

Міністерство освіти і науки України
Одеський національний технологічний університет
Кафедра комп'ютерної інженерії



**ПОЯСНЮВАЛЬНА ЗАПИСКА
ДО КВАЛІФІКАЦІЙНОЇ РОБОТИ**

на тему Розробка сайту інтернет-форуму
(назва кваліфікаційної роботи згідно наказу ОНТУ)

Здобувача Свірської А.І.
(прізвище, ініціали)

2 (скор.) курсу 543б групи

Керівники: к.т.н., доц. Шестопалов С.В.
(посада, прізвище та ініціали)
ст. викл. Сіренко О.І.
(посада, прізвище та ініціали)

Консультанти: _____
(посада, прізвище та ініціали)
Phd, ст.викл. Богданов О.О.
(посада, прізвище та ініціали)

Кваліфікаційна робота допускається до захисту

Рішення кафедри від 05.06 2024 р., протокол № 8
Завідувач кафедри комп. інженерії _____ **Сергій АРТЕМЕНКО**
(назва кафедри) (підпис) (Ім'я ПРІЗВИЩЕ)

Одеса – 2024 рік

ОДЕСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНОЛОГІЧНИЙ УНІВЕРСИТЕТ

Факультет комп'ютерної інженерії, програмування та кіберзахисту
Кафедра комп'ютерної інженерії
Ступінь вищої освіти бакалавр
Спеціальність 123 «Комп'ютерна інженерія»
Освітня програма Розробка ігор та інтерактивних медіа у віртуальній
реальності

ЗАТВЕРДЖУЮ

Зав. кафедри комп'ютерної інженерії
Сергій АРТЕМЕНКО
« 30 » серпня 2023 року

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ ЗДОБУВАЧА

Свірської Анастасії Іванівни

1. Тема роботи Розробка сайту інтернет-форуму

Затверджена наказом університету від « 30 » серпня 2023 р., наказ № 442-03

2 Термін здачі здобувачем закінченої роботи 28 травня 2024 р.

3. Вихідні дані роботи

1. Редактор коду Visual Studio Code. 2. Локальний сервер Open Server

3. Текстовий редактор Microsoft Word. 4. Редактор презентацій Microsoft PowerPoint.

4. Перелік питань, які потрібно розробити

1. Вступ. 2. Збір та аналіз інформації. 3. Проектування системи.

4. Розробка системи. 5. Економічні розрахунки.

6. Охорона праці. 7. Загальні висновки.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)

Слайд 1. Тема роботи Слайд 2. Об'єкт та предмет дослідження

Слайд 3. Актуальність. Слайд 4. Аналоги. Слайд 5. Розробка проекту.

Слайд 6. Користувацький інтерфейс Слайд 7. Функціонал проекту. Слайд 8. Схема БД.

Слайд 9. Економічні розрахунки. Слайд 10. Загальні висновки.

6. Консультанти по роботі, із зазначенням розділів роботи, що стосуються їх

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв
<i>Економіка</i>	<i>Phd, ст. викл. Богданов О.О.</i>		
<i>Охорона праці</i>	<i>д.т.н., проф. Артеменко С.В.</i>		
<i>Нормоконтроль</i>	<i>ст. викл. Сіренко О.І.</i>		

7. Дата видачі завдання 30.08.2023

Керівники

Сергій ШЕСТОПАЛОВ

Олександр СІРЕНКО

Завдання прийняв до виконання

Анастасія СВІРСЬКА

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів кваліфікаційної роботи	Термін виконання етапів роботи	Примітка
1.	<i>Дослідженні предметної області</i>	<i>26.10.2023</i>	
2.	<i>Дослідження існуючих аналогів</i>	<i>30.11.2023</i>	
3.	<i>Дослідження методів реалізації штучного інтелекту</i>	<i>28.01.2023</i>	
4.	<i>Проектування</i>	<i>15.02.2024</i>	
5.	<i>Розробка демонстраційної версії ПЗ</i>	<i>27.03.2024</i>	
6.	<i>Підготовка техніко-економічної частини</i>	<i>15.04.2024</i>	
7.	<i>Підготовка розділу охорони праці</i>	<i>15.04.2024</i>	
8.	<i>Оформлення пояснювальної записки</i>	<i>27.05.2024</i>	
9.	<i>Оформлення графічної частини та лістингу</i>	<i>27.05.2024</i>	

Здобувач-дипломник Анастасія СВІРСЬКА

Керівники роботи Сергій ШЕСТОПАЛОВ

Олександр СІРЕНКО

Несу відповідальність за ідентичність електронного та друкованого варіантів кваліфікаційної роботи, даю згоду на обробку персональних даних та не заперечую проти розміщення кваліфікаційної роботи на офіційних web-ресурсах ОНТУ.

Підтверджую, що в кваліфікаційній роботі відсутні порушення норм академічної доброчесності.

Здобувач-дипломник Анастасія СВІРСЬКА

АНОТАЦІЯ

Кваліфікаційна робота присвячена розробці програмної частини сайту форуму. Правильна реалізація програмної частини є основним моментом у створенні позитивного користувацького досвіду. Платформи для форумів вимагають стабільної роботи та безпеки, тому програмна частина має забезпечити швидкий доступ до контенту та захист даних користувачів.

У першому розділі розглянуті особливості розробки форумів, різні функціональні можливості, проведено аналіз існуючих аналогів.

У другому розділі сформоване технічне завдання. Створено концептуальний та дизайнерський документ з програмної частини, де прописано структуру сайту, основні функції та вимоги до користувацького інтерфейсу.

У третьому розділі обґрунтовано вибір засобів та описано процес розробки програмної частини сайту форуму. Вибрано відповідні технології для забезпечення продуктивної роботи, безпеки.

У четвертому розділі проведена оцінка ефективності розробки сайту форуму, описано економічний та соціальний ефект від розробки проекту.

У п'ятому розділі розглянуто питання охорони праці при розробці та експлуатації сайту.

Результатом роботи є демонстраційна версія сайту форуму, в якій реалізовано реєстрацію користувачів, створення та обговорення тем, а також функції пошуку.

Ключові слова: форум, користувацький інтерфейс, технології.

ABSTRACT

The qualification work is devoted to the development of the software part of the forum website. The correct implementation of the software part is the main point in creating a positive user experience. Platforms for forums require stable operation and security, so the software part must provide quick access to content and protection of user data.

In the first section, the peculiarities of the development of forums, various functional possibilities, and the analysis of existing analogues are considered.

In the second section, the technical task is formed. A conceptual and design document has been created for the software part, where the site structure, main functions and requirements for the user interface are spelled out.

The third section substantiates the choice of tools and describes the process of developing the software part of the forum site. Appropriate technologies are selected to ensure productive work and safety.

The fourth chapter evaluates the effectiveness of the development of the forum site, describes the economic and social effect of the project development.

In the fifth section, the issue of labor protection during the development and operation of the site is considered.

The result of the work is a demo version of the forum site, which implements user registration, creation and discussion of topics, as well as search functions.

Keywords: forum, user interface, technologies.

ЗМІСТ

	стор.
ВСТУП.....	8
РОЗДІЛ 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ	10
1.1 Опис принципів роботи сайтів, взаємодія клієнта та сервера	10
1.2 Опис структур веб сайтів	14
1.3 Опис принципів роботи форумів	16
1.4 Огляд існуючих форумів, їх порівняння	19
1.5 Динамічні та статичні сайти	22
1.6 Складання завдання на диплом	23
Висновки до першого розділу	24
РОЗДІЛ 2 ПРОЕКТНА ЧАСТИНА	25
2.1 Функціональності форуму	25
2.2 Структура форуму	27
2.3 Архітектурні рівнів форуму.....	29
2.4 Основні технології.....	30
2.5 Безпека	31
2.6 Алгоритми функціональних компонентів.....	33
2.7 UML-діаграма послідовності дій на форумі	35
Висновки до другого розділу.....	36
РОЗДІЛ 3 ПРАКТИЧНА РЕАЛІЗАЦІЯ	37
3.1 Створення інтерфейсу\веб дизайну	37
3.2 Розробка бекенду.....	41
3.2.1 Структура скриптов проекту.....	41
3.2.2 Використання <i>POST\GET</i> запитів	43

					<i>КРБ.КІ.1.442-03.3.11</i>					
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>						
<i>Розробив</i>		<i>Анастасія СВИРЬКА</i>			<i>Розробка сайту інтернет-форуму</i>			<i>Літ.</i>	<i>Арк.</i>	<i>Акрушів</i>
<i>Перевірив</i>		<i>Олександр СІРЕНКО</i>								
<i>Рецензент</i>		<i>Денис ПОПОВ</i>								
<i>Нормоконтроль</i>		<i>Олександр СІРЕНКО</i>								
<i>Затвердив</i>		<i>Сергій АРТЕМЕНКО</i>						<i>гр. 543б, ОНТУ</i>		

3.2.3 Розробка та реалізація схеми бази даних	45
3.3 Вибір технологій для реалізації	50
3.4 Реалізація захисту інформації.....	54
3.5 Збірка проекту.....	55
3.6 Тестування та відлагодження форуму	58
Висновки до третього розділу	60
РОЗДІЛ 4 ТЕХНІКО-ЕКОНОМІЧНА ЧАСТИНА	61
4.1 Організаційно-економічне й маркетингове обґрунтування проекту .	61
4.1.1 Порівняльний техніко-економічний аналіз	61
4.1.2.Організаційне обґрунтування проекту	62
4.2 Економічні розрахунки проекту	65
4.2.1 Бізнес план проекту	65
4.2.2 Розрахунки економічної ефективності	67
Висновки до четвертого розділу.....	74
РОЗДІЛ 5 ОХОРОНА ПРАЦІ.....	75
5.1 Шкідливі та небезпечні чинники на робочому місці.....	75
5.2 Попередження впливу шкідливих та небезпечних чинників.	76
5.3 Вимоги до приміщення.	78
5.4 Пожежна безпека	79
Висновки до п'ятого розділу	81
ЗАГАЛЬНІ ВИСНОВКИ.....	82
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	83
ДОДАТОК А ГРАФІЧНІ МАТЕРІАЛИ.....	86
ДОДАТКИ Б ЛІСТИНГ КЛАСІВ.....	86

ВСТУП

У сучасному інформаційному суспільстві Інтернет відіграє важливу роль в обміні знаннями, досвідом та ідеями, пошуку інформації, поглинанню контенту.

В останній час соціальні мережі стають все більш популярними і широко використовуються для спілкування\обміну інформацією і створення спільнот. Соціальні мережі надають користувачам корисний інструмент для спілкування в режимі реального часу, публікації контенту та підтримки контактів з іншими користувачами. Однак, незважаючи на всі свої переваги, форуми також залишаються потрібним інструментом для спілкування та обміну інформацією в Інтернеті. На відміну від соціальних мереж, форуми зазвичай організовані за інтересами(по темам), надаючи користувачам можливість глибше обговорити конкретні питання саме з певної теми, отримати якісні відповіді. Форуми дозволяють користувачам знаходити інформацію і створювати довгострокові дискусії та спільноти з різноманітними, майже завжди з незнайомими людьми, що для соціальних мереж є незручним та вкрай рідкісним явищем.

Дослідницький фокус проекту полягає в розробці форуму, який надає користувачам можливість спілкуватися, ставити питання, обмінюватися ідеями і знаходити відповіді на свої питання.

Об'єкт дослідження – процес створення веб застосунку, а саме форуму, включаючи функціональні, технічні та користувацькі аспекти проекту.

Предметом дослідження є розробка як дизайну інтерфейсу, з урахуванням користувацького досвіду, так і функціональних можливостей, що використовуються у форумі.

Цілями дослідження є:

1. Аналіз потреб користувачів (визначення потреб за цільової аудиторії форуму).
2. Розробка архітектури веб сайту.
3. Створення зручного та привабливого інтерфейсу (інтуїтивно зрозумілий дизайн).

					КРБ.КІ.1.442-03.3.11	Арк.
						8
Змн.	Арк.	№ докум.	Підпис	Дата		

4. Розробка та впровадження методів захисту даних користувачів.
5. Оптимізація продуктивності (забезпечення швидкої і надійної роботи сайту).
6. Впровадження системи управління контентом.

Завдання дослідження:

1. Вивчення аналогів (аналізувати існуючі аналоги, вивчення їх позитивних та негативних сторін).
2. Збір вимог (проектування технічного завдання).
3. Проектування архітектури проекту (визначення основних розділів форуму, створення структури сайту).
4. Дизайн інтерфейсу (створити макет, що був би зручним у використанні та привабливим).
5. Розробка та тестування .

					<i>КРБ.КІ.1.442-03.3.11</i>	<i>Арк.</i>
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		9

РОЗДІЛ 1

АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ДОСЛІДЖЕННЯ ІСНУЮЧИХ АНАЛОГІВ

1.1 Опис принципів роботи сайтів, взаємодія клієнта та сервера

Інтернет – глобальна розподільна мережа, що охоплює багато взаємопов'язаних автономних систем, які взаємодіють через протоколи, апаратне та програмне забезпечення та побудований на клієнт–серверній архітектурі. Архітектура клієнт-сервер є широко розповсюдженою концепцією у створенні розподілених мережних застосунків і передбачає взаємодію та обмін даними між ними. Сервери – вузли, що надає послуги в мережі та дозволяють приєднуватися до себе та перенаправляє інформацію, отримує її чи обробляє як служба чи застосунок. Клієнт – мережний вузол, що користується іншими серверами.[20]

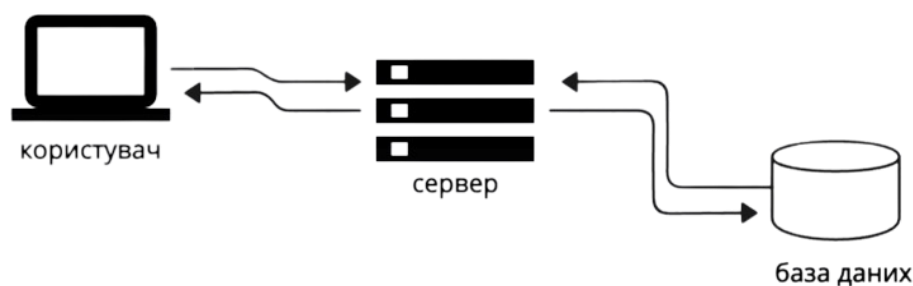


Рис. 1.1 – Схематичне представлення архітектури клієнт-сервер

Клієнтом може бути:

- браузер;
- десктопний застосунок;
- інший сервер.

Робота даної архітектури потребувало логічного розділення на два незалежних рівня – клієнтський та серверний.

Програма клієнтського рівня:

- робити запити на конкретні дані;
- відображати дані;
- не має права змінювати дані.

Програма серверного рівня:

					КРБ.КІ.1.442-03.3.11	Арк.
						10
Змн.	Арк.	№ докум.	Підпис	Дата		

– зберігати інформацію при рестарті системи.

Таким чином бази даних виступають додатковим з'єднувачем в архітектурі, збільшуючи надійність системи.

HTTP (*HyperText Transfer Protocol*) – протокол передачі даних прикладного рівня, що на початку використовувався для передачі HTML файлів, а наразі за допомогою нього можна передавати майже будь-які дані.[19]



Рис. 1.3 – Запрос HTTP

Також *HTTP* використовується як протокол транспортного рівня для передачі інформації протоколами прикладного рівня (наприклад *SOAP*, *XML-RPC*, *WebDAV*) або *API* програмних продуктів використовують його для передачі даних в будь-яких форматів (передача здійснюється через TCP/IP-з'єднання).

В *HTTP* методи не мають суворого значення, тобто він не регулює що саме має робити сервер при отриманні того чи іншого методу. Сервер також не має розуміти всі методи, обов'язковий тільки *GET*.

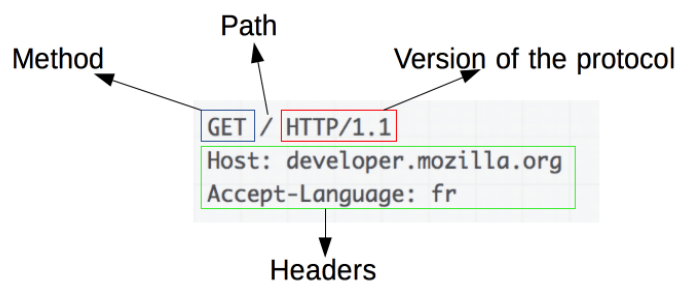


Рис. 1.4 – Складові запису HTTP

Коди стану (три цифри, що визначають результат виконання запроса) звітів *HTTP*[18]:

- від 100 – інформаційні відповіді;
- від 200 – запит успішно виконано;
- від 300 – повідомлення о перенаправленні;
- від 400 – помилка на стороні клієнта;
- від 500 – помилка сервера.

Протокол *HTTP* не має функцій захисту, але має розширення, що реалізує шифрування між веб сторінкою та пристроєм користувача – *HTTPS* (*HyperText Transfer Protocol Secure*). Дані протоколу *HTTPS* передаються поверх криптографічних протоколів *TLS/SSL*.

В яких випадках має використовуватися протокол *HTTPS*:

- якщо на сайті використовуються персональні дані;
- при випадку коли сайт проводить онлайн платежі.

API (application programming interface) – архітектурний стиль, що вказує як найбільш ефективно спілкуватися клієнту та серверу за допомогою *HTTP\HTTPS*[17]



A Blog Post by Ayushi Rawat

Рис. 1.5 – API як посередник між сервером та користувачем

REST API – архітектурний стиль\набір правил для застосунку сервера. *REST API* на сьогодні це найпоширеніший *API*–інтерфейс в Інтернеті. Клієнт надсилає запити на сервер у вигляді даних. Сервер використовує ці дані для запуску внутрішніх функцій і повертає їх клієнту.

					КРБ.КІ.1.442-03.3.11	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		13

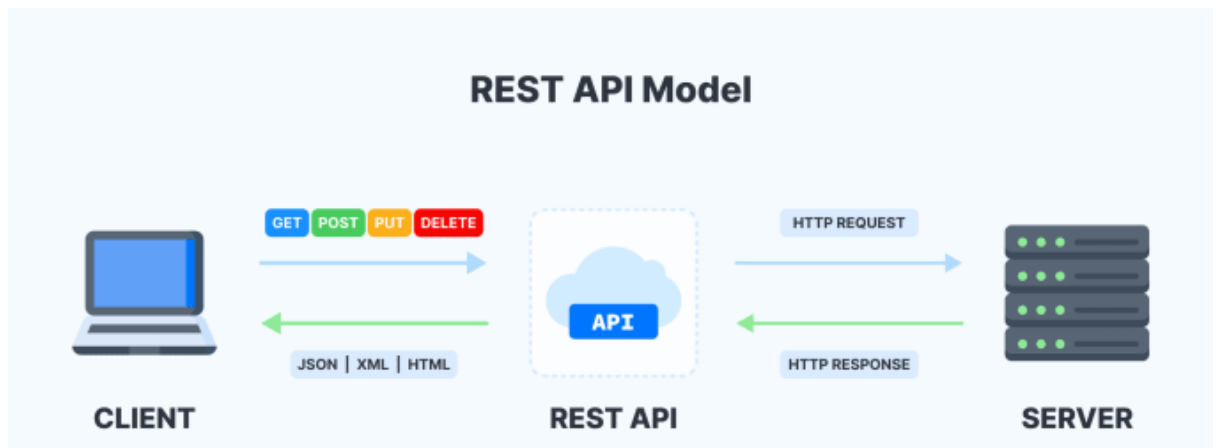


Рис. 1.6 – REST API запити, взаємодія між REST API, клієнтом і сервером

Концепції *REST API*[16]:

- модель взаємодії клієнт–сервер;
- система може бути багаторівневою (мати декілька рівнів\серверів);
- відсутність стану (сервер не повинен мати якийсь стан тобто кожного разу клієнт та сервер спілкуються як перший раз);
- одноманітний уніфікований інтерфейс;
- кешування;
- формат обміну даних – *JSON\XML*;
- версіонування.

1.2 Опис структур веб сайтів

Веб сторінки об'єднують сайти за допомоги гіперпосиланнями. Зручна і зрозуміла структура допомагає користувачеві легко знаходити потрібну інформацію в інтернет ресурсі. Важливо зазначити, що структура сайту може бути зовнішня і внутрішня. Зовнішня структура – макет сторінки із зазначенням розташування на ній блоків. Внутрішня відображає категорії, належність до них певних сторінок і матеріалів.

Внутрішня структура сайту – це схема розташування всіх сторінок ресурсу, його розділів та підрозділів. Тобто певний вид компонування інформації

					<i>КРБ.КІ.1.442-03.3.11</i>	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		14

за допомогою навігації, різних меню, перехресних посилань чи набір *URL*, що розташовані в певній послідовності.

Загалом виділяють п'ять типів структур веб сайтів[15]

Лінійна – всі сторінки з'єднані гіперпосиланнями послідовно.

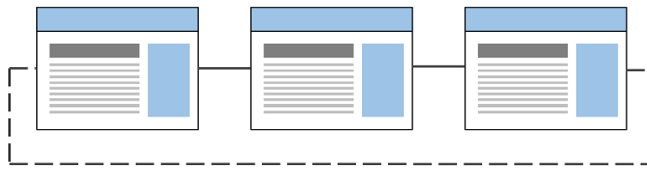


Рис. 1.7 – Лінійна структура сайту

Решітка - розміщення сторінок за принципом ієрархії, посилання дозволяють переміщатися по горизонталі ієрархії та вниз; складно використовувати з великою кількістю сторінок.

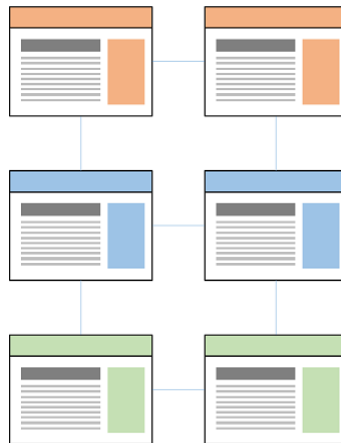


Рис. 1.8 – Структура сайту – решітка

Деревоподібна - сторінки можна поділити на рівні, тобто є початкова, головна сторінка з якої є посилання на сторінки другого рівня, звідти – на одну зі сторінок третього рівня.

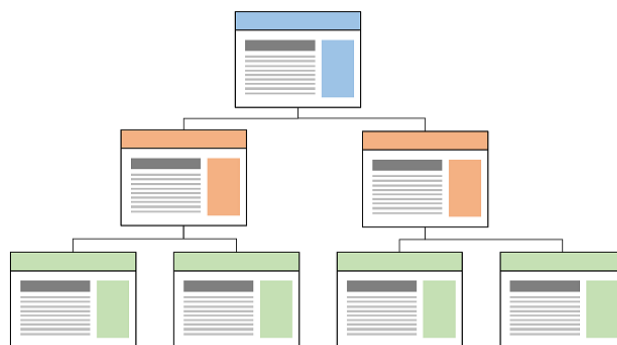


Рис. 1.9 – Деревоподібна структура сайту

Павутиння - неупорядкована, управління сильно ускладнюється з ростом числа сторінок, та довільна.

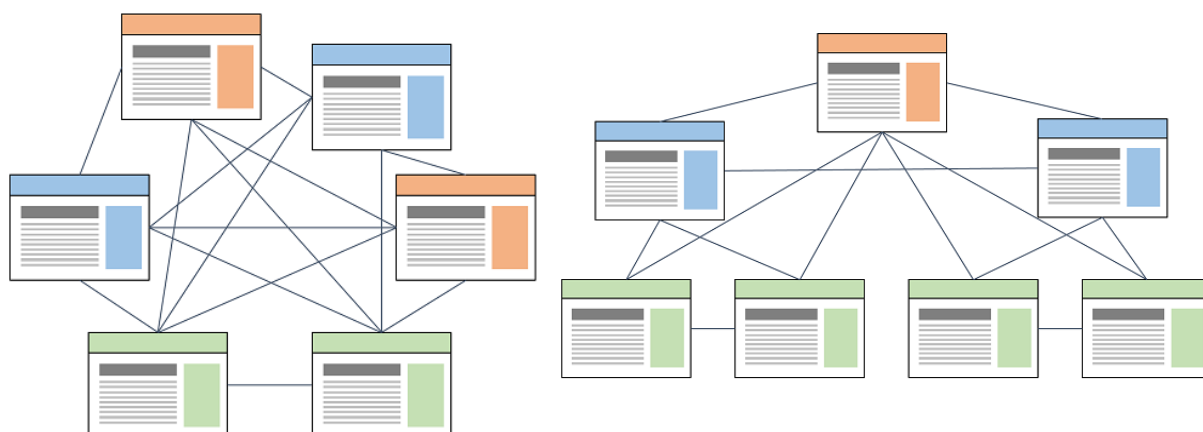


Рис. 1.10 – Довільна структура та структура павутиння сайту

Певні структури притаманні певним видам сайтів та використовуються у певних ситуаціях. Зважаючи, що форум це багатосторінковий сайт раціональним рішенням було б створювати деревоподібну структуру через простоту розуміння.

1.3 Опис принципів роботи форумів

Форум – інтернет сервіс\веб застосунок для спілкування користувачів за певною тематикою, певна форма соціальної мережі, яка більше сконцентрована на спілкування з ціллю пошуку відповіді на специфічне питання чи проблему. Концепція полягає в тому, щоб створювати тему з наступним обговоренням шляхом розміщення повідомлень в середині тем. Теми зберігаються в базах даних форуму, в подальшому можуть бути корисними користувачам Інтернету. Щоб орієнтуватися в незліченній кількості сторінок важливо впровадження зручної та зрозумілої користувачам структури.

Структура сайту – це схема розміщення його сторінок, навігація за допомогою *URL*, що розташовані в певній послідовності та прослідковується логічний зв'язок. Обговорення однієї тематикою об'єднуються для зручності у відповідні розділи, ієрархія веб-форуму має наступний вигляд:



Рис. 1.11 – Приклад ієрархії форуму

Теми та розділи можуть бути статичними та динамічними. Повідомлення зазвичай мають вигляд: «автор – тема – зміст – дата/час». Ці повідомлення та всі відповіді на нього з наступними коментарями утворюють «гілку» чи «топ».

Існують три типи подання повідомлень користувачів у темі:

- деревоподібне (список повідомлень може мати різні рівні ієрархію в залежності від того, кому була надана відповідь, як створення нової гілки на дереві);
- лінійне (повідомлення в рамках однієї теми публікуються один під одним у міру надходження; нові повідомлення містяться останніми);
- топ (повідомлення фільтруються з урахуванням реакцій користувачів)



Рис. 1.12 – Приклад деревоподібного та лінійного подання повідомлень

Особливості веб застосунків форумів, вимоги, що висуваються до застосунку за теперішніми стандартами:

1. Адаптивність (наразі більшість Інтернет трафіку припадає саме на мобільні пристрої, якщо продукт не оптимізувати для смартфонів і планшетів буде втрачена значна кількість потенційних користувачів)[14].
2. Інтуїтивний інтерфейс (звичайно тут все залежить від попереднього досвіду користувача, але інтерфейс має бути зрозумілим і простим, наприклад, зрозуміла ієрархія гілок обговорення, зручний фільтр за темами чи пошук, сортування за кількістю відповідей, датою чи актуальністю тощо).
3. Гармонійно підібрані\налаштовані кольори та шрифти (при створенні дизайну та верстки важливо брати до уваги не тільки різні розміри цифрових пристроїв, ще й можливі вади користувачів, наприклад дальтонізм, що є поширеним захворюванням і при використанні мало контрастних кольорів унеможлиблює використання ресурсу).
4. Підтримка реєстрації і авторизації користувачів, захист акаунтів від злому через двухфакторну ідентифікацію.
5. Налаштування\кастомізація профіля, нікнейм (У користувачів повинна бути можливість налаштовувати сторінки своїх профілів під себе, зміна нікнейму, кольорів сторінки, аватарки, тощо).
6. Система досягнень\рейтинг (хоч і не обов'язкова складова, але може сприяти більше часу проводити на ресурсі, тобто стимулювати розвиток форуму).
7. Зручний текстовий редактор (в застосунки взаємодія в котрому відбувається через текст, мають бути інструменти для мінімального редагування).
8. Можливість прикріплення файлів.

					<i>КРБ.КІ.1.442-03.3.11</i>	<i>Арк.</i>
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		18

9. Захист від спаму, вірусних посилань, шахраїв (впровадження всіх можливих заходів для блокування, попередження негативного впливу на користувачів та їх пристрої).
10. Підтримка різних рівнів доступу (адміністратор, модератор, учасник, гість).

1.4 Огляд існуючих форумів, їх порівняння

Виділяю декілька категорій порівнянь для форумів:

1. Функціональність:

- a) пошук по сайту;
- b) фільтри за темами;
- c) можливість реєстрації;
- d) система управління правами доступу (розподіл користувачів на ролі).

2) Дизайн:

- a) зручність навігації;
- b) загальне візуальне враження;
- c) адаптивність під різні пристрої;
- d) розмір\колір\контрастність кольорів тексту.

3) Можливості:

- a) реакцій на повідомлення;
- b) прикріплення файлів;
- c) особисті повідомлення;
- d) створення публікацій\постів;
- e) вибір мови, теми;
- f) персональне оформлення аккаунту.

Для порівняння виділяю такі форуми:

- 4PDA (<https://4pda.to/forum/index.php?act=idx>);
- REPLACE (<https://replace.org.ua/>);
- Український форум колекціонерів (<https://xn--e1aajfbegedw6d6k.xn--j1amh/index.php>);

					КРБ.КІ.1.442-03.3.11	Арк.
						19
Змн.	Арк.	№ докум.	Підпис	Дата		

– DOU (<https://dou.ua/forums/>);

– WOX (<https://wox.in.ua/>).

Таблиця 1.1

Порівняння форумів за функціональністю

	4PDA	REPLACE	WOX	DOU	Український форум колекціонерів
Пошук	✓	✓	✓	✗	✓
Фільтри	✓	✗	✗	✗	✓
Можливість реєстрації	✓	✓	✓	✓	✓
Система управління правами доступу	✓	✓	✓	✓	✓

Виділяю такі мінуси з критерію функціональність:

1. На *REPLACE* при переході через гіперпосилання веб сторінка відкривається не в новій вкладці, а у вкладці самого форуму, через що його можна втратити з поля зору і відкритих вкладках.
2. Неможливість відповідати на дописи без верифікації у веб застосунку.
3. Незручний пошук, коли при помилці в слові чи запиту словосполучення запит не дасть результату.
4. Складна та довга реєстрація.

Якщо і було унеможливлено відповідати на дописи без аккаунту то процес створення має бути максимально швидким та легким. Наприклад, використання нескладної капчи, як на *REPLACE*, менша кількість обов'язково заповнених полів с усунення непотрібних таких як дата народження, регіон проживання, регіон народження, ім'я бабусі чи кличка домашньої тварини.

					КРБ.КІ.1.442-03.3.11	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		20

E-mail
Поточна дійсна e-mail адреса.

Ім'я користувача
Від 2 до 25 символів.

Столиця України
Захист від спам-ботів

Рис. 1.13 – Скріншот полів реєстрації *REPLACE*

5. Багато пустих гілок з не розпочатими темами. Заповнення сторінки це звичайно важливо, але не пустими гілками і з не розпочатим діалогами.

6. Захована можливість відповіді у теми: або вона губиться серед іншої інформації, або розташована у кінці розмови, що унеможлиблює відповідь, якщо не пролистати всі існуючі відповіді.

7. Система репутації, як заохочення користувачів до взаємодії, корисна, але жодного разу не вказує на правдивість\коректність відповіді.

8. Відсутність можливості реакції на конкретну відповідь, що дало б можливість сортування відповідей за релевантністю.







ФОРУМ	Теми	Повідомлень	Останнє повідомлення
 Експертиза та оцінка	0	0	Немає повідомлень
 Обігові	0	0	Немає повідомлень
 Ювілейні та пам'ятні	6	13	2 гривні 1999 Солов'яненко TarDen  19 грудня 2023, 21:17
 Набори монет	0	0	Немає повідомлень
 Аксесуари для монет України	0	0	Немає повідомлень

Рис. 1.14 – Скріншот з Українського форуму колекціонерів

Відмінності між форумами за дизайном виразні та з спільного можна виділити такі недоліки та вдалі рішення[12]:

1. Багато інформації, що складає візуальний шум та не є важливою для будь-якого пересічного користувача, наприклад якщо час і дата можуть вказати на актуальність питання, то інформація з якого пристрою, країни, скільки повідомлень було надруковано не дасть нам

ніяких корисних даних. Така інформація доречна на сторінці з профілем користувача, а не при обговоренні якоїсь теми.

2. Розміщення тем та відповідей у вигляді таблиці, що дає змогу не губитися через кількість.
3. Якщо оформлення має певний задум, як Український форум колекціонерів, то гарним варіантом було б усунення функції завантаження власного фото для аватарки, щоб зберегти стилістику.
4. Жодний форум не підтримував можливість змінення теми;
5. Більшість має адаптації під різні пристрої.

1.5 Динамічні та статичні сайти

Всі сайти можна умовно розділити на 2 групи, за способом заповнення сторінки інформацією: статичні і динамічні. Статичним називається веб сайт, що складається з незмінних сторінок, тобто користувач буде переглядати сторінку у тому вигляді, в якому вони зберігаються на сервері.

Статичні сайти мають низку переваг:

1. Простота в розробці та підтримці.
2. Швидкість завантаження сторінок на стороні клієнта.
3. Безпека.
4. Дешевий хостинг.
5. Надійність\менший шанс на поломку.

Оскільки статичні сайти не мають складної логіки та механізмів взаємодії з користувачем, то й менше схильні до серйозних несправностей. Але й така обмеженість в функціоналі має недоліки:

1. Масштабованість.
2. Труднощі в оновленні.
3. Малий функціонал.

Статичні сайти, без втручання людини з знаннями *HTML/CSS/JavaScript*, не змінити (це ускладнює підтримку проекту), або не розширити, кожену сторінку потрібно відредагувати окремо. Динамічний сайт складається з динамічних, тобто змінюваних сторінок. Такі сторінки генеруються програмно

					КРБ.КІ.1.442-03.3.11	Арк.
						22
Змн.	Арк.	№ докум.	Підпис	Дата		

в реальному часі на основі запитів користувачів. Прикладом динамічного сайту можуть бути: соціальні мережі, інтернет магазини, форуми, платформи з онлайн курсами, пошукові системи.[13]

Переваги динамічний сторінок:

- гнучкість (сайт змінюється в залежності дій користувача);
- управління вмістом (адміністратори можуть додавати чи редагувати вміст без знань програмування);
- функціональність;
- масштабність (вміст зберігається в базі даних і генерується динамічно, що полегшує управління великими обсягами даних та сторінок).

Також динамічні сайти мають й низку недоліків:

- складність в розробці, потреба значних технічних знань та ресурсів;
- швидкодія в даному випадку є слабкою стороною через прив'язку до баз даних;
- безпека (на даному типі сайтів більше вразливостей);
- вартість хостингу (потреба в більш потужному хостінгу потребує відповідного грошового ресурсу).

Вибір буде залежить від потреб вашого проекту. Статичні сайти більше підходять для простих сайтів або сайтів-візиток, де зміни контенту відбуваються рідко. Для більш складних проектів з високим рівнем інтерактивності і частими оновленнями контенту створюють динамічні сайти.

1.6 Складання завдання на диплом

Виділяю такі критерії за дизайном, яким має відповідати кінцевий продукт:

- максимально контрастні кольори;
- можливість вибору кольорової гами веб сторінки;
- адаптивність для різних пристроїв;
- лише необхідна інформація, прибирання візуального шуму;
- розміщення всією додаткової інформації користувача на сторінки профілю;
- динамічний список категорій тем, який формується користувачами;

					<i>КРБ.КІ.1.442-03.3.11</i>	Арк.
						23
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		

Виділяю такі критерії за функціональністю, яким має відповідати кінцевий продукт:

- фільтрація за датою створення та активністю;
- зручний редактор коментарів;
- швидка та проста реєстрація;
- можливість відповіді у темі без аутифікації;
- реакції на окремі повідомлення користувача, що буде відображати повідомлення вгорі;
- можливість пошуку за ключовими словами;
- створення системи репутації

Висновки до першого розділу

У даному розділі розглянуто ключові елементи функціонування сайтів, форумів включно, структуру, принципи роботи. Порівняли відмінності між динамічними та статичними сайтами, а також проведено порівняння популярних платформ для форумів. Це дає змогу розуміти, як веб сайти взаємодіють з користувачами, як організована внутрішня структура з урахуванням відмінностей у функціоналі і як це впливає на продуктивність.

					<i>КРБ.КІ.1.442-03.3.11</i>	<i>Арк.</i>
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		24

РОЗДІЛ 2

ПРОЕКТНА ЧАСТИНА

2.1 Функціональності форуму

Веб застосунки, а саме форуми, будуть прості у використанні, коли матимуть чіткий та інтуїтивно зрозумілий інтерфейс користувача, що полегшує навігацію та пошук відповідних гілок обговорень. Форум також повинен мати функцію пошуку, яка дозволяє швидко знаходити цікаві теми за власними інтересами. Такі функції, як ланцюжок повідомлень, можливість легко цитувати та відповідати на конкретні повідомлення, а також зрозумілі інструменти модерації як для користувачів, так і для адміністраторів, можуть покращити взаємодію з користувачем. Не слід забувати про адаптивний дизайн, який добре працює як на персональних комп'ютерах\ноутбуках, так і на мобільних пристроях. Чіткі вказівки та правила взаємодії, а також активна модерація для забезпечення повного та інклюзивного середовища є важливими для позитивного досвіду користувача.

Форум - це веб застосунок, де можна реалізувати багато різних функцій, але основними, які притаманні саме форуму є[11]:

1. Створення гілки. Користувачам повинна надаватися можливість створювати гілок для обговорення різних тем чи то для пошуку відповідей на специфічні. при створення такої гілки має вказуватися категорія, до якої входить тема обговорення, заглавне питання чи заголовок та опис та інші додаткові дані.
2. Додавання повідомлень. Знаходячі гілку з обговоренням цікавої теми для користувача повинна надаватися можливість відповіді, коментування, висловлювання власної думки. Це є основним функціоналом, що має бути реалізований на сайті форуму.
3. Редагування та видалення повідомлень. Автори повідомлень можуть мати можливість редагувати або видаляти свої повідомлення. Це

					КРБ.КІ.1.442-03.3.11	Арк.
						25
Змн.	Арк.	№ докум.	Підпис	Дата		

дозволить користувачам коригувати свої думки або виправляти помилки.

4. Підтримка різних рівнів доступу. Різні рівні доступу надають права користування різним функціоналом форуму, наприклад, модератори можуть керувати темами та повідомленнями, а адміністратори мають повний доступ до функціоналу форуму. Але також різні рівні доступу мають низку переваг та недоліків. Якщо користувач не зареєстрований, то форум може обмежити його в функціоналі, надавати функцію участі в обговоренні, створенні\редактури відповідей. з одного боку це є протидією до спаму на форумі, з іншого є негативним користувацьким досвідом. Даний функціонал є основним для форуму, що дозволяє користувачам обмінюватися думками, але задля доброго користувацького досвіду можуть бути реалізовані і другорядні можливості:
5. Приватні повідомлення. Забезпечить приватну комунікацію між користувачами, дозволяючи обговорювати особисті питання та теми, але призводить до відсутності обміну інформацією у відкриті.
6. Можливість редагування власного профілю. Персоналізація профілю надає можливість розміщення про себе достовірної інформації, дозволяє змінювати оформлення за власним смаком.
7. Тегування повідомлень. Дозволяє значно точніше знаходити цікаві теми для обговорення, допомагає швидко орієнтуватися у темі розмови.
8. Рейтинг та вподобайки. Можливість сортування тем за популярністю.
9. Пошук та сортування. Забезпечує швидкий доступ до потрібної інформації\теми, але тільки при коректно побудованому пошуку, в іншому випадку недостатня точність може призвести до неефективного результату. Функції пошуку, сортування чи фільтрації

					<i>КРБ.КІ.1.442-03.3.11</i>	Арк.
						26
Змн.	Арк.	№ докум.	Підпис	Дата		

не є постійними, дані параметри скидаються при наступному відвідуванні форуму.

10. Можливість вставки медіа-файлів. Доповнює текстовий контент візуально, що в деяких моментах може бути дуже ефективно, але також може спричинити перевантаження сторінки.
11. Функція підписки на теми. Легкий доступ до актуальної інформації.
12. Інтеграція з соціальними мережами\поштою для швидкої реєстрації та в подальшому авторизації.
13. Створення анкет для опитування. Може сприяти залученню людей до певного обговорення\проблеми, дає можливість отримати відгуки від користувачів, що зацікавлені в певній категорії\області.

2.2 Структура форуму

Структура форуму є дуже важлива, бо від неї залежить зручність користування веб застосунком.

Структурувати форум можна на такі мітки:

- розділи\категорії;
- теми;
- підтеми;
- теги та ключові слова;
- розміщення важливої інформації на відокремленій сторінці.

Розділи(категорії) поділяють теми обговорення. Наприклад, якщо форум про ігри мав би категорії за жанрами ігор. Розділи можуть мати назву, опис, ключовий тег, що буде прив'язаний до кожної теми\підтеми розділу. Теми розділяють на підтеми для відокремлення конкретних питань. Так само як і розділи, теми можуть мати опис та обов'язково заголовки, ключовий тег, підтема ж матиме додаткова автора та час створення, кількість відповідей. У підтемі користувачі можуть залишати повідомлення. Кожне повідомлення матиме автора, час створення, зміст, при розширеному функціональні медіа-елементи.

					<i>КРБ.КІ.1.442-03.3.11</i>	<i>Арк.</i>
						27
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		

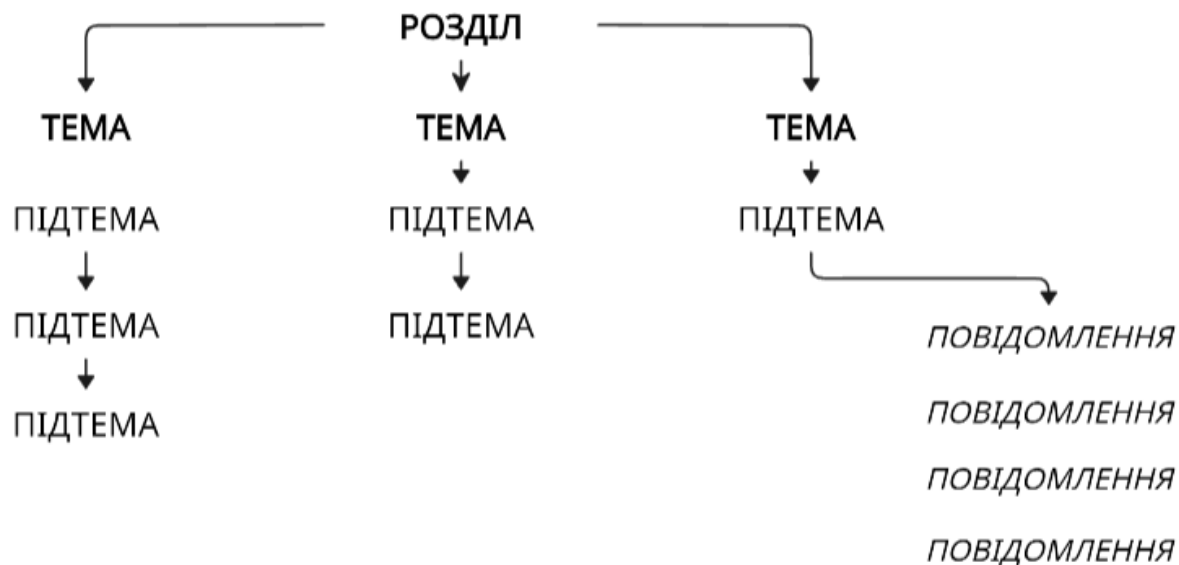


Рис. 2.1 – Схематичне представлення структури форуму

Кожен зареєстрований користувач на форумі має свій профіль. Профіль може містити наступну інформацію:

- дата створення;
- дата останнього відвідування форуму;
- інформація про себе;
- загальна кількість відповідей;
- категорії, якими цікавиться автор;
- перелік створених підтем;
- статус (чи є користувач на форумі в реальному часі);
- рівень доступу (користувач, модератор, адміністратор);
- аватарка.

Дана структура організовує форум таким чином, щоб користувачі з легкістю знаходили та обговорювати теми, в яких зацікавлені, адміністратори та модератори забезпечували порядок і контролювали діяльність на форумі, запобігають порушенню правил користування форуму.

2.3 Архітектурні рівні форуму

Архітектура сайтів – це важливий фактор веб проекту, призначений для структурування та упорядкування інформації, щоб забезпечити зручну навігацію та хороший досвід користувачів.

Frontend та *Backend* є основними частинами веб розробки. Поділ програмних систем на frontend і backend спрощує розробку і ізолює підтримку. *Frontend* — це інтерфейс для взаємодії між користувачем і *backend*, з яким користувач взаємодіє безпосередньо. *Frontend* відповідає за структуру, дизайн і функціональність веб сторінки. *Backend* - це частина веб-сайту або додатку, яка працює за кулісами і відповідає за обробку даних, взаємодію з базою даних, логіку програми та взаємодію з сервером. Backend забезпечує коректну роботу сайту, виконує запити від фронтенду, обробляє дані і повертає результат користувачеві.[10]

Архітектурні рівні форуму:

1. Клієнтська частина (*frontend*):

- a. веб інтерфейс (користувачі взаємодіють з форумом через інтерфейс, де переглядають розділи, теми, підтеми, повідомлення інших користувачів.);
- b. відображення даних\інформації (*frontend* відповідає за відображення даних, таких як повідомлення, профілі користувачів, списки тем та розділів);
- c. взаємодія користувача з застосунком(наприклад, створення нових тем, додавання відповідей, реєстрація\аутифікація).

2. Серверна частина (*backend*):

- a. логіка застосунку (обробка запитів користувачів, операції додавання, редагування, видалення даних.);
- b. керування користувачів (аутифікація, реєстрація, назначення різних прав доступу);

					КРБ.КІ.1.442-03.3.11	Арк.
						29
Змн.	Арк.	№ докум.	Підпис	Дата		

с. взаємодія з БД (зв'язок між базою даних для зберігання та отримання інформації про розділи, теми, повідомлення, користувачів тощо.).

3. База даних:

- а. зберігання даних у таблицях БД;
- б. взаємодія з *backend*.

Ці архітектурні рівні спів пов'язані між собою для функціональності форуму, відображення даних для користувачів та забезпечення безпеки та ефективності роботи системи.



Рис. 2.2 – Таблиця *users*

2.4 Основні технології

Основні технології для створення веб застосунку:

1. *HTML* (*HyperText Markup Language*): мова верстки, що використовується для створення структури веб сторінки[9].
2. *CSS* (*Cascading Style Sheets*): опис стилів\параметрів вигляду елементів *HTML* для оформлення зовнішнього вигляду сторінки[8].

3. *JavaScript* (JS): мова програмування, що використовується для створення інтерактивних елементів на веб сторінці[7].
4. *PHP* (*Hypertext Preprocessor*): Серверна мова програмування, що використовується для обробки запитів, взаємодії з базами даних, генерації динамічного *HTML*-коду, а саме[6]:
 - a. обробка *HTTP*-запитів;
 - b. аутентифікація та авторизація користувачів;
 - c. обробка даних форм сторінок;
 - d. взаємодія з БД;
 - e. генерація динамічних веб сторінок.
5. *SQL* (*Structured Query Language*): мова запитів, що використовується для взаємодії з БД, приймає участь у таких моментах[5]:
 - a. збереження даних користувачів, тем, повідомлень, категорій, створених гілок спілкування тощо;
 - b. виконання запитів на створення, читання, оновлення чи видалення даних;
 - c. забезпечення цілісності та безпеки даних.

Функціональність форуму реалізують дані технології.

Реєстрація чи аутентифікація користувачів відбувається наступним чином: за допомогою *HTML* та *CSS* створюються форми для запитів вводу даних, *PHP* обробляє дані та взаємодіючи з *SQL* зберігає їх в БД. Ті ж самі процеси будуть відбуватися й при реалізації іншого функціоналу будь-то створення тем, відповідь на повідомлення чи зміна аватарки у профілі.

2.5 Безпека

Метою веб безпеки є запобігання тих чи інших атак на застосунок, тобто захист веб сайту від несанкціонованого доступу, використання, модифікації, знищення, розповсюдження даних користувачів або збою веб застосунку. Для ефективної безпеки перш за все слід звернути увагу на розробку, включаючи налаштування сервера, створення паролів та політики оновлення та створення

					<i>КРБ.КІ.1.442-03.3.11</i>	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		31

2.6 Алгоритми функціональних компонентів

Створення форуму, на якому користувачі можуть спілкуватися, вимагає розробки безлічі функціональних елементів і алгоритмів для забезпечення його ефективної роботи. Нижче наведено опис ключових функціональних елементів і алгоритмів, необхідних для роботи таких сайтів.

Алгоритм реєстрація користувача:

1. Користувач вводить свої дані (логін, пароль, адреса електронної пошти).
2. Сервер перевіряє унікальність логіна.
3. Пароль хеширується.
4. Дані зберігаються в базі даних.

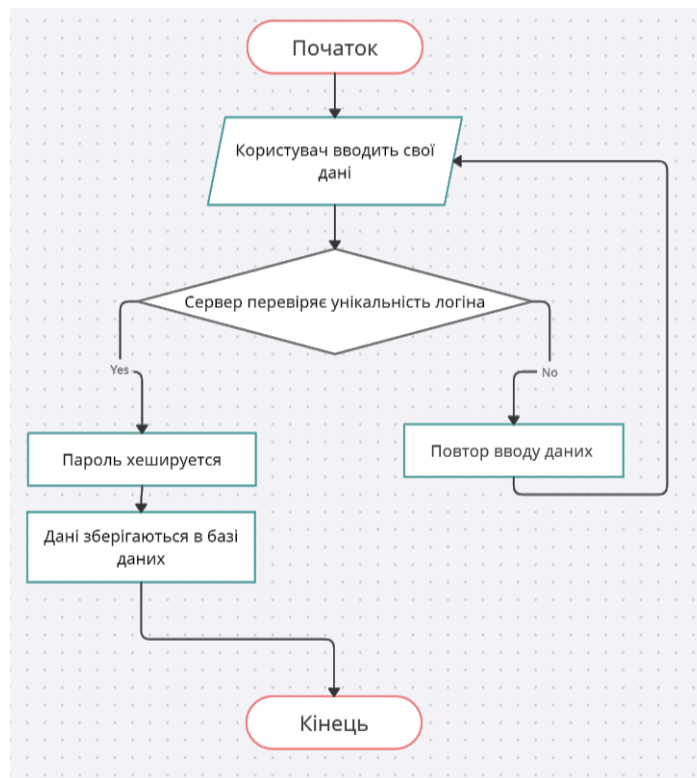


Рис. 2.6 – Алгоритм реєстрації користувача

Алгоритм аутентифікації користувача:

1. Користувач вводить ім'я користувача та пароль.
2. Сервер витягує хеш пароля з бази даних під час входу.
3. Перевірте відповідність між введеним паролем і хешем.
4. Якщо перевірка пройшла успішно, створюється сесія.

2.7 UML-діаграма послідовності дій на форумі

UML (уніфікована мова моделювання) - це уніфікована мова моделювання, яка використовується розробниками програмного забезпечення для візуалізації процесів та систем[1].

Опис дій реєстрація на форумі:

1. Користувач відкриває сторінку реєстрації.
2. Вводить необхідні дані (ім'я користувача, пароль, email).
3. Надсилає дані на сервер.
4. Сервер перевіряє дані, створює новий обліковий запис і надсилає підтвердження користувачу.

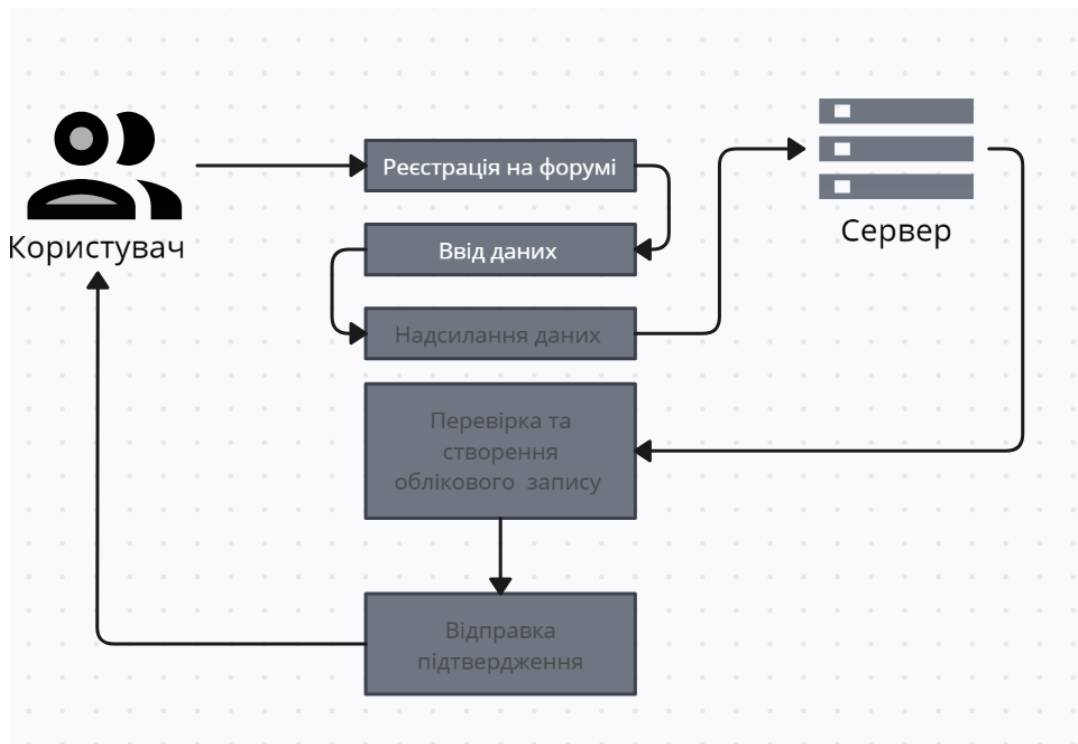


Рис. 2.8 – Діаграма опису дій при реєстрації

Користувач взаємодіє з інтерфейсом, який надсилає запити до серверу. Сервер вже обробляє ці запити та взаємодіє з БД. Кожна дія (реєстрація, вхід, створення теми, додавання відповіді, вихід) включає перевірку і обробку даних, а також відповідний зворотний зв'язок для користувача. Діаграма компонентів є одним з елементів мови моделювання UML і являє собою статичну структурну діаграму, яка декомпозує програмну систему на структурні компоненти і показує

Дану сторінку будуть бачити користувачі при чистому створеному чату. З мірою активності на даній сторінці будуть з'являтися коментарі користувачів.

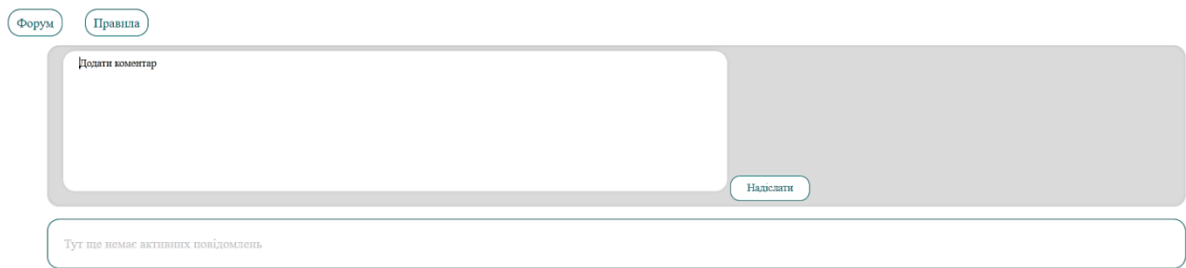


Рис. 3.2 – Пуста сторінка чату

Кожен форум чи будь-яка соціальна мережа мають правила спільноти для позитивного досвіду користувачів. Ці обмеження є необхідними для легального існування форуму.



Рис. 3.3 – Сторінка з правилами форуму

Сторінка користувача має бути функціональною, відображати всі активні чати, опис, якщо користувач його додав, його ім'я. Ця сторінка має бути для користувача активної, з котрої він зможе потрапити до будь-якого чату\обговорення.

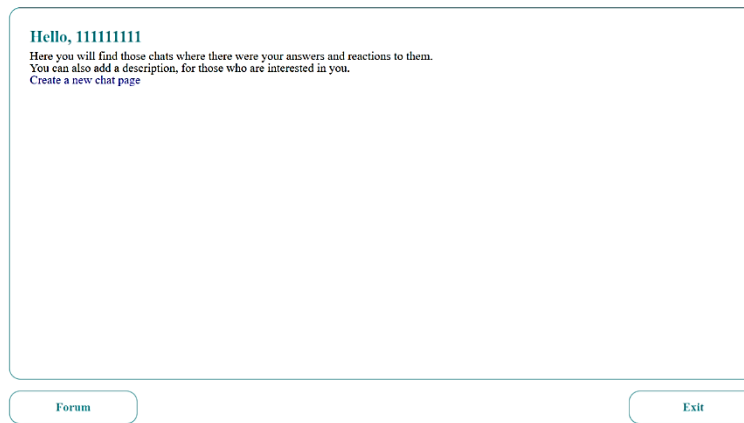


Рис. 3.4 – Сторінка користувача

Щоб забезпечити швидкий перехід до основних тематичних розділів сайту, розробляється меню сайту — список гіперпосилань на його основні сторінки\розділи. Зазвичай меню розташовано в “шапці” сайту, звідки користувач починає свій перегляд, гіперпосилання мають графічно зрозумілий вигляд, анімацію.



Рис. 3.5 – Схема переходу між сторінками

Для створення дизайну сторінок форуму можна використати такі програми\інструменти:

1. *Adobe Photoshop* (програма для роботи з графікою та дизайном, зазвичай використовують як редактор зображень, що в подальшому будуть представлені на сайті).
2. *Figma* (інструмент, що надає спільний робочий простір для дизайну веб сторінок).

					КРБ.КІ.1.442-03.3.11	Арк.
						39
Змн.	Арк.	№ докум.	Підпис	Дата		

3. *Sketch* (програма для швидкого дизайну веб сайту).
4. *Adobe Dreamweaver* (Редактор *HTML*-коду та візуальний редактор для створення веб сторінок).
5. *Canva* (інструмент для швидкого та спрощеного створення графічного контенту веб сторінки).
6. *Colour Contrast Checker* (інструмент для перевірки контрастності кольорів тексту з кольором фону).
7. *Google Fonts* (набір шрифтів) та інші.

Наразі існує дуже багато інструментів для створення дизайну та окрім програм варто зазначити штучний інтелект, що впевнено становиться частиною буденності людини та також може стати чудовим інструментом для створення інтерфейсу сторінки.

Приклади ескізів сторінок реєстрації, входу, перегляду підтем та відповідей:

1. Сторінка реєстрації:

- a. форма з полями для введення ім'я, логіну(є унікальним для кожного користувача), пароль та кнопкою;
- b. посилання на сторінку входу, якщо ти вже зареєстрованих користувач;
- c. повернення на головну сторінку;

2. Сторінка входу:

Рис. 3.6 – Ескіз сторінки входу та сторінки реєстрації

					КРБ.КІ.1.442-03.3.11	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		40

- a. форма з полями для введення логіну та пароля, кнопки;
- b. посилання на сторінку реєстрації;
- c. повернення на головну сторінку.

3.2 Розробка бекенду

3.2.1 Структура скриптов проекту

У проекті використовуються дані файли, що реалізують логіку та функціонал застосунку:

1. *index.php* – це головна сторінка, з якої починається взаємодія користувача с форумом. Окрім переходу, до інших сторінок сайту, реалізована перевірка активної сесії користувача. В залежності від перевірки користувачу буде запропоновано або пройти реєстрацію\вход на сайт, або карта користувача.
2. *log-in.php* – сторінка, реалізовує аутифікацію користувача на сайті, перевіряючи введений логін та пароль з збереженими у відповідній таблиці в БД.
3. *login.php* – сторінка, що пропонує користувачу реєстрацію на сайті. Введі значення паролю\логіну\та ім'я будуть збережені в БД. Введені дані перевіряються на довжину, а до пароля додається сіль, логін не має бути коротшим за 5 символів, пароль не має бути коротшим за 6, ім'я за 3.Також логін є ідентифікатором в таблиці БД, тому повторюватись у користувачів він не може. При цьому довжина активної сесії користувача на сайті триватиме добу, після чого потрібно буде повторно зробити вхід до застосунку. Даний скрипт відповідає за реєстрацію користувачів. Він перевіряє логін та пароль, передані через форму, та встановлює куки та сесії для авторизованих користувачів:

Лістинг 3.1 Реєстрація користувача

```
if(isset($_POST['login']) && isset($_POST['password'])) {
    $login = filter_var(trim($_POST['login']), FILTER_SANITIZE_STRING);
    $password = filter_var(trim($_POST['password']), FILTER_SANITIZE_STRING);
```

					КРБ.КІ.1.442-03.3.11	Арк.
						41
Змн.	Арк.	№ докум.	Підпис	Дата		

```

$hashed_password = md5($password."ty964");
$mysql = new mysqli('localhost', 'root', '', 'registration-web1');
$stmt = $mysql->prepare("SELECT * FROM `users` WHERE `login` = ?");
$stmt->bind_param('s', $login);
$stmt->execute();
$result = $stmt->get_result();
if($result->num_rows === 0) {
    echo "Користувач з даним логіном не знайдено.";
} else {
    $user = $result->fetch_assoc();
    if($user['password'] === $hashed_password) {
        setcookie('user', $login, time() + 3600*24, "/");
        session_start();
        $_SESSION['user_id'] = $user['id'];
        header('Location: /html/card-user.php');
    } else {
        echo "Не правильний пароль.";} }
$stmt->close();
$mysql->close();
} else {
    echo "Необхідно передати логін і пароль";
}

```

4. *exit.php* – реалізація завершення сесії користувача на сайті.

5. *card-user.php* – картка користувача, де відображається опис, ім'я, можливість додавання опису.

6. *branch-sub-theme.php* – сторінка з обговоренням. Користувачі можуть залишати свої відповіді\коментарі, кожен коментар має дату та логін користувача, що є гіперпосиланням на картку користувача.

					КРБ.КІ.1.442-03.3.11	Арк.
						42
Змн.	Арк.	№ докум.	Підпис	Дата		

7. *create-page.php* – створення нового публічного чи приватного чату спілкування.

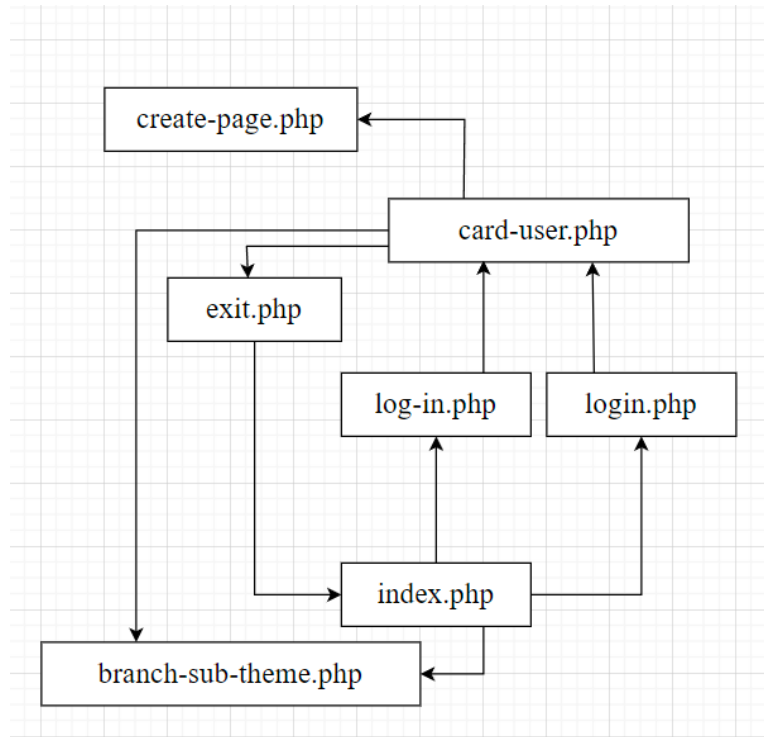


Рис. 3.7 – Схема зв'язків переходів між сторінками

3.2.2 Використання *POST\GET* запитів

Get-запит – використовується для видачі сервером інформації клієнту.

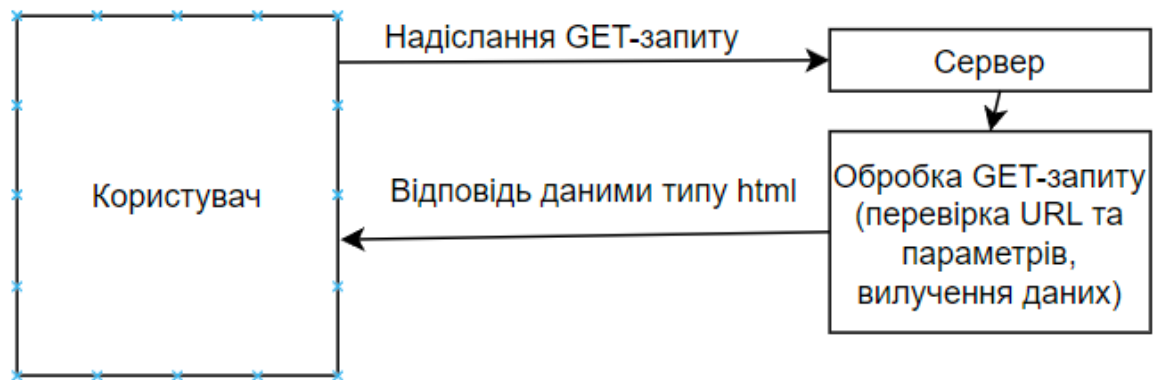


Рис. 3.8 – GET запит

коментарі зберігаються та передаються через змінні: *\$comment*(текст повідомлення), *\$date*(час написання повідомлення), *\$user_login*(логін користувача, що залишив повідомлення). *GET*-запит використовується для отримання сторінки та відображення коментарів, що вже є збереженими у БД.

3.2.3 Розробка та реалізація схеми бази даних

Схема бази даних служить основним архітектурним блоком, що визначає логічну структуру. При розробці це грає важливу роль у забезпеченні організації\збереження інформації форуму. Є різниця між схемою БД та структурою фізичного сховища. Бази даних описують логічну структуру даних. Вона дає уявлення про структури даних і їх взаємозв'язках.

Компоненти баз даних:

1. Таблиці (основна структура у БД, інформація організована у вигляді рядків та стовпців).
2. Поля (визначають тип даних, що зберігається в стовпці таблиці);
3. Ключі (унікальний ідентифікатор, що використовується та встановлення зв'язків).
4. Зв'язки (відносини між різними таблицями БД).
5. Обмеження (використовуються для цілісності даних та визначають права доступу до них).

Під час проектування БД є декілька кроків за якими вони будується. Схеми БД:

1. Логічна схема. Визначення структури та відносин в БД, які дані зберігаються в БД та як вони взаємопов'язані. Використовується на етапі проектування БД для створення загального уявлення про дані та їх взаємозв'язки.
2. Фізична схема. Визначення способу зберігання інформації на фізичному рівні, використовується для оптимізації продуктивності бази даних.

					<i>КРБ.КІ.1.442-03.3.11</i>	<i>Арк.</i>
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		45

3. Схема представлення Визначення рівня доступу до даних, зовнішня схема або рівень представлення, як ці дані відобразатимуться конкретному користувачеві, використовується для створення користувацьких інтерфейсів, відображення даних у зручному для кінцевого користувача форматі.

Можливі схеми таблиць баз даних для веб проекту:

- всі дані зберігаються в одній таблиці. У реалізації ця схема проста, але має недолік - може призвести до дублювання даних та ускладнення звернень до бази даних;
- дані розподіляються між декількома таблицями і пов'язані за допомогою зовнішніх ключів. Наприклад, окремі таблиці для користувачів, повідомлень, тем тощо. Дана схема дозволяє уникнути дублювання даних і забезпечує більш логічну структуру БД.

Різниця між цими схемами полягає у оптимізації швидкодії та впливає на структуру бази даних. Схема, де всі дані в одній таблиці, проста у реалізації, але може призвести до проблем. Розподілення інформації по різних таблицях за змістом забезпечує кращу структуру, але є складніша у проектуванні. Добре продумана схема бази даних є основою успішного проекту. Вона забезпечує: ефективність, надійність, масштабованість, безпеку, гнучкість

Проектування бази даних – це є процесом створення логічних та фізичних структур БД для задоволення потреб користувачів та ТЗ проекту. Це включає в свою чергу призначення сутностей, нормалізацію, створення схем. бази даних. Нормалізація - це розбиття даних на логічно пов'язані таблиці. Призначення сутностей(елементів таблиці БД) є першим кроком у розробці БД і включає в себе визначення ключових елементів, в яких повинна зберігатися інформація.

Робота з сутностями складається з:

1. Ідентифікація об'єкта (ідентифікація всіх об'єктів, представлених в базі даних).
2. Визначення атрибута (кожен об'єкт має властивість\атрибут).

					<i>КРБ.КІ.1.442-03.3.11</i>	Арк.
						46
Змн.	Арк.	№ докум.	Підпис	Дата		

3. Взаємозв'язок між об'єктами (визначає, як об'єкти пов'язані один з одним).

Схема бази даних представляє собою абстрактну структуру бази даних, що включає таблиці, поля, зв'язки.

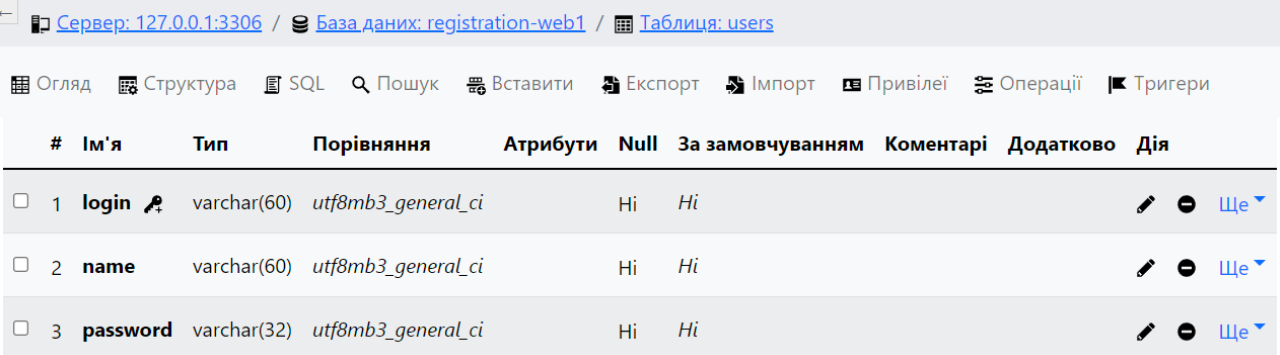
Схема бази даних для форуму може мати наступні таблиці та їх вміст:

1. Таблиця *users*(користувачі):

a. *name*;

b. *login* (унікальне значення для кожного користувача, виступає ідентифікатором);

c. *password* (пароль+"сіть").



#	Ім'я	Тип	Порівняння	Атрибути	Null	За замовчуванням	Коментарі	Додатково	Дія
1	login	varchar(60)	utf8mb3_general_ci		Ні	Ні			Ще
2	name	varchar(60)	utf8mb3_general_ci		Ні	Ні			Ще
3	password	varchar(32)	utf8mb3_general_ci		Ні	Ні			Ще

Рис. 3.10 – Таблиця users

2. Таблиця "Теми":

a. *topic_id* (первинний ключ);

b. *topic_title*;

c. *section_id* (зовнішній ключ, посилається на таблицю "Розділи");

d. *user_id* (зовнішній ключ, посилається на таблицю user);

e. *date*.

3. Таблиця *chat* (повідомлення):

a. *id* (первинний ключ);

b. *data*;

c. *user_id* (зовнішній ключ, посилається на таблицю user);

d. comment.

#	Ім'я	Тип	Порівняння	Атрибути	Null	За замовчуванням	Коментарі	Додатково	Дія
<input type="checkbox"/>	1	id			Ні	Hi		AUTO_INCREMENT	Ще
<input type="checkbox"/>	2	date			Ні	Hi			Ще
<input type="checkbox"/>	3	comment	text	utf8mb4_0900_ai_ci	Ні	Hi			Ще
<input type="checkbox"/>	4	user_id	int		Так	NULL			Ще

Рис. 3.11 – Таблиця chat

До кожного стовпця БД розписується налаштування полів, його тип даних, метод кодування, чи дозволено записувати пусте значення, чи можуть ці значення повторюватися. Дані налаштування полів впливають на взаємодію з даними в БД. Наприклад, якщо це поле, що зберігає вміст коментаря, то налаштовується його тип даних – текст та кількість можливо введених символів, ставимо, що дане поле не може мати пусте (бути не заповненим) значення. Це загальна схема БД та таблиць форуму, яка може існувати, але не є правилом. У проєкті можуть бути додаткові таблиці, інші поля для зберігання додаткової інформації, такої як вподобайки до повідомлень, відповіді на коментарі тощо.

Також варто зазначити низьку критерій оформлення БД, а саме створення взаємодій між таблицями таким чином:

1. Відношення один до одного (з обов'язковим або з необов'язковим зв'язком).
2. Відношення один до багатьох (з обов'язковим або з необов'язковим зв'язком).
3. Відношення багато до одного.
4. Відношення багато до багатьох.

Застосування цих кількох критерій проєктування баз даних будуть доречними для створення ефективної структури БД. Приклади застосувати даних правил з роботою з *MySQL* через *phpMyAdmin*:

1. Створення таблиць. В *phpMyAdmin* є можливість створювати таблиці, визначаючи поля та будуючи зв'язки між ними. При створенні таблиці необхідно враховувати взаємозв'язки між собою, відносини "багато до багатьох", в залежності від конкретних вимог.
2. Встановлення зовнішніх ключів. В *phpMyAdmin* є можливість для встановлення зовнішніх ключів для забезпечення цілісності даних і обміну даними між таблицями. Це дозволяє підтримувати зв'язок між об'єктами відповідно до формальних правил.
3. Нормалізація даних. Застосування нормалізації даних допоможе уникнути надмірності та неоднозначності даних. Розділяйте інформацію на окремі таблиці, щоб покращити продуктивність та забезпечити легкість у обробці даних.
4. Денормалізація.
5. Використання інструментів моделювання. У *phpMyAdmin* можна використовувати інструменти моделювання для візуалізації структури бази даних і відносин між таблицями.

Опис процесу створення структури бази даних:

1. Виберіть сутності: визначте основні сутності, які представляють різні частини системи. В даному випадку це користувачі, сторінки та чати.
2. Визначення полів: визначте необхідні поля для кожної сутності, включаючи ключі та атрибути.
3. Визначення зв'язків: визначте, як сутності взаємодіють, і встановіть відповідні зв'язки між ними.

					КРБ.КІ.1.442-03.3.11	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		49

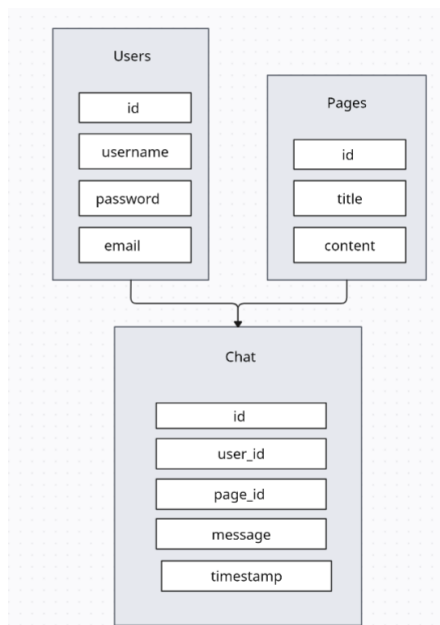


Рис. 3.12 – Структура бази даних

3.3 Вибір технологій для реалізації

Для доступу до даних, представлення інтерфейсу користувача та захисту інформації на форумі є можливість використовувати такі технології:

1. Для доступу до даних можна використовувати:
 - a. *SQL* (*SQL* – мова, що використовується для створення, внесення змін, керування даними баз даних та інше);
 - b. *phpMyAdmin* (*PhpMyAdmin* – інструмент, завдяки котрому можна виконувати різні операції\дії з базами даних *MySQL*, без необхідності прописувати *SQL*-запити власноруч, а за допомогою графічного інтерфейсу).
2. Для інтерфейсу, що представлений користувачу, використовують *HTML* (мова розмітки гіпертексту), *CSS* (каскадні таблиці стилів), *JavaScript* (динамічна, об'єктно-орієнтована мова програмування.).
3. Для створення бізнес-логіки:
 - a. мови програмування (наприклад: *Java*, *Python*, *PHP*);
 - b. фреймворки.
4. Для захисту інформації є такі інструменти\методи:

MySQL — це діалогова мова програмування для здійснення запитів і внесення змін та керування базами даних. *PostgreSQL* — об'єктно-реляційна система керування базами даних, альтернатива *MySQL*. На форумах, де переважають прості операції читання, *MySQL* може бути найкращим вибором через високу швидкість читання та невелику затримку. *PostgreSQL* виконує більш складні запити, але при великому навантаженні на запис він може бути надмірно повільним. Якщо брати до уваги швидку установку та простоту управління БД, *MySQL* буде кращим вибором завдяки своїй простоті. *PostgreSQL* є більш гнучким у налаштуванні, але може виявитися складнішим для початківців розробників та менших команд.

Переваги використання *PHP* та *SQL* для створення веб проекту:

- Головна особливість *PHP* у інтерактивності мови та гнучкості, її можна використовувати в документах, які створені в *HTML* форматі;
- *PHP* є швидкою мовою програмування, котра в порівнянні з іншими легше завантажується через ненад-то швидкий інтернет;
- *PHP* легко підключається до бази даних, забезпечуючи надійне з'єднання;
- *PHP* підтримується багатьма операційними системами, наприклад, такими як Windows, Unix, Linux;
- *PHP* є однією з найпопулярніших мов для веб розробки. Розроблені на основі *PHP* веб програми можна легко запустити на будь-якій платформі. Вона підтримується більшістю хостинг-провайдерів, має велику кількість розробників, що спрощує пошук потрібної інформації для створення проекту;
- *PHP* є легкою для вивчення мовою програмування, що дозволяє швидко створювати та змінювати форум;
- вбудовані функції та інструменти *PHP* полегшують захист веб застосунків від зовнішніх загроз безпеці;

					<i>КРБ.КІ.1.442-03.3.11</i>	Арк.
						52
Змн.	Арк.	№ докум.	Підпис	Дата		

- за допомогою *PHP* є можливість створювати різноманітні функції та модулі для форуму;
- *SQL* дозволяє ефективно працювати з БД та виконувати складні запити;
- *PHP* та *SQL* легко інтегруються з іншими технологіями, що дозволяє розширювати функціонал форум, використовуючи різноманітні розширення.

Використання розглянутих технологій для створення форуму має багато переваг.

Недоліки використання *PHP* та *SQL* для створення форуму:

- *PHP* має вразливістю до атак, таких як *SQL* ін'єкції. Неправильна конфігурація *SQL* запитів може призвести до витіку конфіденційної інформації користувачів;
- *PHP* може бути повільним для обробки великого обсягу даних, що безпосередньо впливає на роботу форуму;
- при збільшенні обсягу користувачів, а з ними і даних, можуть виникнути проблема з масштабуванням. Потрібно виконувати оптимізації та використання кешування для забезпечення ефективної роботи форуму.
- інші мови програмування\технології можуть мати більшу кількість готових рішень та бібліотек для створення форуму, що має можливість спростити розробку;
- *PHP* та *SQL*, обмежуються у деяких аспектах функціональності, наприклад, для роботи з великим обсягом інформації або для застосування в проекті розширених алгоритмів шифрування.

PHP та *SQL* для створення форуму мають недоліки, які при роботі потрібно враховувати.

					<i>КРБ.КІ.1.442-03.3.11</i>	<i>Арк.</i>
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		53

Лістинг 3.2 Додавання коментаря на форум

```
<?php
session_start();
$mysql = new mysqli('localhost', 'root', '', 'registration-web1');
if ($mysql->connect_error) {
    die("Помилка з'єднання: " . $mysql->connect_error);}
if (isset($_POST['comment'])) {
    $comment = $_POST['comment'];
    $date = date("Y-m-d H:i:s");
    $user_login = isset($_SESSION['user_login']) ? $_SESSION['user_login'] : null;
    if ($user_login) {
        // Підготовка SQL-запиту для вставки коментарів
        $stmt = $mysql->prepare("INSERT INTO chat (comment, date, user_id) SELECT ?, ?, login FROM
users WHERE login = ?");
        if (!$stmt) {
            echo "Помилка підготовки запиту: " . $mysql->error;
        } else {
            // Прив'язка параметрів та виконання запиту
            $stmt->bind_param("sss", $comment, $date, $user_login);
            if (!$stmt->execute()) {
                echo "Помилка виконання запиту: " . $stmt->error;} else {
                header("Location: " . $_SERVER['PHP_SELF']);
                exit();}
            $stmt->close();}} else {
        echo "Користувач не авторизований.";}}
$mysql->close();
?>
```

5. Регулярні аудити безпеки.

3.5 Збірка проекту.

Під час розробки веб проекту, будь то сайт чи застосунок, потрібен інструмент для перегляду проекту та тестування. Розробник повинен розуміти, як виглядає макет на працюючому сайті, як працюють різні функції і скрипти, такі як форми, кошики покупок, фільтри і т.д. але купувати для цього місце на хостингу недоцільно.

					КРБ.КІ.1.442-03.3.11	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		55

На початковому етапі розробки веб застосунку використовується локальний сервер. Це дає кілька вагомих переваг:

1. Використання локального сервера дозволяє працювати з проектом без завантаження коду на сервер, а саме швидко вносити зміни і проводити тести без затримок, пов'язаних з віддаленим доступом до сервера. Також це економить грошовий ресурс.
2. При користуванні локального серверу можливість зливу всієї конфіденційної інформації менша, що є пріоритетом.
3. Локальний сервер надає можливість тестувати і налагоджувати веб застосунок, не зачіпаючи діючий сервер. Це дозволяє виявляти та виправляти помилки до того, як програмне забезпечення буде запущено у виробництво і коректно виправляти помилки при вже запущеному застосунку, не роблячи нових помилок і неруйнуючи вже працюючі скрипти.

Використання локального сервера дозволяє ефективно працювати над проектом, забезпечуючи гнучкість роботи. *Open Server* використовується для роботи над проектами і користується популярністю серед розробників, але існує ще кілька можливих варіантів локальних серверів і їх переваг:

1. *Open Server* - це безкоштовний локальний сервер для *Windows*, який включає в себе веб сервер *Apache*, базу даних *MySQL* та *PHP*. Оснащена інтегрованими інструментами для розробки веб проектів та їх тестування. Має простий інтерфейс. *Open Server* підтримує роботу з *PHP*, *Perl*, *Python* та іншими мовами програмування.
2. *XAMPP* - безкоштовна багатоплатформова збірка вебсервера з відкритим початковим кодом, що містить в собі *Apache*, *MySQL*, *PHP* та *Perl*. Він доступний для *Windows*, *macOS* та *Linux* і має широкі можливості для розробки веб додатків. Але є більш повільнішим ніж *Open Server*.

					КРБ.КІ.1.442-03.3.11	Арк.
						56
Змн.	Арк.	№ докум.	Підпис	Дата		

3. WampServer має у збиранні всі необхідні компоненти: *MySQL* (а також модуль *MariaDB*), *PHP*, *Apache* та додаткові інструменти, такі як *phpmyadmin*.

Переваги *OpenServer*:

- приємний та доброзичливий інтерфейс;
- простий у встановленні;
- безкоштовний;
- портативний (можна запускати навіть із флешки);
- крім *Apache* має у складі модуль *Nginx*;
- пропонує на вибір кілька видів СУБД (зокрема *NoSQL*) та інших компонентів.

До недоліків *OpenServer* можна віднести відсутність підтримки 32-бітної систем.

Переваги *XAMPP*:

- зручний та компактний інтерфейс;
- кросплатформність (можна встановити на *Windows*, *Linux* та *MacOS*);
- багатозадачний;
- швидке керування сервісними службами;
- безкоштовний.

Переваги *WampServer*:

- безкоштовний;
- простий у встановленні;
- наявність 64-розрядної та 32-розрядної версій;
- в архіві є старі версії (наприклад, для *Windows XP*);
- є активна діюча спільнота, в якій можна отримати допомогу;

Недоліком *WampServer* є складніші налаштування в порівнянні з аналогами. Переваги *Open Server* порівняно з іншими аналогічними серверами - це простота налаштування, швидкість роботи та доступність для розробників на *Windows*. *Open Server* також має широкі можливості для роботи з різними мовами програмування та базами даних.

Для запуску веб застосунку в інтернет потрібно проробити наступні кроки:

					КРБ.КІ.1.442-03.3.11	Арк.
						57
Змн.	Арк.	№ докум.	Підпис	Дата		

1. Пошук хостинга. Хостинг може бути декількох видів:
 - a. віртуальний хостинг (дешевий і простий в налаштуванні, обмежені ресурси);
 - b. віртуальний виділений сервер (потужний, але вимагає більше знань);
 - c. виділений сервер (максимальний контроль та продуктивність, але дорогий варіант);
 - d. хмарний хостинг (надійний варіант, що часто використовується для великих проектів).
2. Вибір доменного імені, реєстрація ім'я через реєстратор доменів.
3. Налаштування застосунку. В даному етапі кроки наступні:
 - a. вибір і встановлення платформи форуму;
 - b. налаштуйте бази даних для зберігання даних;
 - c. завантаження файлів на сервер;
 - d. налаштуйте базові параметри (ім'я сайту, мова, адміністраторський обліковий запис тощо).
4. Перевірка\налаштування безпеки.

3.6 Тестування та відлагодження форуму

Тестування та відлагодження форуму - це ключова частина у процесі розробки проекту, що передбачає проведення різних видів тестів для перевірки працездатності функцій форуму та виявлення помилок.

Види тестів:

1. Функціональне тестування (перевірка роботи окремих функцій сайту, що повинні виконувати певні завдання). Перевірка роботи основних функцій форуму, таких як створення тем\гілок спілкування, додавання відповідей, реєстрація користувачів, вхід в систему і т.д. Наприклад, реєстрація нового користувача, при внесенні даних у форму має відкриватися вікно картка користувача, дані мають бути збережені в БД.:

					<i>КРБ.КІ.1.442-03.3.11</i>	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		58

Create Account

We are not interested in your e-mail, so if you forget your password, we will not help you recover it

[return](#)

Testing

Test

.....

Register

You no longer exist on the not-wiki?

[Log in](#)

←T→

login

name

password

		login	name	password
<input type="checkbox"/>		111111	111111	39b8cc4481d76b307b5394607048e3c3
<input type="checkbox"/>		Qwerty	Qwerty	381ad4adadf02f43ba1599f35d0854ee
<input type="checkbox"/>		111111111	111111111	98a05ec3e7b1178427c03adb6aaf96bf
<input type="checkbox"/>		Testing	Test	b6b4c3817621d3c0bb8be6184ec8d94d

Hello, Testing

Here you will find those chats where there were your answers and reactions to them. You can also add a description, for those who are interested in you. Create a new chat page

Forum

Exit

Рис. 3.14 – Результат функціонального тестування

2. Юзабіліті тести (оцінка того, як легко користувачі можуть взаємодіяти з елементами сайту, наскільки зручно і практично використовувати ваш сайт).

3. Тестування безпеки. Перевірка форуму на вразливості до атак злому, витоку даних, несанкціонованого доступу.

Тестування забезпечує якість сайту, відсутність помилок в функціональній частині та передбачення відмінностей між браузерами (різні браузери можуть відображати веб сторінки по-різному).

Етапи тестування:

1. Відповідність вимогам. (перевірка, що форум відповідає вимогам, визначеним на етапі технічного завдання. Включно з перевірку функціональності, дизайну, швидкодії та інших параметрів).
2. Відлагодження помилок. (якщо під час тестування виявлені помилки, їх необхідно виправити шляхом відлагодження програмного

										Арк.
										59
Змн.	Арк.	№ докум.	Підпис	Дата						

КРБ.КІ.1.442-03.3.11

забезпечення, а саме виправлення коду, зміни в базі даних чи інші можливі дії для виправлення помилок).

3. Повторні тести. (після внесення змін необхідно повторно протестувати форум, щоб переконатися, що помилки були виправлені та програмне забезпечення працює належним чином без нових викривлень).

Тестування та налагодження є невід'ємною частиною процесу розробки веб сайту\веб застосунку допомагає забезпечити його якість та надійність перед запуском, а також виявити та виправити можливі проблеми. Це можете гарантувати якість, стабільність і продуктивність проекту.

Висновки до третього розділу

Реалізація проекту залежить від вибору технології, комплексного підходу до розробки та впровадження всіх компонентів системи. А також від вибору технології до тестування. Всі етапи є важливими і впливають на кінцевий результат. Інструментарій розробника відіграє вкрай важливу роль, від побудови сторінок, написанню кода до створення грамотної логіки в базах даних. Краще спиратися саме на знайомі інструменти\воми програмування\бібліотеки, або ті, що мають гарну документацією з активним ком'юніті, тоді розробка буде мати кращий вигляд та більш продуману структуру.

					<i>КРБ.КІ.1.442-03.3.11</i>	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		60

РОЗДІЛ 4

ТЕХНІКО-ЕКОНОМІЧНА ЧАСТИНА

4.1 Організаційно-економічне й маркетингове обґрунтування проекту

Впровадження та ефективне використання інформаційних систем та сучасних технологій на форумі може стати перевагою для комерційних проектів, великих та малих підприємств як зворотній зв'язок, оскільки забезпечує інтеграцію з зовнішнім середовищем, сприяє оперативному збору та систематизації інформації. Для створення програмного продукту було проаналізовано основні проблеми в предметній області, проаналізовано існуючі аналогічні продукти, проаналізовано та обґрунтовано вибір засобів реалізації, досліджено моделі та методи проектування. У якості середовища розробки обрано Visual Studio Code, в якості мови програмування – php, javascript, та інструменти верстки – html, css.

4.1.1 Порівняльний техніко-економічний аналіз

Доцільність впровадження проекту - форуму:

1. Економічна доцільність. Даний проект має як витрану сторону(витрати на розробку, підтримку, хостінг) так і можливу прибуткову (монетизація сторінки через платний контент, преміум-послуги, вимкнення реклами, відсоток з продажу товарів через сторінку, рекламу).
2. Соціальна доцільність. Форум дозволяє створювати спільноти однодумців з спільними інтересами, та є патформуою для обміну досвідом між користувачами.
3. Технічна доцільність. Потрібно забезпечити зручний інтерфейс, повний функціонал сайту, яка б допомогла користувачам знаходити потрібну інформацію та взаємодіяти один з одним. Також важливо приділити велику увагу безпеці персональних даних користувачів, особливо, вибору надійного хостингу так само як і оптимізації та

					КРБ.КІ.1.442-03.3.11	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		61

адаптації веб застосунку. Для більшої кількості відвідувачів веб застосунок має мати адаптацію під різні пристрої, це актуально як ніколи, бо більшість Інтернет трафіку споживається саме через мобільні пристрої.

4.1.2. Організаційне обґрунтування проекту

Класифікаційна оцінка проекту:

- клас: мультипроект;
- тип: комерційний;
- вид: комбінований;
- тривалість: середньостроковий;
- за ступенем складності: середня складність та середні розміри;
- рівень: індивідуальний.

Метою даної роботи є покращення користувацького досвіду, шляхом використання нових функцій.

Предметом дослідження є впровадження інноваційних на підходів до форумів та їх вплив на активність користувачів.

Об'єктом дослідження є сам форум як майданчик для спілкування та обміну інформацією.

Результат – розроблений веб додаток.

Етапи виконання розділів дипломної роботи:

- постановка технічного завдання (аналіз існуючих форумів,) (термін: 25 днів);
- техніко-економічна частина (термін: 14 днів);
- проектна частина (термін: 7 днів);
- практична реалізація (термін: 7 днів);

Етапи виконання розділів проекту дипломної роботи:

- технічне завдання (термін: 10-12 днів)

					<i>КРБ.КІ.1.442-03.3.11</i>	Арк.
						62
Змн.	Арк.	№ докум.	Підпис	Дата		

- розробка алгоритму програмного продукту (термін: 27-29 днів);
- розробка робочого проекту (розробка дизайну, верстка сторінок форуму за створеним макетом, створення баз даних та функціональних частин веб застосунку) (термін: 30-35 днів);
- тестування проекту, виправлення помилок (термін: 10-14 днів).

Таблиця 4.1

Життєвий цикл проекту

№ код роботи	Назва роботи	T, дні
1-2	Збір та аналіз даних, існуючих моделей	10
2-3	Встановлення потреби в результатах	9
2-4	Затвердження концепцій	3
3-5	Визначення цілей, мотивів та вимог замовника та власника	11
4-5	Планування предметної області та інших елементів проекту	6
5-6	Розробка та затвердження зведеного плану	7
5-7	Організація виконання робіт	9
6-8	Інформаційний контроль за виконанням робіт	6
7-8	Детальне моделювання	15
8-9	Керівництво і координація робіт, корегування основних показників проекту	30
9-10	Підтвердження закінчення робіт	3
9-11	Експлуатаційні випробування остаточного продукту проекту	5
10-12	Підготовка звітів	7
11-12	Підготовка кадрів для експлуатації програмного продукту	2
12-13	Оцінка результатів проекту	2
13-14	Підготовка керівництва з користування програмного продукту	3

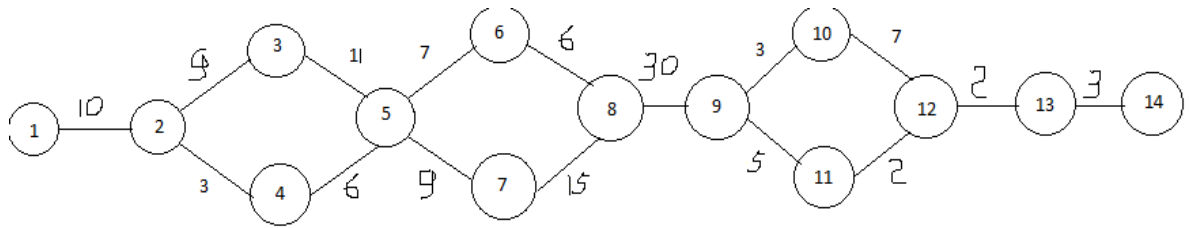


Рисунок 4.1 – Мережевий графік проекту

Надалі проведемо розрахунок параметрів мережевого графіку. Для цього визначимо: ранній строк здійснення роботи (T_i); тривалість роботи (T_{ij}); ранній строк здійснення події (T_i); раннє закінчення робіт (T_{po}), пізній строк здійснення події (T_j); пізнє закінчення робіт (T_{no}), повний резерв часу роботи (R_c); вільний резерв часу роботи (R_c).

Таблиця 4.2

Розрахунок параметрів мережного графік

Попередня робота	Фактична робота	T_{ij}	T_{pi}	T_{po}	T_{ni}	T_{no}	R_c	R_n	R_j
1	2	10	0	10	0	10	0	0	0
2	3	9	10	19	10	19	0	0	0
2	4	3	10	13	10	13	19	19	19
3	5	11	19	30	19	30	0	0	0
4	5	6	13	19	13	19	20	20	20
5	6	7	30	37	30	37	13	13	13
5	7	9	30	39	30	39	0	0	0
6	8	6	37	43	37	43	13	13	13
7	8	15	39	54	39	54	0	1	0
8	9	30	54	84	54	84	0	0	0
9	10	3	84	87	84	87	7	7	7
9	11	5	84	89	84	89	0	0	0
10	12	7	87	94	87	94	7	7	7
11	12	2	89	91	89	91	0	0	0
12	13	2	91	93	91	93	0	0	0
13	14	3	93	96	93	96	0	0	0

Тривалість критичного шляху 96 днів, не перевищує часу на проектування (96 днів), тому оптимізувати мережеву модель немає необхідності.

4.2 Економічні розрахунки проекту

4.2.1 Бізнес план проекту

Ціль проекту - створення онлайн платформи, де користувачі спілкуються та обговорюють різні теми, шукаючи співрозмовників за інтересами, де була б можлива монетизація.

Можливий функціонал, через який буде впроваджена монетизація:

1. Платний(розширений) функціонал (створення гілок спілкування без обмеження, розширені реакції на повідомлення).
2. Відсоток від продажу товарів на сайті.
3. Розміщення реклами.

Опис послуг:

1. Базовий функціонал:

- доступ до всіх публічних гілок розмов;
- можливість коментарів;
- можливість залишати приватне повідомлення для будь-якого користувача;
- створення певної кількості гілок розмов на місяць.

2. Платний функціонал:

- розширені реакції на повідомлення;
- необмежена кількість створення гілок розмов;
- апгрейд зовнішнього виду профілю;
- більша кількість аватарок;
- можливість перемкнути на темну тему форуму.
- розміщення об'яв про продаж товару за тематикою форуму;
- розміщення реклами на певних категоріях.

					<i>КРБ.КІ.1.442-03.3.11</i>	Арк.
						65
Змн.	Арк.	№ докум.	Підпис	Дата		

Цільова аудиторія:

1. Користувачі, що шукають експертної думки та в пошуку спілкування за певними темами.
2. Малі та середні підприємці, які хочуть продавати свої товари через спеціалізовані форуми.
3. Рекламодавці, які бажають досягти цільової аудиторії через тематичні форуми.

Конкуренцію можуть складати інші тематичні форуми, інтернет магазини, інтернет платформи для продажу вживаних речей або ж групи у соціальних мережах. Важливим питанням є план впровадження проекту. Для впровадження та популяризації є можливість використовувати SEO, реклама у соціальних мережах. Щоб проект не втрачав активну аудиторію потрібно підтримка та розвиток, а саме регулярні активності\ конкурси чи спеціальні заходи, наприклад вебінари для користувачів.

Можливі джерела доходу:

- продаж преміум акаунтів (вартість без реклами: \$1 на місяць, прогнозований % користувачів, що купуватимуть: 10%);
- комісія з продажу товарів (середній відсоток комісії: 3% від кожної транзакції.)
- розміщення реклами: (вартість рекламного місця (банери, контекстна реклама): \$100 на місяць).

Можливі витрати:

- розробка та підтримка сайту: \$1000;
- маркетинг та реклама: \$600 на місяць;
- адміністративні витрати (зарплати, хостинг, технічна підтримка): \$300 на місяць;

Ризики та способи їх мінімізації. Ризики:

1. Низька залученість користувачів.
2. Висока конкуренція.

					КРБ.КІ.1.442-03.3.11	Арк.
						66
Змн.	Арк.	№ докум.	Підпис	Дата		

3. Технічні проблеми та збої.

Стратегії мінімізації:

1. Регулярний аналіз та поліпшення UX/UI.
2. Активна маркетингова стратегія та підтримка спільноти.
3. Резервний план на випадок технічних збоїв та кіберзагроз.

4.2.2 Розрахунки економічної ефективності

Тривалість розробки ПП залежить від обсягу ПП, трудомісткості його розробки, кваліфікації виконавців, а також планових строків, які диктуються умовами ринку. У даному проекті розробляється новий програмний продукт, який відповідає аналогу ПЗ оптимізаційних розрахунків с $V_0 = 3000$ умовних машинних команд із трудомісткістю $T_p = 262$ чол/год.

Трудомісткість розробки ПП повинна включати розробку наступних етапів:

- технічного завдання(ТЗ);
- технічного проекту (ТП);
- робочого проекту (РП);
- впровадження (ВН).

Трудомісткість розроблювального ІС визначається на кожному етапі окремо на підставі трудомісткості аналога, з урахуванням складності розробки, ступеня новизни та ступеня використання в розробці стандартних модулів на підставі формул (4.1-4.4):

$$T_{ТЗ} = T_p * L1 * K_n \quad (4.1)$$

$$T_{ТП} = T_p * L2 * K_n \quad (4.2)$$

$$T_{РП} = T_p * L3 * K_n * K_t \quad (4.3)$$

$$T_{ВН} = T_p * L4 * K_n \quad (4.4)$$

T_p – укрупнення норма часу на розробку аналога ПЗ, чол/год, що коректується поправочним коефіцієнтом, що враховує умови розробки ПЗ, тобто в умовах комп'ютера, $K_n=0,7$.

У даному проекті $T_p=262*0,7=183$ чол/год

					<i>КРБ.КІ.1.442-03.3.11</i>	Арк.
						67
Змн.	Арк.	№ докум.	Підпис	Дата		

L_j – питома вага i -го етапу розробки. У даному проекті залежно від ступеня новизни проекту (B): $L_1 = 0,12$; $L_2 = 0,11$; $L_3 = 0,58$; $L_4 = 0,15$.

K_n – поправочний коефіцієнт, що враховує ступінь новизни, у нашому випадку $K_n = 0,7$.

K_t – поправочний коефіцієнт, що враховує ступінь використання в розробці типових програм, $K_t = 0,9$.

При розрахунках прийняті наступні об'єми розробленої документації по етапах проекту:

$N_{ТЗ} = 8$ – кількість сторінок технічного завдання;

$N_{ТП} = 30$ – кількість сторінок технічного проекту;

$N_{рп} = 55$ – кількість сторінок робочого проекту;

$N_{інстр} = 10$ – кількість сторінок інструкції по налагодженню та впровадженню;

$N_{пр} = 85$ – кількість сторінок пояснювальної записки.

Розрахунок трудомісткості розробки ІС:

1. Технічне завдання

$$T_{тз} = T_p * L_1 * K_n = 183 * 0,12 * 0,7 = 15,38 \text{ чол/год}$$

$$T_{кк} = 0,7 * N_{тз} = 0,7 * 8 = 5,6 \text{ чол/год}$$

$$T_{нк} = 0,15 * N_{тз} = 1,2 \text{ чол/год}$$

2. Розробка технічного проекту (алгоритму й блок-схеми)

$$T_{тз} = T_p * L_2 * K_n = 183 * 0,11 * 0,7 = 14,09 \text{ чол/год}$$

$$T_{кк} = 0,7 * N_{тп} = 0,7 * 30 = 21 \text{ чол/год}$$

$$T_{нк} = 0,15 * N_{тп} = 0,15 * 30 = 4,5 \text{ чол/год}$$

3. Розробка робочого процесу

$$T_{тз} = T_p * L_3 * K_n * K_t = 183 * 0,58 * 0,7 * 0,9 = 66,87 \text{ чол/год}$$

$$T_{кк} = 0,7 * N_{рп} = 0,7 * 55 = 38,5 \text{ чол/год}$$

$$T_{нк} = 0,15 * N_{рп} = 0,15 * 55 = 8,25 \text{ чол/год}$$

4. Налагодження і впровадження

$$T_{тз} = T_p * L_4 * K_n = 183 * 0,15 * 0,7 = 19,22 \text{ чол/год}$$

$$T_{кк} = 0,7 * N_{інст} = 0,7 * 10 = 7 \text{ чол/год}$$

					<i>КРБ.КІ.1.442-03.3.11</i>	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		68

$$T_{нк} = 0,15 * N_{инст} = 0,15 * 10 = 1,5 \text{ чел/год}$$

5. Пояснювальна записка

$$T_{тз} = 1,5 * N_{пз} = 1,5 * 85 = 127,5 \text{ чел/год}$$

$$T_{кк} = 0,7 * N_{пз} = 0,7 * 85 = 59,5 \text{ чел/год}$$

$$T_{нк} = 0,15 * N_{пз} = 0,15 * 85 = 12,75 \text{ чел/год}$$

Всього:

$$T_p = 15,38 + 14,09 + 66,87 + 19,22 + 127,5 = 243,06 \text{ чел/год}$$

$$T_{кк} = 5,6 + 21 + 38,5 + 7 + 59,5 = 131,6 \text{ чел/год}$$

$$T_{нк} = 1,2 + 4,5 + 8,25 + 1,5 + 12,75 = 28,2 \text{ чел/год}$$

$$T_{общ} = 243,06 + 131,6 + 28,2 = 402,86 \text{ чел/год}$$

Тривалість розробки ІС у днях визначається за формулою (4.5):

$$T_{III} = \frac{\sum T_{ij}}{8 * 0,73} \quad (4.5)$$

де $\sum T_{ij}$ – сумарна тривалість розробки, год;

8 – тривалість робочого дня (коефіцієнт переказу в робочі дні), ч;

0,73 – коефіцієнт переведення в календарні дні;

T_{ij} – трудомісткість j-го виду робіт по i-му етапу.

$$T_{III} = 402,86 / 5,84 = 69 \text{ дн.}$$

Порівнюючі отримані результати з розрахунками тривалості розробки, вибираємо для подальших розрахунків останнє значення, так як воно, на наш погляд, ближче до фактичної тривалості розробки.

Визначення вартості програмного продукту. Ціна визначається за формулою (4.6):

$$Ц = К * V_{ндр} + Пр, \quad (4.6)$$

де:

$V_{ндр}$ – витрати на розробку програмної продукції;

$К = 1,1$;

$Пр$ – нормативний прибуток, розраховується за формулою (4.7):

$$Пр = С * Рн / 100, \quad (4.7)$$

де:

					КРБ.КІ.1.442-03.3.11	Арк.
						69
Змн.	Арк.	№ докум.	Підпис	Дата		

P_n – норматив рентабельності, % $P_n = 20$ %;

C_m – матеріальні витрати.

$V_{ндр}$ визначаються на підставі складання кошторису витрат на проведення НДР у таблиці.

Витрати по заробітній платі визначаються як сума заробітної плати усіх учасників НДР. Орієнтовний склад учасників, ступінь їх участі у НДР та заробітна плата наведені у таблиці 4.3.

Таблиця 4.3

Орієнтовний склад учасників НДР, їх заробітна плата та ступінь участі

Учасник НДР	Місячна заробітна плата, грн.	Тривалість роботи, міс	Ступінь участі, %
Студент-дослідник	1875	6	90
Науковий керівник роботи	8900	4	10

Основна заробітна плата виконавця з урахуванням окладу і часу.

$$V_{зп} = 8900 * 65,77 * 0,1 / 22 + 1875 * 65,77 * 0,9 / 22 = 7666 \text{ грн}$$

Z_i – середньомісячний оклад

D_p – середня кількість робочих днів ($D_p = 22$)

τ_i - трудомісткість робіт ($\tau_i = 36$)

K_0 – коефіцієнт обліку окладу керівників і консультантів проекту
($K_0 = 0,1$)

Додаткова заробітна плата 10 % від основної заробітної плати.

$$V_{зд} = 7666 * 0,1 = 766,6 \text{ грн.}$$

Єдиний соціальний внесок беруть у розмірі 22 % від величини заробітної плати.

$$ЄСВ = 0,22 * (7666 + 766,6) = 1855,17 \text{ грн.}$$

Витрати, пов'язані зі споживанням електроенергії при розробки програмного продукту, дорівнюють множенню вартості 1 кВт*г електроенергії

					<i>КРБ.КІ.1.442-03.3.11</i>	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		70

та її потреби. Розрахунок надано в таблиці 4.5. Отже, витрати на електроенергію складуть 1584,0 грн.

Таблиця 4.4

Розрахунок витрат на електроенергію.

Напрямок використання	Потреба в е/енергії, кВт*год.	Вартість 1 кВт*год, грн.	Витрати на е/енергію, грн.
Освітлювальні та інші потреби	600	2,64	1584,0

Таблиця 4.5

Кошторис витрат

Найменування статей витрат	Сума витрат, грн.
1. Матеріали	0
2. Паливо та енергія	1584,0
3. Заробітна плата (основна і додаткова)	8432,6
4. Єдиний соціальний внесок	1855,17
5. Інші витрати	1251,47
6. Накладні витрати	4129,87
Виробнича собівартість	17253,11
Адміністративні витрати 3,5 % від виробничої собівартості	603,86
Витрати на збут – 1 % від виробничої собівартості	172,53
Всього повна собівартість	18029,5

Нормативний прибуток:

$$\text{Пр} = (18029,5 - 388) * 0,15 = 2646,23 \text{ грн.}$$

Ціна програмного продукту складе:

$$\text{Ц} = 1,1 \times 18029,5 + 2646,23 = 22\ 478,68 \text{ грн.}$$

Прибуток у ціні ПП становить:

$$\text{П} = 18029,5 \times 0,25 = 4\ 507,38 \text{ грн}$$

Ціна ПП становить:

$$\text{Ц} = 18029,5 \times 1,1 + 4\ 507,38 = 24\ 339,83 \text{ грн}$$

					КРБ.КІ.1.4.42-03.3.11	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		71

4.2.3 Розрахунок капітальних витрат

Розрахунок капітальних витрат, зв'язаних із впровадженням програмного продукту, здійснюється за формулою (4.8):

$$K_2 = K_{\text{пр}} + K_{\text{п}} + K_{\text{ко}} + K_{\text{во}} \quad (4.8)$$

де: $K_{\text{пр}}$ – ціна програмного продукту;

$K_{\text{п}}$ – попередвиробничі витрати;

$K_{\text{ко}}$ – вартість комп'ютерного устаткування;

$K_{\text{во}}$ – вартість допоміжного устаткування, необхідного для надійної роботи
грн;

Ціна програмного продукту становить - 24 339,83 грн.

$K_2 = 24 339,83$ грн.

4.2.4 Розрахунки економічної ефективності

Очікуваний економічний ефект визначається за формулою (4.9) :

$$E_o = (C1 - C2) - E_n \cdot (K2 - K1) \quad (4.9)$$

де $C1, C2$ – поточні витрати відповідно до і після впровадження проекту; ($C1 - C2$) – річна економія на поточних витратах, грн.;

$K2$ – капітальні витрати на впровадження ПП, грн.;

$K1$ – капітальні витрати до впровадження ПП, грн.;

E_n – нормативний коефіцієнт ефективності одноразових витрат (рекомендовано $E_n = 0,25$)

Припустимо, що капітальні витрати до впровадження ПП ($K1$) дорівнюють нулю.

Припустимо, що:

1. Поточні витрати після впровадження проекту ($C2$) дорівнюють 17253,11 грн/рік.
2. Поточні витрати до впровадження проекту ($C1$) були 36624,30 грн/рік (щоб забезпечити відповідну економію).

$$K2 = 24 339,83 \text{ грн}$$

Розрахунок економічного ефекту:

					КРБ.КІ.1.4.42-03.3.11	Арк.
						72
Змн.	Арк.	№ докум.	Підпис	Дата		

$$E_0 = (36624,30 - 17253,11) - 0,25 \cdot 24\,339,83 = 13286,23 \text{ грн}$$

Потім розраховується коефіцієнт ефективності капітальних витрат по формулі(4.10):

$$E = (C_1 - C_2) / K_2 - K_1 \quad (4.10)$$

$$E = 19371,19 / 24\,339,83 = 0,796$$

Так, як $E > E_n$, то проект є ефективним.

Розраховується строк окупності капітальних витрат на впровадження проекту:

$$T = 1 / E = 1 / 0,362 = 2,76 \text{ року}$$

Результати економічних розрахунків відображаються в підсумковій таблиці 4.6

Таблиця 4.6

Техніко-економічні показники програмного продукту

№	Найменування показників	Одиниця виміру	Значення
1	Трудомісткість розробки проекту	днів	25
2	Капітальні витрати	грн.	24339,83
3	Поточні витрати	грн./рік	17253,11
4	Економічний ефект від реалізації проекту	грн./рік	13286,23
5	Строк окупності	років	1,34
6	Економічна ефективність	коефіцієнт	0,745

Висновки до четвертого розділу

Проект має потенціал для створення активної спільноти та виходу на дохід через платний контент та рекламу. Вибір відповідного програмного забезпечення та моделей монетизації є ключовими факторами економічної ефективності проекту. Економічна доцільність проекту залежить від здатності залучити користувачів якісним контентом та ефективно монетизувати платформу.

					КРБ.КІ.1.442-03.3.11	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		74

РОЗДІЛ 5

ОХОРОНА ПРАЦІ

Законодавство про працю регулює умови праці робітників різних професій та напрямків, техніку безпеки на робочому місці, виробничу санітарію, а також час праці\відпочинку та звільнення чи переведення на іншу роботу. Від заходів та норм охорони праці залежать здоров'я та життя працівників, тому їх дотримання є важливою складовою праці. Кожна людина повинен дбати про власну безпеку, в тому числі на своєму робочому місці, а обов'язком роботодавця є забезпечення цієї безпеки життя та збереження здоров'я працівника.

5.1 Шкідливі та небезпечні чинники на робочому місці

Перед початком роботи нового працівника роботодавець згідно зі ст. 29 КЗпП зобов'язаний проінформувати його під розписку про умови праці, наявні на його робочому місці. У тому числі, про всі небезпечні чи шкідливі виробничі фактори, які ще не усунуто, та про можливі наслідки їх впливу на здоров'я працівника, а також про можливі пільги та компенсації за роботу в таких умовах.

Є декілька видів шкідливих та небезпечних чинників, що можуть зашкодити працівнику: фізичні, хімічні, біологічні, психологічні.

Шкідливими факторами при роботі з персональним комп'ютером є

- електростатичне поле;
- шум (багатьма дослідями встановлено, що шум є загальнобіологічним подразником і в певних умовах може впливати на всі системи життєдіяльності людини);
- вібрація (вібрація – це механічні коливання, що призводять до розладу життєвих функцій людини, шкідливо впливають на роботу обладнання та руйнують будівельні конструкції. Вібрація викликає порушення периферійного кровообігу і нервової системи.);
- електромагнітні випромінювання;

					<i>КРБ.КІ.1.442-03.3.11</i>	Арк.
						75
Змн.	Арк.	№ докум.	Підпис	Дата		

- збільшене нервово-емоційне навантаження на працівника;
- збільшення навантаження на зір;
- дрібні стереостатичні рухи кінцівок.

Ці фактори можуть нашкодити здоров'ю працівника, наприклад підвищення артеріального тиску. Комп'ютер, як будь-який електричний прилад, при його неправильному підключенні, може бути джерелом ураження працівника електричним струмом. Не слід користуватися несправною технікою.

5.2 Попередження впливу шкідливих та небезпечних чинників.

Дотримання законодавчих вимог з питань ОП є обов'язковими не тільки роботодавці, а й працівники. Відповідно до ст. 14 Закону про охорону праці до обов'язків працівників входить:

- дбати про особисту безпеку і здоров'я, а також про безпеку і здоров'я оточуючих людей в процесі виконання будь-яких робіт чи під час перебування на території підприємства;
- знати і виконувати вимоги нормативно-правових актів з ОП, правила поведінки з машинами, механізмами, устаткуванням та іншими засобами виробництва, користуватися засобами колективного та індивідуального захисту;
- проходити у встановленому законодавством порядку попередні та періодичні медичні огляди.

Працівник несе безпосередню відповідальність за порушення зазначених вимог. Таким чином, тільки взаємне дотримання роботодавцем і працівником правил з ОП може максимально забезпечити безпечні умови праці для працюючих осіб.

Для робочого місця людини, праця котрої пов'язана з електроприводом, а саме комп'ютером, важливо дотримуватись наступних правил:

- проводити монтаж і експлуатацію електротехнічних приладів з вимкненим блоком живлення;

					<i>КРБ.КІ.1.442-03.3.11</i>	Арк.
						76
Змн.	Арк.	№ докум.	Підпис	Дата		

- не експлуатувати прилад, якщо порушені його цілісність : пошкоджений корпус чи шнур живлення;
- не користуватися нестандартними інструментами при усуненню несправності;
- не перевантажувати електричні з'єднувачі приладами, сила струму яких у сукупності перевищує допустимі значення для таких пристроїв;
- не торкатися мокрими руками та не витирати вологою ганчіркою електричні кабелі, штепсельні розетки, вимикачі, інші електроприлади які ввімкнені в електромережу;
- при залишенні робочого місця перевіряти, щоб всі електроприлади були вимкнені;
- не вмикайте комп'ютер у приміщенні з високою вологістю;
- не залишайте працюючий пристрій без нагляду тривалий час;
- дроти та силові кабелі повинні бути розташовані так, щоб унеможливити наступ на них або поставити щось важке;
- не можна працювати з комп'ютером під час відкритого корпусу системного блоку.

Заходи з охорони праці користувачів ПК необхідно розглядати в трьох основних аспектах: соціальному, психологічному та медичному. У соціальному плані розв'язання цих проблем пов'язане з оптимізацією умов життя, праці, відпочинку, харчування, побуту, розвитком культури, транспорту. Значне місце у профілактиці розладів здоров'я належить психології праці. Тому заходи, пов'язані з формуванням раціональних виробничих колективів, в яких відсутня психологічна несумісність, сприяють зменшенню нервно-психічного перенапруження, підвищенню працездатності та ефективності праці. Особливої значущості у користувачів відеодісплейних терміналів набуває психоемоційний стрес, який в більшій або меншій мірі проявляється у кожного з них.

Значна роль у профілактиці захворювань користувачів ПК відводиться медицині. Існує перелік профілактичних заходів для користувачів ПК, що

					<i>КРБ.КІ.1.442-03.3.11</i>	<i>Арк.</i>
						77
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		

включає як складові первинної профілактики здоров'я (професійний відбір), так і вторинної, яка направлена на зниження ймовірності розвитку перевтоми та перенапруження. Ці комплексні заходи спрямовані на відновлення функціонального стану здорового та опорно - рухового апарату.

5.3 Вимоги до приміщення.

Вимоги до приміщення:

1. Всі приміщення для роботи з ВДТ(візуальний дисплейний термінал) мають відповідати вимогам ДСанПІН 3.3.2.007-98.
2. Розміщення робочих місць з ВДТ ЕОМ і ПЕОМ у підвальних приміщеннях, на цокольних поверхах заборонено.
3. Площа на одне робоче місце становить не менше ніж $6,0 \text{ м}^2$, а об'єм - не менше ніж $20,0 \text{ м}^3$.
4. Приміщення для роботи з ВДТ повинні мати природне та штучне освітлення відповідно до СніП ІІ-4-79.
5. Природне освітлення має здійснюватись через світлові прорізи, орієнтовані переважно на північ чи північний схід, і забезпечувати коефіцієнт природньої освітленості (КПО) ніж 1,5%.
6. Виробничі приміщення повинні обладнуватись шафами для зберігання документів, магнітних дисків, полицями, стелажами, тумбами тощо, з урахуванням вимог до площі приміщень. У приміщеннях з ВДТ слід щоденно робити вологе прибирання.
7. Приміщення з ВДТ мають бути оснащені аптечками першої медичної допомоги.
8. При приміщеннях з ВДТ мають бути обладнані побутові приміщення для відпочинку під час роботи, кімната психологічного розвантаження. (СНІП 2.09.04.- 87).

Також окрему увагу слід приділяти освітленню, що відіграє важливу роль у житті людини. . Біля 90% інформації сприймається через зоровий канал, тому правильно виконане раціональне освітлення має важливе значення для

					КРБ.КІ.1.442-03.3.11	Арк.
						78
Змн.	Арк.	№ докум.	Підпис	Дата		

виконання всіх видів робіт. Раціональне освітлення повинно відповідати таким умовам: бути достатнім (відповідним нормі); рівномірним; не утворювати тіней на робочій поверхні; не засліплювати працюючого; напрямок світлового потоку повинен відповідати зручному виконанню роботи. Це сприяє підтримці високого рівня працездатності, зберігає здоров'я людини та зменшує травматизм.

5.4 Пожежна безпека

Питання пожежної безпеки є актуальним на даний момент, а правила важливою складовою не тільки праці на підприємстві чи виробництві, а побутового життя, особливо це стосується професій, робота котрі безпосередньо пов'язані з електроприладами.

Причини займань та пожеж, що можуть статися під час праці в приміщеннях з ПК:

- несправність техніки (перегрів техніки, коротке замикання струмовідних частин техніки);
- не дотримання правил пожежної безпеки під час праці.

Перш за все треба знати як попередити та завадити виникненню небезпечної ситуації з відкритим вогнем.

Основні вимоги пожежної безпеки:

- шнури до електроприладів, повинні бути справні та без ушкоджень;
- температура зовнішньої поверхні електроопалювальних приладів у найбільш нагрітому місці в нормальному режимі роботи не повинна перевищувати 85°C;
- відстань від приладів електроопалення до горючих матеріалів має становити не менше 0,25 м.

Не дозволяється:

- застосовувати для опалення приміщення нестандартне (саморобне) електронагрівальне обладнання, або лампи розжарювання;

					<i>КРБ.КІ.1.442-03.3.11</i>	Арк.
						79
Змн.	Арк.	№ докум.	Підпис	Дата		

- користуватися несправними розетками, відгалужувальними, вимикачами, іншими пошкодженими електроприладами;
- лампами, скло яких має сліди затемнення або деформування;
- залишати без нагляду ввімкнені в електромережу нагрівальні прилади (праска, електроплитка, електричний чайник тощо);
- використовувати побутові електронагрівальні прилади без негорючих підставок;
- клеювати електропроводку шпалерами або горючими тканинами.

По-друге треба знати як коректно загасити полум'я під напругою. Під час горіння електроприладів треба - висмикнути шнур з розетки, після відключити електроенергію загальним вимикачем. Потім приступати до гасіння.

Для ліквідації невеликих осередків пожеж, а також для гасіння пожеж у початковій стадії їх розвитку силами персоналу об'єктів, застосовуються первинні засоби пожежогасіння. До них відносяться: вогнегасники, пожежний інвентар. Їх застосовують для ліквідації невеликих загорянь до приведення в дію стаціонарних та пересувних засобів гасіння пожежі або до прибуття пожежної команди.

Палаючу проводку можна погасити за допомогою:

- газовими і порошковими вогнегасниками;
- застосовуйте пісок і інші склади, які входять в стандартну комплектацію пожежних щитів на виробничих;
- якщо горять побутові електроприлади - можна використовувати щільні покривала;
- використовуйте підручні негорючі порошки, наприклад:
- сіль, сода або пральний порошок,
- земля з горщиків і інші;

					<i>КРБ.КІ.1.442-03.3.11</i>	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		80

- в зимовий період для гасіння можна використовувати сніг - просто зліпите сніжок і закиньте в осередок загоряння. Такі сніжки безпечні з точки зору ураження від електричного струму, а коротке замикання.

Вогнегасник - найпростіший і ефективний спосіб погасити полум'я на ранніх стадіях. Водні і пінні типи можуть використовуватися для гасіння мережі без напруги, а газові, найчастіше вуглекислотні, і порошкові суміші використовуються для проводки під напругою до 1000 вольт. При цьому необхідно дотримуватися безпечну відстань - не наближатися ближче, ніж на один метр до вогню.

Висновки до п'ятого розділу

Попередження несприятливих наслідків для життя та здоров'я, дотримання правил охорони праці та пожежної безпеки є обов'язковим для кожної людини не тільки на робочому місці але й у побуті. Працівник повинен пам'ятати що недотримання правил створює загрозу для небезпечних ситуацій не тільки для себе, а й оточуючих. Знання цих норм та правил збережуть здоров'я, чи навіть життя.

					<i>КРБ.КІ.1.442-03.3.11</i>	Арк.
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		81

ЗАГАЛЬНІ ВИСНОВКИ

Аналізуючи предметну область, окреслила\сформувала основні характеристики сайтів-форумів. Дослідила основні методи створення контенту.

Розглянуто сучасні подібності таких сайтів-форумів. Виділяю такі переваги: простота використання, активна база користувачів, можливість персоналізації; недоліки: складність конфігурації, необхідність постійного моніторингу контенту та високі витрати на підтримку застосунку.

Проаналізуючи аналоги, їх функціональні можливості, зробила даний висновок: успішні форуми є прості у використанні.

Здійснено постановку задачі з зазначенням бажаного результату.

Був розроблений концептуальний документ проекту. Цей документ описує загальну ідею проекту, його цілі, зазначав цільову аудиторію, описував можливий функціонал. Описано план розвитку, а саме терміни реалізації проекту, представлені потенційні джерела доходів, такі як реклама, преміум-підписка та інтернет-магазин. Цей документ є основою для обговорення з потенційними інвесторами і допомагає їм зрозуміти комерційний потенціал проекту.

Документ про дизайн форуму - описує ключові візуальні особливості сайту: детальний опис користувацького інтерфейсу, макет сторінок, елементи навігації, кольорову палітру. Також визначається функціональність кожної сторінки. Ставляться вимоги до адаптивності дизайну, щоб забезпечити зручність використання форуму на мобільних пристроях і різних екранах.

Проектна документація слугує напрямком дії для розробників і дизайнерів.

					КРБ.КІ.1.442-03.3.11	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		82

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Як будувати UML-діаграми // DOU: [Веб-сайт]. URL: <https://dou.ua/forums/topic/40575/> (дата звернення: 17.04.2024).
2. Безпека веб-сайтів. Загрози безпеці веб-сайтів та поради щодо захисту від шкідливих атак//OniStudio : [Веб-сайт]. URL: <https://www.onistudio.com.ua/bezpeka-veb-sajtiv/> (дата звернення: 24.03.2024).
3. SQL-ін'єкція // Вікіпедія: [Веб-сайт]. URL: <https://uk.wikipedia.org/wiki/SQL%D1%96%D0%BD%27%D1%94%D0%BA%D1%86%D1%96%D1%8F> (дата звернення: 23.03.2024).
4. Що таке https? І навіщо він потрібен кожному сайту // Хостинг Україна: [Веб-сайт]. URL: <https://www.ukraine.com.ua/uk/blog/seo-optimization/chto-takoe-https-i-zachem-on-nuzhen-kazhdomu-sajtu.html> (дата звернення: 08.04.2024).
5. SQL // Вікіпедія: [Веб-сайт]. URL: <https://uk.wikipedia.org/wiki/SQL> (дата звернення: 03.11.2024).
6. PHP // Вікіпедія: [Веб-сайт]. URL: <https://uk.wikipedia.org/wiki/PHP> (дата звернення: 11.03.2024).
7. JavaScript // Вікіпедія: [Веб-сайт]. URL: <https://uk.wikipedia.org/wiki/JavaScript> (дата звернення: 11.03.2024).
8. CSS // Вікіпедія: [Веб-сайт]. URL: <https://uk.wikipedia.org/wiki/CSS> (дата звернення: 11.03.2024).
9. HTML // Вікіпедія: [Веб-сайт]. URL: <https://uk.wikipedia.org/wiki/HTML> (дата звернення: 11.03.2024).
10. Frontend vs backend: відмінності та особливості розробки // Ком'юніті: [Веб-сайт]. URL:

					КРБ.КІ.1.442-03.3.11	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		83

<https://timeweb.com/ru/community/articles/frontend-i-backend-razlichiya-osobennosti-i-trebovaniya-k-specialistam> (дата звернення: 04.04.2024).

11. Вебфорум // Вікіпедія: [Веб-сайт]. URL: <https://uk.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1%D1%84%D0%BE%D1%80%D1%83%D0%BC> (дата звернення: 03.04.2024).

12. Юзабіліті сайту // WebTune: [Веб-сайт]. URL: <https://webtune.com.ua/statti/web-rozrobka/yuzabiliti-sajtu/> (дата звернення: 04.03.2024).

13. Статичні та динамічні web-сайти // ArmedSoft: [Веб-сайт]. URL: <https://armedsoft.com/ua/blog/statychni-ta-dynamichni-web-sayty> (дата звернення: 27.02.2024).

14. Перевірка адаптивності сайту за допомогою браузера // WebTune: [Веб-сайт]. URL: <https://webtune.com.ua/statti/internet-marketing/yak-pereviryty-adaptyvnist-za-dopomogoyu-brauzera/> (дата звернення: 04.03.2024).

15. Структура сайту: основні види та правила їх розробки // WebTune: [Веб-сайт]. URL: <https://webtune.com.ua/statti/web-rozrobka/struktura-sajtu/> (дата звернення: 05.02.2024).

16. REST API як спосіб спілкування компонент веб-додатків // FoxmindEd: [Веб-сайт]. URL: <https://foxminded.ua/shcho-take-rest-api/> (дата звернення: 14.02.2024).

17. What is an API // IBM: [Веб-сайт]. URL: <https://www.ibm.com/topics/api> (дата звернення: 13.02.2024).

18. Список кодів стану HTTP // Вікіпедія: [Веб-сайт]. URL: https://uk.wikipedia.org/wiki/%D0%A1%D0%BF%D0%B8%D1%81%D0%BE%D0%BA_%D0%BA%D0%BE%D0%B4%D1%96%D0%B2_%D1%81%D1%82%D0%B0%D0%BD%D1%83_HTTP (дата звернення: 03.02.2024).

19. HTTP // Вікіпедія: [Веб-сайт]. URL: <https://ru.wikipedia.org/wiki/HTTP> (дата звернення: 05.02.2024).

					<i>КРБ.КІ.1.442-03.3.11</i>	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		84

20. Клієнт-серверна архітектура // Вікіпедія: [Веб-сайт]. URL: <https://uk.wikipedia.org/wiki/%D0%9A%D0%BB%D1%96%D1%94%D0%BD%D1%82%D1%81%D0%B5%D1%80%D0%B2%D0%B5%D> (дата звернення: 02.02.2024).

					КРБ.КІ.1.442-03.3.11	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		85