

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**Сумський державний університет**  
**Факультет електроніки та інформаційних технологій**  
**Кафедра інформаційних технологій**

«До захисту допущено»

В. о. завідувача кафедри

\_\_\_\_\_ Світлана ВАЩЕНКО

\_\_\_\_\_ 2024 р.

**КВАЛІФІКАЦІЙНА РОБОТА**  
**на здобуття освітнього ступеня бакалавр**

зі спеціальності 122 «Комп'ютерні науки»,

освітньо-професійної програми «Інформаційні технології проектування»

на тему: «Вебдодаток ведення документації викладачем. Модуль формування звітності \_\_\_\_\_ про \_\_\_\_\_ підвищення \_\_\_\_\_ кваліфікації викладача»

Здобувача (ки) групи IT-02 Терещенко Валерії Сергіївни  
(шифр групи) (прізвище, ім'я, по батькові)

Кваліфікаційна робота містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело.

\_\_\_\_\_ (підпис)

Валерія ТЕРЕЩЕНКО

(Ім'я та ПРІЗВИЩЕ здобувача)

Керівник \_\_\_\_\_ к. т. н., доц. Світлана ВАЩЕНКО  
(посада, науковий ступінь, вчене звання, ім'я та ПРІЗВИЩЕ)

\_\_\_\_\_ (підпис)

Сумський державний університет  
Факультет електроніки та інформаційних технологій  
Кафедра інформаційних технологій  
Спеціальність 122 «Комп'ютерні науки»  
Освітньо-професійна програма «Інформаційні технології проектування»

**ЗАТВЕРДЖУЮ**

В. о. завідувача кафедри

\_\_\_\_\_ Світлана ВАЩЕНКО

«\_\_\_\_\_» \_\_\_\_\_ 2024 р.

## **З А В Д А Н Н Я**

### **НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА СТУДЕНТУ**

*Терещенко Валерія Сергіївна*

**1 Тема роботи** «Вебдодаток ведення документації викладачем. Модуль формування звітності про підвищення кваліфікації викладача»

**керівник роботи** Ващенко Світлана Михайлівна, к.т.н., доцент,  
затверджені наказом по університету від «07» травня 2024 р. №0482-VI

**2 Строк подання студентом роботи** «26» травня 2024 р.

**3 Вхідні дані до роботи** технічне завдання на розробку вебдодатку формування звітності про підвищення кваліфікації викладачів

**4 Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)** аналіз предметної області, проектування вебдодатку формування звітності про підвищення кваліфікації викладача, розробка вебдодатку формування звітності про підвищення кваліфікації викладача.

**5 Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)** актуальність, постановка задачі, аналіз програмних продуктів-аналогів, вимоги до продукту, архітектура програмного комплексу, структурно-функціональна модель, діаграма варіантів використання, логічна модель бази даних, фізична модель бази даних, перелік реалізованих таблиць бази даних, засоби реалізації, макети сторінок, демонстрація сторінок вебдодатку, адміністративна панель, шаблон звіту про підвищення кваліфікації, результат звіту про підвищення кваліфікації, практична реалізація, тестування, висновки.

## 6. Консультанти розділів роботи:

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв

7. Дата видачі завдання « 08 » квітня 2024 р.

## КАЛЕНДАРНИЙ ПЛАН

№ п/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Визначення актуальності роботи, дослідження предметної області та проведення аналізу аналогів вебдодатків.	08.04.2024 - 11.04.2024	
2	Створення специфікації.	11.04.2024 - 16.04.2024	
3	Проектування структури вебдодатку.	16.04.2024 - 17.04.2024	
4	Розробка макетів сторінок вебдодатку.	17.04.2024 - 23.04.2024	
5	Верстка сторінок вебдодатку.	23.04.2024 - 03.05.2024	
6	Розробка бази даних для вебдодатку.	03.04.2024 - 09.05.2024	
7	Розробка усіх потрібних функцій вебдодатку.	09.05.2024 - 17.05.2024	
8	Тестування методом білого ящика.	17.05.2024 - 20.05.2024	
9	Оформлення пояснювальної записки кваліфікаційної роботи бакалавра	08.04.2024 - 26.05.2024	

Студент

\_\_\_\_\_

(підпис)

Валерія ТЕРЕЩЕНКО

Керівник роботи

\_\_\_\_\_

(підпис)

к.т.н., доц. Світлана ВАЩЕНКО

## АНОТАЦІЯ

Тема кваліфікаційної роботи бакалавра «Вебдодаток ведення документації викладачем. Модуль формування звітності про підвищення кваліфікації викладача».

Пояснювальна записка складається зі вступу, 3 розділів, висновків, списку використаних джерел із 14 найменувань, 4 додатків. Загальний обсяг роботи – 127 сторінок, у тому числі 49 сторінок основного тексту, 2 сторінки списку використаних джерел, 78 сторінок додатків.

Актуальність роботи полягає в вирішенні проблеми автоматизації процесу збору та розрахунку даних щодо підвищення кваліфікації викладачів.

Метою даної кваліфікаційної роботи є розробка вебдодатку автоматизації формування звітності про підвищення кваліфікації викладача кафедри інформаційних технологій, Сумського державного університету.

Результатом проведеної роботи став вебдодаток формування звітності про підвищення кваліфікації викладача.

Застосування цієї розробки дозволить значно скоротити час, який витрачається на формування звітності вручну, тим самим полегшуючи роботу викладачів. Це визначає практичну значимість представленої роботи.

Ключові слова: вебдодаток, підвищення кваліфікації, викладач, звітність, клієнт-серверна архітектура, контрольні показники, накопичувальна система, розробка.

## ЗМІСТ

ВСТУП.....	5
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ .....	7
1.1 Огляд останніх досліджень і публікацій .....	7
1.2 Аналіз програмних продуктів-аналогів .....	11
1.3 Постановка задачі .....	15
1.4 Вибір засобів реалізації .....	16
2 ПРОЕКТУВАННЯ ВЕБДОДАТКУ ФОРМУВАННЯ ЗВІТНОСТІ ПРО ПІДВИЩЕННЯ КВАЛІФІКАЦІЇ ВИКЛАДАЧА.....	17
2.1 Структурно-функціональне моделювання .....	17
2.2 Моделювання варіантів використання вебдодатку .....	20
2.3 Проектування бази даних вебдодатку .....	23
3 РОЗРОБКА ВЕБДОДАТКУ ФОРМУВАННЯ ЗВІТНОСТІ ПРО ПІДВИЩЕННЯ КВАЛІФІКАЦІЇ ВИКЛАДАЧА .....	26
3.1 Архітектура вебдодатку .....	26
3.2 Програмна реалізація.....	27
3.3 Використання програмного додатку.....	36
3.4 Тестування вебдодатку.....	40
ВИСНОВКИ.....	46
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ .....	48
ДОДАТОК А.....	50
ДОДАТОК Б.....	69
ДОДАТОК В .....	84
ДОДАТОК Г.....	125

## ВСТУП

У сучасному вищому навчальному закладі важливою складовою є система підвищення кваліфікації науково-педагогічних працівників. Однак, процес ведення звітності контрольних показників накопичувальної системи обрахунку основних результатів підвищення кваліфікації часто вимагає значних зусиль від викладачів та адміністрації. З метою спрощення цього процесу, потрібно розробити вебдодаток, який спрямований на швидке та ефективне заповнення контрольних показників для викладачів кафедри інформаційних технологій, Сумського державного університету.

Отже, метою даної кваліфікаційної роботи є розробка вебдодатку автоматизації формування звітності про підвищення кваліфікації викладача кафедри інформаційних технологій, Сумського державного університету.

Об'єкт дослідження – процес формування викладачами звітності щодо підвищення кваліфікації за накопичувальною системою в Сумському державному університеті.

Предмет дослідження – технології автоматизації підготовки звітної документації в процесі звітування за підвищення кваліфікації.

Для досягнення мети проекту необхідно виконати наступні задачі:

- визначити актуальність роботи, дослідити предметну область та провести аналіз аналогів вебдодатків;
- спроектувати структуру вебдодатку;
- розробити макети сторінок вебдодатку;
- провести моделювання структури та роботи майбутньої розробки;
- створити бази даних для зберігання інформації про підвищення кваліфікації викладачів;
- реалізувати функціонал для автоматизованого формування звітності;
- виконати тестування вебдодатку.

Розроблений вебдодаток дозволить вирішити актуальну проблему автоматизації процесу збору та розрахунку даних щодо підвищення кваліфікації викладачів. Застосування цієї розробки дозволить значно скоротити час, який витрачається на формування звітності вручну, тим самим полегшуючи роботу викладачів та адміністрації. Це визначає практичну значимість представленої роботи.

# 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

## 1.1 Огляд останніх досліджень і публікацій

Відповідно до статті 18 Закону про освіту України, підвищення кваліфікації включає в себе отримання нових знань та навичок або покращення існуючих у межах своєї професійної діяльності або певної галузі знань [1, ст. 18, п. 6].

Державний нормативний документ України, зокрема «Постанова про деякі питання професійного розвитку науково-педагогічних працівників», визначає поняття підвищення кваліфікації як систему організаційно-педагогічних заходів, спрямованих на розвиток та удосконалення професійної компетентності педагогічних працівників для забезпечення якості освітньої діяльності [2].

Отже, головною метою підвищення кваліфікації є оновлення та поглиблення знань, умінь та навичок науково-педагогічних працівників, що є важливою складовою їхньої професійної діяльності, оскільки це дозволяє покращити якість надання освітніх послуг, впроваджувати сучасні методи навчання для ефективного вирішення сучасних завдань та викликів у відповідній сфері.

З урахуванням публікації [3] можна виділити кілька аспектів, що визначають підвищення кваліфікації педагогічного працівника як важливу складову його роботи:

- актуальність знань. Під час процесу навчання з'являються нові технології та методи, які викладачі повинні вивчати, щоб залишатися в курсі сучасних тенденцій.
- стимулювання саморозвитку. Вивчення нового надає викладачам можливість постійно розвиватися та покращувати свою професійну кар'єру. Постійне навчання допомагає науково-педагогічним працівникам зберігати високий професійний рівень та впевненість у своїх здібностях.
- вплив на здобувачів освіти. Викладачі, які постійно вдосконалюють свою кваліфікацію, можуть більш ефективно впливати на здобувачів та створювати сприятливу навчальну атмосферу.



Підвищення кваліфікації педагогічних та науково-педагогічних працівників може здійснюватися через різноманітні форми і види навчання та професійного розвитку. Наведемо деякі з них [4]:

- навчання за програмами підвищення кваліфікації, які можуть бути спеціально розроблені для вчителів та інших педагогічних працівників з метою оновлення їхніх знань та навичок;
- участь у різноманітних навчальних заходах, таких як семінари, практикуми, тренінги, вебінари та майстер-класи;
- стажування, яке може включати як практичний досвід роботи в інших навчальних закладах, так і участь у наукових дослідженнях та проектах.

Крім того, педагоги можуть підвищувати свою кваліфікацію шляхом участі у програмах академічної мобільності, науковому стажуванні, самоосвіті, отриманні наукових ступенів або вищої освіти [4].

Загальна кількість академічних годин для підвищення кваліфікації протягом п'яти років повинна становити не менше 180 годин [5, с. 3], тому кожні п'ять років науково-педагогічні працівники Сумського державного університету (СумДУ), зокрема кафедри Інформаційних технологій, проходять підвищення кваліфікації.

Згідно діючого «Положення про підвищення кваліфікації співробітників СумДУ, педагогічних та науково-педагогічних працівників інших закладів освіти у Сумському державному університеті» [6] в СумДУ є можливим перерахування результатів підвищення кваліфікації за контрольними показниками накопичувальної системи у такому порядку: спочатку працівники повинні подати до Центру розвитку кадрового потенціалу (ЦРКП) звіт та документи про своє підвищення кваліфікації за певним шаблоном. Потім ЦРКП розглядає ці звіти до першого числа кожного місяця і робить пропозиції щодо затвердження результатів на вчених радах інститутів або факультетів. Після цього, на основі рішень вчених рад, ЦРКП готує проєкт наказу про зарахування результатів підвищення кваліфікації.

Всі можливі види робіт, які можна перераховувати як результати підвищення кваліфікації науково-педагогічного працівника, регламентуються документом «Контрольні показники накопичувальної системи обрахунку основних результатів

підвищення кваліфікації науково-педагогічних працівників Сумського державного університету» [7]. На рисунку 1.1 зображено як виглядає частина таблиці контрольних показників накопичувальної системи СумДУ.

№ п/п	Контрольний показник	Бали	Суб'єкт контролю <sup>1)</sup>	Примітка
<b>1 Навчання та стажування</b>				
1.1	Навчання за програмою підвищення кваліфікації	6×k <sup>2)</sup>	ЦРКП	за винятком врахованого в інших показниках
1.2	Стажування, у т.ч. наукове стажування	6×k <sup>2)</sup>		
1.3	Участь у якості слухача у семінарах, практикумах, тренінгах, вебінарах, майстер-класах тощо	6×k <sup>2)</sup>		
1.4	Участь у програмах академічної мобільності з метою підвищення кваліфікації, викладання за кордоном, виконання спільних проектів та наукове співробітництво			
1.4.1	в іноземних університетах і наукових установах, що входять до TOP-500 рейтингу QS <sup>6)</sup>	20×k <sup>2)</sup>		
1.4.2	в іноземних університетах і наукових установах, що входять до TOP-1000 рейтингу Webometrics <sup>6)</sup>	10×k <sup>2)</sup>		
1.4.3	в іноземних університетах, наукових та інших установах <sup>6)</sup>	6×k <sup>2)</sup>		
1.4.4	участь у програмах академічної мобільності у дистанційній формі	k <sup>2)</sup>		
1.5	Здобуття першого (бакалаврського), другого (магістерського) рівня вищої освіти, третього (освітньо-наукового/освітньо-творчого) рівня або наукового рівня вищої освіти вперше або за іншою спеціальністю у межах професійної діяльності або галузі знань	1000	ВК	

Рисунок 1.1 – Вигляд таблиці контрольних показників накопичувальної системи СумДУ

Для зарахування підвищення кваліфікації за накопичувальною системою викладачі самостійно заповнюють звіт (за встановленим шаблоном в [6]), який передбачає їх досягнення у навчанні та стажуванні, результати навчально-методичної роботи, які визнаються як свідчення професійного зростання науково-педагогічного працівника та інше, щоб подати цей звіт до Центру розвитку кадрового потенціалу (ЦРКП).

Науково-педагогічні працівники самостійно заповнюють показники накопичувальної системи та обраховують основні результати підвищення кваліфікації.

Для прикладу як повинен виглядати звіт про підвищення кваліфікації викладача, візьмемо звіт науково-педагогічного працівника кафедри Інформаційних технологій, Сумського державного університету, Ващенко Світлани Михайлівни 2023 року. На рисунку 1.2 можна побачити табличку, в якій викладач самостійно обчислює та заповнює свої бали та види контрольних показників.

**ЗВІТ ПРО РЕЗУЛЬТАТИ ПІДВИЩЕННЯ КВАЛІФІКАЦІЇ  
ЗА НАКОПИЧУВАЛЬНОЮ СИСТЕМОЮ**  
Ващенко Світлана Михайлівна  
В.о. зав. кафедрою, кафедра інформаційних технологій

звітний період \_\_\_\_\_ року – \_\_\_\_\_ року

Тематичне спрямування<sup>1)</sup> \_\_\_\_\_ інформаційні технології

№ 2)	Контрольний показник	Результати обрахування, бали	
		Пропозиція <sup>3)</sup>	визнання результатів <sup>4)</sup>
1. Навчання та стажування			
1.1.	Навчання за програмою підвищення кваліфікації 1) «ІТ інструменти для викладачів» від компанії <u>GlobalLogic</u> ; липень 2023 р., сертифікат CS № 12048/2023, 18 годин; 6*18= 108 2) «TECH SUMMER FOR TEACHERS BOOTCAMP» від компанії <u>SoftServe</u> ; липень-серпень 2022 р.; сертифікат ТМ №2022/00156; 10 год.; 6*10=60 3) “Інституційна культура академічної доброчесності: національний досвід та кращі практики Європейського союзу ”, сертифікат №101048055-25-44; травень 2022 р.; 6 год.; 6*6 =36 4) підвищення кваліфікації за програмою «Акредитація освітньої програми: кращі практики та проблемні питання», посвідчення СП № 05408289 / 3034-21; листопад 2021 р.; 30 год.; 6*30=180	384	
1.5.	Здобуття другого (магістерського) рівня вищої освіти, за іншою спеціальністю у межах професійної діяльності або галузі знань; диплом М21№024832, виданий 26 лютого 2021 р.	1000	

Рисунок 1.2 – Вигляд звіту про підвищення кваліфікації викладача

Щоб обчислювати бали, науково-педагогічного працівник повинен самостійно ознайомитись із нормативним документом на сайті Сумського державного університету [7].

З цього можна зробити висновок, що викладачам необхідно робити всі розрахунки вручну і заносити результати в документ MS Word. Можна також використовувати таблиці в MS Word для виконання розрахунків, але це вимагає знань роботи з полями і формулами, що може бути складним для пересічного користувача.

Щоб скоротити час на написання звіту можна використовувати додаток MS Excel для введення формул, але все одно залишається незручність формування звітності. У випадку використання MS Excel для розрахунків, також потрібно знати, як користуватися цією програмою.

Однак у будь-якому випадку потрібно постійно звертатися до положення, щоб брати норми розрахунку балів. Це не зручно та приводить до значних зусиль та витрат часу.

Для вирішення цієї проблеми потрібно розробити вебдодаток, який допоможе викладачам швидко та зручно заповнювати звіти про підвищення кваліфікацію.

## **1.2 Аналіз програмних продуктів-аналогів**

Для визначення вимог майбутнього вебдодатку проведено дослідження існуючих аналогів для формування звітності з підвищення кваліфікації викладачів, а саме karazin.ua, РРК.isuo.org, imzo.gov.ua.

На рисунку 1.3 зображено вебдодаток Харківського національного університету імені В. Н. Каразіна [8], на якому немає ніяких обрахунків або генерації звітів, лише інформація про підвищення кваліфікації для вчителів. Така ситуація на усіх офіційних сайтах університетів, як приклад візьмемо один із них.

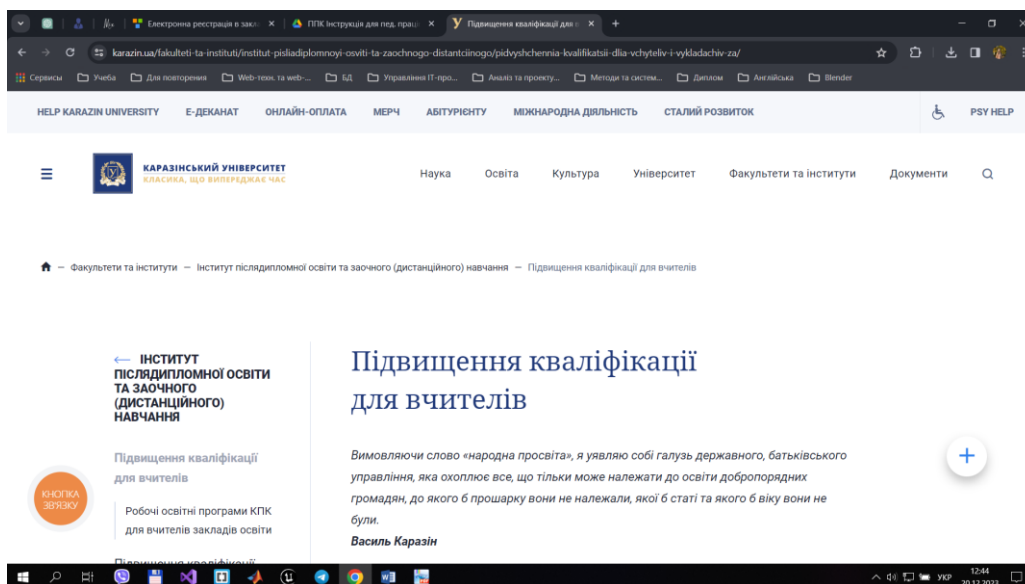


Рисунок 1.3 – Приклад вебдодатку університету імені В. Н. Каразіна

На рисунку 1.4 продукт, який виключно тільки для ЗЗСО (шкіл). Цей вебдодаток [9] дозволяє знайти програми підвищення кваліфікації (ППК), подивитись інформацію про запропоновані ППК та дізнатись їх вартість, отримувати інформацію про свідоцтва про пройдено ППК та обрати ППК з виміром в кількості годин або кредитів. Вебдодаток більше розрахований на пошук та вибір програми для вчителів, а не для розрахунку балів з підвищення кваліфікації.

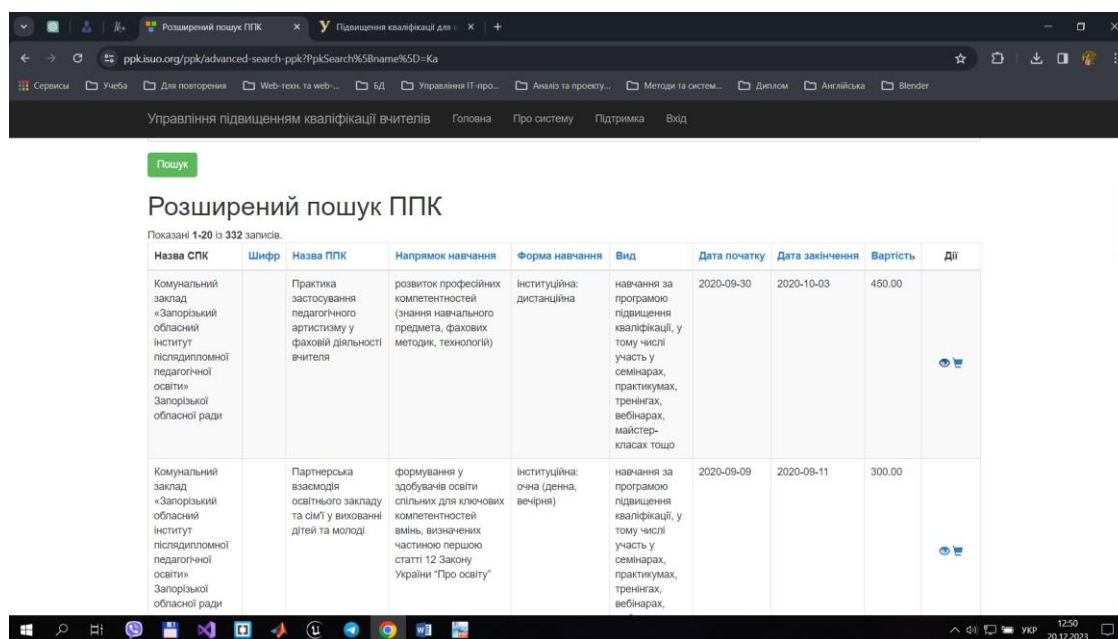


Рисунок 1.4 – Вебдодаток з управління підвищенням кваліфікації вчителів

На цьому сайті можна завантажувати HTML, CSV, Excel, PDF, JSON та текстові файли (рис. 1.5).

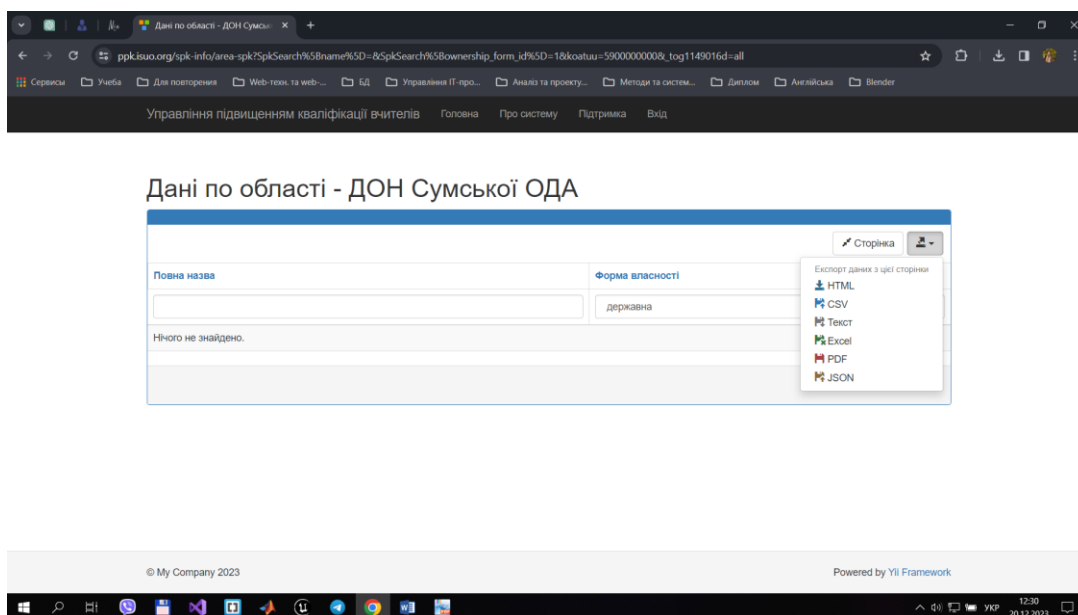


Рисунок 1.5 – Формати файлів, які можна завантажити

На рисунку 1.6 зображено вебдодаток [10], який має освітні послуги з підвищення кваліфікації, сертифікати, освітні програми, тощо, але немає функції формування звітності для викладачів.

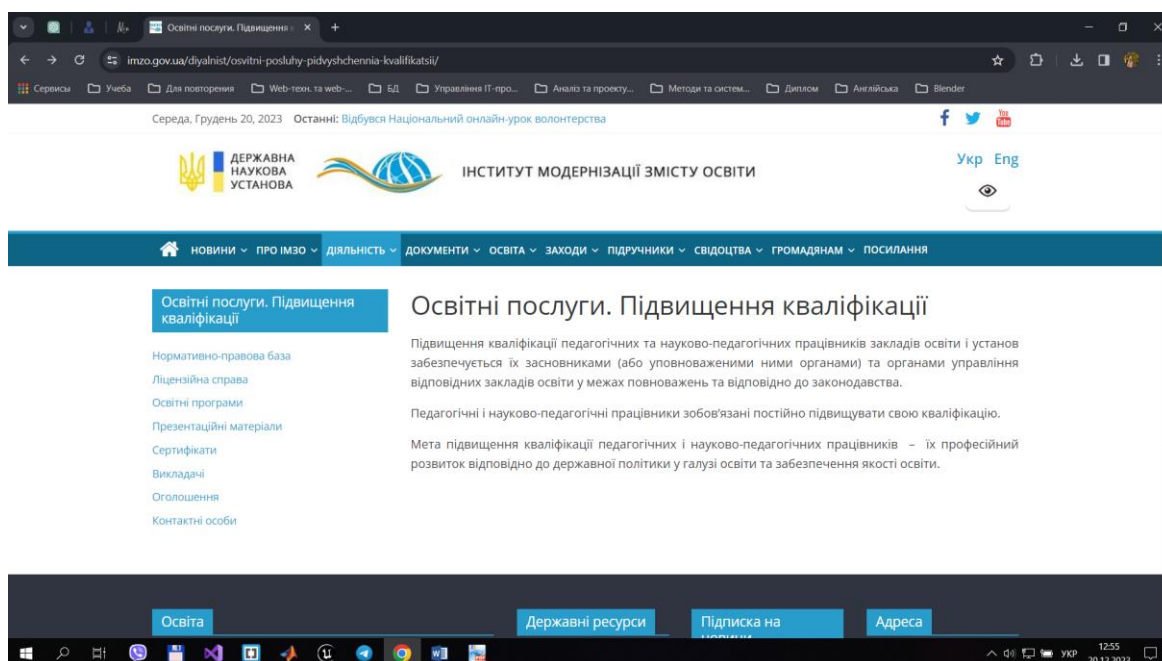


Рисунок 1.6 – Інститут модернізації змісту освіти

Після детального аналізу аналогів веб-сайтів з підвищення кваліфікації, було визначено їх переваги та недоліки. Його результати представлені в таблиці 1.1.

Таблиця 1.1 – Порівняльний аналіз продуктів-аналогів

<b>Характеристика</b>	<b>karazin.ua</b>	<b>РРК.isuo.org</b>	<b>imzo.gov.ua</b>
Основна мета щодо підвищення кваліфікації	Інформація про ПК	Пошук ППК	Освітні послуги по програми ПК
Функціонал для викладачів по формуванню звітності з підвищення кваліфікації	-	-	-
Можливість завантажувати файли	-	+	-

Отже, вебдодатки karazin.ua, РРК.isuo.org, imzo.gov.ua не мають функціоналу який повинен бути в створюваному вебдодатку, але ці сайти мають інші напрямки та корисні з боку вибору програми по підвищенню кваліфікації або освітньої послуги.

Розроблений вебдодаток планується розмістити на офіційному веб-сайті кафедри інформаційних технологій. Аналіз сайтів кафедр різних університетів показав відсутність такого вебдодатку, який буде створювати звіт із підвищення кваліфікації викладачів або навіть просто обраховувати суму контрольних показників накопичувальної системи.

Дана кваліфікаційна робота спрямована на вирішення цієї актуальної проблеми через розробку вебдодатку, який значно полегшить та прискорить процес введення та обробки інформації про підвищення кваліфікації завдяки автоматизації процесу. Створення такого інструменту вирішить питання неефективності самостійного заповнення звітів та допоможе викладачам кафедри інформаційних технологій фокусуватися на професійному розвитку, замість витрат часу на адміністративні аспекти. Для вдосконалення проекту потрібно врахувати позитивні аспекти, які присутні в аналогах, наприклад, різноманітність форматів звітності як у другому аналогу [9].

### 1.3 Постановка задачі

Метою даної кваліфікаційної роботи є розробка вебдодатку автоматизації формування звітності з підвищення кваліфікації викладачів кафедри інформаційних технологій, Сумського державного університету.

Основні вимоги до створюваного програмного продукту є наступними:

- забезпечити автоматичний розрахунок контрольних показників підвищення кваліфікації відповідно до нормативів університету;
- реалізувати систему авторизації викладачів;
- розробити зручний та інтуїтивно зрозумілий інтерфейс для введення контрольних показників;
- розробити формування звітності викладачів в зручному форматі (Word);
- забезпечити адміністративну сторінку, завдяки якій адміністратор вебдодатку зможе редагувати усі дані, які є в базі даних, додавати та видаляти дані.

Для досягнення мети проєкту необхідно виконати наступні задачі:

- спроектувати структуру вебдодатку та змодельовати його роботу;
- розробити макети вебдодатку;
- розробити усі сторінки вебдодатку;
- створити бази даних для зберігання інформації про підвищення кваліфікації викладачів;
- реалізувати функціонал для автоматизованого формування звітності;
- забезпечити безпеку та конфіденційність інформації в системі;
- виконати тестування вебдодатку.

Вимоги до проєкту в цілому, структури вебдодатку, видів забезпечення та функціонування системи описані у технічному завданні на розробку проєкту (додаток А).



## 1.4 Вибір засобів реалізації

Для створення веб-додатку обрано HTML як мову розмітки, оскільки вона є стандартом для створення структури веб-сторінок. HTML дозволяє легко розміщувати текст, зображення та інші мультимедійні елементи на сторінці, що робить його ідеальним для створення каркасу веб-сторінок.

Каскадні таблиці стилів (CSS) обрано для надання веб-сторінкам візуальних ефектів та адаптивності до різних пристроїв. CSS дозволяє задавати вигляд і розміщення елементів на сторінці, що дозволяє створювати привабливий та зручний для користувачів інтерфейс.

JavaScript обрано для надання динамічності веб-сторінкам та взаємодії з користувачем без перезавантаження сторінки. JavaScript дозволяє виконувати різні дії на сторінці, такі як анімація, валідація форм, реакція на події користувача тощо.

PHP обрано для написання модулів, що реалізують логіку роботи вебдодатку, оскільки ця мова програмування добре підходить для обробки форм, доступу до бази даних MySQL та генерації веб-сторінок на серверному боці. PHP є потужним інструментом для створення функціональності, яка взаємодіє з інтерфейсною частиною додатку.

Для зберігання даних обрано реляційну систему управління базами даних (СУБД) MySQL, оскільки ця СУБД є швидкою та надійною, що забезпечує ефективне управління та збереженням інформації користувачів.

Для генерації звітів обрано бібліотеку docxtemplater, яка є бібліотекою JavaScript, яка дозволяє створювати та редагувати документи Word (формат .docx) програмним шляхом. Вона використовує шаблони документів, в яких визначені спеціальні мітки або змінні, які можуть бути замінені на конкретні дані під час виконання скрипту. Це робить її дуже корисною для автоматизації створення документів, таких як звіти, договори, сертифікати та інші документи, які потребують динамічного вмісту.

## 2 ПРОЕКТУВАННЯ ВЕБДОДАТКУ ФОРМУВАННЯ ЗВІТНОСТІ ПРО ПІДВИЩЕННЯ КВАЛІФІКАЦІЇ ВИКЛАДАЧА

### 2.1 Структурно-функціональне моделювання

Для структурно-функціонального моделювання вебдодатку обрано діаграму IDEF0, щоб зрозуміти процес формування звітної документації за результатами підвищення кваліфікації викладача. Інтеграційне визначення для функціонального моделювання, або IDEF0 – це модель, що відображає дії, повноваження та активності процесу. Ця модель використовується для ілюстрації вхідних та вихідних даних, осіб, що приймають рішення, та механізмів процесу. Зазвичай вона застосовується при розробці нових систем для визначення вимог та функціональності [11].

Слово «інтеграційне» в контексті IDEF0 вказує на те, що ця модель призначена для інтеграції різних елементів та аспектів процесу в єдину систему опису. Це означає, що IDEF0 дозволяє об'єднати різні аспекти процесу, такі як дії, повноваження, вхідні та вихідні дані, у єдину концепцію, що допомагає краще зрозуміти та управляти процесом.

Функціональна модель вебдодатку формування звітності з підвищення кваліфікації у нотації IDEF0 представлена на рисунку 2.1.

Визначено наступні основні елементи:

- вхідні дані: дані про викладача (логін та пароль, вчений ступінь та наукове звання, посада викладача, фото за наявності, ПІБ);
- вихідні дані: сформований звіт (файл MS Word, який містить звітність з підвищення кваліфікації викладача (з обрахунками балів)), збережена інформація про звіт в базу даних (шлях до файлу, який зберігається на сервері та його ім'я);
- управління: положення про контрольні показники накопичувальної системи обрахунку основних результатів підвищення кваліфікації науково-педагогічних працівників Сумського державного університету, штат кафедри та шаблон звіту;

– механізми: вебдодаток формування звітності про підвищення кваліфікації викладача, викладач, технічне забезпечення, прикладне програмне забезпечення.

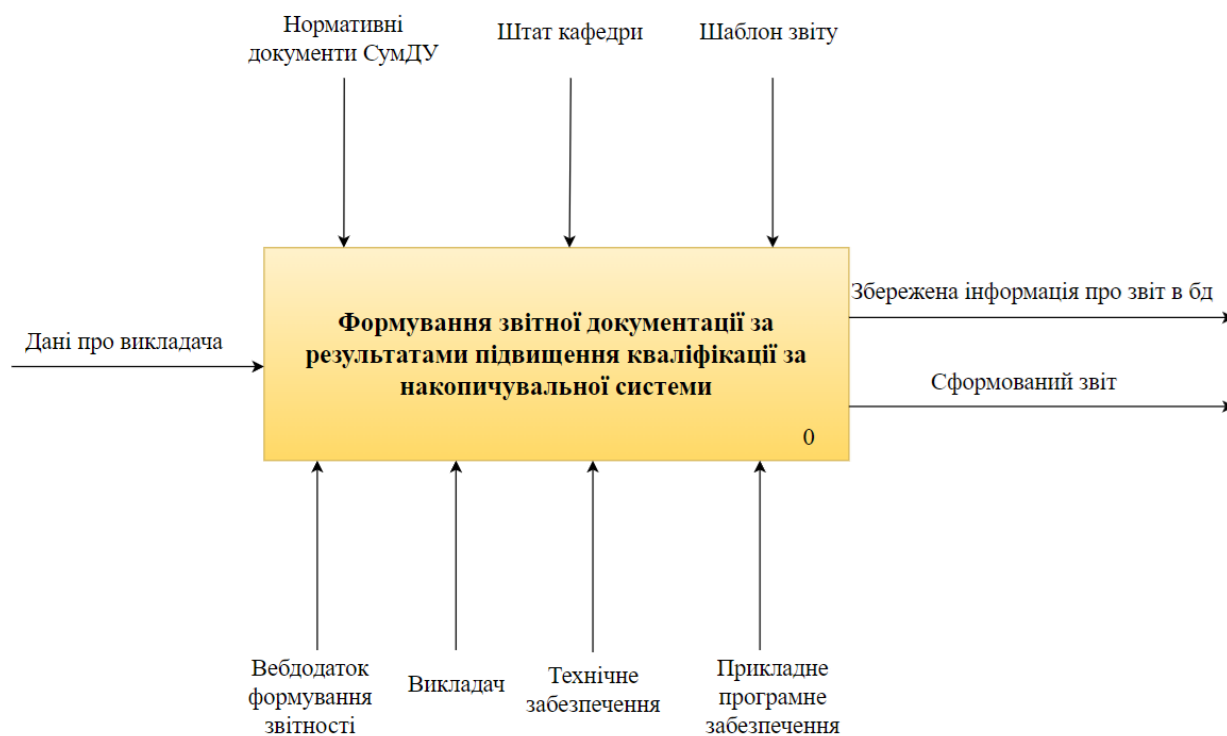


Рисунок 2.1 – Контекстна діаграма формування звітності про підвищення кваліфікації викладача

Для кращого розуміння структури вебдодатку його розгорнуто на окремі підпроцеси. Ці підпроцеси включають в себе авторизацію, введення викладачем своїх даних про підвищення кваліфікацію у форму, обрахунок основних результатів підвищення кваліфікації науково-педагогічних працівників та формування звіту в MS Word.

У таблиці 2.1 для зручності представлено дані кожного з цих підпроцесів.

Таблиця 2.1 – Підпроцеси діаграми декомпозиції для підтримки формування звітності про підвищення кваліфікації викладача за результатами накопичувальної системи

Підпроцес	Вхідні дані	Управління	Механізми	Вихідні дані
Авторизація	Дані про викладача	Штат кафедри	Вебдодаток формування звітності, викладач, технічне забезпечення, прикладне програмне забезпечення	Профіль викладача
Введення даних про підвищення кваліфікацію	—	Нормативні документи СумДУ	Вебдодаток формування звітності, викладач, технічне забезпечення, прикладне програмне забезпечення	Заповнена форма викладачем про підвищення кваліфікацію
Обрахунок основних результатів підвищення кваліфікації науково-педагогічних працівників	—	Нормативні документи СумДУ	Вебдодаток формування звітності, технічне забезпечення, прикладне програмне забезпечення	Розраховані бали контрольних показників викладача
Формування звіту в MS Word	—	Шаблон звіту	Вебдодаток формування звітності, викладач, технічне забезпечення, прикладне програмне забезпечення	Файл MS Word, який містить звітність про підвищення кваліфікації викладача (з обрахунками балів)

Декомпозиція першого рівня зображена на рисунку 2.2

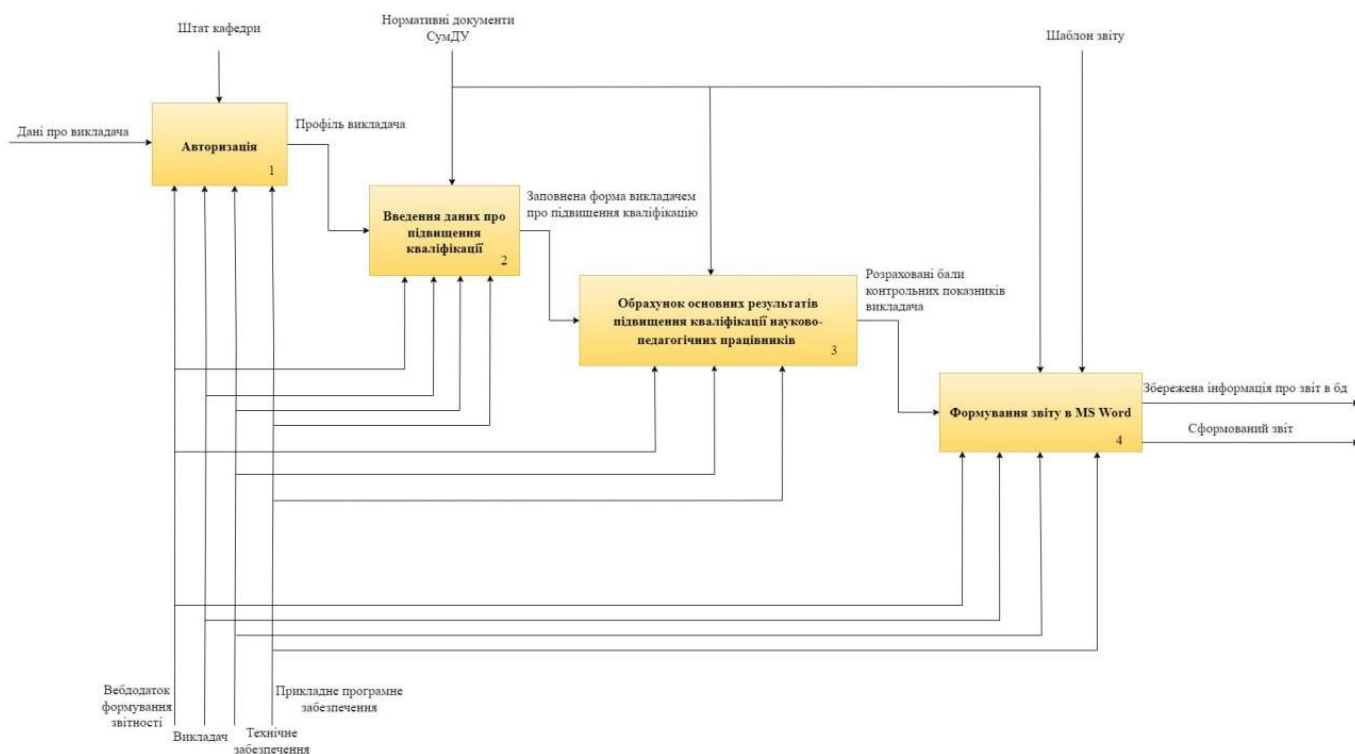


Рисунок 2.2 – Декомпозиція першого рівня

## 2.2 Моделювання варіантів використання вебдодатку

Проаналізувавши процеси, які мають бути у вебдодатку, можна визначити, хто та які дії може здійснювати використовуючи розроблюваний вебдодаток формування звітності. Для цього обрано діаграму варіантів використання (use case diagram), яка є однією з основних моделей в UML (Unified Modelling Language). Вона використовується для опису функціональних вимог до системи. Діаграма візуально представляє взаємодію між користувачами (акторами) та системою через різні сценарії використання (use cases) [12].

У таблиці 2.2 описані усі актори діаграми варіантів використання.

Таблиця 2.2 – Актори системи

Назва актора	Короткий опис
<b>Актори - користувачі:</b>	
<b>Адміністратор</b>	Адміністратор сайту може додавати нових викладачів, редагувати та видаляти їх. Також редагувати, видаляти та додавати контрольні показники та бали за них, які впливають на підвищення кваліфікації. Обов'язково адміністратор повинен авторизуватися.
<b>Викладач</b>	Викладач може переглядати свій особистий профіль, переглядати свої звіти минулих років та створювати новий, після того, усе це після того як викладач авторизується. За потреби він може зберегти звіт з підвищення кваліфікації у файл MS Word, який формується на основі шаблону Template MS Word.
<b>Актори - зовнішні системи:</b>	
<b>БД MySQL</b>	У базі даних MySQL зберігається інформація про контрольні показники накопичувальної системи. Також зберігається інформація про викладачів.
<b>Артефакти:</b>	
Звіт з розширенням docx	Збереження обробленої інформації на основі шаблону звіту в MS Word.

Можливі дії для викладача є наступними:

- авторизація, яка є обов'язковою для усіх інших дій;
- заповнення форми про підвищення кваліфікації;
- формування звітності;
- видалення звіту про підвищення кваліфікації;
- перегляд особистого профілю;

- перегляд звітів про підвищення кваліфікації;
- завантаження звітів;

Можливі дії для адміністратора сайту:

- додавання даних в базу даних;
- редагування даних;
- видалення даних;

Діаграма варіантів використання вебдодатку формування звітності про підвищення кваліфікації зображена на рисунку 2.3.

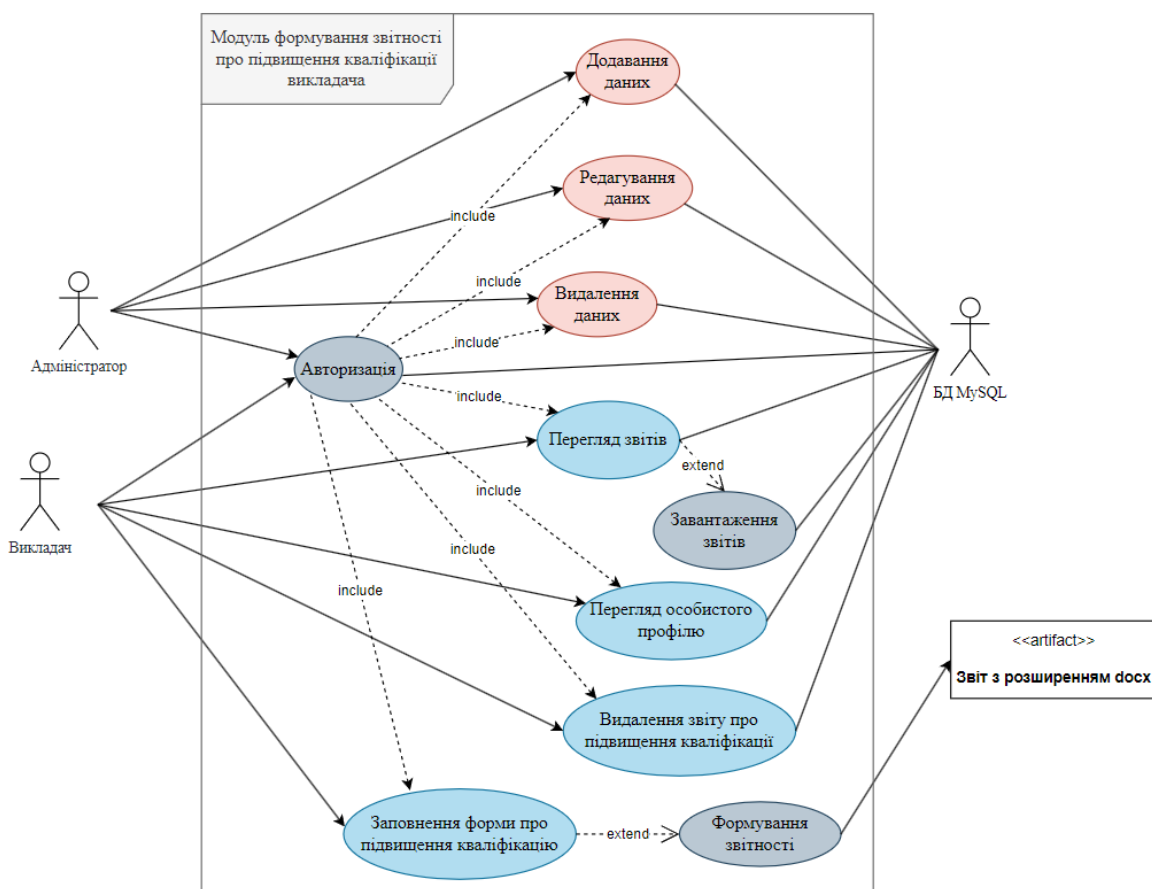


Рисунок 2.3 – Діаграма варіантів та використання вебдодатку формування звітності

## 2.3 Проектування бази даних вебдодатку

Для розроблюваного вебдодатку обов'язково потрібна база даних, яка буде містити дані про викладачів, адміністраторів, контрольних показників накопичувальної системи та сформованих звітів. Обрано реляційну базу даних, що організовує дані у вигляді таблиць. Кожна таблиця складається з рядків і стовпців, де кожен рядок представляє окремий запис, а стовпці містять дані про атрибути цього запису. Ключовим елементом реляційної бази даних є можливість встановлення відносин між таблицями через використання ключів [13].

Усього в базі даних 15 сутностей. Логічна модель представлена на рисунку 2.4.

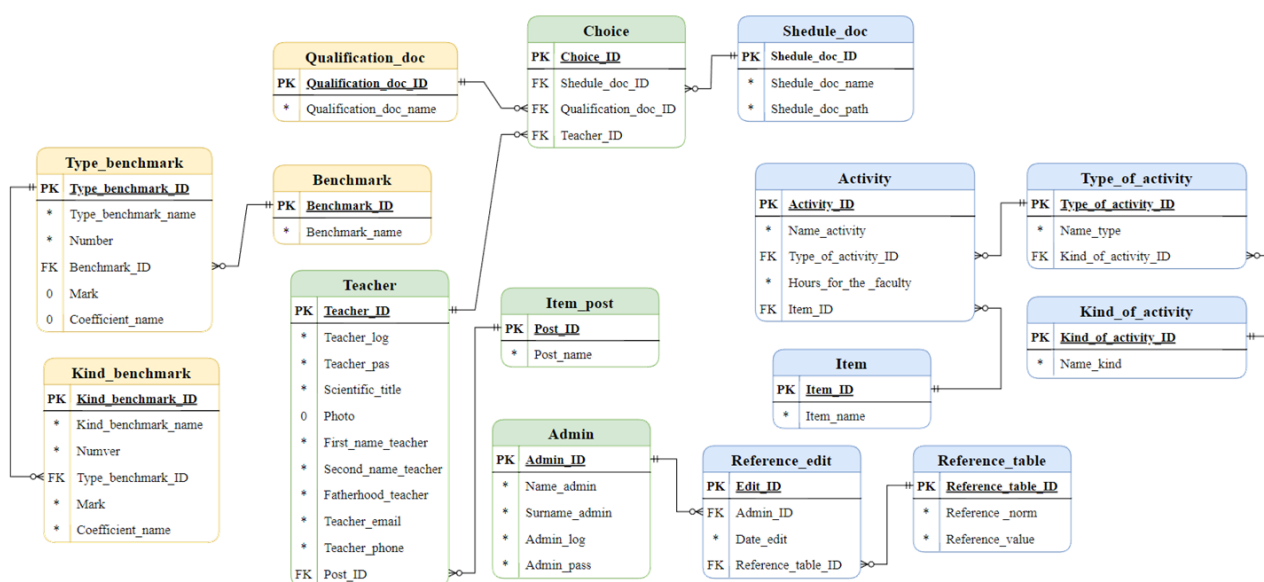


Рисунок 2.4 – Логічна модель бази даних для вебдодатку ведення документації

Для розгортання бази даних обрано вебдодаток PhpMyAdmin. Фізична модель бази даних усього вебдодатку представлена на рисунку 2.5.



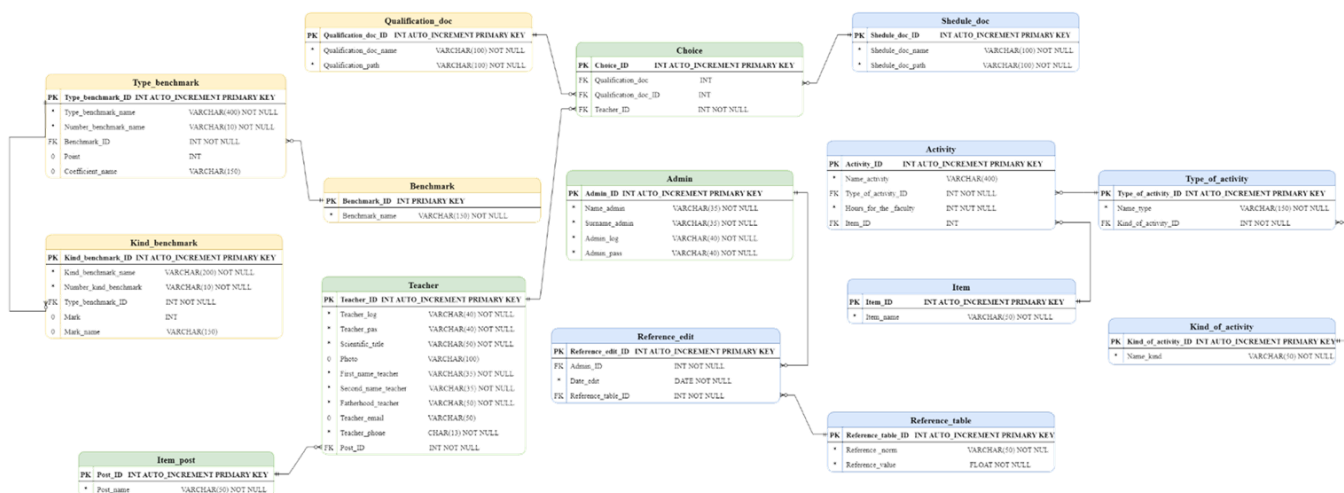


Рисунок 2.4 –База даних вебдодатку ведення документації

Для модулю формування звітності про підвищення кваліфікації викладачем використовуються 8 таблиць, а саме:

- **Qualification\_doc** – сутність для збереження посилань на згенеровані звіти та їх назви;

- **Choice** – сутність для об'єднання вебдодатку модуля формування звітності про підвищення кваліфікації та модулю планування позааудиторної роботи викладача;

- **Benchmark** – зберігає назви усіх видів контрольних показників накопичувальної системи, які взяті з нормативного документу СумДУ [7];

- **Type\_benchmark** – зберігає назви, номери та бали для обрахунків для усіх підтипів контрольних показників;

- **Kind\_benchmark** – зберігає дані про підтипи контрольних показників.

- **Teacher** – сутність для збереження інформації про викладачів, їх логін, пароль, пошта, посада, наукова ступінь, номер телефону, ПІБ та фото;

- **Item\_post** – усі посади;

- **Admin** – дані про адміністратора вебдодатку, його пароль, логін, прізвище та ім'я;

Фізична модель бази даних модулю формування звітності про підвищення кваліфікації викладача зображена на рисунку 2.5.

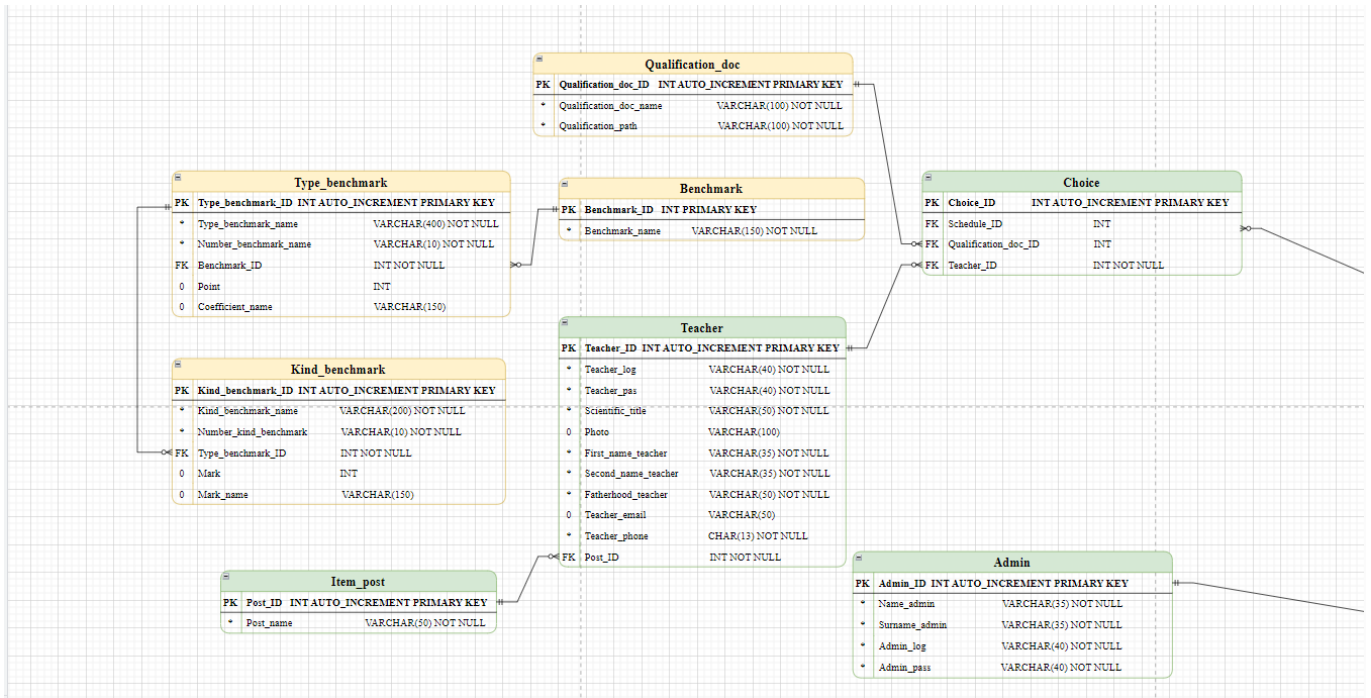


Рисунок 2.5 – База даних для модуля формування звітності про підвищення кваліфікації викладача

## 3 РОЗРОБКА ВЕБДОДАТКУ ФОРМУВАННЯ ЗВІТНОСТІ ПРО ПІДВИЩЕННЯ КВАЛІФІКАЦІЇ ВИКЛАДАЧА

### 3.1 Архітектура вебдодатку

Архітектура програмного продукту визначає взаємодію між різними компонентами та модулями вебдодатку. В основі будь-якого вебдодатку лежить модель клієнт-сервер. У цій моделі клієнт взаємодіє з сервером через Інтернет. На сервері розміщується вебдодаток, він обробляє запити клієнта і надсилає відповіді. Клієнтська частина відповідає за рендеринг користувацького інтерфейсу та забезпечення безперебійної роботи користувача [14].

На рисунку 3.1 зображена клієнт-серверна архітектура розробленого вебдодатку ведення документації викладачем, яка включає в себе два модуля: модуль формування звітності з підвищення кваліфікації та модуль планування позааудиторної роботи викладача. Архітектура складається з трьох основних компонентів: сервер бази даних, вебсервер і робоча станція користувача. Кожен з цих компонентів виконує свою роль у загальній системі, забезпечуючи взаємодію між собою та з користувачем.

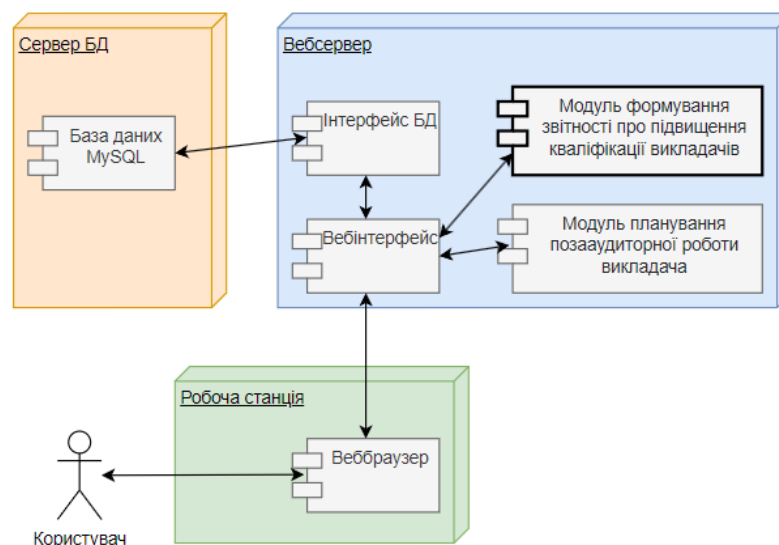


Рисунок 3.1 – Архітектура вебдодатку ведення документації викладачем

Сервер бази даних містить базу даних MySQL, яка використовується для зберігання всіх даних вебдодатку (дані про викладачів, усі контрольні показники накопичувальної системи, звіти про підвищення кваліфікації та звіти з позааудиторної діяльності). Сервер бази даних забезпечує зберігання, управління та доступ до цих даних.

Вебсервер виконує роль посередника між базою даних та користувачем. Він обробляє запити від робочої станції, звертається до бази даних, отримує необхідну інформацію та повертає її користувачеві. Вебсервер включає в себе кілька компонентів:

- інтерфейс БД, який відповідає за взаємодію з базою даних. Він приймає запити від вебінтерфейсу, виконує їх у базі даних та передає результати назад.
- вебінтерфейс, який обробляє запити від користувачів, отримані через веббраузер. Він формує запити до бази даних через інтерфейс БД.
- модуль формування звітності про підвищення кваліфікації викладачів, який дозволяє викладачам створювати та зберігати звіти про їхню професійну підготовку та підвищення кваліфікації.
- модуль планування позааудиторної роботи викладача.

Робоча станція представляє комп'ютер або мобільний пристрій викладача, який використовує вебдодаток. Взаємодія з вебдодатком здійснюється через веббраузер, що забезпечує доступ до всіх функцій системи.

## 3.2 Програмна реалізація

Для початку був розроблений дизайн вебдодатку, для цього використовувався програмний продукт Figma. Дизайн розроблювався посилаючись на офіційний сайт Сумського державного університету, кафедри інформаційні технологій, за вимогами технічного завдання (додаток А). На рисунку 3.2 зображені макети вебдодатку.

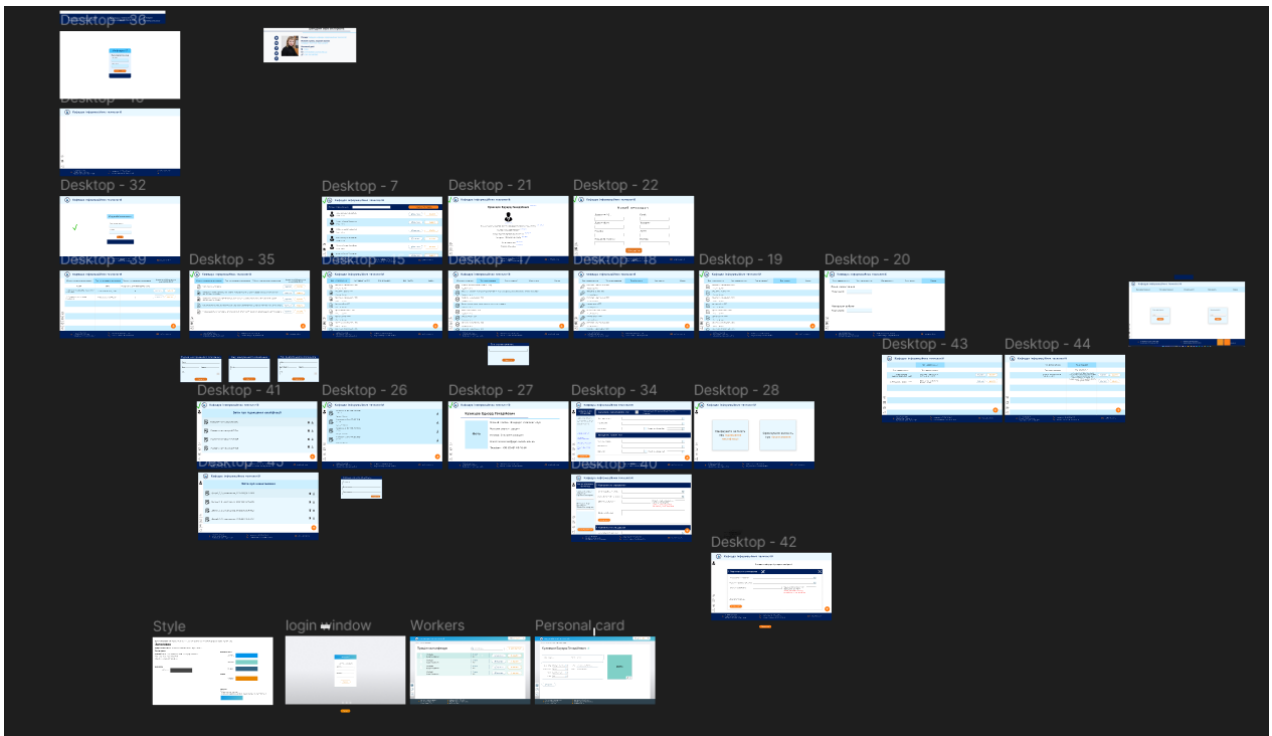


Рисунок 3.2 – Макети вебдодатку ведення документації

Після цього розроблювався інтерфейс вебдодатку за допомогою верстки, структура вебдодатку зроблена мовою розмітки HTML, а стиль каскадними таблицями CSS. Розпочато роботу з верхньої та нижньої частини вебдодатку, так як ці частини є на кожній сторінці вебдодатку. На рисунку 3.3 зображено розмітку верхньої частини вебдодатку, а на рисунку 3.4 частинку низу вебдодатку. Середовищем для програмування обрано програмний додаток Visual Studio Code.

```

<!-- Хедер -->
<header class="header">
  <div class="img"></div>
  <p>Кафедра інформаційних технологій</p>
</header>

```

Рисунок 3.3 – Верхня частина вебдодатку

```

117 <!-- Футер -->
118 <footer class="footer">
119   <div class="location">
120     <p class="imgloc">
121       
122     </p>
123     <p class="adres">
124       вул. Харківська, 116, Суми, Україна, 40007 Головний корпус, 13-й
125       поверх, ауд. Г1303
126     </p>
127   </div>
128

```

Рисунок 3.4 – Нижня частина вебдодатку

Наступним кроком є версткою усіх сторінок, це сторінка авторизації, перегляду звітів, формування звітів, перегляд персональної інформації та усіх сторінок адміністративної панелі (видів, типів та підтипів контрольних показників, редагування інформації про викладачів і т. д.). На рисунку 3.5 зображено зверстану сторінку авторизації.

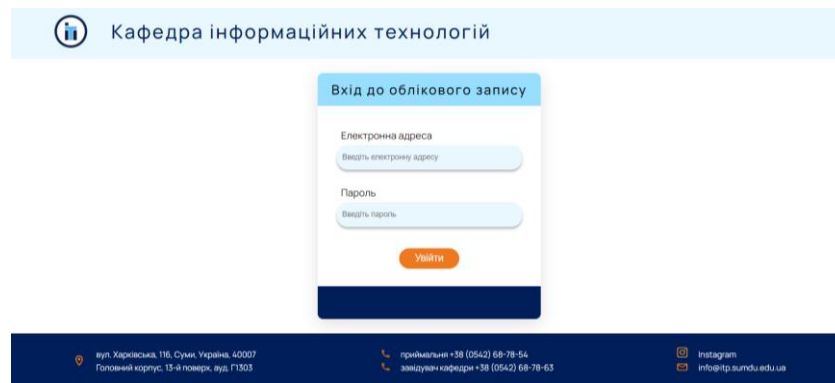


Рисунок 3.5 – Готова сторінка авторизації

Після закінчення фронтенду потрібно розгорнути базу даних, скрипт якої знаходиться в додатку Г. На рисунку 3.6 зображено перелік реалізованих таблиць бази даних. Заповнення відбувається або з адміністративної панелі адміністратором вебдодатку, або безпосередньо в базі даних розробником. Дані про контрольні показники беруться з нормативу СумДУ [7]. Викладачі заповнюються відповідно штату кафедри інформаційних технологій.

Таблиця	Действие	Строки	Тип	Сравнение	Размер	Фрагментировано
Activity	☆	3	InnoDB	utf8mb4_unicode_ci	48.0 КиБ	-
Admin	☆	2	InnoDB	utf8mb4_unicode_ci	16.0 КиБ	-
Benchmark	☆	5	InnoDB	utf8mb4_unicode_ci	16.0 КиБ	-
Choice	☆	1	InnoDB	utf8mb4_unicode_ci	64.0 КиБ	-
Item	☆	4	InnoDB	utf8mb4_unicode_ci	16.0 КиБ	-
Item_post	☆	5	InnoDB	utf8mb4_unicode_ci	16.0 КиБ	-
Kind_benchmark	☆	7	InnoDB	utf8mb4_unicode_ci	32.0 КиБ	-
Kind_of_activity	☆	4	InnoDB	utf8mb4_unicode_ci	16.0 КиБ	-
Qualification_doc	☆	1	InnoDB	utf8mb4_unicode_ci	16.0 КиБ	-
Reference_edit	☆	0	InnoDB	utf8mb4_unicode_ci	48.0 КиБ	-
Reference_table	☆	2	InnoDB	utf8mb4_unicode_ci	16.0 КиБ	-
Schedule_doc	☆	0	InnoDB	utf8mb4_unicode_ci	16.0 КиБ	-
Teacher	☆	5	InnoDB	utf8mb4_unicode_ci	32.0 КиБ	-
Type_benchmark	☆	19	InnoDB	utf8mb4_unicode_ci	32.0 КиБ	-
Type_of_activity	☆	2	InnoDB	utf8mb4_unicode_ci	32.0 КиБ	-
<b>15 таблиц</b>	<b>Всего</b>	<b>60</b>	<b>InnoDB</b>	<b>utf8mb4_unicode_ci</b>	<b>416.0 КиБ</b>	<b>0 Байт</b>

Рисунок 3.6 – Перелік реалізованих таблиць бази даних  
db\_maintaining\_documentation

Далі усім сторінкам, які це потребують, потрібно додати інтерактивності за допомогою мови програмування JavaScript. На прикладі розглянемо сторінку формування звітності. Для того щоб користувачу було усе зрозуміло та зручно, продумано спеціальний дизайн з кнопками додавання нових форм для заповнення інформацією про підвищення кваліфікації викладача, щоб згенерувати звіт. На рисунку 3.7 зображено початкова сторінка формування звітності.

Рисунок 3.7 – Початкова сторінка формування звітності

Для того щоб працювала кнопка додавання контрольних показників написано скрипт, частина коду зображена на рисунку 3.8.

```

592 // Створення нового блоку
593 const newBlock = document.createElement("div");
594 newBlock.className = "formreport";
595
596 newBlock.innerHTML = `
597 <div class="form">
598   <form method="post">
599     <div class="name">
600       <input type="text" name="readonlyInput" value="${selectedOption}"
601         readonly style="border:none; font-size:inherit;">
602       <button class="del">
603         
604       </button>
605     </div>
606     <div class="d">
607       <div class="form-group">
608         <label> Тип контрольного показника </label>
609         <!-- Placeholder for selectElement -->
610         <div id="selectContainer"></div>

```

Рисунок 3.8 – Фрагмент коду додавання блоку з формою для ведення даних

При натисканні на кнопку додавання використовується функція `addEventListener()`, яка відкриває модальне вікно, яке у весь час скрите. Модельне вікно виводить усі види контрольних показників, які є в базі даних (рис. 3.9). Після підтвердження додається блок форми для ведення даних, для цього використовується також функція `addEventListener()` та вставляється блок (рис. 3.8) в контейнер, який прописаний на рисунку 3.10.

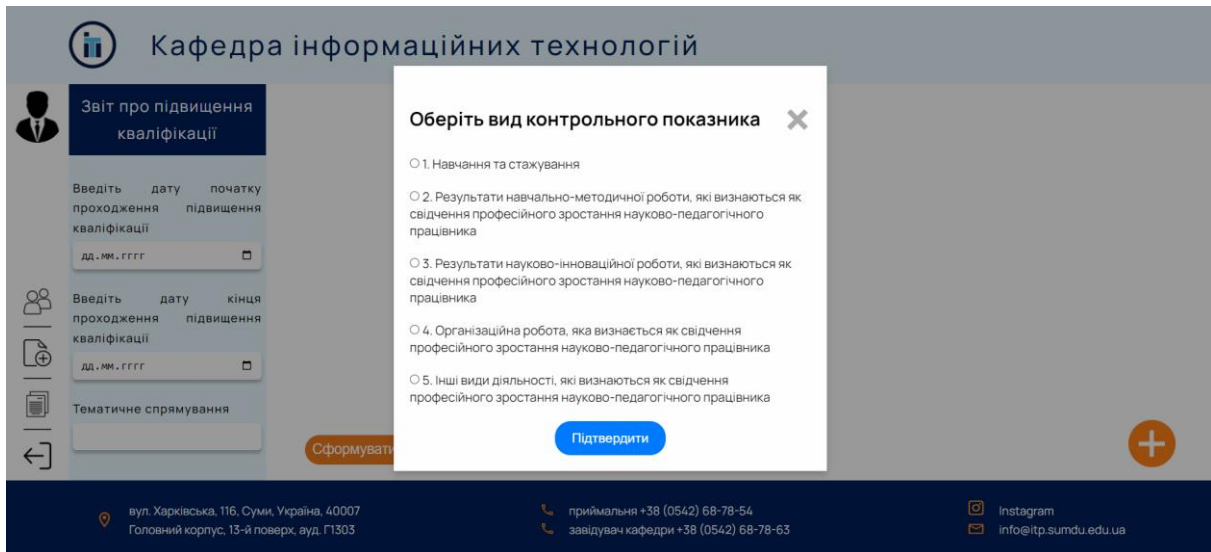


Рисунок 3.9 – Модальне вікно вибору контрольного показника

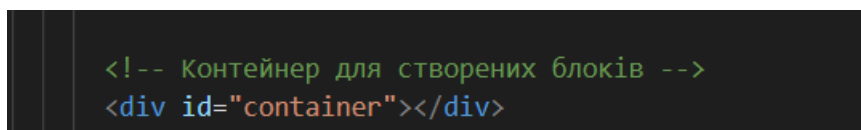


Рисунок 3.10 – Контейнер для додавання блоків з формами

Додавання в контейнер блоків відбувається за допомогою функції `appendChild()`. Для того щоб працювало видалення блоків при натисканні на хрестик використовується фрагмент коду, який зображено на рисунку 3.11.

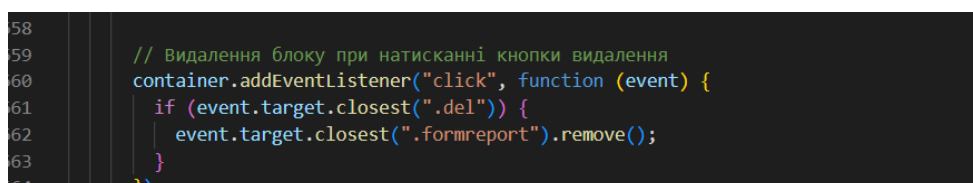
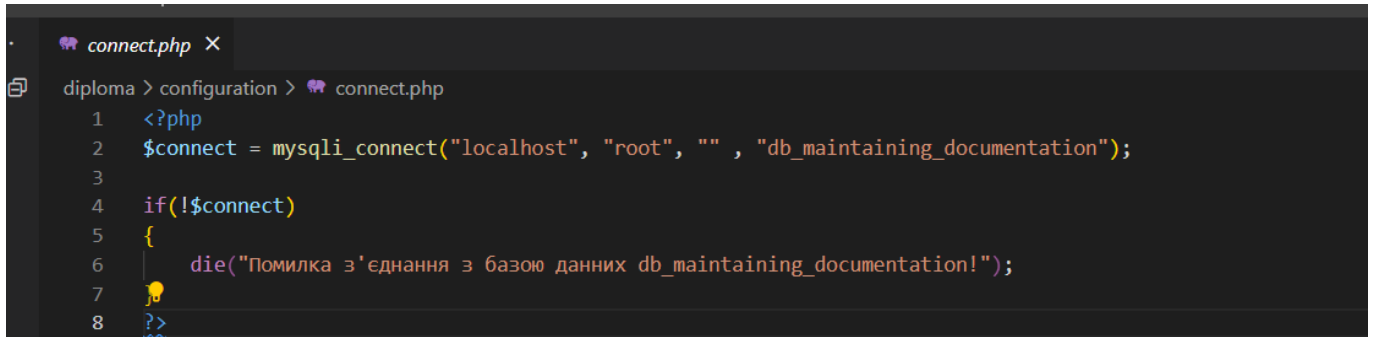


Рисунок 3.11 – Видалення блоків



Як було сказано раніше, інформація в модельне вікно виводиться з бази даних і для цього потрібно підключитись до неї, це вже початок бекенду. Скрипт підключення до бази даних зображено на рисунку 3.11.



```

connect.php ×
diploma > configuration > connect.php
1  <?php
2  $connect = mysqli_connect("localhost", "root", "", "db_maintaining_documentation");
3
4  if(!$connect)
5  {
6      die("Помилка з'єднання з базою даних db_maintaining_documentation!");
7
8  ?>

```

Рисунок 3.11 – Підключення до бази даних

Після авторизації починається сесія і таким чином зрозуміло хто саме використовує вебдодаток, приклад початку сесії зображено на рисунку 3.12



```

1  <?php
2  session_start();
3  require_once "../configuration/connect.php";
4
5  $email = $_POST["email"];
6  $password = $_POST["password"];
7
8  if($_POST["email"] != "" && $_POST["password"] != "")
9  {
10     $check_user = mysqli_query($connect, "SELECT * FROM `Teacher` WHERE `Teacher_log` = '$email' AND `Teacher_pas` = '$password'");
11     if (mysqli_num_rows($check_user) > 0)
12     {
13         $user = mysqli_fetch_assoc($check_user);
14         $_SESSION["user"] =
15         [
16             'Teacher_ID' => $user ['Teacher_ID'],
17             'scientific_title' => $user ['Scientific_title'],
18             'Photo' => $user ['Photo'],
19             'First_name_teacher'=> $user ['First_name_teacher'],
20             'Second_name_teacher'=> $user ['Second_name_teacher'],
21             'Fatherhood_teacher'=> $user ['Fatherhood_teacher'],
22             'Teacher_email'=> $user ['Teacher_email'],
23             'Teacher_phone'=> $user ['Teacher_phone'],
24             'Post_ID'=> $user ['Post_ID']
25         ];
26         header("Location: choice.php");
27     }
28 }

```

Рисунок 3.12 – Авторизація, початок сесії

Основною задачею модулю формування звітності про підвищення кваліфікації є звіт, для цього використовується бібліотека docxtemplater. Спочатку був розроблений шаблон звіту (Template.docx), в який вставлено змінні та масив, який

зберігає усі зібрані дані з усіх форм, що додавав викладач. На рисунку 3.13 зображено шаблон звіту, а на рисунку 3.14 рендеринг засобами JavaScript.

Шаблон за версією 04  
Затверджено наказом  
ректора СуУДП  
№ 0572-1 від 01.07.2020 р.

Начальник ЦРКП

" \_\_\_\_\_ " \_\_\_\_\_ 2023 р.

**ЗВІТ ПРО РЕЗУЛЬТАТИ ПІДВИЩЕННЯ КВАЛІФІКАЦІЇ  
ЗА НАКОПИЧУВАЛЬНОЮ СИСТЕМОЮ**  
{surname} {name} {fatherhood}

{post}, кафедра інформаційних технологій

звітний період \_\_\_\_ {date\_start} \_\_\_\_ року – \_\_\_\_ {date\_end} \_\_\_\_ року

Тематичне спрямування<sup>1)</sup> {thematicfocus}

№ з)	Контрольний показник	Результати оброблення, бали	
		Пропозиція <sup>3)</sup>	визнання результатів <sup>4)</sup>
{#ArrayData2} {mainKey}			
{typeNumbers}	{type} {subtype} {additionalData} {points} * {points_type} = {mark}	{mark}	{/ArrayData2}
Всього за контрольними показниками		{sum}	

ПОГОДЖЕНО:  
Декан факультету ЕдІТ. \_\_\_\_\_ **Юрій ВОЛК**  
(підпис)

Науково-педагогічний працівник \_\_\_\_\_ {name} {surname}.  
(підпис)

Рисунок 3.13 – Шаблон звіту

```

59 });
60 // Render the document
61 doc.render({
62
63   name: `${name2}`,
64   surname: `${secondname2}`,
65   fatherhood: `${fatherhood2}`,
66   post: `${post2.Post_name}`,
67   date_start: `${formattedDate}`,
68   date_end: `${formattedDate2}`,
69   thematicfocus: `${thematicFocus}`,
70   sum: `${sum}`,
71
72   ArrayData2: ArrayData2
73
74 });

```

Рисунок 3.14 – Шаблон звіту

Для зберігання звітів на сервері використовується запит за допомогою функції `fetch()`, спочатку файл відправляється на сервер, потім за допомогою серверної мови програмування `php` виконуються усі дії з цим файлом (зберігання в потрібну папку на сервері, збереження шляху та назви файлу в базу даних). Також файл зберігається і

на локальному комп'ютері викладача за допомогою функції saveAs() на клієнтській частині.

Для захисту сторінок, передбачена перевірка сесій, якщо викладач почистив кеш сесія одразу завершиться, але за замовчуванням викладач залишиться на сторінках вебдодатку неавторизований, для цього на рисунку 3.15 зображена перевірка, яка забезпечить повторну авторизацію, для безпеки даних.

```
// для безпеки сторінки, тільки для викладачів
if(!isset($_SESSION["user"]))
{
    header("Location: authorization.php");
}
?>
```

Рисунок 3.15 – Перевірка сесії для викладача

Також реалізовано адміністративну панель для ведення, редагування та видалення даних про контрольні показники накопичувальної системи СумДУ та викладачів. Увесь код однотипний, тому розглянемо лише одну сторінку. На рисунку 3.16 зображено вивід усіх викладачів із бази даних.

```
<?php foreach($teachers as $teacher): ?>
<div class="doc">
  <div class="namedoc">
    
    <p><?php echo $teacher['Second_name_teacher'] . ' ' . $teacher['First_name_teacher'] . ' ' . $teacher['Fatherhood_teacher'];
  </div>
  <div class="action">
    <div class="knop1">
      <a href="person_information_edit.html">
        <button type="edit">Редагувати</button>
      </a>
    </div>
    <div class="knop2">
      <button type="dell">Видалити</button>
    </div>
  </div>
</div>
```

Рисунок 3.16 – Вивід усіх викладачів із бази даних в адміністративну панель

На рисунку 3.17 зображено вивід із бази даних підтипів контрольних показників для редагування або виведення.

Вид контрольного показника			Тип контрольного показника		Підтип контрольного показника
Назва підтипу	Номер підтипу	Назва типу	Коефіцієнт для розрахунку балу	Назва коефіцієнту для введення	Дія
в іноземних університетах і наукових установах, що входять до TOP-500 рейтингу QS	1.4.1	1.4 Участь у програмах академічної мобільності з метою підвищення кваліфікації, викладання за кордоном, виконання спільних проєктів та наукове співробітництво	6	Кількість годин, зазначених у відповідному документі	Редагувати Видалити

Рисунок 3.17 – Вивід підтипів із бази даних в адміністративну панель

Останнім кроком є адаптивність сайту для мобільних пристроїв, для цього використовуються медіазапити в каскадних таблицях стилів css. Вигляд адаптивності (рис. 3.18).

Рисунок 3.18 – Вигляд вебдодатку для телефонів

У додатку В надано лістинг вебдодатку формування звітності про підвищення кваліфікації. Також у додатку Г надано скрипт для розгортання бази даних.

### 3.3 Використання програмного додатку

Для початку викладачу потрібно авторизуватися, щоб почати користуватися вебдодатком. Після успішної авторизації викладач може переглянути вже сформовані звіти, які зберігаються на сервері, та завантажити їх або видалити. Також може сформувати новий звіт та переглянути персональну інформацію (рис. 3.19).

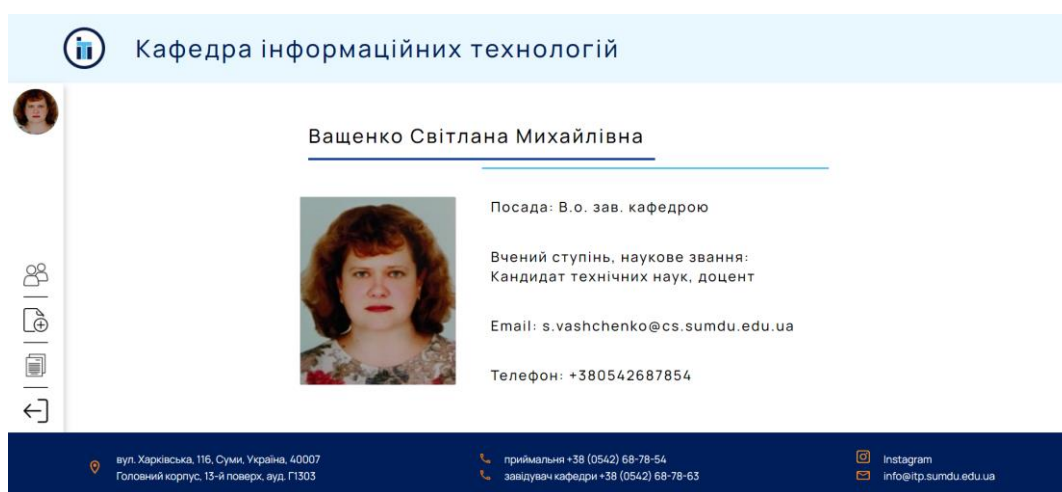


Рисунок 3.19 – Персональна інформація авторизованого викладача

Щоб сформувати звіт потрібно перейти на сторінку, яка зображена на рисунку 3.21. Для цього потрібно натиснути на кнопку «сформувати звітність про підвищення кваліфікації» (рис. 3.20).



Рисунок 3.20 – Вивід підтипів із бази даних в адміністративну панель

Рисунок 3.21 – Сторінка формування звітності про підвищення кваліфікації

Викладач в модальному вікні (рис. 3.9) обирає вид контрольного показника, потім натискає «підтвердити» та з'являється новий блок з формою. Тип в списках залежить від обраного виду контрольного показника, а підтип від типу (не завжди є). Поле ведення для числових показників з'являється відповідна від обраного типу чи підтипу. Додавати таких форм можна безліч. Потім обов'язково потрібно заповнити дату початку, кінця підвищення кваліфікації та тематичне спрямування. Після цього викладач натискає на кнопку «Сформувати звіт» та отримує звіт з підвищення кваліфікації з розширенням .docx., який автоматично завантажується в «Завантаження» на локальний комп'ютер викладача. На рисунку 3.22 зображено яку назву має згенерований звіт. Назва залежить від того, хто генерує звіт.

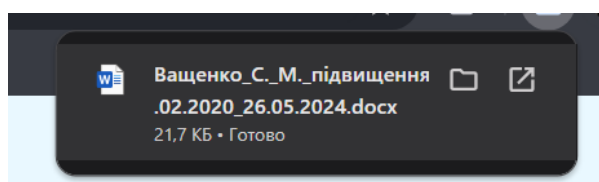


Рисунок 3.22 – Назва згенерованого звіту

Вигляд згенерованого звіту на рисунках 3.23-3.24.

Шаблон за версією 04  
Затверджено наказом  
ректора ЄУАІТ  
№ 0372-І від 01.07.2020 р.

Начальник ЦРКП

\_\_\_\_\_ (підпис) \_\_\_\_\_ (літ'я, прізвище)  
" \_\_\_\_\_ " \_\_\_\_\_ 2023 р.

**ЗВІТ ПРО РЕЗУЛЬТАТИ ПІДВИЩЕННЯ КВАЛІФІКАЦІЇ  
ЗА НАКОПИЧУВАЛЬНОЮ СИСТЕМОЮ**

Вашенко Світлана Михайлівна

В.о. зав. кафедрою, кафедра інформаційних технологій

звітний період \_\_\_\_ 04.02.2020 \_\_\_\_ року – \_\_\_\_ 26.05.2024 \_\_\_\_ року

Тематичне спрямування<sup>1)</sup> інформаційні технології

№ з)	Контрольний показник	Результати обрахування, бали	
		Пропозиція <sup>2)</sup>	визнання результатів <sup>3)</sup>
<b>1. Навчання та стажування</b>			
1.1	Навчання за програмою підвищення кваліфікації  «IT інструменти для викладачів» від компанії GlobalLogic; липень 2023 р., сертифікат CS № 12048/2023, 18 годин; 100*6=600	600	
<b>2. Результати навчально-методичної роботи, які визнаються як свідчення професійного зростання науково-педагогічного працівника</b>			
2.2	Керівництво студентом, який здобув призери ІІ етапу Міжнародної студентської олімпіади, Всеукраїнської студентської олімпіади або Всеукраїнського конкурсу, студентських наукових робіт	200	

Рисунок 3.23 – Згенерований звіт, перша частина

	Керівництво студентом, який здобув призове місце на І етапі (внутрішньовузівському) Всеукраїнської студентської олімпіади Палажченко С.В., Шкура А. В., 2019-2020 р.; Всеукраїнська студентська олімпіада за напрямом підготовки «Комп'ютерні науки»  2*100=200		
Всього за контрольними показниками		800	

ПОГОДЖЕНО:  
Декан факультету ЄдІТ

\_\_\_\_\_ (підпис) Юрій ВОЛК

Науково-педагогічний працівник

\_\_\_\_\_ (підпис) Світлана ВАЩЕНКО

Рисунок 3.24 – Згенерований звіт, друга частина

Також викладач може переглядати, завантажувати та видаляти звіти на сторінці «Переглянути звіти про підвищення кваліфікації» (рис. 3.25).



Рисунок 3.25 – Перегляд звітів

Для адміністратора також потрібно авторизуватися. Приклад використання адміністративної панелі на рисунках 3.26-3.27. Додавання нового викладача. Пароль шифрується за допомогою функції `md5()` в `php`.

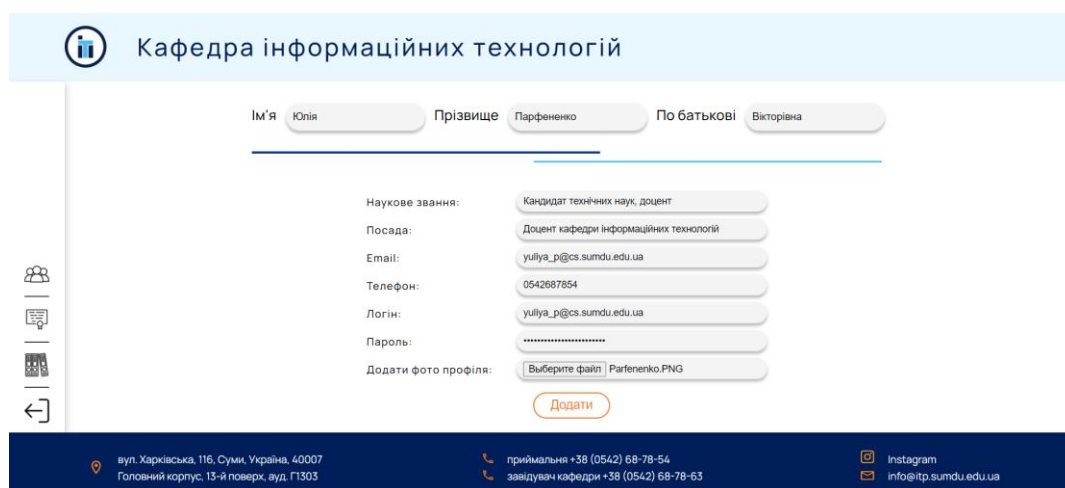


Рисунок 3.26 – Додавання нового викладача

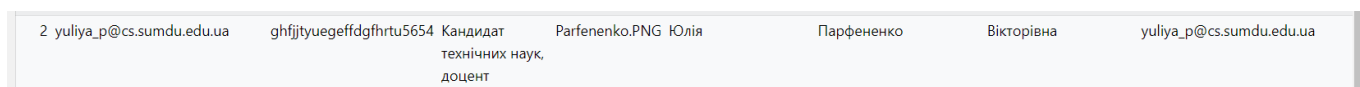


Рисунок 3.27 – Додавання успішно виконано

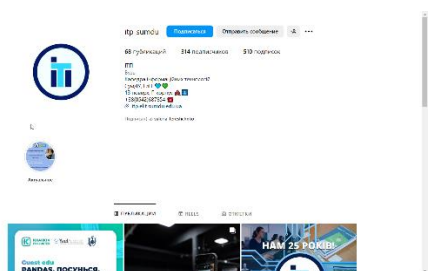
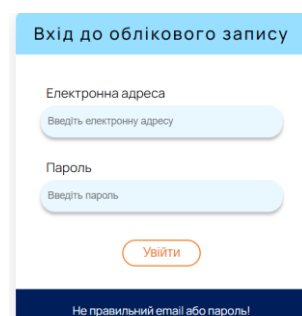


Також можна редагувати та видаляти інформацію з бази даних в адміністративній панелі, на сторінці з викладачами реалізовано пошук по викладачам.

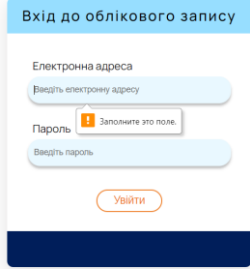
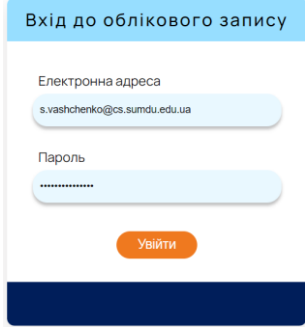


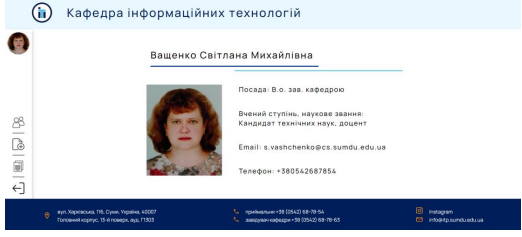
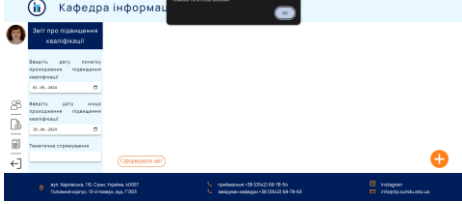
### 3.4 Тестування вебдодатку

Для перевірки працездатності вебдодатку проведено тестування методом білого ящика. Таке тестування як підходить саме для розробників, так як він бачить свій код та може одразу внести зміни, якщо буде знайдено баг. У таблиці 3.1 наведені результати тестування.

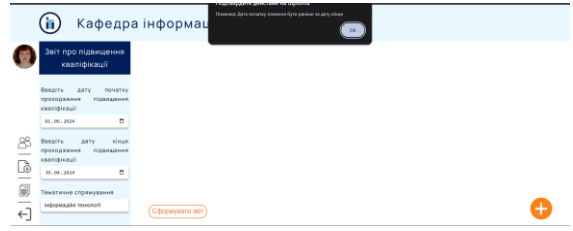
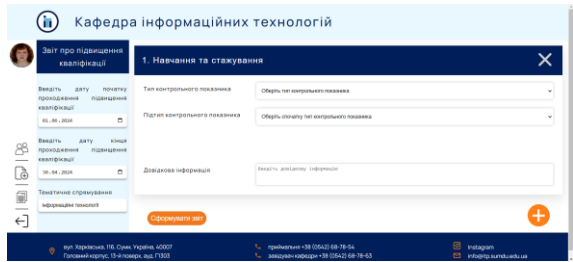
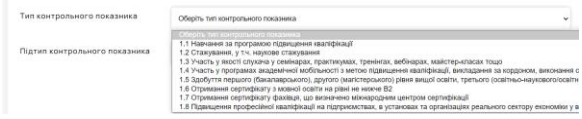
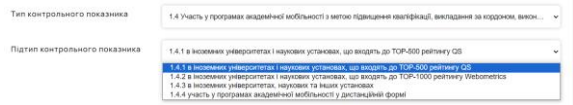
Таблиця 3.1 – Тестування методом білого ящика

№ тесту	Умова, яка тестується	Очікуваний результат	Фактичний результат	0/1
1	Клікабельність в футері на сторінку інстаграма.	Перехід на сторінку інстаграма кафедри інформаційних технологій.	<p>При натисканні на посилання інстаграму відбувся перехід на сторінку, в новій вкладці.</p> 	1
2	Введення невалідних даних в форму авторизації.	Повідомлення про не правильний пароль або пошту.	<p>Повідомлення відбулося.</p> 	1

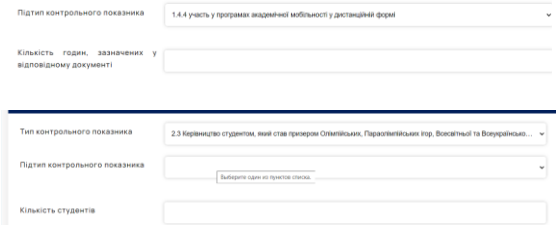
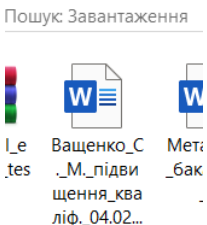
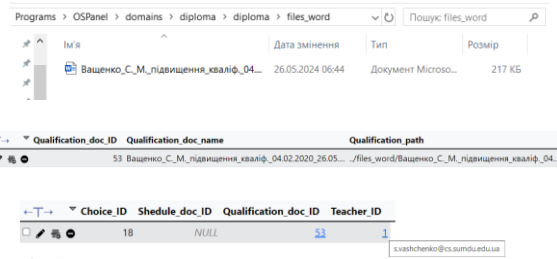
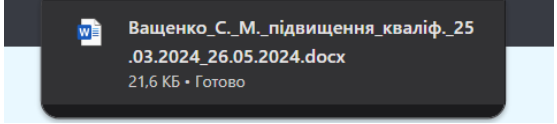
Продовження табл. 3.1

№ тесту	Умова, яка тестується	Очікуваний результат	Фактичний результат	0/1
3	Відправка пустої форми.	Повідомлення про заповнення поля.		1
4	Видимість паролю.	Пароль повинен бути прихований.		1
5	Введення валідних даних в форму для авторизації.	Перехід на сторінку вебдодатку для викладачів.	Перехід відбувся. 	1
6	Відображення фотографії викладача, який авторизувався.	Фото відповідає авторизованому викладачу.	Фото відповідає авторизованому викладачу. 	1
7	Профіль викладача.	Виведення правильної інформації авторизованого викладача.		1
8	Відправлення пустої форми на генерування звіту.	Повідомлення про незаповнені поля.		1

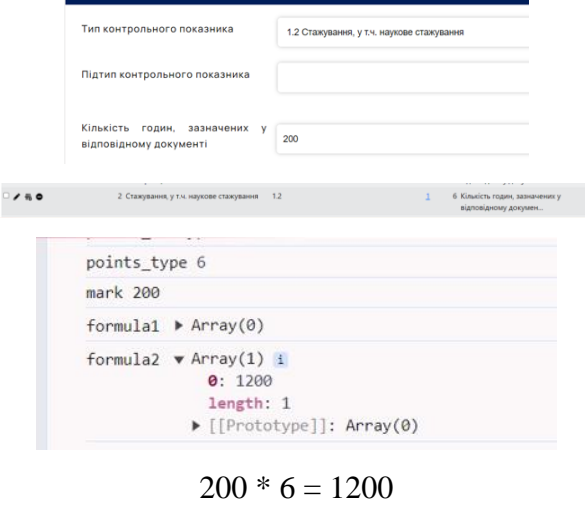

Продовження табл. 3.1

№ тесту	Умова, яка тестується	Очікуваний результат	Фактичний результат	0/1
9	Заповнення дати початку більшої за дату кінця підвищення кваліфікації.	Повідомлення про введення некаретних даних.		1
10	Натискання на кнопку додавання форм на сторінці формування звітності.	Додавання нового блоку з формою з відповідним видом контрольного показника обраним викладачем в модальному вікні.		1
11	Список типів контрольних показників.	Виведення усіх типів контрольних показників з бази даних, які відносяться до обраного виду.		1
12	Список підтипів контрольних показників.	Виведення усіх підтипів (якщо вони є) контрольних показників з бази даних, які відносяться до обраного типу.		1

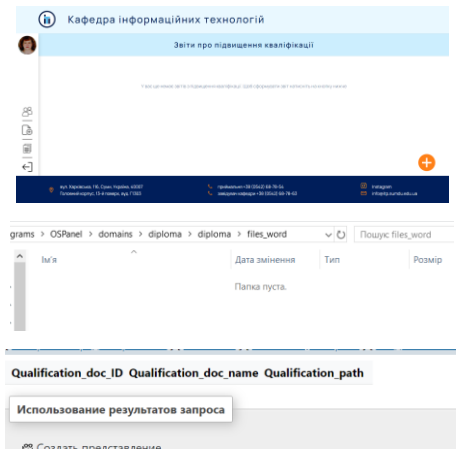
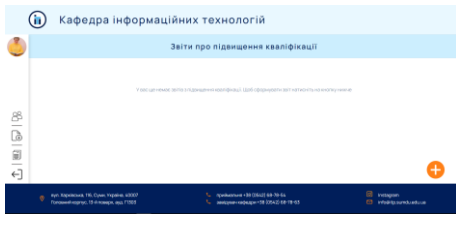
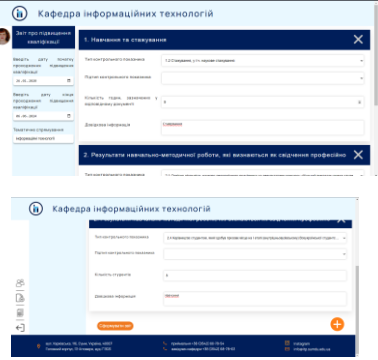
Продовження табл. 3.1

№ тесту	Умова, яка тестується	Очікуваний результат	Фактичний результат	0/1
13	Лейбл числові дані.	Поява введення числових даних (якщо вони є) відповідно від вибору типу або підтипу контрольних показників.		1
14	Збереження сформованого звіту локальному пристрої викладача.	Збереження сформованого звіту локальному пристрої викладача.		1
15	Збереження сформованого звіту на сервері та в бд.	Збереження сформованого звіту на сервері в папці file_word та в бд шлях.		1
16	Збереження сформованого звіту з назвою ім'ям авторизованого користувача та датами початку та кінця підвищення кваліфікації.	Збереження сформованого звіту з назвою ім'ям авторизованого користувача та датами початку та кінця підвищення кваліфікації.		1

Продовження табл. 3.1

№ тесту	Умова, яка тестується	Очікуваний результат	Фактичний результат	0/1
17	Обрахунок балів за контрольні показники.	Для типу 1.2 «Стажування» в базі даних значення коефіцієнта 6, ведення 200, очікуваний результат $200 * 6 = 1200$ .		1
18	Перегляд сформованих звітів.	Перелік усіх звітів, які сформовані авторизованим викладачем та які зберігаються на сервері.		1
19	Завантаження звітів.	Після натискання на кнопку завантажити звіт, звіт завантажується в «Завантаження» на локальному пристрою авторизованого викладача.	Завантаження відбулося.	1

Продовження табл. 3.1

№ тесту	Умова, яка тестується	Очікуваний результат	Фактичний результат	0/1																										
20	Видалення звітів.	Після натискання видалення, видалення звіту з бд та серверу.	Видалення відбулося. 	1																										
21	Відсутність звітів.	Якщо звітів немає у авторизованого викладача, тоді повинно бути повідомлення про це.		1																										
22	Формування звітності.	Дані з форми повинні співпадати з згенерованим звітом.	 <p>В.о. зав. кафедрою, кафедра інформаційних технологій</p> <p>звітний період 26.01.2020 року – 06.06.2024 року</p> <p>Тематичне співручання<sup>а)</sup> інформаційні технології</p> <table border="1"> <thead> <tr> <th rowspan="2">№ з/п</th> <th rowspan="2">Контрольний показник</th> <th colspan="2">Результати обчислення, балів</th> </tr> <tr> <th>Пропозиція<sup>б)</sup></th> <th>визначені результати<sup>в)</sup></th> </tr> </thead> <tbody> <tr> <td colspan="4"><b>1. Навчання та стажування</b></td> </tr> <tr> <td>1.2</td> <td>Стажування, у тч наукове стажування Стажувальника 9*6-54</td> <td>54</td> <td></td> </tr> <tr> <td colspan="4"><b>2. Результати навчально-методичної роботи, які визнаються як свідчення професійного зростання науково-педагогічного працівника</b></td> </tr> <tr> <td>2.4</td> <td>Керівництво студентами, які здобув призове місце на I етапі (внутрішньнауковий) Всеукраїнської студентської олімпіади або Всеукраїнського конкурсу студентських наукових робіт Навчальця 8*20-160</td> <td>160</td> <td></td> </tr> <tr> <td colspan="2">Всього за контрольними показниками</td> <td>214</td> <td></td> </tr> </tbody> </table> <p>ПОГОДЖЕНО:</p>	№ з/п	Контрольний показник	Результати обчислення, балів		Пропозиція <sup>б)</sup>	визначені результати <sup>в)</sup>	<b>1. Навчання та стажування</b>				1.2	Стажування, у тч наукове стажування Стажувальника 9*6-54	54		<b>2. Результати навчально-методичної роботи, які визнаються як свідчення професійного зростання науково-педагогічного працівника</b>				2.4	Керівництво студентами, які здобув призове місце на I етапі (внутрішньнауковий) Всеукраїнської студентської олімпіади або Всеукраїнського конкурсу студентських наукових робіт Навчальця 8*20-160	160		Всього за контрольними показниками		214		1
№ з/п	Контрольний показник	Результати обчислення, балів																												
		Пропозиція <sup>б)</sup>	визначені результати <sup>в)</sup>																											
<b>1. Навчання та стажування</b>																														
1.2	Стажування, у тч наукове стажування Стажувальника 9*6-54	54																												
<b>2. Результати навчально-методичної роботи, які визнаються як свідчення професійного зростання науково-педагогічного працівника</b>																														
2.4	Керівництво студентами, які здобув призове місце на I етапі (внутрішньнауковий) Всеукраїнської студентської олімпіади або Всеукраїнського конкурсу студентських наукових робіт Навчальця 8*20-160	160																												
Всього за контрольними показниками		214																												

Проведено тестування основних функцій модулю формування звітності про підвищення кваліфікації користувацької частини, помилок не виявлено.

## ВИСНОВКИ

У результаті кваліфікаційної роботи бакалавра став вебдодаток формування звітності про підвищення кваліфікації викладача.

Під час виконання даної роботи проведено аналіз процесу оформлення звітності щодо підвищення кваліфікації викладачів Сумського державного університету за накопичувальною системою.

За результатами аналізу вебдодатків-аналогів виявлено, що ці сайти не мають функціоналу, який потрібно реалізувати у створюваному вебдодатку, а саме автоматизоване формування звітності про підвищення кваліфікації викладачів або обрахунок суми контрольних показників накопичувальної системи. Однак, вони відповідають іншим напрямкам та надають корисні освітні послуги у своїх галузях.

Отже, результати аналізу підтвердили актуальність розробки вебдодатку для формування звітності про підвищення кваліфікації викладачів. Були сформульовані вимоги до моделі, постановка задачі, розроблено структуру вебдодатку та макети веб-сторінок.

Обрано засоби реалізації вебдодатку, для створення каркасу веб-сторінок обрано HTML, для надання веб-сторінкам візуальних ефектів та адаптивності обрано каскадні таблиці стилів CSS, для надання динамічності обрано скриптову мову програмування JavaScript, для створення бекенду обрано мову програмування PHP, для реалізації бази даних розроблюваного вебдодатку обрано СУБД MySQL та для генерації звітів бібліотеку docxtemplater.

Проведено планування робіт та проаналізовано можливі ризики при виконання проекту.

Також було виконано проектування вебдодатку та його бази даних. У ході цього процесу було створено кілька важливих діаграм для кращого розуміння структури і функціональності системи. Перш за все, було розроблено контекстну діаграму бізнес-процесів, яка відображає загальну картину взаємодії між основними процесами

вебдодатку в нотації IDEF0. Далі було розроблено діаграму першого рівня декомпозиції IDEF0, що деталізує ці взаємодії.

Крім того, створено діаграму варіантів використання, яка ілюструє основні сценарії взаємодії користувачів з вебдодатком. У процесі проектування бази даних було розроблено як логічну, так і фізичну моделі.

Розроблено діаграму архітектури, яка відображає взаємозв'язок між компонентами системи. На діаграмі зображено клієнт-серверну архітектуру, яка чітко показує взаємодію між клієнтською частиною, серверною частиною та базою даних.

Проведено тестування основних функцій модулю формування звітності про підвищення кваліфікації користувацької частини, помилок не виявлено.

Застосування цієї розробки дозволить значно скоротити час, який витрачається на формування звітності вручну, тим самим полегшуючи роботу викладачів. Це визначає практичну значимість представленої роботи.



## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Про освіту : Закон України від 05.09.2017 р. № 2145-VIII : станом на 24 берез. 2024 р. URL: <https://zakon.rada.gov.ua/laws/show/2145-19#Text> (дата звернення: 30.04.2024).
2. Деякі питання підвищення кваліфікації педагогічних і науково-педагогічних працівників : Постанова Каб. Міністрів України від 21.08.2019 р. № 800 : станом на 31 груд. 2019 р. URL: <https://zakon.rada.gov.ua/laws/show/800-2019-п#Text> (дата звернення: 30.04.2024).
3. Бодя М. С. Декілька причин, чому важливо підвищувати кваліфікацію вчителю. Всеосвіта. URL: <https://vseosvita.ua/blogs/dekilka-prychyn-chomu-vazhlyvo-pidvyshchuvaty-kvalifikatsiiu-vchyteliu-85484.html> (дата звернення: 30.04.2024).
4. Особливості підвищення кваліфікації педагогічних працівників. Знайшов. URL: [https://znaishov.com/News/Details/osoblyvosti\\_pidvyshchennia\\_kvalifikatsii\\_pedagogichnykh\\_pratsivnykiv](https://znaishov.com/News/Details/osoblyvosti_pidvyshchennia_kvalifikatsii_pedagogichnykh_pratsivnykiv) (дата звернення: 30.04.2024).
5. Деякі питання професійного розвитку науково-педагогічних працівників : Наказ від 04.12.2020 р. № 1504. URL: <https://mon.gov.ua/storage/app/uploads/public/5fd/8e5/df0/5fd8e5df039fc332033916.pdf> (дата звернення: 30.04.2024).
6. Положення про підвищення кваліфікації співробітників СумДУ, педагогічних та науково-педагогічних працівників інших закладів освіти у Сумському державному університеті – Сумський державний університет. URL: <https://normative.sumdu.edu.ua/?task=getfile&tmpl=component&id=ba7d76bc-742d-e211-86ae-001a4be6d04a&kind=1&version=1714416145167> (дата звернення: 28.04.2024).
7. Контрольні показники накопичувальної системи обрахунку основних результатів підвищення кваліфікації науково-педагогічних працівників Сумського державного університету – Сумський державний університет. URL: <https://normative.sumdu.edu.ua/?task=getfile&tmpl=component&id=a0d38502->

2cc5-e311-aea9-001a4be6d04a&kind=1&version=1714416145167 (дата звернення: 30.04.2024).

8. Підвищення кваліфікації для вчителів. Харківський національний університет імені В. Н. Каразіна | Каразінський університет. URL: <https://karazin.ua/fakulteti-ta-instituti/institut-pisliadiplomnoyi-osviti-ta-zaochnogo-distantciinogo/pidvyshchennia-kvalifikatsii-dlia-vchyteliv-i-vykladachiv-za/> (дата звернення: 30.04.2024).

9. Розширений пошук ППК. Портал для пошуку та запису на програми підвищення кваліфікації (ППК) для вчителів. URL: <https://ppk.isuo.org/ppk/advanced-search-ppk> (дата звернення: 30.04.2024).

10. Освітні послуги. Підвищення кваліфікації. Інститут модернізації змісту освіти. URL: <https://imzo.gov.ua/diyalnist/osvitni-posluhy-pidvyshchennia-kvalifikatsii/> (дата звернення: 30.04.2024).

11. Gallagher N. Modelling Strategic Processes for a Secondary Experienced Client in the Service Contract Environment : Professional Doctorate. 2023. 370 p.

12. PROCEEDINGS OF 2019 INTERNATIONAL CONFERENCE ON INFORMATION & COMMUNICATION TECHNOLOGY AND SYSTEMS (ICTS). Surabaya, Indonesia : Department of Informatics Faculty of Information and Communication Technology Institut Teknologi Sepuluh Nopember (ITS), 2019. 7 p.

13. Navathe S. B., Elmasri R. Fundamentals of Database Systems, Global Edition. Pearson, 2016. 1272 p.

14. Висоцький А. В. ПОНЯТТЯ КЛІЄНТ-СЕРВІСНОЇ АРХІТЕКТУРИ. м. Дніпро, 3–4 груд. 2020 р. URL: [http://dspace.puet.edu.ua/bitstream/123456789/10007/1/\(Part%201\)%20Conference%20Results%202020.pdf#page=233](http://dspace.puet.edu.ua/bitstream/123456789/10007/1/(Part%201)%20Conference%20Results%202020.pdf#page=233) (дата звернення: 15.05.2024).

## ДОДАТОК А

### ТЕХНІЧНЕ ЗАВДАННЯ

на розробку інформаційної системи  
«Вебдодаток ведення документації викладачем.  
Модуль формування звітності про підвищення  
кваліфікації викладача»

### ПОГОДЖЕНО:

Доцент кафедри інформаційних технологій

\_\_\_\_\_ Ващенко С. М.

Студентка групи ІТ-02

\_\_\_\_\_ Терещенко В. С.

Суми 2024

## **1. Призначення й мета створення вебдодатку**

### **1.1 Призначення вебдодатку**

Вебдодаток призначений для ведення документації викладачем, а саме формування звітності про підвищення кваліфікації за накопичувальною системою.

### **1.2 Мета створення вебдодатку**

Мета даної кваліфікаційної роботи є розробка вебдодатку автоматизації формування звітності про підвищення кваліфікації викладача кафедри інформаційних технологій, Сумського державного університету та спрощення роботи науково-педагогічних працівників.

### **1.3 Цільова аудиторія**

Цільовою аудиторією даного проекту є викладачі кафедри інформаційних технологій Сумського державного університету, які проходять процедуру підвищення кваліфікації кожні п'ять років.

## **2 Вимоги до вебдодатку**

### **2.1 Вимоги до проекту в цілому**

#### **2.1.1 Вимоги до структури й функціонування вебдодатку**

Вебдодаток формування звітності про підвищення кваліфікації викладача повинен бути реалізований за допомогою веб-інструментів та забезпечувати визначений набір функціональних можливостей.

Кінцевий продукт даного проекту має бути представлений вебдодатком, який містить якісне інформаційне наповнення.

#### **2.1.2 Вимоги до персоналу**

Персонал університету не повинен мати особливих технічних навичок для роботи з вебдодатком і його підтримкою. Єдиною вимогою є наявність навичок користування персональним комп'ютером та веб-браузером.

#### **2.1.3 Вимоги до збереження інформації**

Уся інформація надана у вебдодатку повинна зберігатися у базі даних реалізованій засобами системи управління базами даних MySQL.

## **2.1.4 Вимоги до розмежування доступу**

Розроблюваний вебдодаток має бути загальнодоступним у мережі Інтернет. Права доступу до інформації розмежовані за групами користувачів: адміністратор та користувач.

Адміністратор має необмежений доступ до даних з правами перегляду, додавання, редагування та видалення. Також займається додаванням та видаленням користувачів із бази даних. Доступ до адміністративної панелі надається за спеціальним логіном та паролем, який потрібно вводити в форму авторизації (входу).

Користувач (викладач) може переглядати власні дані, а також має доступ до сторінок, пов'язаних з заповненням своєї звітності та завантаження файлів. Не може редагувати чи переглядати інформацію інших викладачів. Щоб користуватися вебдодатком, потрібно авторизуватися, логін та пароль надає адміністратор сайту.

## **2.2 Структура вебдодатку**

### **2.2.1 Загальна інформація про структуру вебдодатку**

До структури вебдодатку входять усі його веб-сторінки та адміністративна панель для користування працівником кафедри інформаційних технологій, Сумського державного університету.

Перелік сторінок вебдодатку наступний:

- сторінка «Авторизація» слугує для авторизації викладача або адміністратора вебдодатку;
- сторінка «Головна» слугує для вибору сформувати нову звітність про підвищення кваліфікації викладача або про навантаження викладача;

- сторінка «Персональні дані» слугує для перегляду даних викладача, який авторизувався;
- сторінка «Перегляд звітів» слугує для перегляду та завантаження викладачем звітів з підвищення кваліфікації минулих років.
- сторінка «Формування звітності» слугує для заповнення пунктів викладачами, які потрібні для звіту та формування файлу формату .doc, який можна завантажити на свій комп'ютер;
- адміністративна панель слугує для додавання, редагування та видалення усієї інформації, яка є в базі даних.

### **2.2.2 Навігаційне меню**

Для зручної навігації повинно бути створене меню, що забезпечить швидке переміщення користувача по всім доступним сторінкам вебдодатку. Меню має бути закріплене і розташоване зліва на кожній сторінці, окрім авторизації.

### **2.2.3 Наповнення вебдодатку (контент)**

Управління контентом вебдодатку має здійснюватися за допомогою адміністративної панелі. Усе інформаційне наповнення вебдодатку має міститися у базі даних. Графічні матеріали та інформацію для наповнення надає Замовник.

### **2.2.4 Дизайн та структура вебдодатку**

Дизайн вебдодатку має бути виконаний в єдиному стилі оформлення (використання блакитних, білих, синіх та помаранчевих кольорів), мінімалізм та функціональне розміщення інформації, зручний перегляд інформації, інтерактивне користування вебдодатком. Також дизайн розробленого вебдодатку повинен співпадати з дизайном головного сайту кафедри інформаційних технологій, Сумського державного університету. Усі фото мають бути чіткими та якісними.

Вебдодаток має бути адаптивним для його перегляду з більшості девайсів: персонального комп'ютеру, телефону, ноутбуку та ін.

Види і розміри шрифтів повинні бути комфортними для перегляду користувачами. Навігаційні елементи повинні забезпечувати однозначне розуміння користувачами їх змісту. Дизайн вебсайту має коректно відображатись в наступних основних браузерях: останні версії Mozilla, Firefox, Chrome, Safari.

Шаблон майбутнього вебдодатку зображено на рисунках А.1-А.6.





Рисунок А.1 – Схема сторінки авторизації



Рисунок А.2 – Схема головної сторінки

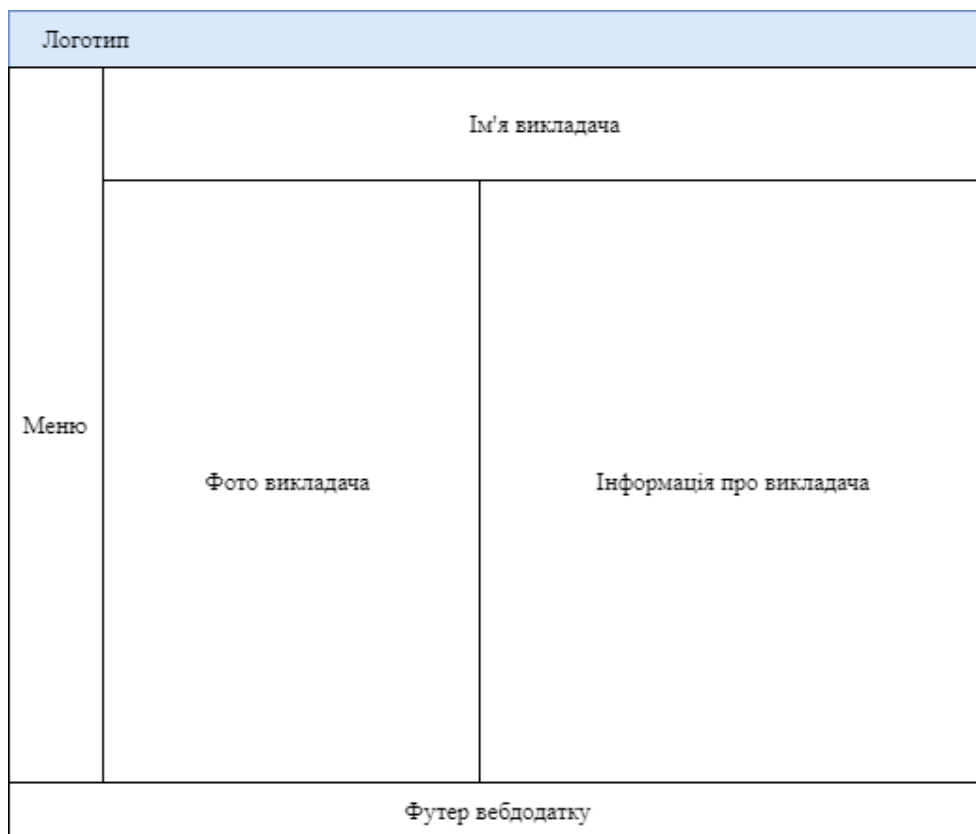


Рисунок А.3 – Схема сторінки персональної інформації викладача

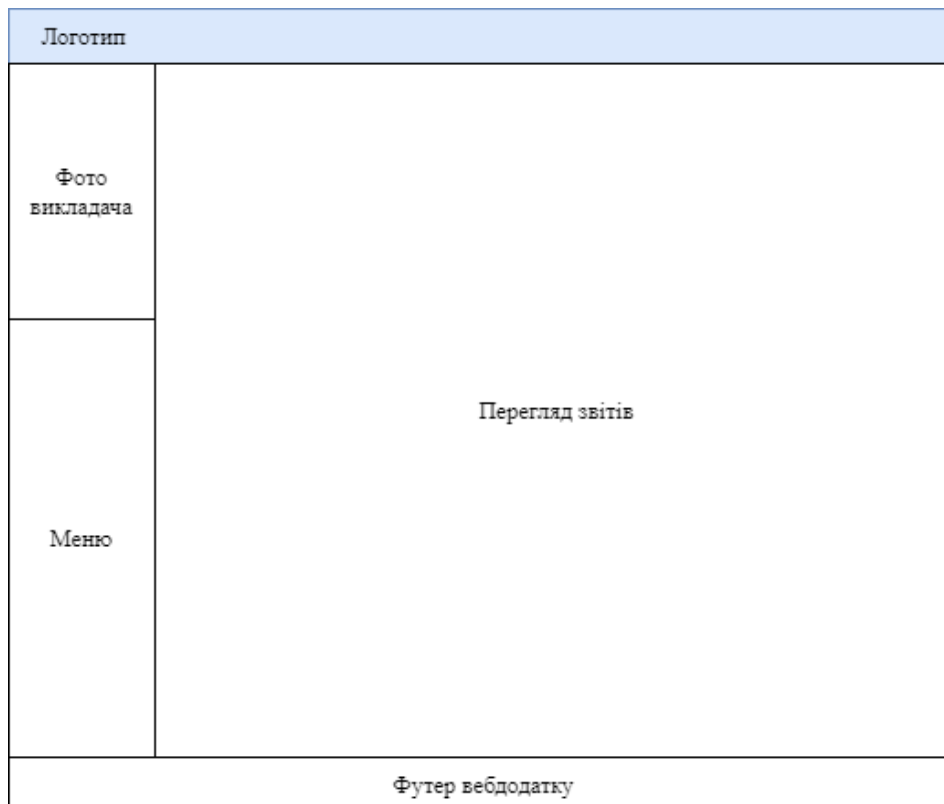


Рисунок А.4 – Схема сторінки перегляду усіх звітів про підвищення кваліфікації викладача

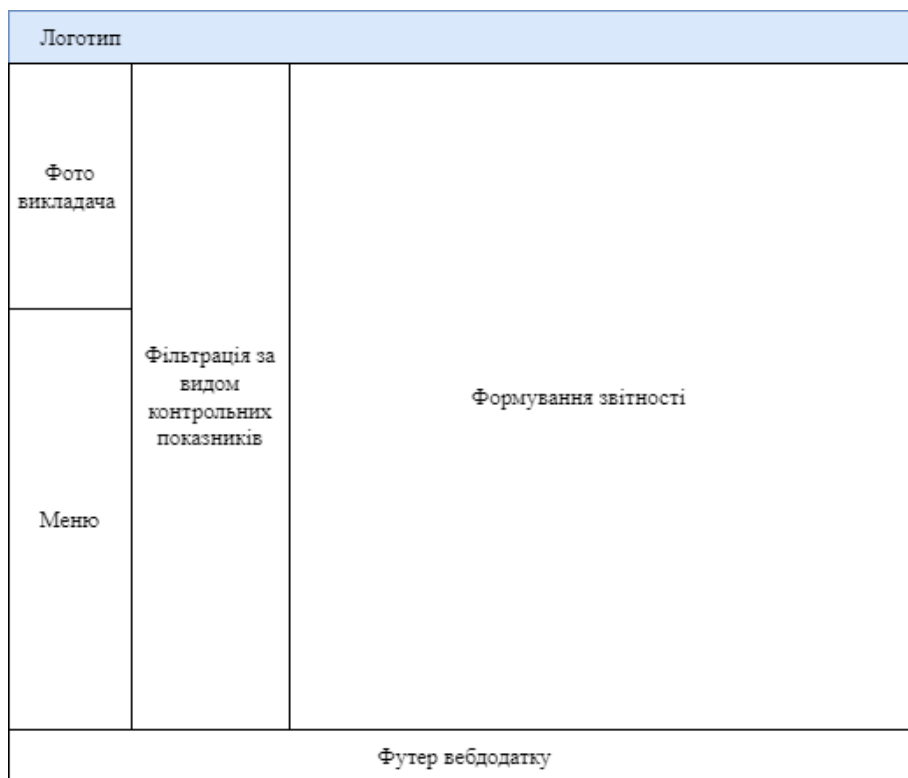


Рисунок А.5 – Схема сторінки формування звітності про підвищення кваліфікації викладача

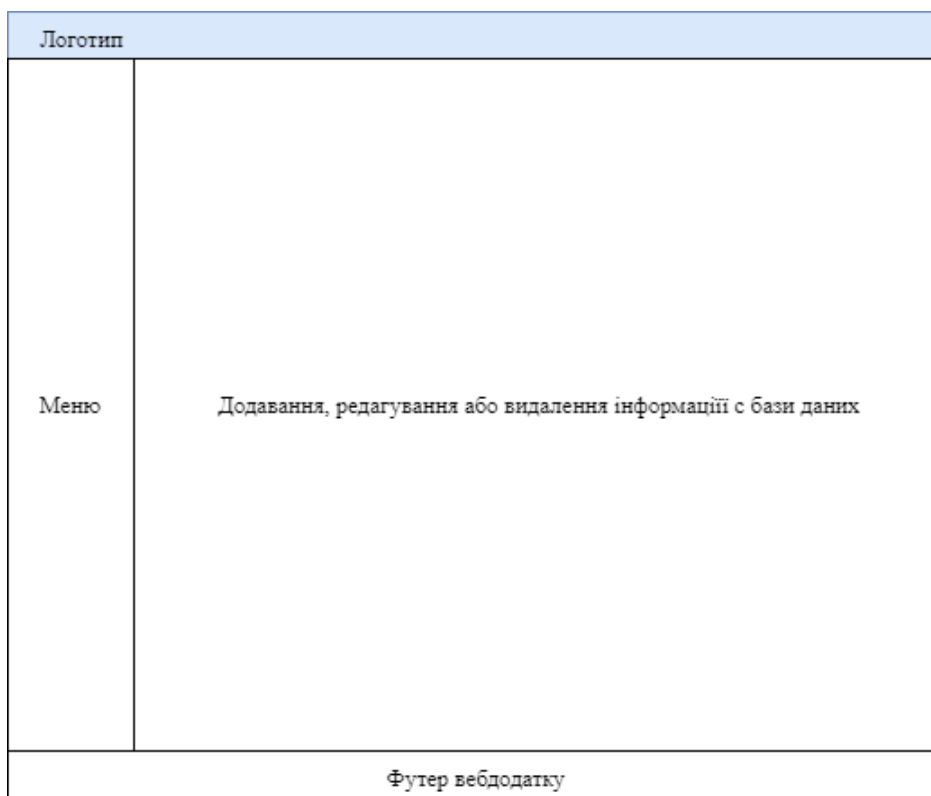


Рисунок А.6 – Схема адміністративної сторінки

Макети сторінок вебдодатку наведені нижче на рисунках А.7 – А.13.

The screenshot shows a light blue header with the department's logo and name: "Кафедра інформаційних технологій". Below the header is a white login form titled "Вхід до облікового запису". The form contains two input fields: "Електронна адреса" and "Пароль", followed by an orange "Вхід" button. At the bottom of the page is a dark blue footer with contact information: "Сумський державний університет, вул. Римського-Корсакова, 2, м. Суми, Україна, 40007", phone numbers for the reception and department, and the email "opr@sumdu.edu.ua".

Рисунок А.7 – Макет сторінки авторизації

The screenshot shows a light blue header with the department's logo and name: "Кафедра інформаційних технологій". On the left side, there is a vertical sidebar with a user profile icon at the top and four navigation icons below. The main content area features two large light blue buttons: "Сформувати звітність про підвищення кваліфікації" and "Сформувати звітність про навантаження". At the bottom is a dark blue footer with the same contact information as in Figure A.7.

Рисунок А.8 – Макет головної сторінки вебдодатку

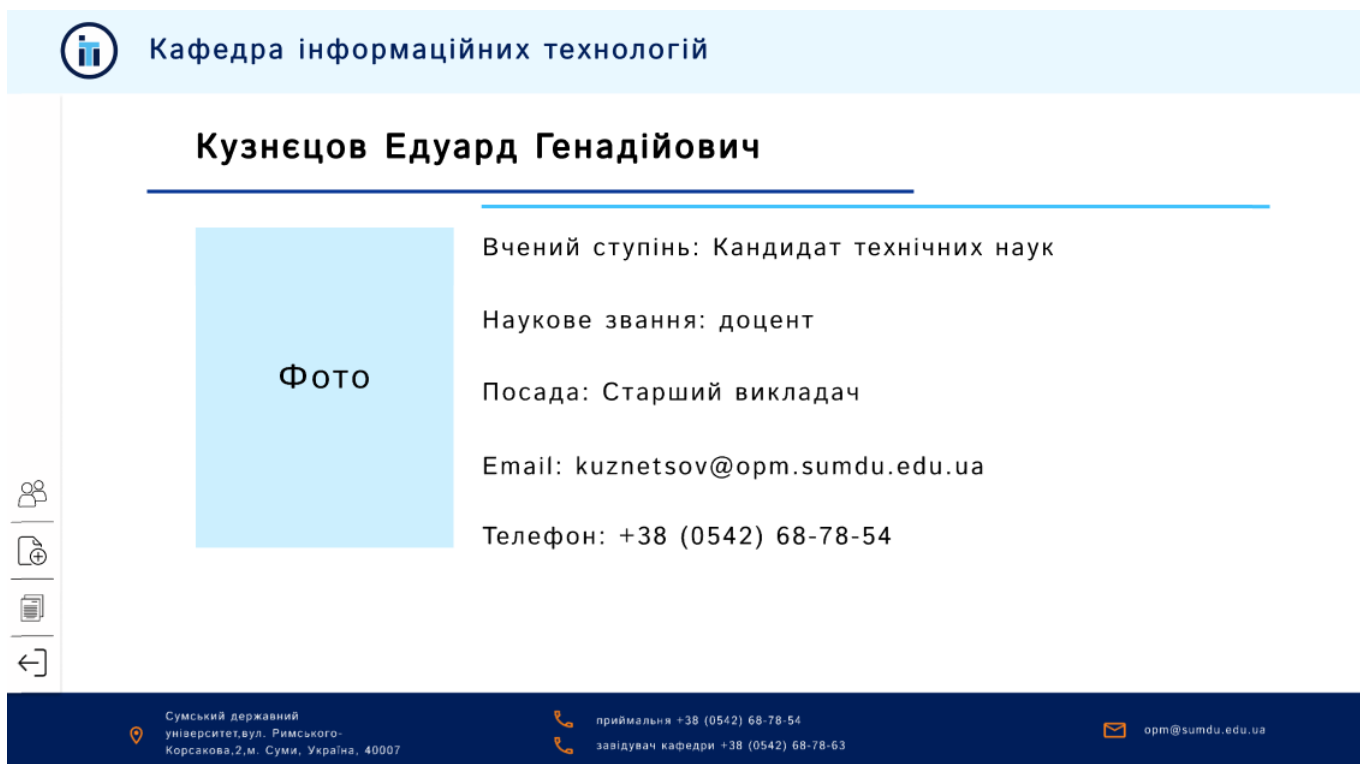



Рисунок А.9 – Макет сторінки «Персональна інформація»



Рисунок А.10 – Макет сторінки «Перегляд звітів»

 Кафедра інформаційних технологій

**Звіт про підвищення кваліфікації**

Навчання та стажування

Результати навчально-методичної роботи, які визнаються як свідчення професійного зростання науково-педагогічного працівника

Результати науково-інноваційної роботи, які визнаються як свідчення професійного зростання науково-педагогічного працівника

Організаційна робота, яка визнається як свідчення професійного зростання науково-педагогічного працівника

Інші види діяльності, які визнаються як свідчення професійного зростання науково-педагогічного працівника

**Зберегти**

### 1 Навчання та стажування

Тип контрольного показника

Підтип контрольного показника

Тематичне спрямування  Кількість годин, зазначених у відповідному документі   
\*назва залежить від вибору типу/підтипу контр. показника

Довідкова інформація

**Додати файл**

### 1 Навчання та стажування


Тип контрольного показника

Сумський державний університет, вул. Римського-Корсакова, 2, м. Суми, Україна, 40007


приймальня +38 (0542) 68-78-54  
завідувач кафедри +38 (0542) 68-78-63


opt@sumdu.edu.ua


Рисунок А.10 – Макет сторінки «Формування звітності»


 Кафедра інформаційних технологій


Пошук працівника  **+ Додати викладача**


 Кузнєцов Едуард Генадійович  
Ставка: 0.2, 0.5 **РЕДАГУВАТИ** **ВИДАЛИТИ**

 Кузнєцов Едуард Генадійович  
Ставка: 0.2, 0.5 **РЕДАГУВАТИ** **ВИДАЛИТИ**

 Кузнєцов Едуард Генадійович  
Ставка: 0.2, 0.5 **РЕДАГУВАТИ** **ВИДАЛИТИ**

 Кузнєцов Едуард Генадійович  
Ставка: 0.2, 0.5 **РЕДАГУВАТИ** **ВИДАЛИТИ**

 Кузнєцов Едуард Генадійович  
Ставка: 0.2, 0.5 **РЕДАГУВАТИ** **ВИДАЛИТИ**

 Кузнєцов Едуард Генадійович  
Ставка: 0.2, 0.5 **РЕДАГУВАТИ** **ВИДАЛИТИ**



Сумський державний університет, вул. Римського-Корсакова, 2, м. Суми, Україна, 40007


приймальня +38 (0542) 68-78-54  
завідувач кафедри +38 (0542) 68-78-63

opt@sumdu.edu.ua

Рисунок А.11 – Макет адміністративної панелі, управління користувачами вебдодатку

 Кафедра інформаційних технологій

Вид контрольного показника	Тип контрольного показника	Підтип контрольного показника	Назва коефіцієнта для розрахунку балу
 1 Навчання та стажування			<input type="button" value="РЕДАГУВАТИ"/> <input type="button" value="ВИДАЛИТИ"/>
 2 Результати навчально-методичної роботи, які визнаються як свідчення професійного зростання науково-педагогічного працівника			<input type="button" value="РЕДАГУВАТИ"/> <input type="button" value="ВИДАЛИТИ"/>
 3 Результати науково-інноваційної роботи, які визнаються як свідчення професійного зростання науково-педагогічного працівника			<input type="button" value="РЕДАГУВАТИ"/> <input type="button" value="ВИДАЛИТИ"/>
 4 Організаційна робота, яка визнається як свідчення професійного зростання науково-педагогічного працівника			<input type="button" value="РЕДАГУВАТИ"/> <input type="button" value="ВИДАЛИТИ"/>
 5 Інші види діяльності, які визнаються як свідчення професійного зростання науково-педагогічного працівника			<input type="button" value="РЕДАГУВАТИ"/> <input type="button" value="ВИДАЛИТИ"/>

Сумський державний університет, вул. Римського-Корсакова, 2, м. Суми, Україна, 40007  приймальня +38 (0542) 68-78-54  завідувач кафедри +38 (0542) 68-78-63  oipm@sumdu.edu.ua

Рисунок А.12 – Макет адміністративної панелі, редагування видів контрольних показників

 Кафедра інформаційних технологій

Вид контрольного показника	Тип контрольного показника	Підтип контрольного показника	Назва коефіцієнта для розрахунку балу
Назва	Вид	Коефіцієнт для розрахунку балу	
1.1 Навчання за програмою підвищення кваліфікації	1 Навчання та стажування	6	<input type="button" value="РЕДАГУВАТИ"/> <input type="button" value="ВИДАЛИТИ"/>
1.2 Стажування, у т.ч. наукове стажування	1 Навчання та стажування	6	<input type="button" value="РЕДАГУВАТИ"/> <input type="button" value="ВИДАЛИТИ"/>
			

Сумський державний університет, вул. Римського-Корсакова, 2, м. Суми, Україна, 40007  приймальня +38 (0542) 68-78-54  завідувач кафедри +38 (0542) 68-78-63  oipm@sumdu.edu.ua

Рисунок А.13 – Макет адміністративної панелі, редагування типів контрольних показників

## 2.2.4 Система навігації (карта вебдодатку)

Карта вебдодатку зображена на рисунку А.14.

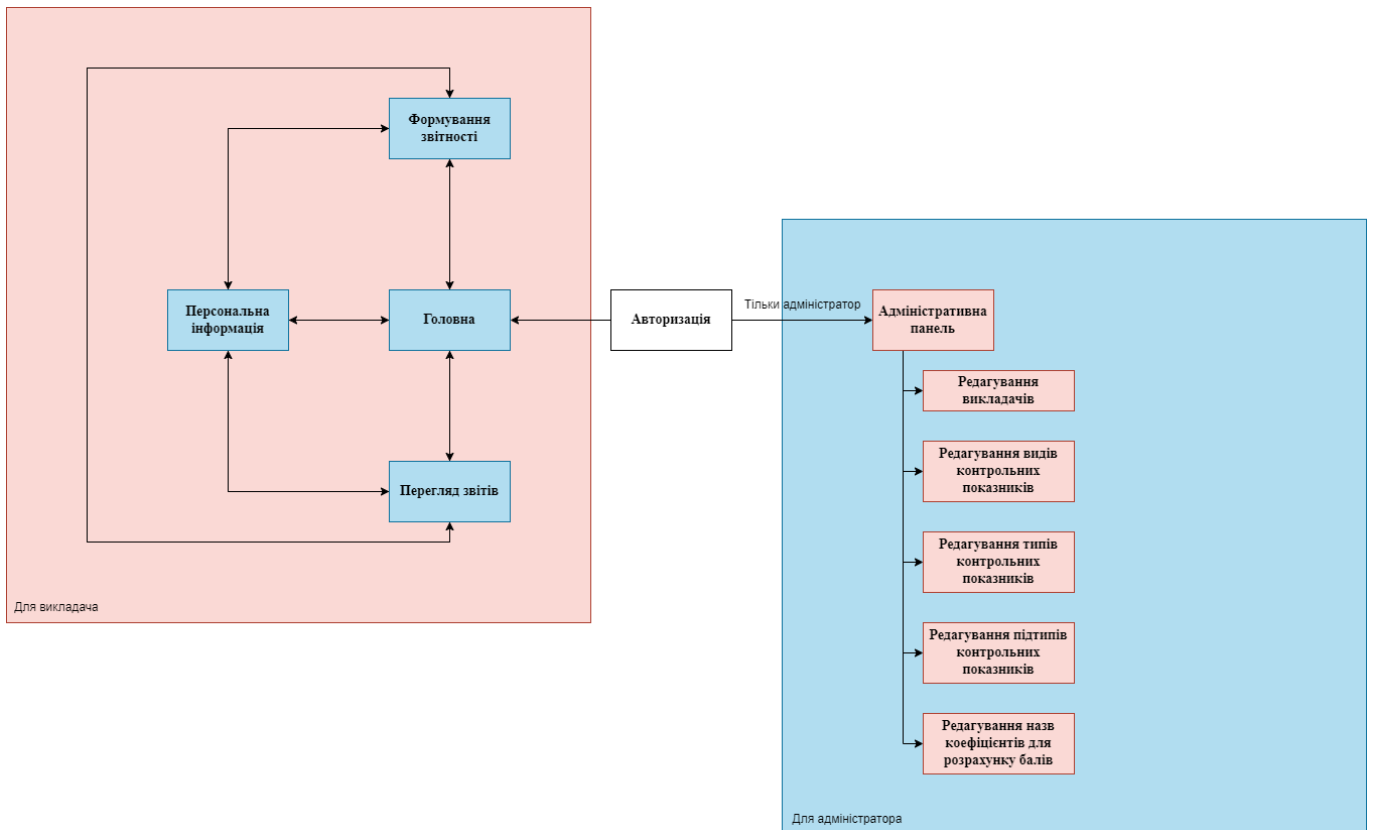


Рисунок А.14 – Система навігації

## 2.3 Вимоги до функціонування системи

### 2.3.1 Потреби користувача

Потреби користувача, визначені на основі рішення замовника, представлені у таблиці А.1.



Таблиця А.1 – Потреби користувача

ІД	Потреби користувача	Джерело
UN-01	Зручна авторизація.	Користувач (викладач)
UN-02	Можливість перегляду особистої інформації.	Користувач (викладач)
UN-03	Можливість вибору типу формування звітності.	Користувач (викладач)
UN-04	Можливість перегляду своїх попередніх звітів з підвищення кваліфікації.	Користувач (викладач)
UN-05	Можливість швидкого та зручного заповнення контрольних показників накопичувальної системи та іншої інформації для формування звітності.	Користувач (викладач)
UN-06	Формування звітів у форматах .pdf або .doc для збереження файлів.	Користувач (викладач)
UN-07	Інтуїтивно зрозумілий інтерфейс та навігація по сайту	Користувач (викладач)
UN-08	Редагування даних.	Адміністратор

### 2.3.2 Функціональні вимоги

Проаналізувавши потреби користувачів було визначено наступні вимоги:

- наявність авторизації користувачів;
- можливість перегляду своїх попередніх звітів з підвищення кваліфікації;
- наявність формування звітів про підвищення кваліфікації з можливістю завантаження файлу в форматі .docx.
- можливість перегляду персональної інформації;

– наявність панелі адміністратора для додавання, редагування та видалення інформації з вебдодатку.

### 2.3.3 Системні вимоги

Даний розділ визначає, розподіляє та вказує на системні вимоги, визначені розробником. Їх перелік наведений в таблиці А.2.

Таблиця А.2 – Системні вимоги

ІД	Системні вимоги	Пріоритет	Опис
SR-01	Веб-браузер (Chrome, Firefox, Safari, Edge тощо).	M	Підтримка сучасних версій веб-браузерів для коректного відображення та роботи з вебдодатком.
SR-02	Доступ до Інтернету.	M	Додаток потребує підключення до Інтернету для роботи.
SR-03	Роздільна здатність екрану.	S	Рекомендована роздільна здатність екрану для зручного відображення веб-сторінок додатку.

Умовні позначення в таблиці А.2:

Must have (M) – вимоги, які повинні бути реалізовані в системі;

Should have (S) – вимоги, які мають бути виконані, але вони можуть почекати своєї черги;

Could have (C) – вимоги, які можуть бути реалізовані, але вони не є центральною ціллю проекту.

## **2.4 Вимоги до видів забезпечення**

### **2.4.1 Вимоги до інформаційного забезпечення**

Вимоги до інформаційного забезпечення:

- інформація про викладачів. Система повинна зберігати інформацію про ПІБ викладачів, дані про їх науковий ступінь, номер телефону, логін та пароль. Ця інформація має бути доступною для формування звітності про підвищення кваліфікації, окрім логіну та паролю, вони слугують для авторизації викладача;
- дані про контрольні показники накопичувальної системи, їх назви, види, типи та підтипи;
- назви коефіцієнтів для розрахунків балів контрольних показників.

Для зберігання інформації потрібно використовувати реляційну базу даних, таку як MySQL. Реляційна база даних дозволяє зберігати дані у вигляді таблиць, що спрощує організацію та доступ до інформації.

До розроблюваного вебдодатку база даних буде підключена за допомогою мови програмування PHP.

### **2.4.2 Вимоги до лінгвістичного забезпечення**

Весь текст у вебдодатку має бути виконаний українською мовою.

### 2.4.3 Вимоги до програмного забезпечення

Для забезпечення стабільної роботи вебдодатку веб-браузер має бути Internet Explorer 7.0 і вище, або Firefox 3.5 і вище, або Opera 9.5 і вище, або Safari 3.2.1 і вище, або Chrome 2 і вище. Потрібно стабільне інтернет-підключення з достатньою швидкістю для завантаження веб-сторінок та медіа-контенту, будь-яка операційна система, така як Windows, macOS, Linux, будь-яка мобільна операційна система, така як Android і iOS, базовий рівень процесора та обсяг оперативної пам'яті (ОЗП), які використовуються в більшості сучасних комп'ютерів і пристроїв.

### 3 Склад і зміст робіт зі створення вебдодатку

Детальний опис етапів створення вебдодатку наведено в таблиці А.3.

Таблиця А.3 – Етапи створення вебдодатку

№	Склад і зміст робіт	Строк розробки
1	Визначення актуальності роботи, дослідження предметної області та проведення аналізу аналогів вебдодатків.	5 днів
2	Створення специфікації.	5 днів
3	Проектування структури вебдодатку.	2 дня
4	Розробка макетів сторінок вебдодатку.	6 днів
5	Верстка сторінок вебдодатку (frontend).	11 днів
6	Розробка бази даних для вебдодатку.	5 днів
7	Розробка усіх потрібних функцій вебдодатку (backend).	10 днів

Продовження табл. А.3

8	Тестування методом білого ящика.	1 день
9	Тестування методом чорного ящика.	1 день
10	Розміщення вебдодатку на хостингу.	1 день
11	Перевірка працездатності.	1 день
12	Написання супровідної документації.	5 днів
13	Реліз вебдодатку.	1 день
	Загальна тривалість робіт	54 днів

#### **4 Вимоги до складу й змісту робіт із введення вебдодатку в експлуатацію**

Вебдодаток має бути затверджено та розміщено на веб-хостингу.

## ДОДАТОК Б

### ПЛАНУВАННЯ РОБІТ

У сучасному вищому навчальному закладі важливою складовою є система підвищення кваліфікації науково-педагогічних працівників. Однак, процес ведення звітності контрольних показників накопичувальної системи обрахунку основних результатів підвищення кваліфікації часто вимагає значних зусиль від викладачів та адміністрації. З метою спрощення цього процесу, потрібно розробити вебдодаток, який спрямований на швидке та ефективно заповнення контрольних показників для викладачів кафедри інформаційних технологій, Сумського державного університету.

Розроблений вебдодаток вирішує актуальну проблему автоматизації процесу збору та розрахунку даних щодо підвищення кваліфікації викладачів. Застосування цієї розробки дозволить значно скоротити час, який витрачається на формування звітності вручну, тим самим полегшуючи роботу викладачів та адміністрації.

**Деталізація мети проєкту методом SMART.** Для успішності та конкурентоспроможності проєкту треба на концептуальному етапі правильно визначити його мету за допомогою SMART-методу. Вона має ширше формулювання. А саме: «Розробка вебдодатку автоматизації формування звітності про підвищення кваліфікації викладачів кафедри інформаційних технологій, Сумського державного університету для скорочення часу та зусиль науково-педагогічних працівників на заповнення звіту до 01 червня 2024 року». Результати деталізації мети методом SMART розміщені у таблиці Б.1.

Таблиця Б.1 – Деталізація мети методом SMART

<b>Specific</b>	Розробка вебдодатку автоматизації формування звітності про підвищення кваліфікації викладачів кафедри інформаційних технологій, Сумського державного університету.
<b>Measurable</b>	Результатом роботи проєкту є оцінка замовника.

Продовження табл. Б.1

<b>Achievable</b>	Мета досяжна, є затверджене технічне завдання від замовника.
<b>Relevant</b>	Для скорочення зусиль та витрат часу на заповнення звіту з підвищення кваліфікації викладачів кафедри інформаційних технологій, Сумського державного університету.
<b>Time-framed</b>	Є конкретний термін – до кінця 4 курсу (01 червня 2024 р.).

**Планування змісту робіт.** WBS (Work Breakdown Structure – ієрархічна структура робіт) – це графічний вигляд елементів проекту, які згруповані ієрархією у єдине ціле з продуктом проекту. Структура декомпозиції робіт орієнтована на досконале виконання робіт по частинам і сама є ключовою частиною проекту, яка спрямована на організацію командної роботи. Елементами декомпозиції можуть бути продукти, дані та послуги. Більше того, WBS забезпечує необхідним каркасом для ретельної оцінки термінів та контролю та графіків роботи.

На найвищому (першому) рівні розміщений продукт проекту. Основні дії та заходи, що забезпечують досягнення мети проекту, зафіксовані на другому рівні декомпозиції. Декомпозиція робіт виконується до тих пір, поки вони не стануть елементарними (простими).

Елементарні роботи – це дії, які мають однозначний чіткий результат, на які призначена відповідальному одна конкретна особа, для якої можна обчислити витрати праці і тривалість виконання.

На рисунку Б.1 представлено WBS з розробки вебдодатку формування звітності про підвищення кваліфікації викладачів.

**Планування структури виконавців.** Наступним етапом після декомпозиції процесів є розробка організаційної структури виконавців або OBS, яка визначається як графічна структура відображення учасників або відповідальних осіб, які беруть участь у реалізації проекту. У ролі відповідальних осіб виступають співробітники, що

відповідають за організацію і виконання елементарної роботи, що зазначена у WBS. Кожну елементарну роботу можна розглядати як окремий проєкт.

На рисунку Б.2 представлено організаційну структуру планування проєкту. Список виконавців, що функціонують в проєкті, описано в таблиці Б.2.



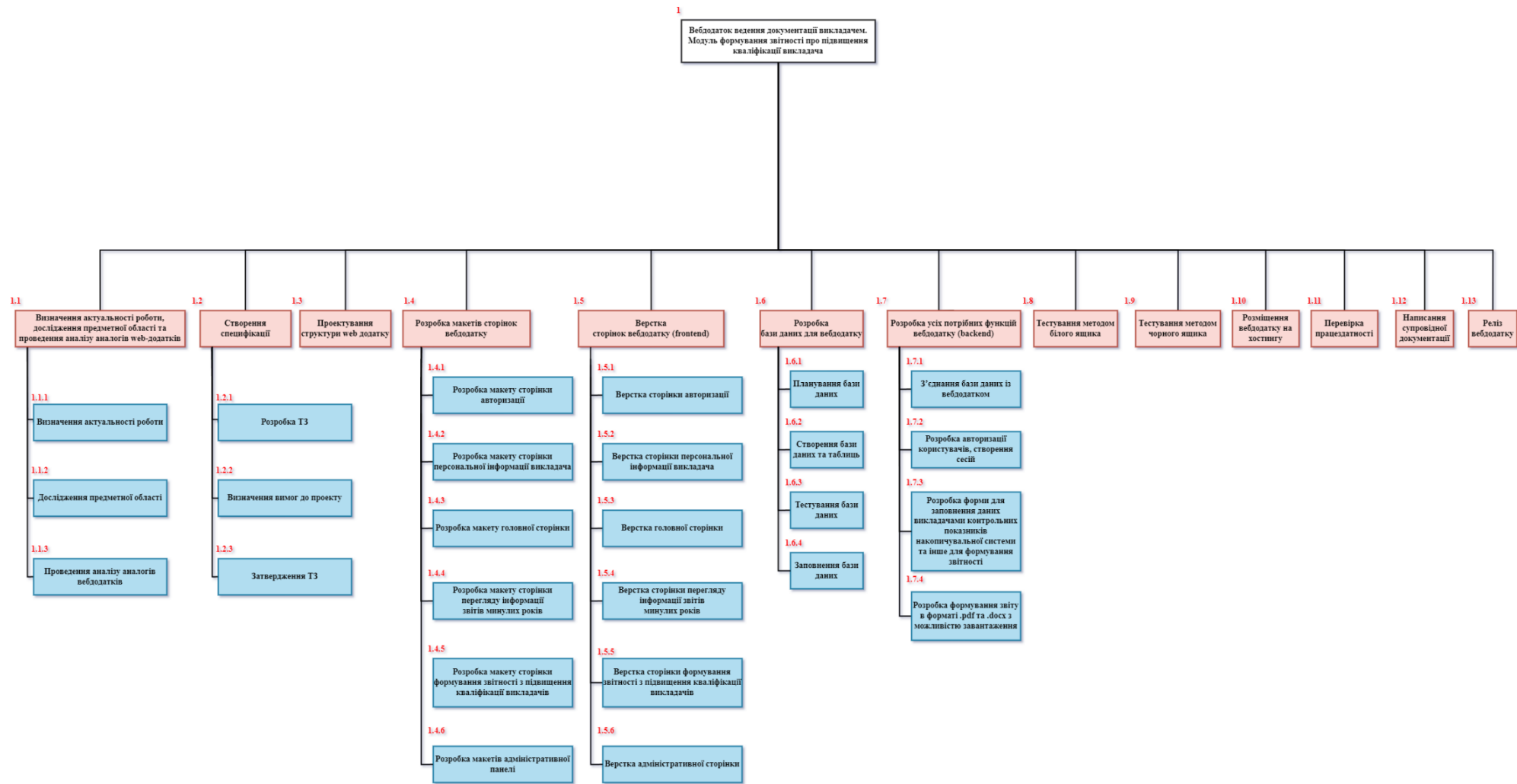


Рисунок Б.1 – WBS-структура робіт проекту

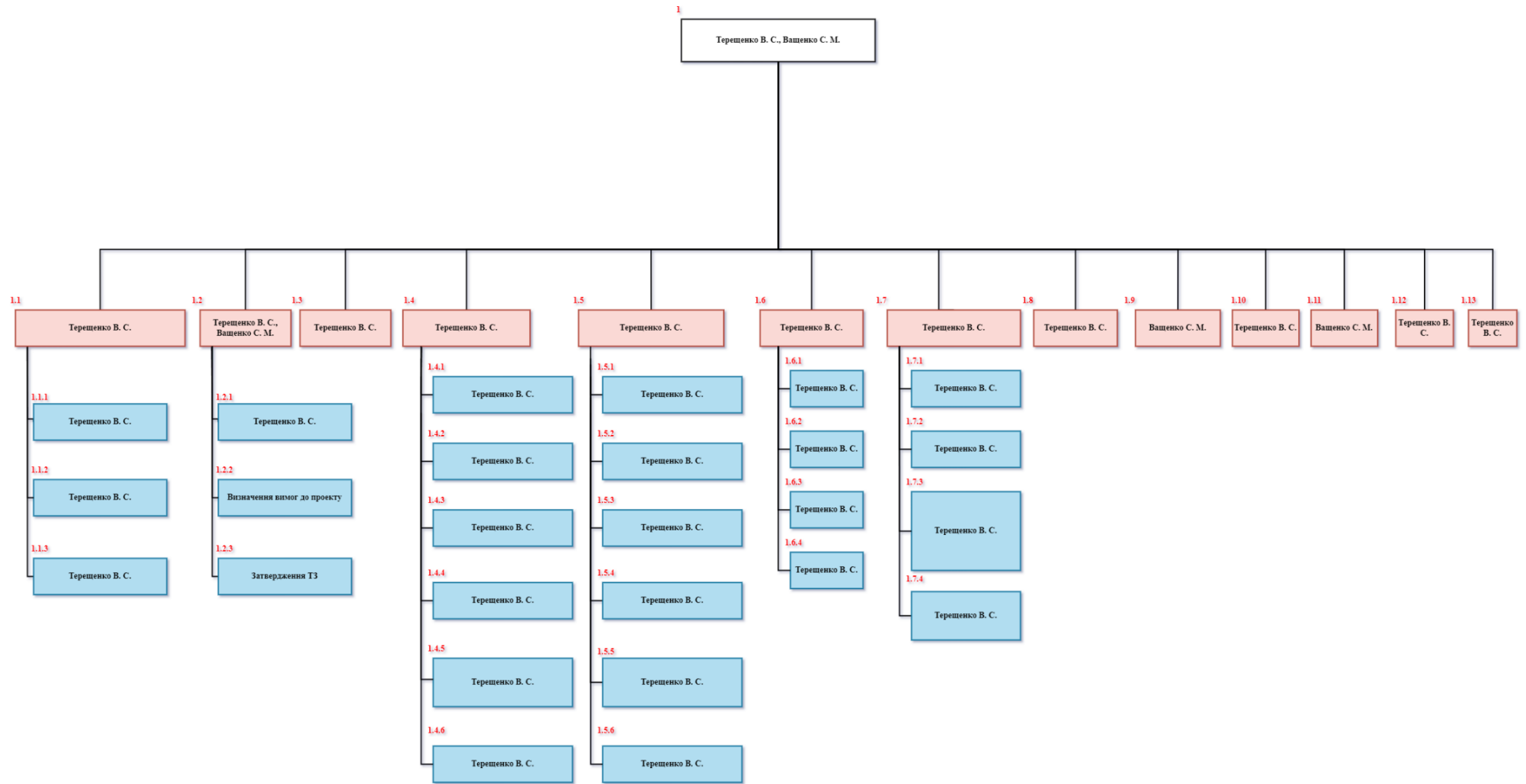


Рисунок Б.2 – OBS-структура робіт проект

Таблиця Б.2 – Виконавці проєкту

<b>Роль</b>	<b>ПІБ</b>	<b>Проектна роль</b>
Розробник	Терещенко В. С.	Виконує frontend та backend розробку.
Проектувальник	Терещенко В. С.	Виконує проектування бази даних та розробляє структуру вебдодатку.
Тестувальник	Ващенко С. М.	Відповідає за тестування функціоналу та дизайну вебдодатку, а саме методом чорного ящика.
Керівник проєкту	Ващенко С. М.	Формує завдання на розробку проєкту.
Менеджер проєкту	Терещенко В. С.	Відповідає за виконання термінів, розподіл ресурсів та завдань між учасниками. Виконує збір та аналіз даних.

**Діаграма Ганта.** Побудова календарного графіку (діаграми Ганта) є одним з важливих етапів планування проєкту, що виглядає як розклад виконання робіт з реальним розподілом дат. Завдяки йому можна отримати достовірне уявлення про тривалість процесів з обмеженнями у ресурсах, урахуванням вихідних днів та свят.

Календарний графік проєкту представлено на рисунках Б.3-Б.6.

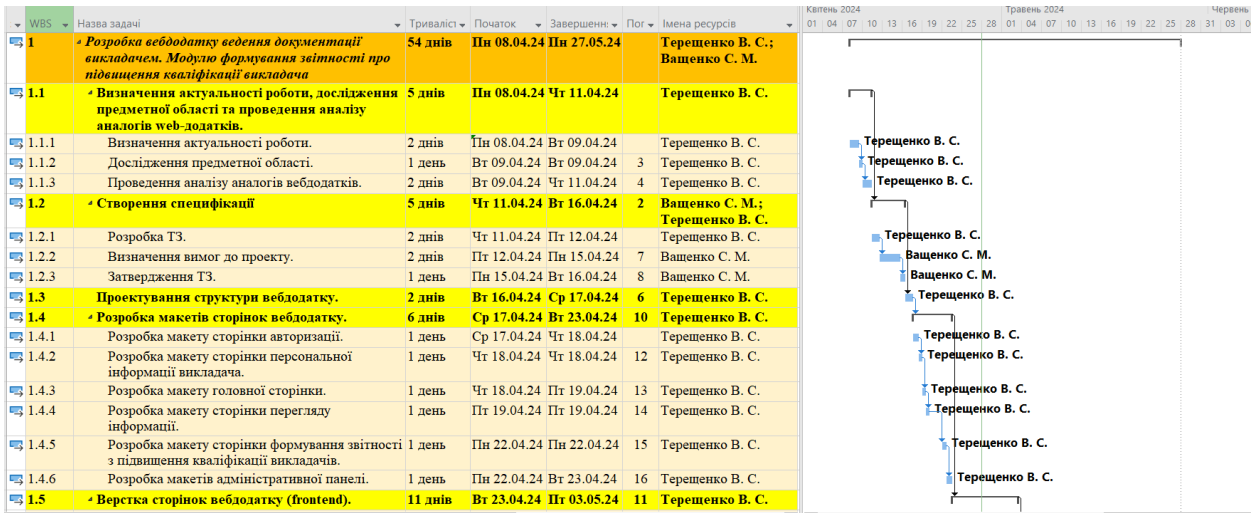


Рисунок Б.3 – Календарний графік проекту



Рисунок Б.4 – Продовження календарного графіку проекту

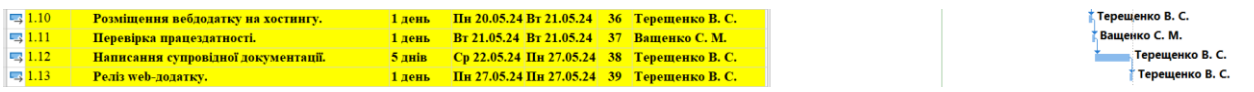


Рисунок Б.5 – Продовження календарного графіку проекту

<b>1</b>	<b>Розробка вебдодатку ведення документації викладачем. Модуль формування звітності про підвищення кваліфікації викладача</b>	<b>54 днів</b>	<b>Пн 08.04.24</b>	<b>Пн 27.05.24</b>	<b>Терещенко В. С.; Вашенко С. М.</b>
<b>1.1</b>	<b>Визначення актуальності роботи, дослідження предметної області та проведення аналізу аналогів вебдодатків.</b>	<b>5 днів</b>	<b>Пн 08.04.24</b>	<b>Чт 11.04.24</b>	<b>Терещенко В. С.</b>
1.1.1	Визначення актуальності роботи.	2 днів	Пн 08.04.24	Вт 09.04.24	Терещенко В. С.
1.1.2	Дослідження предметної області.	1 день	Вт 09.04.24	Вт 09.04.24	3 Терещенко В. С.
1.1.3	Проведення аналізу аналогів вебдодатків.	2 днів	Вт 09.04.24	Чт 11.04.24	4 Терещенко В. С.
<b>1.2</b>	<b>Створення специфікації</b>	<b>5 днів</b>	<b>Чт 11.04.24</b>	<b>Вт 16.04.24</b>	<b>2 Вашенко С. М.; Терещенко В. С.</b>
1.2.1	Розробка ТЗ.	2 днів	Чт 11.04.24	Пт 12.04.24	Терещенко В. С.
1.2.2	Визначення вимог до проекту.	2 днів	Пт 12.04.24	Пн 15.04.24	7 Вашенко С. М.
1.2.3	Затвердження ТЗ.	1 день	Пн 15.04.24	Вт 16.04.24	8 Вашенко С. М.
<b>1.3</b>	<b>Проектування структури вебдодатку.</b>	<b>2 днів</b>	<b>Вт 16.04.24</b>	<b>Ср 17.04.24</b>	<b>6 Терещенко В. С.</b>
<b>1.4</b>	<b>Розробка макетів сторінок вебдодатку.</b>	<b>6 днів</b>	<b>Ср 17.04.24</b>	<b>Вт 23.04.24</b>	<b>10 Терещенко В. С.</b>
1.4.1	Розробка макету сторінки авторизації.	1 день	Ср 17.04.24	Чт 18.04.24	Терещенко В. С.
1.4.2	Розробка макету сторінки персональної інформації викладача.	1 день	Чт 18.04.24	Чт 18.04.24	12 Терещенко В. С.
1.4.3	Розробка макету головної сторінки.	1 день	Чт 18.04.24	Пт 19.04.24	13 Терещенко В. С.
1.4.4	Розробка макету сторінки перегляду інформації.	1 день	Пт 19.04.24	Пт 19.04.24	14 Терещенко В. С.
1.4.5	Розробка макету сторінки формування звітності з підвищення кваліфікації викладачів.	1 день	Пн 22.04.24	Пн 22.04.24	15 Терещенко В. С.
1.4.6	Розробка макетів адміністративної панелі.	1 день	Пн 22.04.24	Вт 23.04.24	16 Терещенко В. С.
<b>1.5</b>	<b>Верстка сторінок вебдодатку (frontend).</b>	<b>11 днів</b>	<b>Вт 23.04.24</b>	<b>Пт 03.05.24</b>	<b>11 Терещенко В. С.</b>
1.5.2	Верстка сторінки персональної інформації викладача.	2 днів	Вт 23.04.24	Чт 25.04.24	19 Терещенко В. С.
1.5.3	Верстка головної сторінки.	1 день	Чт 25.04.24	Чт 25.04.24	20 Терещенко В. С.
1.5.4	Верстка сторінки перегляду інформації звітів.	1 день	Чт 25.04.24	Пт 26.04.24	21 Терещенко В. С.
1.5.5	Верстка сторінки формування звітності з підвищення кваліфікації викладачів.	2 днів	Пт 26.04.24	Пн 29.04.24	22 Терещенко В. С.
1.5.6	Верстка адміністративної сторінки.	4 днів	Пн 29.04.24	Пт 03.05.24	23 Терещенко В. С.
<b>1.6</b>	<b>Розробка бази даних для вебдодатку.</b>	<b>5 днів</b>	<b>Пт 03.05.24</b>	<b>Чт 09.05.24</b>	<b>18 Терещенко В. С.</b>
1.6.1	Планування бази даних.	1 день	Пт 03.05.24	Пт 03.05.24	Терещенко В. С.
1.6.2	Створення бази даних та таблиць.	2 днів	Пт 03.05.24	Пн 06.05.24	26 Терещенко В. С.
1.6.3	Тестування бази даних.	1 день	Вт 07.05.24	Вт 07.05.24	27 Терещенко В. С.
1.6.4	Заповнення бази даних.	1 день	Вт 07.05.24	Чт 09.05.24	28 Терещенко В. С.
<b>1.7</b>	<b>Розробка усіх потрібних функцій вебдодатку (backend).</b>	<b>10 днів</b>	<b>Чт 09.05.24</b>	<b>Пт 17.05.24</b>	<b>25 Терещенко В. С.</b>
1.7.1	З'єднання бази даних із вебдодатком.	1 день	Чт 09.05.24	Чт 09.05.24	Терещенко В. С.
1.7.2	Розробка авторизації користувачів, створення сесій.	2 днів	Чт 09.05.24	Пн 13.05.24	31 Терещенко В. С.
1.7.3	Розробка форми для заповнення даних викладачами контрольних показників накопичувальної системи та інше для формування звітності.	5 днів	Пн 13.05.24	Чт 16.05.24	32 Терещенко В. С.
1.7.4	Розробка формування звіту в форматі .pdf та .docx з можливістю завантаження.	2 днів	Чт 16.05.24	Пт 17.05.24	33 Терещенко В. С.
<b>1.8</b>	<b>Тестування методом білого ящика.</b>	<b>1 день</b>	<b>Пт 17.05.24</b>	<b>Пн 20.05.24</b>	<b>30 Терещенко В. С.</b>
<b>1.9</b>	<b>Тестування методом чорного ящика.</b>	<b>1 день</b>	<b>Пн 20.05.24</b>	<b>Пн 20.05.24</b>	<b>35 Вашенко С. М.</b>
<b>1.10</b>	<b>Розміщення вебдодатку на хостингу.</b>	<b>1 день</b>	<b>Пн 20.05.24</b>	<b>Вт 21.05.24</b>	<b>36 Терещенко В. С.</b>
<b>1.11</b>	<b>Перевірка працездатності.</b>	<b>1 день</b>	<b>Вт 21.05.24</b>	<b>Вт 21.05.24</b>	<b>37 Вашенко С. М.</b>
<b>1.12</b>	<b>Написання супровідної документації.</b>	<b>5 днів</b>	<b>Ср 22.05.24</b>	<b>Пн 27.05.24</b>	<b>38 Терещенко В. С.</b>
<b>1.13</b>	<b>Реліз вебдодатку.</b>	<b>1 день</b>	<b>Пн 27.05.24</b>	<b>Пн 27.05.24</b>	<b>39 Терещенко В. С.</b>

Рисунок Б.6 – Список робіт для побудови діаграми Ганта

**Управління ризиками проєкту.** Під час виконання якісної оцінки ризиків треба визначити ризики, які мають бути усунені якнайшвидше. В залежності від ступеня важливості ризику – реагування буде відповідне. Наступним етапом є виконання кількісного оцінювання ризиків. Кількісне та якісне оцінювання можуть виконувати одночасно або окремо, що залежить від ступеня забезпечення проєкту.

У таблиці Б.3 надано перелік ризиків даного проєкту. Результати оцінки ризиків надано у таблиці Б.4. Таблиця Б.5 представляє шкалу для класифікації ризиків за величиною впливу на проєкт та ймовірністю їх виникнення.

Таблиця Б.3 – Ризики проекту

<b>Номер ризику</b>	<b>Назва (опис) ризику</b>
1	Серйозна помилка, яка може бути знайдена під час тестування вебдодатку, на виправлення якої потрібно багато часу.
2	Проблема з інтеграцією нових технологій у браузер.
3	Зміна вимог до проекту.
4	Зміни в законодавстві, наприклад, зміни в законодавстві щодо захисту даних.
5	Повторне вторгнення ворогів у Сумську область.
6	Поновлення графіків відключення світла або блекаут.
7	Втрата даних та відсутність резервного копіювання.
8	Неефективне управління проектом.
9	Конфлікт між розробником та керівником проекту.
10	Хвороба розробника.

Таблиця Б.4 – Результати визначення ймовірності, впливу та рангу ризиків проекту

<b>Номер ризику</b>	<b>Назва (опис) ризику</b>	<b>Ймовірність (0.1-0.9)</b>	<b>Вплив (0.05-0.8)</b>	<b>Ранг</b>
1	Серйозна помилка, яка може бути знайдена під час тестування вебдодатку, на виправлення якої потрібно багато часу.	0.3	0.8	0.24
2	Проблема з інтеграцією нових технологій у браузер.	0.3	0.4	0.12

Продовження табл. Б.4

Номер ризику	Назва (опис) ризику	Ймовірність (0.1-0.9)	Вплив (0.05-0.8)	Ранг
3	Зміна вимог до проєкту.	0.3	0.2	0.06
4	Зміни в законодавстві, наприклад, зміни в законодавстві щодо захисту даних.	0.1	0.4	0.04
5	Повторне вторгнення ворогів у Сумську область.	0.5	0.8	0.4
6	Поновлення графіків відключення світла або блекаут.	0.7	0.8	0.56
7	Втрата даних та відсутність резервного копіювання.	0.3	0.8	0.24
8	Неефективне управління проєктом.	0.1	0.4	0.04
9	Конфлікт між розробником та керівником проєкту.	0.3	0.4	0.12
10	Хвороба розробника.	0.7	0.8	0.56

Таблиця Б.5 – Шкала оцінювання ризиків за ймовірністю виникнення та величиною впливу

Оцінка	Ймовірність виникнення	Вплив ризику	Тип ризику
1	Низька	Низький	Прийнятні
2	Середня	Середній	Виправдані
3	Висока	Високий	Недопустимі

Для того, щоб знизити негативний вплив ризиків на проєкт треба виконати планування реагування на них. До нього входить оцінка наслідків впливу на проєкт і розробка відповідних заходів. Аналіз виконується за показниками, які описані в таблиці Б.4. У результаті планування заходів реагування на ризики проєкту було отримано матрицю ймовірності виникнення та впливу ризиків (табл. Б.6). Зеленим кольором на матриці позначають прийнятні ризики, жовтим – виправдані, а червоним – недопустимі.

Таблиця Б.6 – Матриця ймовірності та впливу

Ймовірність ризику (Й)	Вплив загрози (ризик)				
	Дуже малий	Малий	Середній	Великий	Дуже великий
	0.05	0.1	0.2	0.4	0.8
0.9	0.045	0.09	0.18	0.36	0.72
0.7	0.035	0.07	R6 (0.14), R10 (0.14)	0.28	0.56
0.5	0.025	0.05	0.1	0.2	0.4
0.3	0.015	0.03	R3 (0.06)	R2 (0.12), R9 (0.12)	0.24
0.1	0.005	0.01	0.02	R4 (0.04), R8 (0.04)	R1 (0.08)

Класифікація ризиків проєкту за рівнем, відповідно до отриманого значення індексу, представлена у таблиці Б.7. У таблиці Б.8 описано ризики та стратегії реагування на кожен із них.



Таблиця Б.7 – Шкала оцінювання за рівнем ризику

<b>№</b>	<b>Назва</b>	<b>Межі</b>	<b>Ризики, які входять (номера)</b>
1	Прийнятні	$0.005 \leq R \leq 0.05$	4, 8
2	Виправдані	$0.05 < R \leq 0.14$	1, 2, 3, 6, 9, 10
3	Недопустимі	$0.14 < R \leq 0.72$	-

Таблиця Б.8 – Ризики проєкту та стратегії реагування

ID ризику	Статус ризику	Опис ризику	Ймовірність виникнення	Вплив ризику	Ранг ризику	Тип стратегії реагування	План А (заходи запобігання виникненню ризику)	План Б (заходи усунення наслідків ризику)
1	Відкритий	Серйозна помилка, яка може бути знайдена під час тестування вебдодатку, на виправлення якої потрібно багато часу.	0.1	0.8	0.08	Зменшення	1. Організувати додаткові тести перед релізом. 2. Проводити регулярні code review.	1. Розробити детальний план виправлення помилок. 2. Запровадити систему моніторингу за помилками.
2	Відкритий	Проблема з інтеграцією нових технологій у браузер.	0.1	0.4	0.04	Зменшення	1. Провести детальний технічний аудит нових технологій для забезпечення їхньої сумісності з різними браузерами. 2. Провести інтенсивне тестування нових технологій на різних браузерах перед їхньою фактичною інтеграцією.	1. Створити бекап-план, який передбачатиме можливість відновлення до попереднього стану у випадку невдачі.
3	Відкритий	Зміна вимог до проєкту.	0.3	0.2	0.06	Зменшення	1. Забезпечити гнучкість у проєкті для внесення змін.	1. Внести зміни відповідно до нових вимог.

Продовження табл. Б.8

ІД ризику	Статус ризику	Опис ризику	Ймовірність виникнення	Вплив ризику	Ранг ризику	Тип стратегії реагування	План А (заходи запобігання виникненню ризику)	План Б (заходи усунення наслідків ризику)
4	Відкритий	Зміни в законодавстві, наприклад, зміни в законодавстві щодо захисту даних.	0.1	0.4	0.04	Прийняття		1. Найняти консультанта з юридичних питань. 2. Провести навчання персоналу щодо нових правил.
6	Відкритий	Поновлення графіків відключення світла або блекаут.	0.7	0.2	0.14	Зменшення	1. Закупівля генераторів, безперебійників. 2. Підключення до провайдеру, у якого є генератор, щоб був інтернет.	1. Розробити план дій під час блекауту.
8	Відкритий	Неефективне управління проектом.	0.1	.4	0.04	Зменшення	1. Застосувати методологію управління, яка найкраще відповідає характеру проекту.	1. Потрібно визначити конкретні заходи, які можна вжити для виправлення ситуації. Це може бути зміна плану, перерозподіл завдань, визначення додаткових ресурсів тощо.

Продовження табл. Б.8

ІД ризику	Статус ризику	Опис ризику	Ймовірність виникнення	Вплив ризику	Ранг ризику	Тип стратегії реагування	План А (заходи запобігання виникненню ризику)	План Б (заходи усунення наслідків ризику)
9	Відкритий	Конфлікт між розробником та керівником проєкту.	0.3	0.4	0.12	Зменшення	<ol style="list-style-type: none"> <li>1. Налагодити гарні відносини між розробником та керівником.</li> <li>2. Дотримуватися ділового етикету спілкування.</li> <li>3. Створити комфортні умови для співпраці.</li> </ol>	<ol style="list-style-type: none"> <li>1. При виявленні непорозуміння потрібно в'яснити, що саме стало причиною непорозуміння обговорити її та створити здорову атмосферу в колективі.</li> </ol>
10	Відкритий	Хвороба розробника.	0.7	0.2	0.14	Зменшення	<ol style="list-style-type: none"> <li>1. Передбачити резерв часу.</li> </ol>	<ol style="list-style-type: none"> <li>2. Організувати дистанційну роботу для розробника.</li> </ol>

## ДОДАТОК В

### Лістинг програмного коду основних модулів вебдодатку.

#### Частина «user»

#### authorizathion.php

```

<?php
session_start();
require_once "../configuration/connect.php";

$email = $_POST["email"];
$password = $_POST["password"];

if($_POST["email"] != "" && $_POST["password"] != "")
{
    $check_user = mysqli_query($connect, "SELECT * FROM `Teacher` WHERE `Teacher_log` = '$email'
AND `Teacher_pas` = '$password'");
    if (mysqli_num_rows($check_user) > 0)
    {
        $user = mysqli_fetch_assoc($check_user);
        $_SESSION["user"] =
        [
            'Teacher_ID' => $user ['Teacher_ID'],
            'Scientific_title' => $user ['Scientific_title'],
            'Photo' => $user ['Photo'],
            'First_name_teacher'=> $user ['First_name_teacher'],
            'Second_name_teacher'=> $user ['Second_name_teacher'],
            'Fatherhood_teacher'=> $user ['Fatherhood_teacher'],
            'Teacher_email'=> $user ['Teacher_email'],
            'Teacher_phone'=> $user ['Teacher_phone'],
            'Post_ID'=> $user ['Post_ID']
        ];
        header("Location: choice.php");
    }

    $check_user = mysqli_query($connect, "SELECT * FROM `Admin` WHERE `Admin_log` = '$email' AND
`Admin_pass` = '$password'");
    if (mysqli_num_rows($check_user) > 0)
    {
        $user = mysqli_fetch_assoc($check_user);
        $_SESSION["user"] =
        [
            'Name_admin' => $user ['Name_admin'],

```

```

    'Surname_admin' => $user ['Surname_admin']
  ];
  header("Location: ../admin/search.html");
}

$_SESSION["msg"] = "<p>Не правильний email або пароль!</p>";
}
?>

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <link
      rel="stylesheet"
      href="https://fonts.googleapis.com/css2?family=Manrope:wght@200;300;400;500;600;700;800&displa
      y=swap"
      />
    <link rel="icon" href="../img/logo.png" type="image/x-icon" />
    <link type="text/css" href="style/authorization.css" rel="stylesheet" />
    <title>Авторизація</title>
  </head>

  <body>
    <div class="all">
      <!-- Хедер -->
      <header class="header">
        <div class="img"></div>
        <p>Кафедра інформаційних технологій</p>
      </header>

      <!-- Основна частина сайту -->
      <div class="main">
        <div class="ma">
          <div class="authorization">
            <div class="vhid">
              <h1>Вхід до облікового запису</h1>
            </div>
            <div class="form">
              <form method="post">
                <div class="form-group">
                  <label for="email">Електронна адреса</label>
                  <input
                    type="email"

```

```

        id="email"
        placeholder="Введіть електронну адресу"
        name="email"
        required
    />
</div>
<div class="form-group">
    <label for="password">Пароль</label>
    <input
        type="password"
        id="password"
        placeholder="Введіть пароль"
        name="password"
        required
    />
</div>
<div class="form-group">
    <div class="knop">
        <button type="submit">Увійти</button>
    </div>
</div>
</form>
</div>
<div class="bl">
    <?php
    if($_SESSION["msg"])
    {
        echo '<p>'. $_SESSION["msg"] . '</p>';
        unset($_SESSION["msg"]);
    }
    ?>
</div>
</div>
</div>
</div>

<!-- Футер -->
<footer class="footer">
    <div class="location">
        <p class="imgloc">
            
        </p>
        <p class="adres">
            вул. Харківська, 116, Суми, Україна, 40007 Головний корпус, 13-й
            поверх, ауд. П1303
        </p>
    </div>
</div>

```

```

<div class="phone">
  <div class="phone1">
    <p class="img">
      
    </p>
    <p class="pr">приймальня +38 (0542) 68-78-54</p>
  </div>
  <div class="phone2">
    <p class="img">
      
    </p>
    <p class="pr">завідувач кафедри +38 (0542) 68-78-63</p>
  </div>
</div>

<div class="email">
  <div class="insta">
    <a href="https://www.instagram.com/itp_sumdu?igshid=fopu9dqxdigb">
      
      <span>Instagram</span>
    </a>
  </div>
  <div class="em">
    <p class="img2"></p>
    <p class="adres">info@itp.sumdu.edu.ua</p>
  </div>
</div>
</footer>
</div>
</body>
</html>

```

## creating\_report.php

```

<?php
session_start();
require_once "../configuration/connect.php";

$Benchmark = mysqli_query($connect, "SELECT * FROM `Benchmark`");
$Benchmark = mysqli_fetch_all($Benchmark);

```



```

$Type_benchmark = mysqli_query($connect, "SELECT * FROM `Type_benchmark`");
$Type_benchmark = json_encode(mysqli_fetch_all($Type_benchmark));

$Kind_benchmark = mysqli_query($connect, "SELECT * FROM `Kind_benchmark`");
$Kind_benchmark = json_encode(mysqli_fetch_all($Kind_benchmark));

// для безпеки сторінки, тільки для адміністраторів
if(!isset($_SESSION["user"]))
{
    header("Location: authorization.php");
}
?>

<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="UTF-8" />
        <meta http-equiv="X-UA-Compatible" content="IE=edge" />
        <meta name="viewport" content="width=device-width, initial-scale=1.0" />
        <link
            rel="stylesheet"
            href="https://fonts.googleapis.com/css2?family=Manrope:wght@200;300;400;500;600;700;800&displa
            y=swap"
            />
        <link rel="icon" href="../img/logo.png" type="image/x-icon" />
        <link type="text/css" href="style/creating_report.css" rel="stylesheet" />
        <title>Ведення документації</title>
    </head>

    <script
src="https://cdnjs.cloudflare.com/ajax/libs/docxtemplater/3.45.0/docxtemplater.js"></script>
    <script src="https://unpkg.com/pizzip@3.1.6/dist/pizzip.js"></script>
    <script
src="https://cdnjs.cloudflare.com/ajax/libs/FileSaver.js/1.3.8/FileSaver.js"></script>
    <script src="https://unpkg.com/pizzip@3.1.6/dist/pizzip-utils.js"></script>

    <body>
        <div class="all">
            <!-- Хедер -->
            <header class="header">
                <div class="img"></div>
                <p>Кафедра інформаційних технологій</p>
            </header>

            <!-- Основна частина сайту -->
            <div class="main">

```

```

<!-- Меню сайту -->
<div class="menu">
  <div class="photoperson">
    <?php
      if($_SESSION['user']['Photo'] != NULL)
      {
        echo '';
      }
      else
      {
        echo '';
      }
    ?>
  </div>
  <div class="menunumber">
    <ul>
      <li>
        <a href="person_information.php">
          
        </a>
      </li>
      <li>
        <a href="choice.php">
          
        </a>
      </li>
      <li>
        <a href="choiceview.php">
          
        </a>
      </li>
      <li>
        <a href="../configuration/exit.php">
          
        </a>
      </li>
    </ul>
  </div>
</div>
<div class="adapt">
  <div class="name">
    <div class="namereport">
      <p>Звіт про підвищення кваліфікації</p>
    </div>
    <div class="date">
      <form method="post" id="dateForm">

```

```

<div class="form-group">
  <label for="date">
    Введіть дату початку проходження підвищення кваліфікації
  </label>
  <input
    type="date"
    id="date_start"
    placeholder="Виберіть дату початку"
    name="date_start"
    required
  />
</div>
<div class="form-group">
  <label for="date">
    Введіть дату кінця проходження підвищення кваліфікації
  </label>
  <input
    type="date"
    id="date_end"
    placeholder="Виберіть дату кінця"
    name="date_end"
    required
  />
</div>
<div class="form-group">
  <label> Тематичне спрямування </label>
  <input
    type="text"
    id="thematicfocus"
    name="thematicfocus"
    required
  />
</div>
</form>
</div>
</div>
<div class="ma">

<!-- <div class="formreport">
  <div class="form">
    <form method="post">
      <div class="name">
        <input type="text" name="readonlyInput" value="1 Навчання та стажування"
          readonly style="border:none; background:transparent; font-size:inherit;">
        <button class="del">
          
        </button>
      </div>
    </form>
  </div>
</div>

```

```

</div>
<div class="d">
  <div class="form-group">
    <label> Тип контрольного показника </label>
    <select name="typebenchmark" required>
      <option value="Kzkzk">Kzkzk</option>
      <option value="Fff">Fff</option>
    </select>
  </div>
  <div class="form-group">
    <label> Підтип контрольного показника </label>
    <select name="subtypebenchmark" id="">
      <option value="Kzkzk">Kzkzk</option>
      <option value="Fff">Fff</option>
    </select>
  </div>
  <div class="form-group">
    <label> Тематичне спрямування </label>
    <input
      type="text"
      id="thematicfocus"
      placeholder="Введіть тематичне спрямування"
      name="thematicfocus"
      required
    />
  </div>
  <div class="form-group" id="form-group1">
    <label>
      Кількість годин, зазначених у відповідному документі
      *назва залежить від вибору типу/підтипу контр. показника
    </label>
    <input
      type="number"
      id="points"
      placeholder=""
      name="points"
      required
    />
  </div>
  <div class="form-group" id="form-group1">
    <label> Довідкова інформація </label>
    <textarea
      type="text"
      name="backgroundinformation"
      placeholder="Введіть довідкову інформацію"
      id="backgroundinformation"
      rows="3"
    >
  </div>

```

```

        >
        </textarea>
    </div>
    <div class="form-group" id="form-group2">
        <label> Додати файл </label>
        <input
            id="addfile"
            placeholder="Виберіть файл"
            name="addfile"
            type="file"
        />
    </div>
</div>
</form>
</div>
</div> -->

<!-- Контейнер для створених блоків -->
<div id="container"></div>
<div class="add">
    <div class="done">
        <button type="button" id="submitAll">Сформувати звіт</button>
    </div>
    <div class="add1">
        <a href="" id="openModal">
            
        </a>
    </div>
</div>
</div>
</div>

<!-- Футер -->
<footer class="footer">
    <div class="location">
        <p class="imgloc">
            
        </p>
        <p class="adres">
            вул. Харківська, 116, Суми, Україна, 40007 Головний корпус, 13-й
            поверх, ауд. П1303
        </p>
    </div>

    <div class="phone">
        <div class="phone1">

```

```

    <p class="img">
      
    </p>
    <p class="pr">приймальня +38 (0542) 68-78-54</p>
  </div>
  <div class="phone2">
    <p class="img">
      
    </p>
    <p class="pr">завідувач кафедри +38 (0542) 68-78-63</p>
  </div>
</div>

<div class="email">
  <div class="insta">
    <a href="https://www.instagram.com/itp_sumdu?igshid=fopu9dqxdigb">
      
      <span>Instagram</span>
    </a>
  </div>
  <div class="em">
    <p class="img2"></p>
    <p class="adres">info@itp.sumdu.edu.ua</p>
  </div>
</div>
</footer>
</div>

<!-- Модальне вікно -->
<div id="modal" class="modal">
  <div class="modal-content">
    <div class="cl">
      <h2>Оберіть вид контрольного показника</h2>
      <span class="close">&times;</span>
    </div>
    <form id="radioForm" method="post">

    <?php
    $benchmark = mysqli_query($connect, "SELECT * FROM `Benchmark`");
    $benchmark = mysqli_fetch_all($benchmark);

    if(count($benchmark) > 0)
    {

```

```

    foreach ($benchmark as $benchmark_key) {
    echo
    '<div class="form-group">
    <label>
    <input
    type="radio"
    name="option"
    value="' . $benchmark_key[0] . ' . ' . $benchmark_key[1] . "'
    required
    />
    ' . $benchmark_key[0] . ' . ' . $benchmark_key[1] . '
    </label>
    </div>';
    }
}
else
{
    echo '<div class="form-group">
    <label>
    Немає видів контрольних показників
    </label>
    </div>';
}
?>

<div class="donnn"><button type="submit" class="submit"
id="submitRadio">Підтвердити</button></div>
</form>
</div>
</div>

<script src="js/modal_creative_report.js"></script>

<script>
document.addEventListener("DOMContentLoaded", function () {
    // Отримуємо модальне вікно та форму
    const modal = document.getElementById("modal");
    const radioForm = document.getElementById("radioForm");
    const container = document.getElementById("container");

    // Відкриття модального вікна (додайте відповідний код для триггеру)
    // modal.style.display = "block";

    // Закриття модального вікна
    document.querySelector(".close").onclick = function () {
        modal.style.display = "none";
    };
};

```

```

// Закриття модального вікна при кліку за межами вікна
window.onclick = function (event) {
    if (event.target == modal) {
        modal.style.display = "none";
    }
};

// Обробка події відправки форми
radioForm.onsubmit = function (event) {
    event.preventDefault();
    const selectedOption = document.querySelector(
        'input[name="option"]:checked'
    ).value;

    //console.log(selectedOption);
    // Витяг даних в джс
    const Type_benchmark = <?php echo $Type_benchmark; ?>;
    const Benchmark = <?php echo $Benchmark; ?>;
    const Kind_benchmark = <?php echo $Kind_benchmark; ?>;
    const parts = selectedOption.split(' '); // для витягу тільки назви Benchmark
    const Benchmark_ID = parts[0];
    const matchedValues = [];
    const type_benchmark_id = []; // айдішники для тайпбенчмарку

    for (const item of Type_benchmark) {
        const Value = item[3]; // Витягуємо айдти FK , Benchmark в Type_benchmark
        const idValue = item[0];
        if (Value === Benchmark_ID)
        {
            matchedValues.push(item[2] + " " + item[1]);
            type_benchmark_id.push(item[0]);
        }
    }

    //console.log(matchedValues);
    //console.log(type_benchmark_id);

    // Створення <select> елемента для типу
    const selectElement = document.createElement('select');
    selectElement.name = 'type';
    selectElement.required = true;

    // Додавання опції за замовчанням
    const defaultOption = document.createElement('option');
    defaultOption.value = '';
    defaultOption.textContent = 'Оберіть тип контрольного показника';

```



```

defaultOption.disabled = true;
defaultOption.selected = true;
selectElement.appendChild(defaultOption);

// Видалення опції за замовчанням після вибору іншої опції
selectElement.addEventListener('change', function() {
    const defaultOption = selectElement.querySelector('option[value=""]');
    if (defaultOption) {
        selectElement.removeChild(defaultOption);
    }
});

// Додавання опцій в елемент <select>
matchedValues.forEach(value => {
    const option = document.createElement('option');
    option.value = value;
    option.textContent = value;
    selectElement.appendChild(option);
});

// Create <select> element для підтипу
const selectElement2 = document.createElement('select');
selectElement2.name = 'subtype';
selectElement2.required = true;

// Додавання опції за замовчанням для селекта підтипу
const defaultOption3 = document.createElement('option');
defaultOption3.value = '';
defaultOption3.textContent = 'Оберіть спочатку тип контрольного показника';
defaultOption3.disabled = true;
defaultOption3.selected = true;
selectElement2.appendChild(defaultOption3);

// Видалення опції за замовчанням після вибору іншої опції
selectElement2.addEventListener('change', function() {
    const defaultOption3 = selectElement2.querySelector('option[value=""]');
    if (defaultOption3) {
        selectElement2.removeChild(defaultOption3);
    }
});

// Додавання обробника події зміни (change) для елемента <select> для типу
selectElement.addEventListener('change', function(event) {
    var name_Kind_benchmark = [];
    selectElement2.innerHTML = ''; // онулювання масиву

    let selectedValue = event.target.value;

```

```

//console.log('Вибране значення:', selectedValue);
selectedValue = String(selectedValue);
const selected = selectedValue.split(' ');
//console.log('Вибране значення:', selected);
const numberSelected = selected[0];

var typeValue = "";
for (const item of Type_benchmark){
  if (numberSelected === item[2])
  {
    typeValue = item[0]; // ID_Type_Benchmark
  }
}

for (const item of Kind_benchmark)
{
  if (item[3] == typeValue)
  {
    name_Kind_benchmark.push(item[2] + " " + item[1]); // Number_kind_benchmark +
Kind_benchmark_name
    //console.log(name_Kind_benchmark);
  }
}

// Додавання опцій в елемент <select> для підтипу
name_Kind_benchmark.forEach(value => {
  const option = document.createElement('option');
  option.value = value;
  option.textContent = value;
  selectElement2.appendChild(option);
});

// Отримуємо батьківський елемент, в який будемо додавати <label> та <input>
const labelContainer = newBlock.querySelector('.labelContainer');

// Перевіряємо, чи вже існує елемент <label>
var labelElement = labelContainer.querySelector('label');

// Якщо елемент <label> ще не існує, створюємо його
if (!labelElement)
{
  labelElement = document.createElement('label');
  labelContainer.appendChild(labelElement);
}

let coefficients = "";
for (const item of Type_benchmark)

```

```

{
  if (typeValue == item[0])
  {
    coefficients = item[5];
  }
}

// Встановлюємо текст для <label>
if(coefficients != null && coefficients !== '')
{
  labelElement.textContent = `${coefficients}`;
  // Перевіряємо, чи інпут з ім'ям 'points' вже існує
  let pointsInput = labelContainer.querySelector('input[name="points"]');
  // Якщо інпут не існує, створюємо його
  if(!pointsInput){
    pointsInput = document.createElement('input');
    pointsInput.type = 'number';
    pointsInput.id = 'points';
    pointsInput.placeholder = '';
    pointsInput.name = 'points';
    labelContainer.appendChild(pointsInput);
  }
}
else
{
  labelElement.textContent = '';
  // Видаляємо інпут, якщо лейбел порожній
  const pointsInput = labelContainer.querySelector('input[name="points"]');
  if(pointsInput)
  {
    pointsInput.remove();
  }
}
});

// Додавання обробника події зміни (change) для елемента <select> для підтипу
selectElement2.addEventListener('change', function(event)
{
  // Отримуємо батьківський елемент, в який будемо додавати <label>
  const labelContainer = newBlock.querySelector('.labelContainer');

  // Перевіряємо, чи вже існує елемент <label>
  var labelElement = labelContainer.querySelector('label');

  // Якщо елемент <label> ще не існує, створюємо його
  if (!labelElement)
  {

```

```

    labelElement = document.createElement('label');
    labelContainer.appendChild(labelElement);
    newBlock.querySelector('.labelContainer').appendChild(labelContainer);
    container.appendChild(newBlock);
}

let selectedValue = event.target.value;

selectedValue = String(selectedValue);
const selected = selectedValue.split(' ');
const numberSelected = selected[0];

for (const item of Kind_benchmark)
{
    if (item[2] == numberSelected)
    {
        selectedValue = item[0];
    }
}

let coefficients = "";
for (const item of Kind_benchmark)
{
    if (selectedValue == item[0])
    {
        coefficients = item[5];
    }
}

// Встановлюємо текст для <label>
if(coefficients != null && coefficients !== '')
{
    labelElement.textContent = `${coefficients}`;
    // Перевіряємо, чи інпут з ім'ям 'points' вже існує
    let pointsInput = labelContainer.querySelector('input[name="points"]');
    // Якщо інпут не існує, створюємо його
    if(!pointsInput)
    {
        pointsInput = document.createElement('input');
        pointsInput.type = 'number';
        pointsInput.id = 'points';
        pointsInput.placeholder = '';
        pointsInput.min = '0'; // Додаємо атрибут min
        pointsInput.name = 'points';
        labelContainer.appendChild(pointsInput);

        pointsInput.addEventListener('input', function () {

```

```

        if (this.value < 0) {
            this.value = 0;
        }
    });

    pointsInput.addEventListener('keydown', function (event) {
        // Заборона введення мінуса та інших недопустимих символів
        if (event.key === '-' || event.key === 'e' || event.key === 'E') {
            event.preventDefault();
        }
    });

    // Запобігання вставці від'ємних чисел через буфер обміну
    pointsInput.addEventListener('paste', function (event) {
        let paste = (event.clipboardData ||
window.clipboardData).getData('text');
        if (paste.indexOf('-') !== -1) {
            event.preventDefault();
        }
    });
}
}
else {
    labelElement.textContent = '';
    // Видаляємо інпут, якщо лейбел порожній
    const pointsInput = labelContainer.querySelector('input[name="points"]');
    if(pointsInput){
        pointsInput.remove();
    }
}
});

// Створення нового блоку
const newBlock = document.createElement("div");
newBlock.className = "formreport";

newBlock.innerHTML = `
<div class="form">
  <form method="post">
    <div class="name">
      <input type="text" name="readonlyInput" value="${selectedOption}"
        readonly style="border:none; font-size:inherit;">
      <button class="del">
        
      </button>
    </div>
  </div>
<div class="d">

```

```

<div class="form-group">
  <label> Тип контрольного показника </label>
  <!-- Placeholder for selectElement -->
  <div id="selectContainer"></div>
</div>
<div class="form-group">
  <label> Підтип контрольного показника </label>
  <!-- Placeholder for selectElement -->
  <div id="selectContainer2"></div>
</div>
<div class="form-group labelContainer" id="form-group1">
  <!-- Placeholder for labelElement -->
  <label></label>
  <!-- Input -->
</div>
<div class="form-group" id="form-group1">
  <label> Довідкова інформація </label>
  <textarea
    type="text"
    name="backgroundinformation"
    placeholder="Введіть довідкову інформацію"
    id="backgroundinformation"
    rows="3"
  ></textarea>
</div>
<!--
<div class="form-group" id="form-group2">
  <label> Додати файл </label>
  <input
    id="addfile"
    placeholder="Виберіть файл"
    name="addfile"
    type="file"
  />
</div> -->
</div>
</form>
</div>
`;

// Append the selectElement to the placeholder div (для типу)
newBlock.querySelector('#selectContainer').appendChild(selectElement);
// Append the selectElement2 to the placeholder div (для підтипу)
newBlock.querySelector('#selectContainer2').appendChild(selectElement2);

// Додавання нового блоку до контейнера
container.appendChild(newBlock);

```

```

// Закриття модального вікна
modal.style.display = "none";
};

// Видалення блоку при натисканні кнопки видалення
container.addEventListener("click", function (event) {
  if (event.target.closest(".del")) {
    event.target.closest(".formreport").remove();
  }
});
});
</script>

<script>
  // Збір даних з усіх форм блоків
  document.getElementById('submitAll').onclick = function () {
    const formReports = document.querySelectorAll('.formreport');
    const collectedData = [];
    let collectedWord = [];

    <?php
      $query = mysqli_query($connect, "SELECT `Type_benchmark_ID`, `Number_benchmark_name`,
`Point` FROM `Type_benchmark`");
      $query2 = mysqli_query($connect, "SELECT `Kind_benchmark_ID`,
`Number_kind_benchmark`, `Mark` FROM `Kind_benchmark`");
      $result = json_encode(mysqli_fetch_all($query));
      $result2 = json_encode(mysqli_fetch_all($query2));
      ?>

    const coef_type = <?php echo $result; ?>;
    const coef_subtype = <?php echo $result2; ?>;

    formReports.forEach(formReport => {
      const form = formReport.querySelector('form');
      const formData = new FormData(form);

      const data = {};
      formData.forEach((value, key) => {
        data[key] = value;
      });

      collectedData.push(data);
    });

    // Отримати значення форми
    const dateForm = document.getElementById('dateForm');

```

```

const formData = new FormData(dateForm);

// Отримати значення дат початку та кінця з FormData
const startDate = new Date(formData.get('date_start'));
const endDate = new Date(formData.get('date_end'));
const thematicFocus = formData.get('thematicfocus');

// Перевірка, чи всі необхідні поля містять значення
if (startDate != "" && endDate != "" && thematicFocus != "")
{
    // Перевірка, чи дата початку є раніше за дату кінця
    if (startDate < endDate)
    {
        // Якщо усі перевірки пройдено успішно, додати дані до об'єкта data
        const data = {};
        formData.forEach((value, key) =>
        {
            data[key] = value;
        });
    }
}

////////////////////////////////////
////////////////////////////////////

console.log('collectedData', collectedData);
let formula1 = [];
let formula2 = [];
let points_subtype = [];
let points_type = [];
let mark = collectedData.map(item => item.points);
const subtypes = collectedData.map(item => item.subtype);
const types = collectedData.map(item => item.type);
const readonlyInputs = collectedData.map(item => item.readonlyInput);
const backgroundinformation = collectedData.map(item =>
item.backgroundinformation);
const addfile = collectedData.map(item => item.addfile);
const collectedWord = [[], [], [], [], []]; // Ініціалізуємо collectedWord як
масив масивів

let n = 0;

function extractNumberFromSubtype(subtype) {
    if (subtype) {
        const result = subtype.split(' ');
        return result[0];
    }
    return null;
}

```



```

}

function extractNumberFromType(type) {
  if (type) {
    const result = type.split(' ');
    return result[0];
  }
  return null;
}

for (let i = 0; i < collectedData.length; i++) {
  const item = collectedData[i];

  collectedWord[0].push(subtypes[i]);
  collectedWord[1].push(new File([], `file${i + 1}`));
  collectedWord[2].push(readonlyInputs[i]);
  collectedWord[3].push(backgroundinformation[i]);

  if (subtypes[i] !== undefined && subtypes[i] !== '') {
    let numbersFromSubtypes = extractNumberFromSubtype(subtypes[i]);

    for (let j = 0; j < coef_subtype.length; j++) {
      const coefItem = coef_subtype[j];
      if (coefItem[1] === numbersFromSubtypes) {
        points_subtype[i] = parseInt(coefItem[2]);
      }
    }

    if (mark[i] !== undefined && mark[i] !== '') {
      const mark1 = parseInt(mark[i]);
      n = (parseInt(points_subtype[i] * mark1));
    } else {
      n = (parseInt(points_subtype[i]));
    }

    collectedWord[4].push(n);
    formulal.push(n);
  }
  else {
    let numbersFromTypes = extractNumberFromType(types[i]);

    for (let j = 0; j < coef_type.length; j++) {
      const coefItem = coef_type[j];
      if (coefItem[1] === numbersFromTypes) {
        points_type[i] = parseInt(coefItem[2]);
      }
    }
  }
}

```

```

        if (mark[i] !== undefined && mark[i] !== '') {
            const mark1 = parseInt(mark[i]);
            n = (parseInt(points_type[i] * mark1));
        } else {
            n = (parseInt(points_type[i]));
        }

        collectedWord[4].push(n);
        formula2.push(n);
    }

    console.log("points_subtype", points_subtype[i]);
    console.log("points_type", points_type[i]);
    console.log("mark", mark[i]);
    console.log("formula1", formula1);
    console.log("formula2", formula2);
}

const CollectedWord = [
    collectedWord[0], // subtypes
    collectedWord[1], // addfile (як файли)
    collectedWord[2], // readonlyInputs
    collectedWord[3], // backgroundinformation
    collectedWord[4] // результат формули
];

console.log('CollectedWord', CollectedWord);
//collectedWord = collectedWord.slice(0, 6);

const newArray = [];

// Ітерація через readonlyInputs та результат формули
for (let i = 0; i < CollectedWord[2].length; i++) {
    let readonlyInput = CollectedWord[2][i];
    let formulaResult = CollectedWord[4][i];

    // Розділення readonlyInput на число та текст, видалення крапки з числа
    let splitInput = readonlyInput.match(/^(\\d+)\\.?\\s*(.*)$/);
    let numberPart = splitInput ? splitInput[1].replace('.', '') : '';
    let textPart = splitInput ? splitInput[2] : readonlyInput;

    // Додавання розділених частин у новий масив
    newArray.push([numberPart, textPart, formulaResult]);
}

<?php

```

```

$Benchmark1 = mysqli_query($connect, "SELECT * FROM `Benchmark`");
$Benchmark1 = json_encode(mysqli_fetch_all($Benchmark1));
?>;

```

```

const Benchmark2 = <?php echo $Benchmark1; ?>;
console.log(Benchmark2);
const mas = new Array(Benchmark2.length).fill(0);
const result = [];

```

```

for (const i of newArray) {
  let foundIndex = -1;
  for (let j = 0; j < Benchmark2.length; j++) {
    if (i[0] == Benchmark2[j][0]) {
      foundIndex = j;
      break;
    }
  }
  if (foundIndex !== -1) {
    mas[foundIndex] += i[2];
  }
}

```

```

console.log(mas);
let sum = mas.reduce((acc, val) => acc + val, 0);
console.log(sum);

```

//////////////////////////////////// Word

```

function loadFile(url, callback) {
  PizZipUtils.getBinaryContent(url, callback);
}

```

```

loadFile(
"Template.docx", // You need to replace this with the actual path to your

```

template

```

function (error, content)
{
  if (error)
  {
    throw error;
  }
  const zip = new PizZip(content);
  const doc = new window.docxtemplater(zip,
  {
    paragraphLoop: true,
    linebreaks: true,

```

```

    });

    let secondname2 = <?php echo
json_encode($_SESSION['user']['Second_name_teacher']); ?>;
    let name2 = <?php echo
json_encode($_SESSION['user']['First_name_teacher']); ?>;
    let fatherhood2 = <?php echo
json_encode($_SESSION['user']['Fatherhood_teacher']); ?>;

    // Отримання дня, місяця та року
    let day = startDate.getDate();
    let month = startDate.getMonth() + 1; // Місяць від 0 до 11, тому додаємо 1
    let year = startDate.getFullYear();

    // Додаємо "0" перед одноцифровими числами
    let dayString = day < 10 ? '0' + day : day.toString();
    let monthString = month < 10 ? '0' + month : month.toString();
    let yearString = year.toString();

    // Використовуємо dayString, monthString, yearString для форматування дати
    let formattedDate = `${dayString}.${monthString}.${yearString}`;

    // Отримання дня, місяця та року
    let day2 = endDate.getDate();
    let month2 = endDate.getMonth() + 1; // Місяць від 0 до 11, тому додаємо 1
    let year2 = endDate.getFullYear();

    // Додаємо "0" перед одноцифровими числами
    let dayString2 = day2 < 10 ? '0' + day2 : day2.toString();
    let monthString2 = month2 < 10 ? '0' + month2 : month2.toString();
    let yearString2 = year2.toString();

    // Використовуємо dayString, monthString, yearString для форматування дати
    let formattedDate2 = `${dayString2}.${monthString2}.${yearString2}`;

    <?php
    $post_Id = $_SESSION['user']['Post_ID'];
    $Item_post = mysqli_query($connect, "SELECT `Post_name` FROM `Item_post`
WHERE `Post_ID` = '$post_Id'");
    $Item_post = mysqli_fetch_assoc($Item_post);
    ?>

    let post2 = <?php echo json_encode($Item_post); ?>;

    console.log('CollectedWord', CollectedWord);

```

```

//console.log('collectedData', collectedData);
for (let i = 0; i < collectedData.length; i++) {
  collectedData[i].additionalData = CollectedWord[4][i];
}
for (let i = 0; i < collectedData.length; i++) {
  collectedData[i].additionalData = CollectedWord[3][i];
  collectedData[i].points_type = points_type[i];
  collectedData[i].mark = mark[i];
}

let formula = [...formula1, ...formula2];
collectedData.forEach((item, index) => {
  if (formula[index] !== undefined) {
    item.mark = formula[index];
  }
});
console.log('rez', collectedData);
const Benchmark3 = <?php echo $Benchmark1; ?>;
const combinedArray = Benchmark3.map(item => {
  return [item[0] + '.', item[1]].join(' ');
});
//console.log(combinedArray);

const rezmasData = combinedArray.reduce((acc, current) => {
  acc[current] = []; // Adding two empty objects as values
  return acc;
}, {});

for (const item of collectedData) {
  const key = item.readonlyInput;
  if (rezmasData[key]) {
    rezmasData[key].push(item);
  }
}

for (const key in rezmasData) {
  rezmasData[key].forEach(obj => {
    delete obj.readonlyInput;
    delete obj.backgroundinformation;
  });
}
console.log(rezmasData);

// const CollectedWord1 = rezmasData.map(item => ({
//   backgroundinformation: item.backgroundinformation,

```

```

//  points: item.points,
//  readonlyInput: item.readonlyInput,
//  type: item.type
//  });
//console.log(points_type);
//console.log(mark);

const CollectedWord1 = [];

for (const key in rezmasData) {
  rezmasData[key].forEach(item => {
    CollectedWord1.push({ mainKey: key, ...item });
  });
}
console.log(CollectedWord1);
const CollectedWord2 = CollectedWord1.map(item => ({
  readonlyInput: item.readonlyInput,
  subtype: item.subtype,
  type: item.type,
  points: item.points,
  mark: item.mark,
  points_type: item.points_type,
  additionalData: item.additionalData
}));

CollectedWord1.forEach(item => {
  // Перевіряємо наявність поля subtype та типу його значення
  if (item.subtype && typeof item.subtype === 'string') {
    // Використовуємо регулярний вираз для витягування чисел
    let subtypeNumbers = item.subtype.match(/\d+(\.\d+)+/);
    // Додаємо новий ключ з витягнутим значенням
    if (subtypeNumbers) {
      item.subtypeNumbers = subtypeNumbers[0];
    }
  }

  // Перевіряємо наявність поля type та типу його значення
  if (item.type && typeof item.type === 'string') {
    // Використовуємо регулярний вираз для витягування чисел
    let typeNumbers = item.type.match(/\d+(\.\d+)+/);
    // Додаємо новий ключ з витягнутим значенням
    if (typeNumbers) {
      item.typeNumbers = typeNumbers[0];
    }
  }
});

```

```

const CollectedWord11 = CollectedWord1.map(item => {
  // Перевіряємо наявність поля subtype та типу його значення
  const subtype = item.subtype && typeof item.subtype === 'string' ?
item.subtype.replace(/\.\/g, '') : '';
  const type = item.type && typeof item.type === 'string' ?
item.type.replace(/\.\/g, '') : '';

  return {
    ...item,
    mark: isNaN(item.mark) ? "1" : item.mark,
    points_type: isNaN(item.points_type) ? "1" : item.points_type,
    subtype,
    type
  };
});
console.log('****', CollectedWord11);
const ArrayData = CollectedWord11.map(item => {
  const subtype = item.subtype && typeof item.subtype === 'string' ?
item.subtype.replace(/\d+/g, '') : '';
  const type = item.type && typeof item.type === 'string' ?
item.type.replace(/\d+/g, '') : '';

  return {
    ...item,
    subtype,
    type
  };
});
console.log('#', ArrayData);
console.log('****', CollectedWord);
const ArrayData2 = ArrayData.map(item => ({
  subtype: item.subtype,
  type: item.type,
  points: item.points,
  mark: item.mark,
  points_type: item.points_type,
  additionalData: item.additionalData,
  mainKey: item.mainKey,
  typeNumbers: item.typeNumbers

}));
// Render the document
doc.render({

  name: `${name2}`,
  surname: `${secondname2}`,

```

```

        fatherhood: `${fatherhood2}`,
        post: `${post2.Post_name}`,
        date_start: `${formattedDate}`,
        date_end: `${formattedDate2}`,
        thematicfocus: `${thematicFocus}`,
        sum: `${sum}`,

        ArrayData2: ArrayData2

    });

    const blob = doc.getZip().generate({
        type: "blob",
        mimeType: "application/vnd.openxmlformats-officedocument.wordprocessingml.document",
        compression: "DEFLATE",
    });

    // Збереження файлу в браузері
    let fileName = secondname2 + "_" + name2[0] + "." + fatherhood2[0] +
        "._підвищення_кваліф._" + formattedDate + "_" + formattedDate2 + ".docx";
    saveAs(blob, fileName);

    content = doc.getZip().generate({ type: "blob" });

    // Відправляємо Blob на сервер
    let formData = new FormData();
    formData.append('file', content, fileName); // 'report.docx' - ім'я файлу,
    яке ви хочете використовувати на сервері

    fetch('save.file.php', {
        method: 'POST',
        body: formData
    })
    .then(response => {
        if (response.ok) {
            return response.text();
        }
        throw new Error('Network response was not ok.');
```



```

    }
    else
    {
        //console.log('Помилка: Дата початку повинна бути раніше за дату кінця');
        alert('Помилка: Дата початку повинна бути раніше за дату кінця');
    }
}
else
{
    alert('Помилка: Не всі поля заповнені');
}
//console.log('Зібрані дані з усіх форм:', collectedData);
}
</script>
</body>
</html>

```

## view\_qualifications.php

```

<?php
session_start();
require_once "../configuration/connect.php";
$_SESSION["msg"] = "<div class = 'm'><p class = 'msg'>У вас ще немає звітів з підвищення
кваліфікації. Щоб сформувати звіт натисніть на кнопку нижче</p></div>";

// для безпеки сторінки, тільки для адміністраторів
if(!isset($_SESSION["user"]))
{
    header("Location: authorization.php");
}
?>

<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8" />
<meta http-equiv="X-UA-Compatible" content="IE=edge" />
<meta name="viewport" content="width=device-width, initial-scale=1.0" />
<link
    rel="stylesheet"

href="https://fonts.googleapis.com/css2?family=Manrope:wght@200;300;400;500;600;700;800&displa
y=swap"
/>
<link rel="icon" href="../img/logo.png" type="image/x-icon" />
<link

```

```

    type="text/css"
    href="style/view_qualifications.css"
    rel="stylesheet"
  />
<title>Ведення документації</title>
</head>

<body>
  <div class="all">
    <!-- Хедер -->
    <header class="header">
      <div class="img"></div>
      <p>Кафедра інформаційних технологій</p>
    </header>

    <!-- Основна частина сайту -->
    <div class="main">
      <!-- Меню сайту -->
      <div class="menu">
        <div class="photoperson">
          <?php
            if($_SESSION['user']['Photo'] != NULL)
            {
              echo '';
            }
            else
            {
              echo '';
            }
          ?>
        </div>
        <div class="menunumber">
          <ul>
            <li>
              <a href="person_information.php">
                
              </a>
            </li>
            <li>
              <a href="choice.php">
                
              </a>
            </li>
            <li>
              <a href="choiceview.php">
                
              </a>
            </li>
          </ul>
        </div>
      </div>
    </div>
  </body>
</html>

```

```

        </a>
    </li>
    <li>
        <a href="../configuration/exit.php">
            
        </a>
    </li>
</ul>
</div>
</div>
<div class="ma">
    <div class="name">
        <p>Звіти про підвищення кваліфікації</p>
    </div>

    <?php
    if($_SESSION['user'] != '')
    {
        $Teacher_ID = $_SESSION['user']['Teacher_ID'];
        $Qualification_doc_result = mysqli_query($connect, "SELECT
d.`Qualification_doc_ID`, d.`Qualification_doc_name`, d.`Qualification_path`
        FROM `Choice` c JOIN Qualification_doc d ON c.Qualification_doc_ID =
d.Qualification_doc_ID WHERE c.`Teacher_ID` = '$Teacher_ID'");

        // Отримання результату запиту
        if (mysqli_num_rows($Qualification_doc_result) > 0)
        {
            $Qualification_doc = mysqli_fetch_all($Qualification_doc_result);
            foreach($Qualification_doc as $name)
            { ?>

            <div class="doc">
                <div class="namedoc">
                    
                    <p><?php echo $name[1] ?></p>
                </div>
                <div class="action" title="Видалити">
                    <?php echo
                    '<a href = "vendor/delete_doc.php?id=' . $name[0] . '">
                        <button title="Видалити">'; ?>
                        
                    </button>
                    </a>
                    <?php echo
                    '<a href = "vendor/load.php?id=' . $name[0] . '">
                        <button class="load" title="Завантажити">
                        '?>
</button>
</a>
</div>
</div>

<?php }

}
else
{
    echo $_SESSION["msg"]; // Якщо таблиця порожня або немає результатів
}
}

?>

<div class="add">
    <div class="add1">
        <a href="creating_report.php">
            
        </a>
    </div>
</div>
</div>
</div>

<!-- Футер -->
<footer class="footer">
    <div class="location">
        <p class="imgloc">
            
        </p>
        <p class="adres">
            вул. Харківська, 116, Суми, Україна, 40007 Головний корпус, 13-й
            поверх, ауд. П1303
        </p>
    </div>

    <div class="phone">
        <div class="phone1">
            <p class="img">
                
            </p>
        </div>
    </div>

```

```

        <p class="pr">приймальня +38 (0542) 68-78-54</p>
    </div>
    <div class="phone2">
        <p class="img">
            
        </p>
        <p class="pr">завідувач кафедри +38 (0542) 68-78-63</p>
    </div>
</div>

<div class="email">
    <div class="insta">
        <a href="https://www.instagram.com/itp_sumdu?igshid=fopu9dqxdigb">
            
            <span>Instagram</span>
        </a>
    </div>
    <div class="em">
        <p class="img2"></p>
        <p class="adres">info@itp.sumdu.edu.ua</p>
    </div>
</div>
</footer>
</div>
</body>
</html>

```

## save.file.php

```

<?php
session_start();
require_once "../configuration/connect.php";

// Перевірка, чи був отриманий файл
if (isset($_FILES['file'])) {
    $file = $_FILES['file'];

    echo "Файл отримано. Ім'я файлу: " . $file['name'] . " Тимчасовий файл: " .
    $file['tmp_name'];

    // Встановлення шляху до кореневої папки проекту для збереження файла
    $uploadDirectory = __DIR__ . '/../files_word/';

```

```

$filePath = $uploadDirectory . basename($file['name']);

// Збереження файлу на сервері
if (move_uploaded_file($file['tmp_name'], $filePath)) {
    echo "Файл було успішно завантажено.";

    // Збереження інформації про файл в базу даних
    $Teacher_ID = $_SESSION['user']['Teacher_ID'];
    $Qualification_doc_name = basename($file['name']);
    $Qualification_path = '../files_word/' . $file['name'];

    // Вставка даних у таблицю Qualification_doc
    $stmt = $connect->prepare("INSERT INTO `Qualification_doc` (`Qualification_doc_name`,
`Qualification_path`) VALUES (?, ?)");
    $stmt->bind_param("ss", $Qualification_doc_name, $Qualification_path);

    if ($stmt->execute()) {
        echo "Інформацію про файл було успішно збережено в базі даних.";

        // Отримання ID щойно вставленого запису
        $Qualification_doc_ID = $connect->insert_id;
        echo "Новий Qualification_doc_ID: " . $Qualification_doc_ID;

        // Вставка даних у таблицю Choice
        $Schedule_ID = null;
        $stmt2 = $connect->prepare("INSERT INTO `Choice` (`Shedule_doc_ID`,
`Qualification_doc_ID`, `Teacher_ID`) VALUES (?, ?, ?)");
        $stmt2->bind_param("iii", $Schedule_ID, $Qualification_doc_ID, $Teacher_ID);

        if ($stmt2->execute()) {
            echo "Інформацію про файл було успішно збережено в базі даних.";
        } else {
            echo "Помилка збереження інформації про файл: " . $stmt2->error;
        }

        $stmt2->close();
    } else {
        echo "Помилка збереження інформації про файл: " . $stmt->error;
    }

    $stmt->close();
    $connect->close();
} else {
    echo "Помилка завантаження файлу.";
}
} else {
    echo "Файл не був отриманий.";
}

```

```
}
?>
```

## person\_information.php

```
<?php
session_start();
require_once "../configuration/connect.php";
?>

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <link
      rel="stylesheet"

href="https://fonts.googleapis.com/css2?family=Manrope:wght@200;300;400;500;600;700;800&displa
y=swap"
  />
    <link rel="icon" href="../img/logo.png" type="image/x-icon" />
    <link
      type="text/css"
      href="style/person_information.css"
      rel="stylesheet"
    />
    <title>Персональна інформація</title>
  </head>

  <body>
    <div class="all">
      <!-- Хедер -->
      <header class="header">
        <div class="img"></div>
        <p>Кафедра інформаційних технологій</p>
      </header>

      <!-- Основна частина сайту -->
      <div class="main">
        <!-- Меню сайту -->
        <div class="menu">
          <div class="photoperson">
            <?php
              if($_SESSION['user']['Photo'] != NULL)
```

```

    {
        echo '';
    }
    else
    {
        echo '';
    }
    ?>
</div>
<div class="menunumber">
    <ul>
        <li>
            <a href="person_information.php">
                
            </a>
        </li>
        <li>
            <a href="choice.php">
                
            </a>
        </li>
        <li>
            <a href="choiceview.php">
                
            </a>
        </li>
        <li>
            <a href="../../configuration/exit.php">
                
            </a>
        </li>
    </ul>
</div>
<div class="ma">
    <div class="name">
        <?php
            echo '<p>' . $_SESSION['user']['Second_name_teacher'] . ' ' .
$_SESSION['user']['First_name_teacher'] .
            ' ' . $_SESSION['user']['Fatherhood_teacher'] . '</p>'
        ?>
    </div>
    <div class="line1">
        <div class="name-line1"></div>
    </div>
    <div class="line2">

```



```

    <div class="name-line2"></div>
</div>
<div class="info">
    <div class="photo">
        <?php
            if($_SESSION['user']['Photo'] != NULL)
            {
                echo '';
            }
            else
            {
                echo '';
            }
        ?>
    </div>
    <div class="person_information">
        <?php
            $id = $_SESSION['user']['Teacher_ID'];
            $query = "SELECT p.Post_name FROM Teacher t JOIN Item_post p ON t.Post_ID =
p.Post_ID WHERE t.Teacher_ID = '$id'";
            $check_user = mysqli_query($connect, $query);

            if (!$check_user)
            {
                die('Query Error: ' . mysqli_error($connect));
            }

            $post = mysqli_fetch_assoc($check_user);

            if ($post)
            {
                echo '<p>Посада: ' . $post['Post_name'] . '</p>';
            }
            else
            {
                echo '<p>Посада не знайдена</p>';
            }

            if ($_SESSION['user']['Scientific_title'])
            {
                echo '<p>Вчений ступінь, наукове звання:<br />' .
$_SESSION['user']['Scientific_title'] . '</p>';
            }
            else
            {
                echo '<p>Вчений ступінь, наукове звання не знайдено</p>';
            }
        ?>
    </div>

```

```

    }

    if ($_SESSION['user']['Teacher_email'])
    {
        echo '<p>Email: ' . $_SESSION['user']['Teacher_email'] . '</p>';
    }
    else
    {
        echo '<p>Email не знайдено</p>';
    }

    if ($_SESSION['user']['Teacher_phone'])
    {
        echo '<p>Телефон: ' . $_SESSION['user']['Teacher_phone'] . '</p>';
    }
    else
    {
        echo '<p>Телефон не знайдено</p>';
    }
    ?>
</div>
</div>
</div>
</div>

<!-- Футер -->
<footer class="footer">
    <div class="location">
        <p class="imgloc">
            
        </p>
        <p class="adres">
            вул. Харківська, 116, Суми, Україна, 40007 Головний корпус, 13-й
            поверх, ауд. П1303
        </p>
    </div>

    <div class="phone">
        <div class="phone1">
            <p class="img">
                
            </p>
            <p class="pr">приймальня +38 (0542) 68-78-54</p>
        </div>
        <div class="phone2">
            <p class="img">
                
            </p>
        </div>
    </div>

```

```

        </p>
        <p class="pr">завідувач кафедри +38 (0542) 68-78-63</p>
    </div>
</div>

<div class="email">
    <div class="insta">
        <a href="https://www.instagram.com/itp_sumdu?igshid=fopu9dqxdigb">
            
            <span>Instagram</span>
        </a>
    </div>
    <div class="em">
        <p class="img2"></p>
        <p class="adres">info@itp.sumdu.edu.ua</p>
    </div>
</div>
</footer>
</div>
</body>
</html>

```

## delete\_doc.php

```

<?php
require_once "../configuration/connect.php";

$doc_id = $_GET["id"];

// Отримання шляху до файлу
$fileToDelete_res = mysqli_query($connect, "SELECT `Qualification_path` FROM
`Qualification_doc` WHERE `Qualification_doc_ID` = '$doc_id'");
$fileToDelete = mysqli_fetch_assoc($fileToDelete_res)['Qualification_path'];

// Перевірка існування файлу
if (file_exists("../" . $fileToDelete)) {
    // Спроба видалення файлу
    if (unlink("../" . $fileToDelete)) {
        echo "Файл було успішно видалено.";
    } else {
        echo "Помилка при видаленні файлу.";
    }
}

```

```

    }
} else {
    echo "Файл не існує.";
}

// Видалення запису з бази даних
mysqli_query($connect, "DELETE FROM `Qualification_doc` WHERE `Qualification_doc_ID` =
'$doc_id'");

mysqli_close($connect);
header("Location: ../view_qualifications.php");
?>

```

## load.php

```

<?php
require_once "../configuration/connect.php";

$doc_id = $_GET["id"];

// Отримання шляху до файлу
$file_res = mysqli_query($connect, "SELECT `Qualification_path`, `Qualification_doc_name` FROM
`Qualification_doc` WHERE `Qualification_doc_ID` = '$doc_id'");
$file = mysqli_fetch_assoc($file_res);

$filePath = "../" . $file['Qualification_path'];
$fileName = $file['Qualification_doc_name'];

// Перевірка існування файлу
if (file_exists($filePath)) {
    // Встановлення заголовків для завантаження файлу
    header('Content-Type: application/octet-stream');
    header('Content-Disposition: attachment; filename="' . $fileName . '"');
    header('Expires: 0');
    header('Cache-Control: must-revalidate');
    header('Pragma: public');
    header('Content-Length: ' . filesize($filePath));

    // Відправлення файлу користувачу
    readfile($filePath);
    exit;
} else {
    echo "Файл не існує.";
}

mysqli_close($connect);

```

?>

## **connect.php**

```
<?php
$connect = mysqli_connect("localhost", "root", "" , "db_maintaining_documentation");

if(!$connect)
{
    die("Помилка з'єднання з базою даних db_maintaining_documentation!");
}
?>
```

## **exit.php**

```
<?php
session_start();
unset($_SESSION["user"]);
session_unset();
header("Location: ../user/authorization.php");
?>
```

## ДОДАТОК Г

### Лістинг програмного коду створення сутностей в базі даних

```

CREATE TABLE Admin (
    Admin_ID INT AUTO_INCREMENT PRIMARY KEY,
    Name_admin VARCHAR(35) NOT NULL,
    Surname_admin VARCHAR(35) NOT NULL,
    Admin_log VARCHAR(40) NOT NULL,
    Admin_pass VARCHAR(40) NOT NULL);

CREATE TABLE Reference_table (
    Reference_table_ID INT AUTO_INCREMENT PRIMARY KEY,
    Reference_norm VARCHAR(50) NOT NULL
    Reference_value FLOAT NOT NULL);

CREATE TABLE Reference_edit (
    Reference_edit_ID INT AUTO_INCREMENT PRIMARY KEY,
    Date_edit DATE NOT NULL,
    Admin_ID INT NOT NULL,
    Reference_table_ID INT NOT NULL,
    FOREIGN KEY (Admin_ID) REFERENCES Admin(Admin_ID)
    ON UPDATE CASCADE ON DELETE CASCADE,
    FOREIGN KEY (Reference_table_ID) REFERENCES Reference_table(Reference_table_ID)
    ON UPDATE CASCADE ON DELETE CASCADE);

CREATE TABLE Item (
    Item_ID INT AUTO_INCREMENT PRIMARY KEY,
    Item_name VARCHAR(50) NOT NULL);

CREATE TABLE Kind_of_activity (
    Kind_of_activity_ID INT AUTO_INCREMENT PRIMARY KEY,
    Name_kind VARCHAR(50) NOT NULL);

CREATE TABLE Item_post (
    Post_ID INT AUTO_INCREMENT PRIMARY KEY,
    Post_name VARCHAR(50) NOT NULL);

CREATE TABLE Qualification_doc (
    Qualification_doc_ID INT AUTO_INCREMENT PRIMARY KEY,
    Qualification_doc_name VARCHAR(100) NOT NULL,
    Qualification_path VARCHAR(100) NOT NULL
);

CREATE TABLE Shedule_doc (
    Shedule_doc_ID INT AUTO_INCREMENT PRIMARY KEY,

```

```

    Shedule_doc_name VARCHAR(100) NOT NULL,
    Shedule_path VARCHAR(100) NOT NULL
);

CREATE TABLE Type_of_activity (
    Type_of_activity_ID INT AUTO_INCREMENT PRIMARY KEY,
    Name_type VARCHAR(50) NOT NULL,
    Kind_of_activity_ID INT NOT NULL,
    FOREIGN KEY (Kind_of_activity_ID) REFERENCES Kind_of_activity(Kind_of_activity_ID)
    ON UPDATE CASCADE ON DELETE CASCADE);

CREATE TABLE Activity (
    Activity_ID INT AUTO_INCREMENT PRIMARY KEY,
    Type_of_activity_ID INT NOT NULL,
    Item_ID INT NOT NULL,
    Name_activity VARCHAR(50),
    Hours_for_the_faculty INT NOT NULL,
    FOREIGN KEY (Type_of_activity_ID) REFERENCES Type_of_activity(Type_of_activity_ID)
    ON UPDATE CASCADE ON DELETE CASCADE,
    FOREIGN KEY (Item_ID) REFERENCES Item(Item_ID)
    ON UPDATE CASCADE ON DELETE CASCADE);

CREATE TABLE Teacher (
    Teacher_ID INT AUTO_INCREMENT PRIMARY KEY,
    Teacher_log VARCHAR(40) NOT NULL,
    Teacher_pas VARCHAR(40) NOT NULL,
    Scientific_title VARCHAR(50) NOT NULL,
    Photo VARCHAR(100),
    First_name_teacher VARCHAR(35) NOT NULL,
    Second_name_teacher VARCHAR(35) NOT NULL,
    Fatherhood_teacher VARCHAR(35) NOT NULL,
    Teacher_email VARCHAR(50),
    Teacher_phone CHAR(13) NOT NULL,
    Post_ID INT NOT NULL,
    FOREIGN KEY (Post_ID) REFERENCES Item_post(Post_ID)
    ON UPDATE CASCADE ON DELETE CASCADE);

CREATE TABLE Benchmark (
    Benchmark_ID INT PRIMARY KEY,
    Benchmark_name VARCHAR(150) NOT NULL);

CREATE TABLE Type_benchmark (
    Type_benchmark_ID INT AUTO_INCREMENT PRIMARY KEY,
    Type_benchmark_name VARCHAR(150) NOT NULL,
    Number_benchmark_name VARCHAR(10) NOT NULL,
    Benchmark_ID INT NOT NULL,
    Point INT,

```

```

Coefficient_name VARCHAR(50),
FOREIGN KEY (Benchmark_ID) REFERENCES Benchmark (Benchmark_ID)
ON UPDATE CASCADE ON DELETE CASCADE
);

CREATE TABLE Kind_benchmark (
    Kind_benchmark_ID INT AUTO_INCREMENT PRIMARY KEY,
Kind_benchmark_name VARCHAR(150) NOT NULL,
    Number_kind_benchmark VARCHAR(10) NOT NULL,
    Type_benchmark_ID INT NOT NULL,
    Mark INT,
    Mark_name VARCHAR(50),
    FOREIGN KEY (Type_benchmark_ID) REFERENCES Type_benchmark (Type_benchmark_ID)
    ON UPDATE CASCADE ON DELETE CASCADE
);

CREATE TABLE Choice (
    Choice_ID INT AUTO_INCREMENT PRIMARY KEY,
    Shedule_doc_ID INT,
    Qualification_doc_ID INT,
    Teacher_ID INT NOT NULL,
    FOREIGN KEY (Shedule_doc_ID) REFERENCES Shedule_doc (Shedule_doc_ID)
    ON UPDATE CASCADE ON DELETE CASCADE,
    FOREIGN KEY (Qualification_doc_ID) REFERENCES Qualification_doc (Qualification_doc_ID)
    ON UPDATE CASCADE ON DELETE CASCADE,
    FOREIGN KEY (Teacher_ID) REFERENCES Teacher (Teacher_ID)
    ON UPDATE CASCADE ON DELETE CASCADE
);

```