

Міністерство освіти і науки України  
Національний технічний університет  
«Дніпровська політехніка»

Інститут електроенергетики

(інститут)

Факультет інформаційних технологій

(факультет)

Кафедра Програмного забезпечення комп'ютерних систем

(повна назва)

ПОЯСНЮВАЛЬНА ЗАПИСКА  
кваліфікаційної роботи ступеня

магістра

(назва освітньо-кваліфікаційного рівня)

студента	<i>Горячева Антона Владиславовича</i> (ПІБ)		
академічної групи	<i>122М-22-2</i> (шифр)		
спеціальності	<i>122 Комп'ютерні науки</i> (код і назва спеціальності)		
освітньої програми	<i>«122 Комп'ютерні науки»</i> (назва освітньої програми)		
на тему:	<i>Розробка та обґрунтування веб-додатку з дослідженням рівня «емоційного інтелекту» користувачів</i>		

Керівники	Прізвище, ініціали	Оцінка за шкалою		Підпис
		рейтин говою	інституційною	
розділів кваліфікаційної роботи				
спеціальний	<i>проф. Мещеряков Л.І.</i>			

Рецензент				
Нормоконтролер	<i>доц. Гуліна І.Г.</i>			

Дніпро  
2023



#### 4 ЕТАПИ ВИКОНАННЯ РОБІТ

Найменування етапів робіт	Строки виконання робіт (початок – кінець)
Аналіз теми та постановка задачі	05.09.2023-24.09.2023
Аналіз засобів створення програмного забезпечення для проєктування веб-додатку та вивчення та аналіз усіх можливостей діагностики Н. Холла	25.09.2023-11.10.2023
Розробка програмного забезпечення та дослідження ефективності запропонованих рішень	12.10.2023-07.12.2023

Завдання видав

\_\_\_\_\_

(підпис)

*Мещераков Л.І.*

\_\_\_\_\_

(прізвище, ініціали)

Завдання прийняв до виконання

\_\_\_\_\_

(підпис)

*Горячев А.В.*

\_\_\_\_\_

(прізвище, ініціали)

Дата видачі завдання: 05.09.2023 р.

Термін подання кваліфікаційної роботи до ЕК 09.12.2023

## РЕФЕРАТ

**Пояснювальна записка:** 70 с., 9 рис., 2 дод., 48 джерела.

**Об'єкт дослідження:** Процес дослідження рівня "емоційного інтелекту" користувачів та його обґрунтування.

**Мета роботи:** Метою є розробка вебдодатку та проведення дослідження рівня "емоційного інтелекту" користувачів за допомогою бібліотеки React та психологічного тесту Н. Холла.

У роботі була створена платформа для взаємодії користувача з тестом Н. Холла. Додаток реалізований за допомогою мови програмування JavaScript, використовуючи бібліотеку React та тест Н. Холла для діагностики "емоційного інтелекту". Метою розробки веб-додатку є надання можливості не лише проводити тестування, але й отримувати обґрунтування результатів.

Важливо відзначити, що сучасна розробка вимагає серйозного, кваліфікованого та якісного підходу до користувачів. У зв'язку з цим були розроблені нові методи тестування "емоційного інтелекту", що сприятимуть покращенню взаємодії з платформою.

### **Ключові аспекти:**

**Технічна реалізація:** Веб-додаток розроблено на мові програмування JavaScript з використанням бібліотеки React, що забезпечує ефективну та інтерактивну роботу.

**Діагностика "емоційного інтелекту":** Застосовано тест Н. Холла, який дозволяє визначити рівень "емоційного інтелекту" користувачів, надаючи більше високорівневого розуміння їхнього емоційного стану.

**Список ключових слів:** тести, емоційний інтелект, вебдодаток, React, Н. Холл, JavaScript.

## ABSTRACT

**Explanatory note:** 88 pages, 30 pictures, 3 appendices, 52 sources.

**Research object:** The process of investigating the level of "emotional intelligence" of users and its justification.

**Objective of the work:** The aim is to develop a web application and conduct research on the level of users' "emotional intelligence" using the React library and the psychological test by N. Hall.

A platform for user interaction with the N. Hall test was created in the work. The application is implemented using the JavaScript programming language, employing the React library and the N. Hall test for diagnosing "emotional intelligence." The purpose of developing the web application is to provide not only the ability to conduct testing but also to receive justification for the results.

It is important to note that modern development requires a serious, qualified, and quality approach to users. In this regard, new methods of "emotional intelligence" testing were developed to improve interaction with the platform.

### **Key Aspects:**

**Technical Implementation:** The web application is developed in the JavaScript programming language using the React library, ensuring efficient and interactive operation.

**Diagnosis of "Emotional Intelligence":** The N. Hall test is applied, allowing the determination of users' "emotional intelligence" level, providing a higher-level understanding of their emotional state.

**Keywords:** tests, emotional intelligence, web application, React, N. Hall, JavaScript.

# ЗМІСТ

РЕФЕРАТ	4
ABSTRACT	5
ВСТУП	9
РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ ТА ПОСТАНОВКА ЗАВДАННЯ	12
1.1 Тестування в життєвому циклі програмного забезпечення	12
1.2 Актуальність мануального тестування	16
1.3 Існуючі підходи до управління ресурсами	17
1.4 Порівняльний аналіз засобів досягнення ефективності, масштабованості та швидкості роботи додатка	19
1.5 Аутентифікація та авторизація	25
1.6 Висновки	27
РОЗДІЛ 2. ЗБІР, СИСТЕМАТИЗАЦІЯ ТА ДОСЛІДЖЕННЯ ІНФОРМАЦІЇ ПРО СУЧАСНІ МОДЕЛІ ІНФОРМАЦІЙНИХ СИСТЕМ НА БАЗІ СУКУПНОСТІ ТЕХНОЛОГІЙ SPRING	29
2.1 Основи побудови ІС у Spring Framework	29
2.2 Архітектура Spring Framework	31
2.3 Структура діяльності автономного додатку	33
2.4 Обґрунтування рівня залежності та зв'язаності	35
2.5 Технології Spring для побудови додатка	37
2.5.1 Платформа Spring Boot	37
2.5.2 Захист додатку з Spring Security	40
2.6 Аналіз безпечності та швидкості алгоритмів шифрування	44
2.7 Обґрунтування вибору механізму асинхронного програмування	48
2.8 Висновки	49
РОЗДІЛ 3. РОЗРОБКА ЗАСТОСУНКУ ДЛЯ АВТОМАТИЗАЦІЇ ЗМІНИ СТАНУ РЕСУРСІВ ТЕСТУВАННЯ	52
3.1 Структура інформаційної системи автоматизації зміни стану ресурсів тестування.	52

3.2 Структура та опис бази даних.	54
3.3 Опис ІС testerossa та алгоритмів її функціонування.	56
3.4 Висновки.	63
ВИСНОВКИ	64
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	65
ДОДАТОК А ЛІСТИНГ ПРОГРАМИ	69
ДОДАТОК Б ВІДГУК КЕРІВНИКА	86
ДОДАТОК В ПЕРЕЛІК ФАЙЛІВ НА ДИСКУ	88

## ВСТУП

**Актуальність роботи.** У сучасному інформаційному суспільстві, де технології відіграють ключову роль у повсякденному житті, питання емоційного інтелекту стають дедалі важливішими. Емоційний інтелект уособлює здатність розуміння та управління власними емоціями, а також емпатію та ефективну взаємодію з емоціями оточуючих. У цьому контексті актуальним стає дослідження рівня емоційного інтелекту користувачів та розробка інструментів для його оцінки.

**Об'єкт дослідження.** Метою даної роботи є розробка веб-додатка для вимірювання рівня емоційного інтелекту користувачів із використанням бібліотеки React та психологічного тесту Н. Холла. Об'єктом дослідження виступає процес дослідження рівня емоційного інтелекту користувачів та його обґрунтування.

**Наукові очікувані результати.** Практична цінність роботи полягає у створенні платформи щодо тестування рівня емоційного інтелекту користувачів. Розроблений додаток реалізований з використанням мови програмування JavaScript, бібліотеки React та тесту Н. Холла для діагностики емоційного інтелекту. Очікується, що результати дослідження призведуть до глибшого розуміння емоційного інтелекту користувачів та нададуть основу розробки нових методів його тестування.

### **Етапи виконання роботи:**

- 1. Аналіз теми та постановка задачі (01.09.2023-30.09.2023):** У цей період проводиться аналіз предметної галузі та формулювання завдань, які потрібно вирішити у ході дослідження.
- 2. Аналіз засобів створення програмного забезпечення та вивчення діагностики Н. Холла (01.10.2023-31.10.2023):** На цьому етапі визначаються інструменти для розробки веб-додатку, а також вивчаються всі можливості діагностики Н. Холла.
- 3. Розробка програмного забезпечення та дослідження ефективності рішень (01.11.2023-15.11.2023):** На даному етапі здійснюється створення веб-додатку та проведення експериментів для



оцінки ефективності запропонованих рішень.

**Завдання наукового дослідження.** Основним завданням є створення універсальної та ефективної платформи для тестування емоційного інтелекту користувачів. Це включає адаптацію сучасних технологічних рішень, таких як бібліотека React, до потреб психологічного тестування. Програма повинна надавати користувачам можливість не тільки пройти тест, але й отримати якісні дослідні обґрунтування свого рівня емоційного інтелекту.

**Наукові результати.** Результати дослідження призведуть до створення інноваційного інструменту, здатного не лише вимірювати рівень емоційного інтелекту, а й надавати цінні психологічні висновки для користувачів. Розроблений додаток може стати основою для подальших досліджень у галузі взаємозв'язку технологій та психології, а також сприяти розвитку нових підходів до вивчення та підвищення емоційного інтелекту.

## РОЗДІЛ 1

# ТЕХНОЛОГІЇ ТА МЕТОДИ ДЛЯ РОЗВ'ЯЗАННЯ ПОСТАВЛЕНОЇ ЗАДАЧІ

### 1.1 Теоретичні аспекти психології в онлайн-формат

У сучасному світі, де віртуальна реальність стає все більш невід'ємною частиною нашого повсякденного життя, застосування психологічних принципів та методів у онлайн-середовищі набуває великого значення. Цей розділ літературного огляду присвячений розгляду основних аспектів та напрямків використання психології в онлайн-форматі.

Психологічні аспекти взаємодії в онлайн-середовищі. Одним із ключових аспектів використання психології в онлайн-середовищі є вивчення взаємодії між користувачами та віртуальним оточенням. Психологічні принципи допомагають розуміти, як сприймається інформація в онлайн-режимі, як виникають емоції під час взаємодії з віртуальними об'єктами, та як це впливає на поведінку користувачів.

Психологічні тести в онлайн-форматі. Особливий акцент у літературному огляді робиться на застосуванні психологічних тестів в онлайн-форматі. Аналізується ефективність та надійність таких тестів, їх психометричні властивості та можливі обмеження. Освітлюються теоретичні підходи до створення та використання психологічних тестів у віртуальному середовищі.

Етичні питання використання психології в онлайн-середовищі. Розглядаються етичні аспекти застосування психології в онлайн-форматі, зокрема, питання конфіденційності даних, дозволів на проведення психологічних досліджень в мережі, та впливу віртуального взаємодії на психічне здоров'я користувачів.

Переваги та виклики використання психології в онлайн-форматі.

У цьому розділі розглядаються позитивні аспекти використання психології в онлайн-середовищі, такі як можливість проведення дистанційних психологічних

консультацій, а також виклики, з якими можуть зіткнутися дослідники та практики при роботі в цьому сегменті.

Сучасні тенденції та перспективи розвитку. Заключний підрозділ літературного огляду присвячений аналізу сучасних тенденцій використання психології в онлайн-середовищі та прогнозуванню перспектив розвитку даної галузі.

Тест Холла на эмоциональный интеллект (EQ)

1 1 из 30

Когда позволяет время, я обращаюсь к своим негативным чувствам и разбираюсь, в чем проблема.

- Полностью согласен
- В основном согласен
- Отчасти согласен
- Отчасти не согласен
- В основном не согласен
- Полностью не согласен

Далее Завершить

Автор: Методика – Н. Холл, адаптация – Е. Ильин

Рис. 1.1. Приклад тесту в онлайн-форматі

## 1.2 Технології веброзробки

Технології веброзробки займають важливе місце у створенні сучасних інтернет-ресурсів та додатків. В даному розділі роботи розглядається широкий спектр інструментів та технологій, які використовуються для розробки веб-додатків і можуть бути застосовані у створенні вебсайту з тестом по психології.

HTML є стандартною мовою розмітки для створення структури веб-сторінок. Вона визначає різноманітні елементи та їхні взаємодії, включаючи заголовки, параграфи, таблиці, форми та інші. Його основною функцією є створення логічної структури документа, яка може бути інтерпретована браузером.

HTML еволюціонувало з часом, додавши нові теги та атрибути, які дозволяють розробникам створювати більш динамічні та зручні для користувача сторінки. Зокрема, HTML5 вводить нові можливості, такі як аудіо та відео вбудування, а також покращені можливості форм та графіки.

CSS відповідає за стилізацію та зовнішній вигляд веб-сторінок, відокремлюючи оформлення від структури. За допомогою CSS можна задавати кольори, розміри шрифтів, відступи, рамки та інші аспекти дизайну. Каскадна природа CSS дозволяє використовувати різні таблиці стилів для різних елементів та рівнів структури.

Сучасні підходи включають у себе адаптивний дизайн, який дозволяє сторінкам коректно відображатися на різних пристроях та екранах. Також використовуються препроцесори CSS, такі як Sass або Less, які дозволяють розробникам використовувати змінні, вкладені правила та інші покращення, що спрощують процес написання стилів.

### **JavaScript: роль та вплив у веб-розробці**

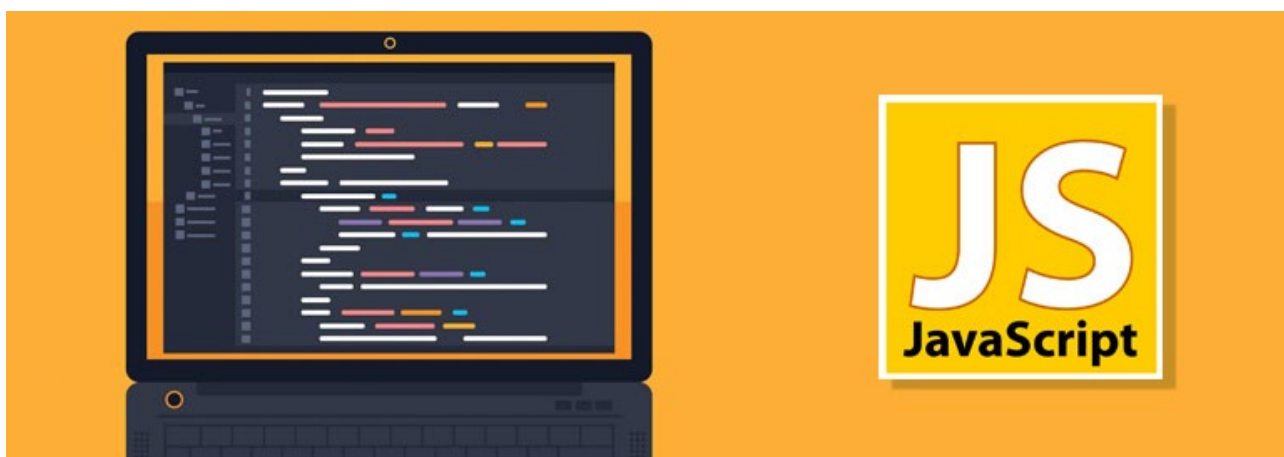


Рис. 1.2. Логотип JavaScript

JavaScript є мовою програмування, яка відіграє ключову роль у веб-розробці, надаючи можливість динамічного та інтерактивного взаємодії користувачів з веб-сторінками. У цьому розділі розглядається різноманітність використання JavaScript та його вплив на розробку сучасних веб-додатків.

JavaScript використовується для динамічної зміни вмісту сторінок без їх повного перезавантаження. Це досягається за допомогою DOM, який дозволяє змінювати вміст та структуру HTML-документа за допомогою JavaScript. Такий підхід створює плавні та швидкі переходи між сторінками, покращуючи взаємодію з користувачем.

JavaScript у поєднанні з технікою AJAX чи Asynchronous JavaScript and XML дозволяє виконувати асинхронні запити до сервера без повного перезавантаження сторінки. Це важливо для швидкої та ефективної передачі даних між клієнтом та сервером, забезпечуючи користувачам плавний та безперебійний досвід взаємодії з вебсайтом.

JavaScript дозволяє структурувати код за допомогою функцій та об'єктів, що полегшує розробку та підтримку проекту. Використання об'єктно-орієнтованого підходу сприяє створенню чистого та організованого коду, що полегшує спільну роботу команди розробників.

З'явлення сучасних фреймворків, таких як React, дало нові можливості для швидкого та ефективного створення інтерфейсів. React використовується для компонентної розробки, що спрощує управління станом додатку та робить його більш масштабованим.

**Сучасні фреймворки та інструменти:** В сучасному веб-розробці існує багато фреймворків та інструментів, які пропонують інноваційні підходи до створення інтерактивних та ефективних веб-додатків. В даному розділі розглянемо деякі з них та їхній вплив на розробку веб-сайтів з психологічними тестами.

**Angular.** Angular, розроблений командою Google, є одним із найпопулярніших фреймворків для створення великих та складних веб-додатків. Він визначає строгий підхід до організації коду за допомогою компонентного підходу. Angular також має вбудовані засоби для роботи з HTTP-запитами, формами та іншими типовими задачами, що можуть бути важливими для розробки психологічних тестів в онлайн-форматі.

**Vue.js.** Vue.js - це легкий та гнучкий JavaScript-фреймворк, який зосереджений на простоті та легкості використання. Він дозволяє швидко інтегрувати компоненти в існуючий проект та легко масштабувати додаток. Vue.js може бути привабливим вибором для розробки веб-сайту з тестами, оскільки він пропонує чистий та зрозумілий API.

**Svelte.** Svelte представляє собою новий підхід до створення веб-додатків, де код написаний на високорівневому мові, і компілюється в ефективний JavaScript код під час збірки. Його головна перевага - відсутність рантайм-бібліотек у фінальному коді, що може збільшити продуктивність. Для психологічних тестів, де важлива швидкість та ефективність, Svelte може бути цікавим варіантом.

**React.** React, розроблений Facebook, є одним з найпопулярніших бібліотек для створення інтерфейсів користувача. Його використання у контексті розробки вебсайту з психологічними тестами є обґрунтованим завдяки його компонентному підходу, віртуальному DOM та великому спільноті розробників. Вибір оптимального фреймворку для проекту

**Вибір оптимального фреймворку для проекту:** Вибір конкретного фреймворку чи бібліотеки повинен бути обґрунтованим з урахуванням специфіки проекту, його масштабів, потреб користувачів та інших чинників. У цьому розділі буде здійснено аналіз і порівняння основних характеристик різних фреймворків для визначення оптимального вибору для створення веб-сайту з психологічними тестами.

### 1.3 Бібліотека React та чому була обрана саме вона



Рис. 1.3. Логотип React

Бібліотека React, була розроблена компанією Facebook, стала однією з найпопулярніших та широко використовуваних технологій у сфері веб-розробки. Її унікальні особливості та ефективність при створенні інтерактивних інтерфейсів забезпечують високий рівень користувацького досвіду у веб-додатках, зокрема, у випадку створення вебсайту з тестами по психології.

React визначається своєрідним підходом до розробки, відомим як компонентний. Замість традиційного підходу до створення веб-сторінок, React розбиває інтерфейс на компоненти - невеликі, самодостатні елементи, які легко керувати та перевикористовувати. Це спрощує процес розробки, дозволяючи розробникам концентруватися на кожному елементі окремо та полегшує супровід коду.

Важливим елементом React є віртуальний DOM, який дозволяє ефективно оновлювати сторінку, не перезавантажуючи її повністю. Це робить React ідеальним

для створення динамічних веб-додатків, таких як психологічні тести, де важлива швидкість та миттєвість відгуку.

Однією з суттєвих переваг React є його ефективна реакція на зміни стану додатка. Коли користувач взаємодіє з тестом чи відповідає на питання, React автоматично оновлює лише ті частини інтерфейсу, які потребують змін. Це зменшує навантаження на ресурси та робить веб-додаток ефективним у використанні.

React активно підтримується великою спільнотою розробників, що сприяє обміну досвідом та вирішенню проблем. Наявність безлічі готових бібліотек та рішень для React дозволяє розробникам швидко та ефективно реалізовувати різноманітні функціональності, що є актуальним у випадку створення вебсайту з психологічними тестами.

React є дуже адаптабельним до різних потреб веб-розробки. Він може використовуватися для створення невеликих веб-сайтів із складними інтерфейсами, а також для великих проєктів зі складною логікою та інтенсивними завданнями.

## **1.4 Використання React в психологічних тестах**

Використання бібліотеки React у сфері психологічних тестів відкриває безліч можливостей для покращення взаємодії та користувацького досвіду. Нижче представлені конкретні приклади використання React у розробці вебсайтів з психологічними тестами:

1. За допомогою React можна реалізувати динамічність тестових питань, де наступне питання виводиться на основі відповіді користувача на попереднє. Це дозволяє персоналізувати тест для кожного користувача, адаптуючи його до його попередніх відповідей та реакцій.
2. Психологічні тести можуть включати складні інтерфейси з різними типами



питань, графіками та візуалізаціями. Завдяки компонентному підходу React, створення та управління складними інтерфейсами стає значно простішим завданням, що сприяє розширенню можливостей психологічних тестів.

3. React дозволяє створювати інтерактивні елементи для отримання результатів тестів. Зміна стану та відображення результатів в реальному часі допомагає користувачам отримати зрозумілу та негайну зворотню інформацію.
4. React пропонує ефективні рішення для адаптації веб-додатків до різних типів пристроїв. Це особливо важливо для психологічних тестів, оскільки користувачі можуть взаємодіяти з тестами на різних пристроях, від комп'ютерів до смартфонів.
5. React допомагає реалізувати інструменти для збору та аналізу даних користувачів. Це дозволяє проводити більш глибокий аналіз відповідей на питання та психологічні параметри, що допомагає вдосконалювати тести та розуміти поведінку користувачів.
6. React дозволяє ефективно реалізувати систему обробки помилок та повідомлень для користувачів. Це особливо важливо у психологічних тестах, де важливо забезпечити користувачів підтримкою та допомогою у випадках невірних або неочікуваних відповідей.
7. React має широкий екосистему готових компонентів, які можна легко інтегрувати у психологічні тести. Це може бути, наприклад, календар для визначення дати чи слайдер для вибору параметрів. Використання готових рішень спрощує розробку та розширює функціональність додатку.
8. React може легко інтегруватися з різноманітними засобами зберігання даних, включаючи бази даних, хмарні рішення та інші сервіси. Це дозволяє збирати та аналізувати дані з тестів у реальному часі, а також надає можливість додаткового наукового дослідження.

## РОЗДІЛ 2

# **ВИМОГИ ДО ФУНКЦІОНАЛУ ВЕБСАЙТУ ТА СЕРВЕРНОЇ ЧАСТИНИ**

## **2.1 Вимоги до функціоналу вебсайту**

У рамках реалізації вебсайту із тестом по психології передбачається впровадження ряду ключових функціональних можливостей:

### **Авторизація та Реєстрація:**

- Створення системи авторизації та реєстрації користувачів для забезпечення персонального доступу.
- Збереження інформації про користувачів, включаючи основні дані та історію тестувань.

### **Відображення Тестів:**

- Створення інтерфейсу для відображення різноманітних психологічних тестів.

### **Проходження Тестів:**

- Механізм проведення тестів з автоматичним збереженням відповідей користувачів.
- Інтерактивний та зручний інтерфейс для заповнення тестових питань.

## **2.1 Вимоги до серверної частини**

Для забезпечення ефективної роботи вебсайту та обробки тестових відповідей користувачів, серверна частина повинна включати наступний функціонал:

### **Реєстрація та Аутентифікація**

- Реалізація механізму реєстрації нових користувачів та їх подальшої аутентифікації.

- Захист від несанкціонованого доступу та збереження конфіденційності даних.

### **Зберігання Даних Користувачів та Результатів Тестувань**

- Створення бази даних для зберігання основних даних користувачів та результатів їх тестувань.
- Ефективне управління базою даних для забезпечення швидкого доступу до інформації.

### **Логіка Тестування та Обробка Результатів**

- Розробка логіки обробки відповідей користувачів на психологічні тести.
- Аналіз результатів та виведення відповідної інформації для користувача.

### **API для Взаємодії із Клієнтською Частиною**

- Розробка API для забезпечення взаємодії між клієнтською та серверною частинами.
- Документування API для зручного використання та розширення функціоналу.

## **ОСОБЛИВОСТІ ПЕРЕЖИВАННЯ СУБ'ЄКТИВНОГО ВІДЧУТТЯ САМОТНОСТІ**

### **3.1 Психологічні переживання суб'єктивного відчуття самотності**

У сучасному світі термін "самотність" стає все більш поширеним, а інтерес до вивчення цієї проблеми постійно зростає. Вчені стверджують, що відчуття самотності відіграє ключову роль у формуванні особистості, і тому вивчення самотності як психологічного та соціального явища наразі займає важливе місце в наукових досліджах. Знамениті вчені, такі як Г.С. Салліван, І. Ялом, З. Фрейд, Е. Фромм, В. Франкл, а також вітчизняні науковці, зокрема С.Г. Корчагіна, С.Л. Вербицька, Є.М. Заворотних, О.В. Неумоева-Колчеданцева, І.М. Слободчиков, Н.В. Хамітов, М.А. Давидова та інші, займалися трактуванням поняття самотності.

Важливо відзначити, що протягом століття виникає питання про те, що саме є "самотність особистості", проте науковці ще не доходять до загального висновку. Ймовірно, згоди не буде, поки існують дві кардинально протилежні точки зору на спрямованість самотності. Наразі існують розходження думок, одні вчені вважають, що самотність має негативний відтінок, тоді як інші аргументують позитивні аспекти цього досвіду. Зараз вважається, що переживання самотності є подвійним явищем, і тому вкрай складно надати йому універсальне пояснення.

У психологічних словниках самотність розглядається як психогенний фактор, що впливає на емоційний стан та психічне здоров'я людини, проявляючись у довготривалій фізичній або емоційній ізоляції. За думкою Є.М. Заворотних та М.М. Мовчан, самотність є багатоплановим, амбівалентним психічним явищем і може розглядатися як відчуття, стан, процес чи відношення.

Для нашого дослідження важливо розуміти сутність суб'єктивного відчуття самотності. Це відчуття виникає, коли людина вважає себе інакшою, відторгнутою, не в своєму елементі, що призводить до утворення психологічного бар'єру у спілкуванні з іншими. Самотність часто виникає через стійку думку особистості,

що вона не може побудувати близькі стосунки з ким-небудь протягом певного періоду життя. З'явлення цього суб'єктивного відчуття не залежить від кількості міжособистісних контактів, стану психічного здоров'я, переважаючого емоційного настрою і обумовлено особистісними властивостями особи.

Загалом, самотність розглядається як соціально-психологічне явище, емоційний стан людини, який викликає сильне суб'єктивне відчуття і передає особливу форму самосвідомості, виявляючи розрив між соціальними відносинами та внутрішнім світом особистості.

Після аналізу літературних джерел можна визначити, що більшість дослідників акцентують увагу на негативних аспектах самотності, визначаючи її руйнівний характер. Серед таких дослідників Ф. Фромм-Рейхман, Е. Фромм, М. Мід, К. Голдберг, Г. Зілбург, Г.С. Салліван, Р.С. Немов, В.І. Кошкарлова . Інші науковці виявляють амбівалентне ставлення до проблеми самотності, а деякі, наприклад, А. Маслоу та К. Мустакас, вважають, що переживання самотності може бути єдиним шляхом до самовдосконалення особистості.

З метою глибшого розуміння явища самотності доцільно розглянути кілька, на нашу думку, найбільш зацікавлюючих і вичерпних класифікацій видів самотності. Відомий учений І. Ялом розробив власну класифікацію видів самотності залежно від типу ізоляції:

- 1) Внутрішньоособистісна самотність, коли людина намагається втекти від власних бажань;
- 2) Міжособистісна самотність, коли людина розриває свої соціальні зв'язки з різних причин;
- 3) Екзистенціальна самотність, коли людина не знаходить сенсу у своєму житті.

З свого боку психолог Дж. Янг виділяє три типи самотності:

- 1) Хронічна самотність, яка триває довго через нездатність індивіда

встановлювати тісні міжособистісні зв'язки;

2) Ситуативна самотність, що є тимчасовою і викликана конкретною подією у житті людини;

3) Перехідна самотність, що є невеликими періодами смутку, які раптово виникають і так само раптово проходять без помітних наслідків.

Є.М. Заворотних описує три рівні самотності як суб'єктивний досвід:

1) Низький рівень самотності, коли людина не відчуває себе ізольованою від інших;

2) Комфортний середній рівень самотності, коли цей стан існує, але не викликає особливого дискомфорту;

3) Деструктивний рівень самотності, який викликає негативні почуття і переживання у особистості.

Дослідниця також стверджує, що відчуття самотності не залежить від кількості міжособистісних контактів, друзів чи близького оточення, оскільки це суб'єктивне переживання, що визначається особистісними характеристиками людини.

С.Г. Корчагіна, оцінивши проблему самотності, розробила власну класифікацію, виділяючи:

1) відчужуючу самотність – це перебування особи в тривалому стані самотності, обумовленому її особистісними рисами;

2) дифузну самотність, яка виражає страх перед загальною самотністю, що призводить до розчинення особистості в своєму оточенні;

3) дисоціюючу самотність – це найбільш руйнівний вид самотності, що спостерігається у людей з експресивною психікою та нестабільним світосприйняттям.

Л.Ю. Айвазян вказує на наявність двох протилежних понять, які слід

вивчати по-різному. Вона розрізняє відчуття самотності фізично або соціально ізольованої особистості від самого суб'єктивного відчуття самотності. Дослідниця вважає, що ці поняття мають багато спільного, але не є ідентичними. Багато вчених не роблять цього розгорнення у своїх дослідженнях, що призводить до суперечок в теоретичних джерелах та наукових роботах. Л.Ю. Айвазян також спрямовувала свої дослідження на вивчення зв'язку між суб'єктивним відчуттям самотності та розвитком особистості, вказуючи, що це явище є нормативним і відіграє позитивну роль у житті людини.

Ми знаємо, що найбільш вразлива група людей, яка відчуває суб'єктивну самотність, - це підлітки. Проте ще в ранньому віці діти можуть стикатися з цими почуттями, коли формується їхня особистість. Якщо дорослі не звернуть увагу на це вчасно, це почуття може значно загостритися в підлітковому віці, що негативно впливає на навчання, відносини з батьками та сверстниками.

Підлітковий вік - це критичний період у житті кожної людини. У цей період остаточно закріплюються всі риси характеру та способи емоційної реакції, які в подальшому впливатимуть на життя вже дорослої особи. Цей етап переходу від дитинства до зрілості супроводжується виникненням багатьох проблем: демонстративна невпевненість в собі, конфлікти з важливими дорослими, орієнтація на групу сверстників, а також відчуття неприйняття та самотності.

Учені стверджують, що відчуття самотності може виникати у підлітка, коли він не встановлює емоційної близькості з важливими для нього ровесниками, коли його ігнорують або відштовхують. У дітей старшого підліткового віку це відчуття може загостритися, негативно впливаючи на їхнє психічне здоров'я і ускладнюючи процес становлення особистості. Ситуації непорозуміння з однолітками, конфлікти з батьками, вікова криза, відсутність сенсу життя, перше кохання, а також булінг можуть сприяти поглибленню відчуття самотності та відчуженості серед старших підлітків.

Ю.В. Никоненко провела дослідження особливостей переживання самотності у підлітків. Вона визначила, що у цьому періоді життя діти

переоцінюють своє оточення, порівнюючи друзів з власними ідеалами, цінностями і нормами, і віддаляються від невідповідних осіб. За словами дослідниці, самотність може позитивно впливати на інтеграцію підлітків у суспільство, оскільки вони вчаться аналізувати та формувати своє оточення, що сприяє їхньому розвитку.

Отже, визначено, що відчуття самотності найгостріше переживається у підлітковому віці. Більшість дослідників зосереджуються на цій групі, стверджуючи, що самотність негативно впливає на особистість підлітка. Проте слід врахувати, що самотність у підлітковому віці є поширеним явищем, і важливо, як підліток знаходить шляхи виходу з цього стану. Слухаючи себе і свої бажання, використовуючи відчуття самотності конструктивно, підліток може виявити своє покликання, глибше захопитися хобі, розвивати вміння та успішно подолати відчуття самотності, знаходячи друзів зі спільними інтересами.

Визначено, що відчуття самотності не завжди має негативне забарвлення, і свідомі діти можуть ефективно працювати з цим відчуттям для власної користі. Однак не всі підлітки готові самотійно вирішувати цю проблему. Таким чином, вивчення дослідниками самотності у підлітковому віці, спостереження та підтримка батьків та вчителів, а також адаптивна робота шкільних психологів можуть допомогти вчасно розпізнати перші симптоми і допомогти підлітку усвідомити свій стан, що, в результаті, сприятиме успішній соціалізації

### **3.2 Загальна характеристика поняття «емоційний інтелект»**

Вчені давно вказують на вплив емоційних виявів на поведінку та мислення людини, і вперше термін "емоційний інтелект" був введений Дж. Мейером та П. Селовеєм у 90-х роках ХХ століття в їхній науковій статті. На початковому етапі вивчення феномен емоційного інтелекту розглядався як частина соціального інтелекту. Вчені розробили першу модель емоційного інтелекту, включаючи ідентифікацію та вираження емоцій, емоційну саморегуляцію та використання



емоційно забарвленої інформації у мисленні та діяльності. Пізніше їхню модель розширили, визначивши, що до емоційного інтелекту входять когнітивні здібності, пов'язані з обробкою емоційної інформації, такі як ідентифікація емоцій, емоційне сприяння мисленню, розуміння емоцій і керування емоціями.

Їхня концепція стала основою першої теорії емоційно-інтелектуальних здібностей. Після виходу науково-популярної книги Д. Гоулмана тема емоційного інтелекту стала широко популярною, а сам вчений стверджував, що успіх людини залежить не тільки від розумових здібностей, але і від здібностей до самопізнання, емоційної саморегуляції та вміння розпізнавати та висловлювати свої почуття.

Хоча існують різні підходи до розуміння емоційного інтелекту, важливими компонентами є емоційні здібності, вольові якості, соціальні вміння та навички, а також самосвідомість. Теорія Р. Бар-Она ввела поняття емоційно-соціального інтелекту, описуючи його як сукупність міжособистісних та внутрішньоособистісних здібностей, навичок і умінь, які визначають поведінку. Бар-Он також вперше ввів поняття коефіцієнту емоційності (EQ) та описав п'ять компонентів емоційного інтелекту: інтраперсональний, інтерперсональний, управління стресом, здатність до адаптації та загальний настрій. Вищезазначені дослідники, такі як Р. Робертс, М. Зайднер та Д. Люсин, висвітлюють дві концепції емоційного інтелекту:

1) В рамках моделей здібностей емоційний інтелект визначається як набір здібностей, які вимірюються за допомогою тестів із завданнями на правильні та неправильні відповіді.

2) Змішані моделі емоційного інтелекту тлумачать його як складний психічний утвір, який включає як когнітивні, так і особистісні компоненти.

Модель емоційного інтелекту Д. Люсина, разом із його двокомпонентною теорією емоційного інтелекту, є найбільш поширеною в практиці. Він виділяє два аспекти у структурі емоційного інтелекту: внутрішньоособистісний (усвідомлення та керування власними емоціями) та міжособистісний (усвідомлення й керування

емоціями інших людей).

У вітчизняній науці теорія про єдність емоцій та інтелекту знайшла своє відображення в працях таких вчених, як Л.С. Виготський, С.Л. Рубінштейн, О.М. Леонтьєв. Л.С Виготський вважав, що емоції є більш складними, ніж інстинкти, і визначив інтелектуальний відрізок між зародженням думки і безпосередньою дією як "сміслові переживання," близьке до поняття "емоційний інтелект".

О.М. Леонтьєв розширив цю ідею, стверджуючи, що мислення включає моменти емоційної регуляції, де емоційні сигнали надають подіям чуттєве забарвлення і наповнюють їх особистісним сенсом. С.Л. Рубінштейн також відзначав, що емоції є невід'ємною частиною пізнавального процесу, об'єднуючи емоційний і інтелектуальний компоненти, подібно до пізнавальних процесів. О.К. Тихоміров ввів поняття "емоційне мислення," яке, фактично, перетинається з "емоційним інтелектом," оскільки специфіка емоційного мислення полягає в тому, що всі емоційні переживання, відчуття та стани особистості включаються у процес вирішення задач.

Емоційний інтелект можна розглядати як складову соціального інтелекту. У своїй концепції соціального інтелекту Савенков О.І. виділяє три основні групи критеріїв: когнітивні, емоційні та поведінкові. Кожна з цих груп може бути розглянута як:

1. Когнітивні аспекти: соціальні знання, соціальна інтуїція, соціальне прогнозування.
2. Емоційні компоненти: емоційна чуттєвість, емоційна залученість, емоційний контроль, емпатія та здатність до саморегуляції.
3. Поведінкові аспекти соціальне сприйняття, соціальна взаємодія, соціальна адаптація.

Виникає проблема чіткого визначення меж між емоційним і соціальним інтелектом. З одного боку, емоційний компонент присутній у структурі соціального інтелекту, відображаючи емоційні переживання у міжособистісній

взаємодії. З іншого боку, емоційний інтелект включає соціальнокомунікативні аспекти, пов'язані з розумінням і передбаченням емоцій у контексті спілкування.

Вчені вважають, що розвиток емоційного інтелекту можливий. Дослідження, проведені С.А. Праховою та М.Н. Макаренко, свідчать про позитивні результати у розвитку емоційного інтелекту у дітей молодшого шкільного віку. Також існують програми, наприклад, розроблена С.М. Могиластою, які спрямовані на підвищення рівня емоційного інтелекту серед старшокласників. С.О. Подофей вважає, що для успішної соціалізації дітей необхідна налагоджена психологічно-педагогічна робота в освітніх закладах, яка включає програми розвитку емоційного інтелекту. Ці програми повинні систематично впроваджуватися та адаптуватися відповідно до потреб батьків, вчителів та учнів.

Загалом, концепція емоційного інтелекту відображає ідею єдності емоційних та інтелектуальних процесів. Емоційний інтелект розглядається як здатність розуміти та управляти емоціями, розвиток якої може позитивно впливати на міжособистісні відносини та успішність в різних сферах діяльності.

### **3.3 Взаємозв'язок емоційного інтелекту та суб'єктивного відчуття самотності**

Текст анализирует внимание, уделяемое изучению эмоционального интеллекта, как со стороны зарубежных, так и отечественных ученых. Однако информации о исследованиях, направленных на выявление связи между эмоциональным интеллектом и субъективным ощущением одиночества, крайне мало. Эксперт в области эмоционального интеллекта Д. Гоулман отмечает связь между неспособностью воспринимать эмоциональные сигналы, реагировать на них, и чувством одиночества и отчужденности у детей. Он подчеркивает, что если дети не могут распознавать и правильно интерпретировать эмоциональные сигналы сверстников, то они не смогут адекватно реагировать, что обязательно приведет к недопониманию.

Експерти Р. Канаи и Б. Бахрами установили неразрывную связь между чувством одиночества и объемом серой материи в задней части скрени. Эта область отвечает за овладение социальными навыками, способность распознавать и интерпретировать свои и чужие эмоции, а также проводить первичную оценку настроения собеседника по мимическому выражению лица. В случае недостаточного развития этой части мозга человек испытывает трудности в идентификации эмоциональных сигналов окружающих, что приводит к уменьшению социальных контактов и усилению чувства непонимания, отчужденности и, в конечном итоге, к устойчивому ощущению одиночества.

Исследование также указывает на важную роль матери в развитии эмоционального интеллекта ребенка. Взаимодействие матери с ребенком в первые годы жизни закладывает фундаментальные навыки и основные стратегии поведения, позволяющие ребенку разбираться в своих чувствах и контролировать эмоции. Таким образом, успешное выполнение материнской роли в дальнейшем определит, будет ли ребенок чувствовать себя одиноким и непонятым в обществе, или же сможет успешно вливаться в коллектив, заводить друзей и хорошо чувствовать себя в социуме.

І.М. Андрєєва визначила, що у сім'ях, де батьки відкрито можуть обговорювати проблеми, діти вчаться розуміти власні емоції та самостійно регулювати внутрішні переживання. Але потрібно зауважити, що вирішення проблем має відбуватися не криками, а аргументами, коли батьки пояснюють один одному, які почуття вони переживають і які емоції викликала дана ситуація в кожного з них, а дитина спостерігає за цим і вчиться на їхньому прикладі. В сім'ях, які благополучні в емоційному плані, найчастіше виникає певна виховна стратегія, в якій головна увага переключається з поведінки дитини на її почуття та переживання.

В.В. Овсяннікова досліджувала зв'язок переживання самотності з рівнем рефлексії у підлітків. На її думку, інтенсивність та глибина переживань самотності прямо залежить від вміння усвідомлювати свої можливості та користуватися

власними ресурсами. Вона бачить два варіанти розвитку подій: у разі, якщо дитина має адекватну самооцінку, реально оцінює себе та свої здібності, свідомо підходить до вирішення будь-яких життєвих питань, то її рефлексія буде розвиватися і навіть перебування на самоті йтиме їй на користь. Якщо ж у підлітка будуть проявлятися супутні проблеми, а саме негативні настрої, агресивні прояви, ірраціональне мислення, нелюбов до себе, то ці переживання загостряться і самотність стане єдиним виходом.

В. Зарицька також виділяє інтегральну специфіку емоційного інтелекту, яка проявляється в здатності розпізнавати, відслідковувати та регулювати власні й емоції інших, а також використовувати емоції для підвищення ефективності діяльності і вирішення проблем в спілкуванні. Вона робить акцент на те, що відносно високий рівень емоційного інтелекту відкриває перед індивідом безліч можливостей, до яких входить найважливіша – здатність легко зближуватися з людьми на емоційній основі. Таке зближення безсумнівно приводить до успішної соціалізації особистості, а також профілактики суб'єктивного почуття самотності.

На підставі теоретичного аналізу літератури ми встановили, що самотність – це комплексний психологічний явище. На сьогодні існують різні точки зору серед вчених. Деякі стверджують, що самотність негативно впливає на життя людини, тоді як інші аргументують позитивні аспекти цього відчуття. Автори, такі як Е. Фромм, М. Мід, Г.С. Салліван, Р.С. Немов, В.І. Кошкарлова, підтверджують руйнівний вплив самотності на особистість. Інші вчені висловлюють амбівалентний підхід до проблеми, а деякі, наприклад, А. Маслоу та К. Мустакас, стверджують, що відчуття самотності може сприяти внутрішньому самовдосконаленню.

Цікавим є висловлення Є.М. Заворотних, яка розрізняє три рівні самотності як суб'єктивного переживання: низький рівень, коли людина не відчуває себе ізольованою; комфортний середній рівень, який рідко виникає і не викликає особливого дискомфорту; та деструктивний рівень, який призводить до негативних почуттів та переживань.

Згідно з Є.М. Заворотних, відчуття самотності більше залежить від особистісних характеристик, аніж від кількості соціальних контактів. Самотність може бути розглядана як відчуття, стан, процес або відношення. Як відчуття, вона виникає, коли людина відчуває себе інакшою, утворюючи психологічний бар'єр у спілкуванні з іншими.

Дослідження також вказують на те, що найбільш вразлива група до відчуття самотності - підлітки. У цьому періоді формуються основні риси характеру та емоційних реакцій, які вплинуть на подальше життя. Підлітки можуть відчувати самотність, якщо не встановлюють емоційних зв'язків з однолітками або відчують непорозуміння зі сторони батьків.

Дослідження феномену емоційного інтелекту, проведені авторами, такими як Дж. Мейєр, П. Саловей, Д. Гоулман, Р. Бар-Он, Д. В. Люсін, підкреслюють важливість правильного сприйняття та передачі емоційних сигналів у відчутті самотності. Це особливо актуально в дитячому віці, де значущу роль у розвитку емоційного інтелекту відіграють батьки, визначаючи, як діти будуть взаємодіяти зі суспільством та вбудовувати соціальні зв'язки в майбутньому.

### **3.4 Проведення практичного дослідження та обґрунтування використання психодіагностичних методів**

Дане дослідження ґрунтується на теоретичних концепціях, що стосуються емоційного інтелекту, які розроблені відомими зарубіжними дослідниками, такими як Д. Гоулман, Дж. Мейєр, П. Саловей, Р. Бар-Он, а також вітчизняними ученими, зокрема О.М. Леонтьєвим, Д. Люсіним, І.М. Андрєєвою та М.А. Холодною. Окрім цього, враховані погляди на суб'єктивне відчуття самотності від І. Ялома, Е. Фромма і Дж. Янга, а також їхніх вітчизняних колег, таких як Є.М. Осін, О.В. Неумоевої-Колчеданцевої, Д.О. Леонтьєва та С.Г. Корчагіної.

Для вивчення зв'язку між емоційним інтелектом та суб'єктивним відчуттям самотності ми обрали вибірку з осіб у віці 14-15 років, які навчаються в дев'ятому

класі. Загальна кількість опитаних склала 48 осіб. Для проведення емпіричного дослідження ми підготували анкети, де кожна з п'яти методик розміщена на окремому аркуші паперу для зручності подальших обчислень; досліджувані вказували свої прізвище, ім'я та по батькові на кожному аркуші.

Тестування тривало 60 хвилин, включаючи оголошення інструкцій до кожної методики, час їхнього проходження, а також збір анкет і обговорення майбутніх зустрічей для повідомлення результатів дослідження та узгодження подальших кроків співпраці з окремими учасниками.

Для проведення емпіричного дослідження взаємозв'язку емоційного інтелекту та особистого відчуття самотності у підлітків, ми скористалися рядом методик:

1. Диференціальний опитувальник переживання самотності Є. М. Осіна та Д. О. Леонтьєва.
2. Опитувальник для визначення типу самотності С. Г. Корчагіної.
3. Опитувальник особистого відчуття самотності Д. Рассела та М. Фергюсона.
4. Опитувальник ЕМІн Д. Люсіна.
5. Тест емоційного інтелекту EQ Н. Холла

Диференціальний опитувальник переживання самотності представляє інноваційний психодіагностичний інструмент, розроблений для аналізу самотності як багатовимірного явища, що включає негативні та позитивні аспекти. Самотність, яку приймають і розвивають, може призводити до особистісного зростання, тоді як намагання уникати неї може гальмувати цей процес.

Методика була розроблена у 2013 році Є.М. Осіним та Д.О. Леонтьєвим і включає два варіанти – повний (40 питань) та короткий (24 питання) опитувальники, що дає можливість ретельніше вивчити самотність респондентів або використовувати методику для індивідуальної діагностики.

Опитувальник для визначення типу самотності С.Г. Корчагіної, представлений у 2008 році, дозволяє визначити тип самотності та глибину її

переживання. Методика включає чотири діагностичні шкали: загальний стан самотності, дифузну, відчужуючу та дисоційовану самотність.

Дифузна самотність виникає, коли людина ідентифікує себе з іншими і починає втрачати власну ідентичність. Відчужуюча самотність занурює людину у власний світ, обмежуючи соціальні контакти, тоді як дисоційована самотність, характеризується залежністю і частіше виникає у експресивних особистостей, може поглибитися до клінічної форми без вчасної допомоги.

Опросник суб'єктивного відчуття самотності, розроблений Д. Расселом та М. Фергюсоном у 1978 році, був адаптований Н.С. Водоп'яноюю. Цей метод призначений для визначення рівня особистого відчуття самотності та соціальної ізоляції у людини. Його основа полягає в уявленні про самотність як особистісний феномен, де почуття самотності розглядається як щось суб'єктивне, власне, індивідуальне і часто унікальне переживання. Одна з ключових ознак самотності - це відчуття повного занурення у внутрішній світ. Це почуття несхоже на інші, воно повністю поглиблює особистість, ізолює її від соціуму та направляє всі думки всередину себе.

Для визначення рівня емоційного інтелекту ми обрали тест ЕМІН Д. Люсіна, один з найвідоміших опитувальників у цій області. Автор спочатку розробив свою модель емоційного інтелекту і потім створив відповідний тест. Згідно з його визначенням, емоційний інтелект - це здатність ідентифікувати власні та чужі емоції, а також управляти ними. Автор також вводить поняття внутрішньоособистісного і міжособистісного емоційного інтелекту, які тісно пов'язані. Тест був розроблений на протязі кількох етапів, з першою версією, опублікованою в 2004 році, і останньою, доповненою версією в 2006 році, де надійність шкал була визнана високою. Вчений продовжує розширювати тест та вдосконалювати його. Автор стверджує, що його модель емоційного інтелекту відрізняється від інших, оскільки вона абсолютно незалежна від особистісних характеристик респондентів.

Щодо наступної методики визначення емоційного інтелекту, представленої



Є.П. Ільїним, варто зауважити, що вона є найбільш використовуваною серед дослідників. Н. Холл використовував цю методику для визначення основних можливостей розуміння внутрішніх переживань особистості, які відображаються у її емоційній сфері, а також для виявлення здатності людини керувати своїми та чужими емоціями на основі прийняття відповідних рішень. Методика спрямована на визначення рівня розвитку емоційної сфери особистості і ґрунтується на загальних теоретичних уявленнях про емоційний інтелект. Тест EQ Н. Холла складається з 30 тверджень і включає 5 шкал: емоційна обізнаність, управління своїми емоціями, самомотивація, емпатія та розпізнавання емоцій інших людей.

### **3.5 Методи математичної обробки**

Для встановлення зв'язку між емоційним інтелектом та суб'єктивним відчуттям самотності серед підлітків юнацького віку, ми використовували методи математичної статистики. По-перше, ми обрахували та проаналізували результати кожного з 48 респондентів за п'ятьма методиками (див. Додаток А). Потім ми внесли їхні дані до програми SPSS для подальшої математичної обробки.

На початку дослідження ми визначили тип розподілу даних, зокрема, чи вони є нормальними чи виявляють відхилення від нормального розподілу. Для цього використовувався критерій Колмогорова-Смірнова, який порівнює емпіричний розподіл з теоретичним або один емпіричний розподіл з іншим. Якщо ймовірність того, що розподіл відповідає нормальному, перевищує 0,05, це свідчить про нормальний розподіл. У випадку, коли це число менше 0,05, ми вважаємо результати неінформативними.

Після визначення виду розподілу ми перейшли до виявлення кореляційних зв'язків між показниками. Кореляція - це міра ступеня та напрямку зв'язку між значеннями двох змінних. У випадку, коли дані мають нормальний розподіл, ми використовували критерій Пірсона. Це параметричний показник для метричних змінних. У випадку, коли дані не мають нормального розподілу, ми

використовували критерій Спірмена, який є непараметричним показником для рангової шкали.

Крім того, для оцінки впливу тренінгу на емоційний інтелект окремої групи підлітків ми використовували t-тест для парних вибірок. Тестова величина, отримана з розподілу Стюдента, визначала ступінь впливу та зміни показників після тренінгу. Паралельно з цим ми проводили цей тест для контрольної групи, яка не брала участь у програмі корекції емоційного інтелекту, для чистоти експерименту.

#### **РОЗДІЛ 4**

# АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ ТА ПОСТАНОВКА ЗАВДАННЯ

## 4.1 Інформаційна модель

Для реалізації поставленої задачі першим кроком було розроблено схеми функціонування веб-платформи.

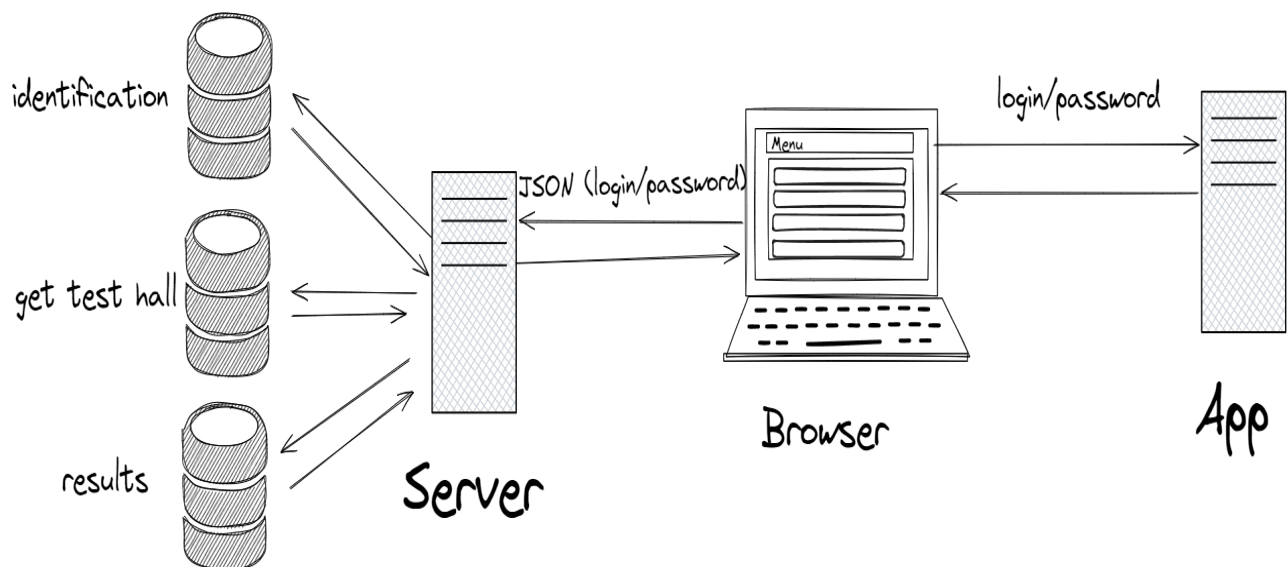


Рис. 4.1. Схема

## 4.2 Тестування в життєвому циклі програмного забезпечення

Файл 'Header.js' реалізує React-компонент, який відповідає за відображення верхньої частини веб-сторінки, також відомої як "шапка" (header). Давайте проаналізуємо код і запропонуємо назву файлу "Header.js":

```

4   class header extends React.Component {
5   >   constructor(props) {...
8   }
9
10  componentDidMount() {
11    var value = null;
12    value = JSON.parse(sessionStorage.getItem("user"));
13    if (value) {
14      this.setState({ isLogin: true, user: value });
15    }
16  }
17
18 >  render() {...
37  }
38 }
39
40 > function Authorized({ children, user }) {...
56 }
57
58 > function Unauthorized({ children }) {...
66 }
67
68 const sign_out = () => {
69   sessionStorage.removeItem("user");
70   window.location.reload();
71 };
72
73 export default header;

```

Рис. 4.2. Файл Header.js

Основні риси:

#### 1. Керування станом:

- 1) Компонент містить локальний стан, представлений об'єктом з полями `isLogin` та `user`.
- 2) `isLogin` відповідає за відстеження стану входу користувача.
- 3) `componentDidMount` використовується для перевірки наявності користувача в сеансі та оновлення стану.

#### 2. Навігаційне меню:

- 1) Навігаційне меню містить посилання на "Головну" та "Тести".

#### 3. Умовне рендеринг:

- 1) Залежно від стану входу користувача відображається або компонент `Authorized`, або `Unauthorized`.

#### 4. Компонент `Authorized`:

- 1) Відображається, якщо користувач увійшов в систему.
- 2) Показує привітання з ім'ям користувача та посилання для виходу.

## 5. Компонент `Unauthorized`:

- 1) Відображається, якщо користувач не увійшов в систему.
- 2) Пропонує посилання для входу.

## 6. Функція виходу (`sign\_out`):

- 1) Видаляє інформацію про користувача з сеансу та перезавантажує сторінку.

### 4.3 Головна сторінка сайту

Компонент головної сторінки веб-сайту з використанням React. Містить інформацію про сайт, його місію, особливості та інструкції для користувачів. Фон сторінки змінюється за допомогою зображення.

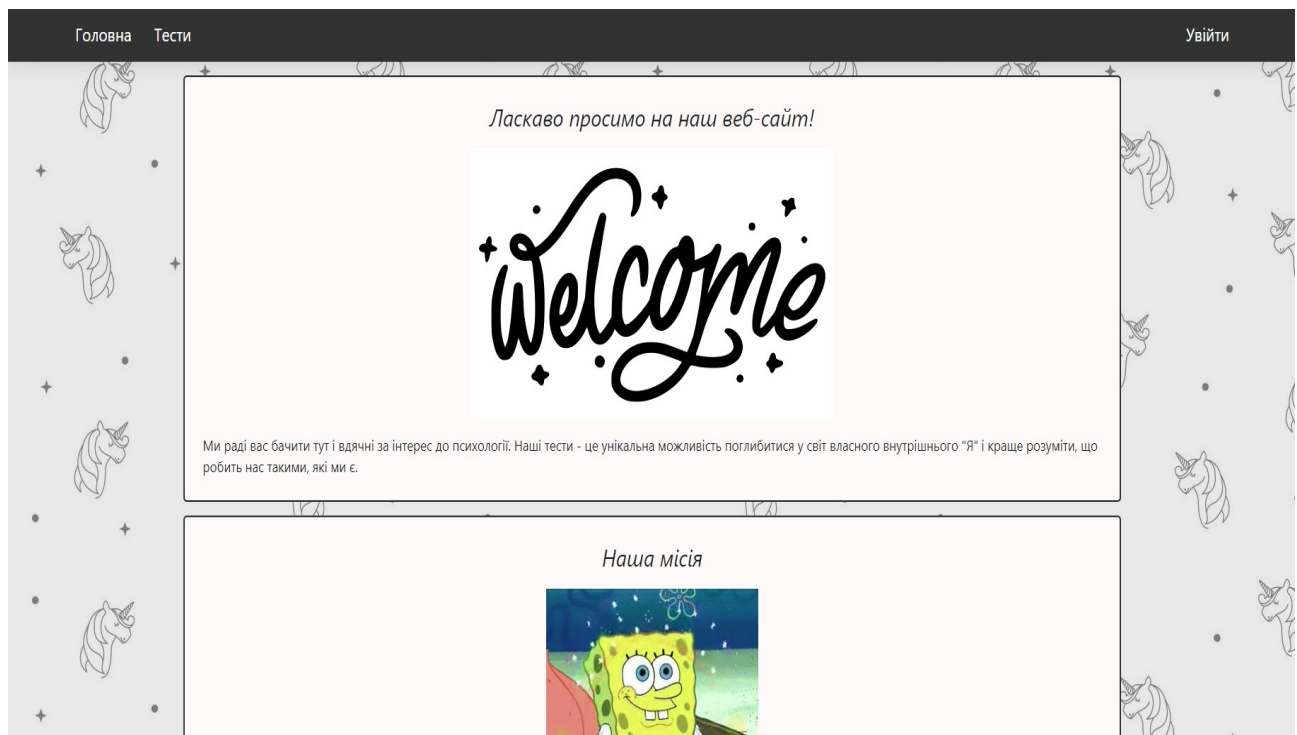


Рис. 4.3. Файл home.js

Основні елементи цього коду:

1. Зображення та стилізація: Код містить посилання на зображення для фону (wallpaper.png), логотипу (welcome.png), та анімованого

зображення (rainbow.gif). Також імпортований файл стилів home.css, ймовірно, містить стилі для цього компонента.

2. React-класовий компонент: Компонент наслідується від React.Component і має метод render, який відповідає за відображення компонента.
3. Вміст головної сторінки: Головна сторінка складається з кількох блоків (<div className='post'>), кожен з яких містить інформацію про веб-сайт. Кожен блок має назву, зображення та текстовий опис.
4. Зміна фону сторінки: Відзначається зміна фону сторінки через властивість backgroundImage в стилі document.body. Фон встановлюється за допомогою посилання на зображення wallpaper.

#### 4.4 Галерея тестів на емоційний інтелект

Цей файл з назвою gallery.js представляє собою React-компонент для галереї тестів на емоційний інтелект (EQ). Назва "бібліотека" (library) може бути використана для цього компонента, оскільки він відображає кілька тестів для користувачів.

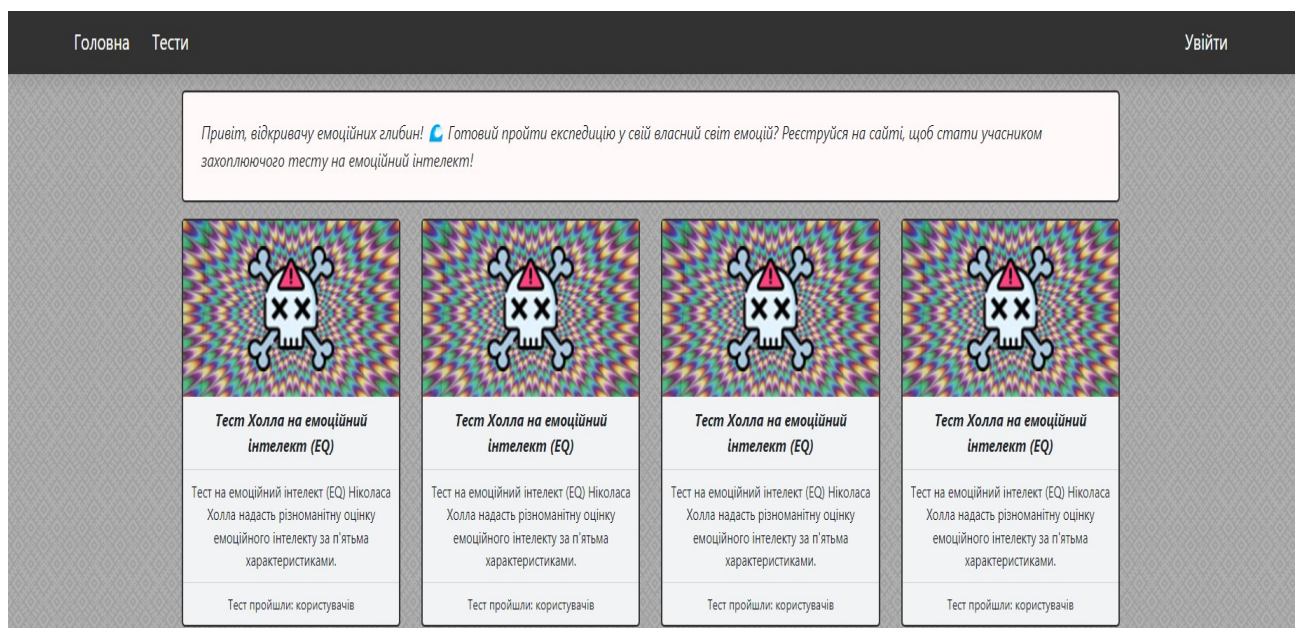


Рис. 4.4. Файл gallery.js

Основні аспекти коду:

1. Зображення та стилізація: Компонент використовує зображення та CSS-стилі для оформлення. Фонове зображення (wallpaper.png) встановлюється як тло для всього документу.
2. Стан та авторизація: Є стан компонента, який визначає, чи виконаний вхід в систему (авторизація користувача). Залежно від цього стану, відображається відповідний контент.
3. Компоненти тестів: Є кілька блоків, які представляють тести. Кожен блок включає зображення, назву тесту, опис та лічильник пройдених тестів.
4. Повідомлення для користувача: Якщо користувач не увійшов в систему, відображається повідомлення зі спробою вперше залучити його до участі в тестах.
5. Функціональні компоненти: Є три функціональні компоненти: `Message_for_the_user`, `Authorized` та `Unauthorized`. Вони використовуються для відображення повідомлення для користувача, авторизованого зображення для тесту та неавторизованого зображення відповідно.

#### **4.5 Сторінка с тестом Н. Холла**

Файл `test_hall.js` містить React-компонент, призначений для відображення та відправлення результатів тестування. Компонент використовується для взаємодії з сервером, отримуючи та відправляючи дані тестування.

**Вопрос 26: Я здатний поліпити настрої інших людей.**

Полностью не згоден     В основном согласен     Отчасти согласен     Отчасти не согласен     В основном не согласен     Полностью не согласен

**Вопрос 27: Зі мною можна порадитися з питань відносин між людьми.**

Полностью не згоден     В основном согласен     Отчасти согласен     Отчасти не согласен     В основном не согласен     Полностью не согласен

**Вопрос 28: Я добре налаштовуюся на емоції інших людей.**

Полностью не згоден     В основном согласен     Отчасти согласен     Отчасти не согласен     В основном не согласен     Полностью не согласен

**Вопрос 29: Я допомагаю іншим використовувати їх спонукання для досягнення особистих цілей.**

Полностью не згоден     В основном согласен     Отчасти согласен     Отчасти не согласен     В основном не согласен     Полностью не согласен

**Вопрос 30: Я можу легко відключитися від переживання неприємностей.**

Полностью не згоден     В основном согласен     Отчасти согласен     Отчасти не согласен     В основном не согласен     Полностью не согласен

Отправить

Рис. 4.5. Файл test\_hall.js

#### Опис коду:

1. У компоненті визначено стан, який містить дані тесту, а також відповіді на кожне запитання тесту.
2. При завантаженні компоненту викликається метод `componentDidMount`, який виконує запит на сервер за допомогою `fetch` для отримання даних тесту та оновлення стану компоненту.
3. Для обробки вибору відповідей використовується метод `handleRadioChange`, який оновлює стан відповідно до обраної опції.
4. При надсиланні форми викликається метод `handleSubmit`, який



відправляє відповіді на сервер для подальшої обробки.

5. Відображення тестових запитань та вибору відповідей відбувається в блоку `render`, де дані з серверу використовуються для створення відповідних форм інтерфейсу.
6. Кожне запитання має шість варіантів відповідей, представлених у вигляді радіо-кнопок.
7. Кнопка "Отправить" вказує на можливість відправки введених відповідей на сервер

#### 4.6 Компонент ідентифікації користувача

Цей код - це компонент React з інтерфейсом для ідентифікації користувача. Файл називається "identification.js".

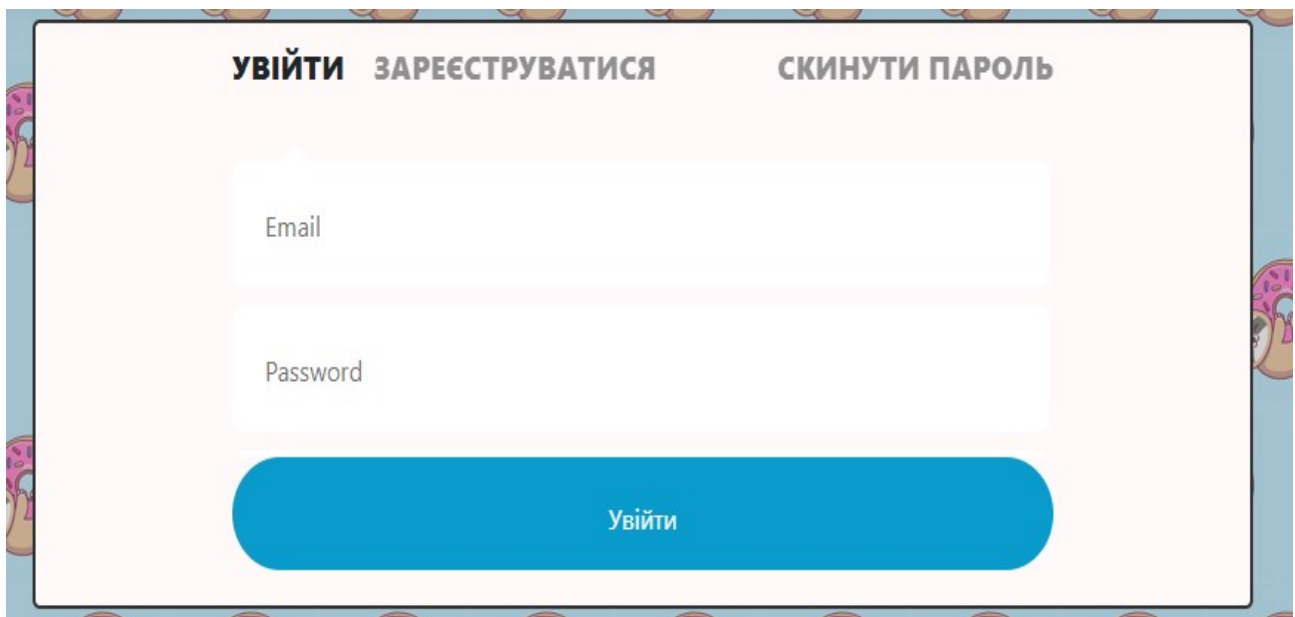


Рис. 3.6. Файл identification.js

Клас "Identification", який є підкласом `React.Component`. Він містить стан компонента з полями "radio", "login", "password" та "repass", які відображають вибір типу дії (увійти, зареєструватися, скинути пароль), логін, пароль та повторний пароль відповідно.

Розмітка форми для введення даних користувача, яка містить поля для вибору типу дії, введення електронної пошти, паролю та повторення паролю.

Вивід кнопки для відправки даних форми.

## 4.7 Сам сервер

Цей код представляє собою сервер, який використовується для взаємодії з базою даних за допомогою Express та Cors у середовищі програмування JavaScript з використанням Node.js. Сервер прослуховує порт 3030 і забезпечує API для автентифікації, реєстрації користувачів, скидання паролів, отримання даних тестового зали та відправки відповідей на тестовий зали.

Кожен ендпоінт обробляє дані від клієнта, передає їх функціям модуля "database" для виконання відповідної операції з базою даних і повертає результати клієнту. У разі виникнення помилок під час обробки вони обробляються, і відповідь містить об'єкт з полем "error". У протилежному випадку результати операції відправляються у вигляді об'єкта з полем "name".

```
const database = require("./database");

const express = require("express");

const cors = require("cors");

const app = express();

app.use(cors());

/***** Запуск сервера *****/

app.listen(3030, () => {

  console.log("Я живой!!");

});
```

```
/****** Работа с таблицей `users` *****/
```

```
app.post("/sign_in", express.json(), (request, results) => {
```

```
  database.sign_in(request.body, (error, r) => {
```

```
    if (error) {
```

```
      results.send({ error: error });
```

```
      return;
```

```
    }
```

```
    results.send({ name: r });
```

```
  });
```

```
});
```

```
app.post("/sign_up", express.json(), (request, results) => {
```

```
  database.sign_up(request.body, (error, r) => {
```

```
    if (error) {
```

```
      results.send({ error: error });
```

```
      return;
```

```
    }
```

```
    results.send({ name: r });
```

```
  });
```

```
});
```

```
app.post("/reset", express.json(), (request, results) => {
```

```
database.reset(request.body, (error, r) => {  
  if (error) {  
    results.send({ error: error });  
    return;  
  }  
  
  results.send({ name: r });  
});  
});  
  
app.get("/test_hall", (request, results) => {  
  database.test_hall((error, r) => {  
    if (error) {  
      results.send({ error: error });  
      return;  
    }  
  
    results.send({ name: r });  
  });  
});  
  
app.post("/answers_test_hall", express.json(), (request, results) => {  
  database.answers_test_hall(request.body, (error, r) => {
```

```
if (error) {  
    results.send({ error: error });  
    return;  
}  
  
results.send({ name: r });  
});  
});
```

#### **4.8 Взаємодія з базою даних**

Цей код - це набір функцій для взаємодії з базою даних MySQL у зв'язку з користувачами та тестовою залюю. Основні функції виконують дії, такі як вхід користувача (`sign_in`), реєстрація нового користувача (`sign_up`), та скидання паролю (`reset`). Також є функції для отримання даних з таблиці `test_hall` та обробки відповідей до тестового залу (`test_hall` та `answers_test_hall` відповідно).

Основний клас `Database` використовується для взаємодії з базою даних MySQL. Він містить метод `database_access`, який встановлює підключення до бази даних, виконує SQL-запит та передає результат у вигляді виклику зазначеної функції зворотного виклику (`callback`).

Для забезпечення безпеки коду, слід використовувати параметризовані запити для уникнення SQL-ін'єкцій. Також, потрібно забезпечити правильне оброблення помилок та відключення підключення до бази даних після виконання запиту.

Загалом, код здається функціональним, але йому можна було б додати деякі заходи безпеки та оптимізації.

```

const mysql = require("mysql");

module.exports = {
  /******* Работа с таблицей `users` *****/
  sign_in: (user, callback) => {
    var database = new Database(`SELECT * FROM users where
login='${user.login}' and password='${user.password}'`);
    database.database_access(callback);
  },
  sign_up: (user, callback) => {
    var database = new Database(`INSERT INTO users(login, password)
VALUES('${user.login}', '${user.password}')`);
    database.database_access(callback);
  },
  reset: (user, callback) => {
    var database = new Database(`UPDATE users SET password='123456'
WHERE login='${user.login}'`);
    database.database_access(callback);
  },
  /******* Работа с таблицей `test_hall` *****/
  test_hall: (callback) => {
    var database = new Database("SELECT * FROM `test_hall`");
    database.database_access(callback);
  },
  answers_test_hall: (answers, callback) => {
    //var database = new Database(`SELECT * FROM users where
login='${user.login}' and password='${user.password}'`);
    //database.database_access(callback);
    const answer = answers.answers;

```

```
let array = [];  
  
for (let number in answer) {  
  if (answer.hasOwnProperty(number)) {  
    let question = answer[number];  
    array.push(question["question" + number]);  
  }  
}  
},  
};
```

```
class Database {  
  constructor(request) {  
    this.request = request;  
  }  
}
```

```
/* Обращение к базе данных */
```

```
database_access(callback) {  
  var connection = mysql.createConnection({ host: "localhost", user: "root",  
password: "", database: "database" });
```

```
  connection.connect((error) => {  
    if (error) {  
      callback(error);  
    }  
  })
```

```
  connection.query(this.request, function (error, results) {  
    if (error) {  
      callback(error);  
    } else {
```

```
    callback(null, results);  
  }  
});
```

```
    connection.end();  
  });  
}  
}
```



## ВИСНОВКИ

Загальною метою дипломної роботи є не лише розробка функціонального веб-додатку, а й створення основи для подальших досліджень та розвитку у галузі вивчення та застосування емоційного інтелекту у веб-розробці.

В результаті дослідження та розробки веб-додатку з аналізом рівня емоційного інтелекту користувачів за допомогою бібліотеки React можна визначити, що такий підхід виявляється ефективним та потенційно корисним для подальшого розвитку в галузі психології та інтерфейсного дизайну.

Використання бібліотеки React виявилось не тільки зручним, але й дозволило значно полегшити розробку та підтримку веб-додатку. Компонентний підхід React дозволяє ефективно управляти станом додатку та забезпечує швидку реакцію на зміни, що важливо для аналізу емоцій користувачів.

Розроблений веб-додаток успішно здійснює аналіз рівня емоційного інтелекту користувачів, базуючись на їхньому поведінкові та взаємодії з додатком. Це відкриває можливості для подальших досліджень в області розуміння та оптимізації інтерфейсів для кращого взаєморозуміння та задоволення користувачів.

Висновки дозволяють припустити, що подальший розвиток веб-додатків, спрямованих на аналіз емоційного інтелекту користувачів, має великий потенціал у сферах психології, маркетингу, та інтерфейсного дизайну. Такі інструменти можуть використовуватися для покращення якості взаємодії між користувачами та різноманітними веб-сервісами.

Процес розробки веб-додатку підтверджує важливість планування та адаптації підходів до конкретних вимог та потреб користувачів. Застосування принципів React сприяло стабільності та зручності розробки, адже вони дозволили зосередитися на функціональності, що пов'язана з аналізом емоцій, і виключити зайві технічні труднощі.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Айвазян Л.Ю. Особливості прояву самотності як психологічного явища. Вісник Національного університету оборони України, 2013 №5 (36). С. 147- 151.
2. Рыжов Д.М. Современные представления об эмоциональном интеллекте. Вестник ЧГПУ им. И.Я Яковлева. 2012. № 1 (73). Ч. 2. С. 142-147.
3. Коваленко О.Г. Суб'єктивне відчуття самотності осіб похилого віку. Проблеми сучасної психології, 2013. Вип. 19. С. 239-250.
4. Железнякова. Ю.В. Самотність у підлітковому віці як соціальнопсихологічна проблема. Вісник Дніпропетровського університету. Серія «Педагогіка і психологія», 2013. Вип. 19. С. 67-74.
5. Андреева И.Н. Концептуальное поле понятия «эмоциональный интеллект». Вопросы психологии.. 2009. № 4. С. 131-141.
6. Никоненко Ю.В. Теоретичний аналіз проблеми дослідження самотності в сучасній психології. Актуальні проблеми психології, 2012. Вип. 35. С. 177-185.
7. Люсин Д.В. Современные представления об эмоциональном интеллекте // Социальный интеллект: Теория, измерение, исследования / под ред. Д.В. Люсина, Д.В. Ушакова. - М.: Институт психологии РАН, 2012.
8. Болтунова А.И. Особенности межличностного общения среди подростков. Молодой ученый. 2014. № 1. С. 492-494.
9. Ялом И. Экзистенциальная психотерапия. М.: Независимая фирма «Класс», 1999. 576 с.
10. Айвазян Л.Ю. Взаємозв'язок суб'єктивного відчуття самотності і розвитку особистості. Проблеми сучасної психології, 2011. Вип 13. С. 12-21.
11. Тримаскина И.В. Тренинг эмоционального интеллекта и развития личной эффективности. СПб.: Речь, 2010. 157 с.
12. Выготский Л.С. Мышление и речь. СПб : Питер, 2017. 432 с.
13. Могиляста С.М. Аналіз теоретичних підходів до дослідження емоційного інтелекту особистості. Актуальні проблеми психології. Психологія обдарованості. 2018. Вип. 14. Том VI. С. 362-369

14. Подофей С.О. Эмоциональный интеллект та особливості його становлення. Науковий вісник Херсонського державного університету. Серія “Психологічні науки, 2019 Вип. 3. Том.І. С. 260-265
15. Андреева И.Н. Эмоциональный интеллект как феномен современной психологии. – Новополюцк: ПГУ, 2011. 388 с
16. Леонтьев А.Н. Деятельность, сознание, личность. М.: Смысл, 2005. 352 с.
17. Слободчиков И.М. Современные исследования переживания одиночества. Психологическая наука и образование. 2007. №3. С. 27-35.
18. Железнякова Ю.В. Предпосылки развития чувства. Личность, семья и общество: вопросы педагогики и психологии : сб. ст. по матер. XXXVI междунар. науч.-практ. конф. Новосибирск: СибАК, 2014. № 1(36).
19. Овсяннікова В.В. Суб'єктивні переживання самотності підлітків. Проблеми сучасної психології, 2012. № 2. С. 99-106.
20. Мантикова А.В., Тен Ю.В. Состояние подросткового одиночества в контексте личностных особенностей. Наука. Мысль. 2014. №12. С. 18-33.
21. Андреева И.Н. Эмоциональный интеллект в структуре индивидуальности. Философия и социальные науки. 2010. № 3. С. 8-12.
22. Фетискин Н.П., Козлов В.В., Мануйлов Г.М. Социально-психологическая диагностика развития личности и малых групп. М.: Изд-во Института Психотерапии. 2002.
23. Авдеева Н.Н. Большой психологический словарь. 2003. 666 с.
24. Моисеева Т.А. Особенности субъективного чувства одиночества в подростковом возрасте. Молодой ученый. 2019. № 4 (242). С. 158-161.
25. Белкина О.А. Феноменология эмоционального интеллекта. Вестник Московского городского педагогического университета. Серия: Педагогика и психология. 2009. № 4. С. 47-62.
26. Остапенко Р.И. Математические основы психологии: учебно-методическое пособие для студентов и аспирантов психологических и педагогических специальностей вузов. Воронеж. : ВГПУ, 2010. 76 с.
27. Ильин Е.И. Эмоции и чувства. СПб Питер, 2001.

28. Янг Д.И. Одиночество, депрессия и когнитивная терапия: теория и ее применение. Лабиринты одиночества: сборник статей / Под ред. Н.Е. Покровского. М.: Прогресс, 1989. 593с.
29. “Flying your drone safely and legally.” URL: <https://tc.canada.ca/en/aviation/drone-safety/flying-your-drone-safelylegally>. (Дата звернення 17.11.2022).
30. Н. Tian, Т. Wang, Y. Liu, X. Qiao, and Y. Li, “Computer vision technology in agricultural automation —A review,” *Information Processing in Agriculture*. 2020, doi: 10.1016/j.inpa.2019.09.006.
31. “Gatwick Airport drone attack: Police have ‘no lines of inquiry’ - BBC News.” URL: <https://www.bbc.com/news/uk-england-sussex-49846450>.
32. P. Viola and M. Jones, “Rapid object detection using a boosted cascade of simple features,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2001, vol. 1, doi: 10.1109/cvpr.2001.990517.
33. “Drone law in the UK | Nottinghamshire Police.” URL: <https://www.nottinghamshire.police.uk/advice/drone-law-uk>. (Дата звернення 17.11.2022).
34. “GitHub: The top 10 programming languages for machine learning - TechRepublic.” URL: <https://www.techrepublic.com/article/github-the-top10-programming-languages-for-machine-learning/>. (Дата звернення 09.04.2022).
35. R. Girshick, J. Donahue, Т. Darrell, and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2014, doi: 10.1109/CVPR.2014.81.
36. T. W. Hui, X. Tang, and C. C. Loy, “A lightweight optical flow CNN – Revisiting data fidelity and regularization,” *arXiv*. 2019, doi: 10.1109/tpami.2020.2976928.
37. “models/tf1\_detection\_zoo.md at master · tensorflow/models.” URL: [https://github.com/tensorflow/models/blob/master/research/object\\_detection/g3doc/tf1\\_detection\\_zoo.md](https://github.com/tensorflow/models/blob/master/research/object_detection/g3doc/tf1_detection_zoo.md) (Дата звернення 15.04.2022).

38. "TFRecord and tf.train.Example | TensorFlow Core." URL: [https://www.tensorflow.org/tutorials/load\\_data/tfrecord](https://www.tensorflow.org/tutorials/load_data/tfrecord). (Дата звернення 17.04.2022.)
39. "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," arXiv. 2018. S. Bai, J. Z. Kolter, and V. Koltun.
40. "A Practical Guide to Support Vector Classification," BJU Int., 2008. C.-J. L. Chih-Wei Hsu, Chih-Chung Chang.
41. Y. LeCun et al., "Backpropagation Applied to Handwritten Zip Code Recognition," *Neural Comput.*, vol. 1, no. 4, pp. 541–551, Dec. 1989, doi: 10.1162/neco.1989.1.4.541.
42. C. L. Zitnick and P. Dollár, "Edge boxes: Locating object proposals from edges," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2014, doi: 10.1007/978-3-319-10602-1\_26.
43. "Gang who flew drones carrying drugs into prisons jailed - BBC News." URL: <https://www.bbc.com/news/uk-england-45980560>. (Дата звернення 17.12.2020).
44. N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proceedings - 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005, 2005*, vol. I, pp. 886–893, doi: 10.1109/CVPR.2005.177.
45. "SSD: Single Shot MultiBox Detector," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 9905 LNCS, pp. 21–37, Dec. 2015, doi: 10.1007/978-3-319-46448-0\_2. W. Liu et al.
46. "Selective search for object recognition," *Int. J. Comput. Vis.*, 2013, doi: 10.1007/s11263-013-0620-5. J. R. R. Uijlings, K. E. A. Van De Sande, T. Gevers, and A. W. M. Smeulders.
47. "Deep sparse rectifier neural networks," in *Journal of Machine Learning Research*, 2011. X. Glorot, A. Bordes, and Y. Bengio.

## ЛІСТІНГ ПРОГРАМИ - my-app

**header.js**

```
import React from "react";
import "./header.css";

class header extends React.Component {
  constructor(props) {
    super(props);
    this.state = { isLogin: false, user: null };
  }

  componentDidMount() {
    var value = null;
    value = JSON.parse(sessionStorage.getItem("user"));
    if (value) {
      this.setState({ isLogin: true, user: value });
    }
  }

  render() {
    return (
      <nav className='navbar'>
        <div className='left_navbar'>
          <ul>
            <li className='navbar_item'>
              <a href='/'>ГОЛОВНА</a>
            </li>
            <li className='navbar_item'>
              <a href='/gallery'>Тести</a>
            </li>
          </ul>
        </div>
      </nav>
    );
  }
}
```

```

    <div className='right_navbar'>
      <ul>{this.state.isLogin ? <Authorized user={this.state.user} /> : <Unauthorized
/>}</ul>
    </div>
  </nav>
);
}
}

```

```

function Authorized({ children, user }) {
  return (
    <React.Fragment>
      <li className='navbar_item'>
        <a href='#'>
          <span>Доброго дня, </span>
          {user.nickname}
        </a>
      </li>
      <li className='navbar_item'>
        <a href='#' onClick={sign_out}>
          Вийти
        </a>
      </li>
    </React.Fragment>
  );
}

```

```

function Unauthorized({ children }) {
  return (
    <React.Fragment>
      <li className='navbar_item'>
        <a href='/identification'>Увійти</a>
      </li>
    </React.Fragment>
  );
}

```

```
);  
}
```

```
const sign_out = () => {  
  sessionStorage.removeItem("user");  
  window.location.reload();  
};
```

```
export default header;
```

.....

## **home.js**

```
import React from "react";
```

```
/* image */
```

```
import wallpaper from "./wallpaper.png";
```

```
import rainbow from "./rainbow.gif";
```

```
import welcome from "./welcome.png";
```

```
/* css */
```

```
import "./home.css";
```

```
class home extends React.Component {
```

```
  render() {
```

```
    document.body.style.backgroundImage = "url(" + wallpaper + ")";
```

```
    return (
```

```
      <>
```

```
        <div className='page'>
```

```
          <div className='container'>
```

```
            <div className='post'>
```

```
              <div className='name'>
```

```
                <span>Ласкаво просимо на наш веб-сайт!</span>
```

```
              </div>
```

```
              <img src={welcome} alt='your image' />
```



```
<div className='text'>
```

```
  <span>Ми раді вас бачити тут і вдячні за інтерес до психології. Наші тести - це  
унікальна можливість поглибитися у світ власного внутрішнього "Я" і краще розуміти, що робить  
нас такими, які ми є.</span>
```

```
</div>
```

```
</div>
```

```
<div className='post'>
```

```
  <div className='name'>
```

```
    <span>Наша місія</span>
```

```
  </div>
```

```
  <img src={rainbow} alt='your image' width='300px' />
```

```
  <div className='text'>
```

```
    <span>
```

```
      Ми прагнемо створити простий та натхненний шлях до самопізнання. У світі  
швидкого темпу життя іноді ми втрачаємо контакт з собою. Наша місія - допомогти вам  
відновити цей контакт, надаючи захопливі та інформативні тести, що ґрунтуються  
на останніх дослідженнях у галузі психології.
```

```
    </span>
```

```
  </div>
```

```
</div>
```

```
<div className='post'>
```

```
  <div className='name'>
```

```
    <span>Що робить нас особливими ✨</span>
```

```
  </div>
```

```
  <div className='text'>
```

```
    <span>
```

```
      Ми приділяємо особливу увагу якості наших тестів і їх науковому ґрунту.  
Наша команда експертів у галузі психології ретельно розробляє кожен тест, враховуючи  
різноманіття людського досвіду. Ми впевнені, що розуміння себе - ключ до кращої
```

```
      взаємодії з навколишнім світом. 🌍🔍
```

```
    </span>
```

```
  </div>
```

```
</div>
```

```

<div className='post'>
  <div className='name'>
    <span>Як користуватися сайтом 🌐</span>
  </div>
  <div className='text'>
    <span>
      Просто пройдіть наші захоплюючі тести, відкрийте для себе нові дивовижні
      аспекти своєї особистості та пориньте у захоплюючий світ психології. Дякуємо за вибір нашого
      сайту. Давайте разом вирушимо у захоплюючу подорож світом свідомості!
      🚀 ✨
    </span>
  </div>
</div>
</div>
</div>
</div>
</div>
</div>
</div>
</div>
</div>
);
}
}

```

```
export default home;
```

```
.....
```

### **gallery.js**

```
import React from "react";
```

```
/* image */
```

```
import wallpaper from "./wallpaper.png";
```

```
import phaser from "./icons/phaser.png";
```

```
import point from "./icons/point.png";
```

```
/* css */
```

```
import "./gallery.css";
```

```

class library extends React.Component {
  constructor(props) {
    super(props);
    this.state = { isLogin: false, user: null };
  }

  componentDidMount() {
    var value = null;
    value = JSON.parse(sessionStorage.getItem("user"));
    if (value) {
      this.setState({ isLogin: true, user: value });
    }
  }

  render() {
    document.body.style.backgroundImage = "url(" + wallpaper + ")";

    return (
      <div className='page'>
        {this.state.isLogin ? null : <Message_for_the_user />}

        <div class='container'>
          <div class='row'>
            <div class='col-md-3'>
              <div className='testing'>
                <div className='image'>{this.state.isLogin ? <Authorized /> : <Unauthorized
/>}</div>

                <div className='test_name'>Тест Холла на емоційний інтелект (EQ)</div>
                <hr className='hr' />
                <div className='description'>Тест на емоційний інтелект (EQ) Ніколаса Холла
надасть різноманітну оцінку емоційного інтелекту за п'ятьма характеристиками.</div>
                <hr className='hr' />
                <div className='counter'>Тест пройшли: {} користувачів</div>
              </div>
            </div>
          </div>
        </div>
      </div>
    );
  }
}

```

```
</div>
```

```
<div class='col-md-3'>
```

```
<div className='testing'>
```

```
<div className='image'>{this.state.isLogin ? <Authorized /> : <Unauthorized
```

```
/>}</div>
```

```
<div className='test_name'>Тест Холла на емоційний інтелект (EQ)</div>
```

```
<hr className='hr' />
```

```
<div className='description'>Тест на емоційний інтелект (EQ) Ніколаса Холла  
надасть різноманітну оцінку емоційного інтелекту за п'ятьма характеристиками.</div>
```

```
<hr className='hr' />
```

```
<div className='counter'>Тест пройшли: {} користувачів</div>
```

```
</div>
```

```
</div>
```

```
<div class='col-md-3'>
```

```
<div className='testing'>
```

```
<div className='image'>{this.state.isLogin ? <Authorized /> : <Unauthorized
```

```
/>}</div>
```

```
<div className='test_name'>Тест Холла на емоційний інтелект (EQ)</div>
```

```
<hr className='hr' />
```

```
<div className='description'>Тест на емоційний інтелект (EQ) Ніколаса Холла  
надасть різноманітну оцінку емоційного інтелекту за п'ятьма характеристиками.</div>
```

```
<hr className='hr' />
```

```
<div className='counter'>Тест пройшли: {} користувачів</div>
```

```
</div>
```

```
</div>
```

```
<div class='col-md-3'>
```

```
<div className='testing'>
```

```
<div className='image'>{this.state.isLogin ? <Authorized /> : <Unauthorized
```

```
/>}</div>
```

```
<div className='test_name'>Тест Холла на емоційний інтелект (EQ)</div>
```

```
<hr className='hr' />
```

```
<div className='description'>Тест на емоційний інтелект (EQ) Ніколаса Холла  
надасть різноманітну оцінку емоційного інтелекту за п'ятьма характеристиками.</div>
```

```
<hr className='hr' />
```

```
<div className='counter'>Тест пройшли: {} користувачів</div>
```

```

        </div>
      </div>
    </div>
  </div>
</>
);
}
}

```

```
function Message_for_the_user({ children }) {
```

```
  return (
```


```
    <React.Fragment>
```

```
      <div class='container'>
```

```
        <div class='post'>
```

```
          <div className='text'>
```

```
            <span className='message_for_the_user'>Привіт, відкривачу емоційних глибин!
```

 Готовий пройти експедицію у свій власний світ емоцій? Реєструйся на сайті, щоб стати учасником захоплюючого тесту на емоційний інтелект!</span>

```
          </div>
```

```
        </div>
```

```
      </div>
```

```
    </React.Fragment>
```

```
  );
```

```
}
```

```
function Authorized({ children }) {
```

```
  return (
```

```
    <React.Fragment>
```

```
      <a href='test_hall'>
```

```
        <img src={phaser} alt='your image' />
```

```
      </a>
```

```
    </React.Fragment>
```

```
  );
```

```
}
```

```
function Unauthorized({ children }) {
  return (
    <React.Fragment>
      <img src={phaser} alt='your image' />
    </React.Fragment>
  );
}
```

```
export default library;
```

```
.....
```

```
test_hall.js
```

```
import React from "react";
```

```
/* css */
```

```
import "./test_hall.css";
```

```
class home extends React.Component {
```

```
  constructor(props) {
```

```
    super(props);
```

```
    this.state = {
```

```
      data: [],
```

```
    };
```

```
  }
```

```
  componentDidMount() {
```

```
    fetch("http://localhost:3030/test_hall")
```

```
      .then((response) => response.json())
```

```
      .then((data) => {
```

```
        this.setState({ data: data.name });
```

```
      });
```

```
  }
```

```

handleRadioChange = (questionId, event) => {
  const { name, value } = event.target;
  this.setState((prevState) => ({
    answers: {
      ...prevState.answers,
      [questionId]: { ...prevState.answers[questionId], [name]: value },
    },
  }));
};

```

```

handleSubmit = (event) => {

```

```

  //console.log(this.state.answers);

```

```

  const answers = { answers: this.state.answers };

```

```

  // fetch("http://localhost:3030/answers_test_hall", { method: "POST", headers: { "Content-
Type": "application/json" }, body: JSON.stringify(answers) })

```

```

  fetch("http://localhost:3030/answers_test_hall", { method: "POST", headers: { "Content-
Type": "application/json" }, body: JSON.stringify(answers) })

```

```

  .then((res) => res.json())

```

```

  .then((data) => {

```

```

    //sessionStorage.setItem("user", JSON.stringify(data.name[0]));

```

```

    //window.location = "/";

```

```

  });

```

```

  event.preventDefault();

```

```

};

```

```

render() {

```

```

  return (

```

```

    <

```

```

      <div className='page'>

```

```

        <div className='container-fluid'>

```

```

          <form onSubmit={this.handleSubmit}>

```

```

            <div className='submit'>

```

```

              <button type='submit' className='btn btn-primary'>

```

Отправить

</button>

</div>

{this.state.data.map((post) => (

<div className='post\_test' key={post.id}>

<div className='test\_hall'>

<div className='question'>

<span>

Вопрос {post.id}: {post.question}

</span>

</div>

<div className='answer'>

<div className='row'>

{[1, 2, 3, 4, 5, 6].map((index) => (

<div className='col-md-2' key={index}>

<input type='radio'

name={`question\${post.id}`}

id={`answer\_\${index}\_\${post.id}`}

className='form-check-input'

value={`option\${index}`}

onChange={(e) => this.handleRadioChange(post.id, e)} />

<label

htmlFor={`answer\_\${index}\_\${post.id}`}>{post[ `answer\_\${index}` ]}</label>

</div>

))}

</div>

</div>

</div>

</div>

))}

<div className='submit'>

<button type='submit' className='btn btn-primary'>

Отправить

</button>

</div>

</form>



```
        </div>
      </div>
    </>
  );
}
}
```

```
export default home;
```

```
.....
```

### **identification.js**

```
import React from "react";
```

```
import request from "./request.js";
```

```
/* image */
```

```
import wallpaper from "./wallpaper.png";
```

```
/* css */
```

```
import "./identification.css";
```

```
class Identification extends React.Component {
```

```
  constructor(props) {
```

```
    super(props);
```

```
    this.state = {
```

```
      radio: "signin",
```

```
      login: "",
```

```
      password: "",
```

```
      repass: "",
```

```
    };
```

```
    this.handleSubmit = this.handleSubmit.bind(this);
```

```
    this.handleRadio = this.handleRadio.bind(this);
```

```
    this.handleEmail = this.handleEmail.bind(this);
```

```
    this.handlePass = this.handlePass.bind(this);
```

```
    this.handleRepass = this.handleRepass.bind(this);
```

```
  }
```

```
////////////////////////////////////
```

```
handleRadio(event) {  
  this.setState({ radio: event.target.value });  
}
```

```
handleEmail(event) {  
  this.setState({ login: event.target.value });  
}
```

```
handlePass(event) {  
  this.setState({ password: event.target.value });  
}
```

```
handleRepass(event) {  
  this.setState({ repass: event.target.value });  
}
```

```
handleSubmit(event) {  
  if (this.state.radio === "signin") {  
    request.sign_in(this.state.login, this.state.password);  
  } else if (this.state.radio === "signup") {  
    request.sign_up(this.state.login, this.state.password, this.state.repass);  
  } else if (this.state.radio === "reset") {  
    console.log("Я живой!!");  
    request.reset(this.state.login);  
  }  
}
```

```
event.preventDefault();  
}
```

```
render() {  
  document.body.style.backgroundImage = "url(" + wallpaper + ")";  
  
  return (  

```

```

<div className='page'>
  <div className='container custom_container'>
    <div className='post'>
      <form className='identification' onSubmit={this.handleSubmit}>
        <input id='signin' name='action' type='radio' value='signin'
onChange={this.handleRadio} defaultChecked={true} />
        <label htmlFor='signin'>Увійти</label>
        <input id='signup' name='action' type='radio' value='signup'
onChange={this.handleRadio} />
        <label htmlFor='signup'>Зареєструватися</label>
        <input id='reset' name='action' type='radio' value='reset'
onChange={this.handleRadio} />
        <label htmlFor='reset'>Скинути пароль</label>
        <div id='wrapper'>
          <div id='arrow'></div>
          <input id='email' placeholder='Email' type='text' value={this.state.email}
onChange={this.handleEmail} />
          <input id='pass' placeholder='Password' type='password' value={this.state.pass}
onChange={this.handlePass} />
          <input id='repass' placeholder='Repeat password' type='password'
value={this.state.repass} onChange={this.handleRepass} />
        </div>
        <button type='submit'>
          <span>
            Скинути пароль
          <br />
            Увійти
          <br />
            Зареєструватися
          </span>
        </button>
      </form>
    </div>
  </div>
</div>

```

```
);  
}  
}
```

```
export default Identification;
```

```
.....
```

### **request.js**

```
const request = {
```

```
  sign_in: function (login, password) {
```

```
    const user = { login: login, password: password };
```

```
    fetch("http://localhost:3030/sign_in", { method: "POST", headers: { "Content-Type":  
"application/json" }, body: JSON.stringify(user) })
```

```
      .then((res) => res.json())
```

```
      .then((data) => {
```

```
        sessionStorage.setItem("user", JSON.stringify(data.name[0]));
```

```
        window.location = "/";
```

```
      });
```

```
},
```

```
  sign_up: function (login, password, repass) {
```

```
    const user = { login: login, password: password };
```

```
    fetch("http://localhost:3030/sign_up", { method: "POST", headers: { "Content-Type":  
"application/json" }, body: JSON.stringify(user) })
```

```
      .then((res) => res.json())
```

```
      .then((data) => {
```

```
        console.log(data);
```

```
        window.location.reload();
```

```
      });
```

```
},
```

```
  reset: function (login) {
```

```
    console.log("reset");
```

```
    const user = { login: login };
```

```
    fetch("http://localhost:3030/reset", { method: "POST", headers: { "Content-Type":
```

```
"application/json" }, body: JSON.stringify(user) })
    .then((res) => res.json())
    .then((data) => {
      console.log(data);
      window.location.reload();
    });
  },
};
export default request;
```

.....

app.js

```
import { BrowserRouter as Router, Switch, Route } from "react-router-dom";
import Header from "./main/header";
```

*/\* Страницы \*/*

```
import identification from "./component/identification/identification";
import test_hall from "./component/test hall/test_hall";
import gallery from "./component/gallery/gallery";
import home from "./component/home/home";
```

*/\* Шрифты \*/*

```
import "./main/styles/fonts.css";
```

*/\* Стили сайта \*/*

```
//import './main/styles/main.css';
```

```
function app() {
```

```
  return (
```

```
    <>
```

```
    <Header username='FFF' />
```

```
    <Router>
```

```
      <Switch>
```

```
        <Route exact path='/identification' component={identification} />
```

```
    <Route exact path='/test_hall' component={test_hall} />
    <Route exact path='/gallery' component={gallery} />
    <Route exact path="/" component={home} />
  </Switch>
</Router>
</>
);
}

export default app;
```

## ЛІСТІНГ ПРОГРАМИ - server

**index.js**

```
const database = require("./database");
const express = require("express");
const cors = require("cors");
const app = express();
app.use(cors());

/***** Запуск сервера *****/
app.listen(3030, () => {
  console.log("Я живой!!");
});

/***** Работа с таблицей `users` *****/
app.post("/sign_in", express.json(), (request, results) => {
  database.sign_in(request.body, (error, r) => {
    if (error) {
      results.send({ error: error });
      return;
    }

    results.send({ name: r });
  });
});

app.post("/sign_up", express.json(), (request, results) => {
  database.sign_up(request.body, (error, r) => {
    if (error) {
      results.send({ error: error });
      return;
    }
  });
});
```

```
    results.send({ name: r });
  });
});
app.post("/reset", express.json(), (request, results) => {
  database.reset(request.body, (error, r) => {
    if (error) {
      results.send({ error: error });
      return;
    }

    results.send({ name: r });
  });
});

app.get("/test_hall", (request, results) => {
  database.test_hall((error, r) => {
    if (error) {
      results.send({ error: error });
      return;
    }

    results.send({ name: r });
  });
});

app.post("/answers_test_hall", express.json(), (request, results) => {
  database.answers_test_hall(request.body, (error, r) => {
    if (error) {
      results.send({ error: error });
      return;
    }

    results.send({ name: r });
  });
});
```



```

app.get("/list_users", (request, results) => {
  database.list_users((error, r) => {
    if (error) {
      results.send({ error: error });
      return;
    }

    results.send({ name: r });
  });
});

/***** Работа с таблицей `games` *****/
app.get("/list_games", (request, results) => {
  database.list_games((error, r) => {
    if (error) {
      results.send({ error: error });
      return;
    }

    results.send({ name: r });
  });
});

app.post("/games", express.json(), (request, results) => {
  database.games(request.body, (error, r) => {
    if (error) {
      results.send({ error: error });
      return;
    }

    results.send({ name: r });
  });
});

/***** Работа с таблицей `message` *****/

```

```
app.get("/list_message", (request, results) => {
  database.list_message((error, r) => {
    if (error) {
      results.send({ error: error });
      return;
    }

    results.send({ name: r });
  });
});

app.post("/add_message", express.json(), (request, results) => {
  console.log("/add_message");
  database.add_message(request.body, (error, r) => {
    if (error) {
      results.send({ error: error });
      return;
    }

    results.send({ name: r });
  });
});

app.post("/games/level", express.json(), (request, results) => {
  database.level(request.body, (error, r) => {
    if (error) {
      results.send({ error: error });
      return;
    }

    results.send({ name: r });
  });
});
```

.....

```

database.js
const mysql = require("mysql");

module.exports = {
  /******* Работа с таблицей `users` *****/
  sign_in: (user, callback) => {
    var database = new Database(`SELECT * FROM users where login='${user.login}' and
password='${user.password}'`);
    database.database_access(callback);
  },
  sign_up: (user, callback) => {
    var database = new Database(`INSERT INTO users(login, password)
VALUES('${user.login}', '${user.password}')`);
    database.database_access(callback);
  },
  reset: (user, callback) => {
    var database = new Database(`UPDATE users SET password='123456' WHERE
login='${user.login}'`);
    database.database_access(callback);
  },

  /******* Работа с таблицей `test_hall` *****/
  test_hall: (callback) => {
    var database = new Database("SELECT * FROM `test_hall`");
    database.database_access(callback);
  },
  answers_test_hall: (answers, callback) => {
    //var database = new Database(`SELECT * FROM users where login='${user.login}' and
password='${user.password}'`);
    //database.database_access(callback);
    const answer = answers.answers;
    let array = [];

    for (let number in answer) {
      if (answer.hasOwnProperty(number)) {

```

```
    let question = answer[number];
    array.push(question["question" + number]);
  }
}
},
```

```
class Database {
  constructor(request) {
    this.request = request;
  }
}
```

```
/* Обращение к базе данных */
```

```
database_access(callback) {
  var connection = mysql.createConnection({ host: "localhost", user: "root", password: "",
database: "database" });
```

```
  connection.connect((error) => {
    if (error) {
      callback(error);
    }
  })
```

```
  connection.query(this.request, function (error, results) {
    if (error) {
      callback(error);
    } else {
      callback(null, results);
    }
  });
```

```
  connection.end();
});
}
```

## ПЕРЕЛІК ФАЙЛІВ НА ОПТИЧНОМУ НОСІЇ

Ім'я файлу	Опис
Пояснювальні документи	
Диплом_Горячева.doc	Пояснювальна записка до проекту. Документ Word.
Диплом_Горячева.pdf	Пояснювальна записка до проекту в форматі PDF.
Програма	
Program.rar	Архів, що містить коди програми для запуску проекту.
Презентація	
Презентація_Горячева.ppt	Презентація для проекту.

# ВІДГУК КЕРІВНИКА

## НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ «ДНІПРОВСЬКА ПОЛІТЕХНІКА»

Факультет інформаційних технологій  
Кафедра програмного забезпечення комп'ютерних систем

### ВІДГУК

Наукового керівника Мещерякова Леоніда Івановича, д.т.н., проф. каф. ПЗКС  
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання, посада, місце роботи)

### на магістерську роботу

студент Горячева Антона Владиславовича

а

\_\_\_\_\_ (прізвище, ім'я, по батькові)

курсу II

122М-22-2

групи

спеціальност

122 Комп'ютерні науки

і

освітньої програми

«122 Комп'ютерні науки»

на тему Розробка та обґрунтування веб-додатку з дослідженням

рівня «емоційного інтелекту» користувачів

**Актуальність теми**

**Розглянута магістерська кваліфікаційна робота**

**присвячена знаходженню та реалізації нового розширеного методу проведення**

мануального тестування. Даний підхід є перспективним у тестуванні в корпоративних системах завдяки запровадженню належного рівня безпеки інформації. На сьогоднішній момент сфера розробки, а разом із нею тестування, розвивається швидкими темпами, таким чином магістерська робота відзначається актуальністю.

Мета досліджень Полягає в підвищенні автоматизованості, ефективності, швидкості та доступної для користувача варіативності сценаріїв для повного тестування інформаційних систем, за допомогою використання мови Java та можливостей фреймворку Spring.

Коротка характеристика розділів роботи Перший розділ роботи складається з аналітичного розбору існуючих методик тестування та створення програмного забезпечення за темою магістерської роботи. Другий розділ містить огляд шляхів виконання наймасштабніших завдань забезпечення реалізації створюваного застосунку. Третій розділ присвячено тонкощам програмної реалізації застосунку для управління ресурсами тестування.

Практичне значення роботи Отримані результати роботи є актуальними у проектуванні додатків з управління ресурсами тестування в корпоративних системах. Також надані результати дослідження є підставою для більш поглибленого вивчення проблем, пов'язаних з використанням ресурсів у мануальному тестуванні.

Зауваження та В роботі недостатньо повно оглянуто реалізацію недоліки програмного інтерфейсу користувача, а також процес реєстрації в системі може бути поліпшено відправкою листа.

Висновки та оцінка Магістром було проведено порівняльний аналіз існуючих засобів для зміни стану ресурсів тестування, виявлено недоліки та реалізовано програмний апарат для вирішення досліджених проблем. Під час виконання магістерської кваліфікаційної роботи студентка Стешенко А.А. постала кваліфікованим та впевненим спеціалістом, який знаходить оптимальні рішення у складних технічних питаннях. Вважаю, що магістерська кваліфікаційна робота

заслуговує оцінку «відмінно», а Стешенко А. А. – присвоєння кваліфікації «магістра» по спеціальності комп'ютерних наук.

Науковий  
керівник

Мещеряков Л.І., док. техн. наук, проф., проф. каф. ПЗКС

---

(прізвище, ім'я, по батькові, посада, місце роботи)

«\_\_» \_\_\_\_\_ 20\_\_ р.

---