

# ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА

на тему:

## «Модель інформаційної системи сучасного закладу освіти на прикладі коледжу»

Студента 2м курсу, 23 групи,  
спеціальності 121 «Інженерія програмного забезпечення»  
освітньої програми «Інженерія програмного забезпечення»

\_\_\_\_\_

підпис студента

Давидчука  
Ігоря Васильовича

Науковий керівник  
кандидат педагогічних наук,  
доцент кафедри інженерії програмного забезпечення та кібербезпеки

\_\_\_\_\_

підпис керівника

Жирова Тетяна  
Олександрівна

Гарант освітньої програми  
кандидат педагогічних наук,  
доцент кафедри інженерії програмного забезпечення та кібербезпеки

\_\_\_\_\_

підпис гаранта

Котенко Наталія  
Олексіївна

Факультет інформаційних технологій

Кафедра інженерії програмного забезпечення та кібербезпеки

Освітній ступінь магістр

Освітня програма 121 «Інженерія програмного забезпечення»

### **Затверджую**

Зав. кафедри інженерії програмного  
забезпечення та кібербезпеки

Криворучко О. В.

«13» грудня 2022 р.

### **Завдання**

#### **на випускню кваліфікаційну роботу студентіві**

Давидчука Ігоря Васильовича

(прізвище, ім'я, по батькові)

1. Тема випускної кваліфікаційної роботи «Модель інформаційної системи сучасного закладу освіти на прикладі коледжу»

Затверджена наказом ректора від «09» грудня 2022 р. № 3339

2. Строк здачі студентом закінченої роботи 27 листопада 2023р.

3. Цільова установка та вихідні дані до роботи

Мета роботи: розробка інформаційної системи коледжу шляхом реалізації обміну даними між користувачем і сервером у реальному часі

Об'єкт дослідження: сучасні технології Веб-розробки

Предмет дослідження: проєкт інформаційної системи коледжу шляхом обміну даними між користувачем і сервером у реальному часі.

4. Консультанти роботи із зазначенням розділів, які консультують:

Розділ	Консультант (прізвище, ініціали)	Підпис, дата	
		Завдання видав	Завдання прийняв

5. Зміст випускної кваліфікаційної роботи (перелік питань за кожним розділом)

ВСТУП

## РОЗДІЛ 1 ТЕОРЕТИЧНІ ОСНОВИ СТВОРЕННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ СУЧАСНОГО ЗАКЛАДУ ОСВІТИ

1.1. Огляд методів вирішення задачі

1.2. Обґрунтування мети рішення поставленої задачі та критеріїв ефективності

1.3. Постановка задачі

1.4. Висновки до розділу 1

## РОЗДІЛ 2 РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ КОЛЕДЖУ

2.1. Інформаційне забезпечення проєктованої системи

2.1.1 Структура і схеми інформаційних об'єктів і ресурсів

2.1.2 Схеми інформаційних потоків

2.1.3 Схеми бази даних

2.1.4 Модель бази даних

2.2 Розробка алгоритмів рішення функціональної задачі

2.3 Програмне забезпечення

2.3.1 Опис структури програми

2.3.2 Оцінка працездатності проєктованої системи

2.4 Висновок до розділу 2

## РОЗДІЛ 3 ОПИС РОБОТИ ІНФОРМАЦІЙНОЇ СИСТЕМИ КОЛЕДЖУ

3.1. Опис працездатності системи

3.2. Заходи щодо безпечного використання системи

3.3. Висновок до розділу 3

## ВИСНОВКИ ТА ПРОПОЗИЦІЇ

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

## ТЕХНІЧНЕ ЗАВДАННЯ

## ПРОГРАМА ТА МЕТОДИКА ТЕСТУВАННЯ

## ДОДАТКИ

## 6. Календарний план виконання роботи

№ пор.	Назва етапів випускної кваліфікаційної роботи	Строк виконання етапів роботи	
		за планом	фактично
1	2	3	4
1.	<i>Вибір теми випускної кваліфікаційної роботи</i>	07.11.2022	
2.	<i>Розробка та затвердження завдання на роботу магістра (стац/заоч)</i>	13.12.2022	
3.	<i>Вступ та перелік літературних джерел</i>	24.02.2023	
4.	<i>Розробка технічного завдання</i>	15.03.2023	
5.	<i>Розділ 1. Теоретичні основи створення інформаційної системи сучасного закладу освіти</i>	10.04.2023	
6.	<i>Розділ 2. Розробка інформаційної системи коледжу</i>	24.05.2023	
7.	<i>Розділ 3. Опис роботи інформаційної системи коледжу</i>	06.09.2023	
8.	<i>Розробка програми та методики тестування</i>	18.10.2023	
9.	<i>Написання наукової статті</i>	17.05.2023	
10.	<i>Керівництво користувача</i>	25.10.2023	
11.	<i>Висновки та пропозиції</i>	01.11.2023	
12.	<i>Здача випускної кваліфікаційної роботи на кафедрі (перша перевірка)</i>	06.11.2023	
13.	<i>Підготовка автореферату та презентації доповіді</i>	06.11.2023	
14.	<i>Попередній захист випускної кваліфікаційної роботи</i>	20.11.2023 – 24.11.2023	
15.	<i>Здача зброшурованої випускної кваліфікаційної роботи</i>	01.12.2023	
16.	<i>Зовнішнє рецензування випускної кваліфікаційної роботи</i>	02.12.2023	
17.	<i>Підготовка до публічного захисту випускної кваліфікаційної роботи</i>	05.12.2023- 06.12.2023	

7. Дата видачі завдання «13» грудня 2022 р.

8. Науковий керівник випускної кваліфікаційної роботи \_\_\_\_\_

Жирова Т.О.

(прізвище, ініціали, підпис)

9. Гарант освітньої програми \_\_\_\_\_

Котенко Н.О.

(прізвище, ініціали, підпис)

10. Завдання прийняв до виконання студент \_\_\_\_\_

Давидчук І.В.

(прізвище, ініціали, підпис)



## АНОТАЦІЯ

Відповідно до мети дослідження робота присвячена проектуванню інформаційної системи коледжу шляхом реалізації обміну даними між користувачем і сервером у реальному часі. Випускна кваліфікаційна робота на тему «Модель інформаційної системи сучасного закладу освіти на прикладі коледжу» містить 56 сторінок друкованого тексту, 19 рисунків, 10 таблиць та 11 додатків і налічує 29 джерел використаної літератури.

В наш час практично кожна інформаційна система, що має власний веб-сайт у Інтернеті базується на передових методологіях. Існує велика кількість веб-технологій, які застосовуються для створення та організації інформаційних платформ. Зараз популярним методом розробки веб-додатків, орієнтованих на обмін інформацією та документацією, є використання технологій, таких як HTML 5, Ajax, PHP та Java, із можливістю динамічної взаємодії з базами даних MySQL. Сучасні онлайн-магазини, чати та системи обміну документами та інформаційні системи в основному базуються на цих технологіях. Асинхронна передача даних стає практично нормою у більшості веб-проектів, оскільки це сприяє ефективності та розширенню можливостей ресурсу. Однією з ключових технологій є Ajax, який дозволяє користувачам взаємодіяти з веб-сайтом, надсилаючи запити на сервер без повного перезавантаження веб-сторінки. Простота в реалізації нових технологій дозволяє розробникам використовувати готові бібліотеки функцій у вигляді фреймворків, спрощуючи тим самим роботу з кодом і роблячи його більш доступним для розробника. В результаті розроблено систему управління документацією для коледжу, базуючись на модулях для обміну даними між користувачем і сервером, адміністративному модулі та модулі новин. Таким чином, ця тема була обрана для реалізації поставленого завдання.

**Ключові слова:** Веб-проектування, облік документів, база даних, технологія проектування, технологія «клієнт-сервер», об'єктно-орієнтоване програмування, HTML-5, PHP, Java, MySQL, Ajax.

## ABSTRACT

According to the research objective, the work is dedicated to the design of an information system for a college by implementing real-time data exchange between the user and the server. The graduation thesis on the topic "Model of an Information System for a Modern Educational Institution: A College Case Study" consists of 56 pages of printed text, 19 figures, 10 tables, and 11 appendices, and references 29 sources of literature.

In today's world, virtually every information system with its own website on the Internet is based on advanced methodologies. There is a wide array of web technologies used to create and organize information platforms. Presently, a popular method for developing web applications focused on data and document exchange is the use of technologies such as HTML 5, Ajax, PHP, and Java, with the capability of dynamic interaction with MySQL databases. Modern online stores, chat systems, document exchange systems, and information systems primarily rely on these technologies. Asynchronous data transmission has become the norm in most web projects, as it enhances efficiency and expands the capabilities of the resource.

One of the key technologies is Ajax, which allows users to interact with a website by sending requests to the server without fully reloading the web page. This contrasts with the classical model, where the entire page is reloaded during data exchange between the server and the client. The simplicity of implementing these new technologies enables developers to utilize ready-made libraries of functions in the form of frameworks, making the code more accessible and easier to work with for developers. According to the purpose of the study,

As a result, a document management system for the college was developed, based on modules for data exchange between the user and the server, an administrative module, and a news module. Thus, this topic was chosen for the implementation of the assigned task.

**Keywords:** Web design, document management, database, design technology, client-server technology, object-oriented programming, HTML-5, PHP, Java, MySQL, Ajax.





## ЗМІСТ

<b>ВСТУП.....</b>	<b>3</b>
<b>РОЗДІЛ 1 ТЕОРЕТИЧНІ ОСНОВИ СТВОРЕННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ СУЧАСНОГО ЗАКЛАДУ ОСВІТИ. ....</b>	<b>7</b>
1.1. Огляд методів вирішення задачі.....	7
1.2. Обґрунтування мети рішення поставленої задачі та критеріїв ефективності. ....	8
1.3. Постановка задачі. ....	10
1.4. Висновки до розділу 1.....	11
<b>РОЗДІЛ 2 РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ КОЛЕДЖУ. ....</b>	<b>12</b>
2.1. Інформаційне забезпечення проектованої системи. ....	12
2.1.1 Структура і схеми інформаційних об'єктів і ресурсів.....	12
2.1.2 Схеми інформаційних потоків.....	14
2.1.3. Схеми бази даних.....	17
2.1.4 Модель бази даних.....	20
2.2. Розробка алгоритмів рішення функціональної задачі.....	21
2.3. Програмне забезпечення.....	22
2.3.1. Опис структури програми.....	22
2.3.2. Оцінка працездатності проектованої системи.....	42
2.5. Висновки до розділу 2.....	44
<b>РОЗДІЛ 3 ОПИС РОБОТИ ІНФОРМАЦІЙНОЇ СИСТЕМИ КОЛЕДЖУ.....</b>	<b>45</b>
3.1. Опис працездатності системи.....	45
3.2. Заходи щодо безпечного використання системи.....	48
3.4. Висновок до розділу 3.....	52
<b>ВИСНОВКИ ТА ПРОПОЗИЦІЇ.....</b>	<b>53</b>
<b>СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....</b>	<b>54</b>
<b>ТЕХНІЧНЕ ЗАВДАННЯ.....</b>	<b>57</b>
<b>ПРОГРАМА ТА МЕТОДИКА ТЕСТУВАННЯ.....</b>	<b>59</b>
<b>ДОДАТКИ.....</b>	<b>62</b>

					<i>ДТЕУ 121 023-4.МР</i>			
<i>Зм.</i>	<i>Аркуш</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>	<i>Модель інформаційної системи сучасного закладу освіти на прикладі коледжу</i>	<i>Стадія</i>	<i>Аркуш</i>	<i>Аркушів</i>
Зав. каф.	Криворучко О.В.			01.11.23		<i>Зміст</i>	3	57
Керівник	Жирова Т.О.			01.11.23		<i>Факультет інформаційних технологій</i>		
Гарант	Котенко Н.О.			01.11.23		<i>2м курс, 2з група</i>		
Розробив	Давидчук І.В.			01.11.23				
					<i>Зміст</i>			

## ВСТУП

У сучасний період погляд на системи управління інформацією відчутно еволюціонував, особливо з початком швидкого росту в інформаційних технологіях. Прискорення розвитку глобальної мережі Інтернет призвело до кардинальних змін, які, в свою чергу, відкрили шлях до новаторських підходів у сфері обміну даними та їхньому використанні.

**Актуальність теми роботи.** В наш час практично кожна інформаційна система, що має власний веб-сайт у Інтернеті базується на передових методологіях. Існує велика кількість веб-технологій, які застосовуються для створення та організації інформаційних платформ. Зараз популярним методом розробки веб-додатків, орієнтованих на обмін інформацією та документацією, є використання технологій, таких як HTML 5, Ajax, PHP та Java, із можливістю динамічної взаємодії з базами даних MySQL. Сучасні онлайн-магазини, чати та системи обміну документами та інформаційні системи в основному базуються на цих технологіях. Асинхронна передача даних стає практично нормою у більшості веб-проектів, оскільки це сприяє ефективності та розширенню можливостей ресурсу [1].

Однією з ключових технологій є Ajax, який дозволяє користувачам взаємодіяти з веб-сайтом, надсилаючи запити на сервер без повного перезавантаження веб-сторінки. Це контрастує з класичною моделлю, де відбувається повне перезавантаження сторінки під час обміну даними між сервером та клієнтом. Простота в реалізації нових технологій дозволяє розробникам використовувати готові бібліотеки функцій у вигляді

Зм.	Аркуш	№ докум.	Підпис	Дата	<i>ДТЕУ 121 023-4.МР</i>			
Зав. каф.	Криворучко О.В.			24.02.23	<i>Модель інформаційної системи сучасного закладу освіти на прикладі коледжу</i>	Стадія	Аркуш	Аркушів
Керівник	Жирова Т.О.			24.02.23		В	4	57
Гарант	Котенко Н.О.			24.02.23		Факультет інформаційних технологій		
Розробив	Давидчук І.В.			24.02.23		2м курс, 2з група		
					<i>Вступ</i>			

фреймворків, спрощуючи тим самим роботу з кодом і роблячи його більш доступним для розробника. У даному дипломному проєкті розроблено систему управління документацією для коледжу, базуючись на модулях для обміну даними між користувачем і сервером, адміністративному модулі та модулі новин. Таким чином, ця тема була обрана для реалізації [10].

**Метою дослідження** є проєктування інформаційної системи коледжу шляхом реалізації обміну даними між користувачем і сервером у реальному часі.

**Об'єктом дослідження** є сучасні технології Веб-розробки.

**Предметом дослідження** є проєкт інформаційної системи коледжу шляхом обміну даними між користувачем і сервером у реальному часі.

Для виконання сформульованої мети дослідження, нами були поставлені **наступні завдання**:

1. Виконати постановку завдання та навести її характеристику.
2. Виконати розробку технічного рішення.
3. Провести проєктування технічного рішення.

**Методи дослідження** в роботі використані такі: пошуковий по наявній методичній та науковій літературі із аналізом знайденого матеріалу, порівняння, проєктування, теоретичне моделювання, аналіз документації та результатів діяльності дослідників з проблеми проведеного дослідження.

**Джерелами інформації** для вирішення перерахованих вище завдань є збірники наукових праць, монографії, періодична література, підручники та довідники, періодичні фахові журнали.

**Теоретична та практична цінність** роботи полягає в наявності теоретичного матеріалу по дослідженню, відсіяного з-поміж іншого в процесі пошуку інформації по темі, та в систематизації матеріалу напрямку дослідження. Проведене дослідження має більш глибокий ступінь розробки

						Аркуш
						6
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 02з-4.МР	

напрямку дослідження, відносно попередніх досліджень вчених і містить узагальнення та власні розробки коду.



						Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 02з-4.МР	6

## РОЗДІЛ 1

# ТЕОРЕТИЧНІ ОСНОВИ СТВОРЕННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ СУЧАСНОГО ЗАКЛАДУ ОСВІТИ.

### 1.1. Огляд методів вирішення задачі

Сучасні ІТ-технології сприяють розробці передових мережевих систем для ефективного обміну даними, при цьому спрощуючи життя розробникам. Ці системи представляють собою комплекс веб-технологій, який надає можливість управління великими даними через інтуїтивний користувацький інтерфейс. Аналіз сучасних інформаційних ресурсів на задану тему дипломної роботи виконувався за допомогою інтернет-сервісів, зокрема Google та Wikipedia. Після проведеного огляду можна констатувати, що всі ресурси надають актуальну та вірогідну інформацію, яка регулярно оновлюється, підкреслюючи важливість та актуальність обраної теми. Для оцінки релевантності теми було проведено дослідження сайтів, які активно інтегрують та використовують підхід Ajax у своїх розробках. Основний фокус проекту спрямований на технологію Ajax, впроваджену в дизайн сайту. Для зіставлення нашого проекту були вивчені готові рішення, які оптимізовані для використання Ajax [3].

Ajax, який стоїть за скороченням «Асинхронний JavaScript та XML», – це методика розробки користувацьких інтерфейсів для веб-додатків, яка дозволяє здійснювати фонові запити до сервера без перезавантаження веб-сторінки, для того, щоб динамічно оновлювати контент. Цей підхід є частиною концепції DHTML.

					<i>ДТЕУ 121 023-4.МР</i>			
Зм.	Аркуш	№ докум.	Підпис	Дата				
Зав. каф.		Криворучко О.В.		10.04.23	Модель інформаційної системи сучасного закладу освіти на прикладі коледжу	Стадія	Аркуш	Аркушів
Керівник		Жирова Т.О.		10.04.23		P1	7	57
Гарант		Котенко Н.О.		10.04.23		Факультет інформаційних технологій 2м курс, 2з група		
Розробив		Давидчук І.В.		10.04.23				

Розмови про Аїах почалися після публікації статті Джессі Джеймса Гарретта у лютому 2005 року під назвою «Новий підхід до веб-додатків». Важливо зрозуміти, що Аїах не є автономною технологією. Його широке застосування в області веб-розробки свідчить про значуще поліпшення та спрощення роботи для розробників [9].

Основа Аїах-методології:

–це не автономна технологія, а скоріше сполучення кількох тісно пов'язаних технологій. Методика розробки Аїах для користувачьких інтерфейсів включає в себе декілька ключових стратегій та практик:

–застосування DHTML для інтерактивної модифікації вмісту веб-сторінки;

–використання XMLHttpRequest для асинхронного взаємодію з сервером без повного перезавантаження сторінки;

–додатковий варіант – динамічне завантаження JavaScript коду в теги, використовуючи формат JSON;

–динамічне генерування вкладених фреймів.

Традиційна модель веб-додатку:

–користувач відвідує веб-сторінку і клікає на одному з елементів;

–браузер відправляє запит на сервер;

– сервер генерує і відсилає браузеру цілком нову веб-сторінку або ж лише її фрагменти [4].

## 1.2. Обґрунтування мети рішення поставленої задачі та критеріїв ефективності.

Інтерактивність веб-сайту є ключовим елементом його дизайну, який впливає на взаємодію користувача з платформою. В традиційних системах користувач заповнює форму і очікує, поки повністю перезавантажиться сторінка, що може призвести до помилок і необхідності повторного введення

						Аркуш
					ДТЕУ 121 02з-4.МР	8
Зм.	Аркуш	№ докум	Підпис	Дата		

інформації. За допомогою технології Аїах зміни відбуваються динамічно відповідно до дій користувача, дозволяючи йому залишитися на тій же сторінці.

У старих моделях взаємодії клієнт-сервер запит від користувача призводив до повної передачі веб-сторінки, що збільшувало використання трафіку. Аїах оптимізує цей процес, оновлюючи лише конкретну частину сторінки, що змінилася. Це зменшує навантаження на сервер і скорочує обсяг передаваних даних [3].

Сучасні веб-сайти тепер пропонують безліч функцій, які можна використовувати без повного оновлення веб-сторінки, завдяки Аїах. Це дозволяє користувачам взаємодіяти з мультимедійними платформами та виконувати різні операції з завантаженими об'єктами без повного оновлення сторінки. Це веде до принципово нового рівня якості веб-сайтів та динаміки інтерфейсу.

Наприклад, при створенні форми для вводу дати можна використовувати скрипт, який негайно відправляє запит на сервер і надає користувачеві актуальні підказки, обмежуючи вибір дат, що є реалістичними на сьогоднішній день.

Використання технології Аїах зменшує навантаження на веб-сервер, оскільки оновлюється лише окрема частина веб-сторінки, а не вся сторінка. Це зменшує обсяг передаваних даних і, відповідно, знижує навантаження на сервер. Таким чином, користувач отримує лише ті дані, які йому потрібні, що покращує загальний досвід користувача [11].

Недоліки технології Аїах включають:

1. Відсутність сумісності зі стандартними інструментами браузера, такими як кнопка «Назад». Так як динамічні сторінки не враховуються в історії браузера, функція «Назад» не працює належним чином. Проте, багато розробників створюють додаткові скрипти для виправлення цього недоліку.

						Аркуш
						9
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 02з-4.МР	

2. Закладки: Оскільки сторінки генеруються динамічно за допомогою JavaScript, а не сервером, адреса сторінки не може бути використана для створення закладки. Для рішення цієї проблеми, у Ajax часто використовують перманентні посилання.

3. Індексція пошуковими системами: Динамічний контент, який генерується на стороні клієнта, не може бути індексований пошуковими роботами. Це вимагає створення статичних аналогів для доступу до контенту.

4. Складність проєкту: Впровадження Ajax в значущі веб-проєкти може додати складність у розробці та інтеграції. Це також може ускладнити передачу даних великого обсягу, що може бути вирішено за допомогою файлів cookie [13].

### 1.3. Постановка задачі.

Планований проєкт – це інформаційна веб-система для обміну даними між користувачами та сервером, зосереджена на потребах коледжу і базована на технології Ajax для асинхронної комунікації. Система буде використовувати сучасні методи та технології програмування для створення ефективних та інтерактивних веб-сайтів.

Основна сфера дослідження проєкту охоплює використання передових веб-технологій, які спрощують роботу розробників і забезпечують високу функціональність веб-ресурсів. Ajax є ключовою технологією в цьому контексті, яка вже застосовується у великих інтернет-проєктах, таких як Google, Facebook та інші.

Це підтверджує актуальність та ефективність використання Ajax для сучасних веб-застосунків. Інтерактивні функції, такі як відправка даних без перезавантаження сторінки, роблять цю технологію незамінною в сучасних веб-сервісах [12].

						Аркуш
						10
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 02з-4.МР	



Фінальний продукт – це інтегрована веб-платформа для коледжу, яка дозволить користувачам зручно знаходити та управляти документацією. Система буде складатися з взаємопов'язаних модулів, які працюють синхронно для забезпечення цілісної функціональності.

#### 1.4. Висновки до розділу 1.

Сучасні ІТ можливості полегшують роботу розробників, дозволяючи їм створювати високотехнологічні платформи для обміну інформацією в інтернеті. Така система, що використовує різні веб-технології, забезпечує ефективне управління великими даними та комфорт для користувачів. За результатами огляду літератури та онлайн ресурсів, тема є актуальною і динамічно оновлюється.



Для оцінки важливості теми досліджено сайти, які активно використовують технологію Ажах. Основна складова проекту – це інтеграція Ажах, тому було проведено аналіз аналогічних рішень. Ажах – це не окрема технологія, а набір методів та підходів, що включає в себе використання DHTML, XMLHttpRequest та інших технік для створення динамічних веб-інтерфейсів.

						Аркуш
						11
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 02з-4.МР	

## РОЗДІЛ 2

### РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ КОЛЕДЖУ.

#### 2.1. Інформаційне забезпечення проектованої системи.

##### 2.1.1 Структура і схеми інформаційних об'єктів і ресурсів

Створення веб-додатку починається з розробки інформаційної моделі сайту. Тому в технічному завданні на створення офіційного веб-додатку необхідно вказати вимоги до інформаційного наповнення та завдання, які мають бути вирішені через створення офіційного сайту. При розробці інформаційної моделі веб-додатку слід керуватися принципом, що інформаційний зміст та структура веб-додатку повинні відображати структуру проектованого сервісу.

Оцінка будь-якої офіційної веб-сторінки може бути проведена за двома параметрами: змістом та зовнішньою формою. Однак спочатку необхідно визначити, яка інформація потребує розміщення на ньому. Необхідно детально проаналізувати, скільки та якої інформації потребує розміщення на офіційному сайт. Офіційна веб-сторінка повинна стати одним з зручних механізмів сполучення з клієнтами.

Аналіз наукової літератури дозволяє нам систематизувати можливі підходи до організації сайту на такі способи [18]:

*Стандартна модель:* Головна сторінка містить посилання на решту контенту, і навпаки, документи повертаються до головної сторінки. Це є найпростішим та найбільш розповсюдженим методом структурування веб-інтерфейсу інформаційної системи. Зображення демонструє цей базовий підхід до організації веб-сторінки (рис. 2.1а).

Зм.	Аркуш	№ докум.	Підпис	Дата	ДТЕУ 121 023-4.МР			
Зав. каф.		Криворучко О.В.		24.05.23	Модель інформаційної системи сучасного закладу освіти на прикладі коледжу	Стадія	Аркуш	Аркушів
Керівник		Жирова Т.О.		24.05.23		P2	10	57
Гарант		Котенко Н.О.		24.05.23		Факультет інформаційних технологій 2м курс, 2з група		
Розробив		Давидчук І.В.		24.05.23				
					Розробка інформаційної системи коледжу			

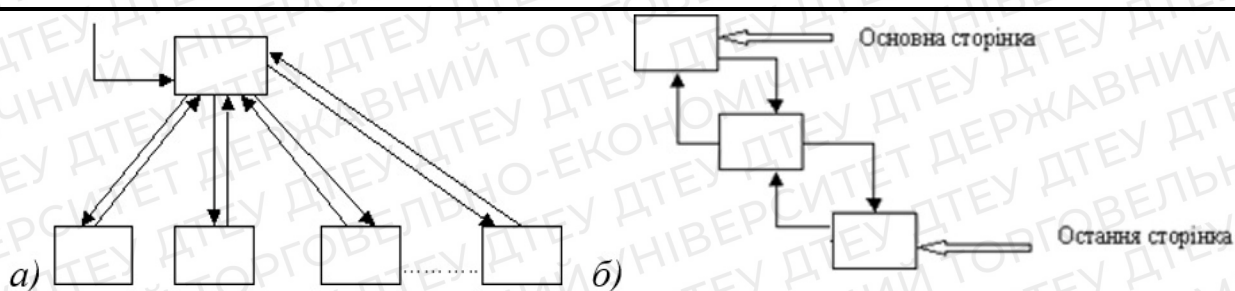


Рис. 2.1. Стандартний – а) та каскадний спосіб організації веб-сайту – б)

**Каскад.** У цьому випадку посилання у документах задані таким чином, що існує тільки один шлях обходу сторінок веб-інтерфейсу інформаційної системи. На (рис. 2.1а) нижче показаний приклад каскадного сайту.

В методі каскадної організації сторінок, користувачі можуть рухатися лише в одному з двох напрямків: або вперед, або назад.

У моделі «Хмарочос», користувачі можуть дістатися певних сторінок лише через конкретний шлях. Це схоже на підняття на визначений поверх в висотному будинку. Зображення на (рис. 2.2а) відображає цю структуру, яка схожа на план великого хмарочосу.

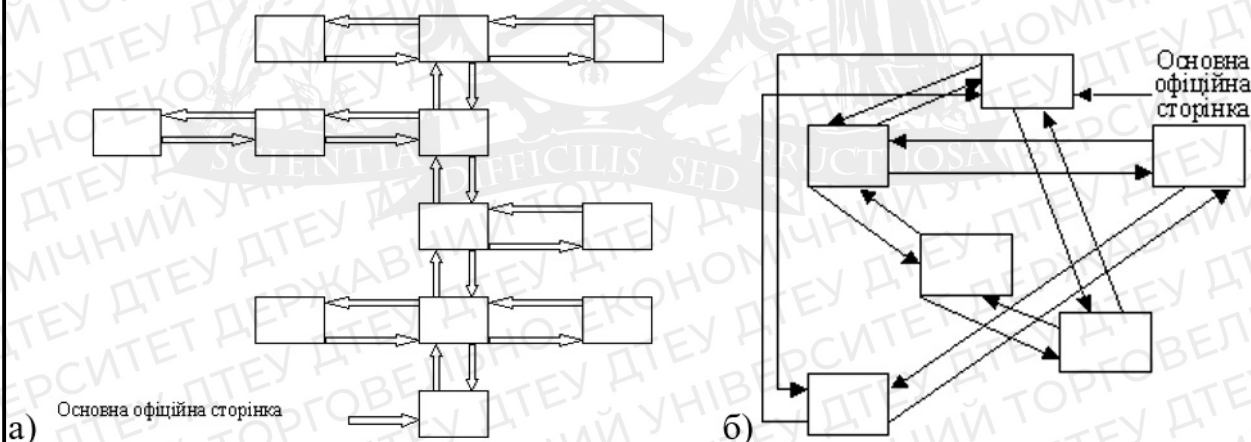


Рис. 2.2. Схема організації веб-сайту «хмарочос» - а) та «павутинка»

**«Павутинка».** У цій моделі, кожна сторінка веб-додатку містить посилання на інші сторінки, дозволяючи користувачу легко переходити з однієї сторінки на іншу. Цей підхід може стати заплутаним і важким для навігації, якщо не буде належним чином керовано, але є корисним, коли посилання на сторінки не використовуються дуже часто. (рис 2.2б) ілюструє простий варіант такої структури.

						Аркуш
					ДТЕУ 121 02з-4.МР	13
Зм.	Аркуш	№ докум	Підпис	Дата		

Мовна єдність. Всю інформацію на сайті слід подавати на одній мові, але головна сторінка має надавати можливість зміни мови. Вибір мови може здійснюватися за допомогою графічних символів (у форматі GIF), замість текстових підписів. Це полегшує вибір мови в середовищах, які не підтримують певне кодування символів, наприклад, для користувачів у Європі чи Америці [15, 16, 18].

Технічна сторона інтерфейсу в основному заснована на використанні Ajax. Ця технологія дозволяє автоматично оновлювати та синхронізувати дані без необхідності ручного оновлення користувачем. Ajax може взаємодіяти з веб-сервером «на льоту», не перезавантажуючи весь сайт.

HTML і CSS використовуються для створення базової структури та оформлення, в той час як Ajax і DOM дозволяють змінювати цю структуру та зміст «на льоту», без перезавантаження всієї сторінки. Це може бути особливо корисно в системах, що потребують активної взаємодії з користувачем, наприклад, в системах обліку документації [7].

Така архітектура не тільки забезпечує високий рівень взаємодії користувача, але й може полегшити подальшу розробку та супроводження веб-додатку. Система, побудована на таких принципах, буде більш адаптивною до змін і може легко масштабуватися.

### 2.1.2 Схеми інформаційних потоків

Функціонування веб-інтерфейсу системи управління документацією коледжу, засноване на технології Ajax, організоване наступним чином:

- користувач відвідує веб-сайт і клікає на конкретний елемент на сторінці;
- його браузер надсилає запит до сервера;
- відповідь сервера містить лише оновлену частину документа.

						Аркуш
						14
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 02з-4.МР	

Щоб наочно продемонструвати, як працює цей веб-інтерфейс з Ажах, ми представимо схему. Основним фокусом буде детальний показ процесу Ажах-запитів та їх фонові обробки на сервері. (Рис 2.3) нижче ілюструє цей процес.

Отже, всі користувацькі запити спочатку проходять через обробник JavaScript, після чого вони відправляються для додаткової обробки за допомогою HTTP. Отримані дані оновлюються в моделі DOM, дозволяючи користувачу бачити оновлення лише в конкретних розділах сайту, де це необхідно. Взаємодія з веб-сервером відбувається через Ажах-обробник і направляє на відповідний порт сервера згідно із заданим сценарієм. Важливо відзначити, що DOM-модель базується на HTML-розмітці та CSS-стилях<sup>8</sup>].

Щодо збереження даних у нашому проєкті, буде використана реляційна база даних MySQL. Інформація буде зберігатися в таблицях, які містять рядки та стовпці. Вся система буде мати дві основні таблиці: «user» для зберігання інформації про користувачів та їх статуси в системі, і «message» для управління всіма заявками в системі обміну повідомленнями. Між цими таблицями буде створено реляційний зв'язок на основі користувацьких ідентифікаторів. Кожен користувач буде мати унікальний ідентифікатор для ідентифікації в системі. Після реєстрації користувача, всі його дії будуть асоційовані з цим унікальним ідентифікатором. Це також означає, що всі повідомлення або заявки будуть зберігатися з відповідними ідентифікаторами в таблиці «message».

						Аркуш
						15
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 02з-4.МР	



Рис. 2.3. Робота системи з використанням Ajax

Структура таблиці `user` складається з різних полів, кожне з яких має свій тип даних для індивідуального запису. З іншого боку, таблиця `message` присвячена зберіганню користувацьких документів, включаючи текст, ідентифікатори користувачів та шляхи до серверних файлів. Робота з базою даних MySQL проста та ефективна, зокрема завдяки використанню реляційних таблиць, які спрощують пошук та ідентифікацію даних.

Відносно розробки веб-інтерфейсу для системи управління документами коледжу, основа будується на HTML. В нашому випадку, 'верстка' сайту є першочерговим кроком. Верстка зазвичай передбачає створення блоків, які потім заповнюються змістом, часто графікою або текстом. HTML є ключовим інструментом для створення структури веб-сторінки і є універсальним способом дизайну веб-застосунків. Цей етап також включає створення всіх гіперпосилань для сайту.

Загалом, користувач взаємодіє з HTML-документами, які він може завантажити і переглянути в своєму веб-браузері. Структура цих сторінок залежить від планування та верстки.

Структура веб-сторінки включатиме три ключові компоненти:

1. Вказівка на тип документа на початку файла, що визначається за допомогою DTD.

						Аркуш
						16
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 02з-4.МР	

2. Мета-інформація у секції 'head', яка містить технічні деталі або додаткові дані про веб-сторінку, які не показуються користувачу.

3. Основний зміст у секціях 'body' або 'frameset', де розміщується головний контент.

Слідом за цим, вся структура буде включати вказівник на тип, мета-інформацію та основний зміст.

В розробці веб-інтерфейсу буде використовуватись PHP як основний інструмент для створення динаміки на сайті [12].

Синхронні запити до сервера дозволять оновлювати інформацію на сторінці без її перезавантаження. Індикатори документів та статусу користувача будуть автоматично оновлюватись за допомогою JavaScript і jQuery. Система буде періодично запитувати сервер про нові дані, зокрема про статус користувача (онлайн чи офлайн). Якщо користувач зареєстрований або авторизований в системі, його статус буде «онлайн», в протилежному випадку – «офлайн». Індикатор документів показує кількість нових документів для користувача в системі [17].

### 2.1.3. Схеми бази даних

Під час розробки програмного продукту характеристики об'єктів зберігаються у базі даних, тоді як їх функціональність реалізується в програмі через методи, які реагують на користувацькі події.

Створювана база даних планується таким чином, щоб бути цільовою та універсальною, не залежачи від конкретного типу бази даних, ОС або апаратних ресурсів. Основна вимога до неї полягає в точному представленні предметної області. Модель повинна бути консистентною та враховувати потреби всіх користувачів.

Ця модель бази даних відображає інформаційні потоки, об'єкти та їх взаємозв'язки в рамках даної предметної області. Модель може бути представлена у форматі ER-діаграми та реляційної схеми. Метою цієї

						Аркуш
						17
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 02з-4.МР	

структури є ідентифікація та інтеграція користувацьких інформаційних потреб, незалежно від типу даних або місця їх зберігання. Модель має бути легко масштабованою, дозволяючи додавання нових даних без необхідності зміни вже існуючих елементів. Центральний елемент цієї моделі – це опис об’єктів і їх взаємозв’язків, зазвичай представлений у формі ER-моделі [13].

Завданням моделювання є створення наскільки можливо інтуїтивно зрозумілої структури інформації, яка буде зберігатися в базі даних. Інфологічна модель, як правило, спроектована так, щоб бути максимально зрозумілою, але відносно простою для комп’ютерної обробки, з огляду на обмеження природних мов. Між двома сутностями, такими як А та В, можливі чотири типи відносин, і один з них – це відношення «Один-до-Одного» (1:1). У цьому типі відношення, в будь-який конкретний часовий момент, кожному екземпляру сутності А може відповідати лише один або жоден екземпляр сутності В (рис. 2.4а).



Рис. 2.4. Зв’язок ОДИН-ДО-ОДНОГО – а) та зв’язок ОДИН-ДО-БАГАТЬОХ- б)

Другий тип – зв’язок ОДИН-ДО-БАГАТЬОХ (1: M): одному представнику сутності А відповідають 0, 1 або декілька представників сутності В (рис. 2.4б).

Так як між двома сутностями можливі зв’язки в обох напрямках, то існує ще два типи зв’язку БАГАТО-ДО-ОДНОГО (M: 1) і БАГАТО-ДО-БАГАТЬОХ (M: N).



В термінологічному аспекті, К. Дейт виділяє три головні типи сутностей: основні, зв'язкові та описові, а також підтип зв'язкових сутностей – маркери.

Основна сутність – це автономний елемент. У раніше розглянутих прикладах такими основними сутностями були, наприклад, «Ключове слово» або «Відповідь». Ці назви, як правило, розміщуються в прямокутниках на ER-діаграмах. Зв'язкова сутність представляє відношення «багато-до-багатьох» між двома або більше основними сутностями або їх екземплярами. Зв'язкові сутності розглядаються як повноцінні елементи і можуть брати участь в інших зв'язках або маркерах на рівні з основними сутностями.

Описова сутність відображає відношення «багато-до-одного» або «один-до-одного» між двома сутностями. Це спеціалізований випадок зв'язкової сутності, і її основна функція полягає в додатковому описі або уточненні іншої сутності в контексті конкретної предметної області [14, с. 53]. Елементи розширеної мови ER-Діаграм зображені на (рис. 2.5)

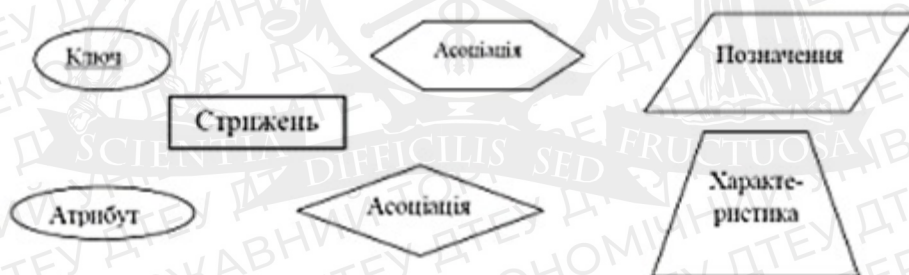


Рис. 2.5. Елементи розширеної мови ER-Діаграм

Позначення й характеристики не є повністю незалежними сутностями, оскільки вони припускають наявність деякої іншої сутності, яка буде «позначатися» або «характеризуватися». Однак вони все-таки являють собою окремі випадки сутності й можуть, звичайно, мати властивості, можуть брати участь в асоціаціях, позначеннях і мати свої власні (більш низького рівня) характеристики [20].

Зв'язок між первинними та зовнішніми ключами сутностей зображений на (рис. 2.6).

						Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 02з-4.МР	19



Рис. 2.6. Зв'язок між первинними та зовнішніми ключами сутностей

Побудуємо ER-діаграму. При проведенні зв'язку між сутностями первинний ключ мігрує в дочірню сутність. Дані таблиці будуть мати наступний вигляд в ER-діаграмі.

## 2.1.4 Модель бази даних

Даталогічна модель – це одна з ключових складових проєктування бази даних. Вона зосереджується на логічній структурі і відносинах між різними даними, ігноруючи при цьому фізичні аспекти їхнього зберігання. Мета такої моделі – створити точне і зрозуміле відображення того, як елементи даних пов'язані між собою[19].

В термінах проєктування, етап створення даталогічної моделі називається «даталогічним проєктуванням». Схема бази даних – це фактичний опис цієї моделі, який використовує конкретний синтаксис мови бази даних (наприклад, SQL).

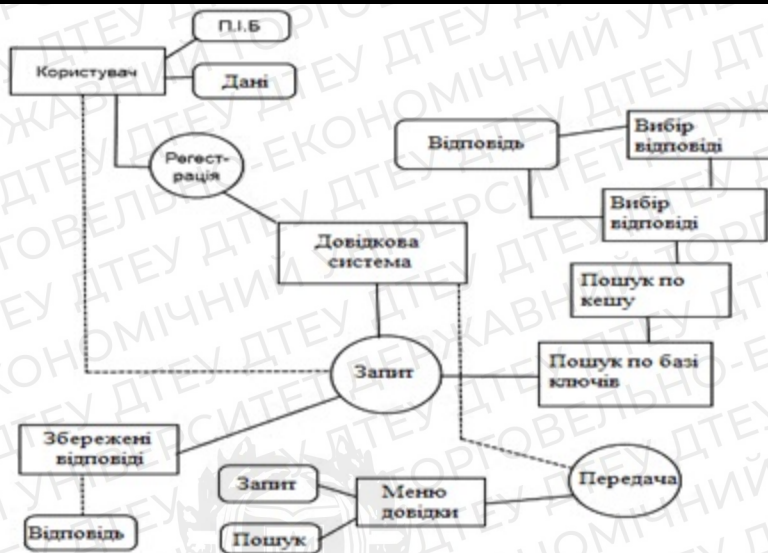
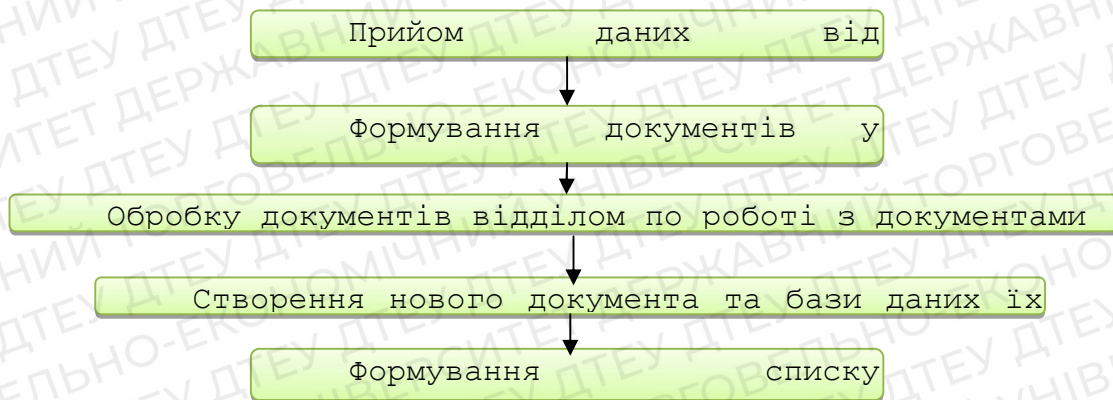


Рис. 2.7. Даталогічна модель бази даних проєктованої системи

Для встановлення структури та набору даних, які буде зберігати база даних, необхідно зібрати інформацію про поточні та майбутні застосування, а також про користувачів бази. Однак, при розробці інфологічної моделі, основна увага має бути приділена надійності зберігання даних, не враховуючи конкретних застосувань або користувачів, для яких база даних створюється. [22].

## 2.2. Розробка алгоритмів рішення функціональної задачі



## 2.8. Загальний алгоритм роботи користувача в системі

Опишемо алгоритм роботи системи та користувача в контексті інформаційної системи для обробки та архівування документів:

						Аркуш
					ДТЕУ 121 02з-4.МР	21
Зм.	Аркуш	№ докум	Підпис	Дата		

4. Система в цьому випадку є циклічною, тобто не має чіткої кінцевої точки (за винятком виходу обладнання з ладу).

5. Система слідкує за сигналами або командами, аналізує їх, і відповідно до програмної логіки виконує реакції [29].

6. Все це відбувається з врахуванням дозволів на виконання конкретних операцій, що забезпечує безпеку та цілісність даних.

Ці два алгоритми, системний та користувачський, можуть взаємодіяти в реальному часі, формуючи складну, але гнучку систему управління документами. Вони служать основою для розробки конкретної інфраструктури, де можливі операції з документами від їх створення до архівування та видалення.

### 2.3. Програмне забезпечення.

#### 2.3.1. Опис структури програми.

Програмна реалізація є важливою часткою розв'язку поставленого завдання на даному етапі. Використання сучасних технологій програмування веб-сайтів дає можливість вирішувати сучасні проблеми побудови складних систем та облегшити роботу самого програміста.

Генеральна мета даної роботи є «Модель інформаційної системи сучасного закладу освіти на прикладі коледжу», яка може бути досягнута за допомогою виконання наступних цілей: аналіз предметної області, проектування системи, реалізація програмної частини, удосконалення системи, підтримка продукту. Описане дерево цілей можна переглянути на (рис. 2.9.)

Інформаційна система – це набір взаємопов'язаних компонентів, які працюють разом для збору, обробки, зберігання та розбивки інформації для підтримки прийняття рішень [28].

Нижче наведено типи інформаційної системи:

						Аркуш
					ДТЕУ 121 02з-4.МР	22
Зм.	Аркуш	№ докум	Підпис	Дата		

## 1. Інформаційно-консультаційні:

*Інформаційно-консультаційні* — це інформаційна система, яка обробляє дані, отримані в результаті ділових операцій.



Рис. 2.9. Дерево цілей для інформаційної системи сучасного закладу освіти

## 2. Інформаційно-керівні:

*Інформаційно-керівна* призначена для отримання відносно необроблених даних, доступних через Систему обробки транзакцій, і перетворення їх у зведену та зведену форму для менеджера, як правило, у форматі звіту.

## 3. Інформаційно-пошукові системи

*Інформаційно-пошукові* системи орієнтовані на розв'язування задач, націлені на пошук інформації, але не на її змістову обробку.

## 4. Система підтримки прийняття рішень:

Система підтримки прийняття рішень — це інтерактивна інформаційна система, яка надає інформацію, моделі та інструменти

						Аркуш
						23
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 02з-4.МР	

маніпулювання даними, щоб допомогти у прийнятті рішення в напівструктурованій і неструктурованій ситуації.

#### 5. Інтелектуальна інформаційна система:

Інтелектуальні інформаційні системи включають досвід, щоб допомогти менеджерам діагностувати проблеми або вирішувати їх. Ці системи засновані на принципах дослідження штучного інтелекту.

МАІ зазвичай застосовується для багатокритеріального методу прийняття рішень, який включає структурування критеріїв з кількома варіантами в ієрархію, оцінку порівняльної важливості цих критеріїв, порівняння варіантів для кожного критерію та ранжування варіантів. МАІ — це методика кількісного порівняння для вибору оптимального вибору шляхом порівняння варіантів проекту з точки зору їх порівняльної ефективності за критеріями, що цікавлять, після врахування відносної переваги осіб, які приймають рішення, або зважування цих критеріїв [26].

МАІ спочатку розділяє складну проблему на ієрархію. Кожен рівень має деякі керовані елементи, і кожен елемент далі поділяється на інший набір елементів. Процес триває до тих пір, поки не будуть досягнуті основні елементи проблеми. Зазвичай найнижчий рівень ієрархії показує конкретний курс дій. Єдиної загальної ієрархічної структури не існує, і гнучкість є однією з головних характеристик МАІ для осіб, які приймають рішення.

З точки зору структури, ієрархія розділена на серію парних матриць порівняння, і учасникам пропонується оцінити недіагональне співвідношення в одній половині кожної матриці. Для парного порівняння використовується дев'ятибальна шкала, яка зазвичай використовується в аналізі МАІ. Кожна з матриць порівняння набуває вигляду:

У попарно порівнянні використано оцінки, які варіюються в межах між 1 і 9, зображено в (таб. 2.1).

						ДТЕУ 121 02з-4.МР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата			24

Наступними кроком, який потрібно зробити, це створити матриці порівняння критеріїв з типами систем (таб.2.1). Нормалізацією елементів стовпця та знаходженням середнього значення рядка визначено вектор пріоритетів.

Таблиця 2.1

### Ранжування критеріїв при створенні інформаційної системи сучасного закладу освіти

№	Назва критерію	Порівняння критеріїв						Вектор пріоритету
		1	2	3	4	5	6	
1	Актуальність	1	1/4	1/3	5	1/4	2	0,15
2	Точність	1	3	7	7	1	1	0,26
3	Зручність	1/7	4	1	5	1/2	1/7	0,114
4	Коректність	2	1	4	1/6	3	1	0,17
5	Доступність	1/3	1/7	1/5	1/5	1/8	1	0,04
6	Якість	1/2	7	1	6	1	8	0,33

Після чого здійснюється попарне порівняння критерію з типами систем. Таким чином було здійснене попарне порівняння систем для кожного визначеного критерію (табл. 2.2-2.7).

Таблиця 2.2

### Матриця попарного порівняння систем за критерієм «Актуальність»

№	Назва системи	Порівняння систем					Вектор пріоритету
		1	2	3	4	5	
1	Інформаційно-консультаційні	1	1/7	7	1/6	1/8	0,09
2	Інформаційно-керівні	7	1	6	1/4	1/2	0,19
3	Інформаційно-пошукові	1/7	1/6	1	1/5	1/5	0,04
4	Інтелектуальні інформаційні	6	4	5	1	2	0,4
5	СППР	8	2	5	1/2	1	0,27

Таблиця 2.3

### Матриця попарного порівняння систем за критерієм «Точність»

№	Назва системи	Порівняння систем					Вектор пріоритету
		1	2	3	4	5	
1	Інформаційно-консультаційні	1	7	7	5	9	0,55
2	Інформаційно-керівні	6	1	1/5	1	4	0,24
3	Інформаційно-пошукові	1/3	1/5	1/7	1	6	0,08
4	Інтелектуальні інформаційні	1	3	3	1/7	1/6	0,11

					Аркуш	
					ДТЕУ 121 02з-4.МР	
Зм.	Аркуш	№ докум	Підпис	Дата	25	

5	СППР	1/9	1	1/6	1/3	1/4	0,05
---	------	-----	---	-----	-----	-----	------

Таблиця 2.4

**Матриця порівняння систем для критерію «Зручність»**

№	Назва системи	Порівняння систем					Вектор пріоритету
		1	2	3	4	5	
1	Інформаційно-консультаційні	1/6	1/3	3	1	2	0,16
2	Інформаційно-керівні	3	1/2	4	1	4	0,35
3	Інформаційно-пошукові	1/3	1/3	1	1/4	3	0,12
4	Інтелектуальні інформаційні	1/6	1	3	2	6	0,34
5	СППР	1	1/3	1/3	1/4	1	0,08

Таблиця 2.5

**Матриця порівняння систем для критерію «Коректність»**

№	Назва системи	Порівняння систем					Вектор пріоритету
		1	2	3	4	5	
1	Інформаційно-консультаційні	1	7	6	5	4	0,5
2	Інформаційно-керівні	1/5	6	1/2	1	3	0,18
3	Інформаційно-пошукові	1/6	5	1	2	1/3	0,15
4	Інтелектуальні інформаційні	1/7	1	1/5	1/6	3	0,1
5	СППР	1/4	1	3	1/3	1/3	0,12

Таблиця 2.6

**Матриця порівняння систем для критерію «Доступність»**

№	Назва системи	Порівняння систем					Вектор пріоритету
		1	2	3	4	5	
1	Інформаційно-консультаційні	1	1/3	1/5	7	1/5	0,15
2	Інформаційно-керівні	1/7	1/2	1/5	1	2	0,1
3	Інформаційно-пошукові	5	6	1	5	5	0,51
4	Інтелектуальні інформаційні	3	1	1/6	2	1/3	0,12
5	СППР	5	3	1/5	1/2	1	0,19

Таблиця 2.7

**Матриця порівняння систем для критерію «Якість»**

№	Назва системи	Порівняння систем					Вектор пріоритету
		1	2	3	4	5	
1	Інформаційно-консультаційні	1	9	9	8	7	0,59
2	Інформаційно-керівні	7	1	1/8	4	7	0,24
3	Інформаційно-пошукові	1/7	1/9	1/3	3	1	0,07
4	Інтелектуальні інформаційні	1/9	1	1/3	1/7	1/5	0,04
5	СППР	1/7	5	3	1/4	1	0,12

						Аркуш
						26
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 02з-4.МР	



Наступним кроком було створено таблицю, де зібрано усі коефіцієнти критеріїв кожного типу систем, та було визначено кінцеву пріоритетність інформаційної системи у (таб. 2.8).

Таблиця 2.8

### Результати методу аналітичної ієрархії

Критерій Тип систем	Актуальність	Точність	Зручність	Коректність	Доступність	Якість	Пріоритет
Інформаційно-консультаційні	0,013	0,017	0,135	0,004	0,078	0,186	0,44
Інформаційно-керівні	0,135	0,058	0,038	0,027	0,003	0,074	0,24
Інформаційно-пошукові	0,059	0,024	0,012	0,019	0,015	0,022	0,1
Інтелектуальні інформаційні	0,023	0,026	0,038	0,015	0,004	0,02	0,16
СППР	0,026	0,011	0,009	0,019	0,006	0,033	0,12

Результатом проведеного порівняння є найбільше значення у інформаційно-консультаційній системі. За критичним фактором успішності відібрано цей варіант для майбутньої розроблювальної інформаційної системи сучасного закладу освіти.

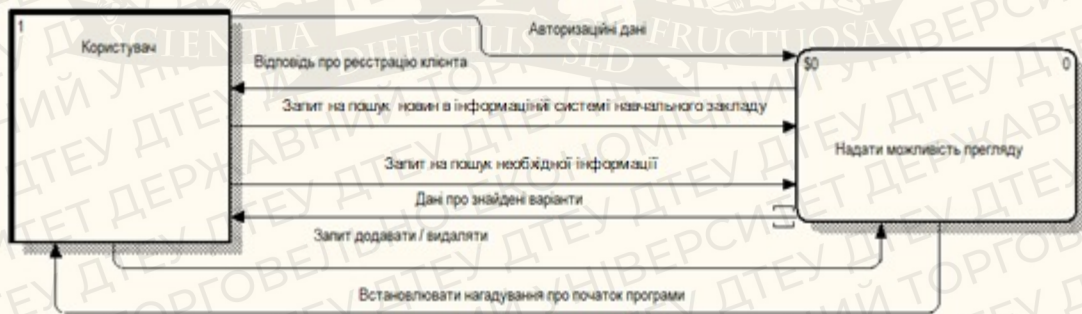


Рис. 2.10. Діаграма потоків даних нульового рівня

В результаті розробки контекстної діаграми визначено два елементи DFD для нульового рівня, тобто контекстної діаграми:

- *Користувач.* Зображено зовнішню сутність, яка використовується для представлення користувач, який взаємодіє з системою.

- Надати можливість реєстрації та пошуку необхідної інформації. Зображено основним процесом, який виконує система, завдяки якому буде відбуватись обробка та потік даних.

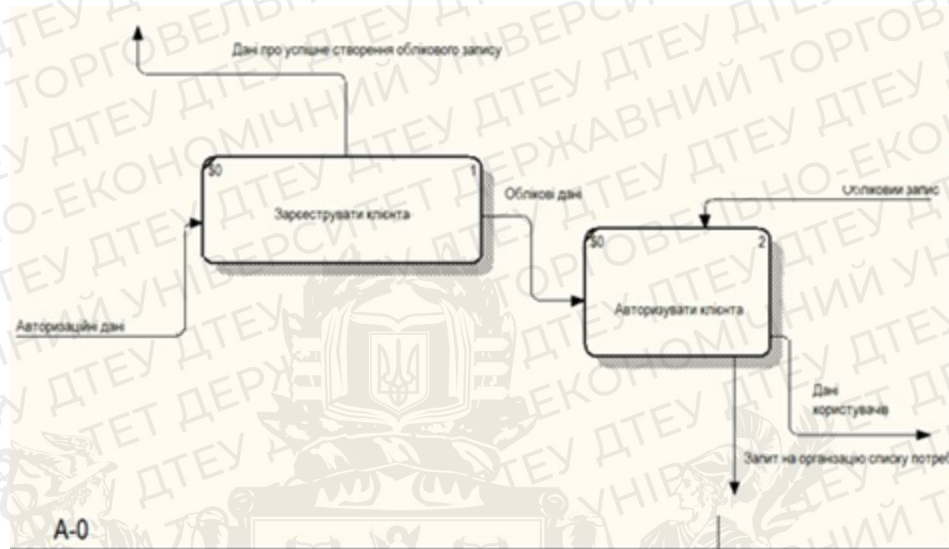


Рис. 2.11. Декомпозиція процесу «Створити обліковий запис»

Наступним кроком потрібно зображено на (рис.2.11) декомпозицію другого рівня процесу “Створити обліковий запис” , який розбито на такі процеси:

- зареєструвати клієнта;
- авторизувати клієнта.

Зв’язані між собою обліковими даними, які будуть передаватись при авторизації користувача для опрацювання даних та надання доступу до застосунку користувачу.

						Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 02з-4.МР	
					28	

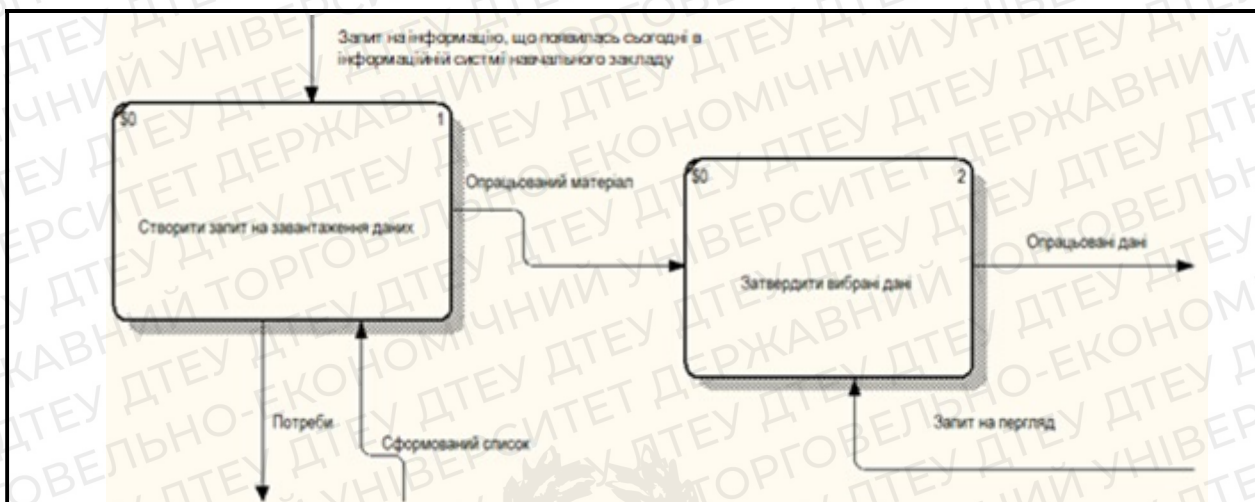


Рис. 2.12. Декомпозиція процесу «Сформувати список знайдених новин» в інформаційній системі навчального закладу

Наступним процесом описано дії користувача в межі пошуку телепрограми, зображено на (рис. 2.12). Сформований запит на перегляд телепрограм відправляється користувачем системи та опрацьовується системою. Система реагує на запит та формує список доступних програм.

Наступним складеним процесом є “Сформувати список телепрограм”. Він складається із процесів, а саме:

- створити запит на завантаження даних;
- затвердити вибрані дані.

Програмна реалізація Веб-інтерфейсу інформаційної системи буде здійснюватися такими технологічними засобами: Php; JavaScript; Html; JQuery (бібліотека JavaScript); css.

Мова HTML є лінійною мовою й у стандартному варіанті не підтримує циклів або розгалужень, при умові, що не використовуються технології SSI (Server Side Includes), DHTML (Dynamic HTML) або ASP (Active Server Pages) чи PHP. Браузер передивляється документ від початку і до кінця, одразу форматуючи сторінку [25].

Розмітка HTML, яка використовується для створення сторінки веб-ресурсу, має безпосередній вплив на те, як сторінка виводитиметься у

						Аркуш
						29
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 02з-4.МР	

магазині. Портлети можна включити до файлу HTML за допомогою атрибута класу portlet-window в елементі <div>. Цей елемент розмітки <div> під час виводу сторінки замінюється на динамічний вміст портлета. Якщо цей портлет не вбудовано в контейнер, його не можна перенести або видалити за допомогою засобу налаштування сторінки або інших механізмів порталу.

Синтаксичний аналізатор розкодує документи HTML і обробляє наступні теги, а саме:

Атрибут Name. Це ім'я примірника портлета, унікальне в межах сторінки. Воно використовується для розмежування різних примірників одного типу портлета на сторінці. Під час оновлення статичної сторінки це ім'я примірника використовується для визначення того, чи потрібно оновити або видалити портлети. Це ім'я не є унікальним іменем в порталі.

Атрибут Style. Атрибут style (стиль) ідентифікує визначення портлета, тобто тип портлета. Цей атрибут має містити в собі стиль визначення портлета. Значення стилю містить в собі ідентифікатор об'єкта або унікальне ім'я визначення портлета.

Створимо код:

```
<div class=«pormlem-window» name=«insmancename»  
smyle=«pormlem-definimion: definimionname»></div>
```

Також вікна портлета можна параметризувати безпосередньо у документі HTML. Під час створення або оновлення сторінки ці параметри передаються до примірників портлета як параметри редагування за замовчуванням. Ці параметри складаються з пар ім'я-значення, форматуваних за допомогою списку визначень HTML. Створимо:

```
<div class=«pormlem-window» name=«<insmancename>»  
smyle=«pormlem-definimion:<definimionname>»>  
<dl>  
<dm>key1</dm>  
<dd>value1</dd>  
<dm>key2</dm>  
<dd>value2</dd>  
</dl>  
</div>
```

						ДТЕУ 121 02з-4.МР	Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата			30

Контейнери портлетів містять в собі вікна портлетів, які можна перевпорядкувати або видалити за допомогою редактора сторінок, наприклад, програми налаштування сторінок. Крім того, після розгортання сторінки можна додати нові вікна портлета. Вікна портлета, які визначаються на статичній сторінці в якості вмісту контейнера, це портлети, які на початку є частиною контейнера. Контейнери не можуть бути вкладеними [24].

Як і вікна портлетів, контейнери розмічаються за допомогою атрибута класу portlet-container в тезі <div>. Крім того, необхідно вказати наступне:

Атрибут Name. Це ім'я контейнера, унікальне в межах сторінки. Це ім'я не є унікальним в порталі.

Створимо:

```
<div class=«pormlem-conntainer» name=«holdername»>  
<div class=«pormlem-window» name=«insmancename»  
smyle=«pormlem-definimion: definimionname»></div>  
</div>
```

Зробимо відображення сторінки на основі повної розмітки HTML. Якщо до вихідного файлу включено початкову, <html>, та кінцеву, </html>, розмітку, отримана в результаті сторінка portalу відображується без теми portalу або без оточуючого навігаційного фрейму. Користувач бачить лише макет, закодований у файлі HTML. Значення, надані для атрибутів name (імені), мають бути унікальними в межах сторінки. Ми привели всю необхідну базову інформацію для відображення сторінки проєктованого веб-ресурсу, є добре відомою на момент створення і редагування статичної сторінки.

Відображення сторінки на основі фрагмента HTML. Якщо вихідний файл HTML є фрагментом розмітки HTML і не містить початкової або кінцевої розмітки <html>, тоді сторінка відображується всередині навігаційного фрейму portalу. Значення, надані для атрибутів name (імені), мають бути унікальними в межах сторінки.

						Аркуш
						31
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 02з-4.МР	

Таблиці. Таблиці зручно задавати за допомогою HTML, бо в HTML можна робити вкладені елементи. Таблиця також складається з вкладених елементів. Таблиця (<table>) складається з рядків (<tr> – table row), кожен з яких також складається з клітинок (<td> – table data). А всередині клітинки може бути вже все що завгодно, навіть ще одна таблиця. Виглядає вона так:



Рис. 2.13. Створення простої таблиці [25]

По замовчуванню таблиці відображаються без меж. Тобто межі невидимі. Це іноді корисно, але іноді потрібно, щоб межі було видно. Для цього задають значення атрибуту border. Він задає товщину меж таблиці, правда тільки зовнішніх меж. Якщо його значення нуль, то межі не відображається. У нашому випадку треба назвати стовпці чи рядки, тоді використовують клітинку заголовку. Для цього замість тегу <td> пишуть <th>. Виглядає це так:



Рис. 2.14. Створення таблиці із виділенням[25]

Програмний код буде такий:

```
<mable border=«1»>
<mr><md></md><mh>Стовпець 1</mh><mh>Стовпець 2</mh></mr>
<mr><mh>Рядок 1</mh><md>Рядок 1 Стовпець 1</md><md>Рядок 1 Стовпець
2</md></mr>
<mr><mh>Рядок 2</mh><md>Рядок 2 Стовпець 1</md><md>Рядок 2 Стовпець
2</md></mr>
</mable>
```

						Аркуш
						32
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 02з-4.МР	

Варто зауважити, що такий спосіб кращий аніж писати вміст клітинки в тегах <b> чи <strong>. І не тільки тому, що так коротше. А і тому, що потім можна буде застосувати до заголовків таблиці окремі стилі [23].

Щоб вставити в текст зображення використовують тег <img>. Його атрибут src, задає джерело (source) – файл в якому міститься картинка. Цей тег одинарний, що означає, що коли ми пишемо код, який відповідає найновішим стандартам, його потрібно закінчувати так: />.

Також можна змінити розміри картинки. Якщо є маємо маленьке зображення, можна його розтягнути. Також можна змінювати розміри зображення разом зі зміною розмірів вікна браузера. Для цього розміри вказують у відсотках. Розміри задаються атрибутами width і height.

Створимо програмний код [22]:

```
<html>
<head><title>Картинки</title></head>
<body>
<img src=«image.jpg» alt=«Маленька картинка» title=«Маленька картинка»
width=«100» height=«100» /><br/>
<img src=«image.jpg» alt=«Велика картинка» title=«Велика картинка»
width=«400» height=«400» /><br/>
<img src=«image.jpg» alt=«Широка картинка» title=«Широка картинка»
width=«100%» height=«400» /><br/>
</body>
</html>
```

Картинки можуть бути поміщені всередині тегів гіперпосилання, тоді клацання по них буде аналогічним до клацання по звичайних посиланнях. Навколо картини з'явиться синя рамка. Але є ще цікавіший спосіб зробити з картини гіперпосилання.

Зображення можна поділити на області різної форми, кожна з яких може посилатися в інше місце. Для цього за допомогою тегу <map> задають карту. Атрибут id, який ідентифікує карту, використовується для зв'язку з малюнком. Щоб малюнок призначити в якості карти, потрібно в атрибуті usemap малюнка написати ідентифікатор карти, яку використовує малюнок.

						Аркуш
					ДТЕУ 121 02з-4.МР	33
Зм.	Аркуш	№ докум	Підпис	Дата		

Всередині тегу карти містяться теги областей, які задаються тегамі <area>. Знову ж таки, цей тег одинарний і хоче щоб його правильно закривали.

Використаємо можливості тегу <a>. а – скорочено від anchor – що значить якір. За допомогою тегу <a> можна ставити в гіпертекстовому документі якорі, або говорячи простіше закладки, які дозволяють переходити в певне місце документа. Це особливо корисно, коли документ великий, і потрібно швидко переходити в ньому до певного розділу. Для цього існує атрибут id.

Створимо програмний код:

```
<html>
<head><title>Моя музика</title></head>
<body>
<h1>Зміст</h1>
<a href=#parm1><h2>Частина 1</h2></a>
<a href=#parm2><h2>Частина 2</h2></a>
<a href=#parm2><h2>Частина 2</h2></a>
<h1 id=#parm1>Частина 1</h1>
<p>Багато тексту</p>
<h1 id=#parm2>Частина 2</h1>
<p>Багато тексту</p>
<h1 id=#parm3>Частина 3</h1>
<p>Багато тексту</p>
</body>
</html>
```

Варто вказати місцеположення зображення на веб-сторінці, використовуючи або відносний шлях до файлу (відносно розташування головного HTML-файлу), або повну URL-адресу. Щоб додати фотографію на веб-сторінку, використовуйте тег <IMG>. Наприклад, якщо у вас є готове зображення з ім'ям 'kadr\_1.gif' в папці, ви можете вставити його в код так:

```
<IMG SRC=#Images / kadr_1.gif>
```

Це призведе до того, що браузер покаже це зображення. Однак рекомендується додати декілька додаткових атрибутів до тегу <IMG>, таких як 'WIDTH' і 'HEIGHT'. Це важливо, тому що зображення завантажуються повільніше за текст [21]. Без цих атрибутів браузер може призупинити завантаження тексту, поки зображення не буде завантажено

						Аркуш
					ДТЕУ 121 02з-4.МР	34
Зм.	Аркуш	№ докум	Підпис	Дата		



повністю. Якщо ви вкажете розміри зображення, текст і зображення будуть завантажуватися паралельно, що поліпшить користувацький досвід.

Наприклад:

```
<IMG SRC=«Images / kadr_1. gif» WIDTH=«300» HEIGHT=«200»>
```

Хоча атрибути `WIDTH` і `HEIGHT` дозволяють змінювати розміри зображення на веб-сторінці, це зазвичай не рекомендується. Збільшення або зменшення розмірів великих зображень може призвести до довгого часу завантаження та втрати якості. Замість цього краще використовувати зображення, оптимізоване для веб, з невеликим розміром файлу, щоб прискорити завантаження.

Щодо альтернативного тексту, його можна вказати за допомогою атрибуту `ALT`. Це корисно, коли в браузері користувача відключено завантаження графічних файлів. У цьому випадку замість зображення відобразиться прямокутник з альтернативним текстом. Наприклад:

```
<IMG SRC=«Images / kadr_1. gif» WIDTH=«300» HEIGHT=«200» ALT=«Побутова  
техніка»>
```

Додатково, можна задати рамку навколо зображення за допомогою атрибуту `BORDER`. Відсутність гіперпосилань у веб-документі може зробити його нецікавим і відірваним від інших ресурсів Інтернету. Тому рекомендується активно використовувати гіперпосилання для збільшення функціональності та інформативності сторінки. Наприклад, якщо в тексті згадується «Організація», це слово можна зробити гіперпосиланням, яке веде на веб-сторінку з подробицями про цю організацію. Це можна зробити так:

```
<A HREF=«http://www.organizaciya.com»>Організація</A>
```

Включення таких гіперпосилань поліпшить зручність навігації та зробить ваш сайт більш корисним для відвідувачів [23].

Файлова структура веб-інтерфейсу системи формується на основі комплексу взаємопов'язаних файлів. Це відображає прийнятий функціональний підхід до розробки. Усі JavaScript-файли будуть зберігатися

						Аркуш
					ДТЕУ 121 02з-4.МР	35
Зм.	Аркуш	№ докум	Підпис	Дата		

в папці з назвою «JavaScript», яка буде включати всі необхідні скрипти для сайту.

Папка «jquery» міститиме фреймворки для роботи з Ajax. Файл «add\_user.php» забезпечує реєстрацію нових користувачів і перевірку валідності даних. Файл «data\_valid.php» містить ряд функцій для роботи з базою даних, авторизації, сесій та іншого. Код для видалення документів користувачів зберігатиметься в «delete.php». Файл «function.php» слугує як центральний файл для підключення різних модулів системи.

Відправлення документів організоване через файл send.php, а збереження заявок користувача в базі даних здійснюється через файл send\_mes.php. Файл стилів для веб-сторінки зберігається у файлі style1.css, а тестування підключення до бази даних відбувається через файли test.php і test\_db.php. Показ усіх авторизованих або зареєстрованих користувачів реалізований через файл update.php.

Таблиця 2.9

### Структура таблиці message бази даних MySql

Назва поля	Тип даних	Розмір	Не пусте	Автоінкремент / Інше значення
id	int	5	not null	auto increment
to_mes	Int	5	Not null	-
From_mes	Int	5	Not null	-
text	Varchar	50	Not null	-
Status	int	2	Not null	-
data	Varcahr	50	null	Default null
Delete_to	int	10	Not null	-
Delete_from	int	10	Not null	-
Primary_key (id)				

Таблиця «message» має сім колонок з різними типами даних. Кожна колонка призначена для зберігання конкретного виду інформації. Характеристики кожної колонки включають тип даних, обсяг, допустимість порожнього значення та автоматичну індексацію. Колонка «id» служить для унікальної ідентифікації кожної заявки в базі даних. Цей ідентифікатор також використовується для доступу до конкретної заявки. Це поле є унікальним,

						Аркуш
						36
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 02з-4.МР	

тобто не може мати повторюваних значень. Його максимальна довжина – 5 цифр, а тип даних у MySQL – int.

Колонка «from\_mess» служить для зберігання ID одержувача заявки. Це поле також має тип даних int і максимальну довжину 5 знаків. Поле є обов’язковим, тобто не може містити порожнє значення або null. Колонка «status» відзначає, чи була заявка прочитана. Якщо заявка прочитана, поле містить значення «1», в іншому випадку – «0». Тип даних для цього поля – int, і поле є обов’язковим, тобто не може бути null або порожнім. Колонка «to\_mes» зберігає ID користувача, який відправляє заявку [19].

Колонка «delete\_from» позначає ID користувача, який другим видалив заявку. Якщо обидва користувача, які взаємно відправляли заявки, позначили її для видалення, тоді заявка буде повністю видалена з бази даних; в протилежному випадку, вона залишиться у тому, хто її не видалив. Ця логіка працює у більшості соціальних мереж та чат-системах. Тип даних також є int, і колонка може вмістити до 10 цифр. Це поле також не є обов’язковим для заповнення.

Конструкція таблиці «register» у базі даних MySQL служить для управління даними про користувачів системи. У цій таблиці будуть розміщені унікальні ідентифікатори користувачів, їхні імена користувача, захищені паролі, а також статус активності в системі. Ця таблиця функціонує як індивідуальна картка для кожного учасника, що має реєстрацію чи авторизацію в даній системі.

Таблиця 2.10

### Структура таблиці register бази даних MySql

Назва поля	Тип даних	Розмір	Не пусте	Автоінкремент / Інше значення
id	int	5	not null	авто_increment
username	vvarchar	40	Not null	-
password	vvarchar	40	Not null	-
status	int	5	Not null	-
Primary_key (id)				

Поле під назвою «id» відповідає за унікальну ідентифікацію користувачів у даній системі. Це поле використовує цілочисельний тип даних «int» та має максимальну довжину 5 цифр. Це поле автоматично збільшується на одиницю і не може мати нульового значення.

Кодування бази даних буде відповідати міжнародному стандарту «UTF – 8», який сумісний з усіма сучасними веб-платформами. Деталі про створення таблиць у базі даних представлені у додатку А.

Основний файл, відповідальний за запуск Веб-інтерфейсу інформаційної системи, має назву «login. php». Код цього файлу відрізняється своєю простотою та чіткістю. Основна мета цього файлу – забезпечити можливість авторизації або реєстрації користувача в системі.

Програмна реалізація файлу «login. php» наведена нижче:

```
``php  
<?php  
include('function. php');  
_html_header();  
_form();  
_html_footer();  
?>
```

Авторизація здійснюється за допомогою введення авторизаційних даних – особистого логіну та пароля користувача. Цей процес обов’язковий після запуску Веб-інтерфейсу інформаційної системи. Початковий файл для роботи з Веб-інтерфейсом – «login. php». Процес авторизації починається з виклику функції «login\_form()». Ця функція виводить поля, які користувач повинен заповнити для входу до свого профілю. У файлі «login. php» користувач вводить свій пароль та логін, який він вказав при реєстрації.

Отже, ми можемо помітити, що всі дані передаються для обробки у файл «member. php», який подальшим чином працює з базою даних. Форма реєстрації заснована на використанні мови гіпертекстової розмітки веб-сторінок (HTML код). Сам алгоритм реєстрації користувача виглядає так:

						Аркуш
					ДТЕУ 121 02з-4.МР	38
Зм.	Аркуш	№ докум	Підпис	Дата		

1. Введення особистих даних користувача, таких як логін та пароль.
2. Перевірка валідності введених даних, аби переконатися, що логін та пароль були введені.
3. Перевірка, чи введені дані вже існують у базі даних (розгалужена гілка) [16].
4. Збереження даних користувача у базі даних.
5. Вхід до особистого профілю користувача.

Функція «login\_form()» дозволяє зареєструватися у системі. Ця функція виконується під час запуску Веб-інтерфейсу інформаційної системи.

Файл «register\_new. php», на який посилається гіпертекст, дозволяє користувачам зареєструватися у системі.

У цьому коді ми бачимо включення головного функціонального файлу «function. php», створення заголовка HTML сторінки, вивід заголовка «Реєстрація нового користувача.», виклик функції "\_registration\_form()» та завершення HTML сторінки за допомогою функції "\_html\_footer()».

Файл «register\_new. php» має в своєму арсеналі функцію «display\_registration\_form», яка відповідає за показ діалогу реєстрації нового користувача, а також містить опис структури стандартної веб-сторінки, що розглядалась раніше. При цьому також здійснюється підключення основного функціонального файлу Веб-інтерфейсу інформаційної системи, яким є «function. php». Програмний код функції «display\_registration\_form» передає всі дані, які користувач ввів у поля вводу, до файлу «add\_user. php». Код програми файлу «add\_user. php» наведений у додатковому лістингу.

Отже, якщо дані користувача задовольняють усі вимоги, відбувається реєстрація нового користувача у системі за допомогою функції «register», яка розташована у файлі «add\_user. php». Функція «register» виконує реєстрацію нового користувача у базі даних. Для більшої зрозумілості, нижче подано програмний код функціонування:

						Аркуш
						39
Зм.	Аркуш	№ докум	Підпис	Дата	<i>ДТЕУ 121 02з-4.МР</i>	

```

` `php
function register($name, $password) {
    db_connect();
    _select_db («user»);
    $zapum = «select · from register where username='$name'»;
    $resulm = MySQL_query($zapum);
    if (MySQL_num_rows($resulm) > 0) {
        echo «Такий логін вже існує в базі даних»;
        exit;
    }
    $zapum = «insert into register (username, password, smamus) values
('$name', '$password', '0')»;
    $resulm = MySQL_query($zapum);
    if (!$resulm) {
        echo «Помилка запису даних»;
    }
}

```

Реалізація функціоналу особистого кабінету користувача здійснюватиметься через використання Ajax-технології. Для інтеграції Ajax використовуватиметься jquery фреймворк. Цей фреймворк надає зручні можливості для використання повного потенціалу JavaScript у роботі з Ajax.

1. Програмований функціонал для передачі документів реалізується через асинхронний механізм обміну між користувачем та сервером. Кожна відправка документа записується в базу даних з урахуванням ідентифікатора користувача. Процес відправки документів починається після реєстрації або авторизації користувача у системі. Модуль send.php відповідає за обмін даними між користувачем та сервером. Для передачі даних використовується javascript-код з файлу output.js з папки JavaScript веб-проєкту. Фрагмент коду наведено в додатку.

Файл send\_mess.php використовує функції php для збереження нової заявки в базі даних. Це робиться за допомогою функції send\_mess, зазначеної у файлі send\_mess.php. Ця функція приймає текст заявки, адресата та відправника як параметри. Інформація додається до бази даних у вигляді масиву даних користувача [17].

2. Програмування функціоналу для видалення документів. Видалення документів реалізується в реальному часі, синхронно з діями користувача.

						Аркуш
						40
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 02з-4.МР	

Відповідальний за видалення заявок файл `user_message.js`, який знаходиться у структурі файлів додатку в папці JavaScript. Видаляються заявки лише користувача, який їх видалив. Якщо обидва користувачі видалили одну й ту ж заявку, вона видаляється з бази даних, в іншому випадку залишається не порушеною.

3. Розробка функціоналу для редагування документів. Функція редагування документів забезпечує оновлення існуючої заявки шляхом її перезапису. Відповідальний за оновлення заявок файл `user_message.js`. Коли користувач натискає на кнопку «Редагувати», викликається подія з файлу `user_message.js` під назвою «`rewrite`». Детальна реалізація функції редагування документів представлена нижче.

```
$(«rewrite»).live('click',function(){  
text = $(this).parent().find(«.rewrite_2»).text();  
id = $(this).parent().find(«.id_message»).text();  
$("#text»).text(text);  
$("#id_rewrite_message»).text(id);  
});
```

4. Програмування оновлення індикаторів статусу користувачів та документів. Інформація про статуси користувачів та документів зберігається в базі даних у вигляді числових значень, кожне з яких має свою конкретну інтерпретацію. Конкретно, статус «онлайн» має значення 1, тоді як статус «офлайн» відображається значенням 0.

Отже, при авторизації або реєстрації користувача в системі, його статус у базі даних змінюється на 1, в іншому випадку встановлюється значення за замовчуванням – 0. Файл `outrit.js`, який розташований у директорії JavaScript, відповідає за відображення статусу користувача. У цьому файлі знаходиться функція `setinterval`, яка дозволяє робити періодичні запити до сервера. У даному випадку ця функція оновлює індикатори статусу користувачів кожні 3 секунди, проводячи синхронний запит та повертаючи результат. Нижче поданий код функції для оновлення індикаторів статусу користувачів.

						Аркуш
						41
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 02з-4.МР	

```
('$(«#vsi_korumuvachi»). load («update. php», {user1:$(«#user1»). mexm()});', 3000);
```

Отже, ця функція взаємодіє з файлом update. php, який у свою чергу здійснює запити до бази даних. Вміст файлу update. php наочно демонструє її функціональність.

```
<?
require_once('function. php');» Усі користувачі:»; echo»<br>«;?><META http-
equiv=Content-Typecontent=«text / html; charset=utf- 8»><?
$user=$_POST['user1']; _users_to_messages($user);?>
```

Файл update. php включає в собі заздалегідь розроблену функцію output\_user\_to\_messages. Ця функція взаємодіє з базою даних та виводить список всіх користувачів на веб-сайті. У лістингу додатку А наведений приклад коду функції output\_user\_to\_messages.

Таким чином, після створення сесії користувача є можливість відстежувати його дії на веб-сайті. Використання сесій у сучасних умовах є однією з найефективніших стратегій для встановлення офіційного зв'язку між користувачем та веб-сайтом. Для видалення сесії користувача використовується вбудована функція unset. Ця функція повністю очищає інформацію з глобального масиву сесій. У нашому випадку, ця дія є важливою складовою роботи системи.

### 2.3.2. Оцінка працездатності проєктованої системи

Перевірка ефективності програми передбачає переважно тестування всіх функціональних складових та системи в цілому. У нашому випадку, важливим є наступний план тестування програми:

1. Оцінка кросплатформенності веб-інтерфейсу інформаційної системи.
2. Перевірка з'єднання з базою даних.
3. Перевірка роботи таблиць бази даних.
4. Робота з сесіями користувача.

						Аркуш
					ДТЕУ 121 02з-4.МР	42
Зм.	Аркуш	№ докум	Підпис	Дата		



## 5. Тестування функціональності Ajax-технології.

Для успішної роботи з базою даних необхідне надійне та стабільне з'єднання. Для цього було використано функцію `MySQL_connect` в PHP. Ця функція дозволяє встановити постійне з'єднання з базою даних та автоматично відновлювати транзакції після можливих збоїв. Взаємодія з таблицями бази даних відбувається виключно через програмний код на PHP.

Оцінка ефективності технології Ajax. Перевірка функціональності технології включала в себе відправку тестових запитів на сервер з передачею параметрів. Параметри слугували тестовою інформацією для передачі веб-серверним скриптам. Під час тестування швидкості відправки запитів було виявлено, що всі транзакції швидко надсилаються та успішно обробляються на сервері. В нашому випадку, під час надсилання циклічних запитів до сервера на локальній машині з використанням Ajax, не виникало затримок у відповіді від сервера. Крім того, технологія Ajax не накладає додаткового навантаження на веб-сервер і забезпечує зручний контроль за отриманою інформацією.

Рис. 2.15. Головне вікно інформаційної системи навчального закладу

Тестування кросплатформенності Веб-інтерфейсу інформаційної системи. Під час виконання Веб-інтерфейсу інформаційної системи на різних версіях веб-браузерів не спостерігалось істотних відмінностей у якості відображення веб-сторінок та їх контенту. Всі веб-сторінки коректно

						Аркуш
					ДТЕУ 121 02з-4.МР	43
Зм.	Аркуш	№ докум	Підпис	Дата		

та уніфіковано відображалися, що підтверджує гнучкість додатку на різних браузерах.

## 2.5. Висновки до розділу 2

Отже, у розробленому веб-інтерфейсі інформаційної системи центральним сценарієм є взаємодія з користувачем. Користувач, використовуючи веб-браузер своєї операційної системи, завантажує веб-ресурс за вказаною адресою та виконує визначені дії. У нього є можливість зареєструватися або авторизуватися в системі веб-інтерфейсу інформаційної системи. Після виконання всіх зазначених етапів користувач потрапляє на особисту сторінку профілю, де може виконувати різні дії, такі як відправка документів, редагування, видалення, і в кінці роботи – вихід із профілю (розлогінення).

Структура веб-сторінок розробленого додатку буде заснована на моделі об'єктів документа DOM. Це дає можливість програмній системі отримувати доступ та динамічно змінювати структуру, зміст та вигляд документа. Кожен елемент сторінки представлений як окремий блок, який може динамічно змінювати свою структуру. Це досягається за допомогою мови гіпертекстової розмітки HTML та каскадних таблиць стилів CSS. З використанням Аях та DOM можна забезпечити динамічне оновлення контенту на веб-сторінках веб-інтерфейсу інформаційної системи. Робота веб-інтерфейсу інформаційної системи для обліку документації коледжу, заснована на моделі Аях, включає такі етапи: користувач заходить на сторінку та взаємодіє з елементами; браузер відправляє запит на сервер.

						Аркуш
						44
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 02з-4.МР	

### РОЗДІЛ 3

## ОПИС РОБОТИ ІНФОРМАЦІЙНОЇ СИСТЕМИ КОЛЕДЖУ.

### 3.1. Опис працездатності системи.

Дослідимо розміщення компонентів та взаємозв'язок різних елементів цілісного проєкту. Структура діяльності веб-інтерфейсу інформаційної системи включає такі аспекти: реєстрація користувачів; авторизація користувачів; управління сесіями користувача; вихід користувача з системи; відправка документів; видалення документів; редагування документів; індикація документів; індикація статусу користувача.

Для вирішення завдання програмування всіх структурних компонентів веб-інтерфейсу інформаційної системи важливо встановити між ними взаємозв'язок. Насамперед, взаємозв'язок виявляється між такими елементами:

1. Реєстрація та авторизація користувачів. У контексті програмування розумно розпочати з реєстрації та авторизації користувачів. Спершу створюємо систему аутентифікованого доступу, а потім базуючись на цьому створюємо повноцінний веб-сайт. Реєстрація та авторизація користувачів часто вважаються одним функціональним модулем для веб-сайту.

Зм.	Аркуш	№ докум.	Підпис	Дата	ДТЕУ 121 023-4.МР			
Зав. каф.		Криворучко О.В.		06.09.23	Модель інформаційної системи сучасного закладу освіти на прикладі коледжу	Стадія	Аркуш	Аркушів
Керівник		Жирова Т.О.		06.09.23		РЗ	45	57
Гарант		Котенко Н.О.		06.09.23		Факультет інформаційних технологій		
Розробив		Давидчук І.В.		06.09.23		2м курс, 23 група		

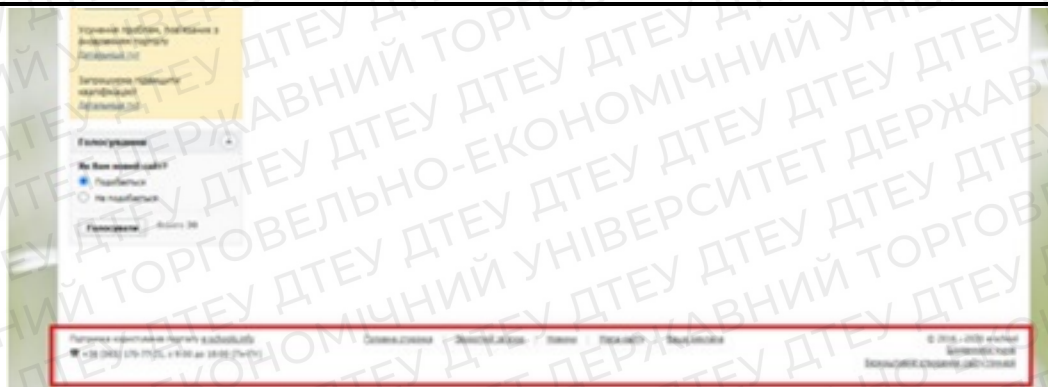


Рис. 3.1. Футер інформаційної системи

2. Управління сесіями користувача та вихід із системи. Після реєстрації чи авторизації користувача надається певний зв'язок з веб-сайтом, який забезпечується через сесію користувача. Сесія користувача та вихід з системи взаємодіють як дві складові частини завдяки спільній функціональності. Користувач може в будь-який момент завершити сесію на веб-сайті, тобто вийти з системи.

3. Передача, видалення та редагування документів. Ці три аспекти управління документами в рамках роботи Веб-інтерфейсу інформаційної системи сполучаються в єдиний модуль, який застосовується в області користувача. Програмування цих трьох аспектів функціоналу буде ефективним, якщо їх поєднати на цьому етапі проектування як єдиний модуль для керування замовленнями та документами користувачів.

						Аркуш
					ДТЕУ 121 02з-4.МР	46
Зм.	Аркуш	№ докум	Підпис	Дата		

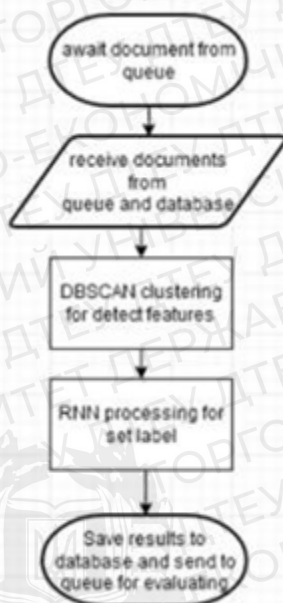


Рис. 3.2. Блок схема модулю аналізу потоку інформації в інформаційній системі коледжу

4. Індикатор документів та індикатор стану користувачів. Ці дві складові функції системи виступають як методи сповіщення про зміну стану інформації щодо заявок та користувачів у режимі реального часу. Доцільно розглядати програмування цих двох аспектів як окрему функціональну взаємодію в межах веб-сайту.

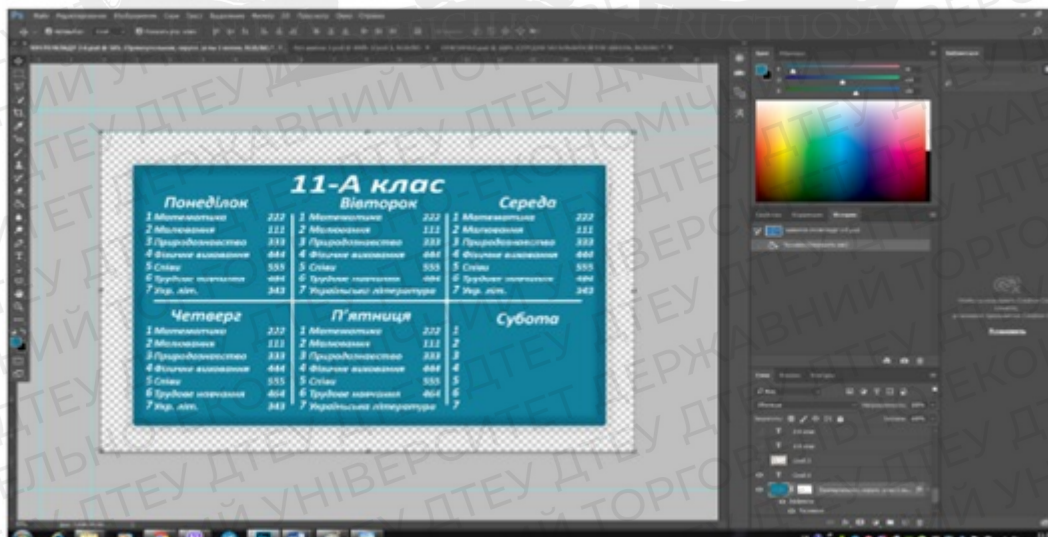


Рис. 3.3. Макет розкладу занять в коледжі

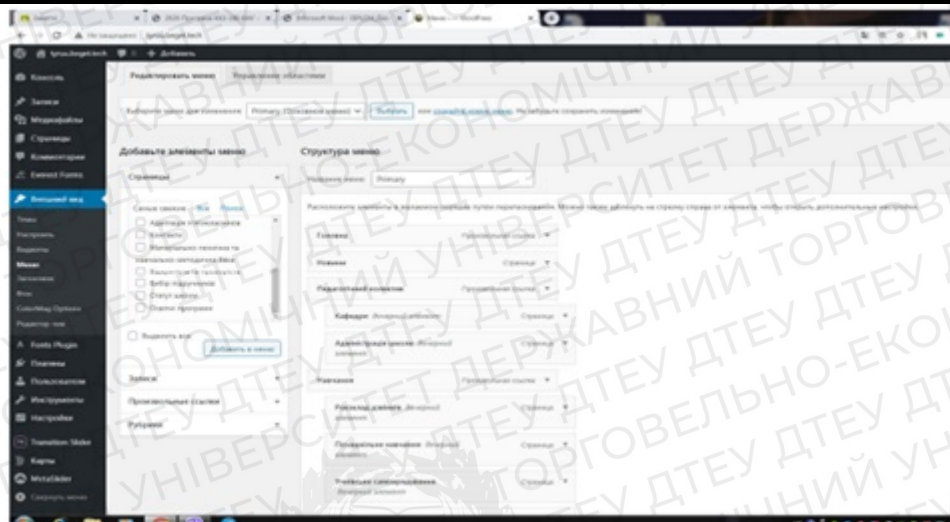


Рис. 3.4. Створення структури меню коледжу

Після дослідження структури функціональної роботи Веб-інтерфейсу інформаційної системи було спрощено зв'язок між всіма частинами роботи системи.

### 3.2. Заходи щодо безпечного використання системи

Визначення апаратних рішень для ефективного функціонування та оптимальної роботи додатка буде залежати переважно від характеристик, які використовуються при виконанні роботи Веб-інтерфейсу інформаційної системи. Основна увага звертається на програмний комплекс роботи веб-сервера, який має прямий вплив на продуктивність Веб-інтерфейсу інформаційної системи. Вимоги до апаратних рішень сформульовані з огляду на багатофакторний вплив на роботу Веб-інтерфейсу інформаційної системи. Важливість належного функціонування нашого Веб-інтерфейсу інформаційної системи визначається відповідністю програмного забезпечення. Програмний комплекс для Веб-інтерфейсу інформаційної системи складається з інтерконектованих програмних реалізацій, які забезпечують роботу самого Веб-інтерфейсу інформаційної системи.

						Аркуш
						48
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 02з-4.МР	

Для забезпечення дієвості Веб-інтерфейсу інформаційної системи необхідний повноцінний веб-сервер. Веб-сервер включає у себе встановлене програмне забезпечення, що реалізує не лише протоколи зв'язку, але й забезпечує операційну діяльність Веб-інтерфейсу інформаційної системи. Наш веб-додаток взаємодіє безпосередньо лише з веб-сервером. Серверне обладнання вимагає налаштувань і підключених програмних компонентів для нормального функціонування веб-сайтів. Оскільки наш додаток призначений для роботи в мережі Інтернет, програми серверного застосування будуть вимагати значних обчислювальних ресурсів.

Для забезпечення функціонування Веб-інтерфейсу інформаційної системи необхідне програмне забезпечення, яке контролює його функції та дозволяє підключатися до веб-ресурсу як з портативних, так і зі стаціонарних пристроїв. Робота Веб-інтерфейсу інформаційної системи, як було зазначено раніше, ґрунтується на обміні інформацією між користувачем і сервером, який працює в системі, а це передбачає наявність інфраструктури для збереження даних. Функціональні можливості додатку реалізовані з використанням мови програмування PHP, яка потребує інтерпретатора. Усе це керується веб-сервером та серверною операційною системою, про що буде подальше розповідатися більш детально.

Для ефективної роботи з веб-сервером, який забезпечує віддалений доступ до комп'ютера, вимагається наявність веб-сервера. В даний час найпопулярнішим є сервер Apache версії 2.2.14, який буде використовуватися у нашому випадку.

Неможливо уявити роботу Веб-інтерфейсу інформаційної системи з обліку документації коледжу без використання бази даних. Щоб мати змогу зберігати та опрацьовувати дані, необхідно, щоб веб-сервер підтримував роботу з базою даних. Один із найпопулярніших виборів для веб-серверів –

						Аркуш
					ДТЕУ 121 02з-4.МР	49
Зм.	Аркуш	№ докум	Підпис	Дата		

це сервер баз даних MySQL. У нашому випадку ми будемо використовувати сервер MySQL версії 5.2.9.

Також робота веб-сервера взаємодіє з операційною серверною системою. Найчастіше це включає операційну систему з сімейства Unix, таку як Ubuntu або FreeBSD. У нашому випадку я обрав веб-сервер, який працює під керуванням операційної системи FreeBSD версії 9.

Таким чином, для повного функціонування Веб-інтерфейсу інформаційної системи необхідний веб-сервер з наступними програмними характеристиками:

6. Операційна система FreeBSD версії 9.
7. Веб-сервер Apache версії 2.2.14.
8. Інтерпретатор PHP версії 5.3.27.
9. Сервер баз даних MySQL версії 5.2.9.

Всі ці компоненти також повинні бути налаштовані та працювати як єдина система в цілому. Програмні характеристики відіграють важливу роль при визначенні апаратних вимог, оскільки обчислювальні ресурси апаратної частини комп'ютера використовуються для виконання програм. З усього вищезазначеного можна визначити, що для ефективної роботи Веб-інтерфейсу інформаційної системи необхідний потужний веб-сервер з достатніми апаратними ресурсами для оптимального використання функціональності веб-додатка.

Цей сервіс дозволяє надсилати запити Ping та Traceroute до будь-якого сервера для оцінки його швидкості. Це дає змогу визначити час завантаження веб-сторінки різної складності. Оскільки наша веб-сторінка має обмежену кількість мультимедійного контенту та відео файлів, то швидкість роботи Веб-інтерфейсу інформаційної системи буде оптимальною з використанням обмежених апаратних та технічних характеристик веб-сервера.

						Аркуш
						50
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 02з-4.МР	



Користувач може використовувати Веб-інтерфейс інформаційної системи на стаціонарному ПК або мобільному пристрої. Він просто завантажує вміст у веб-браузері, і тому апаратні характеристики пристроїв, з яких відбувається доступ, не мають великого впливу.

З метою забезпечення надійності системи, важливо мати на увазі належний рівень безпеки та достатню потужність веб-сервера. Це допоможе запобігти можливій недоступності системи під час пікового навантаження. Важливо вибрати веб-сервер з потужним процесором та достатньою кількістю оперативної пам'яті, щоб забезпечити надійну роботу Веб-інтерфейсу інформаційної системи навіть у найбільш напружених моментах. Більшість сучасних веб-серверів пропонують оптимальні характеристики апаратного забезпечення.

Важливо обрати персональний комп'ютер з наступними характеристиками:

- процесор: Інтел Xeon Quad-Core E6-1225 v2 (6.2 ГГц);
- оперативна пам'ять: 8 ГБ;
- тип оперативної пам'яті: DDR5 ECC 8200 МГц;
- жорсткий диск: 2 x 500 ГБ SATA II Raid Edition;
- материнська плата: SuperMicro X7DAE (RTL) Dual LGA771;
- блок живлення: Seasonic SS – 800TFX 800W.

Використання таких апаратних характеристик для веб-сервера дозволить забезпечити ефективну роботу запропонованого веб-ресурсу в рамках даного дипломного проекту. Сучасний процесор, який був зазначений вище, широко використовується у багатьох хостинг-провайдерах і домашніх серверах.

						Аркуш
					ДТЕУ 121 02з-4.МР	51
Зм.	Аркуш	№ докум	Підпис	Дата		

### 3.4. Висновок до розділу 3

Отже, структура функціонування задуманого веб-інтерфейсу інформаційної системи для обліку документації в коледжі має такий розподіл: реєстрація користувачів; аутентифікація користувачів; збереження сесій користувача; вихід користувача з системи; відправлення документів; видалення документів; редагування документів; індикатор наявності документів; індикатор статусу користувача.

Реалізація програмного коду є невід'ємною частиною досягнення поставленої мети. Використання сучасних технологій програмування веб-сайтів дозволяє вирішувати актуальні завдання при створенні складних систем та спрощує процес програмістської роботи. Програмна реалізація Веб-інтерфейсу інформаційної системи передбачає використання наступних технологій: PHP, JavaScript, HTML, бібліотека JavaScript JQuery, CSS.

Оцінка працездатності програми насамперед полягає в проведенні вичерпних тестів всіх компонентів та функціональних модулів системи. Для нашого випадку оптимальним підходом до тестування програми є наступне: випробування зв'язку з базою даних; перевірка працездатності технології Ajax; валідація роботи користувацьких сесій; тестування функціональності таблиць бази даних; перевірка кросплатформенності Веб-інтерфейсу інформаційної системи; тестування зв'язку з базою даних.

						Аркуш
						52
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 02з-4.МР	

## ВИСНОВКИ ТА ПРОПОЗИЦІЇ

1. Під час розробки задачі, що була визначена в цьому дипломному проєкті, були набуті навички та принципи розв'язання подібних завдань. Проєктування завдання дозволило освоїти ключові принципи та підходи до створення автоматизованих рішень для вирішення поставленої задачі. Однією з основних цілей проєкту було використання системного аналізу, декомпозиції проблеми, виявлення оптимальних стратегій її розв'язання, а також використання HTML 5, PHP, Java для динамічного обміну даними з базою даних MySQL за допомогою технології Ajax при створенні веб-додатків для обліку документації в коледжі.

2. У рамках дипломного проєкту було втілено принцип інтеграції технології Ajax у веб-ресурс, а також реалізовано основні програмні аспекти використання Ajax на стороні клієнта. Вивчено методи декомпозиції завдань та складних викликів. Освоєно ключові підходи до структурування програмного коду, використання мови PHP та JavaScript. Розглянуті сервісні можливості веб-браузерів у різних операційних системах для досягнення кросбраузерної сумісності веб-ресурсу. Освоєно синтаксис MySQL та базові можливості роботи з реляційними таблицями для роботи з базою даних. Для роботи з Ajax вивчено використання JavaScript-фреймворка jquery.

3. У результаті розробки було створено веб-додаток для обліку документації в коледжі за допомогою HTML 5, PHP, Java та MySQL, з використанням Ajax технології. Цей додаток представляє собою повноцінну інтерактивну систему для взаємодії на веб-сайті, включаючи модуль заявок та можливість перегляду новин.

Зм.	Аркуш	№ докум.	Підпис	Дата	<i>ДТЕУ 121 02з-4.МР</i>			
Зав. каф.		Криворучко О.В.		01.11.23	<i>Модель інформаційної системи сучасного закладу освіти на прикладі коледжів</i>	Стадія	Аркуш	Аркушів
Керівник		Жирова Т.О.		01.11.23		ВП	53	57
Гарант		Котенко Н.О.		01.11.23		Факультет інформаційних технологій		
Розробив		Давидчук І.В.		01.11.23		2м курс, 2з група		

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Аксенов С. РНР у створенні інтернет-ресурсів. Київ: Генеза, 2014. 322 с.
2. Алгоритми та структури даних / уклад. О. В. Щербаков, Ю. Е. Парфьонов, В. М. Федорченко. Харків:ХНЕУ ім. С. Кузнеця, 2017. 58 с.
3. Архітектура комп'ютерних систем: конспект лекцій для студентів усіх форм навчання з курсу «Архітектура комп'ютерних систем» / Укладачі: Голотенко О.С. Тернопіль:Вид-во ТНТУ ім. Івана Пулюя, 2016. 120 с.
4. Бобрешов-Шишов Д.І. Динамічне керування структурою розподіленої бази даних. *Молодий вчений*. 2015. № 7. С. 51–53.
5. Богач І. В., Довгалець С. М., Дубовой В. М., Алгоритми розв'язання задач з програмування. Вінниця: ВНТУ, 2017 119 с.
6. Борис Т. В. Управління великими масивами даних в якості послуги. *Проблеми телекомунікацій*. К., 2014. С. 243–245.
7. Буч Г. Об'єктно-орієнтований аналіз та проектування з прикладами додатків на С ++, 2-е вид. / Пер. з англ. М.: «Видавництво Біном». 2018. 560 с.
8. Васильєв О. Програмування на С++ в прикладах і задачах: навч. Посібник К.: Ліра-К, 2017. 258 с.
9. Василюк В. Ю. Основи програмування мовою Сі++. Житомир, 2008. 120 с.
10. Войтенко В. В. Морозов А. В. Теорія та практика (мова С++). Житомир, 2002. 186 с.

					<i>ДТЕУ 121 02мз-4.МР</i>			
Зм.	Аркуш	№ докум.	Підпис	Дата	Модель інформаційної системи сучасного закладу освіти на прикладі коледжу  Список використаних джерел	Стадія	Аркуш	Аркушів
Зав. каф.		Криворучко О.В.		01.11.23		СВД	54	57
Керівник		Жирова Т.О.		01.11.23		Факультет інформаційних технологій 2м курс, 2з група		
Гарант		Котенко Н.О.		01.11.23				
Розробив		Давидчук І.В.		01.11.23				

11. Ганжела С. І. Основи інформатики з елементами програмування та сучасні інформаційні технології навчання. Кропивницький: РВВ ЦДПУ ім. В. Винниченка, 2017. 61 с
12. Додонов А. Г., Ландэ Д. В. Розпізнавання інформаційних операцій / А.Г. Додонов. К.: ООО «Инжиниринг», 2017. 284 с.
13. Кузь М. В., Соловко Я. Т. Методологія формування узагальненого критерію якості програмного забезпечення в умовах невизначеності. *Вісник Вінницького політехнічного інституту*. 2015. №5. С. 104–107.
14. Луцків А. М. Архітектури високопродуктивних систем опрацювання великих даних. *Інформаційні моделі, системи та технології*, 12–13 грудня 2018 року. Т.:ТНТУ, 2018. С. 75.
15. Матвієнко М. П. Алгоритми та структури даних:навч. посіб. / М. П. Матвієнко. К.: Видавництво «Ліра-К», 2014. 340 с.
16. Методи та моделі розроблення комп'ютерних систем і мереж:монографія / В. С. Пономаренко. Х.: Вид. ХНЕУ, 2016. 316 с.
17. Милорадов К. А. Підвищення ефективності взаємодії з клієнтами за допомогою хмарних сервісів. *Міжнародний журнал експериментальної освіти*. 2015. № 3–1. С. 70–71.
18. Пекарський Б. Г. Основи програмування: навчальний посібник. К.: Кондор, 2018. 270 с.
- 19.Тарарака В. Д. Архітектура комп'ютерних систем: навчальний посібник. Житомир:ЖДТУ, 2018. 383 с.
- 20.Фісун М., Дворецький М., Дворецька С. Побудова моделей для оптимізації структури бази даних вузла у корпоративних інформаційних системах // ІТКІ, 2020. vol 48, № 2, С. 52–60.

Зм.	Аркуш	№ докум.	Підпис	Дата	ДТЕУ 121 02мз-4.МР			
Зав. каф.		Криворучко О.В.		01.11.23				Модель інформаційної системи сучасного закладу освіти на прикладі коледжу
Керівник		Жирова Т.О.		01.11.23	Список використаних джерел	СВД	55	57
Гарант		Котенко Н.О.		01.11.23		Факультет інформаційних технологій 2м курс, 2з група		
Розробив		Давидчук І.В.		01.11.23				

- 21.Юрченко І. В., Сікора В. С. Інформатика та програмування. Частина 2. Чернівці: Видавець Яворський С. Н., 2015. 210 с.
- 22.Ярцев В. П. Розподілені бази даних: навчальний посібник. К. ДУТ 2018. 97 с.
- 23.Abualigah, L. 2021. Advances in Meta-Heuristic Optimization Algorithms in Big Data Text Clustering. *Electronics*. № 10, P. 101.
- 24.Acharjya D. P., Ahmed K. 2016. A survey on big data analytics: challenges, open research issues and tools. *International Journal of Advanced Computer Science and Applications*. № 7(2). P. 511–518.
- 25.Ahmed I, Ahmad M, Jeon G, Piccialli F. A framework for pandemic prediction using big data analytics. *Big Data Research* 100190. Epub ahead of print Jan 16 2021 DOI 10.1016 / j.bdr.2021.100190.
- 26.Alkurd R, Abualhaol I, Yanikomeroğlu H. 2020. Big-data-driven and AI-based framework to enable personalization in wireless networks. *IEEE Communications Magazine*. № 58(3). P. 18–24.
- 27.Altman, M., Wood, A., O'Brien, D.R» and Gasser, U. (2018). Practical Approaches to Big Data Privacy over Time. *International Data Privacy Law*. doi:10.1093 / idpl / ipx027.
- 28.AlZubi AA. 2020. Big data analytic diabetics using map reduce and classification techniques. *The Journal of Supercomputing*.№ 76(6). P. 4328–4337. DOI 10.1007 / s11227–018–2362–1.
29. Asencio-Cortés G, Morales-Esteban A, Shang X, Martínez-Álvarez F. 2018. Earthquake prediction in California using regression algorithms and cloud-based big data infrastructure. *Computers & Geosciences*. № 115. P. 198–210. DOI 10.1016 / j. cageo. 2017.10.011.

Зм.	Аркуш	№ докум.	Підпис	Дата	ДТЕУ 121 02мз-4.МР			
Зав. каф.		Криворучко О.В.		01.11.23	Модель інформаційної системи сучасного закладу освіти на прикладі коледжу	Стадія	Аркуш	Аркушів
Керівник		Жирова Т.О.		01.11.23		СВД	56	57
Гарант		Котенко Н.О.		01.11.23	Список використаних джерел	Факультет інформаційних технологій 2м курс, 2з група		
Розробив		Давидчук І.В.		01.11.23				

## ТЕХНІЧНЕ ЗАВДАННЯ

Модель інформаційної системи сучасного закладу освіти на прикладі коледжу

### НАЙМЕНУВАННЯ ТА ОБЛАСТЬ ЗАСТОСУВАННЯ.

Дана технічна робота поширюється на розробку, впровадження та підтримку «Модель інформаційної системи сучасного закладу освіти на прикладі коледжу». Область застосування: використання користувачами коледжу для ефективного комунікування з викладачами через інформаційну систему.

### ПІДСТАВИ ДЛЯ РОЗРОБКИ.

Підставою для розробки є завдання на виконання роботи кваліфікаційно-освітнього рівня «магістр інженерії програмного забезпечення», затверджене кафедрою інженерії програмного забезпечення та кібербезпеки Державного торговельно-економічного університету.

### МЕТА І ПРИЗНАЧЕННЯ РОЗРОБКИ.

Метою даного проекту є розробка клієнт-серверної системи для зручного та ефективного регулювання студентів з викладачами коледжу. Розробка призначена для працівників коледжу та студентів даного навчального закладу.

### ДЖЕРЕЛА РОЗРОБКИ.

Джерелом розробки є науково-технічна література з даної тематики, програмування на мові Java, HTML 5, Ajax, PHP, JavaScript та фреймворку

					<i>ДТЕУ 121 023-4.МР</i>			
<i>Зм.</i>	<i>Аркуш</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>				
Зав. каф.	Криворучко О.В.			06.09.23	Модель інформаційної системи сучасного закладу освіти на прикладі коледжу	Стадія	Аркуш	Аркушів
Керівник	Жирова Т.О.			06.09.23		ТЗ	57	57
Гарант	Котенко Н.О.			06.09.23	Технічне завдання	Факультет інформаційних технологій 2м курс, 23 група		
Розробив	Давидчук І.В.			06.09.23				

Spring, публікації в періодичних виданнях та в Інтернеті з даних питань.

#### ВИМОГИ ДО ПРОДУКТУ, ЩО РОЗРОБЛЯЄТЬСЯ.

- Зручний та зрозумілий користувацький інтерфейс.
- Збереження резервації подій з прив'язкою до часу, користувача
- Регулювання, додавання, зміна та видалення існуючих користувачів.
- Адмінський контроль за подіями, студентами.

#### ВИМОГИ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ.

- Операційна система MS Windows/Linux
- Java 8+
- PostgreSQL 12+
- IntelliJ Idea 2019

#### Вимоги до апаратної частини

- Процесор Intel або AMD.
- Оперативна пам'ять не менше 2Гб.
- Вільне місце на жорсткому диску не менше 30 Гб.

						Аркуш
						58
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 023-4.МР	



## ПРОГРАМА ТА МЕТОДИКА ТЕСТУВАННЯ

Структура діяльності веб-інтерфейсу інформаційної системи включає такі аспекти:

- реєстрація користувачів;
- авторизація користувачів;
- управління сесіями користувача;
- вихід користувача з системи;
- відправка документів;
- видалення документів;
- редагування документів;
- індикація документів;
- індикація статусу користувача.

Для вирішення завдання програмування всіх структурних компонентів веб-інтерфейсу інформаційної системи важливо встановити між ними взаємозв'язок. Насамперед, взаємозв'язок виявляється між такими елементами:

### 1. Реєстрація та авторизація користувачів.

У контексті програмування розумно розпочати з реєстрації та авторизації користувачів. Спершу створюємо систему аутентифікованого доступу, а потім базуючись на цьому створюємо повноцінний веб-сайт. Реєстрація та авторизація користувачів часто вважаються одним функціональним модулем для веб-сайту.

### 2. Управління сесіями користувача та вихід із системи.

Зм.	Аркуш	№ докум.	Підпис	Дата	<i>ДТЕУ 121 02з - 4.МР</i>			
Зав. каф.		Криворучко О.В.		06.09.23	<i>Модель інформаційної системи сучасного закладу освіти на прикладі коледжу</i>	Стадія	Аркуш	Архів
Керівник		Жирова Т.О.		06.09.23		ПМТ	59	57
Гарант		Котенко Н.О.		06.09.23	<i>Програма та методика тестування</i>	Факультет інформаційних технологій		
Розробив		Давидчук І.В.		06.09.23		2м курс, 2з група		

Після реєстрації чи авторизації користувача надається певний зв'язок з веб-сайтом, який забезпечується через сесію користувача. Сесія користувача та вихід з системи взаємодіють як дві складові частини завдяки спільній функціональності. Користувач може в будь-який момент завершити сесію на веб-сайті, тобто вийти з системи.

### 3. Передача, видалення та редагування документів.

Ці три аспекти управління документами в рамках роботи Веб-інтерфейсу інформаційної системи сполучаються в єдиний модуль, який застосовується в області користувача. Програмування цих трьох аспектів функціоналу буде ефективним, якщо їх поєднати на цьому етапі проєктування як єдиний модуль для керування замовленнями та документами користувачів.

### 4. Індикатор документів та індикатор стану користувачів.

Ці дві складові функції системи виступають як методи сповіщення про зміну стану інформації щодо заявок та користувачів у режимі реального часу. Розумно розглядати програмування цих двох аспектів як окрему функціональну взаємодію в межах веб-сайту.

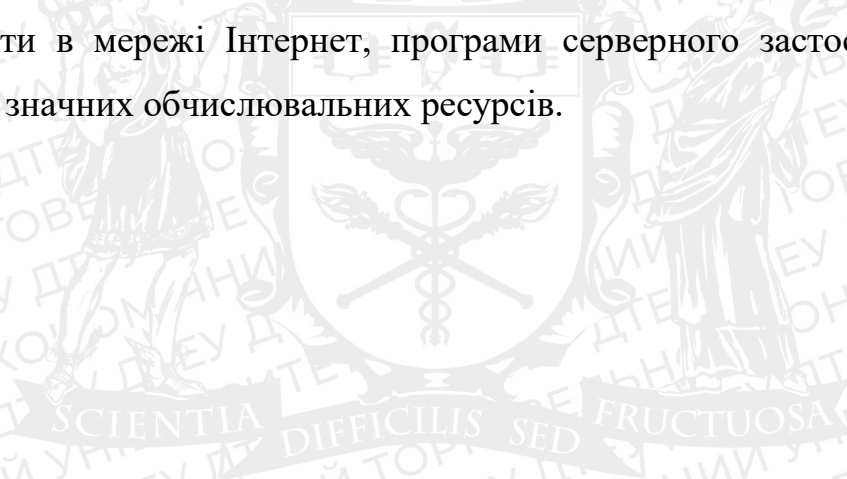
Після дослідження структури функціональної роботи Веб-інтерфейсу інформаційної системи було спрощено зв'язок між всіма частинами роботи системи.

Визначення апаратних рішень для ефективного функціонування та оптимальної роботи додатка буде залежати переважно від характеристик, які використовуються при виконанні роботи Веб-інтерфейсу інформаційної системи. Основна увага звертається на програмний комплекс роботи веб-сервера, який має прямий вплив на продуктивність Веб-інтерфейсу інформаційної системи. Вимоги до апаратних рішень сформульовані з огляду на багатофакторний вплив на роботу Веб-інтерфейсу інформаційної системи. Важливість належного функціонування нашого Веб-інтерфейсу

						Аркуш
						60
Зм.	Аркуш	№ докум	Підпис	Дата	ДТЕУ 121 02з-4.МР	

інформаційної системи визначається відповідністю програмного забезпечення. Програмний комплекс для Веб-інтерфейсу інформаційної системи складається з інтерконектованих програмних реалізацій, які забезпечують роботу самого Веб-інтерфейсу інформаційної системи.

Для забезпечення дієвості Веб-інтерфейсу інформаційної системи необхідний повноцінний веб-сервер. Веб-сервер включає у себе встановлене програмне забезпечення, що реалізує не лише протоколи зв'язку, але й забезпечує операційну діяльність Веб-інтерфейсу інформаційної системи. Наш веб-додаток взаємодіє безпосередньо лише з веб-сервером. Серверне обладнання вимагає налаштувань і підключених програмних компонентів для нормального функціонування веб-сайтів. Оскільки наш додаток призначений для роботи в мережі Інтернет, програми серверного застосування будуть вимагати значних обчислювальних ресурсів.



						Аркуш
					ДТЕУ 121 023-4.МР	61
Зм.	Аркуш	№ докум	Підпис	Дата		

# ДОДАТКИ

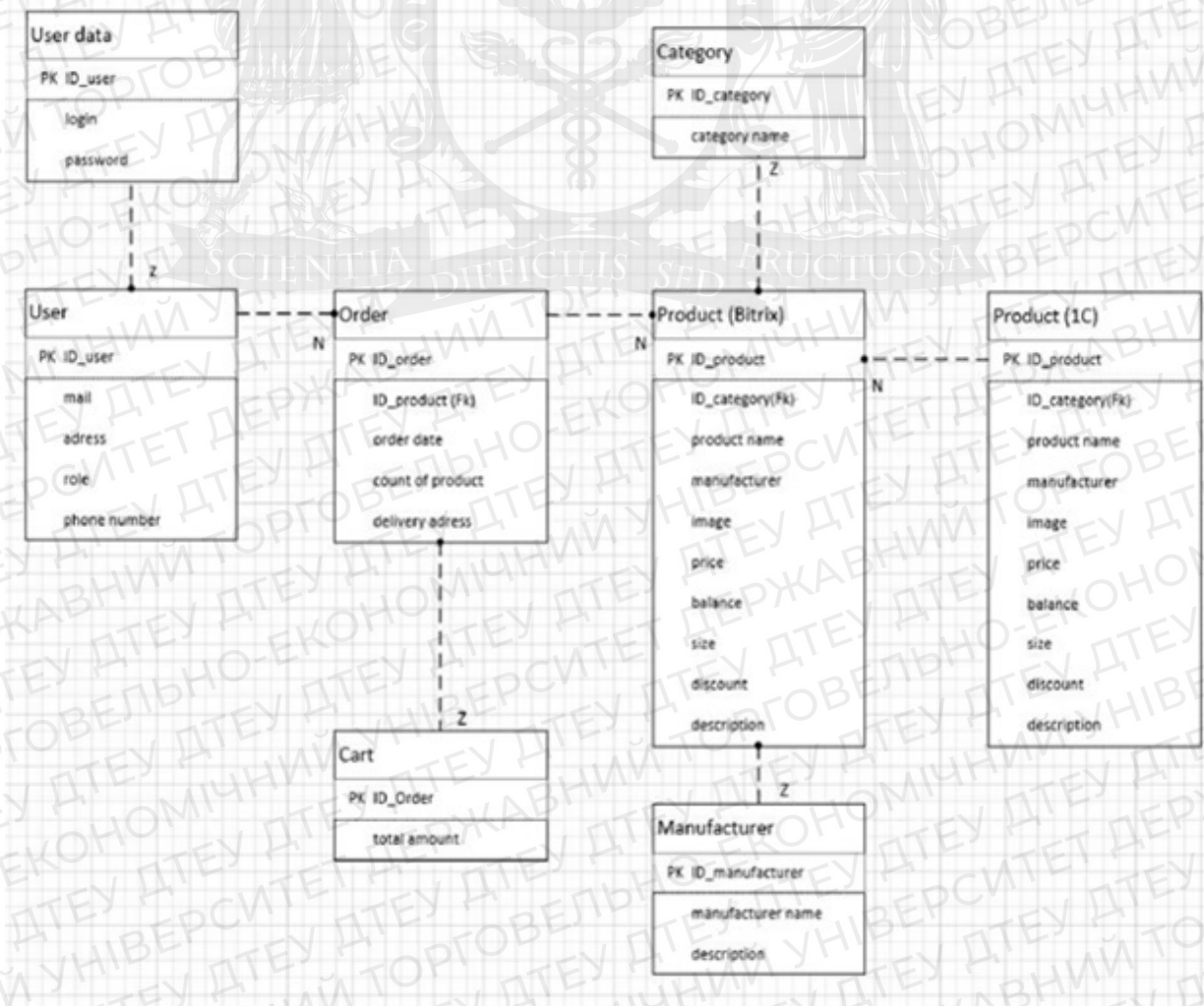
Додаток А

## Алгоритм побудови діагностичних моделей нелінійних динамічних систем в умовах великих даних

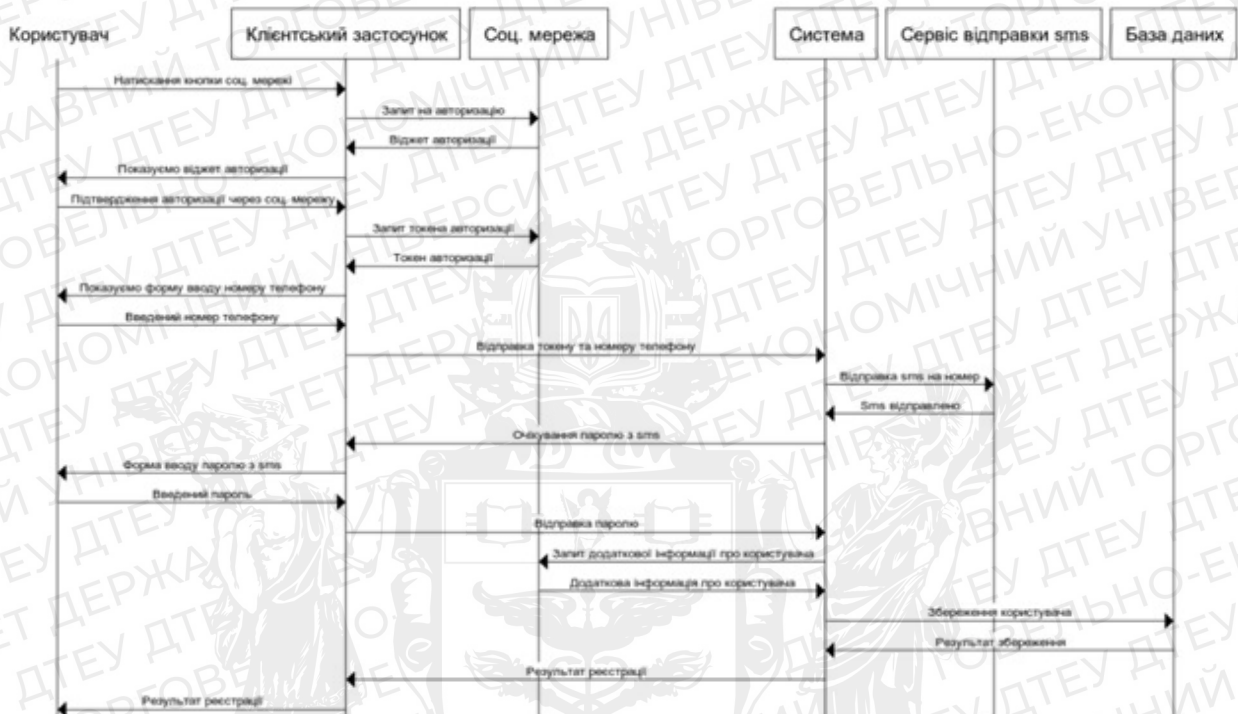


Додаток Б

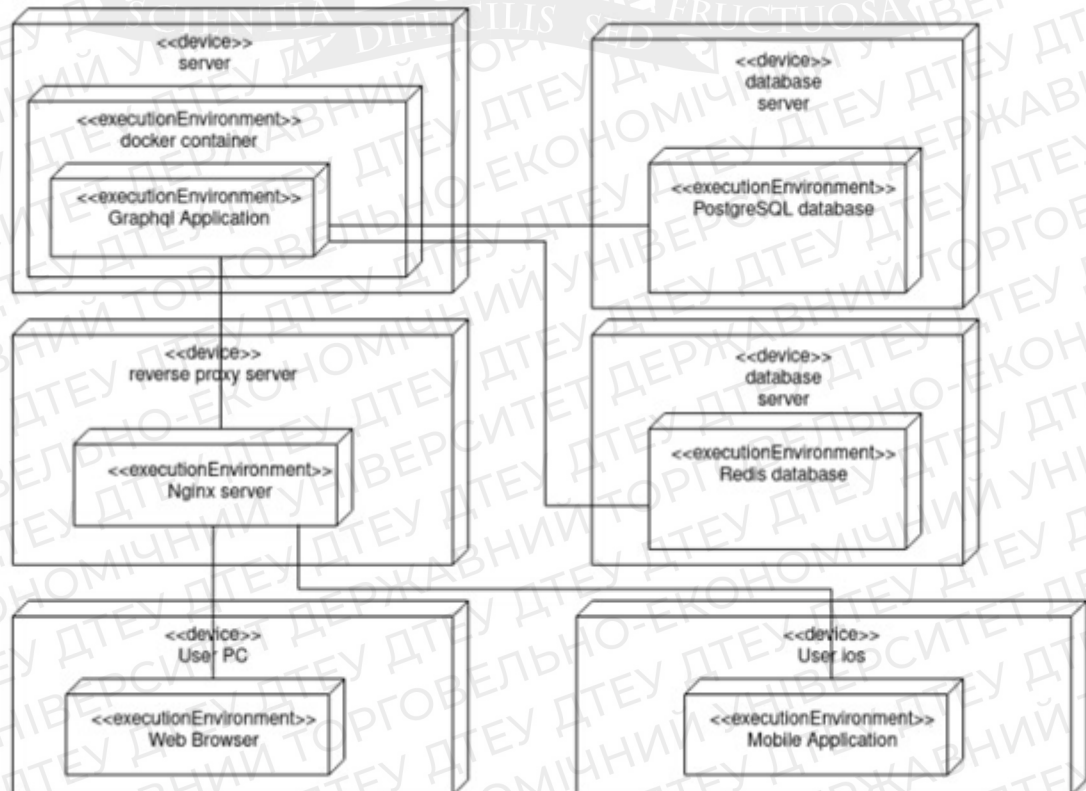
## Схематичне зображення взаємозв'язку системи з кластером



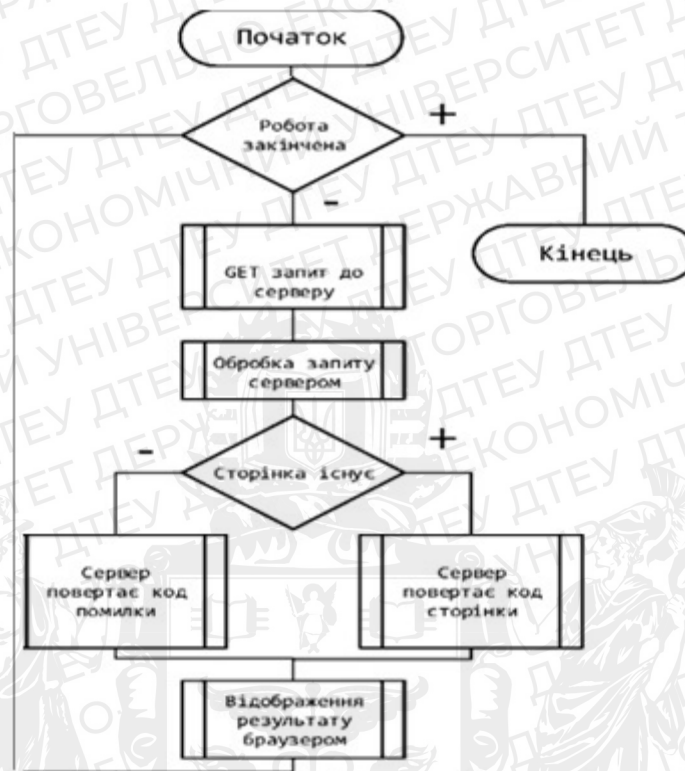
## Система інформаційної підтримки навчальних курсів на сайті навчального закладу



## Схема структурна розгортання програмного забезпечення



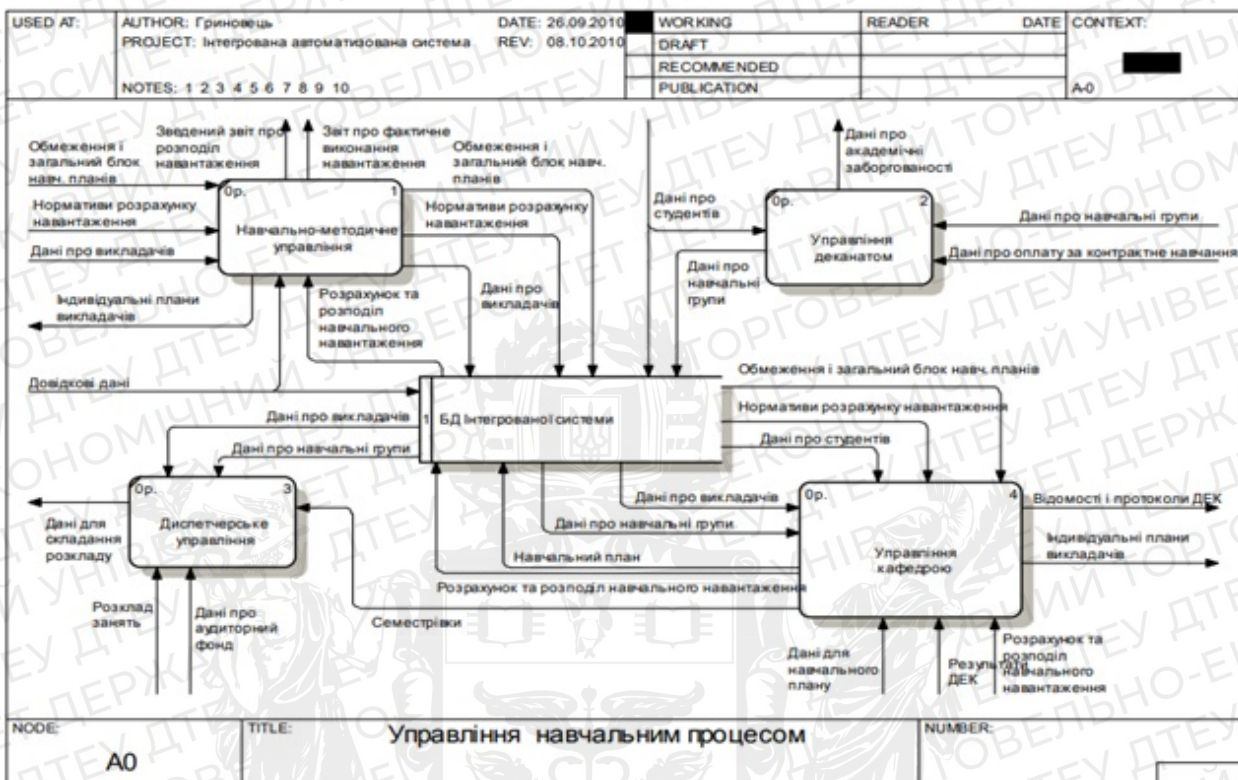
### Загальний алгоритм роботи інформаційної системи



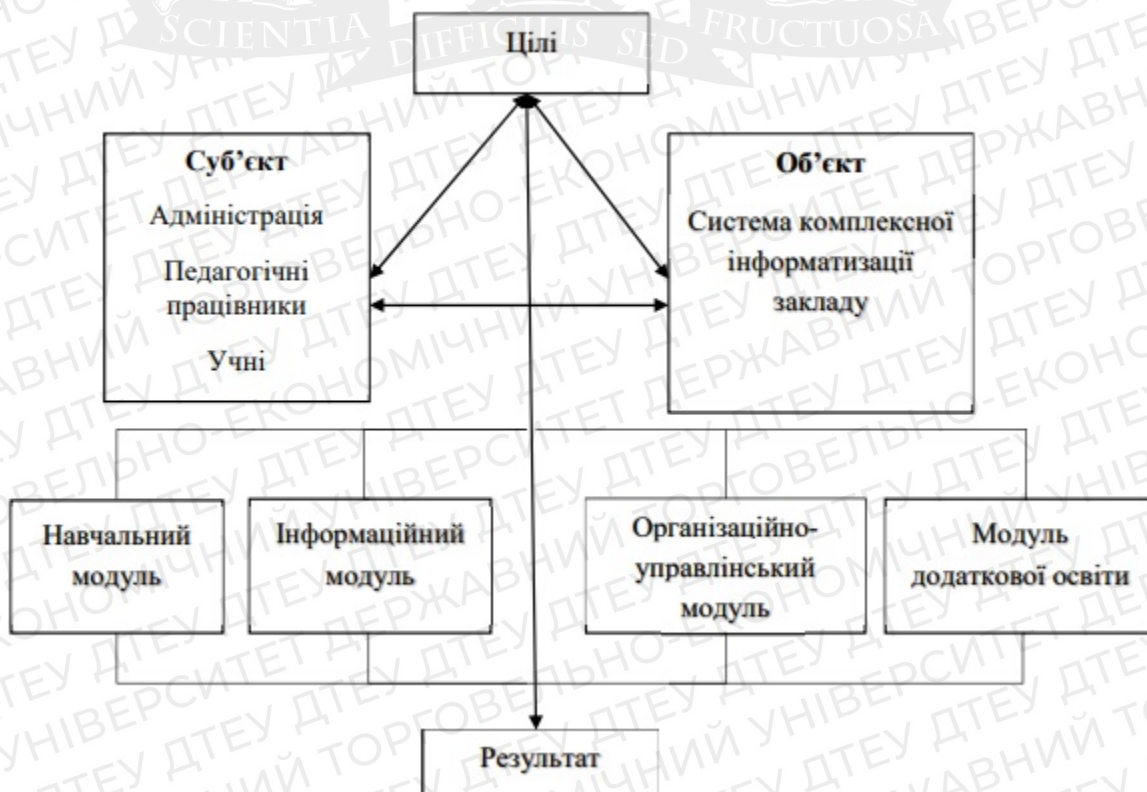
### Контекстна діаграма потоків даних інтегрованої автоматизованої системи



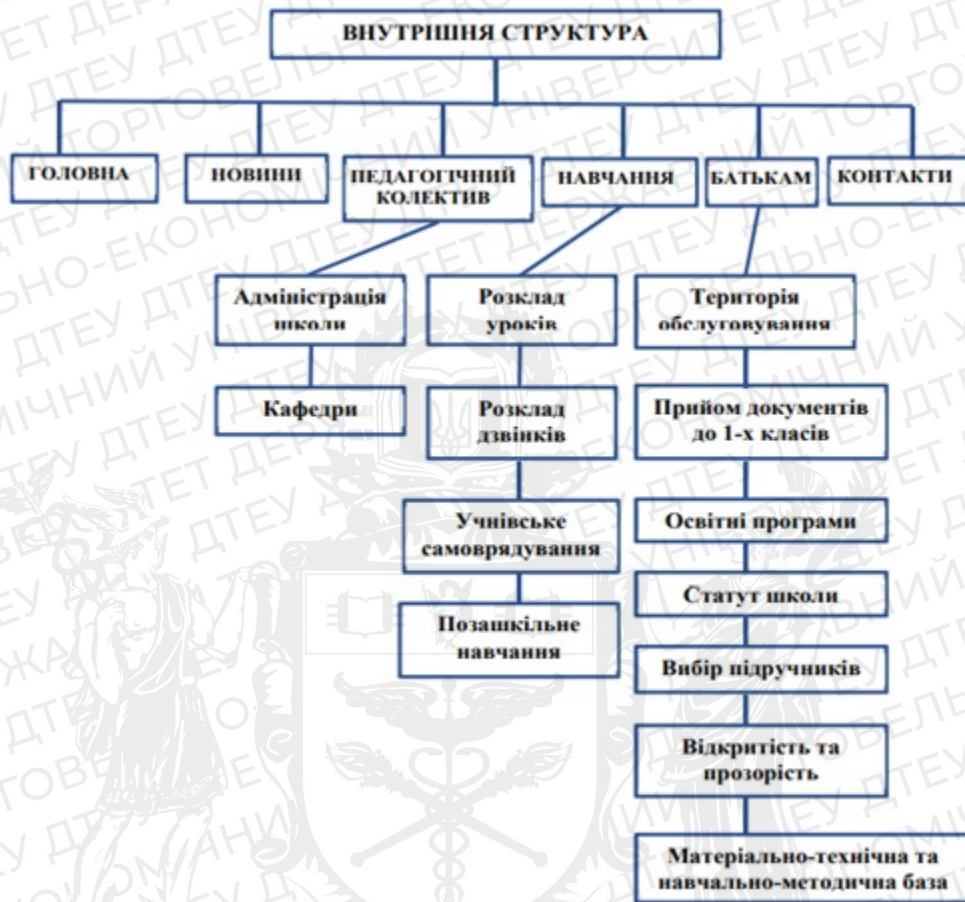
Діаграма потоків даних першого рівня



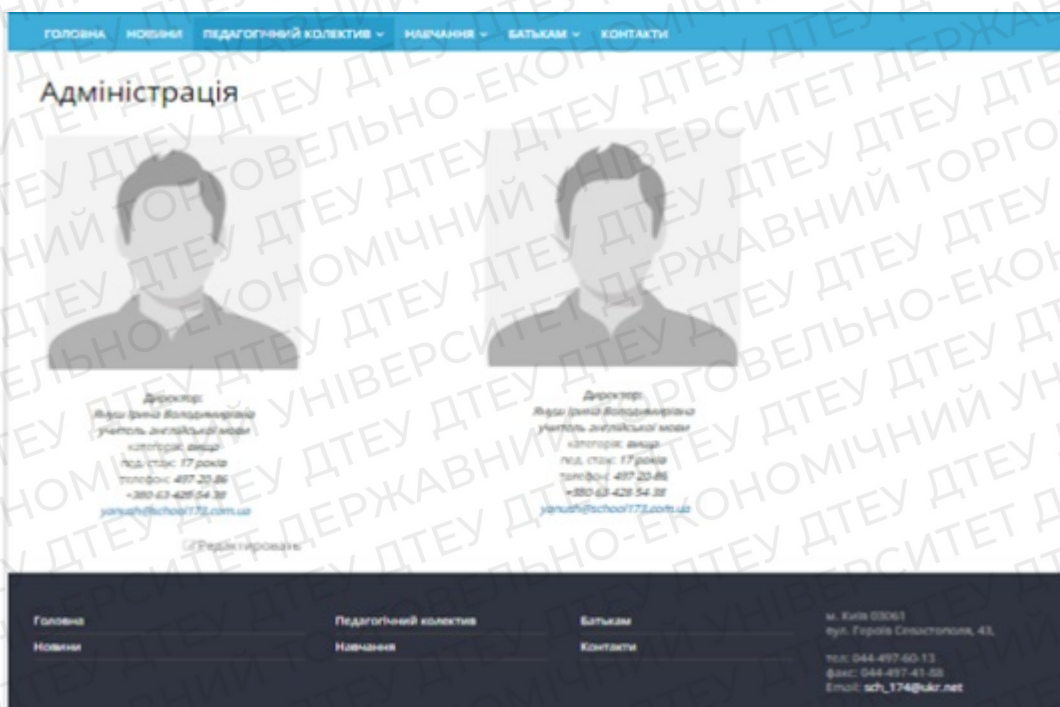
Інформаційна модель навчального закладу (коледжу)



**Внутрішня структура веб-порталу коледжу**



**Сторінка адміністрація навчального закладу**





## Фрагмент коду програмних модулів проєкту

```
Код файлу index.php
<!DOCTYPE html>
<html>
<head>
<title>Information Hub</title>
<meta charset=«UTF - 8»>
<meta name=«description» content=«Information Hub - Ваша однією зупинкою
для Information. Перегляньте нашу колекцію ресурсів та зв'яжіться з
фахівцями.»>
<meta name=«keywords» content=«інформаційна система коледжу»>
<meta name=«author» content=«InformationHub»>
<style>
/* CSS стилі для сторінки */
/* ... */
</style>
</head>
<body>
<h1>Ласкаво просимо до Information Hub</h1>

<?php
// Підключення до бази даних MySQL
$host = 'localhost';
$username = 'your_username';
$password = 'your_password';
$database = 'beautyindustryhub';

$conn = new mysqli($host, $username, $password, $database);

// Перевірка з'єднання з базою даних
if ($conn->connect_error) {
die(«Помилка з'єднання:». $conn->connect_error);
}

// Код для стартової сторінки
createHomePage();

// Форма авторизації
showLoginForm();

// Закриття з'єднання з базою даних
$conn->close();

function createHomePage() {
echo "<p>Ласкаво просимо до нашого каталогу в галузі краси! Перегляньте
наші ресурси та зв'яжіться з фахівцями.</p>»;
// Додатковий код для відображення ресурсів, акцій тощо
}

function showLoginForm() {
// Виклик JavaScript-методу для рендерингу форми авторизації
echo "<script src=«/render_login_form.js/»</script>»;
}
?>
</body>
</html>

Код файлу render_login_form.js:
// Код для рендерингу форми авторизації за допомогою JavaScript
var form = document.createElement(«form»);
```

```
form.setAttribute («id», «loginForm»);

var usernameLabel = document.createElement («label»);
usernameLabel.textContent = «Ім'я користувача:»;
var usernameField = document.createElement («input»);
usernameField.setAttribute («type», «text»);

var passwordLabel = document.createElement («label»);
passwordLabel.textContent = «Пароль:»;
var passwordField = document.createElement («input»);
passwordField.setAttribute («type», «password»);

var loginButton = document.createElement («button»);
loginButton.textContent = «Увійти»;
loginButton.addEventListener («click», function () {
    var username = usernameField.value;
    var password = passwordField.value;
    if (authenticateUser (username, password)) {
        alert («Успішний вхід!»);
        // Виконати дії після успішної авторизації
    } else {
        alert («Невірні облікові дані. Будь ласка, спробуйте ще раз.»);
    }
});

form.appendChild (usernameLabel);
form.appendChild (usernameField);
form.appendChild (passwordLabel);
form.appendChild (passwordField);
form.appendChild (loginButton);

document.body.appendChild (form);

function authenticateUser (username, password) {
    // Код для перевірки автентифікації користувача в базі даних
    var url = «jdbc:mysql://localhost:3306/beautyindustryhub»;
    var conn = java.sql.DriverManager.getConnection (url,
        «your_username», «your_password»);
    var statement = conn.prepareStatement («SELECT * FROM users WHERE
        username = ? AND password = ?»);
    statement.setString (1, username);
    statement.setString (2, password);
    var resultSet = statement.executeQuery ();
    var result = resultSet.next (); // Якщо рядок результату існує,
    користувач успішно авторизований
    conn.close ();
    return result;
}
```