для більш ефективної його роботи.

Практичне значення отриманих результатів:

- Розроблено алгоритми та методи для інформаційно-навчальної моделі використання порталу та оптимізації його роботи.

Рисунок Г.4 – Наукова новизна

Аналоги

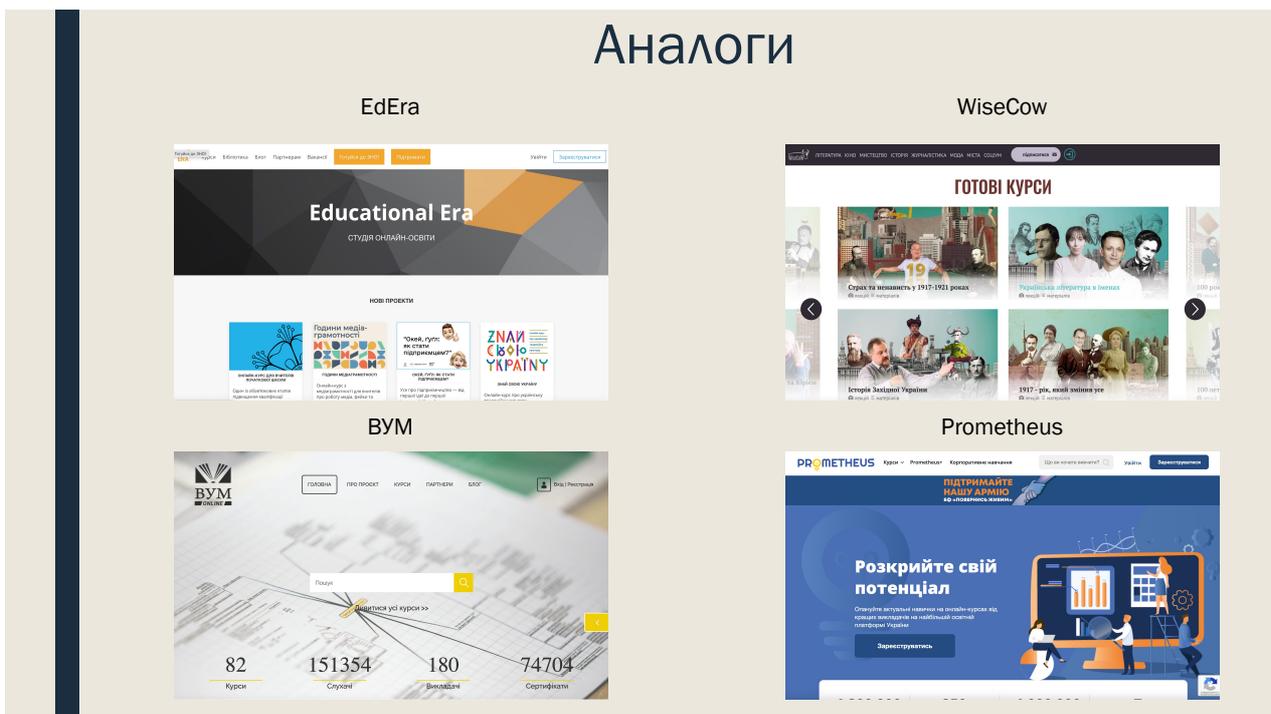


Рисунок Г.5 – Аналоги

Основні
вимоги щодо
проекування
веб-додатку
курсів для
дистанційного
навчання

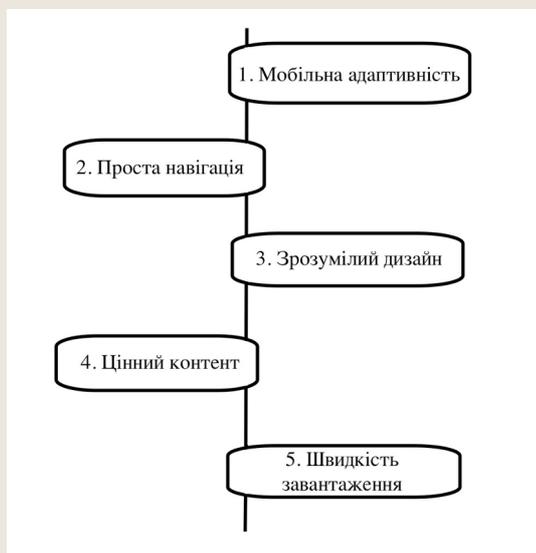


Рисунок Г.6 – Основні вимоги щодо проектування

Інформаційно-навчальна система (ІНС) – це людино-машинний комплекс, що реалізує сценарії навчальної діяльності, та підготовлених знань (структурованої інформації та системи вправ для її осмислення та закріплення), що призначений для управління навчальною діяльністю, метою якої є оволодіння знань, умінь та навичок.

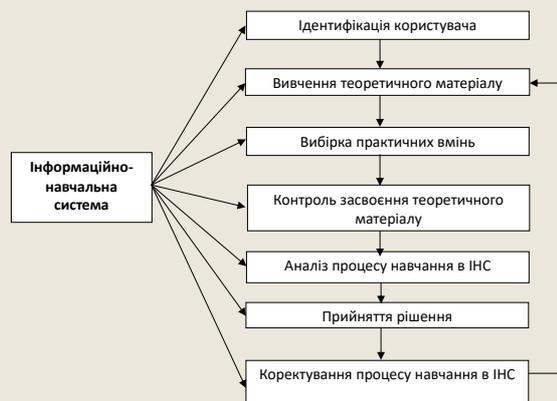


Рисунок Г.7 – Схема процесу навчання в інформаційно-навчальній системі

Основне призначення навчання, отже, і навчальної системи - оволодіння вміннями, а не тільки знаннями. Механізм здійснення діяльності є вирішення завдань. Отже, переважна більшість навчальної системи – тренуюча.

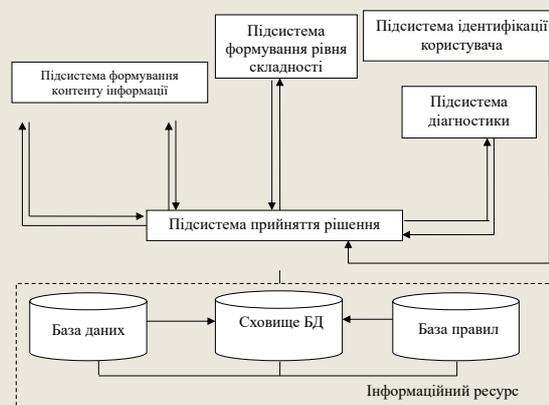


Рисунок Г.8 – Узагальнена структурна схема інформаційно-навчальної системи

Граф складності контенту при послідовному вивченні навчальних елементів

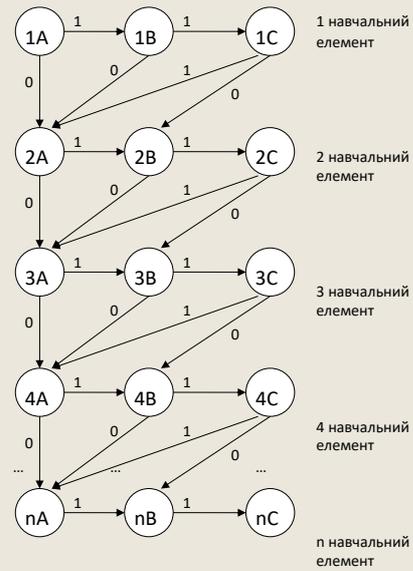


Рисунок Г.9 –Граф складності контенту при послідовному вивченні навчальних елементів

Робота з базами даних

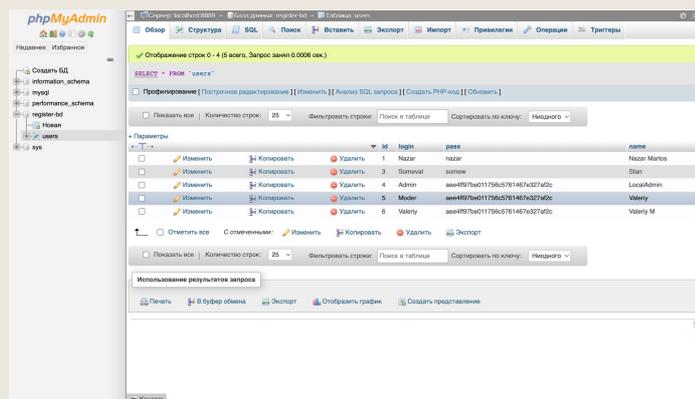


Рисунок Г.10 – База даних реєстрації

Блок-схема алгоритму роботи блоку авторизації

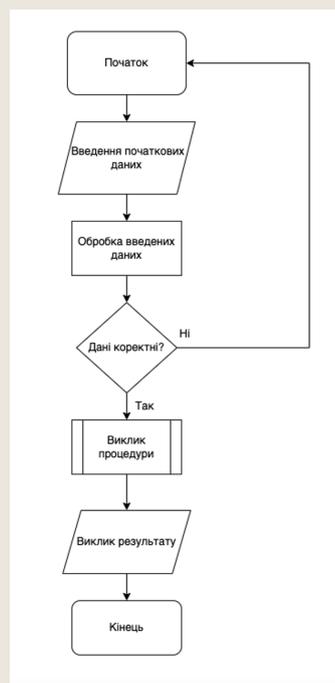


Рисунок Г.11 – Блок-схема алгоритму роботи блоку авторизації

Головна сторінка додатку

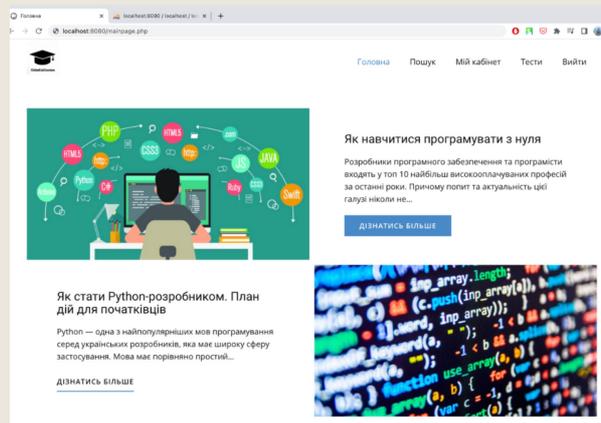


Рисунок Г.12 – Головна сторінка додатку

Сторінка кабінету користувача

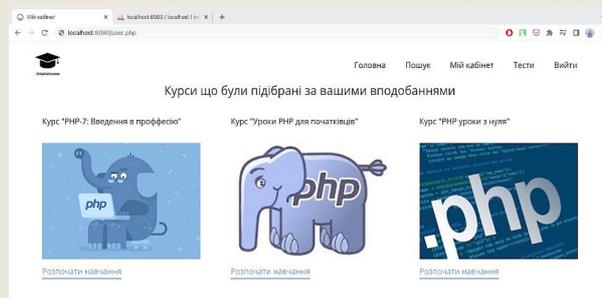


Рисунок Г.13 – Сторінка кабінету користувача

Форма додавання тестів

A screenshot of a web browser showing a form titled "Додавання тесту". The form contains several input fields and buttons. The fields are: "Назва тесту", "Додати мету", "Питання #1", "Відповідь #1", "Кількість балів за питання #1", "Дати відповідь користувачу", "Результат #1", "Бит бал #1", and "Бит бал #2". There are "Додати тест" and "Зберегти" buttons. The browser address bar shows "localhost:8080/admin.php?id=1".

Рисунок Г.14 – Форма додавання тестів

Результат проходження тестування

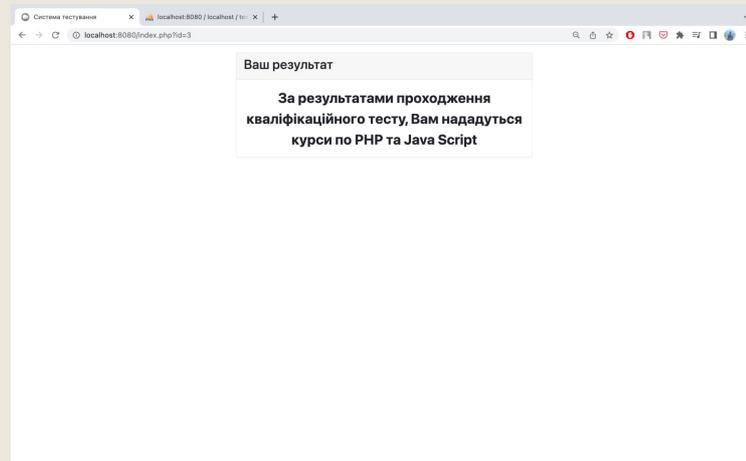


Рисунок Г.15 – Результат проходження тестування

Висновки

Під час виконання бакалаврської дипломної роботи було:

- Проведено порівняльний аналіз аналогів та визначено найбільш ефективний підхід до реалізації власного програмного продукту;
- Було визначено основні вимоги щодо проектування порталу курсів для дистанційного навчання
- Було розроблено модель інформаційно-навчальної системи
- Розроблено основні програмні модулі додатку
- Розроблено зрозумілий та комфортний інтерфейс додатку
- Проведено тестування програмних модулів порталу курсів для дистанційного навчання

Рисунок Г.16 – Висновки

Апробації та публікації

- Основні положення бакалаврської дипломної роботи доповідалися та обговорювалися на науково-технічній конференції підрозділів Вінницького національного технічного університету (Вінниця 2022).
- Основні результати дослідження опубліковані в науковій роботі: Мартос Н.М. Особливості веб-програмування у сфері дистанційного навчання. Матеріали науково-технічної конференції підрозділів Вінницького національного технічного університету (НТКП ВНТУ). Вінниця, 2022.

Рисунок Г.17 – Апробація та публікації