

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Інформаційно-аналітичних технологій та менеджменту
(повна назва)

Кафедра Інформатики
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА Пояснювальна записка

рівень вищої освіти перший (бакалаврський)

РОЗРОБКА ВЕБЗАСТОСУНКУ ДЛЯ ПІДТРИМКИ МЕНТАЛЬНОГО ЗДОРОВ'Я

(тема)

Виконав:
здобувач 4 року навчання,
групи ІТІНФ-21-3

Карабута М. О.
(прізвище, ініціали)

Спеціальність 122 Комп'ютерні науки
(код і повна назва спеціальності)

Тип програми освітньо-професійна

Освітня програма Інформатика
(повна назва освітньої програми)

Керівник доц. Шафроненко А. Ю.
(посада, прізвище, ініціали)

Допускається до захисту

Завідувач кафедри інформатики _____
(підпис)

Кобилін О. А.
(прізвище, ініціали)

2025 р.

Харківський національний університет радіоелектроніки

Факультет Інформаційно-аналітичних технологій та менеджментуКафедра ІнформатикиРівень вищої освіти перший (бакалаврський)Спеціальність 122 Комп'ютерні науки
(код і повна назва)Тип програми освітньо-професійнаОсвітня програма Інформатика
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)

«_____» _____ 2025 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУздобувачеві Карабуті Микиті Олександровичу
(прізвище, ім'я, по батькові)1. Тема роботи Розробка вебзастосунку для підтримки ментального здоров'я

затверджена наказом університету від 19 травня 2025 року № 381Ст

2. Термін подання здобувачем роботи до екзаменаційної комісії 01 червня 2025 р.

3. Вихідні дані до роботи науково-методична та науково-технічна література, матеріали конференцій, дані інтернет-мережі.

4. Перелік питань, що потрібно опрацювати в роботі _____

1. Аналіз предметної області.

2. Огляд фреймворків javascript для створення вебзастосунків.

3. Програмна реалізація вебзастосунку.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (п.5 включається до завдання за рішенням випускової кафедри) Актуальність проблеми розробки вебзастосунка з ментального здоров'я, постановка задачі, огляд фреймворків javascript, опис програмної реалізації вебзастосунка.

6. Консультанти розділів роботи (п.6 включається до завдання за наявності консультантів згідно з наказом, зазначеним у п.1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Строк / терміни виконання етапів роботи	Примітка
1	Отримання завдання на кваліфікаційну роботу	07.04.2025	
2	Аналіз завдання, підбір літератури	08.04.25-10.04.25	
3	Аналіз літератури з досліджуваної проблеми	11.04.25-14.04.25	
4	Аналіз технічних засобів	15.04.25-20.04.25	
5	Розробка вебзастосунку	21.04.25-27.04.25	
6	Програмна реалізація	28.04.25-11.05.25	
7	Оформлення пояснювальної записки	12.05.25-20.05.25	
8	Перевірка на нормоконтроль	21.05.25-01.06.25	
9	Перевірка на плагіат	21.05.25-01.06.25	
10	Рецензування	21.05.25-01.06.25	
11	Підготовка презентації та доповіді	21.05.25-18.06.25	
12	Занесення роботи в електронний архів	02.06.25-18.06.25	
	Попередній захист кваліфікаційної роботи	02.06.25-18.06.25	

Дата видачі завдання 7 квітня 2025 р.

Здобувач _____
(підпис)

Керівник роботи _____ доц. Шафроненко А. Ю.
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ/ABSTRACT

Пояснювальна записка до кваліфікаційної роботи: 60 с., 18 рис., 45 джерел.

МЕНТАЛЬНЕ ЗДОРОВ'Я, БІБЛІОТЕКА REACT, ПСИХОЕМОЦІЙНА ПІДТРИМКА, ФРЕЙМБОРК JAVASCRIPT, СТЕК.

Об'єктом роботи є процес розробки вебзастосунку, спрямованого на популяризацію знань про ментальне здоров'я та покращення емоційного благополуччя користувачів за допомогою цифрових технологій.

Мета роботи – розробити вебзастосунок, що сприятиме поширенню інформації про ментальне здоров'я, забезпечуватиме зручний доступ до тематичного контенту та інтерактивних можливостей для користувачів, а також сприятиме підвищенню рівня обізнаності та саморефлексії щодо психічного благополуччя.

В результаті роботи є створення вебзастосунку, що має на меті стати доступним, корисним та інтерактивним ресурсом, який не лише популяризує знання про психічне здоров'я, але й створює простір для самоаналізу, самопідтримки та розвитку.

MENTAL HEALTH, REACT LIBRARY, PSYCHO-EMOTIONAL SUPPORT, JAVASCRIPT FRAMEWORK, STACK.

The object of the work is the process of developing a web application aimed at popularizing knowledge about mental health and improving the emotional well-being of users using digital technologies.

The purpose of the work is to develop a web application that will contribute to the dissemination of information about mental health, provide convenient access to thematic content and interactive opportunities for users, and also contribute to increasing the level of awareness and self-reflection regarding mental well-being.

The result of the work is the creation of a web application that aims to become an accessible, useful, and interactive resource that not only popularizes knowledge about mental health but also creates a space for introspection, self-support, and development.

ЗМІСТ

Перелік умовних позначень, символів, одиниць, скорочень і термінів	6
Вступ.....	7
1 Аналіз предметної області та постановка завдання	9
1.1 Аналіз сучасного стану психічного здоров'я в Україні.....	12
1.2 Огляд існуючих цифрових рішень	13
1.3 Можливості та переваги вебзастосунку	17
1.4 Постановка задачі	19
2 Огляд фреймворків javascript для створення вебзастосунків.....	21
2.1 Фронтенд-фреймворк Angular.....	21
2.2 JavaScript-фреймворк Vue.js	23
2.3 Бібліотека React	26
2.4 Ember.js	28
2.5 D3.js	30
3 Опис програмної реалізації.....	33
3.1 Основні принципи розробки вебзастосунків	33
3.2 Приклади стеків технологій веброзробки	37
3.3 Обґрунтування вибору засобів розробки вебзастосунку.....	40
3.4 Програмна реалізація	42
Висновки.....	54
Перелік джерел посилання	56

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

ВПО – внутрішньо переміщені особи

ПТСР – посттравматичний стресовий розлад

CLI – Command Line Interface (командний інтерфейс)

SPA – Single-Page Application (односторінковий вебзастосунок)

MVC – Model-View-Controller (модель-представлення-контролер)

HTML – HyperText Markup Language (мова розмітки)

CSS – Cascading Style Sheets (мова стилів)

JSX – JavaScript XML

DOM – Document Object Model (модель об'єкта документа)

API – Application Programming Interface

PWA – Progressive Web Application

ВСТУП

У сучасному світі стрімкі зміни в технологіях і глобальні події кардинально впливають на спосіб життя, психологічний стан і спосіб взаємодії людей. Розвиток інтернет-технологій, мобільних застосунків та онлайн-платформ відкриває перед суспільством нові можливості для підтримки психічного здоров'я, пропонуючи альтернативні інструменти для полегшення стресу та зміцнення емоційного балансу.

Швидкість сучасних подій та постійне зростання інформаційного потоку ставлять перед людьми складні завдання з адаптації та збереження психологічної стійкості. Сучасні цифрові рішення дозволяють не тільки оперативно реагувати на виклики, але й навчатися керувати власними емоціями за допомогою інтерактивних сервісів, які пропонують комплекси методик релаксації, медитації та підтримки через онлайн-консультації. Такий підхід відкриває нову сторінку в системі підтримки психічного здоров'я, де традиційні методи поєднуються з інноваціями.

Одним із важливих напрямків у сфері цифрової психологічної допомоги є використання вебзастосунків, які полегшують доступ до ресурсів підтримки незалежно від місця перебування користувача. За допомогою таких додатків можна не лише проводити регулярний самоконтроль настрою, а й брати участь у сеансах з медитації, практикувати техніки дихання та отримувати консультації кваліфікованих фахівців у режимі онлайн. Це допомагає створити персоналізований комплекс заходів, спрямований на зниження рівня стресу та покращення загального психоемоційного стану.

Цифрові технології дозволяють демократизувати доступ до психологічної підтримки. Вони допомагають людям знаходити ефективні способи подолання життєвих труднощів, не обмежуючись традиційними методами лікування. Інтернет-консультації, мобільні щоденники настрою та інтерактивні платформи дозволяють виявляти проблемні моменти на ранніх стадіях, що є надзвичайно важливим у попередженні серйозних психічних

розладів. Такий підхід створює умови для більш гнучкого та оперативного реагування на потреби користувачів.

Поєднання класичних підходів до психологічної підтримки з цифровими інструментами відкриває нові перспективи для ефективного лікування та профілактики. Онлайн-платформи дозволяють не лише отримувати консультації в режимі реального часу, а й вести систематичний моніторинг емоційного стану, що дозволяє краще розуміти індивідуальні потреби кожної людини. Така інтеграція сприяє формуванню цілісної системи підтримки, яка враховує як традиційні, так і інноваційні підходи до збереження психічного здоров'я.

Цифрові технології стають ключовим чинником у зміні підходу до підтримки психічного здоров'я. В умовах глобальних викликів та швидких змін вони відкривають нові можливості для ефективної роботи з емоційним станом користувачів. Онлайн-платформи та мобільні застосунки не лише розширюють доступ до кваліфікованої психологічної допомоги, а й сприяють зниженню рівня стресу, формуючи нові стандарти у сфері підтримки внутрішньої гармонії. Сучасне суспільство, впроваджуючи такі інновації, крок за кроком наближається до створення стійкої системи ментального здоров'я, що відповідає вимогам часу.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАВДАННЯ

У сучасному світі глобалізація, стрімкий технологічний розвиток та постійна соціально-політична нестабільність створюють низку викликів для психічного здоров'я кожної людини [1-8]. Психоемоційна стабільність стає не лише особистою потребою, але й основою для активної участі у суспільному, професійному та особистісному житті. Особливо гостро ця проблема постала у зв'язку з війною в Україні, яка принесла біль, розлуку, масове переміщення населення та поглиблення почуття невизначеності і страху [9]. В таких умовах багато людей потребують професійної психологічної підтримки, адже травматичний досвід і щоденна боротьба з внутрішніми демонами часто залишають глибокий слід.

Сучасна епоха характеризується високою швидкістю потоку інформації. Завдяки глобалізації ми отримуємо новини з усіх куточків світу практично миттєво, що створює додаткове навантаження на психіку. Інформаційне перевантаження, зокрема негативні новини, може спричиняти хронічний стрес, тривожність та інші психоемоційні порушення. Технологічний прогрес, з появою мобільних пристроїв та високошвидкісного інтернету, значно змінив характер повсякденного життя. Постійна доступність до цифрових технологій змушує людину швидко адаптуватися до нових умов і приймати рішення майже в режимі реального часу. Проте така швидкість життя, попри свої переваги, також може призводити до відчуття перенавантаженості, вигорання і зниження загальної якості життя [10, 11].

Особливої уваги заслуговує аспект віртуалізації соціальних контактів, коли традиційні особисті зустрічі поступово замінюються цифровими комунікаціями. Хоча технології дають можливість підтримувати зв'язок на відстані, вони часто не забезпечують повноцінної емоційної підтримки, котра необхідна для збереження психічного здоров'я. Соціальні мережі та месенджери, дозволяючи бути на зв'язку з друзями та колегами, тим не менш

не можуть замінити тієї глибини емоційного контакту, яка виникає при спілкуванні віч-на-віч.

Соціально-політична нестабільність також вносить свій вклад у психоемоційний стан населення [9-11]. Умови невизначеності, економічні кризи та глобальні конфлікти створюють постійне відчуття небезпеки і страху за майбутнє [2]. Війна в Україні стала яскравим прикладом того, як розрив соціальних зв'язків, втрата близьких і примусове переміщення можуть спричинити низку психологічних розладів [3-8]. Люди, які пережили військові дії та пов'язані з ними травматичні події, часто стикаються з посттравматичним стресовим розладом, депресією та загальною тривожністю. Така ситуація погіршується і через руйнування традиційних соціальних структур, що зазвичай слугують джерелом підтримки та допомоги, що ще більше ускладнює доступ до професійної психологічної допомоги.

Незважаючи на високий попит, кваліфікована психологічна підтримка залишається недоступною для широких верств населення, особливо в малих населених пунктах. Географічні бар'єри, недостатня кількість фахівців і обмежені ресурси створюють значні труднощі на шляху до отримання необхідної допомоги. До того ж, соціальна стигматизація звернення за психологічною підтримкою залишається актуальною проблемою в багатьох регіонах, що змушує багатьох осіб уникати необхідної консультації, що в подальшому погіршує їхній стан.

У цьому контексті цифровізація суспільства пропонує нові можливості для вирішення проблем психічного здоров'я. Мобільні застосунки та онлайн-платформи стають важливим інструментом, що дозволяє забезпечити дистанційний доступ до психологічної інформації, інструментів самопомоги та безпосереднього зв'язку з фахівцями [9, 10]. Онлайн-консультації, чат-боти та відеоконференції відкривають можливості отримати кваліфіковану допомогу незалежно від місцевого розташування пацієнта, що особливо актуально для людей, які мешкають у віддалених регіонах.

Вебзастосунки пропонують широкий спектр інструментів самопомоги. Серед них – техніки релаксації, медитації, вправи для управління стресом, щоденники настрою, які дозволяють користувачам самостійно контролювати свій стан і знижувати ризик розвитку більш серйозних психічних розладів. Крім того, такі платформи є майданчиком для популяризації культури психічного здоров'я. Проведення освітніх кампаній, семінарів і вебінарів сприяє підвищенню обізнаності населення щодо методів підтримки психічного благополуччя та зменшення стигматизації звернень за допомогою [11].

Цифрові технології також допомагають розвантажити традиційну систему охорони здоров'я [12-16]. Попередня цифрова консультація дозволяє визначити потреби людини і спрямувати її до кваліфікованого фахівця, що сприяє більш ефективному використанню ресурсів і скороченню часу очікування [16-22]. Безпека та конфіденційність даних користувачів стають пріоритетом для розробників мобільних додатків, що знижує бар'єр для звернення за допомогою в онлайн-середовищі.

Перспективи розвитку цифрових технологій у сфері психічного здоров'я виглядають надзвичайно обнадійливо. Інтеграція з традиційною системою охорони здоров'я дозволить створити єдину мережу підтримки, де мобільні застосунки стають першою лінією контакту для осіб, що потребують психологічної допомоги. Завдяки використанню штучного інтелекту [9, 13-25], такі платформи здатні проводити аналіз емоційного стану користувачів та надавати персоналізовані рекомендації, що значно підвищує ефективність наданої допомоги. Розвиток соціальних платформ всередині мобільних застосунків створить можливості для обміну досвідом та підтримки серед людей, які стикаються з подібними проблемами, що є важливим доповненням до професійного лікування.

Таким чином, психічне здоров'я стає ключовим чинником успішної адаптації та функціонування в умовах сучасних глобальних викликів. Забезпечення доступу до комплексної підтримки громадян, особливо через

цифрові технології, є необхідною умовою для створення стійкого та гармонійного суспільства. Мобільні застосунки, що дозволяють отримувати психологічну допомогу дистанційно, не лише оптимізують роботу системи охорони здоров'я, але й сприяють формуванню культури, де підтримка психічного благополуччя стає природною складовою повсякденного життя кожного громадянина.

1.1 Аналіз сучасного стану психічного здоров'я в Україні

Проблематика психічного здоров'я в Україні останніми роками набула особливої актуальності, що зумовлено як соціально-економічними змінами, так і впливом військових дій [3-8]. За офіційними даними Міністерства охорони здоров'я України, станом на листопад 2022 року кількість звернень громадян до психологів і психіатрів перевищила 650 тисяч осіб, що вдвічі більше порівняно з 2019 роком (302 тисячі). Така динаміка свідчить про зростаючу потребу населення у психологічній підтримці.

Проте, незважаючи на збільшення запитів, рівень фактичного звернення за професійною допомогою залишається критично низьким. Лише 2% українців, які регулярно відчувають ознаки психоемоційного виснаження (стрес, тривожність, дратівливість), звертаються до фахівців. Це означає, що існує значна кількість осіб, які залишаються без належної допомоги, що може призводити до хронічних порушень психічного здоров'я, зниження якості життя та соціальної ізоляції.

Серед основних бар'єрів, що стримують звернення до психологів, варто виділити:

- фінансові труднощі – високий вартісний бар'єр для приватних консультацій;
- географічну недоступність – особливо в сільських та прифронтових районах, де фахівці або відсутні, або їх недостатньо;

- суспільні упередження – стигматизація психічних розладів і страх бути осудженим;
- нестачу кваліфікованих спеціалістів – нерівномірне розподілення кадрів і обмежений доступ до якісної психотерапевтичної допомоги;
- інформаційну ізолюваність – багато людей не знають, куди і як звернутися по допомогу.

Вказані фактори актуалізують необхідність у розвитку альтернативних каналів підтримки психічного здоров'я, одним із найперспективніших напрямів є використання цифрових технологій. Вебзастосунки, мобільні додатки та онлайн-платформи здатні забезпечити доступність, конфіденційність і гнучкість у наданні інформаційної та психологічної підтримки. Вони мають потенціал компенсувати нестачу традиційних ресурсів і одночасно слугувати зручним інструментом профілактики та раннього втручання.

1.2 Огляд існуючих цифрових рішень

На міжнародному ринку представлено велику кількість вебзастосунків, що мають на меті покращення психічного здоров'я та емоційного стану користувачів. Ці програми охоплюють широкий спектр напрямів: від медитацій і дихальних практик до ведення щоденників настрою, психологічної самопомоги та взаємодії з сертифікованими фахівцями.

Деякі вебзастосунки орієнтовані на зниження рівня стресу, тривожності, подолання депресивних станів або поліпшення якості сну. Інші ж спеціалізуються на розвитку навичок усвідомленості (mindfulness), підвищенні рівня самоповаги, формуванні позитивного мислення та стабілізації емоційного фону.

Користувачі можуть обирати програми відповідно до своїх індивідуальних потреб – від простих щоденних вправ до комплексних

терапевтичних курсів, розроблених за участі психотерапевтів та науковців. Крім того, багато з таких застосунків мають зручний інтерфейс, функцію персоналізації та підтримують багатомовність, що робить їх доступними для широкої аудиторії в усьому світі.

Серед найвідоміших:

– Headspace – сучасний цифровий інструмент, який допомагає користувачам долати стрес, підвищувати рівень концентрації та покращувати якість сну завдяки методикам медитації та уважності. Створений на основі багаторічних досліджень у сфері нейронаук і психології, цей вебзастосунок став популярним серед людей, які прагнуть інтегрувати практики усвідомленості у своє повсякденне життя (рис. 1.1);

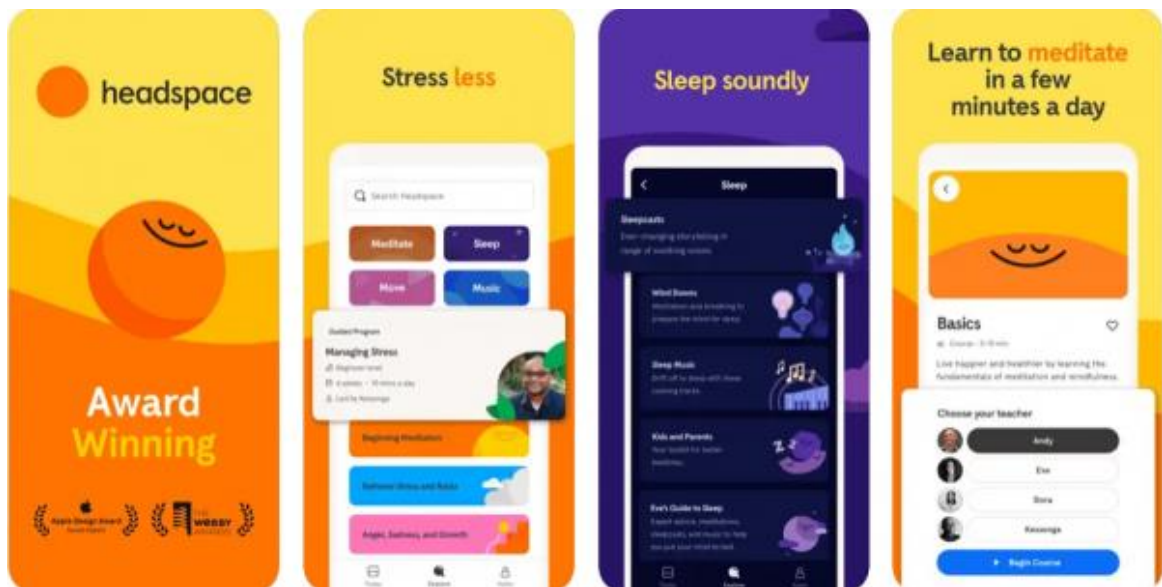


Рисунок 1.1. – Скріншоти застосунку Headspace

– Calm – це один із провідних цифрових інструментів, створених для того, щоб допомогти користувачам знизити рівень стресу, покращити якість сну та загальне емоційне благополуччя за допомогою технік медитації, уважності та релаксації. Цей вебзастосунок здобув широку популярність завдяки інтуїтивно зрозумілому інтерфейсу та насиченому функціоналу, який орієнтований як на новачків, так і на досвідчених користувачів (рис. 1.2);



Рисунок 1.2 – Застосунок Calm

– Moodpath – це сучасний вебзастосунок, що виступає у ролі цифрового щоденника настрою, розроблений для моніторингу психоемоційного стану користувача та раннього виявлення ознак стресу, депресії та тривожних розладів. Завдяки інтегрованим інструментам самоаналізу та зручному інтерфейсу, Moodpath дозволяє не лише відстежувати зміни настрою, але й отримувати персоналізовані рекомендації щодо покращення емоційного благополуччя (рис. 1.3).

Moodpath надає можливість користувачам щодня реєструвати свій емоційний стан за допомогою простих питань, які спрямовані на визначення рівня стресу та загального психоемоційного стану. Заповнення щоденника займає лише декілька хвилин, але систематичне ведення записів дозволяє виявити закономірності та тренди в настрої;

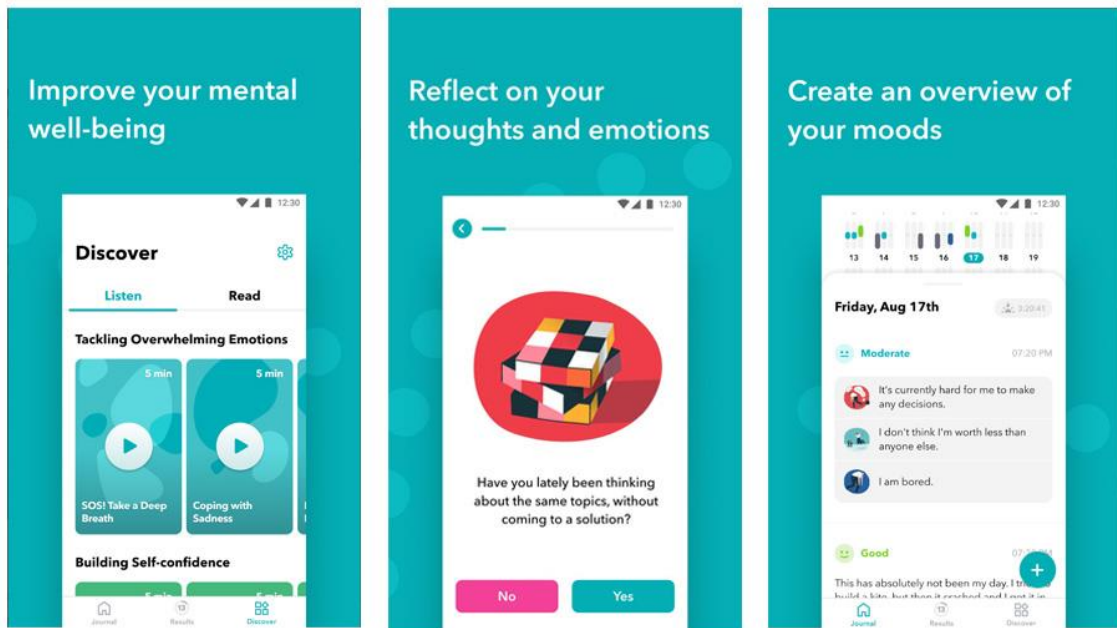


Рисунок 1.3 – Застосунок Moodpath

– BetterHelp – це сучасна онлайн-платформа, що надає можливість отримання психологічної підтримки через інтернет. У відповідь на зростаючі потреби населення у доступній та зручній допомозі, ця платформа об'єднує кваліфікованих фахівців з різних напрямків психології та психотерапії, забезпечуючи конфіденційний зв'язок між користувачами та спеціалістами (рис. 1.4).

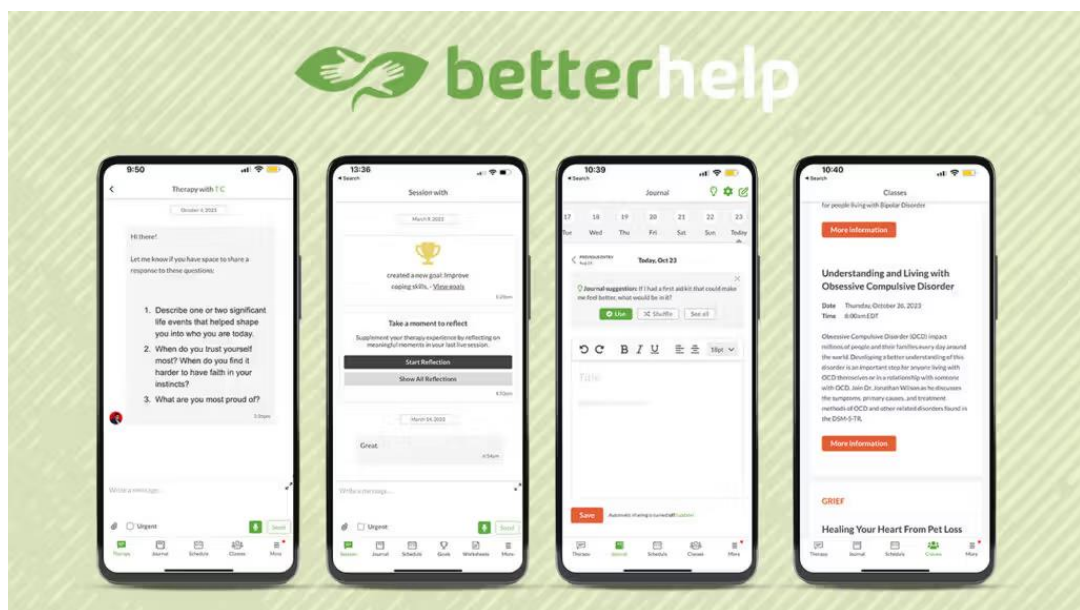


Рисунок 1.4 – Застосунок BetterHelp

1.3 Можливості та переваги вебзастосунку

Вебзастосунок у сфері психічного здоров'я може стати надзвичайно ефективним інструментом, завдяки поєднанню інноваційних технологій і глибокого розуміння потреб користувача. Його ефективність базується на кількох ключових перевагах, що роблять такий формат доступним і корисним для широкого кола людей.

Однією з головних переваг мобільного застосунку є його цілодобова доступність. Користувач може звернутися по допомогу в будь-який час і з будь-якого місця, не обмежуючись робочим графіком спеціаліста або необхідністю відвідувати консультацію офлайн. Це особливо важливо для людей, які живуть у віддалених районах або не мають змоги регулярно відвідувати психолога.

Вебзастосунок дозволяє взаємодіяти з сервісом анонімно, що знижує бар'єри звернення по психологічну допомогу. Страх бути осудженим або неправильно зрозумілим часто стримує людей від звернення до фахівців, особливо в культурах, де тема ментального здоров'я залишається табу. Завдяки цифровому формату користувач почувається більш захищено й вільно.

Багато застосунків пропонують автоматизовані функції, такі як щоденні нагадування, вправи на розслаблення, дихальні практики, короткі тести для самодіагностики тощо. Це допомагає користувачу формувати сталі звички турботи про власний психологічний стан, інтегруючи ці практики в повсякденне життя без додаткового зусилля.

Сучасні додатки використовують гейміфікацію та інтерактивні елементи для підвищення залученості. Наприклад, це можуть бути візуальні щоденники настрою, інтерактивні курси самопізнання, міні-ігри на розвиток емоційного інтелекту, або чат-боти для миттєвої підтримки. Такий підхід робить процес підтримки психічного здоров'я цікавим і мотивуючим.

Застосунок може адаптувати контент до конкретних потреб користувача. Алгоритми машинного навчання аналізують відповіді, стиль взаємодії та прогрес користувача, надаючи індивідуалізовані рекомендації, вправи або ресурси. Це підвищує ефективність програми та сприяє більш глибокому залученню.

Таким чином, вебзастосунок не лише забезпечує зручний та інноваційний спосіб підтримки ментального здоров'я, а й може значно зменшити навантаження на систему охорони здоров'я загалом. Цифрові рішення дозволяють охопити більше людей з меншими витратами, зменшуючи бар'єри доступу до допомоги й водночас підвищуючи якість життя користувачів.

Проаналізувавши вебплатформи, присвячені тематиці ментального здоров'я, можна дійти висновку, що більшість із них або є суто інформаційними ресурсами, або передбачають фінансову участь користувача. Такий формат може стати серйозною перешкодою для осіб, які вже мають ментальні труднощі, зокрема через обмежені фінансові можливості чи психологічний бар'єр.

Попри їх ефективність, ці застосунки мають низку обмежень для українського користувача:

- відсутність локалізації українською мовою;
- висока вартість підписки;
- відсутність урахування ментальних, культурних і військових реалій в Україні.

Тому виникає потреба у створенні національного продукту, адаптованого до контексту українського суспільства, потреб і викликів, який забезпечив би відкритий доступ до перевіреної інформації про психічне здоров'я, мав блогівий розділ з актуальними темами, а також функціонал збереження користувачів у базі даних для персоналізованої взаємодії.

1.4 Постановка задачі

У сучасному світі вебзастосунки набувають дедалі більшої популярності та стають невід'ємною частиною цифрової екосистеми. Інтернет-ресурси вже давно перетворилися на одне з головних джерел інформації для людей у XXI столітті. Практично всі аспекти нашого життя мають своє відображення у глобальній мережі, і щоразу, коли виникає потреба в нових знаннях чи даних, ми насамперед звертаємося саме до онлайн-простору.

Об'єктом роботи є процес розробки вебзастосунку, спрямованого на популяризацію знань про ментальне здоров'я та покращення емоційного благополуччя користувачів за допомогою цифрових технологій.

Мета роботи – розробити вебзастосунок, що сприятиме поширенню інформації про ментальне здоров'я, забезпечуватиме зручний доступ до тематичного контенту та інтерактивних можливостей для користувачів, а також сприятиме підвищенню рівня обізнаності та саморефлексії щодо психічного благополуччя.

Для досягнення поставленої мети потрібно вирішити наступні завдання:

- розробити інтуїтивно-зрозумілий інтерфейс – забезпечення зручного користувацького досвіду для людей різного віку та технічного рівня підготовки;
- розробити функцію реєстрації – створення облікового запису з можливістю персоналізації;
- розробити систему автентифікації – захист доступу до персональних даних користувача;
- розробити кабінет користувача – можливість змінювати особисті налаштування, переглядати власну активність тощо;
- розробити вільний доступ до блогу – усі користувачі зможуть ознайомлюватися з публікаціями без обов'язкової реєстрації;
- розробити функцію управління контентом – створення, редагування, видалення та модерація статей адміністратором або автором;

- розробити пошукову систему – фільтрація статей за категоріями, авторами, ключовими словами;
- розробити психологічні тести – інтеграція опитувальників, що допомагають оцінити емоційний стан користувача.

Цей вебзастосунок має на меті стати доступним, корисним та інтерактивним ресурсом, який не лише популяризує знання про психічне здоров'я, але й створює простір для самоаналізу, самопідтримки та розвитку. Він також може відігравати важливу роль у зниженні соціального стигматизування ментальних проблем.

Основною аудиторією є:

- внутрішньо переміщені особи (ВПО);
- військовослужбовці та ветерани;
- молодь та студенти, які часто зазнають емоційного навантаження;
- особи, які проживають у сільській місцевості або мають обмежений доступ до медицини;
- особи з підозрою на депресію, тривожні розлади, ПТСР.

2 ОГЛЯД ФРЕЙМВОРКІВ JAVASCRIPT ДЛЯ СТВОРЕННЯ ВЕБЗАСТОСУНКІВ

JavaScript – це одна з основних мов програмування, яка лежить в основі сучасної веброзробки, зокрема фронтенду. Саме вона забезпечує інтерактивність та динаміку вебінтерфейсів, дозволяючи створювати зручні, функціональні та привабливі для користувача вебзастосунки. З кожним роком екосистема JavaScript стрімко розвивається, і разом з нею зростає кількість інструментів та бібліотек, що оптимізують робочий процес розробника. Одним із найбільш важливих компонентів цієї екосистеми є фреймворки – структуровані середовища, які пропонують готові рішення для розробки, організації коду, керування станом, маршрутизації та взаємодії з сервером.

На сьогоднішній день існує чимало популярних JavaScript-фреймворків [26-29], кожен з яких має свої унікальні особливості, переваги та сфери застосування. Їхнє використання дозволяє скоротити час розробки, покращити продуктивність вебзастосунків і забезпечити більш зручну підтримку коду в майбутньому.

2.1 Фронтенд-фреймворк Angular

Angular – це потужний фронтенд-фреймворк з відкритим кодом, який розробляється та підтримується компанією Google [29-33]. Він створений для побудови масштабованих, продуктивних і добре структурованих вебзастосунків. Angular належить до так званих «повноцінних» фреймворків, оскільки включає в себе всі необхідні інструменти для розробки – від двостороннього зв'язку даних і системи маршрутизації до вбудованої підтримки форм, валідації, HTTP-запитів та управління залежностями.

Ключові особливості Angular:

– Angular реалізовано на основі мови програмування TypeScript, яка є надбудовою над JavaScript та забезпечує статичну типізацію, підтримку об'єктно-орієнтованих підходів (класів, інтерфейсів, наслідування) та інші механізми, що сприяють підвищенню надійності програмного забезпечення та покращенню його масштабованості, що забезпечує кращу читаємість коду, полегшує його тестування і супровід, а також знижує ймовірність помилок у процесі розробки;

– фреймворк використовує компонентну модель програмування, в якій логіка, шаблони та стилі інтерфейсу об'єднані в окремі функціональні одиниці – компоненти. Такий підхід забезпечує модульність, повторне використання коду, ізолюваність функцій та зручність у тестуванні окремих частин застосунку;

– двостороннє зв'язування даних (Two-way Data Binding). Angular підтримує автоматичну синхронізацію між моделлю даних (дані прикладної логіки) та представленням (графічний інтерфейс користувача), що значно зменшує обсяг рутинного коду та підвищує ефективність розробки динамічних вебінтерфейсів. Такий механізм є особливо корисним під час реалізації інтерактивних форм і реактивних елементів інтерфейсу;

– інтегрований інструментарій розробника. Angular надає розробникам CLI (Command Line Interface) – командний інтерфейс, що дозволяє автоматизувати створення, налаштування та супровід проєктів. CLI включає інструменти для генерації компонентів, сервісів, модулів, а також для проведення тестування та оптимізації кінцевого продукту, що суттєво прискорює розробницький процес;

– архітектура Angular побудована на основі модульного підходу, де код організовується в логічні одиниці – модулі, що дозволяє краще структурувати проєкт, розмежовувати функціональність, забезпечувати повторне використання модулів та їхнє ізолюване тестування, що особливо важливо у великих корпоративних застосунках.

Фреймворк Angular є ефективним інструментом для розробки складних, масштабованих вебзастосунків із розвиненою структурою та високими вимогами до інтерфейсу користувача. Його функціональні можливості, такі як компонентна архітектура, підтримка модульності та інтегровані засоби для керування станом застосунку, зумовлюють широке застосування Angular у низці прикладних сфер.

Зокрема, Angular широко використовується для створення корпоративних вебзастосунків, які потребують високої надійності, масштабованості та зручного адміністрування. Фреймворк також є придатним для розробки систем управління взаємовідносинами з клієнтами, де важливо забезпечити інтерактивність, адаптивність і збереження стану інтерфейсу під час взаємодії з великою кількістю даних.

Крім того, Angular активно застосовується під час створення адміністративних панелей (dashboard-систем), що вимагають динамічного оновлення даних та гнучкої візуалізації, а також інструментів для обробки та керування даними, де важлива інтеграція з формами та механізмами валідації.

До прикладів глобальних компаній, які використовують Angular у своїх проєктах, належать Google, Microsoft, Forbes, Upwork та інші, що свідчить про високу довіру до цього фреймворка з боку технологічно орієнтованих організацій.

2.2 JavaScript-фреймворк Vue.js

Vue.js – це відкритий прогресивний JavaScript-фреймворк, розроблений Еваном Ю (Evan You), який орієнтований на створення інтерфейсів користувача та односторінкових вебзастосунків [34-37]. Його головною особливістю є поступовість. Vue.js легко інтегрується у вже існуючі проєкти, а також може бути використаний як повноцінний фреймворк для складних програмних систем.

Vue.js поєднує в собі переваги найкращих практик із таких фреймворків, як React і Angular, водночас залишаючись легким для вивчення та використання. Його архітектура базується на використанні реактивних даних, компонентному підході та шаблонному синтаксисі, що значно спрощує процес побудови інтерфейсів.

Однією з фундаментальних особливостей фреймворка Vue.js є реактивність – концепція, що передбачає автоматичне оновлення інтерфейсу користувача у відповідь на зміну стану даних. Це досягається за допомогою вбудованого механізму спостереження, який відстежує зміни у даних моделі та синхронізує їх із представленням (view) у режимі реального часу.

У Vue.js реактивність реалізована завдяки використанню спеціальних обгорт (проху-об'єктів) та системи залежностей. Кожна властивість стану додається до реактивної системи, що дозволяє Vue автоматично визначати, які компоненти залежать від яких даних, і оновлювати лише ті частини інтерфейсу, які зазнали змін. Це суттєво зменшує витрати на рендеринг і підвищує продуктивність застосунку.

Таким чином, реактивність Vue.js забезпечує декларативне програмування – розробник описує, який вигляд має інтерфейс у певному стані, а система сама піклується про оновлення DOM, щойно стан змінюється. Це знижує складність коду, усуває необхідність прямої маніпуляції з DOM і забезпечує більшу узгодженість між даними та їхнім поданням.

Однією з ключових архітектурних особливостей фреймворка Vue.js є компонентна структура, що передбачає модульну організацію коду шляхом поділу інтерфейсу користувача на окремі, ізольовані функціональні одиниці-компоненти. Кожен компонент у Vue.js являє собою самодостатню сутність, яка інкапсулює шаблон (структуру HTML), логіку (JavaScript) та стилі (CSS), пов'язані з певною частиною інтерфейсу.

Завдяки такому підходу, розробка застосунків стає значно гнучкішою та масштабованішою. Компоненти можуть бути багаторазово використані в межах одного проєкту, легко тестуватися окремо, а також підтримуватися

незалежно один від одного. Це підвищує читаємість коду, спрощує його обслуговування і дозволяє ефективно працювати над великими проектами в команді.

Компонентна модель у Vue.js реалізована за допомогою декларативного синтаксису, що дозволяє чітко описувати взаємозв'язки між компонентами, їхні властивості (props), внутрішній стан (data) та події (events). Крім того, Vue підтримує як локальну, так і глобальну реєстрацію компонентів, що забезпечує додаткову гнучкість при організації архітектури застосунку.

У підсумку, компонентна структура Vue.js сприяє побудові структурованих, підтримуваних і повторно використовуваних елементів інтерфейсу, що є важливим фактором ефективної розробки сучасних вебзастосунків.

Vue.js – це прогресивний JavaScript фреймворк, який дозволяє легко інтегруватися в існуючі проекти. Одна з основних переваг Vue полягає в тому, що він може бути інтегрований поетапно, без необхідності повної перебудови або переписування наявного коду. Це робить його ідеальним вибором для поступових змін у старих проектах, де не хочеться або неможливо повністю переходити на нові технології.

Vue.js має не тільки гнучкий і простий інтерфейс для розробки інтерфейсів користувача, але й потужну екосистему, яка включає ряд офіційних бібліотек для забезпечення різноманітних функціональних можливостей. Це робить Vue ще більш привабливим для розробників, оскільки ви отримуєте не тільки базовий фреймворк для побудови інтерфейсів, але й набір інструментів для вирішення складних задач, таких як маршрутизація, керування станом, а також розширена функціональність.

Vue.js знаходить широке застосування у створенні інтерфейсів для вебзастосунків, мобільних додатків (за допомогою NativeScript або Quasar) та навіть у проектах, де необхідна швидка розробка прототипів. Фреймворк активно використовують компанії як: Alibaba, Xiaomi, Behance, Grammarly, що підтверджує його надійність і гнучкість у комерційному середовищі.

2.3 Бібліотека React

React (або React.js) – це високопродуктивна JavaScript-бібліотека з відкритим кодом, розроблена компанією Meta (раніше Facebook) для створення динамічних інтерфейсів користувача [29, 30, 32, 34]. React забезпечує ефективну, гнучку та масштабовану розробку односторінкових вебзастосунків (SPA), де інтерфейс має постійно реагувати на зміну стану без необхідності повного перезавантаження сторінки.

На відміну від повнофункціональних фреймворків, React фокусується виключно на представницькому рівні (view) у моделі архітектури MVC (Model-View-Controller), що забезпечує вузьку спеціалізацію, високу продуктивність та простоту інтеграції з іншими інструментами.

Основна ідея React полягає в побудові інтерфейсу користувача як сукупності незалежних, повторно використовуваних компонентів, що дозволяє значно спростити організацію коду, покращити його структуру та забезпечити гнучкість у процесі подальшого масштабування проекту.

Однією з центральних технологічних особливостей React є використання віртуального DOM. Традиційна модель взаємодії з DOM у браузерах передбачає прями й часті маніпуляції з деревом елементів, що може призводити до значного навантаження на систему при оновленні великої кількості елементів. React, натомість, створює віртуальне представлення DOM у пам'яті, яке є копією реального DOM, але значно легшим і швидшим в обробці. Під час зміни стану застосунку React оновлює спершу Virtual DOM, після чого порівнює його з попередньою версією, виявляє відмінності (diffing algorithm) і вносить лише ті зміни, які є справді необхідними. Такий підхід значно підвищує ефективність рендерингу та зменшує кількість операцій над реальним DOM, що позитивно впливає на швидкодію застосунку, особливо в складних інтерфейсах з великою кількістю інтерактивних елементів.

Ще однією фундаментальною рисою React є компонентний підхід до побудови інтерфейсу. Компонент у React – це автономна одиниця, яка

інкапсулює в собі логіку та візуальне представлення певної частини користувацького інтерфейсу. Компоненти можуть мати власний стан, приймати вхідні параметри (props) та бути вкладеними один в одного, формуючи ієрархічну структуру всього інтерфейсу. Завдяки цьому підходу розробник має можливість створювати повторно використовувані блоки, які легко підтримувати, тестувати та вдосконалювати. Поділ логіки на компоненти забезпечує більш чисту архітектуру коду та зменшує кількість дублікацій, що особливо важливо в довгостроковій перспективі.

Крім того, React дотримується концепції одностороннього потоку даних. Це означає, що дані в системі передаються в одному напрямку – зверху вниз, від батьківських компонентів до дочірніх. Такий механізм передачі даних сприяє прозорості взаємодії між компонентами, дозволяє чітко контролювати зміни стану та забезпечує передбачуваність поведінки застосунку.

У випадку, коли необхідно організувати двосторонню взаємодію, React надає інструменти для підняття зворотного зв'язку (callback-функції), що викликаються дочірніми компонентами для впливу на батьківські. Це гарантує структурованість логіки обробки даних і значно знижує ймовірність виникнення помилок через неконтрольовану зміну стану.

Окремої уваги заслуговує синтаксичне розширення JSX (JavaScript XML), що є невід'ємною частиною розробки на React. JSX дозволяє писати розмітку, схожу на HTML, безпосередньо всередині JavaScript-коду. Це робить код більш читабельним і декларативним, оскільки структура інтерфейсу стає очевидною вже на етапі перегляду коду. JSX компілюється у звичайні JavaScript-функції, які створюють елементи React, тому він поєднує зручність розмітки з потужністю мовного середовища. Такий підхід також спрощує інтеграцію логіки та представлення, дозволяючи уникати надмірного поділу між шаблонами та скриптами.

Узагальнюючи вищезазначене, можна зробити висновок, що React є гнучким і потужним інструментом для створення сучасних вебзастосунків. Його архітектурні рішення, такі як віртуальний DOM, компонентна модель,

односторонній потік даних і JSX, не лише сприяють ефективній реалізації складних інтерфейсів, а й покращують досвід розробки загалом. Ці особливості роблять React надзвичайно популярним серед розробників усього світу та активно використовуваним у комерційних і наукових проектах.

2.4 Ember.js

Ember.js є потужним фреймворком JavaScript, призначеним для створення амбітних вебзастосунків із багатим функціоналом [35-41]. Він був розроблений як рішення для побудови масштабованих односторінкових застосунків (SPA), у яких важлива висока продуктивність, чітка структура та розширюваність. На відміну від легших бібліотек, Ember пропонує повний набір інструментів «із коробки», що дозволяє розробникам зосередитися безпосередньо на бізнес-логіці застосунку, не витрачаючи час на інтеграцію сторонніх компонентів або побудову інфраструктури вручну.

Однією з ключових концепцій Ember є дотримання принципу «конвенція понад конфігурацію» (convention over configuration). Це означає, що фреймворк має заздалегідь визначені правила і шаблони, яких має дотримуватися розробник. Завдяки цьому значно зменшується кількість ручного налаштування, а розробка стає швидшою та послідовнішою. Розробникам не потрібно вирішувати, як організовувати структуру проекту – Ember надає чіткі рекомендації та інструменти, що забезпечують єдиний підхід до побудови застосунку.

Важливим елементом архітектури Ember є його потужна система маршрутизації. У фреймворку маршрутизатор не просто відповідає за відображення різних сторінок – він відіграє центральну роль у керуванні станами застосунку. Кожен маршрут пов'язаний з конкретним шаблоном, контролером і модельними даними, що дозволяє реалізовувати складну логіку

переходів та обробки даних у межах чітко визначеної структури. Такий підхід полегшує налагодження застосунків і сприяє їхній модульності.

Компонентна модель у Ember також розвинена, проте на відміну від React або Vue, вона тісно інтегрована з шаблонами Handlebars, які використовуються для опису інтерфейсу, що забезпечує високий рівень декларативності: логіка взаємодії з користувачем описується через прості шаблонні конструкції, що значно полегшує підтримку та читання коду. Водночас Ember підтримує повторне використання компонентів, що дозволяє створювати гнучкі та масштабовані інтерфейси.

Ember CLI – офіційний інструмент командного рядка – є ще однією сильною стороною фреймворку. Він забезпечує інтегроване середовище для генерації коду, тестування, розгортання та управління залежностями. Ember CLI підтримує модульну структуру, автоматичне оновлення проєктів, генерацію типових елементів застосунку (наприклад, контролерів, моделей, маршрутів) і вбудовану систему тестування, що робить його незамінним інструментом для розробників, які прагнуть до високої продуктивності та надійності.

Ще однією перевагою Ember є його зріла екосистема. Завдяки активному розвитку та підтримці спільноти, а також суворій регламентованості структури, фреймворк забезпечує стабільність і передбачуваність, що є критично важливим для корпоративних та довготривалих проєктів. Ember орієнтований на довгострокову підтримку та поступову еволюцію, завдяки чому розробники можуть бути впевнені у майбутній сумісності своїх застосунків із новими версіями.

Загалом, Ember.js – це фреймворк, що найкраще підходить для великих застосунків з чітко визначеною архітектурою. Його особливості, такі як принципи конвенції, централізована маршрутизація, потужний CLI, декларативні шаблони та компонентна модель, роблять його надійним вибором для команд, які прагнуть до структурованої, стабільної та масштабованої розробки.

2.5 D3.js

У світі, де дані відіграють ключову роль у прийнятті рішень, вміння правильно їх візуалізувати – не менш важливе, ніж їх збирання чи аналіз. Саме тут на допомогу приходять D3.js (Data-Driven Documents) – одна з найпотужніших бібліотек JavaScript для візуалізації даних у вебсередовищі.

D3.js – це відкрита JavaScript-бібліотека, яка дозволяє створювати інтерактивні, динамічні графіки та візуалізації, використовуючи сучасні вебтехнології: SVG, HTML та CSS. Вона надає гнучкий інструментарій для роботи з даними – від простої прив'язки значень до елементів DOM до створення складних інтерактивних інфографік [39-41].

D3.js не просто будує графіки – вона працює на рівні нижче за більшість інших бібліотек (наприклад, Chart.js чи Google Charts), даючи розробнику повний контроль над тим, як саме буде виглядати і поводитись візуалізація.

Алгоритм роботи з D3 виглядає приблизно так:

Крок 1. Завантажити дані.

Крок 2. Прив'язати ці дані до DOM-елементів.

Крок 3. Встановити атрибути (висота, ширина, колір, положення тощо) відповідно до значень даних.

Крок 4. Додати анімації чи обробку подій.

Переваги D3.js:

– гнучкість – повний контроль над кожним пікселем. Однією з головних переваг D3.js є його максимальна гнучкість. На відміну від багатьох бібліотек візуалізації, які надають готові компоненти типу «стовпчиковий графік» чи «лінійна діаграма», D3 дозволяє створювати візуалізацію з нуля, використовуючи SVG, HTML або навіть Canvas. Це означає, що розробник має повний контроль над зовнішнім виглядом: колір, розмір, форма, шрифт, позиція – усе налаштовується вручну або на основі даних;

– інтерактивність – підтримка подій, анімацій, реактивних змін. D3.js створений для того, щоб дані були живими. Він підтримує:

- 1) обробку подій (наприклад, кліки, наведення, скрол);
- 2) анімації та переходи (smooth transitions);
- 3) оновлення в режимі реального часу (наприклад, при зміні даних або розміру вікна).

Все це дозволяє створювати справжню взаємодію з користувачем. Ви можете реалізувати фільтрацію, вибір діапазону, додавання підказок (tooltip), зміни кольору при наведенні тощо;

– інтеграція – легко поєднується з іншими бібліотеками (React, Vue, Angular). Хоча D3 сам по собі є потужним інструментом, він також добре інтегрується з фреймворками, такими як:

- 1) React – можна використовувати D3 разом з React-компонентами, зберігаючи керування станом у React, а рендеринг – через D3;
- 2) Vue – гнучкість D3 дозволяє працювати всередині Vue-компонентів, наприклад, в lifecycle-хуках (mounted);
- 3) Angular – D3 добре працює в Angular-модулі як стороння бібліотека для побудови кастомної візуалізації.

Це дає змогу використовувати D3 як «візуальний движок» у великих проєктах із сучасною архітектурою;

– потужність – підходить для великих проєктів з великою кількістю даних. D3.js є не лише інструментом для створення графічних візуалізацій у вебсередовищі, а й потужною платформою для попередньої обробки, трансформації та аналізу даних безпосередньо в браузері.

Однією з ключових особливостей бібліотеки є її здатність ефективно працювати з масштабами. Вбудовані функції масштабування дозволяють трансформувати числові, часові або категоріальні дані у відповідні координати, адаптуючи їх до доступного простору візуалізації, що дає змогу точно й коректно розміщувати графічні елементи в межах вікна перегляду, що особливо важливо при побудові складних багатовимірних візуалізацій.

Крім того, D3.js підтримує обробку великих масивів даних завдяки ефективній реалізації прив'язки даних до елементів DOM. Цей механізм забезпечує високу продуктивність навіть у випадках, коли обсяг даних є значним. Динамічне оновлення візуалізації здійснюється без необхідності повного перерендерингу елементів, що дозволяє створювати інтерактивні та чутливі до змін даних інтерфейси.

Окремої уваги заслуговують вбудовані засоби математичної та статистичної обробки. Бібліотека надає інструменти для виконання типових статистичних операцій, таких як обчислення середнього значення, медіани, агрегування даних, а також реалізацію більш складних підходів, включно з побудовою лінійної регресії, кривих Безьє, генерацією контурів і областей, що значно розширює можливості D3.js і дозволяє використовувати її не лише як інструмент для презентації результатів аналізу, а й як частину самого процесу аналізу даних у клієнтському середовищі.

3 ОПИС ПРОГРАМНОЇ РЕАЛІЗАЦІЇ

3.1 Основні принципи розробки вебзастосунків

Основні принципи розробки вебзастосунків визначають якість, ефективність та зручність використання створеного програмного продукту. Дотримання цих принципів є критично важливим для досягнення стабільної роботи, безпеки, масштабованості та позитивного користувацького досвіду.

Одним із ключових принципів є розділення логіки застосунку, що передбачає чітке розмежування між клієнтською (front-end) та серверною (back-end) частинами (рис. 3.1), що забезпечує кращу підтримуваність коду, сприяє повторному використанню компонентів і спрощує тестування [32, 40].

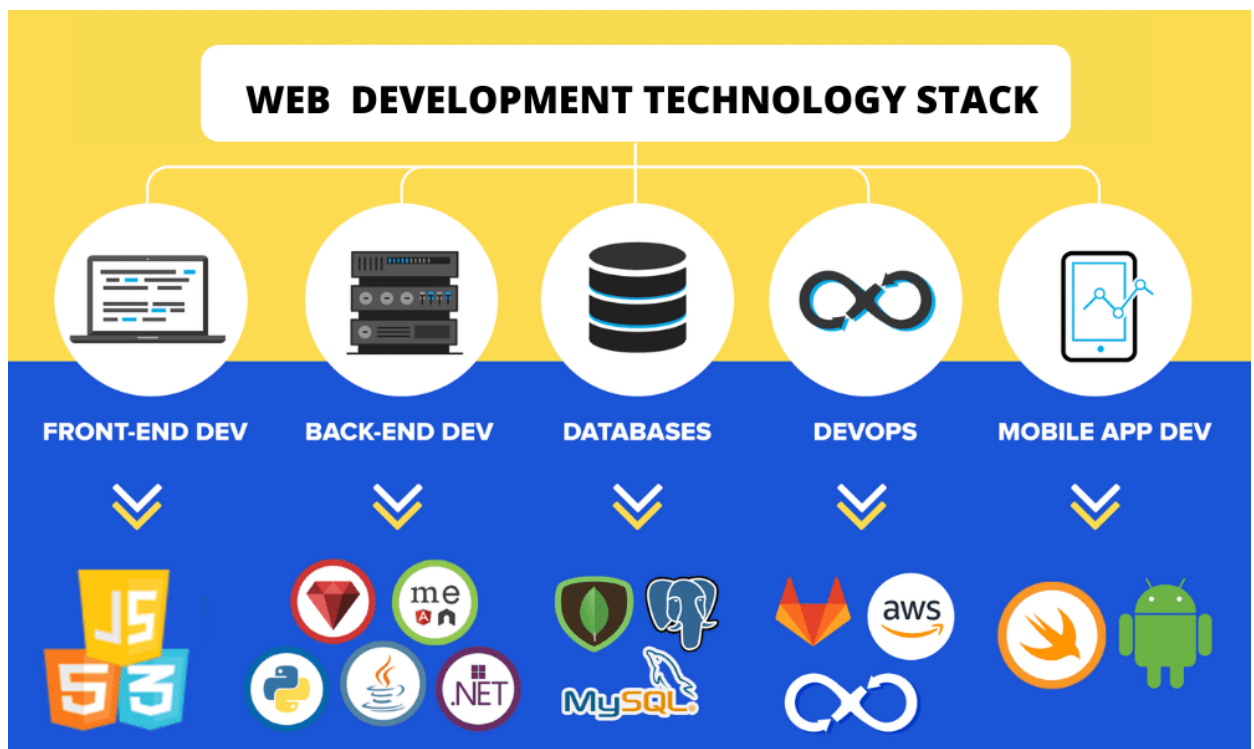


Рисунок 3.1 – Технологія розробки вебзастосунків

Front-end (клієнтська частина вебзастосунку) складається з компонентів і технологій, які відповідають за відображення, взаємодію з користувачем та обробку подій у веббраузері. Основними складовими фронтенду є:

– HTML (HyperText Markup Language) – мова розмітки, що визначає структуру вебсторінки. Вона описує, які елементи повинні бути присутні на сторінці: заголовки, абзаци, кнопки, форми, таблиці тощо;

– CSS (Cascading Style Sheets) – мова стилів, яка відповідає за зовнішній вигляд HTML-елементів: кольори, шрифти, розміри, позиціонування, адаптивність та анімації. CSS дозволяє створювати візуально привабливі інтерфейси;

– JavaScript – мова програмування, що надає вебсторінці динамічність та інтерактивність. Завдяки JavaScript можливо реалізувати реакцію на дії користувача (кліки, введення, навігація), оновлення вмісту без перезавантаження сторінки, валідацію форм тощо;

– фреймворки та бібліотеки JavaScript – це інструменти, які спрощують розробку фронтенду;

– шаблони та компоненти UI – системи, що допомагають формувати інтерфейс з готових елементів (наприклад, кнопок, форм, модальних вікон). До них належать Bootstrap, Tailwind CSS, Material UI тощо;

– інструменти для збірки та оптимізації такі як Webpack, Vite, Babel або Parcel, що автоматизують обробку коду, мінімізують файли, трансформують сучасний JavaScript у сумісний із браузерами, та пришвидшують завантаження;

– API-клієнти – засоби для взаємодії з сервером. JavaScript (через fetch, axios тощо) дозволяє відправляти запити до серверного API та отримувати дані у форматі JSON або XML для відображення на сторінці.

Таким чином, front-end охоплює усі елементи взаємодії користувача з вебзастосунком: від структури і стилю до логіки поведінки та обміну даними з сервером.

Back-end (серверна частина вебзастосунку) – це частина системи, яка працює «за лаштунками» і відповідає за логіку, обробку запитів, зберігання та управління даними. Основне завдання бекенду – це обробка даних, що

надходять від користувача, взаємодія з базами даних та виконання серверної логіки, яку не видно безпосередньо на інтерфейсі.

Складові back-end:

– сервер – це комп'ютер або програма, яка приймає запити від користувачів (зазвичай через HTTP/HTTPS), обробляє їх і надсилає відповідь. Сервери можуть бути фізичними або віртуальними і можуть працювати в хмарних середовищах;

– програмне забезпечення серверної частини (серверні фреймворки та мови програмування) – бекенд-логіка, яка обробляє запити користувача, часто використовує певний фреймворк для створення серверних застосунків. Це можуть бути такі мови та фреймворки, як:

- 1) Node.js (JavaScript);
- 2) Python (Django, Flask);
- 3) Ruby (Ruby on Rails);
- 4) PHP (Laravel, Symfony);
- 5) Java (Spring);
- 6) C# (.NET);

– база даних – це система для зберігання та управління даними, які використовуються вебзастосунком. Дані можуть бути зберігані в реляційних (SQL) або нереляційних (NoSQL) базах даних. Наприклад: SQL (MySQL, PostgreSQL, SQLite), NoSQL (MongoDB, Redis, Cassandra);

– API (Application Programming Interface) – це набір інтерфейсів, який дозволяє різним частинам програми чи іншим програмам взаємодіяти між собою. API в бекенді дозволяє взаємодіяти з клієнтською частиною (front-end) через HTTP запити (наприклад, через REST або GraphQL API);

– логіка бізнес-процесів – бекенд реалізує основну логіку застосунку, яка включає роботу з даними, виконання алгоритмів, обробку запитів користувача та забезпечення правильності виконання операцій на сервері;

- аутентифікація та авторизація – забезпечення безпеки вебзастосунку. Аутентифікація перевіряє особу користувача (зазвичай через логін та пароль), а авторизація визначає, які дії він може виконувати в системі;
- підключення до зовнішніх сервісів – бекенд може взаємодіяти з іншими системами або сервісами, такими як платіжні шлюзи, зовнішні API, хмарні сервіси, для реалізації додаткової функціональності;
- моніторинг і логування – системи моніторингу допомагають відстежувати стан серверу, виявляти помилки, аналізувати ефективність роботи вебзастосунку.

Загалом, back-end є основою будь-якого динамічного вебзастосунку, де зберігається вся важлива логіка і дані, а також забезпечується взаємодія між сервером та клієнтською частиною.

Також важливо дотримуватись принципу модульності, тобто організувати код у вигляді незалежних функціональних блоків, що можуть працювати окремо та легко масштабуватись.

Іншим важливим аспектом є інтуїтивний та адаптивний інтерфейс, який забезпечує комфортну взаємодію з користувачем незалежно від типу пристрою. Адаптивність означає, що вебзастосунок повинен коректно відображатися на різних розмірах екранів, зокрема мобільних.

Безпека – ще один фундаментальний принцип. Необхідно впроваджувати захист від поширених атак, таких як XSS, CSRF, SQL-ін'єкції, та забезпечувати безпечну авторизацію й аутентифікацію користувачів.

Також важливим є принцип масштабованості, який полягає в здатності системи витримувати збільшення навантаження без втрати продуктивності, що досягається завдяки оптимальній архітектурі, ефективному використанню кешування, балансуванню навантаження та горизонтальному масштабуванню серверів.

Крім того, сучасна розробка орієнтована на реактивність інтерфейсу, тобто здатність системи швидко реагувати на дії користувача, оновлюючи інформацію в реальному часі без повного перезавантаження сторінки. Це

забезпечується завдяки використанню JavaScript-фреймворків, таких як React, Vue або Angular.

Усі ці принципи формують цілісну методологію розробки, яка дозволяє створювати ефективні, надійні та зручні для користувачів вебзастосунки.

3.2 Приклади стеків технологій веброзробки

У сучасній веброзробці використання стеків технологій дозволяє створювати ефективні, масштабовані та зручні у підтримці вебзастосунки. Стек технологій – це набір інструментів, мов програмування, фреймворків та бібліотек, які працюють разом для розробки клієнтської та серверної частин застосунку. Існує багато різноманітних стеків, однак найбільш поширеними є ті, що поєднують сучасні фронтенд- та бекенд-технології, забезпечуючи повний цикл розробки.

Одним із найвідоміших є MERN стек (рис. 3.2), до якого входять:

- MongoDB;
- Express.js;
- React;
- Node.js.

MongoDB виступає як нереляційна база даних, що дозволяє зберігати документоорієнтовані дані у форматі JSON.

Express.js – це мінімалістичний серверний фреймворк для Node.js, який забезпечує ефективну маршрутизацію та обробку HTTP-запитів.

React використовується для побудови інтерфейсу користувача, завдяки компонентному підходу та реактивному оновленню даних.

Усі ці технології базуються на JavaScript, що спрощує розробку завдяки єдиній мові для обох частин застосунку – клієнтської та серверної.

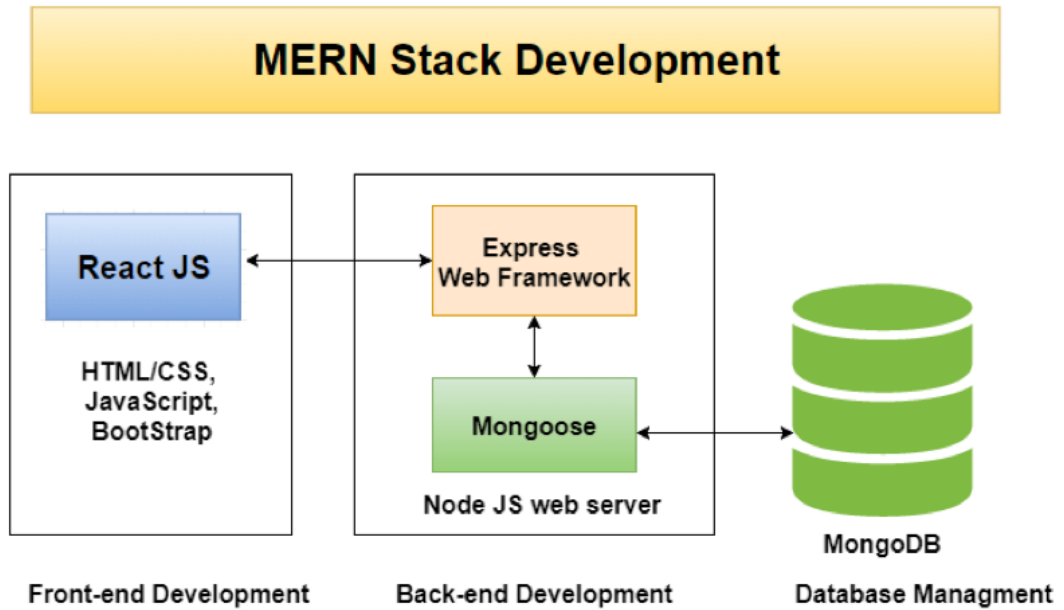


Рисунок 3.2 – MERN стек

Іншим поширеним стеком є MEAN (рис. 3.3), який аналогічний MERN, але замість React використовує Angular – повноцінний фронтенд-фреймворк від Google.

Angular забезпечує строгішу архітектуру та більший рівень абстракції, що робить його зручним для побудови великих корпоративних застосунків.

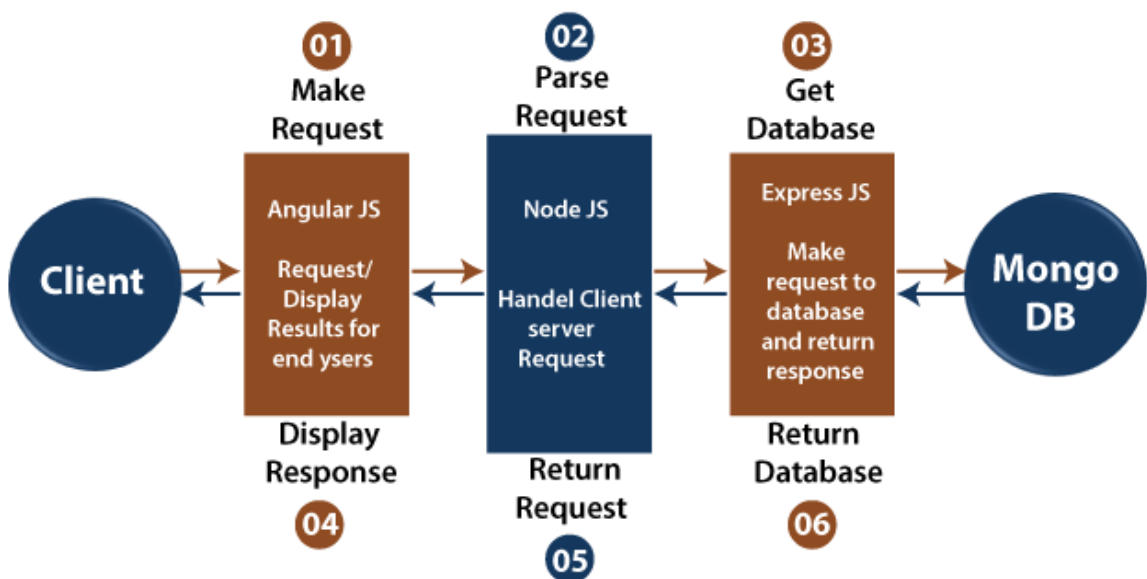


Рисунок 3.3 – MEAN стек

Ще одним прикладом є LAMP стек (рис. 3.4), що об'єднує Linux як операційну систему, Apache як вебсервер, MySQL як систему керування базами даних і PHP як серверну мову програмування. Цей стек є класичним рішенням у веброботці, яке активно використовується для побудови динамічних вебсайтів і CMS-систем.

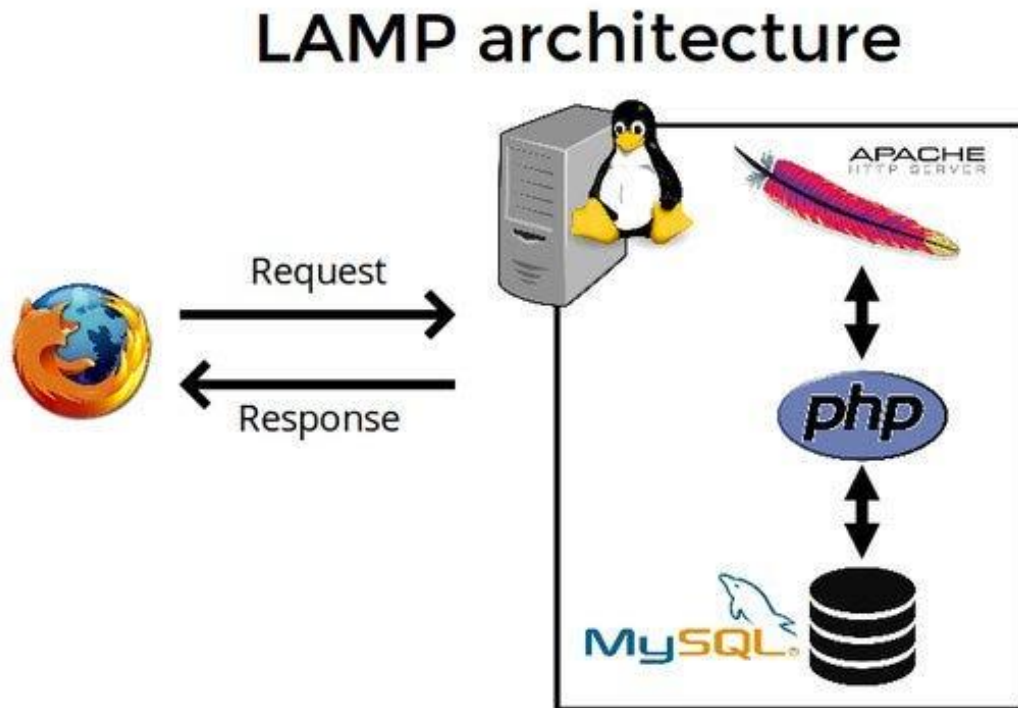


Рисунок 3.4 – LAMP стек

Альтернативою є LEMP стек, який замість Apache використовує вебсервер Nginx. Завдяки більшій продуктивності та здатності обробляти високе навантаження, LEMP вважається ефективнішим рішенням для масштабованих проєктів.

Також варто відзначити JAMstack – сучасний підхід до розробки статичних або напівдинамічних вебзастосунків, у якому акцент робиться на JavaScript, API та попередньо згенерованому HTML (Markup) (рис. 3.5). Цей стек дозволяє досягати високої швидкості завантаження, безпеки та простоти розгортання завдяки використанню серверів CDN та хмарних сервісів.

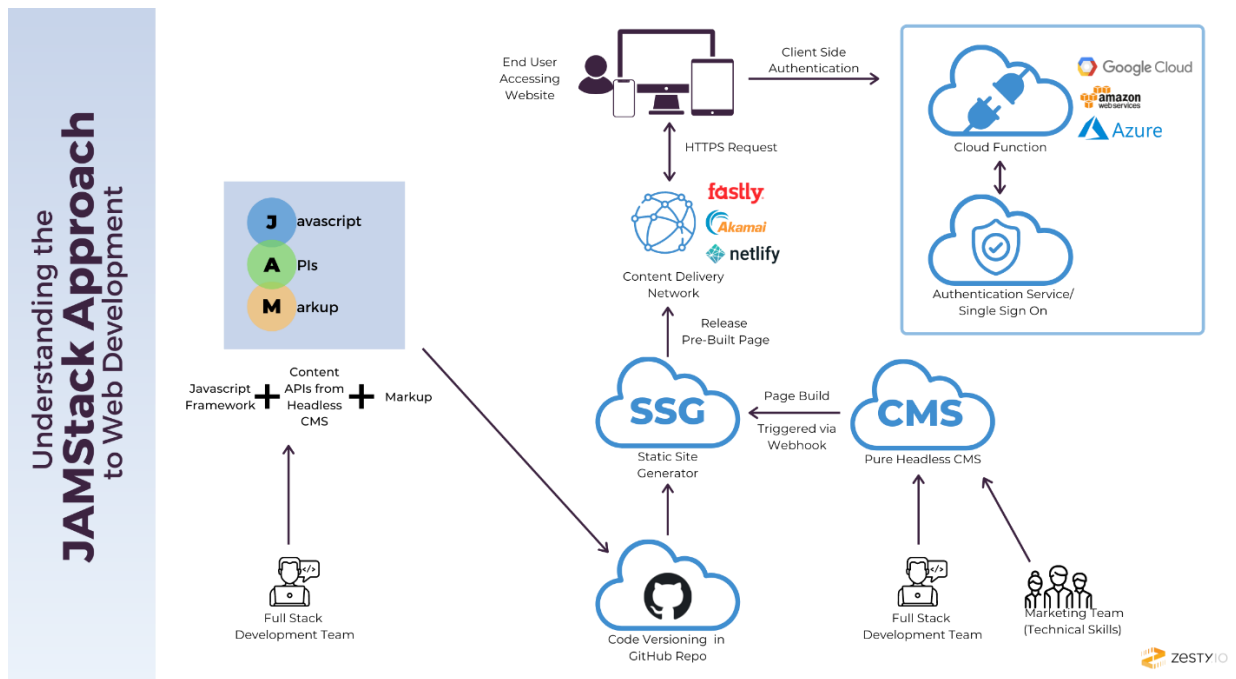


Рисунок 3.5 – JAMstack

Вибір стеку технологій залежить від типу проєкту, вимог до продуктивності, досвіду команди та очікуваних функціональних можливостей.

Сучасні стекові підходи забезпечують високу гнучкість, масштабованість та підтримку новітніх архітектур, таких як мікросервіси, SPA (Single Page Application) або PWA (Progressive Web Application). Завдяки цьому вони є основою успішної розробки вебзастосунків у різних галузях.

3.3 Обґрунтування вибору засобів розробки вебзастосунку

Для розробки вебзастосунку було обрано MERN Stack (рис. 3.6) [42-44], до складу якого входять MongoDB, Express.js, React.js та Node.js. Цей стек є одним із найпопулярніших у сучасній веброзробці, оскільки поєднує в собі потужність, гнучкість і високу продуктивність. Його перевагою є використання єдиної мови програмування – JavaScript – як на клієнтській, так і на серверній частині.

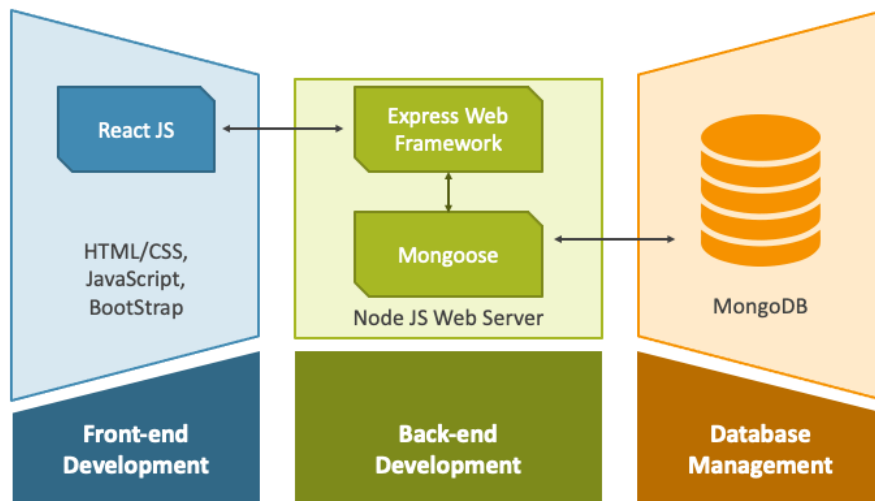


Рисунок 3.6 – Розробка вебзастосунку за допомогою MERN Stack

MongoDB виступає як документоорієнтована база даних NoSQL, яка дозволяє гнучко зберігати структури даних у форматі JSON-подібних документів. Це забезпечує високу масштабованість і швидкість доступу до інформації, що є особливо важливим для динамічних вебзастосунків.

Express.js виконує роль серверного фреймворку для Node.js і відповідає за обробку HTTP-запитів, маршрутизацію, керування middleware-компонентами та реалізацію API. Завдяки своїй простоті та розширюваності Express дозволяє створювати надійні серверні частини з чіткою логікою обробки даних.

React.js застосовується для побудови інтерфейсу користувача. Його компонентна структура дозволяє повторно використовувати елементи, ефективно керувати станом застосунку та динамічно оновлювати вміст без перезавантаження сторінки. Крім того, React має широкую екосистему та підтримку з боку спільноти, що робить його зручним інструментом для створення інтерактивних SPA (Single Page Application).

Node.js виступає як середовище виконання JavaScript на сервері. Воно забезпечує неблокуючу, подієво-орієнтовану модель обробки запитів, що робить його ідеальним для розробки високопродуктивних, масштабованих мережевих застосунків. Node.js також дозволяє ефективно працювати з

асинхронним кодом, що є перевагою при взаємодії з базами даних або зовнішніми API.

Загалом, обраний стек MERN забезпечує повноцінне середовище для розробки сучасних вебзастосунків із гнучкою архітектурою, високою швидкістю роботи та зручною інтеграцією між клієнтською та серверною частинами. Його використання сприятиме реалізації функціонального, масштабованого та зручного у підтримці програмного продукту.

3.4 Програмна реалізація

Технічне завдання має на меті розробку онлайн-платформи для надання інформації щодо ментального здоров'я. Основною метою цього вебзастосунку є створення зручного, інформативного та безпечного середовища, де користувачі зможуть отримувати достовірну інформацію про психоемоційний стан, психологічну підтримку, способи самопомоги, контакти спеціалістів та інші ресурси, спрямовані на покращення ментального здоров'я.

Розроблений вебзастосунок покликаний забезпечити повноцінну взаємодію користувача з інформаційним контентом, а також надати персоналізовані можливості для керування обліковим записом та взаємодії з системою. З цією метою передбачено реалізацію наступного функціонального набору:

- автентифікація та реєстрація користувача. Платформа передбачає систему реєстрації нових користувачів із валідацією введених даних (електронна пошта, пароль тощо). Після створення облікового запису користувач може виконати вхід у систему (автентифікацію) для отримання доступу до персоналізованих функцій. Передбачено реалізацію захищеного зберігання паролів за допомогою хешування;

- редагування профілю. Зареєстрований користувач має можливість переглядати та змінювати власні особисті дані: ім'я, аватар, контактну

інформацію тощо. Цей функціонал спрямований на персоналізацію досвіду користування застосунком;

– додавання, редагування та видалення статей, їх публікація. У вебзастосунку необхідно реалізувати модуль створення авторського контенту. Користувачі, які мають відповідні права (наприклад, автор або адміністратор), можуть створювати нові статті, редагувати вже опубліковані матеріали або видаляти їх. Після написання статті вона може бути збережена як чернетка або опублікована у відкритому доступі;

– фільтрація статей за категоріями та авторами. Для зручної навігації в інформаційному розділі передбачено систему категоризації контенту. Користувач може здійснювати пошук або фільтрацію статей за темами (наприклад, тривожність, депресія, самопомога) або за іменем автора, що сприяє швидкому доступу до релевантної інформації;

– проходження тесту. Окремий модуль дозволяє користувачам проходити психологічні тести, розроблені для оцінки емоційного стану, рівня стресу, схильності до тривожних розладів тощо. Після завершення тестування користувач отримує короткий результат або рекомендації на основі відповідей. Результати можуть зберігатися в обліковому записі для подальшого перегляду (опціонально).

Таким чином, реалізовані функціональні можливості створюють інтерактивне, персоналізоване та інформативне середовище для користувачів, що відповідає загальній меті платформи – підтримці ментального здоров'я шляхом надання якісного цифрового контенту та інструментів самодіагностики.

Дизайн вебзастосунку відіграє ключову роль у забезпеченні зручності взаємодії користувача з системою, формуванні позитивного досвіду використання та підвищенні загальної ефективності сприйняття інформаційного контенту. У розробці інтерфейсу було враховано сучасні принципи UX/UI-дизайну, орієнтовані на простоту, естетику та функціональність.

У верхній частині інтерфейсу розміщується навігаційна панель, яка забезпечує швидкий доступ до основних розділів вебзастосунку, таких як головна сторінка, розділ зі статтями, тестування, профіль користувача, контактна інформація тощо. Навігація реалізована таким чином, щоб користувач міг легко орієнтуватися в межах платформи без необхідності багаторівневих переходів (рис. 3.7).



Рисунок 3.7 – Головна сторінка вебзастосунку

На навігаційній панелі користувач має змогу швидко орієнтуватися між основними функціональними розділами вебзастосунку.

Інтерфейс панелі спроектовано таким чином, щоб забезпечити інтуїтивно зрозумілий доступ до ключових можливостей платформи. Зокрема, на панелі розміщено такі розділи:

- «TEST» – перехід до модулю проходження психологічних тестів. У цьому розділі користувач може пройти інтерактивні опитувальники, спрямовані на самодіагностику ментального стану. Після завершення тесту система генерує інтерпретацію результату або рекомендації, що дозволяє користувачу краще зрозуміти власний емоційний стан;

- «BLOG» – перехід до інформаційного блогу. У цьому розділі відображаються всі опубліковані статті, розділені за категоріями або авторами. Користувач має можливість ознайомлюватися з матеріалами,

залишати коментарі (за наявності авторизації), а також здійснювати пошук за ключовими словами або тематиками;

– «CREATE» – функціональний розділ для створення нового контенту. Доступ до цього розділу мають користувачі з відповідними правами (наприклад, автори чи адміністратори). Інтерфейс дозволяє створювати нові публікації, додавати заголовки, текст, зображення, обирати категорії, а також зберігати статті як чернетки або публікувати їх безпосередньо на платформі.

Незареєстрований користувач має змогу переглядати блог (рис. 3.8), статті, а також проходити тест. Для публікації власної статті необхідно зареєструватися.

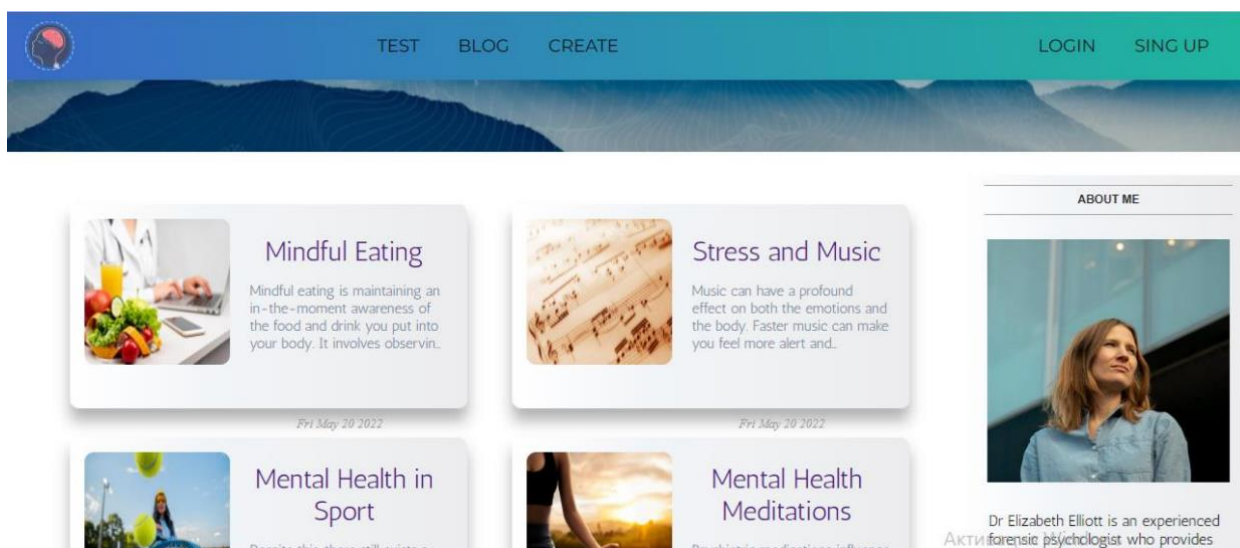


Рисунок 3.8 – Вкладка «BLOG»

Для перегляду повного вмісту статті користувачеві необхідно натиснути на відповідну публікацію у загальному списку блогу або результатах фільтрації. Після натискання система виконує перенаправлення на окрему сторінку, присвячену конкретній статті.

На цій деталізованій сторінці (рис. 3.9) статті відображається повний текстовий вміст матеріалу з урахуванням форматування, яке забезпечує зручне читання: з абзацами, заголовками, списками або ілюстраціями (за наявності). Крім самого контенту, на сторінці додатково виводиться інформація про

автора статті, що дає змогу користувачам дізнатися, ким створено матеріал, а також дата та час публікації, що дозволяє орієнтуватися в актуальності поданої інформації.

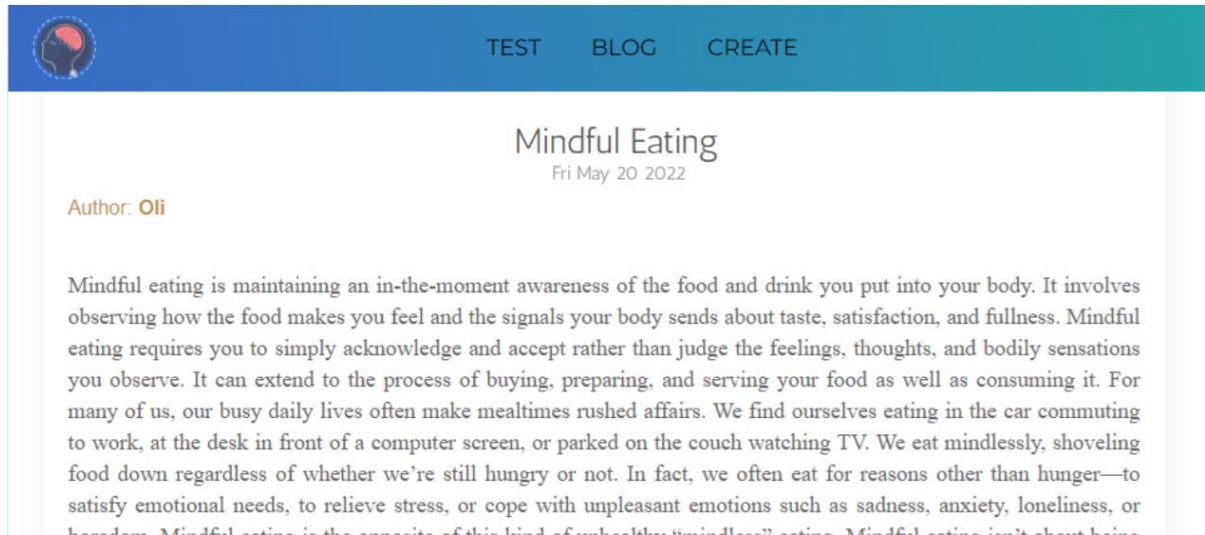


Рисунок 3.9 – Сторінка статті

Такий підхід до реалізації навігації між контентом забезпечує інтуїтивно зрозумілу взаємодію користувача з платформою та сприяє покращенню загального досвіду користування. Завдяки перетворенню імені автора на активне посилання користувачі можуть легко ознайомлюватися з іншими матеріалами, створеними тією ж особою, що особливо актуально у випадках, коли певний автор демонструє експертність або високу якість викладення інформації.

Це не лише сприяє глибшому залученню аудиторії, а й формує більш персоналізований підхід до споживання інформаційного контенту. Авторські сторінки можуть містити коротку біографію, тематику публікацій, рейтинг автора або відгуки, що додатково стимулює довіру до контенту та підтримує взаємодію в межах платформи. Таким чином, реалізація активного посилання на ім'я автора виконує не лише навігаційну, а й інформаційно-комунікативну функцію, підвищуючи як ефективність пошуку релевантних матеріалів, так і загальний рівень користувацького задоволення.

Після натискання відкривається окрема сторінка з переліком усіх статей, написаних цим автором (рис. 3.10). Матеріали подаються у вигляді карток або списку з короткими анотаціями (заголовок, вступ, дата публікації), що дозволяє швидко ознайомитися зі змістом і перейти до повної версії за потреби. Додатково може бути відображена коротка інформація про автора, така як його аватар, ім'я та біографічна довідка.

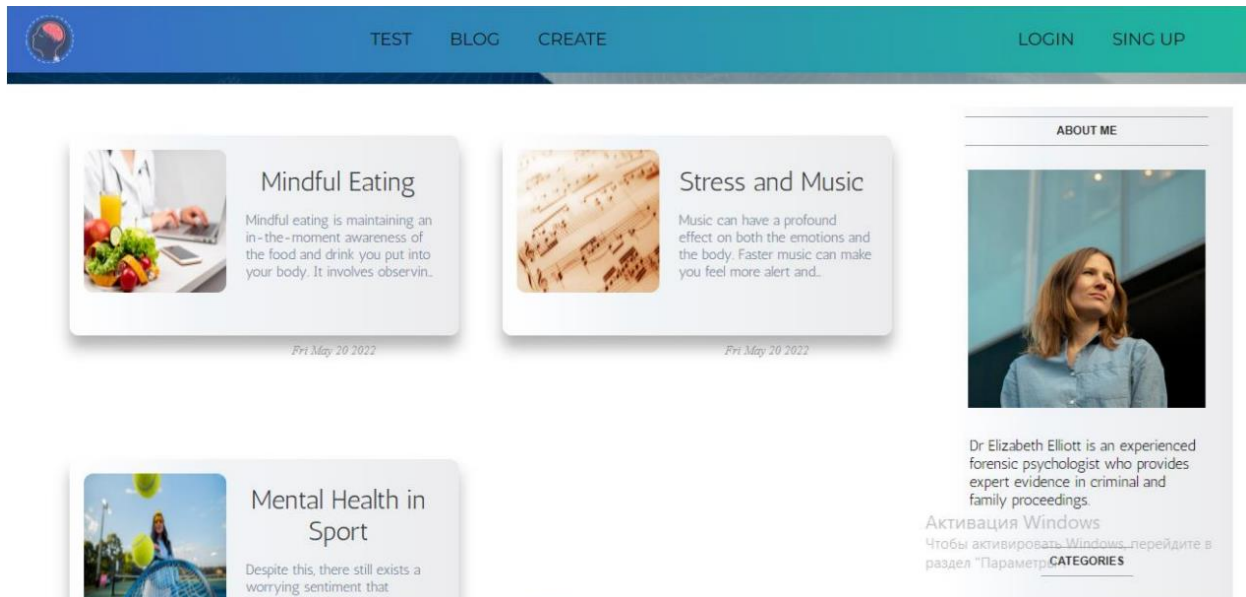


Рисунок 3.10 – Статті обраного автора

Такий механізм полегшує навігацію між матеріалами конкретного автора, дає змогу користувачам стежити за його публікаціями, формує персоналізовану стрічку контенту та сприяє підвищенню залученості до платформи.

Такий підхід забезпечує структуровану подачу контенту, розділяючи короткий анонс у списку публікацій та повну версію статті на окремій сторінці. Це сприяє кращій організації інтерфейсу, покращує взаємодію з користувачем і дозволяє кожній публікації мати власний постійний шлях доступу, що є важливим для навігації, поширення та збереження матеріалів.

Таким чином, навігаційна панель виконує роль основного інструменту перемикання між функціональними частинами вебзастосунку, сприяючи

зручності, швидкості доступу та логічності структури взаємодії користувача з платформою.

Форми реєстрації та входу до системи мають просту та логічну структуру з підказками, що забезпечує легке заповнення. Всі форми оформлені відповідно до загального стилю платформи з використанням адаптивної верстки.

Таким чином, реалізація механізму реєстрації на платформі відповідає сучасним вимогам до зручності, функціональності та безпеки цифрових сервісів. Забезпечення чіткої, логічно побудованої структури інтерфейсу реєстрації дозволяє значно підвищити доступність ресурсу для користувачів з різним рівнем цифрової грамотності. Наявність візуально виокремленої кнопки «SIGN UP», розташованої у верхній частині навігаційної панелі, спрощує початкову взаємодію та орієнтацію в інтерфейсі, що є ключовим фактором у процесі залучення нових користувачів.

Форма реєстрації розроблена з урахуванням принципів UX/UI-дизайну, що включають простоту сприйняття, мінімізацію когнітивного навантаження та передбачуваність дій. Користувач отримує змогу швидко ввести необхідну інформацію без надмірної кількості кроків, а реалізовані механізми валідації забезпечують своєчасне виявлення помилок та надання зворотного зв'язку у зручній для сприйняття формі. Це, у свою чергу, сприяє зменшенню кількості відмов та перерваних спроб реєстрації.

Завдяки автоматичному перенаправленню до форми авторизації (рис. 3.11) або особистого кабінету після успішного створення облікового запису, забезпечується логічна послідовність етапів взаємодії, що знижує ризик дезорієнтації користувача та сприяє ефективному включенню до подальшого користування функціоналом платформи.

Такий підхід не лише оптимізує процес реєстрації, а й формує позитивне перше враження про сервіс, що особливо важливо для вебзастосунку, який має на меті створення безпечного та підтримуючого середовища для покращення ментального здоров'я.

Узгодженість, адаптивність та інтерактивність інтерфейсу в контексті процесу реєстрації свідчать про високий рівень уваги до потреб користувача, що є критично важливим чинником успішності цифрової платформи в сфері охорони психічного добробуту.

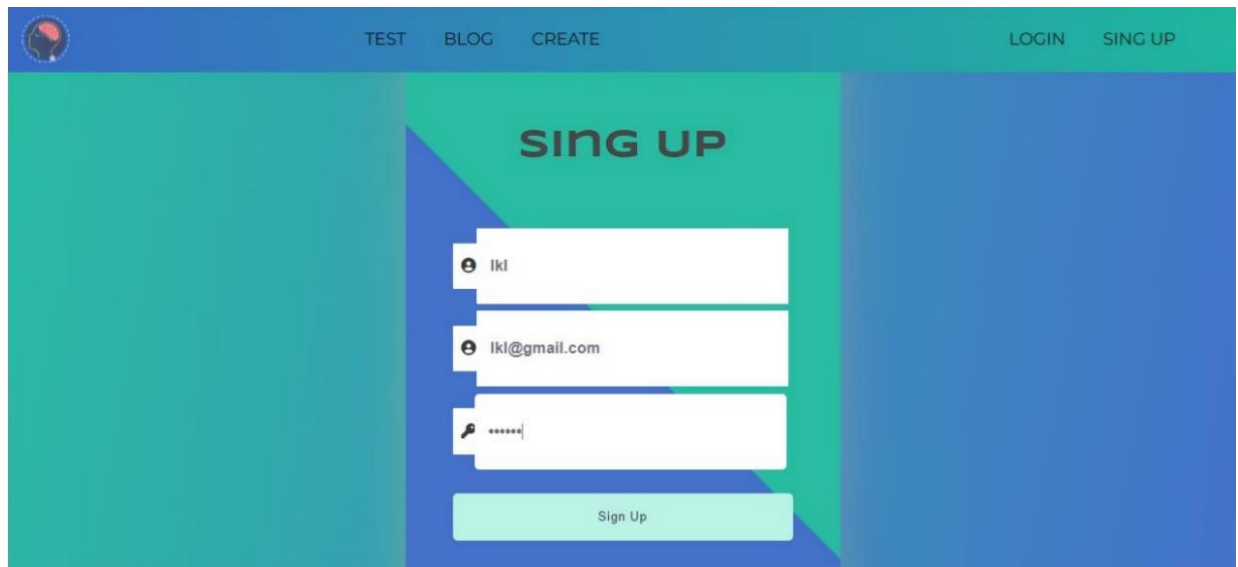
The image shows a web registration form with a teal and blue background. At the top, there are navigation links: 'TEST', 'BLOG', 'CREATE', 'LOGIN', and 'SING UP'. The main heading is 'SING UP'. Below it are three input fields: the first contains 'iki', the second contains 'iki@gmail.com', and the third is a password field with a lock icon and masked characters '*****'. A light green 'Sign Up' button is positioned at the bottom of the form.

Рисунок 3.11 – Форма для реєстрації

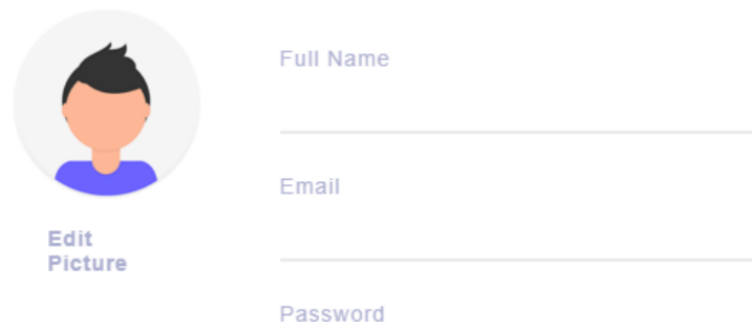
Після успішного завершення процесу реєстрації система автоматично перенаправляє користувача на головну сторінку платформи або в його персональний кабінет. Це забезпечує безперервність та зручність користувацького досвіду, оскільки користувач відразу потрапляє в місце, де може продовжити взаємодію з платформою. Таке автоматичне перенаправлення знижує кількість кроків, необхідних для переходу до основного контенту, і створює більш інтуїтивно зрозумілий та ефективний процес реєстрації. Для підвищення прозорості можна також передбачити відображення сповіщення чи анімації, щоб користувач знав про успішне завершення реєстрації перед тим, як його автоматично перенаправить система. Це дозволяє не тільки покращити зручність, але й забезпечити плавність переходів між етапами використання платформи.

У цьому розділі користувач може продовжити взаємодію із платформою в розширеному функціональному середовищі, зокрема отримати доступ до

персоналізованого контенту, проходити психологічні тести, переглядати результати тестування, редагувати профіль, а також – за наявності відповідних прав – створювати та керувати авторськими публікаціями.

Такий підхід сприяє формуванню гнучкого, індивідуалізованого користувацького середовища, що відповідає потребам сучасного цифрового простору у сфері ментального здоров'я (рис. 3.12).

Profile Settings



The image shows a user profile settings interface. On the left, there is a circular profile picture placeholder with a blue shirt and black hair, and a button labeled "Edit Picture". To the right, there are three input fields labeled "Full Name", "Email", and "Password".

Рисунок 3.12 – Особистий кабінет користувача

У разі, якщо під час реєстрації користувач вводить ім'я, яке вже зареєстроване в системі, після надсилання форми з'являється відповідне повідомлення про помилку, що інформує про неможливість використання зазначеного імені. Такий механізм реалізовано для забезпечення унікальності облікових записів, запобігання дублюванню користувачів і підвищення безпеки системи. Повідомлення містить підказку, яка дозволяє користувачу оперативно змінити введені дані, наприклад, скорегувати ім'я або обрати інший унікальний варіант.

Якщо ж усі дані заповнені коректно, перевірку валідації пройдено успішно, і конфліктів із базою користувачів не виявлено, система завершує процес створення нового облікового запису. Після цього користувач автоматично перенаправляється до сторінки авторизації («LOGIN»), де йому пропонується здійснити вхід до системи, використовуючи вказані під час

реєстрації логін (електронну пошту або унікальне ім'я) та пароль. Такий підхід забезпечує логічну послідовність етапів входу до системи та сприяє зручному переходу до персоналізованої взаємодії з вебзастосунком (рис. 3.13).

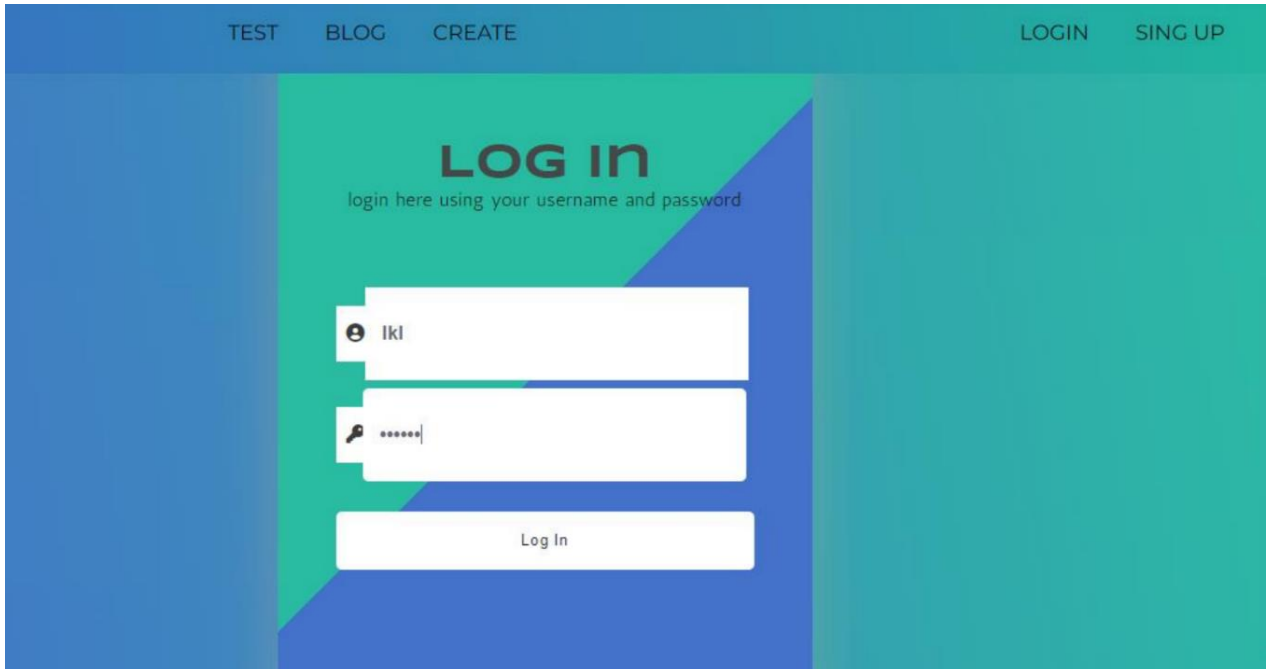


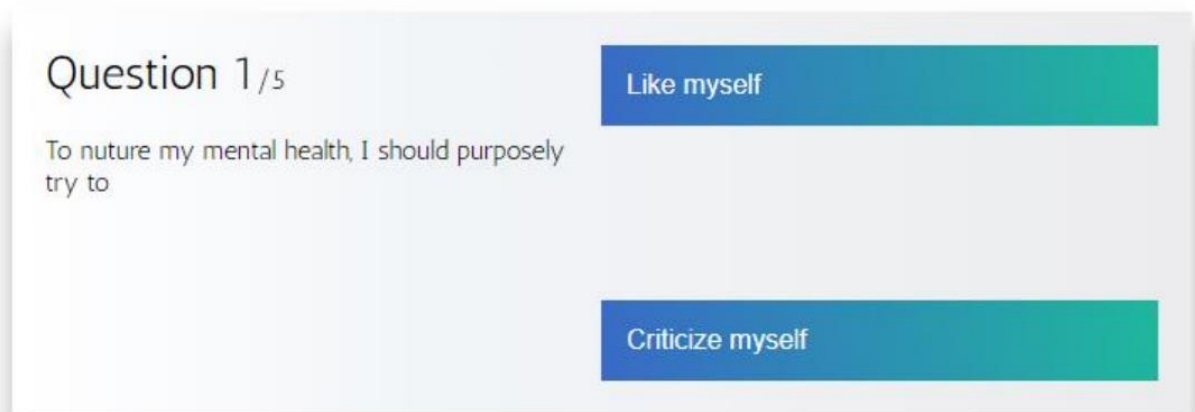
Рисунок 3.13 – Форма авторизації «Login»

Такий функціональний і структурний підхід до побудови вебзастосунку створює цілісну, інтуїтивно зрозумілу та зручну для користувача платформу, що відповідає сучасним вимогам до інформаційних систем у сфері охорони психічного здоров'я. Реалізація фіксованого меню значно покращує навігацію, сприяючи швидкому доступу до основних розділів незалежно від місця перебування на сторінці, а функція тестування дозволяє інтерактивно залучити користувача до процесу самооцінки, формуючи усвідомлене ставлення до власного емоційного стану.

У контексті цифрової трансформації охорони здоров'я, важливою перевагою платформи є її здатність об'єднувати інформаційний, комунікативний та діагностичний функціонал у межах єдиного середовища. Завдяки цьому користувач отримує не лише доступ до достовірної інформації, а й інструменти для самоспостереження та подальшого прийняття рішень

щодо власного психоемоційного благополуччя. Це відповідає провідним міжнародним підходам до створення eHealth-рішень, орієнтованих на користувача, і підкреслює потенціал вебзастосунку як дієвого інструменту підтримки ментального здоров'я у цифрову епоху (рис. 3.14).

How much do you know about caring for your mental health?



The image shows a screenshot of a digital assessment interface. At the top, it asks 'How much do you know about caring for your mental health?'. Below this, a question is displayed: 'Question 1 / 5' followed by the text 'To nurture my mental health, I should purposely try to'. There are two large, teal-colored buttons for response: 'Like myself' and 'Criticize myself'.

Рисунок 3.14 – Тест

Загалом, розроблена онлайн-платформа зосереджена на комплексному підході до інформування, самодіагностики та підтримки ментального здоров'я користувачів.

Поєднання функціональних модулів, таких як інтерактивні психологічні тести, авторський блог, персоналізований профіль користувача та система авторизації, створює ефективне середовище для реалізації ключових завдань цифрової психологічної підтримки.

Завдяки інтеграції сучасних UX/UI рішень, логічній структурі навігації, зручному інтерфейсу та дотриманню принципів безпеки персональних даних, платформа відповідає актуальним вимогам до інформаційно-освітніх вебресурсів у сфері охорони психічного здоров'я.

Таким чином, представлений вебзастосунок не лише сприяє підвищенню рівня обізнаності населення щодо психоемоційного благополуччя, але й створює передумови для формування цифрової культури

дбайливого ставлення до власного психологічного стану. Запропоноване рішення демонструє потенціал використання інформаційних технологій для підтримки ментального здоров'я, а також може стати основою для подальшого розвитку систем превентивної психогігієни в онлайн-середовищі.

ВИСНОВКИ

Розробка вебзастосунку для підтримки ментального здоров'я виявилася важливим кроком у розвитку цифрових технологій для вирішення проблем психоемоційного благополуччя. У сучасному світі, де стрес, тривожність та депресії є одними з основних викликів для багатьох людей, інноваційні підходи до надання допомоги та підтримки через інтернет стають надзвичайно актуальними. Цей вебзастосунок орієнтований на полегшення доступу до ресурсів для збереження та покращення ментального здоров'я, що є важливим аспектом як для тих, хто вже стикається з проблемами психоемоційного стану, так і для тих, хто бажає проводити профілактичні заходи.

Ключовими функціями вебзастосунку є інтеграція інформаційних ресурсів, доступ до психологічних тестів та можливість для користувачів взаємодіяти з контентом, що безпосередньо сприяє підвищенню усвідомленості стосовно власного емоційного стану. Завдяки зручному інтерфейсу, користувачі можуть швидко і без зайвих труднощів отримати доступ до необхідної інформації, зокрема статей, порад, рекомендацій щодо самопомоги та іншого корисного контенту, який може полегшити процес боротьби зі стресом чи іншими психологічними труднощами.

Однією з важливих складових застосунку є система психологічних тестів, що дозволяють користувачам оцінити свій емоційний стан. Тести розроблені таким чином, щоб бути простими для розуміння і надавати не лише точні результати, але й рекомендації для подальших дій. Така функція не лише допомагає користувачам краще розуміти своє психоемоційне здоров'я, але й дає їм можливість вчасно звернутися за професійною допомогою або вжити необхідних заходів для поліпшення свого стану. Це стає важливим інструментом профілактики, оскільки користувачі можуть самостійно відслідковувати свої психологічні зміни і, таким чином, діяти запобіжно.

Для підтримки безпеки даних користувачів у застосунку застосована надійна система автентифікації та реєстрації, що дає змогу зберігати особисті

дані в захищеному середовищі. Це важливо для створення довіри серед користувачів, оскільки ментальне здоров'я є досить чутливою темою, і багато людей можуть не бажати ділитися своїми переживаннями чи емоційними проблемами без гарантії їх конфіденційності.

Інтерактивність є ще однією перевагою цього вебзастосунок. Користувачі можуть створювати власний контент – статті, поради, міркування – що дозволяє формувати спільноту підтримки, де люди можуть обмінюватися досвідом та порадами щодо покращення психоемоційного благополуччя. Таке середовище сприяє підвищенню почуття єдності і меншій соціальній ізоляції серед користувачів, що є важливим аспектом у боротьбі з депресією та іншими розладами.

Не менш важливим є те, що цей застосунок дозволяє покращити доступність допомоги, оскільки люди з різних куточків можуть скористатися ресурсами, не залишаючи дому, що особливо важливо для тих, хто не має змоги звертатися до психолога чи психотерапевта через географічні чи фінансові обмеження. Завдяки цьому вебзастосунок робить професійну психологічну допомогу більш доступною та ефективною.

Таким чином, цей вебзастосунок є важливим елементом сучасної стратегії підтримки ментального здоров'я, що об'єднує технології, психологічні дослідження та принципи самопомочі. З його допомогою користувачі отримують змогу не лише слідкувати за власним емоційним станом, а й покращувати його завдяки доступу до цінної інформації та підтримки. Подібні проекти здатні значно покращити якість життя та сприяти зниженню рівня стресу та тривожності серед широкої аудиторії, що робить цей вебзастосунок надзвичайно актуальним у сучасному цифровому світі.

Результати дослідження апробовано у вигляді тез доповіді під час Міжнародного молодіжного форуму «РАДІОЕЛЕКТРОНІКА І МОЛОДЬ У ХХІ СТОЛІТТІ» [45].

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Ковальська, Н., Жук, А., & Корженко, В. (2024). Вплив дистресу війни в Україні на ментальне здоров'я. *Grail of Science*, 37, 406-410.
2. Ткачишина, О. Р. (2023). Проблема ментального здоров'я в Україні: психологічний аналіз. *Habitus*, (53), 207-211.
3. Лавріненко, Л. (2024). Ментальне здоров'я школярів в умовах воєнного стану. *Вісник Національного університету «Чернігівський колегіум» імені ТГ Шевченка*, 182(26), 133-138.
4. Андріяшевська, О., & Доренська, А. (2024). ЕКОЛОГІЧНА БЕЗПЕКА В УМОВАХ ВІЙНИ: МЕНТАЛЬНЕ ЗДОРОВ'Я СТУДЕНТСТВА. In *К64 Конкурентоспроможна модель інноваційного розвитку економіки України: матеріали VII Міжнар. наук.-практ. конф., м. Кропивницький, 07-08 лист. 2024 р./М–во освіти і науки України, Центральноукраїн. нац. техн. ун–т.–Кропивницький: ЦНТУ, 2024. – 410 с. (р. 275).*
5. Артьоменко, В., Жовтенко, О., Стасій, Я., & Пірон-Думітраску, М. (2024). Мікробіом людини та ментальне здоров'я: новітні погляди. *Сімейна Медицина. Європейські практики*, (2), 29-35.
6. Чепурко, Г., & Соболев, А. (2023). Ментальне здоров'я українців: виклики воєнного часу. *Collection of scientific papers «ΛΟΓΟΣ»*, (May 26, 2023; Boston, USA), 179-182.
7. Інжиєвська, Л. А. (2024). Ментальне здоров'я та його підтримка в умовах військових дій в Україні: електронний навчальний курс.
8. Асєєва, Ю. О., Возний, Д. В., & Волинчук, Т. І. (2024). ПСИХІЧНЕ ЗДОРОВ'Я МОЛОДІ В УМОВАХ ВІЙНИ: РЕЗУЛЬТАТИ ПСИХОДІАГНОСТИКИ ТА ТЕРАПЕВТИЧНИХ ІНТЕРВЕНЦІЙ. *Ментальне здоров'я*, (3), 19-25.
9. Teo, Y. N., Yong, E., Gautam, A., Chaulagain, R., & Yong, K. H. (2023). Artificial intelligence in future mental health industry. *Journal of Chitwan Medical College*, 13(4), 119-121.

10. Gritsch, S., Batchelor, T. T., & Gonzalez Castro, L. N. (2022). Diagnostic, therapeutic, and prognostic implications of the 2021 World Health Organization classification of tumors of the central nervous system. *Cancer*, 128(1), 47-58.
11. Latoo, J., Mistry, M., Wadoo, O., Islam, S. M. S., Jan, F., Iqbal, Y., ... & Alabdulla, M. (2022). Why mental health service delivery needs to align alongside mainstream medical services. *Asian journal of psychiatry*, 71, 103053.
12. Rankin, J. (2005). Mental health in the mainstream. *Criminal Justice Matters*, 61(1), 10-11.
13. Shafronenko, A. Y., Bodyanskiy, Y. V., & Holovin, O. O. (2023). CLUSTERIZATION OF DATA ARRAYS BASED ON THE MODIFIED GRAY WOLF ALGORITHM. *SCIENCE OF THE TOTAL ENVIRONMENT*, 873, 73-79.
14. Шафроненко, А. Ю., & Бодянський, Є. В. (2023). Адаптивний підхід до нечіткої кластеризації на основі еволюційної оптимізації алгоритму сірих вовків. *Збірник наукових праць Харківського національного університету Повітряних Сил*, (1 (75)), 77-81.
15. Bodyanskiy, Y. V., Pliss, I. P., & Shafronenko, A. Y. (2022). Кластеризація масивів даних на основі комбінованої оптимізації функцій щільності розподілу та еволюційного методу котячих зграй. *Radio Electronics, Computer Science, Control*, (4), 61-61.
16. Бодянський, Є. В., Плісс, І. П., & Шафроненко, А. Ю. (2022). Швидка нечітка правдоподібна кластеризація на основі аналізу піків щільності розподілу даних. *Радіoeлектроніка, інформатика, управління*, (1), 76-81.
17. Shafronenko, A. Y., Kasatkina, N. V., Bodyanskiy, Y. V., & Shafronenko, Y. O. (2023). CREDIBILISTIC ROBUST ONLINE FUZZY CLUSTERING IN DATA STREAM MINING TASKS. *Radio Electronics, Computer Science, Control*, (3), 97-97.

18. Шафроненко, А. Ю., & Бодянський, Є. В. (2023). Нечітка достовірна кластеризація великих масивів даних з гіпереліпсоїдальними класами з довільною орієнтацією осей. *Наука і техніка Повітряних Сил Збройних Сил України*, (1 (50)), 93-99.
19. Шафроненко, А. Ю., Бодянський, Є. В., & Руденко, Д. О. (2023). Модифікований рекурентний метод достовірної нечіткої кластеризації з використанням оптимізаційної процедури на основі косяків риб. *Системи обробки інформації*, (1 (172)), 92-96.
20. Bodyanskiy, Y., Shafronenko, A., & Pliss, I. (2022). Clusterization of vector and matrix data arrays using the combined evolutionary method of fish schools. *System research and information technologies*, (4), 79-87.
21. Bodyanskiy, Y. V., Pliss, I. P., Shafronenko, A. Y., & Kalynychenko, O. V. (2022). Нечітка довірча кластеризація даних на основі аналізу щільності розподілу даних та їх піків. *Radio Electronics, Computer Science, Control*, (3), 58-58.
22. Шафроненко, А. Ю., & Бодянський, Є. В. (2022). Адаптивна кластеризація багатоекстремальних масивів даних з використанням модифікованого алгоритму риб'ячої зграї. *АСУ та прилади автоматики*, (178), 33-37.
23. Bodyanskiy, Y., Pliss, I., & Shafronenko, A. (2022). Adaptive neurofuzzy clustering of distorted data based on prototype-centroid strategy using evolutionary procedures. *Artificial Intelligence*, 27, 239-244.
24. Bodyanskiy, Y. V., Shafronenko, A., & Klymova, I. (2021, April). Adaptive Recovery of Distorted Data Based on Credibilistic Fuzzy Clustering Approach. In *COLINS* (pp. 6-15).
25. Shafronenko, A., Bodyanskiy, Y. V., Klymova, I., & Holovin, O. (2020, May). Online credibilistic fuzzy clustering of data using membership functions of special type. In *CMIS* (pp. 744-753).
26. Levlin, M. (2020). DOM benchmark comparison of the front-end JavaScript frameworks React, Angular, Vue, and Svelte.

27. Delcev, S., & Draskovic, D. (2018, May). Modern javascript frameworks: A survey study. In 2018 Zooming Innovation in Consumer Technologies Conference (ZINC) (pp. 106-109). IEEE.
28. Gizas, A., Christodoulou, S., & Papatheodorou, T. (2012, April). Comparative evaluation of javascript frameworks. In Proceedings of the 21st International Conference on World Wide Web (pp. 513-514).
29. Odeniran, Q., Wimmer, H., & Du, J. (2024). Javascript frameworks—a comparative study between react.js and angular.js. In *Interdisciplinary Research in Technology and Management* (pp. 319-327). CRC Press.
30. Cincović, J., & Punt, M. (2020). Comparison: Angular vs. React vs. Vue. Which framework is the best choice? Zdravković, M., Konjović, Z., Trajanović, M.(Eds.) *ICIST 2020 Proceedings*, 250-255.
31. Jameel, S. A. (2022). *Comparison of Modern JavaScript Web Frameworks*.
32. Khati Chhetri, S. (2024). *Comparative Study of Front-end Frameworks: React and Angular*.
33. Uygur, M. F., & Yalçın, N. (2024, October). Comparative Analysis on React and Preact Javascript Frameworks. In 2024 9th International Conference on Computer Science and Engineering (UBMK) (pp. 1-6). IEEE.
34. Wohlgethan, E. (2018). *Supporting web development decisions by comparing three major javascript frameworks: Angular, react and vue.js* (Doctoral dissertation, Hochschule für Angewandte Wissenschaften Hamburg).
35. Duvander, J., & Romhagen, O. (2019). What affects the choice of a JavaScript framework: Interviews with developers.
36. Tulsyan, R., Shukla, P., Singh, T., & Kumar, A. (2024, February). The Impact of JavaScript Frameworks on Website Performance and User Experience. In 2024 IEEE International Conference on Big Data & Machine Learning (ICBDML) (pp. 299-305). IEEE.
37. Ambler, T., Cloud, N., & Hawkes, R. A. (2015). *JavaScript Frameworks for Modern Web Dev* (pp. 155-189). Apress.

38. Ivanova, S., & Georgiev, G. (2019, May). Using modern web frameworks when developing an education application: a practical approach. In 2019 42nd International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO) (pp. 1485-1491). IEEE.
39. Bauer, M. (2021). Hello Framework! A heuristic method for choosing front-end JavaScript frameworks.
40. Ferreira, F., Borges, H. S., & Valente, M. T. (2022). On the (un-) adoption of JavaScript front-end frameworks. *Software: Practice and Experience*, 52(4), 947-966.
41. Retzius, S., & Sundholm, E. (2022). Development of an evaluation model for client-side JavaScript Frameworks.
42. Bawane, M., Gawande, I., Joshi, V., Nikam, R., & Bachwani, S. A. (2022). A Review on Technologies used in MERN stack. *Int. J. Res. Appl. Sci. Eng. Technol*, 10(1), 479-488.
43. Aggarwal, S., & Verma, J. (2018). Comparative analysis of MEAN stack and MERN stack. *International Journal of Recent Research Aspects*, 5(1).
44. Shukla, S. K., Dubey, S., Rastogi, T., & Srivastava, N. (2022). Application using MERN Stack. *International journal for modern trends in science and technology*, 8(06), 102-105.
45. Карабута, М.О. (2025). Розробка вебзастосунку для підтримки ментального здоров'я. *Радіоелектроніка та молодь у XXI столітті: тези доповідей 29-го Міжнародного молодіжного форуму (Харків 16–19 квітня 2025р.)* Харків: ХНУРЕ, 2025. Т. 7. С. 58-59.