

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет комп'ютерних наук
(повна назва)

Кафедра програмної інженерії
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА Пояснювальна записка

рівень вищої освіти перший (бакалаврський)

Програмна система для проведення соціальних опитувань. Front-end
(тема)

Виконав:
студент 4 курсу, групи ПЗПІ-20-6

Печій Р.Р.
(прізвище, ініціали)

Спеціальність 121 – Інженерія програмного
забезпечення
(код і повна назва спеціальності)

Тип програми освітньо-професійна

Освітня програма Програмна інженерія
(повна назва освітньої програми)

Керівник доц. кафедри ПІ Валенда Н.А.
(посада, прізвище, ініціали)

Допускається до захисту
Зав. кафедри

(підпис)

З.В.Дудар
(прізвище, ініціали)

2024 р.

Харківський національний університет радіоелектроніки

Факультет _____ комп'ютерних наук
Кафедра _____ програмної інженерії
Рівень вищої освіти _____ перший (бакалаврський)
Спеціальність _____ 121 – Інженерія програмного забезпечення
Тип програми _____ Освітньо-професійна
Освітня програма _____ Програмна Інженерія
(шифр і назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)

«___» _____ 2024 р.

**ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

студентові _____ Печію Ростиславу Романовичу
(прізвище, ім'я, по батькові)

1. Тема роботи _____ Програмна система для проведення соціальних опитувань.
Front-end

Затверджена наказом по університету від _____ 20.05. 2024р. № 471 Ст

2. Термін подання студентом роботи до екзаменаційної комісії _____ 15.06.2024

3. Вихідні дані до роботи Розробити програмну систему для проведення соціальних опитувань, а саме такий елемент системи: Front-end частина.

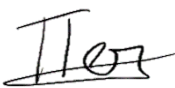
4. Перелік питань, що потрібно опрацювати в роботі

Вступ, аналіз предметної галузі, формування вимог до програмної системи, архітектура та проектування програмного забезпечення, опис прийнятих програмних рішень, тестування розробленого програмного забезпечення, висновки, додатки.

КАЛЕНДАРНИЙ ПЛАН

| № | Назва етапів роботи | Термін виконання етапів роботи | Примітка |
|----|------------------------------------|--------------------------------|-----------------|
| 1 | Аналіз предметної галузі | 09.04.2024 | <i>виконано</i> |
| 2 | Створення специфікації ПЗ | 16.04.2024 | <i>виконано</i> |
| 3 | Проектування ПЗ | 24.04.2024 | <i>виконано</i> |
| 4 | Розробка ПЗ | 23.05.2024 | <i>виконано</i> |
| 5 | Тестування ПЗ | 01.06.2024 | <i>виконано</i> |
| 6 | Оформлення пояснювальної записки | 06.06.2024 | <i>виконано</i> |
| 7 | Підготовка презентації та доповіді | 08.06.2024 | <i>виконано</i> |
| 8 | Попередній захист | 13.06.2024 | <i>виконано</i> |
| 9 | Нормоконтроль, рецензування | 13.06.2024 | <i>виконано</i> |
| 10 | Здача роботи у електронний архів | 15.06.2024 | <i>виконано</i> |
| 11 | Допуск до захисту у зав. кафедри | 15.06.2024 | <i>виконано</i> |

Дата видачі завдання 08 травня 2024р.

Студент  Печій Р. Р.
(підпис)

Керівник роботи _____ доц. кафедри ІІІ Валенда Н.А.
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка до кваліфікаційної роботи бакалавра, 87 стор., 37 рис., 7 табл., 10 джерел, 3 додатки.

ДОСЛІДЖЕННЯ, ОПИТУВАННЯ, ТАРГЕТУВАННЯ, REACTJS.

Метою роботи є розробка клієнтської частини програмної системи для проведення соціальних опитувань, що дозволить користувачам ефективно створювати та публікувати свої опитування, змінювати їх таргетування, зовнішній вигляд і переглядати статистику.

Технології розробки клієнтської частини базуються на використанні мови програмування TypeScript, використовуючи фреймворк ReactJS. Для створення компонентів була використана бібліотека MaterialUI.

У результаті роботи було спроектовано програмну архітектуру клієнтської частини системи «PureSurvey», а також визначено функціональні вимоги до системи.

RESEARCH, SURVEYS, TARGETING, REACTJS.

The aim of the work is to develop the client side of a software system for conducting social surveys, which will allow users to effectively create and publish their surveys, change their targeting, appearance, and view statistics.

The client-side development technologies are based on the TypeScript programming language using the ReactJS framework. The MaterialUI library was used to create components.

As a result of the work, the software architecture of the client part of the «PureSurvey» system was designed, and the functional requirements for the system were determined.

Я, Печій Ростислав Романович, студент гр. ПЗПІ-20-6, здобувач вищої освіти на першому (бакалаврському) рівні кафедри «Програмна інженерія», заявляю: моя кваліфікаційна робота на тему «Програмна система для проведення соціальних опитувань. Front-end», що буде представлена до екзаменаційної комісії для публічного захисту, виконана самостійно, в ній не містяться елементи плагіату і вона може бути опублікована в електронному архіві відкритого доступу EIAr KhNURE. Усі запозичення з друкованих та електронних джерел мають відповідні посилання.

Я ознайомлений із діючим положенням «Про протидію академічному плагіату в ХНУРЕ», згідно з яким виявлення плагіату є підставою для відмови до допуску кваліфікаційної роботи до захисту та застосування дисциплінарних заходів.

ЗМІСТ

| | |
|--|----|
| Вступ..... | 7 |
| 1 Аналіз предметної галузі..... | 8 |
| 1.1 Аналіз предметної галузі..... | 8 |
| 1.2 Виявлення та вирішення проблем..... | 11 |
| 1.3 Постановка задачі..... | 14 |
| 2 Формування вимог до програмної системи..... | 16 |
| 2.1 Постановка мети..... | 16 |
| 2.2 Загальні обмеження..... | 16 |
| 2.3 Припущення та залежності..... | 18 |
| 2.4 Функціональні вимоги..... | 19 |
| 3 Архітектура та проектування програмного забезпечення..... | 21 |
| 3.1 UML проектування ПЗ..... | 21 |
| 3.2 Проектування архітектури ПЗ..... | 24 |
| 3.3 Проектування UI..... | 25 |
| 4 Опис прийнятих програмних рішень..... | 34 |
| 4.1 Архітектура проекту..... | 34 |
| 4.2 Розробка функцій обробки даних..... | 36 |
| 4.3 Розробка основних компонентів..... | 40 |
| 5 Тестування розробленого програмного забезпечення..... | 51 |
| 5.1 Розробка тестових випадків..... | 51 |
| Висновки..... | 57 |
| Перелік джерел посилання..... | 58 |
| Додаток А. Звіт результатів перевірки на унікальність тексту в базі ХНУРЕ..... | 60 |
| Додаток Б. Слайди презентації..... | 61 |
| Додаток В. Специфікація програмного продукту..... | 68 |

ВСТУП

В епоху цифрових трансформацій і стрімкого розвитку онлайн-технологій, соціологічні та маркетингові дослідження все частіше проводяться за допомогою спеціалізованих програмних систем. Традиційні методи збору даних, такі як особисті інтерв'ю чи поштові розсилки, поступово відходять на другий план, поступаючись місцем більш зручним та ефективним онлайн-опитуванням[1].

Програмні системи для проведення опитувань дозволяють швидко створювати, розповсюджувати та збирати відповіді респондентів з усього світу. Вони забезпечують високий рівень автоматизації процесів, економлячи час та ресурси дослідників. Крім того, такі системи надають потужні інструменти для аналізу та візуалізації зібраних даних, полегшуючи виявлення трендів та закономірностей.

Метою створення клієнтської частини створюваної програмної системи для проведення опитувань є забезпечення зручного та інтуїтивно зрозумілого інтерфейсу для взаємодії користувачів та респондентів. Це включає простий та ефективний процес створення опитувань з легким налаштуванням типу питань, логіки, зовнішнього вигляду та локалізації, а також надання зручного адаптивного інтерфейсу для проходження опитувань на різних пристроях. Важливим є також візуалізація результатів у зрозумілій формі за допомогою діаграм та фільтрів для полегшення аналізу зібраних даних, забезпечення безпеки та конфіденційності збору відповідей, даних, а також забезпечення сучасного адаптивного дизайну, що відповідає стандартам веб-розробки і гарантує позитивний досвід користувачів.

Однією з головних переваг онлайн-опитувань є можливість спрямувати напрямок на конкретні групи респондентів за демографічними, географічними чи іншими критеріями. Це забезпечує високу релевантність і якість даних для дослідження. Водночас респонденти можуть брати участь в опитуваннях у зручний для них час і в комфортних умовах, що підвищує достовірність їхніх відповідей.

1 АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ

1.1 Аналіз предметної галузі

Соціальні дослідження є важливим інструментом для вивчення людської поведінки, думок і переконань. У глобальному розумінні, соціологічне дослідження охоплює будь-яку діяльність, спрямовану на формування, розвиток, уточнення чи спростування теорій у соціологічній галузі, незалежно від методів збору та аналізу даних. В Україні цей термін часто плутають з опитуванням громадської думки, проте насправді репрезентативне опитування є лише одним з багатьох методів соціального дослідження.

Проведення соціальних опитувань часто використовуються відповідними дослідниками для розширення знань у різних сферах або перевірки гіпотез, що може принести користь суспільству чи компанії. Зібрана від людей інформація допомагає владі та бізнесу ефективніше розробляти та впроваджувати певні зміни, що можуть покращити життя клієнтів чи громадян.[2]

В Україні переважають проведення кількісних досліджень. Кількісні дослідження покладаються на аналіз числових даних, зібраних від репрезентативної вибірки цільової аудиторії. Дані збираються за допомогою опитувань, анкетувань, інтерв'ю або експериментів. Метод кількісного збору даних полягає в отриманні числових показників, які можна легко проаналізувати за допомогою статистичних програм.

Опитування є гнучким інструментом, що дозволяє задати велику кількість питань на одну тему одному респонденту та зручно додавати нові питання під час його створення.

Загалом, проведення соціальних досліджень є складним процесом який складається з багатьох етапів та вимагає ретельного планування, впровадження методологічно обґрунтованих підходів та глибокого аналізу зібраних даних. Незважаючи на певні обмеження окремих методів, таких як опитування, їх комбінація та вдосконалення дозволяють отримувати цінні знання про людську

поведінку, установки та переконання, що мають практичне значення для різних сфер життя суспільства.

Проведення маркетингової кампанії підприємства нерозривно пов'язане з онлайн-опитуваннями. Цей метод дослідження ринку стрімко набирає популярності, і все вказує на те, що ця тенденція збережеться і в майбутньому. Онлайн-опитування пропонують підприємствам неперевершену швидкість і ефективність збору даних, дозволяючи в реальному часі відстежувати зміни в поведінці споживачів та їхніх уподобаннях. Більше не потрібно чекати тижнями чи місяцями на результати, онлайн-платформи надають можливість отримувати інформацію тут і зараз, дозволяючи миттєво реагувати на нові тренди та запити ринку. Крім того, онлайн-опитування забезпечують значну економію коштів у порівнянні з традиційними методами, що робить їх доступними для компаній будь-якого розміру.[3] В епоху глобалізації онлайн-опитування надають можливість з легкістю охоплювати широку аудиторію, незалежно від її географічного розташування, допомагаючи підприємствам краще розуміти потреби міжнародного ринку.

Впровадження цифрових технологій може значно полегшити деякі аспекти проведення досліджень, особливо в контексті обробки та аналізу великих масивів даних. Розробка спеціалізованих програмних рішень, орієнтованих на потреби дослідників, здатна підвищити ефективність усього процесу та забезпечити більш якісні результати.

Одним із основних напрямків для вдосконалення є створення систем для проведення онлайн-опитувань. Такі системи можуть інтегрувати весь необхідний функціонал – від розробки анкет та налаштування логіки опитування до збору даних, їх аналізу та візуалізації результатів. Важливими складовими є зручний інтерфейс для створення та редагування опитувальників, можливості налаштування дизайну та локалізації, функції таргетування для досягнення релевантної аудиторії, а також потужні аналітичні інструменти для обробки та інтерпретації даних.

Розробка подібних програмних рішень, що враховують специфічні потреби та вимоги дослідників, може значно підвищити продуктивність та якість проведення соціальних досліджень, надаючи доступ до сучасних технологій збору, обробки та представлення даних.

Успішне проведення опитування залежить не лише від цікавої теми та релевантних питань, але й від грамотно спроектованого дизайну, адже візуально приваблива та зручна анкета збільшує залученість респондентів, мотивує їх до завершення опитування та підвищує якість отриманих даних. При створенні дизайну опитувань дослідники керуються принципами простоти та лаконічності, використовуючи чіткі формулювання питань, уникаючи професійної термінології, розбиваючи опитування на логічні блоки та забезпечуючи достатньо вільного простору. Візуальна привабливість досягається за допомогою приємної кольорової гам, а також використання чітких шрифтів, які приємно читати респонденту. Адаптивність до різних пристроїв та розмірів екранів є обов'язковою умовою для зручного проходження опитування.

Дослідження показують, що певні аспекти дизайну можуть суттєво впливати на готовність учасників завершити опитування та надати достовірну інформацію.

Наприклад, формат анкети має вплив на якість проходження респондентами: багатосторінкові анкети з розташуванням питань «одне питання на сторінці» без промотування, як правило, демонструють кращі результати, ніж односторінкові. Респонденти почуваються менш перевантаженими, коли бачать невелику кількість питань на одній сторінці, і це збільшує ймовірність, що вони продовжать заповнювати анкету. [4]

Тип питань також має значення. Хоча табличні питання можуть бути ефективними для збору великої кількості даних, їх надмірне використання може відштовхнути респондентів, роблячи анкету громіздкою та нудною. Рекомендується використовувати табличні питання лише тоді, коли це дійсно необхідно.

Індикатор прогресу допомагає респондентам орієнтуватися в анкеті та розуміти, скільки питань ще залишилось. Дослідження показують, що наявність

індикатора прогресу може позитивно впливати на рівень завершення опитування, але важливо обрати правильний тип індикатора. Лінійний індикатор або індикатор типу "спочатку швидкий, потім повільний" вважаються більш ефективними, ніж індикатор типу "спочатку повільний, потім швидкий".

Кольорова схема опитування також має значення. Яскраві та агресивні кольори можуть відволікати та дратувати респондентів, тоді як спокійні та приємні кольори створюють більш комфортну атмосферу і сприяють концентрації.

В цілому, дизайн онлайн-опитувань повинен бути простим, зрозумілим та привабливим. Важливо уникати перевантаження інформацією, використовувати зручні елементи навігації та забезпечити адаптивність до різних пристроїв. Дотримання цих принципів допоможе створити ефективне опитування, яке дозволить отримати максимальну кількість відповідей та зібрати якісні дані.

1.2 Виявлення та вирішення проблем

Проведення соціальних опитувань можна покращити використавши цифрові технології, які допоможуть спростити роботу у впровадженні дослідження в широкі маси людей, спрощуючи збір та обробку інформації.

На що слід звернути увагу, то це складність створення та редагування анкет традиційним способом. Розробка паперових або статичних електронних форм є трудомістким та негнучким процесом, оскільки будь-які зміни у формулюваннях питань, порядку їх розташування чи варіантів відповідей вимагають значних зусиль. Рішенням може стати інтуїтивно-зрозумілий сервіс зі створенням гнучких опитувань, який дозволить швидко створювати, редагувати та впорядковувати питання та в межах одного опитування.

Також необхідно взяти до уваги складність впровадження логіки відображення питань у багатьох опитуваннях, де наступні запитання можуть з'являтися або приховуватися залежно від попередніх відповідей респондента. Реалізація такої логіки вручну є складним і трудомістким завданням. В нашому проекті можлива реалізація даної функціональності у майбутньому. Відповідним рішенням може стати наявність у системі візуального зручного інструменту для

налаштування логіки відображення питань на основі заданих правил та відповідей опитуваних, що дозволить динамічно змінювати шлях проходження анкети для кожного респондента.

Особливу увагу необхідно звернути на проблему низької залученості та високого відсотка недопроходжень опитувань через незручність інтерфейсу. Застарілий або погано адаптований під мобільні пристрої інтерфейс анкети може значно знизити рівень залучення респондентів і призвести до високої кількості незавершених опитувань. Рішенням цієї проблеми може стати можливість створення сучасного, інтуїтивного та адаптивного веб-інтерфейсу опитувань з урахуванням принципів юзабіліті, який би коректно відображався на різних пристроях та екранах, забезпечуючи зручну навігацію та заповнення.

Також треба проаналізувати проблему з переглядом отриманої інформації від респондентів. Застарілі інструменти візуалізації або їх повна відсутність значно ускладнюють сприйняття та аналіз зібраних даних. Спрощений інтерфейс без можливості сортування, фільтрації та гнучкого налаштування відображення результатів може призвести до невірної інтерпретації даних та ускладнити пошук важливих закономірностей. Рішенням цієї проблеми може стати створення інструментів візуалізації, наприклад, діаграми. Важливим аспектом є можливість гнучкого налаштування відображення результатів, фільтрації та сегментації даних за різними параметрами, що допоможе користувачам легко знаходити потрібну інформацію з проведених опитувань

Створення програмної системи для соціальних опитувань, яка врахує описані проблеми та запропонує відповідні рішення, дозволить значно спростити процес роботи як для клієнтів які створюють ці опитування, так і для респондентів, а також забезпечить зручну і інтуїтивно зрозумілу візуалізацію зібраних даних для отримання цінних даних.

Для більш детального аналізу предметної області було вирішено провести аналіз аналогічного сервісу для проведення опитувань, такого як SurveyMonkey.

SurveyMonkey - це популярний онлайн-сервіс для створення та проведення опитувань, який використовують як окремі особи, так і компанії для збору даних,

дослідження ринку, зворотного зв'язку та інших цілей. Платформа дозволяє створювати опитування різної складності, від простих анкет до комплексних досліджень з розгалуженою логікою, пропонуючи широкий вибір типів питань, матриці відповідей, можливість додавати зображення та відео. Сервіс також надає можливості для таргетування аудиторії, розповсюдження опитувань через email, вбудовування в веб-сайти, збору відповідей через мобільні пристрої та володіє потужним інструментом аналізу результатів з діаграмами та звітами даних.

Незважаючи на популярність сервісу, він має деякі недоліки, які роблять його зручність використання більш важким. Одним з головних недоліків сервісу може бути обмежена можливість кастомізації. Платформа дозволяє вибирати з кількох тем, при чому користувачі можуть змінювати лише деякі параметри в стилі опитування. Це позбавляє їх свободи в оформленні, обмежуючи їх можливості в створенні візуальної частини для опитування, яка буде притаманна користувачеві. Приклад створення візуальної частини опитування на платформі SurveyMonkey представлений на рисунку 1.1

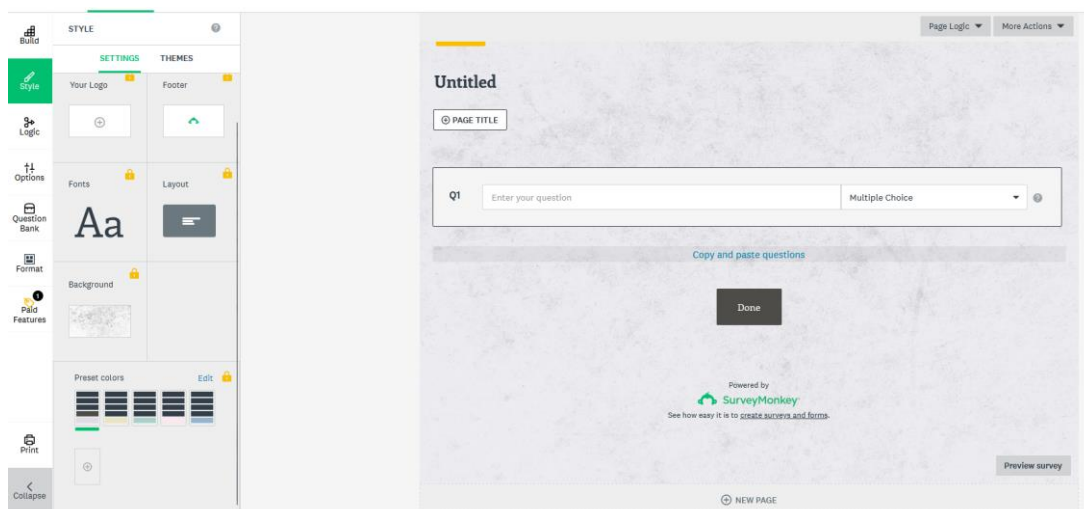


Рисунок 1.1 – Створення стилю опитування в сервісі SurveyMonkey

Наша програмна система планує вирішити цю проблему, надаючи користувачам можливість створювати власні стилі, використовуючи інструменти для налаштування кольорів, параметрів опитування та інших елементів оформлення які містяться у заздалегідь створених шаблонах системи. Це дозволить зробити опитування більш унікальними та привабливими для користувачів.

Аналог не надає достатньо гнучкої логіки для створення опитувань з складними розгалуженнями. Це може обмежити можливості досліджень, де необхідно змінювати порядок питань залежно від відповідей респондента. SurveyMonkey не надає достатньої кількості інструментів для реалізації такої функціональності. Наша команда планує впровадити в майбутніх версіях проекту розширену логіку для створення опитувань з складними розгалуженнями, що дозволить створювати більш інтерактивні опитування, що відповідають потребам наших користувачів.

Було помічено відсутність інтеграції з іншими сервісами. Аналог не дозволяє легко вбудувати опитування в інші сайти. Створювана програмна система планує забезпечити інтеграцію в інші сайти або сервіси, що дозволить користувачам збирати дані з різних джерел та автоматизувати процеси збору та аналізу даних.

Також одним з основних недоліків SurveyMonkey є відсутність створення перекладів для опитувань. Користувачі платформи не можуть створювати опитування з різними перекладами для питань та варіантів відповідей. Наша програмна система планує забезпечити створення нових мов для опитування, щоб дозволити користувачам створювати опитування для різних груп респондент, збільшуючи доступність та ефективність проведення опитування.

Підсумовуючи аналіз аналогу програмної системи для проведення опитувань можна зробити висновок що при дотриманні вимог до інтерфейсу та функціональних можливостей допоможе створити ефективну і зручну програмну систему для проведення опитувань, яка перевершить існуючі рішення на ринку.

1.3 Постановка задачі

Під час виконання роботи має бути спроектована система для проведення соціальних опитувань онлайн. Вона має містити наступні функції:

а) функції для користувачів:

- 1) створення груп опитувань;
- 2) створення опитувань і питань в ньому;
- 3) редагування зовнішнього вигляду опитування;

- 4) отримання інформації про відгуки опитувань;
 - 5) отримання статистики стосовно проходження опитування;
 - 6) створення таргетингів для опитувань.
- б) функції для адміністраторів:
- 1) створення шаблонів опитувань;
 - 2) перегляд інформації користувачів;
 - 3) редагування даних опитувань користувачів.

Система має забезпечувати різні варіанти налаштувань опитування, а саме:

- таргетинг;
- зовнішній вигляд;
- тривалість проведення;

Розроблений інтерфейс має бути інтуїтивно зрозумілим. Для користувача має бути створена можливість отримати підтримку від адміністраторів у вигляді зворотного зв'язку.

1.3.1 Цільова аудиторія

Основну аудиторію створюваної програмної системи складають люди віком від 18 до 50 років, які професійно займаються дослідженнями, аналізом даних, маркетингом, а також ті, хто прагне отримати зворотний зв'язок від користувачів чи клієнтів та оцінити ефективність та/або успішність своєї роботи. Цей сервіс насамперед приверне увагу маркетологів, соціологів, бізнес-аналітиків, спеціалістів з обслуговування клієнтів, викладачів, менеджерів проектів, а також представників некомерційних організацій. Додатково сервіс буде більш зручним для тих, хто вже має досвід роботи з онлайн-опитуваннями та прагне отримати подібний або більший функціонал та зручний інтерфейс для проведення опитувань будь-якої складності.

2 ФОРМУВАННЯ ВИМОГ ДО ПРОГРАМНОЇ СИСТЕМИ

2.1 Постановка мети

Метою виконання роботи є розробка клієнтської частини для програмної системи PureSurvey, яка дозволить користувачам ефективно створювати, редагувати та публікувати опитування, налаштовувати їх зовнішній вигляд та таргетинг, а також отримувати доступ до детальної статистики по проведених опитуваннях. Клієнтська частина має бути інтуїтивно зрозумілою та зручною у використанні, забезпечуючи приємний користувацький досвід, незалежно від рівня технічних навичок.

Основними задачами розробки є створення зручного інтерфейсу для створення, редагування та публікації опитувань з різними типами питань та варіантів відповідей, реалізація гнучкої системи налаштувань, яка надасть користувачам можливість визначати зовнішній вигляд опитувань, вибираючи кольори, шрифти та змінювати інші елементи дизайну, впровадження таргетингу, що забезпечить можливість визначати цільову аудиторію опитувань за місцезнаходженням, а також впровадити візуалізацію статистики, надаючи користувачам інструменти для аналізу результатів опитувань у вигляді діаграм з можливістю фільтрації та розділення даних. Крім того, важливо забезпечити адаптивність сайту, забезпечуючи коректну роботу клієнтської частини в різних браузерах та на різних пристроях, включаючи комп'ютери, планшети та смартфони.

Клієнтська частина має бути інтегрована з серверною частиною за допомогою API, що забезпечить надійну та безпечну передачу даних.

2.2 Загальні обмеження

Клієнтська частина програмної системи повинна відповідати наступним обмеженням:

а) функціональні обмеження:

- 1) залежність від серверної частини. Клієнтська частина повністю залежить від серверної частини (backend) для виконання основних

функцій, таких як аутентифікація, авторизація, зберігання даних, обробка опитувань, та ін. Клієнтська частина не може функціонувати без стабільного з'єднання з сервером та коректного функціонування API;

- 2) обмеження браузерів. Функціональність клієнтської частини може бути обмежена можливостями веб-браузера користувача. Наприклад, деякі старі браузери можуть не підтримувати всі необхідні технології (HTML5, CSS3, JavaScript), що може призвести до некоректної роботи або обмеженої функціональності програмної системи;
 - 3) відсутність офлайн режиму. Клієнтська частина не може працювати в офлайн режимі, оскільки потребує постійного з'єднання з сервером для отримання даних та виконання операцій.
- б) технічні обмеження:
- 1) обмеження продуктивності. Продуктивність клієнтської частини залежить від продуктивності пристрою користувача (комп'ютер, планшет, смартфон) та швидкості інтернет-з'єднання. На пристроях з низькою продуктивністю або при повільному інтернет-з'єднанні застосунок може працювати повільно або некоректно;
 - 2) обмеження розміру екрану: На пристроях з різними розмірами екрану інтерфейс клієнтської частини може відображатися не оптимально, що може спричинити проблему використання певних функцій застосунку.
- а) обмеження безпеки:
- 1) залежність від безпеки серверної частини. Безпека клієнтської частини залежить від безпеки серверної частини. Якщо серверна частина буде зламана, це може призвести до витоку даних користувачів або інших серйозних проблем.

Вказані обмеження допоможуть забезпечити оптимальну продуктивність розробки та відтворення безпеки програмної системи, а також передбачити можливі проблеми, пов'язані з різними інструментами користувачів.

2.3 Припущення та залежності

Розробка клієнтської частини (frontend) для створюваної програмної системи може спиратися на певні припущення та залежності, які необхідно враховувати під час проектування проекту:

а) припущення:

- 1) наявність стабільного API. Припускається, що серверна частина надає стабільний, добре документований API (Application Programming Interface) для взаємодії з клієнтською частиною. Для створюваної системи описуване API має бути чітке та структуроване для подальшої зручної розробки системи а також забезпечувати надійну аутентифікацію під час виконання запитів;
- 2) сучасні обладнання користувачів. припускається, що користувачі використовують сучасні пристрої, які підтримують усі потрібні технології для відтворення клієнтської частини створюваної програмної системи. Це дозволить використовувати сучасні технології веб-розробки для створення інтерактивного та привабливого інтерфейсу;
- 3) доступність бібліотек. Припускається, що використовувані бібліотеки та фреймворки (ReactJS, MaterialUI) будуть доступні та підтримуватимуться протягом всього життєвого циклу проекту. Оскільки для забезпечення стабільності та безпеки проекту потрібна постійна підтримка продукту використовуючи одні й ті ж інструменти розробки, а також для можливості подальшого розширення функціональності продукту;
- 4) користувачі знайомі з системами для створення онлайн-опитувань. Припускається, що більшість користувачів вже мають досвід створення або проходження онлайн-опитувань та знайомі з основними концепціями та термінологією, оскільки більша частина

сайту розрахована на досвічених користувачів з метою зменшення відображення великої кількості інформації на екрані.

б) залежності:

- 1) серверна частина. Клієнтська частина повністю залежить від серверної частини, яка відповідає за зберігання даних опитувань, користувачів, результатів опитувань та іншої інформації. Серверна частина також виконує обробку запитів від клієнтської частини, аналізує дані та надає необхідну інформацію для відображення в інтерфейсі;
- 2) інтернет-з'єднання. Клієнтська частина потребує стабільного інтернет-з'єднання для взаємодії з сервером та завантаження даних;
- 3) безпека. Клієнтська частина повинна бути захищена від різних видів атак на систему, оскільки це може призвести до витоку даних користувачів або зламу системи, яка приведе до непрацезданості системи;
- 4) оновлення браузерів. Подальша розробка клієнтської частини має передбачати можливі оновлення браузерів користувачів та зміни в стандартах веб-розробки.

Для створення хорошої системи необхідно врахувати надані припущення та залежності для того щоб забезпечити надійну взаємодію з серверною частиною та передбачити можливі проблеми, пов'язані з використанням зовнішніх сервісів та бібліотек.

2.4 Функціональні вимоги

Під час виконання кваліфікаційної роботи треба спроектувати систему для проведення опитувань. Дана система має надавати користувачам наступні функції:

- створення груп опитувань користувачами;
- створення опитувань і питань в них користувачами;
- редагування зовнішнього вигляду опитування користувачами;
- отримання інформації про відгуки опитувань користувачами;

- отримання статистики стосовно проходження опитувань користувачами;
- створення таргетингів для опитувань користувачами;
- створення шаблонів опитувань адміністраторами;
- перегляд інформації користувачів адміністраторами;
- редагування даних опитувань користувачів адміністраторами;
- налаштування таргетингу опитувань;
- налаштування зовнішнього вигляду опитувань;
- налаштування тривалості проведення опитувань.

3 АРХІТЕКТУРА ТА ПРОЄКТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

3.1 UML проектування ПЗ

Система має дві ролі користувачів – звичайний користувач, що створює опитування та адміністратор. Для відображення функціональних можливостей кожної ролі було розроблено та спроектовано відповідні Use Case діаграми. На рисунку 3.1 зображено діаграму Use Case для користувача сервісу.

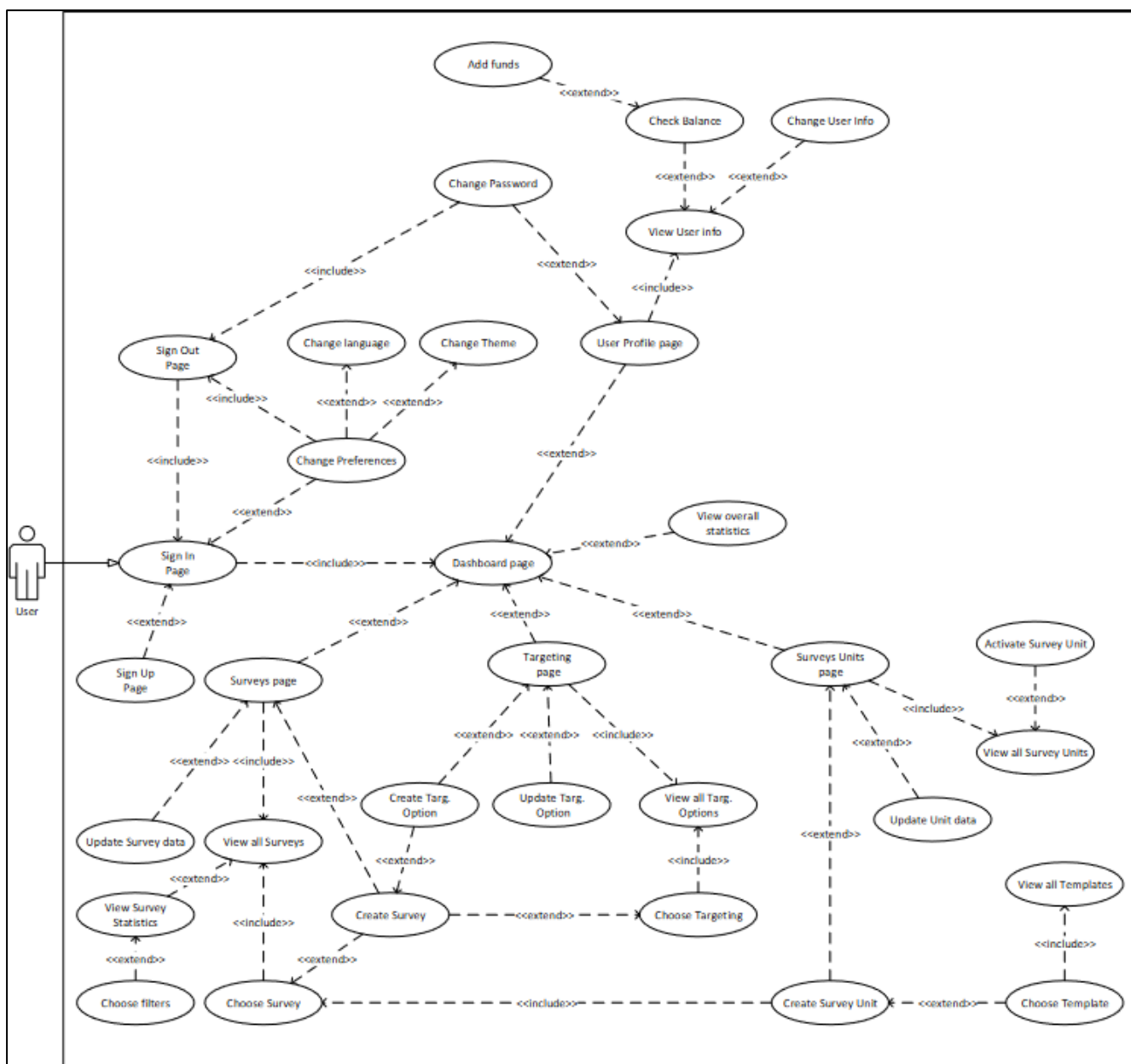


Рисунок 3.1 – Use Case діаграма для ролі користувача сервісу

Відповідно до спроектованої діаграми, користувач має наступні можливості:

- реєстрація аккаунту та вхід в обліковий запис з використанням логіну та паролю;
- зміна мови та стилю інтерфейсу;
- перегляд опитувань з можливістю редагування та створювання нових опитувань і питань в ньому;
- перегляд статистики опитування;
- перегляд групи опитувань з можливістю редагування та створювання нових груп опитувань;
- редагування зовнішнього вигляду певної групи опитувань;
- можливість вбудовування опитування у вигляді блоку на сайті;
- зміна таргетингу опитування та створення нових параметрів таргетингу;
- перегляд загальної статистики;
- керування аккаунтом – зміна паролю облікового запису, подовження підписки на повноцінне користування сервісом;

На рисунку 3.2 зображено Use Case діаграму для ролі адміністратора.

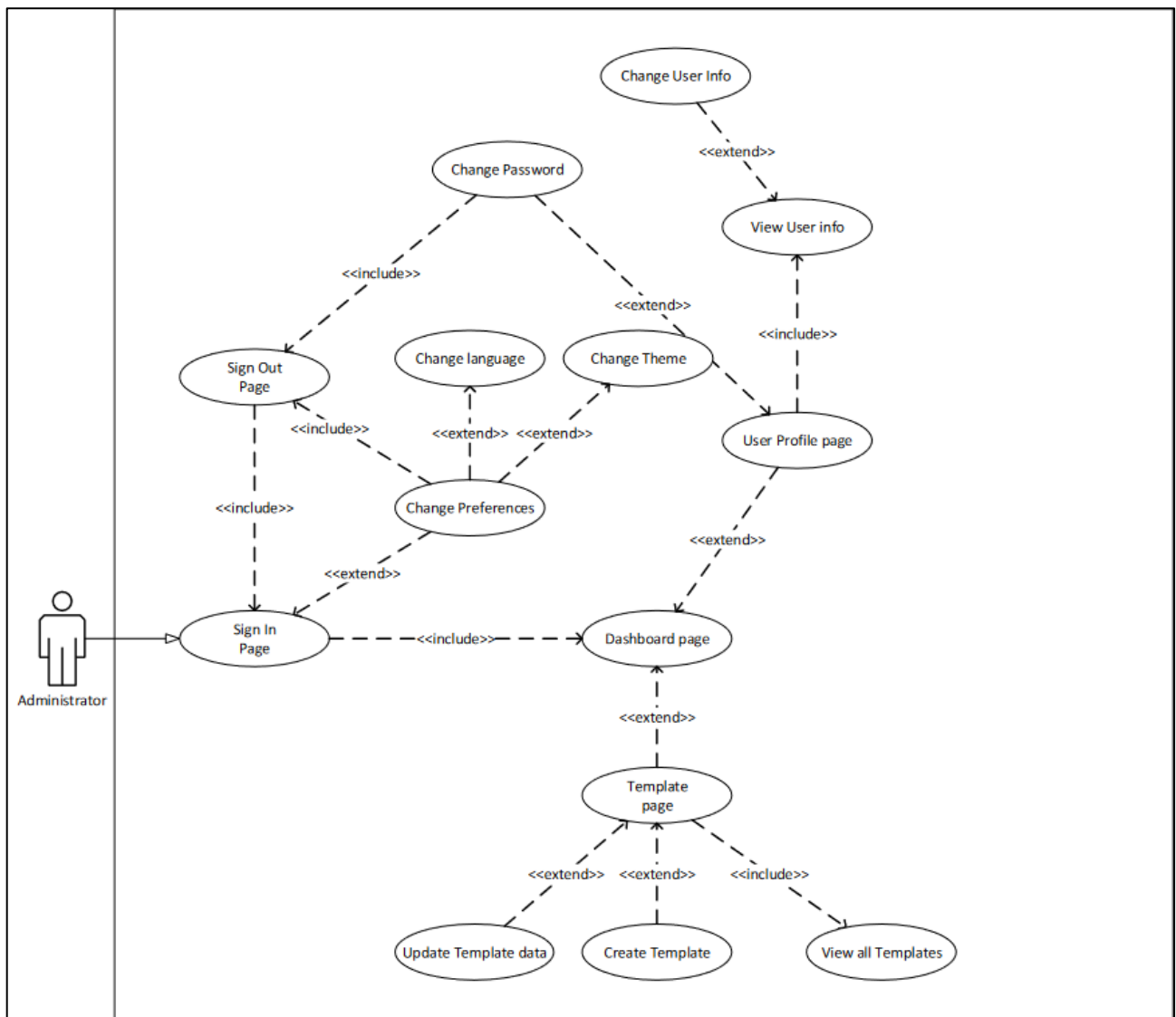


Рисунок 3.2 – Use Case діаграма для ролі адміністратора

Відповідно до спроектованої діаграми, адміністратор має наступні можливості:

- вхід в обліковий запис з використанням логіну та паролю;
- зміна мови та стилю інтерфейсу;
- перегляд наявних у системі шаблонів для груп опитувань, їх редагування та створення нових;
- керування аккаунтом – зміна паролю облікового запису та особистих даних адміністратора;

Також для адміністратора буде додатково створено відповідний блок функціоналу для відповідей на запитання користувачів.

3.2 Проектування архітектури ПЗ

Створюваний сервіс використовуватиме стандартну трирівневу архітектуру з додатковим використанням високонавантаженої частини. [5] На рисунку 3.3 зображено діаграму розгортання програмної системи.

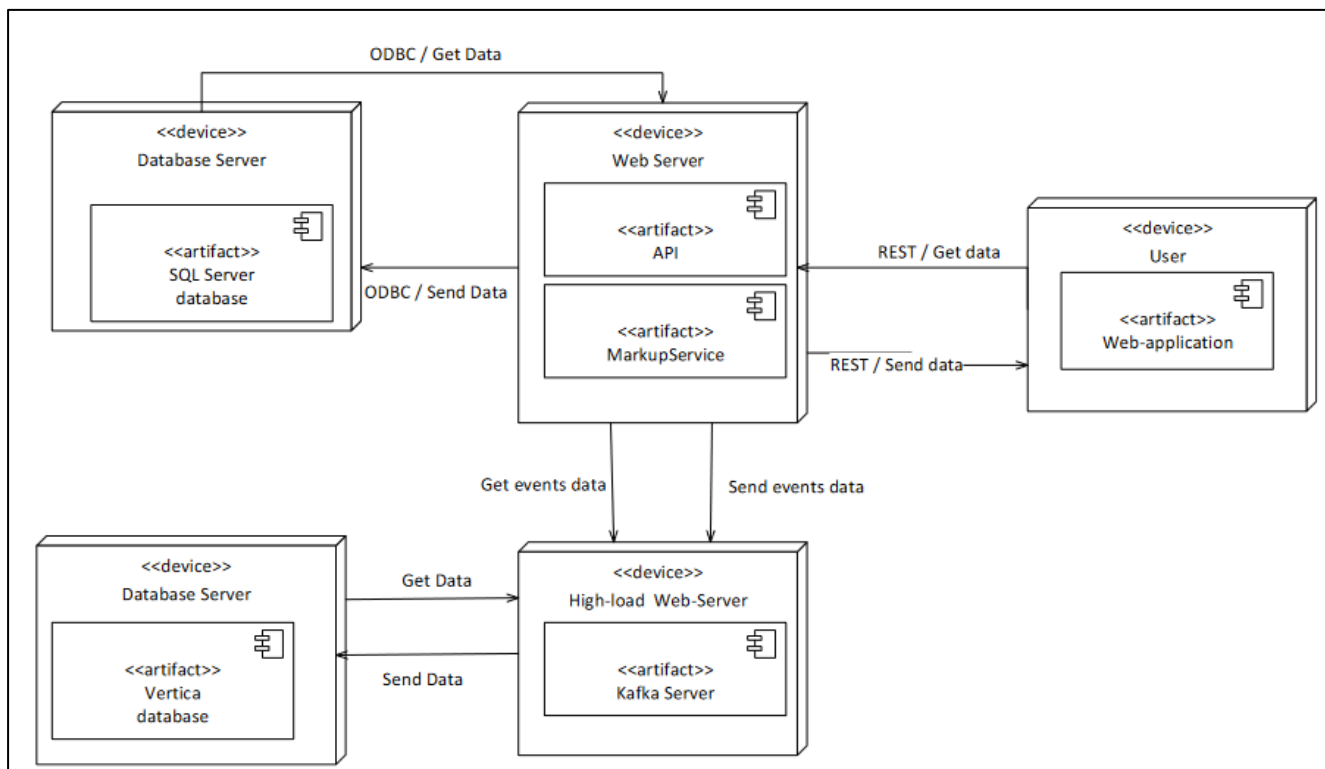


Рисунок 3.3 – Діаграма розгортання

На найнижчому рівні знаходиться основна база даних для зберігання даних користувачів та даних опитувань міститься в СКБД MSSQL. Також додатково додана база даних СКБД Vertica яка відповідає за збереження згенерованих подій під час проходження опитування.

На другому рівні знаходиться два сервери з бізнес-логікою. Основний сервер написаний на .NET відповідає за передачу даних про користувачів та опитувань на клієнтську частину. Додатково було створено високонавантажена частина – Kafka сервер, який відповідає за обробку подій, надісланих клієнтами, які проходять опитування.

На верхньому рівні розроблено клієнтську частину, яка написана на мові TypeScript[6] використовуючи фреймворк ReactJS[7]. Дані технології дають

можливість створювати сторінки, використовуючи компоненти, які можуть бути повторно використані в різних частинах інтерфейсу. А також фреймворк дає можливість використовувати переваги реактивності сторінок, що дає можливість оновлювати дані в реальному часі не перезавантажуючи сторінки.

3.3 Проектування UI

3.3.1 Опис загального стилю та дизайну

Для клієнтської частини обрано сучасний та функціональний стиль дизайну – матеріальний дизайн, який забезпечує зручний та інтуїтивно зрозумілий користувацький інтерфейс.

Material Design (або Матеріальний дизайн) – це концепція дизайну, розроблена компанією Google, яка базується на принципах фізичного світу, таких як світло, тінь, матеріал та рух. Цей стиль характеризується чіткими лініями, виразними кольорами та простою навігацією, що робить його ідеальним для створення зручних та естетично привабливих веб-застосунків. Матеріальний дизайн обраний за його зручність, інтуїтивність та сучасний вигляд. [8]

Для реалізації матеріального дизайну в проекті використовується бібліотека Material UI[9], яка надає широкий набір готових компонентів інтерфейсу, таких як кнопки, поля введення, меню, переходи та інші. Це значно спрощує та прискорює розробку клієнтської частини, забезпечуючи єдиний стиль та високу якість інтерфейсу. Бібліотека Material UI використовується для спрощення та прискорення розробки, а також для забезпечення єдиного стилю інтерфейсу.

Основними кольорами інтерфейсу є синій (#2196f3) та білий, як передбачено стандартною палітрою Material UI. Синій колір асоціюється з надійністю, стабільністю та професіоналізмом, що підкреслює серйозність та значимість досліджень, проведених за допомогою системи. Білий колір забезпечує контраст та чіткість, роблячи інтерфейс легким для сприйняття.

Для відображення тексту в інтерфейсі використовується шрифт Roboto, який також є стандартним для Material UI. Roboto – це сучасний та універсальний шрифт, який добре читається на екранах різних розмірів та роздільної здатності.

Для створення макетів інтерфейсу використовується інструмент Figma. Figma – це платформа для створення дизайнів інтерфейсів, яка надає широкий набір інструментів для створення макетів, прототипів та співпраці в команді. [10]

В цілому, обраний стиль та дизайн клієнтської частини орієнтовані на створення зручного, інтуїтивно зрозумілого та візуально привабливого інтерфейсу, який допоможе користувачам ефективно працювати з системою та отримувати максимальну користь від її функціоналу.

3.3.2 Створення макетів

Для клієнтської частини було створено світлу та темну кольорові схеми з перемикачем теми. Надалі буде показана світла схема сервісу. Користувача зустрічає перша сторінка – авторизації, яка зображена на рисунку 3.3.

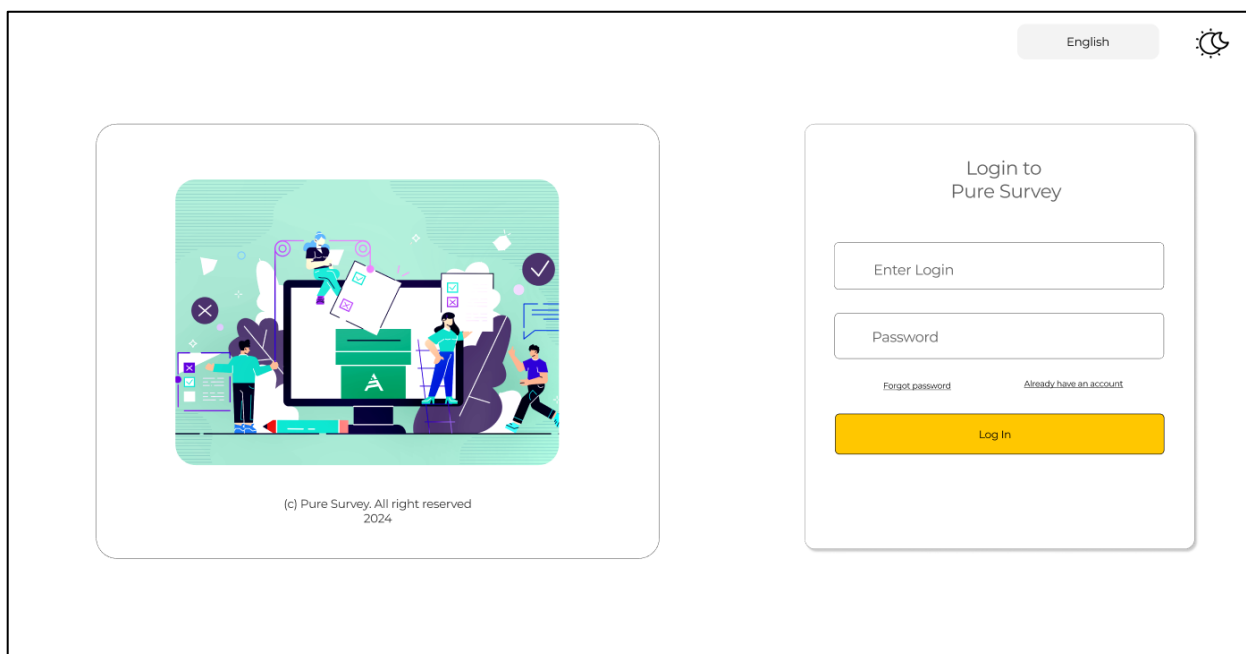


Рисунок 3.4 – Макет сторінки авторизації

Додатково клієнт може обрати можливість створення нового акаунту і користувача буде перенесено на сторінку реєстрації нового облікового запису. Інтерфейс зображено на рисунку 3.5

The mockup shows a registration form with the following elements:

- Language selector: English
- Dark mode toggle: Moon icon
- Form title: Create new account
- Input fields: Login, Password, Last Name, First Name
- Submit button: Create (yellow)
- Link: [Already have an account](#)

Рисунок 3.5 – Макет сторінки реєстрації

Сторінки для перегляду даних (список опитувань, список груп, деталі опитування тощо) можуть бути спроектовані за єдиним подібним макетом представленим на рис. 3.6.

The mockup shows a data table interface with the following elements:

- Header: Typography
- Section title: Lorem Ipsum
- Table structure:

| <input type="checkbox"/> | Lorem Ipsum | Lorem Ipsum | Lorem Ipsum | Lorem Ipsum | |
|--------------------------|-------------|-------------|----------------------------|----------------------------|--------------------------------|
| <input type="checkbox"/> | Lorem Ipsum | 20/06/2024 | 📍 Lorem Ipsum | 📊 Lorem Ipsum | Lorem Ipsum STATS |
| <input type="checkbox"/> | Lorem Ipsum | 20/06/2024 | 📍 Lorem Ipsum | 📊 Lorem Ipsum | Lorem Ipsum STATS |
| <input type="checkbox"/> | Lorem Ipsum | 20/06/2024 | 📍 Lorem Ipsum | 📊 Lorem Ipsum | Lorem Ipsum STATS |
| <input type="checkbox"/> | Lorem Ipsum | 20/06/2024 | 📍 Lorem Ipsum | 📊 Lorem Ipsum | Lorem Ipsum STATS |
| <input type="checkbox"/> | Lorem Ipsum | 20/06/2024 | 📍 Lorem Ipsum | 📊 Lorem Ipsum | Lorem Ipsum STATS |

Footer: Rows per page: 10 | 1-5 of 13 | < >

Рисунок 3.6 – Макет сторінки для перегляду даних

Загалом на сторінці будуть знаходитись наступні компоненти:

- заголовок сторінки: Відображає назву сторінки;

- таблиця даних: Відображає дані у вигляді таблиці з можливістю перегляду додаткової інформації або виконання дій з кожної одиниці даних (редагування, видалення тощо);
- кнопка для додавання: Дозволяє додати новий об'єкт (опитування, групу тощо);
- пагінація: Елементи інтерфейсу для навігації між сторінками даних, якщо їх кількість перевищує певний ліміт або для обмеження кількості виведених результатів на сторінку.

Заголовок веб-додатку буде забезпечувати швидкий доступ до ключових функцій та налаштувань. Він розташований у верхній частині екрану і складається з двох основних частин: навігаційного меню та блоку налаштувань профілю.

Для забезпечення оптимального відображення навігаційного меню на різних пристроях, особливо на мобільних з меншим розміром екрану, буде використано Burger Menu.

Burger Menu – це іконка, що складається з трьох горизонтальних ліній, яка при натисканні відкриває випадаюче меню з пунктами навігації. Такий підхід дозволяє економити простір на екрані та зберігати чіткий та лаконічний дизайн інтерфейсу.

Приклад макету для випадаючого меню представлений на рисунку 3.7

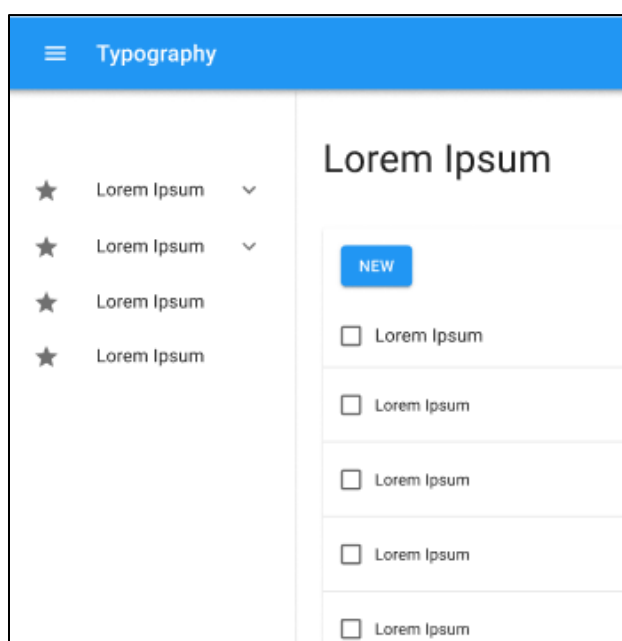


Рисунок 3.7 – Макет навігаційного меню додатку

Випадаюче меню для навігації може містити ті ж пункти, що і стандартне навігаційне меню:

- опитування: Перехід до списку всіх створених опитувань;
- групи: Перехід до списку груп опитувань;
- таргетинги: Перехід до списку створених таргетингів;
- статистика: Перехід до сторінки загальної статистики по опитуванням;

Також на заголовку міститься блок налаштування профілю – також випадаюче меню, яке буде показане при натисненні на значок профілю користувачу. Приклад блоку налаштувань представлений на рисунку 3.8

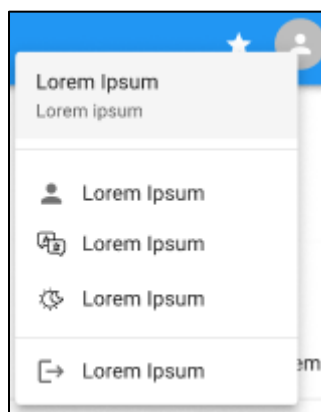


Рисунок 3.8 – Макет для блоку налаштувань додатку

Дане випадаюче меню може містити наступну інформацію:

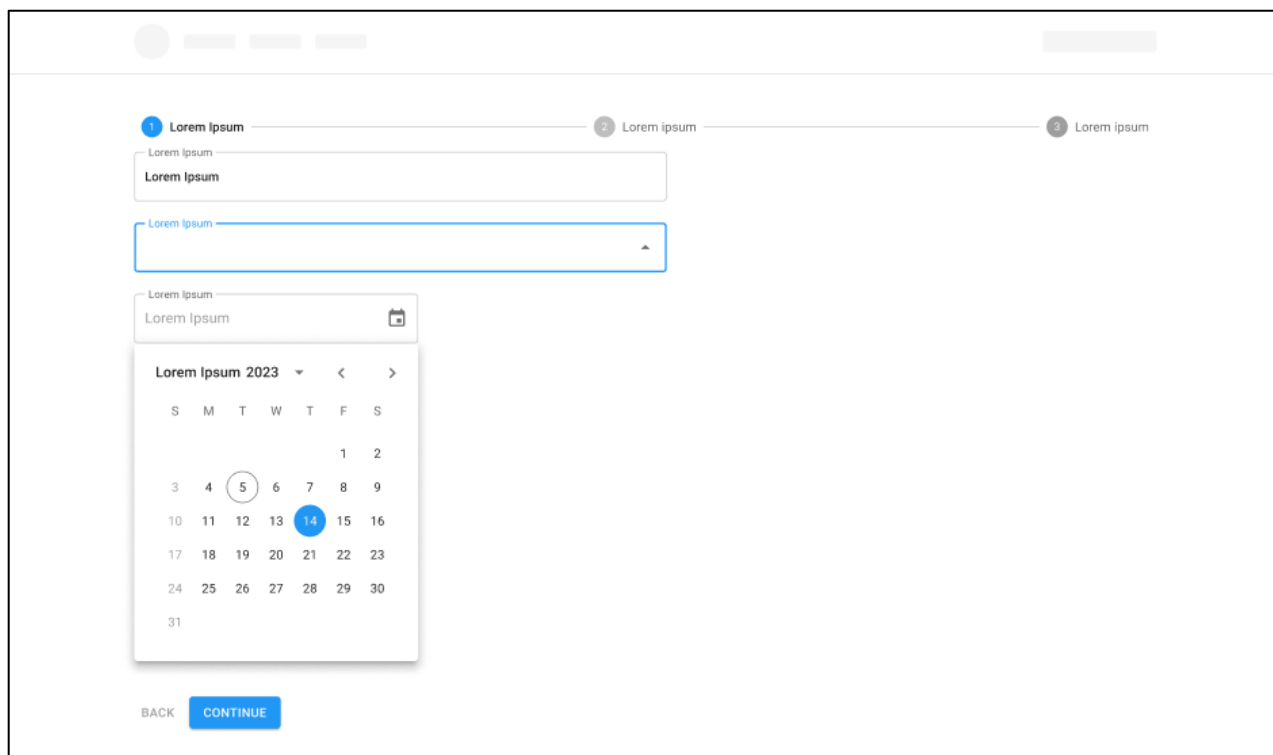
- особиста інформація. В списку може бути наявна коротка інформація про користувача, наприклад ім'я, телефон або пошта. Це потрібно для власної ідентифікації облікового запису, оскільки він може бути використаний багатьма людьми, які можуть мати як персональний, так і бізнес-акаунт компанії;
- різні типи налаштувань. В списку можуть бути присутні функціональні можливості для зміни різних налаштувань користувача, наприклад, зміна мови додатку чи стилю, або вихід з облікового запису.

Далі був спроектований макет для сторінки створення опитування. Сторінка для створення опитування буде спроектована з розбиттям даних на етапи, для внесення інформації поетапно, що дає покращення навігації та відображення

прогресу створення опитування. Ми розділимо процес заповнення даних на три логічних кроки:

- а) основна інформація: буде містити поле для назви опитування та дату закінчення активації опитування;
- б) таргетинг: буде містити поле з вибором наявних для користувача країн;
- в) питання: буде містити форму з додаванням питань, варіантів відповідей і перекладів до них;

Сторінки для заповнення основної інформації та таргетингу можуть бути спроектовані за подібним макетом представленим на рисунку 3.9. Нижче заголовка макету буде відображений прогрес створення опитування, який буде змінюватися відповідно до поточного етапу.



The image shows a wireframe of a three-step survey creation process. At the top, a progress bar indicates the current step (1) is active. Below the progress bar, there are three main sections: 1. A text input field for the survey name. 2. A dropdown menu for selecting the survey end date. 3. A calendar widget for selecting the date. The calendar shows the month of May 2023, with the 5th and 14th highlighted. At the bottom, there are 'BACK' and 'CONTINUE' buttons.

Рисунок 3.9 – Макет для етапів з загальною інформацією

На макеті будуть міститися поля різних типів для введення відповідних даних для опитування. Також на сторінці додані відповідні кнопки для переходів між етапами форми створення опитування. Для форми створення та заповнення було створено макет з приблизним виглядом форми створення опитування, який представлений на рисунку 3.10

Рисунок 3.10 – Макет форми створення опитування

Форма буде мати наступні компоненти:

- текст питання: поле введення тексту для питання;
- тип питання: випадаючий список з типами питань (один варіант, кілька варіантів, текстове поле, шкала оцінювання, тощо);
- для кожного з варіантів відповідей: поля для введення варіантів відповідей з можливістю додавання та видалення;
- переклад питання: вкладки для введення перекладу питання та варіантів відповідей на обрані мови. Це було спроектовано для зменшення полів для введення та кількості інформації на формі;
- відповідні кнопки: на формі присутні потрібні кнопки для додавання нового питання, варіанту відповіді тощо;
- перехід: буде створено перехід між різними питаннями для редагування їх в будь-який момент часу.

Для інших об'єктів буде створена подібний макет сторінки для створення чи редагування інформації. Приклад такого макету представлений на рисунку 3.11.

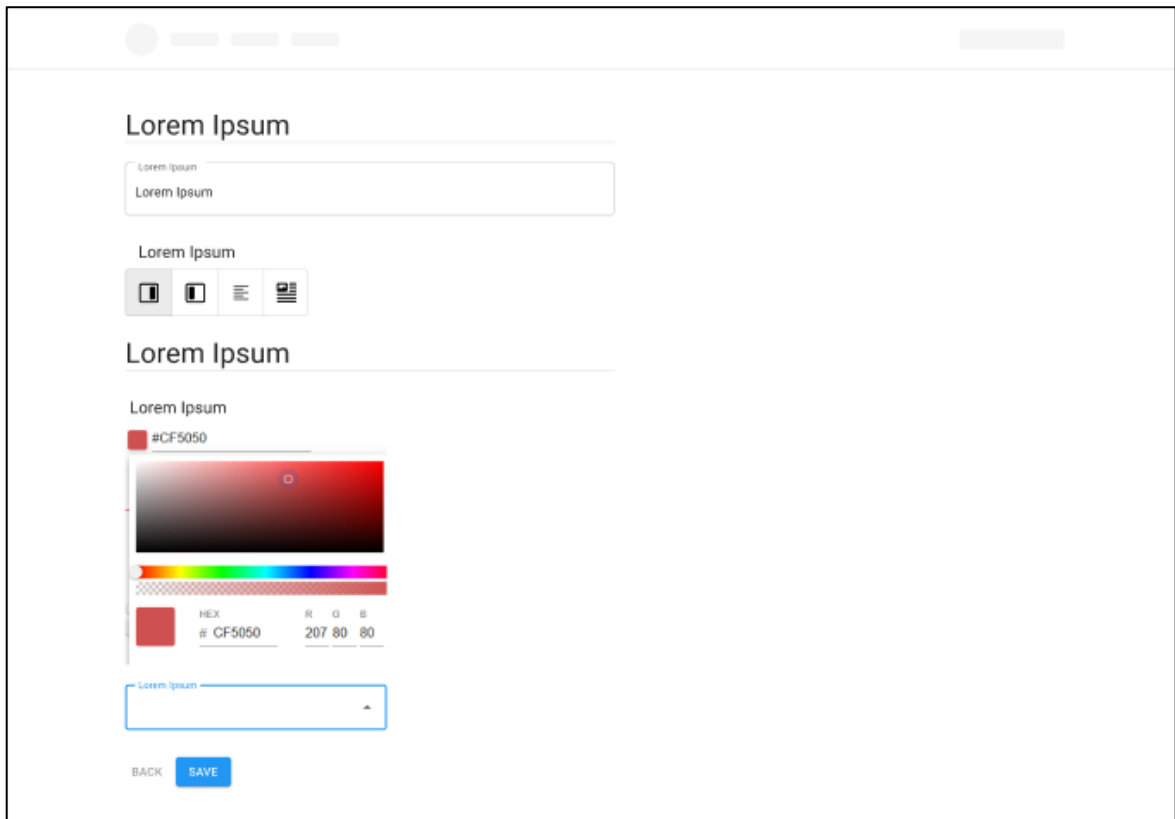


Рисунок 3.11 – Макет форми створення різних типів об’єктів

Сторінка статистики опитування дозволяє користувачам глибоко проаналізувати результати проведеного опитування. У верхній частині сторінки відображається загальна інформація стосовно опитування, на які воно було націлене.

Центральним елементом сторінки є блок візуалізації, де дані представлені у вигляді інтерактивних діаграм. Користувач може обрати конкретне питання з опитування, використовуючи випадаючий список, і діаграми будуть оновлені відповідно до обраного питання, відображаючи розподіл відповідей.

Приклад макету представлений на рисунку 3.12

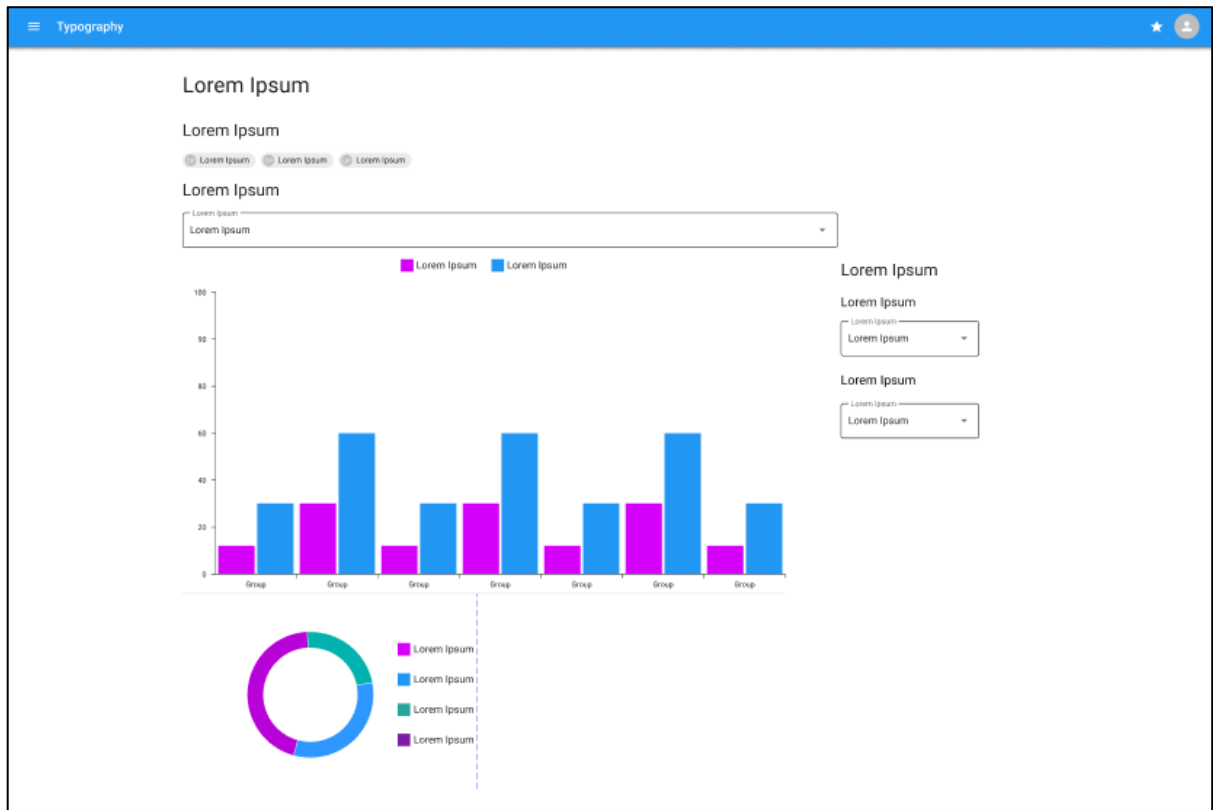


Рисунок 3.12 – Макет оформлення статистики опитування

Для більш детального аналізу, користувач може застосувати фільтри за певними параметрами, що дозволяє зосередитися на статистиці для конкретної групи.

4 ОПИС ПРИЙНЯТИХ ПРОГРАМНИХ РІШЕНЬ

4.1 Архітектура проекту

4.1.1 Вибір фреймворку та інструментів розробки

Для розробки клієнтської частини було обрано фреймворк ReactJS разом з мовою програмування TypeScript, який дозволяє розробляти інтерфейс додатку у вигляді окремих компонентів, які можна повторно використовувати та комбінувати, що робить код структурованішим та легким для подальшої підтримки.

Для реалізації матеріального дизайну та створення інтерфейсу використовується бібліотека MaterialUI, яка надає широкий набір готових компонентів інтерфейсу, таких як кнопки, поля введення, меню, діалогові вікна, таблиці тощо. Це значно спрощує та прискорює розробку інтерфейсу, оскільки не потрібно створювати ці елементи з нуля. Всі компоненти MaterialUI стилізовані відповідно до принципів матеріального дизайну, що забезпечує сучасний та зручний вигляд застосунку.

Приклад коду з використанням обох інструментів можна побачити нижче:

```
export default function CreateAppearancePage() {
  const nav = useNavigate();
  const [templates, setTemplates] = useState<ITemplateResponse[]>([]);
  const [appearance, setAppearance] =
  useState<IUnitAppearanceRequest>({name:"", templateId:0, type:"", params:{}});
  const [selectedTemplate, setSelectedTemplate] =
  useState<ITemplateResponse>({defaultParams:{}, name:"", id:0, templatecode:""});

  useEffect(() => {
    getTemplates();
  }, []);

  const getTemplates = async () => {
    let response = await getSystemTemplates();
    setTemplates(handleGetTemplatesResponse(response));
  }
  return (
    <FormControl sx={{mt:3}} fullWidth>
    <TextField sx={{ mb: 2 }}
      id="nameInput"
      label="Appearance Name"
      name="name"
    />
  )
}
```

```
    value={appearance.name}  
    onChange={handleNameChange}  
></TextField>  
</FormControl>  
)  
}
```

Використання віртуального DOM в ReactJS спрощує оновлення інтерфейсу, що призводить до підвищення продуктивності застосунку. Замість того, щоб оновлювати весь DOM при кожній зміні, ReactJS може оновлювати лише ті частини, які змінилися під час роботи з додатком, що дозволяє зменшити кількість операцій та прискорити роботу застосунку.

4.1.2 Організація файлової структури

Файлова структура проекту "PureSurvey" організована з урахуванням принципів модульності та поділу за відповідальністю, що спрощує навігацію та підтримку коду. Код поділений на незалежні модулі, що відповідають за різні частини застосунку, наприклад, компоненти інтерфейсу, сервіси для взаємодії з API, стилі та допоміжні функції. Кожен модуль має чітко визначену відповідальність, що спрощує розуміння його ролі в системі. Компоненти та допоміжні функції створені таким чином, щоб їх можна було легко повторно використовувати в різних частинах застосунку.

Файлову структуру проекту можна побачити на рисунку 4.1

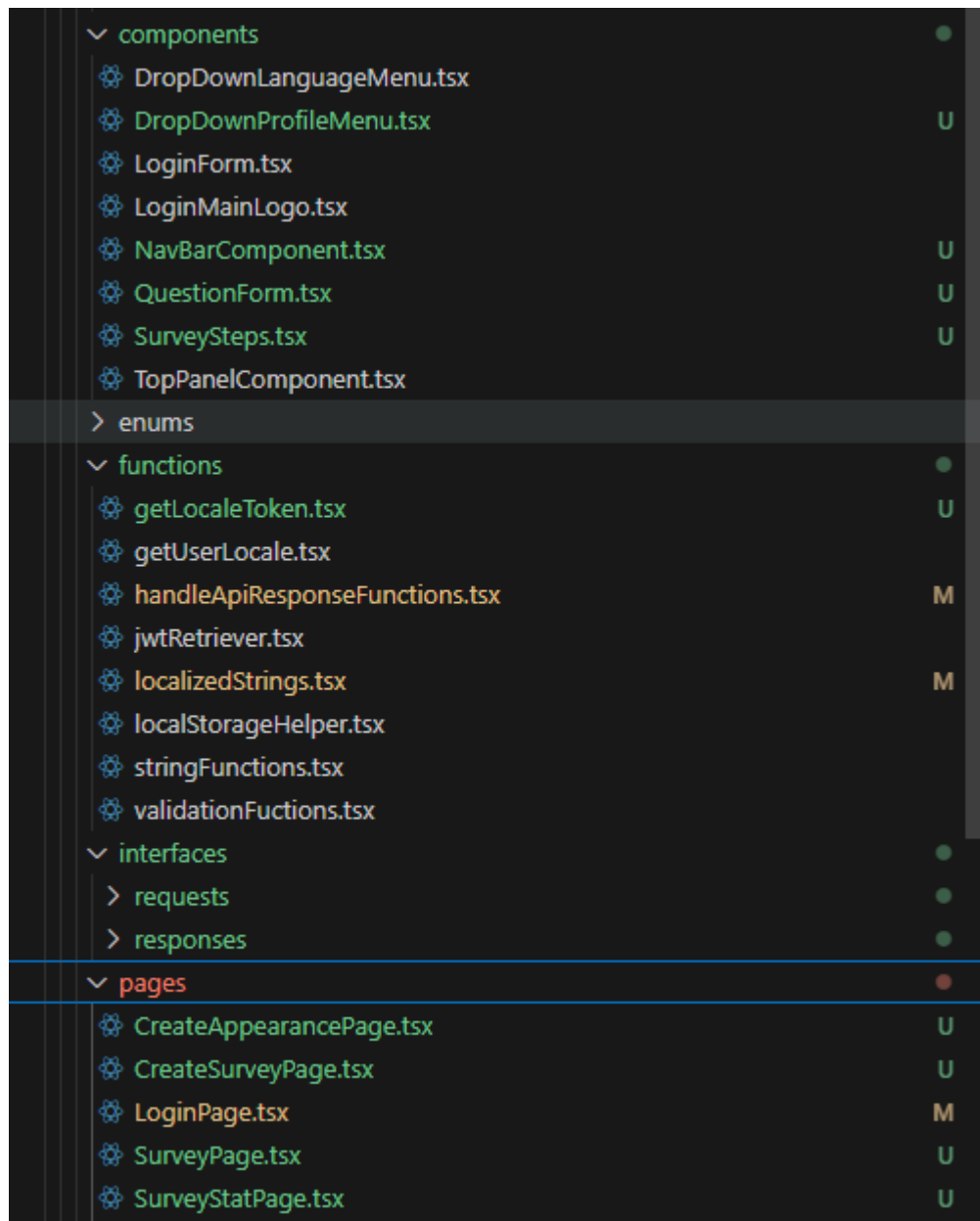


Рисунок 4.1 – Файлова структура проекту

Така організація файлової структури дозволяє ефективно управляти кодом проекту, робить його більш зрозумілим та полегшує спільну роботу над проектом в команді.

4.2 Розробка функцій обробки даних

4.2.1 Отримання даних з API

Для отримання даних з API, таких як інформація про опитування, групи, таргетинги, статистика та інші, використовується окремо створений компонент

FetchData, в якому містяться відповідні функції для відправлення запитів до API, кожна з яких відповідає за конкретний тип запиту і відповідні дані, які отримуються чи надсилаються на сервер. Приклад коду однієї з функцій представлений нижче:

```
export const addSurvey = async (data: ISurveyRequest) => {
  const response = await fetch(`${BASE_URL}/Survey/Survey`, {
    method: 'POST',
    headers: {
      'Authorization' : `Bearer ${getLocaleToken()}`,
      'Content-Type': 'application/json',
      'Accept-Language': getUsersLocale()
    },
    body: JSON.stringify(data)
  });

  const responseJson = await response.json()
  return responseJson;
}
```

Після отримання даних від серверу відбувається їх обробка використовуючи створений компонент handleApiResponseFunctions, який аналізує отримані дані, перевіряє їх на помилки та видає дані отримані від серверу. Приклад коду однієї з функції для оброблення на помилки наведений нижче.

```
export function handleGetTemplatesResponse(res: any): ITemplateResponse[] {
  if (res instanceof Object && 'errors' in res && 'status' in res) {
    const errorResponse = res as IValidationErrorResponse;

    const errorMessages = getValidationErrorMessages(errorResponse);

    errorMessages.forEach((errorMessage) => {
      toast.error(localizeValidationError(errorMessage));
    });
    return [];
  }
  else if (res as IBaseResponse<ITemplateResponse[]>) {
    return handleBaseResponse(res);
  }
  return [];
}
```

Під час виконання даної функції спочатку перевіряється отримана відповідь на наявність помилок в ній. Якщо API повернув помилку, вона класифікується та обробляється шляхом відображення користувачеві зрозумілих повідомлень про

помилки з вказівкою на конкретні поля, які необхідно виправити. У разі помилок на стороні сервера, користувач також отримує загальне повідомлення про помилку.

Якщо була отримана відповідь без помилок, то функція передає значення, отримані від сервера до відповідного компонента ReactJS, у відповідному форматі зазначеному у функції. Таким чином, функції даного компонента виконують роль посередника між API та інтерфейсом користувача.

4.2.2 Авторизація та реєстрація

Для забезпечення доступу до функціональності системи та захисту даних користувачів на клієнтській частині було розроблено систему авторизації та реєстрації користувача. Процес реєстрації або авторизації відбувається за допомогою заповнення форми, вказуючи свою електронну пошту, пароль та іншу необхідну інформацію. Приклад заповнення форми авторизації представлений на рисунку 4.2.

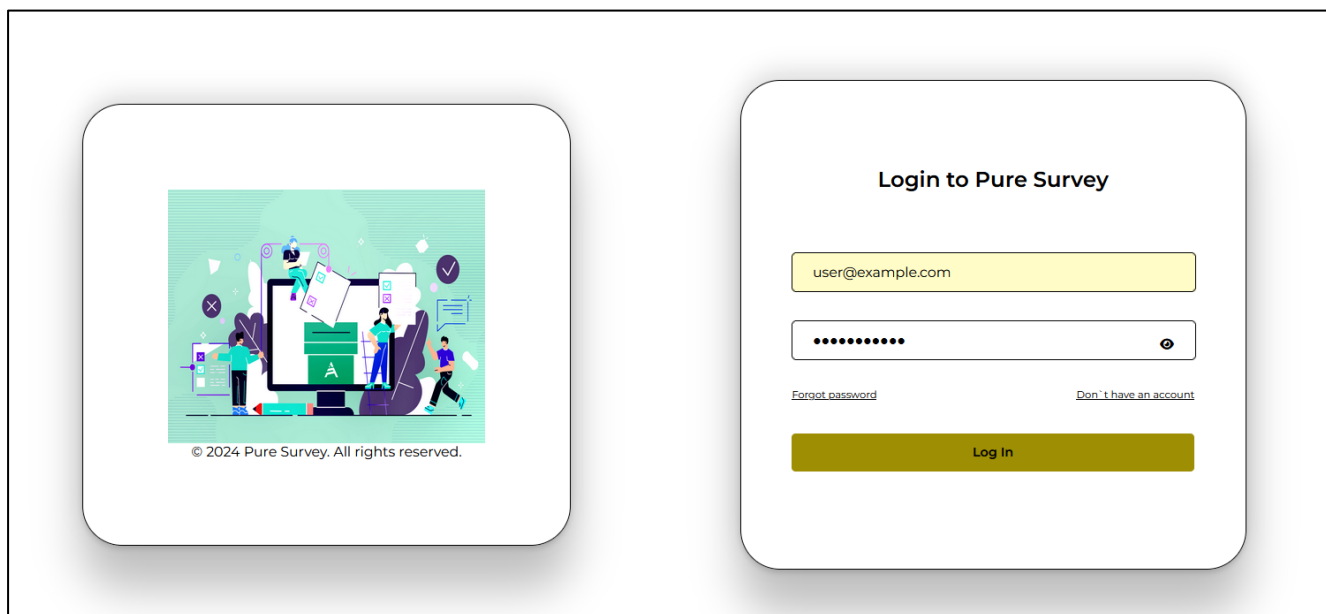


Рисунок 4.2 – Форма авторизації користувача

Після успішної реєстрації або авторизації користувач отримує JWT-токен, який зберігається в локальному сховищі браузера (localStorage). Це дозволяє системі автоматично аутентифікувати користувача при наступних відвідуваннях сайту без необхідності повторного введення логіна та паролю. Перед кожним

запитом до сервера клієнтська частина використовує наявний токен у системі і виконує його валідацію. Якщо токен невалідний або його термін дії закінчився, користувач перенаправляється на сторінку авторизації.

Також у додатку передбачений захист сторінок від переходів неавторизованих користувачів на них використовуючи розроблену систему авторизації. Приклад коду захисту можна побачити нижче.

```
const ProtectedRoute = () => {
  const isAuthenticated = authService.isAuthenticated();
  return isAuthenticated ? <Outlet /> : <Navigate to="/login" replace />
};
```

Можна побачити що спочатку функція перевіряє на те чи авторизований користувач і в залежності від результату пересилає його на сторінку повторної авторизації на сайті або видає результат з відповідним посиланням, на яке відбувався перехід користувачем.

4.2.3 Валідація даних

У разі виявлення помилок валідації користувачеві відображаються зрозумілі та інформативні повідомлення про помилки, використовуючи бібліотеку React Toastify. Вона надає зручний інтерфейс для повідомлень, який відповідає стилістиці дизайну додатку та окремо можна налаштовувати відповідні параметри контейнеру. Для відображення повідомлень використовується компонент `<ToastContainer>`, який необхідно додати до основного компонента який відображається користувачеві. Приклад коду контейнеру та його використання представлений нижче.

```
<ToastContainer
  position="top-right"
  autoClose={5000}
  hideProgressBar={false}
  newestOnTop={false}
  closeOnClick
  rtl={false}
  pauseOnFocusLoss={false}
  draggable
  pauseOnHover
  theme={props.theme === 'dark' ? 'dark' : 'light'}
```

```
 />  
 ...  
 toast.error(localizeValidationError(errorMessage));  
 ...
```

Приклад відображеної помилки, яка була отримана з серверу, показана на рисунку 4.3.

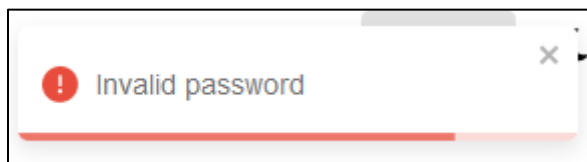


Рисунок 4.3 – Відображення помилки, отриманої від серверу

Реалізація валідації даних на клієнтській частині спроектована у вигляді первинної валідації даних, безпосередньо в браузері користувача. Це дозволяє швидко повідомити користувача про помилки та запобігти непотрібним запитам до сервера. А також після того вони відправляються на сервер для повторної перевірки – це забезпечує додатковий рівень захисту від помилок під час роботи з додатком.

4.3 Розробка основних компонентів

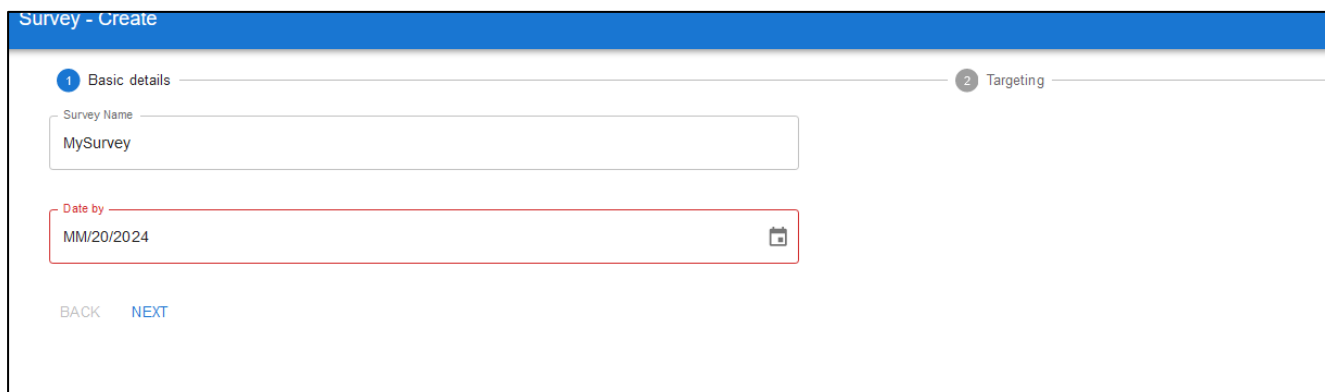
4.3.1 Створення опитувань

Для забезпечення гнучкості та зручності цього процесу в клієнтській частині реалізовано інтерактивний інтерфейс з використанням компонента Stepper.

Компонент Stepper розбиває процес створення опитування на три логічних кроки: "Basic details", "Targeting" та "Question". Кожен крок відображається на окремій сторінці, що дозволяє користувачеві зосередитися на конкретному етапі та уникнути перевантаження інформацією.

На сторінці створення опитування розміщені кнопки, які забезпечують навігацію між етапами. Кнопки "Back" та "Next" відповідають за перехід вперед та назад по етапах створення опитування.

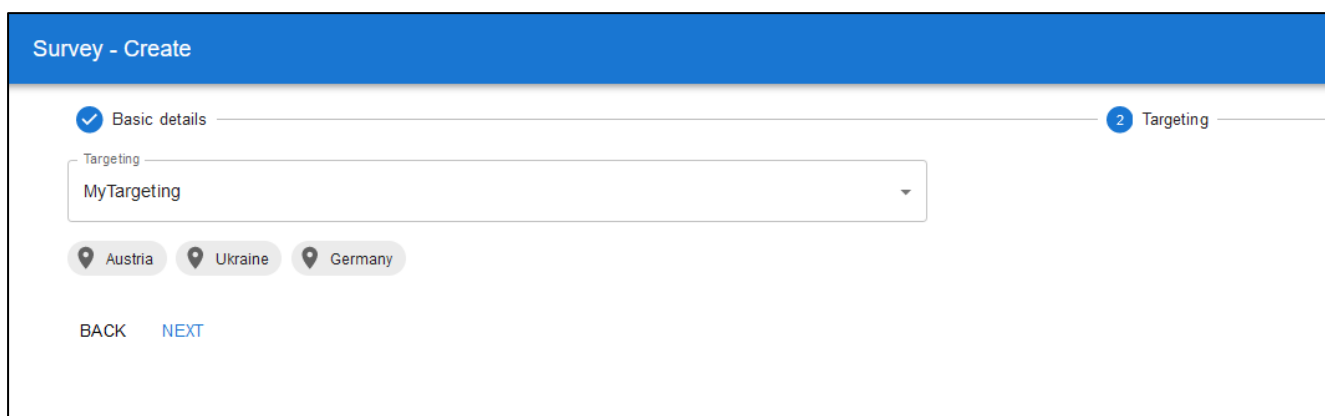
На першому кроці користувач вводить основну інформацію про опитування, таку як назва та дата закінчення. Приклад форми для введення даних зображений на рисунку 4.4



The screenshot shows a web interface titled "Survey - Create". It has two steps: "1 Basic details" and "2 Targeting". In the "Basic details" step, there is a text input field for "Survey Name" with the value "MySurvey" and a date input field for "Date by" with the value "MM/20/2024". Below the inputs are "BACK" and "NEXT" buttons. The "Targeting" step is currently inactive.

Рисунок 4.4 – Інтерфейс сторінки заповнення даних опитування

Користувач має ввести відповідні дані опитування, а саме назву та дату по яке опитування буде активним. Після чого переходить на сторінку з вибором таргетингу опитування, а саме вибір з наявних груп країн створених користувачем. Обравши потрібний таргетинг користувачу будуть виведені на екран відповідні країни, які знаходяться у переліку з даним таргетингом. Приклад відображення обраного значення можна побачити на рисунку 4.5



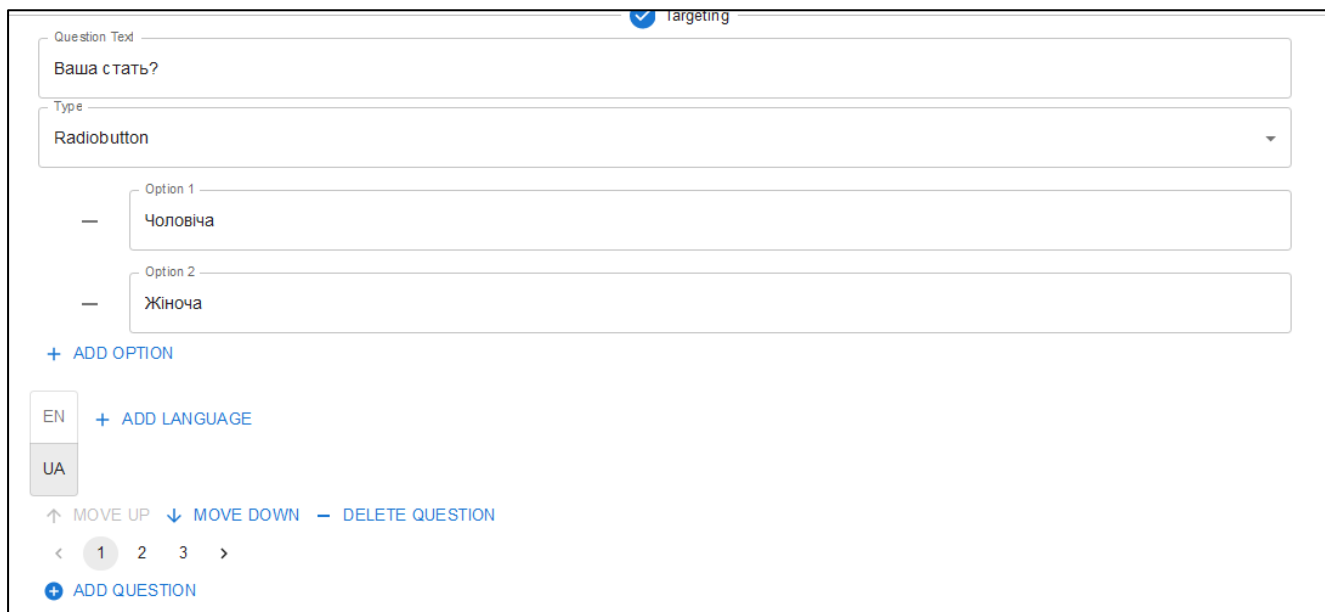
The screenshot shows the "Survey - Create" interface at the "2 Targeting" step. The "Basic details" step is now completed, indicated by a checkmark. The "Targeting" dropdown menu is open, showing "MyTargeting". Below the dropdown, there are three location pins for "Austria", "Ukraine", and "Germany". "BACK" and "NEXT" buttons are visible at the bottom.

Рисунок 4.5 – Інтерфейс відображення країн задіяних в таргетингу

В кінці користувач зустрічається з формою для створення опитувань. Тут реалізовано гнучкий механізм динамічного створення форм, що дозволяє

користувачеві легко формувати опитування будь-якої складності. При натисканні кнопки для додавання нового питання викликається функція, яка додає новий об'єкт до масиву усіх питань цього опитування.

Приклад інтерфейсу форми представлений на рисунку 4.6



The screenshot shows a web interface for adding a question. At the top right, there is a 'targeting' status indicator. The main form consists of several sections: a 'Question Text' field containing 'Ваша стаття?'; a 'Type' dropdown menu set to 'Radiobutton'; two 'Option' fields, 'Option 1' with 'Чоловіча' and 'Option 2' with 'Жіноча'; a '+ ADD OPTION' button; a language selection section with 'EN' and 'UA' buttons and a '+ ADD LANGUAGE' button; and navigation controls including 'MOVE UP', 'MOVE DOWN', 'DELETE QUESTION', a pagination indicator showing '1', '2', '3', and an 'ADD QUESTION' button at the bottom.

Рисунок 4.6 – Інтерфейс форми для додавання питань

Для зменшення кількості інформації яка буде відображена на екрані, наприклад, кількість полів для введення варіантів відповідей для кожного питання і його перекладу, було спроектовано відповідну форму з панеллю керування та переміщення між питаннями і їх перекладами. Користувач може переміщуватися між питаннями використовуючи компонент `Pagination`, де відображений порядковий номер поточного питання. Користувач може відповідно змінити порядок питання натиснувши на відповідні кнопки «Move Up» та «Move Down», а також видалити поточне питання.

Додатково для користувача присутня можливість додавання перекладу для опитування. Це відбувається за допомогою компоненту-перемикача `ToggleButton`, який відображає код мови у списку кнопок перемикача. Додавання відбувається за допомогою відповідної кнопки, яка виводить на екран поле для вводу значення коду мови. Відповідно для цього була додана функціональність при зміні значень

полів, в якій визначається код мови, яка обрана користувачем і відповідно будуть заповненні значення об'єктів до відповідного перекладу. Приклад програмного коду для створення мови питання наведений нижче.

```

const newLanguage = prompt('Enter new language code (e.g., "fr" for
French):')?.toLocaleLowerCase();
if (newLanguage && !availableLanguages.includes(newLanguage)) {
  setAvailableLanguages([...availableLanguages, newLanguage]);
  const updatedQuestions = [...questions];
  updatedQuestions[questionIndex].questionLine.translations.push({
languageCode: newLanguage, translationText: '' });
  updatedQuestions[questionIndex].questionOptions.forEach((option: {
translations: ITranslationLine[]; }) => {
    option.translations.push({ languageCode: newLanguage, translationText:
'' });
  });
  setQuestions(updatedQuestions);
}

```

Після створення питання користувачеві виводяться згенеровані поля для вводу тексту питання та варіантів відповідей, які він може додати натиснувши на відповідні кнопки для додавання варіанту відповіді, а для видалення – на значок зліва від поля для введення варіанту відповіді. Під час введення даних у поля відбувається запит на зміну відповідних значень у створених об'єктах питань, використовуючи. Приклад програмного коду оновлення варіанту відповіді наведений нижче.

```

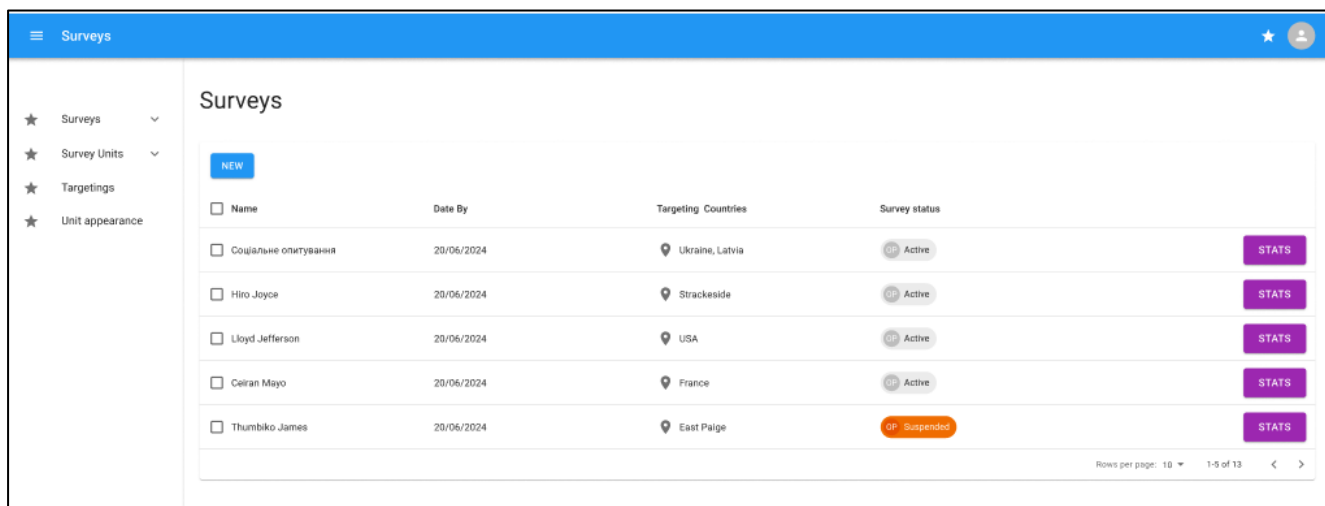
const handleChangeOption = () => {
  const updatedQuestions = [...questions];
  const translation =
updatedQuestions[questionIndex].questionOptions[idx].translations.find((t: {
languageCode: string; }) => t.languageCode === language);
  if (translation) {
    translation.translationText = e.target.value;
  } else {
    updatedQuestions[questionIndex].questionOptions[idx].translations.push({
languageCode: language, translationText: e.target.value });
  }
  setQuestions(updatedQuestions);
};

```

Після завершення всіх трьох кроків користувач натискає кнопку «Save», яка відправляє запит на створення опитування на сервер. Після чого компонент

обробляє відповідь від API, перевіряє її на помилки та перенаправляє користувача на сторінку списку опитувань в разі успішного створення.

Для кожного з основних об'єктів були створені сторінки для перегляду інформації у табличному вигляді, в якій міститься основна інформація яка потрібна користувачу. Приклад інтерфейсу сторінки для опитувань представлений на рисунку 4.7



| <input type="checkbox"/> | Name | Date By | Targeting Countries | Survey status | |
|--------------------------|----------------------|------------|---------------------|---------------|-------|
| <input type="checkbox"/> | Соціальне опитування | 20/06/2024 | Ukraine, Latvia | Active | STATS |
| <input type="checkbox"/> | Hiro Joyce | 20/06/2024 | Strackeside | Active | STATS |
| <input type="checkbox"/> | Lloyd Jefferson | 20/06/2024 | USA | Active | STATS |
| <input type="checkbox"/> | Celran Mayo | 20/06/2024 | France | Active | STATS |
| <input type="checkbox"/> | Thumbiko James | 20/06/2024 | East Paige | Suspended | STATS |

Рисунок 4.7 – Інтерфейс сторінки зі списком опитувань

У таблиці зі списком опитування, у кожного зі створених опитувань коротко написана інформація, а саме: назва опитування, дата по яку буде активне це опитування, таргетовані країни (назва таргетингу) та статус опитування. Також користувач може переглянути статистику певного опитування, натиснувши на кнопку «STATS» яка відповідає за кожне опитування у списку. Вигляд інтерфейсу статистичних даних по певному опитуванню зображено на рисунку 4.8

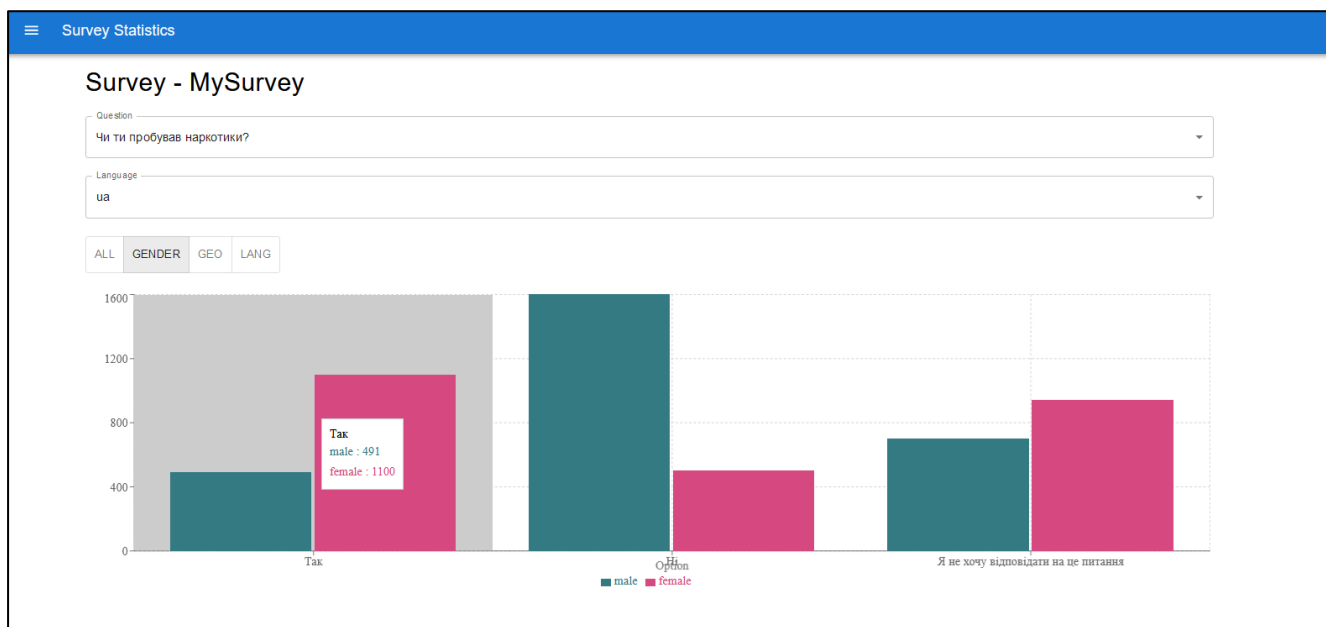


Рисунок 4.8 – Інтерфейс сторінки статистики опитування

Сторінка статистики опитування дозволяє користувачам проаналізувати отримані результати. В верхній частині сторінки розташовано назву опитування та список країн, на які воно було націлене.

Для зручності аналізу користувач може обрати конкретне питання з опитування з випадючого списку та застосувати фільтри за країною, статтю та мовою респондентів, щоб зосередитися на статистиці для конкретної групи. І додатково обрати мову перекладу опитування з випадючого списку нижче для перегляду у зручному форматі для користувача. Частина програмного коду для фільтрації даних представлена нижче.

```
const transformData = (): { [key: string]: any }[] => {
  if (!stats) return [];

  let transformedData: { [key: string]: any }[] = [];

  if (filter === 'Gender') {
    const groupedData = stats.statsForGender.reduce((acc: { [key: string]: any }, stat) => {
      const optionName = getOptionName(stat.optionId);
      if (!acc[optionName]) acc[optionName] = { option: optionName };
      acc[optionName][stat.gender] = stat.answered;
      return acc;
    }, {});
  }
  ...
}
```

Центральним елементом сторінки є блок візуалізації, де дані представлені у вигляді інтерактивної стовпчастої діаграми, яка дозволяє візуалізувати розподіл відповідей на обране питання за варіантами відповідей і переглянути отримані значення по результатам опитування. Для реалізації діаграми була використана бібліотека `recharts` і компоненту `BarChart` який дозволяє створювати стовпчасті діаграми, змінюючи потрібні параметри для відображення на сторінці.

Інтерактивність діаграми реалізована за допомогою `React hooks` (`useState`, `useEffect`) для відстеження змін обраного питання та фільтрів. При зміні цих параметрів викликається функція `fetchStats`, яка завантажує статистику з API за допомогою `getQuestionStats` з компонента `FetchData`. Отримані дані обробляються та трансформуються в формат, потрібний для бібліотеки `recharts`, яка використовується для рендерингу діаграм. Зміна стану компонента, що зберігає дані статистики, автоматично викликає перерендеринг діаграм, виконуючи динамічне оновлення візуалізації відповідно до обраних параметрів. Нижче наведений програмний код для відображення діаграми на сторінці:

```
const renderChart = () => {
  const data = transformData();
  if (data.length === 0) return null;

  const keys = Object.keys(data[0]).filter(key => key !== 'option');

  return (
    <ResponsiveContainer width="100%" height={400}>
      <BarChart data={data}>
        <CartesianGrid strokeDasharray="3 3" />
        <XAxis dataKey="option" >
          <Label value="Option" position="insideBottom" />
        </XAxis>
        <YAxis />
        <Tooltip />
        <Legend />
        {keys.map(key => (
          <Bar key={key} dataKey={key} fill={key === 'answered' ? "#8884d8"
: getRandomColor()} />
        ))}
      </BarChart>
    </ResponsiveContainer>
  );
};
```

Для зовнішнього вигляду групи опитувань було реалізована спеціальна сторінка створення дозволяє користувачам створювати унікальні зовнішні вигляди для груп опитувань, використовуючи різні шаблони та змінюючи їх налаштування.

Основними елементами сторінки є:

- а) вибір шаблону: користувач може обрати шаблон з випадаючого списку, які наявні у системі.
- б) налаштування параметрів: після вибору шаблону користувачеві відображаються поля для налаштування параметрів зовнішнього вигляду, специфічних для обраного шаблону.
- в) попередній перегляд: користувач може переглянути зовнішній вигляд опитування з обраним шаблоном та налаштуваннями в режимі реального часу.

При завантаженні сторінки, застосунок автоматично отримує список доступних шаблонів з сервера за допомогою функції `getSystemTemplates` з компонента `FetchData.tsx`. Отримані дані обробляються та формуються у список об'єктів, які потім використовуються для відображення випадаючого списку з доступними шаблонами.

Коли користувач обирає шаблон, застосунок визначає його параметри, збережені в властивості `defaultParams`, та динамічно створює поля введення для налаштування цих параметрів. Залежно від типу параметра, використовуються різні компоненти: `MuiColorInput` для вибору кольору та `TextField` для введення тексту. Такий підхід дозволяє гнучко налаштовувати зовнішній вигляд опитувань, забезпечуючи інтуїтивно зрозумілий інтерфейс для користувача. Програмний код генерування полів наведено нижче:

```
{transformParams(selectedTemplate?.defaultParams).map((param: Param) => (
  (param.name.toLowerCase().includes("color"))?(
    <FormControl sx={{mt:3}} fullWidth>
      <MuiColorInput label={param.name}
        format="hex">
```

```

    value={appearance.params[param.name]||''}
    onChange={(value) => handleParamChange(param.name, value)} />
  </FormControl>
):(

  <FormControl sx={{mt:3}} fullWidth>
  <TextField
  label={param.name}
  value={appearance.params[param.name]||''}
  InputProps={{
    endAdornment: <InputAdornment position="end">px</InputAdornment>
  }}
  onChange={(event) => handleParamChange(param.name,
event?.target.value)} />
  </FormControl>
  )})}

```

Також на сторінці створення користувач може переглянути попередній вигляд питання опитування. З серверу функція отримує HTML-код шаблону з властивості обраного шаблону. Потім вона проходить по всіх параметрах зовнішнього вигляду, збережених в об'єкті створеного зовнішнього вигляду, та замінює відповідні значення в HTML-кодi шаблону на значення, введені користувачем. Приклад інтерфейсу сторінки створення представлений на рисунку 4.9.

Appearance Name
MyAppearance

Template
DefaultTemplate

Color1\$
#ff1216

Sradius1\$
20 px

BACK SAVE

Survey Template

What is the highest mountain on the Earth?

- Mount Everest
- Makalu
- Manaslu
- Broad Peak

Prev Next

Рисунок 4.9 – Інтерфейс сторінки створення зовнішнього вигляду групи опитувань

Таким чином, користувач може бачити попередній перегляд зовнішнього вигляду опитування в режимі реального часу, що дозволяє йому експериментувати з різними шаблонами та налаштуваннями та створювати унікальний дизайн.

Додатково користувач може натиснути на значок профілю на навігаційному мені і далі виникне випадаюче меню, в якому написана коротка інформація про обліковий запис і вибір функціоналу, а саме: зміна мови, стилю сайту, перехід до профілю користувача і виходу з облікового запису. Інтерфейс випадаючого меню представлений на рисунку 4.10.

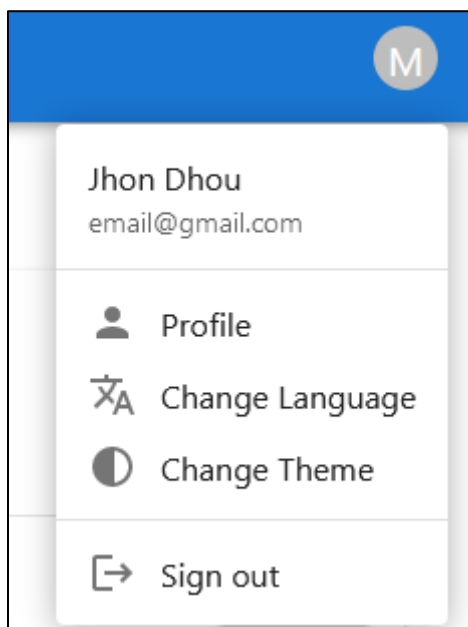


Рисунок 4.10 – Інтерфейс випадаючого меню з налаштуваннями користувача

Таким чином, було детально розглянуто прийняті програмні рішення для реалізації клієнтської частини системи. Особливу увагу приділено опису розроблених функцій обробки даних, включаючи отримання даних з API, реалізацію авторизації та детально розглянуто створення ключових компонентів системи, таких як форми для створення опитувань, інструменти налаштування зовнішнього вигляду та функціонал для перегляду статистики.

Реалізовані рішення забезпечують гнучкість, масштабованість та високу продуктивність системи, роблячи її зручним інструментом для проведення онлайн-опитувань.

5 ТЕСТУВАННЯ РОЗРОБЛЕНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

5.1 Розробка тестових випадків

Для тестування клієнтської частини було обрано тип тестування за допомогою мануальних тестів, оскільки вони найкраще підходять для оцінки якості користувацького інтерфейсу з точки зору зручності та правильності роботи компонентів.

Даний вид тестування дає можливість виконувати дії з точки зору користувача та виконати типові сценарії взаємодії з інтерфейсом, виявляючи потенційні проблеми з навігацією, розташуванням елементів та загальним сприйняттям інформації.

В таблицях 5.1-5.7 наведений список основних тест-кейсів, який був розроблений під час тестування клієнтської частини.

Таблиця 5.1 – Тест-кейс №1

| Параметр тестування | Значення |
|----------------------|---|
| Назва тесту | Авторизація з коректними даними |
| Опис тесту | Перевірка успішного входу користувача з коректними даними. |
| Компонент системи | Авторизації та реєстрації |
| Кроки відтворення | 1. Відкрити сторінку авторизації. 2. Ввести валідну електронну пошту зареєстрованого користувача. 3. Ввести коректний пароль. 4. Натиснути кнопку "Увійти". |
| Очікуваний результат | 1. Система успішно авторизує користувача та перенаправляє на відповідну сторінку (залежно від ролі користувача). |
| Фактичний результат | Відповідає очікуваному |
| Статус | Тест-кейс відтворений |
| Коментар | - |

Таблиця 5.2 – Тест-кейс №2

| | |
|----------------------|---|
| Параметр тестування | Значення |
| Назва тесту | Авторизація з некоректним паролем |
| Опис тесту | Перевірка блокування авторизації з некоректним паролем. |
| Компонент системи | Компонент авторизації та реєстрації |
| Кроки відтворення | 1. Відкрити сторінку авторизації. 2. Ввести валідну електронну пошту зареєстрованого користувача. 3. Ввести некоректний пароль. 4. Натиснути кнопку "Увійти". |
| Очікуваний результат | 1. Система блокує авторизацію та відображає повідомлення про помилку. 2. Повідомлення про помилку вказує на некоректний пароль. |
| Фактичний результат | Відповідає очікуваному |
| Статус | Тест-кейс відтворений |
| Коментар | - |

Таблиця 5.3 – Тест-кейс №3

| | |
|----------------------|---|
| Назва тесту | Створення опитування з заповненням всіх полів |
| Опис тесту | Перевірка коректної роботи форми створення опитування та збереження даних. |
| Компонент системи | Компонент управління опитуваннями |
| Кроки відтворення | 1. Авторизуватися як зареєстрований користувач. 2. Перейти на сторінку створення опитування. 3. Заповнити всі поля форми, включаючи назву, дату закінчення, таргетинг та питання. 4. Натиснути кнопку "Зберегти". |
| Очікуваний результат | 1. Система успішно створює нове опитування та зберігає його дані. 2. Користувач отримує повідомлення про успішне створення опитування. |

Продовження таблиці 5.3

| | |
|---------------------|--|
| | 3. Користувач перенаправляється на сторінку списку опитувань. |
| Фактичний результат | 1. Була виведена помилка про неправильний формат дати, яка була відправлена запитом на створення опитування. |
| Статус | Тест-кейс відтворений, баг виправлений. |
| Коментар | Було знайдено помилку під час форматуванні отриманого значення дати і додано відповідний метод для зміни формату, який буде коректний під час відправлення запиту. |

Таблиця 5.4 – Тест-кейс №4

| | |
|----------------------|---|
| Назва тесту | Додавання нового перекладу до опитування |
| Опис тесту | Перевірка коректної роботи функції додавання нового перекладу для усіх опитувань. |
| Компонент системи | Компонент створення опитувань |
| Кроки відтворення | 1. Авторизуватися як зареєстрований користувач. 2. Перейти на сторінку редагування опитування. 3. Натиснути кнопку для додавання перекладу. 4. Заповнити всі поля форми включаючи текст питання, варіанти відповідей та їх переклади. 5. Натиснути кнопку "Зберегти". |
| Очікуваний результат | 1. Система успішно додає новий переклад до опитування. 2. Питання відображається в списку питань з перекладами на всі обрані мови. |
| Фактичний результат | Відповідає очікуваному |
| Статус | Тест-кейс відтворений |
| Коментар | - |

Таблиця 5.5 – Тест-кейс №5

| | |
|----------------------|---|
| Назва тесту | Перегляд списку груп опитувань |
| Опис тесту | Перевірка відображення списку створених користувачем груп опитувань. |
| Компонент системи | Компонент управління групами опитувань |
| Кроки відтворення | 1. Авторизуватися як зареєстрований користувач. 2. Перейти на сторінку "Групи". |
| Очікуваний результат | 1. Система відображає список створених користувачем груп опитувань. 2. Кожна група опитувань відображає свою назву та списки опитувань в ньому. |
| Фактичний результат | Відповідає очікуваному |
| Статус | Тест-кейс відтворений |
| Коментар | - |

Таблиця 5.6 – Тест-кейс №6

| | |
|----------------------|--|
| Назва тесту | Перегляд статистики по гендерам |
| Опис тесту | Перевірка відображення статистики по гендерам по всіх опитуванням користувача. |
| Компонент системи | Компонент перегляду статистики |
| Кроки відтворення | 1. Авторизуватися як зареєстрований користувач. 2. Перейти на сторінку зі списком опитувань і обрати потрібне опитування. 3. Перейти на сторінку статистики опитування, натиснувши на кнопку "STATS". 4. Обрати потрібне питання зі списку наявних в опитування і групування по гендеру. |
| Очікуваний результат | 1. Система відображає статистику по гендерам по всіх опитуванням користувача, включаючи кількість переглядів, кількість відповідей тощо. |

Продовження таблиці 5.6

| | |
|---------------------|---|
| Фактичний результат | Отримані коректні значення для статистики, але для перекладів варіантів відповідей – відображались зі зламаним кодуванням |
| Статус | Тест-кейс відтворений, баг виправлено |
| Коментар | Було виправлено кодування для бази даних програмної системи для потрібних полів, оскільки вони підтримували лише латинські символи кодування. |

Таблиця 5.7 – Тест-кейс №7

| | |
|----------------------|--|
| Назва тесту | Створення Appearance з невалідними значеннями параметрів |
| Опис тесту | Перевірка валідації полів параметрів шаблону при створенні Appearance. |
| Компонент системи | Компонент управління Appearance |
| Кроки відтворення | 1. Авторизуватися як зареєстрований користувач. 2. Перейти на сторінку "Створити Appearance". 3. Ввести унікальну назву. 4. Обрати шаблон, тип розташування та налаштувати параметри, вводячи невалідні значення (наприклад, текст в полі для вибору кольору). 5. Натиснути кнопку "Зберегти". |
| Очікуваний результат | 1. Система блокує створення Appearance та відображає повідомлення про помилку. 2. Повідомлення про помилку вказує на невалідні значення в полях параметрів. 3. В попередньому перегляді будуть використані останні валідні значення. |
| Фактичний результат | Відповідає очікуваному |
| Статус | Тест-кейс відтворений |
| Коментар | - |

Мануальне тестування вимагає більших зусиль, але без нього ми не зможемо переконатися в тому, чи можлива автоматизація взагалі.. Проведення тестування для перевірки максимально можливої кількості шляхів виконання програми із використанням мінімального числа тест-кейсів вимагає серйозних аналітичних навичок.

ВИСНОВКИ

Під час виконання кваліфікаційної роботи було розроблено клієнтську частину програмної системи "PureSurvey", що надає користувачам зручний та інтуїтивно зрозумілий інтерфейс для створення, налаштування та аналізу онлайн-опитувань.

В процесі роботи було проведено детальний аналіз предметної області, визначено цільову аудиторію та її потреби, а також проаналізовано існуючі аналоги та їх недоліки. На основі отриманої інформації було сформовано перелік функціональних та нефункціональних вимог до системи, які були задокументовані в SRS документі.

Було спроектовано архітектуру клієнтської частини з використанням фреймворку ReactJS та мови програмування TypeScript, що забезпечує сучасний дизайн, гнучкість та модульність коду. Реалізовано основні функції системи, такі як авторизація, реєстрація, створення та редагування опитувань, їх груп, налаштування зовнішнього вигляду, таргетинг, перегляд та експорт статистики.

Особлива увага була приділена створенню зручного та інтерактивного інтерфейсу для роботи з питаннями опитування, дозволяючи динамічно додавати, видаляти та редагувати питання, а також налаштовувати їх переклад на різні мови.

В цілому, розроблена клієнтська частина програмної системи відповідає сучасним стандартам веб-розробки та надає користувачам зручний та ефективний інструмент для проведення онлайн-опитувань.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Опитування – Вікіпедія. Вікіпедія [Електронний ресурс]. URL: <https://uk.wikipedia.org/wiki/Опитування> (дата звернення: 01.06.2024).
2. Іващук І. М. Як соціальні опитування впливають на думку українських громадян під час передвиборчої кампанії / Інеса Миколаївна Іващук // Сучасні політичні процеси: глобальний та національний виміри : матер. Міжнар. наук.-практ. інтернет-конф. (м. Одеса, 28 верес. 2018 р.) / НУ "ОЮА". – Одеса, 2018. – С. 38–41.
3. Войтович С. Я., Ковальчук О. В. Формування стратегічних пріоритетів маркетингової діяльності підприємства на ринку на основі опитування споживачів. Економічні науки. Серія "Регіональна економіка". 2022. № 19(75). С. 42–50.
4. Сидоров, М. С. Про деякі особливості проведення опитувань у Інтернеті. Актуальні проблеми соціології, психології, педагогіки: Збірник наукових праць. 2012. вип. 16. С. 67-76.
5. Клієнт-серверна архітектура. QATestLab [Електронний ресурс]. URL: <https://training.qatestlab.com/blog/technical-articles/client-server-architecture/> (дата звернення: 01.06.2024).
6. JavaScript with syntax for types. TypeScript: JavaScript With Syntax For Types [Електронний ресурс]. URL: <https://www.typescriptlang.org/> (дата звернення: 01.06.2024).
7. React – JavaScript-бібліотека для створення користувацьких інтерфейсів [Електронний ресурс]. URL: <https://uk.legacy.reactjs.org/> (дата звернення: 01.06.2024).
8. Content design – Material Design 3. Material Design [Електронний ресурс]. URL: <https://m3.material.io/foundations/content-design/overview> (дата звернення: 01.06.2024).
9. Material UI: React components that implement Material Design [Електронний ресурс]. URL: <https://mui.com/material-ui/> (дата звернення: 01.06.2024).

10. Kopf B. The Power of Figma as a Design Tool [Электронный ресурс]. URL: <https://www.toptal.com/designers/ui/figma-design-tool> (дата звернення: 01.06.2024).

ДОДАТОК А.

Звіт результатів перевірки на унікальність тексту в базі ХНУРЕ

UNICHECK
by Turnitin

| | |
|--|----------------------------------|
| Ім'я користувача: Олійник Олена Володимирівна каф. ПІ | ID перевірки: 1016337172 |
| Дата перевірки: 09.06.2024 08:26:55 EEST | Тип перевірки: Doc vs Library |
| Дата звіту: 09.06.2024 08:39:39 EEST | ID користувача: 100012353 |

Назва документа: 2024_Б_ПІ_ПЗПІ_20_6_Печій_Р_Р

Кількість сторінок: 50 Кількість слів: 7626 Кількість символів: 62068 Розмір файлу: 1.06 MB ID файлу: 1016138113

Виявлено модифікації тексту (можуть впливати на відсоток схожості)

1.26%
Схожість

Найбільша схожість: 0.29% з джерелом з Бібліотеки (ID файлу: 1015160027)

Пошук збігів з Інтернетом не проводився

1.26% Джерела з Бібліотеки 141 Сторінка 52

0% Цитат

Вилучення цитат вимкнене

Вилучення списку бібліографічних посилань вимкнене

0% Вилучень

Немає вилучених джерел

Модифікації

Виявлено модифікації тексту. Детальна інформація доступна в онлайн-звіті.

Підозріле форматування 11 сторінок

Рисунок А.1 – Результати перевірки на унікальність тексту в базі ХНУРЕ

ДОДАТОК Б.
Слайди презентації

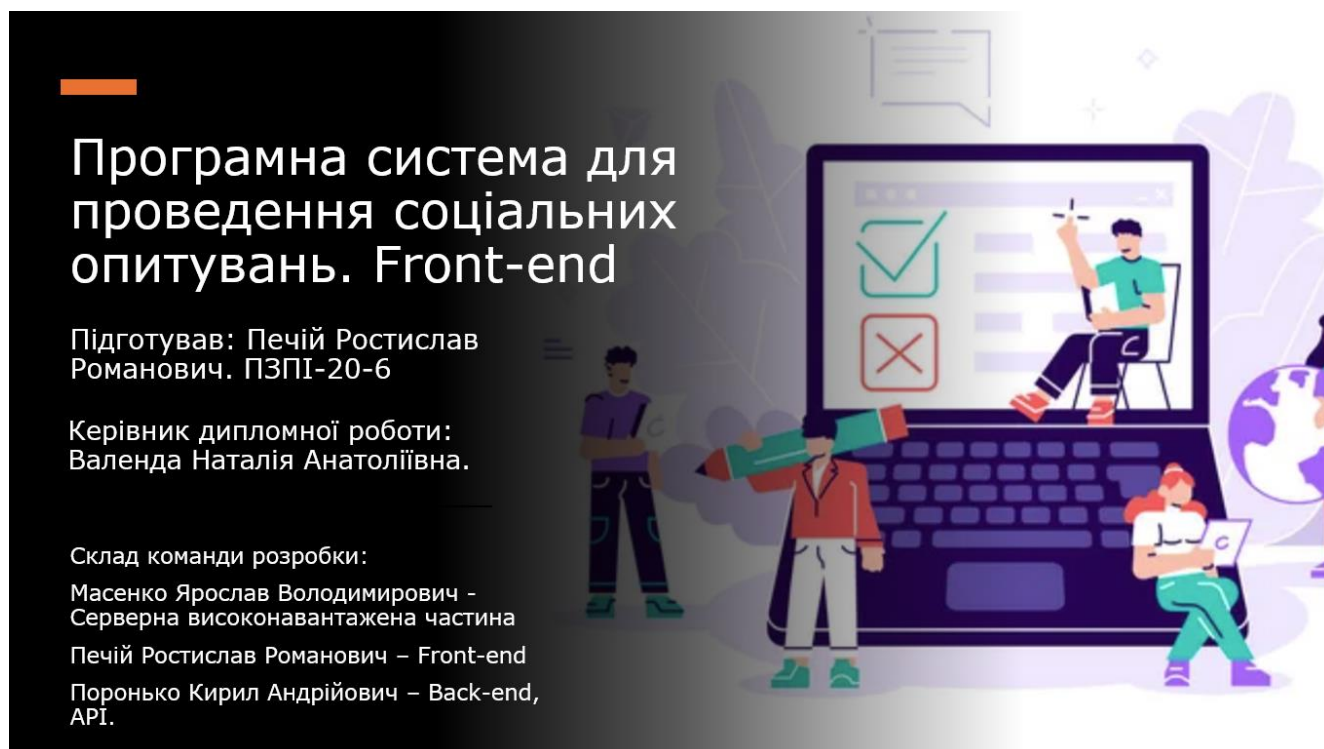


Рисунок Б.1 – Слайд №1

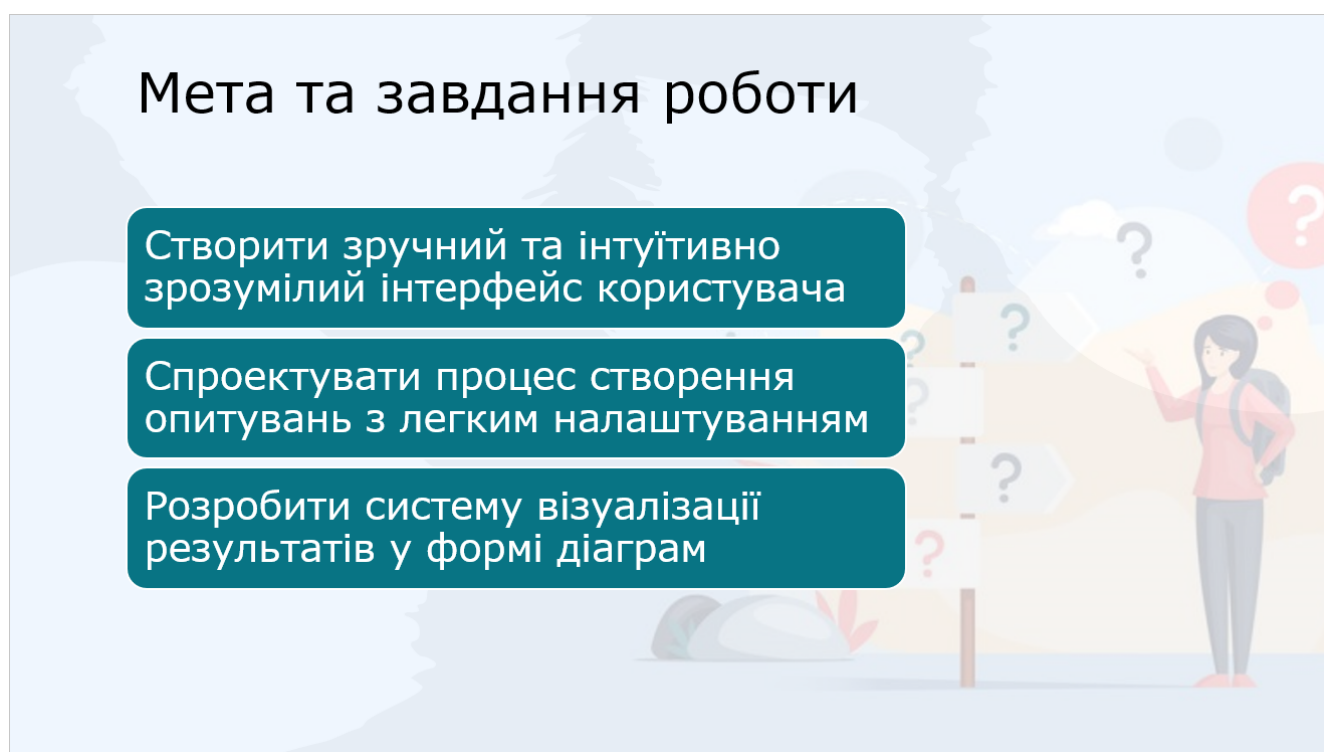


Рисунок Б.2 – Слайд №2

Актуальність роботи

Зростаюча потреба на проведення та створення гнучких анкет

Розробка візуально привабливої анкети може збільшити якість отриманих даних

Використання сучасних технологій для створення форм анкет

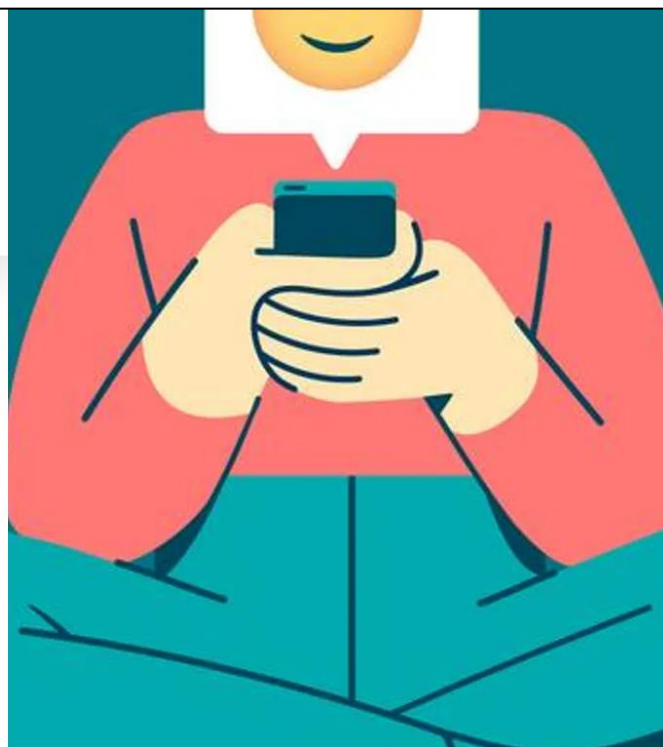


Рисунок Б.3 – Слайд №3

Огляд існуючих рішень

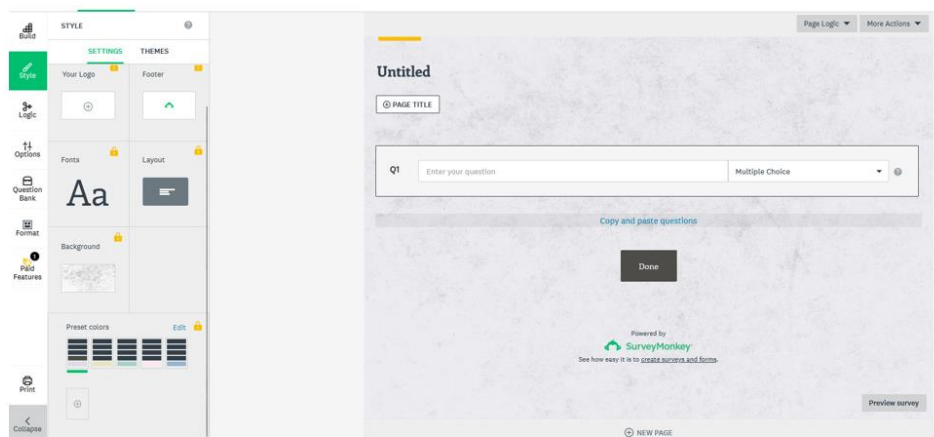


Рисунок Б.4 – Слайд №4

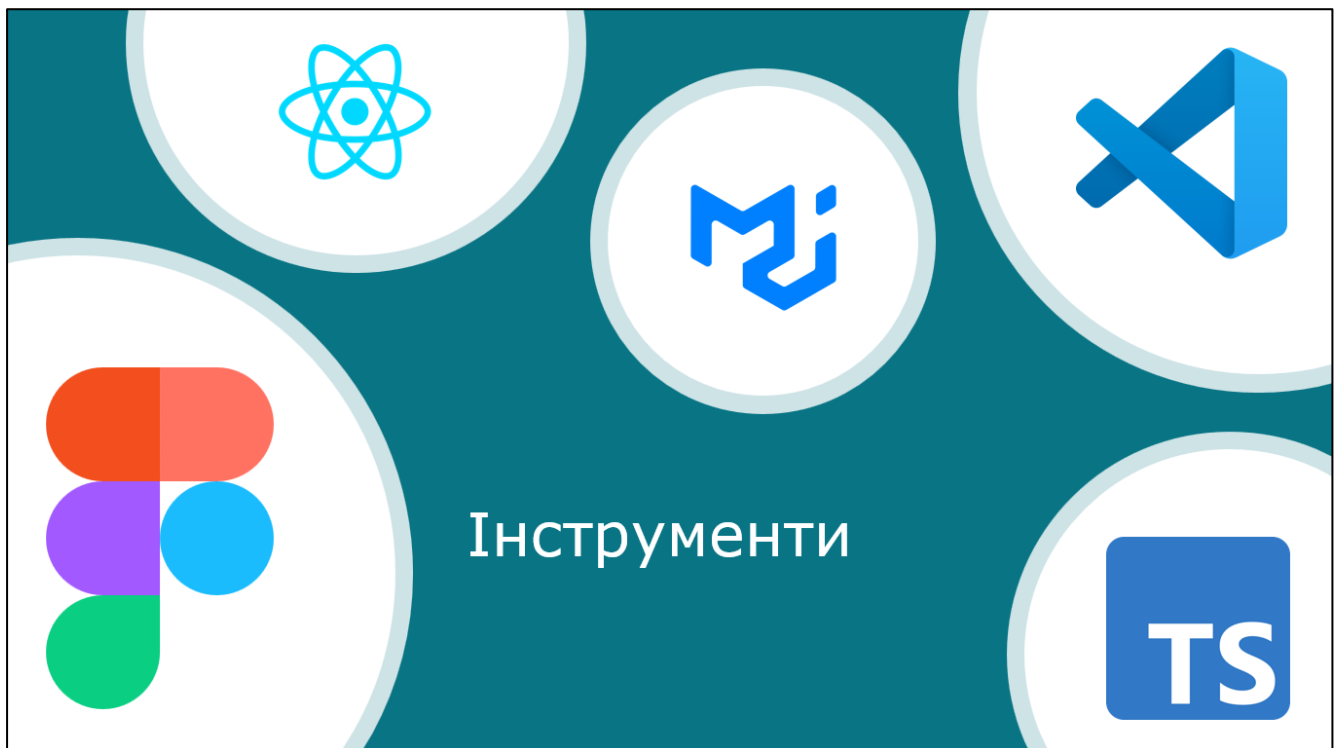


Рисунок Б.5 – Слайд №5

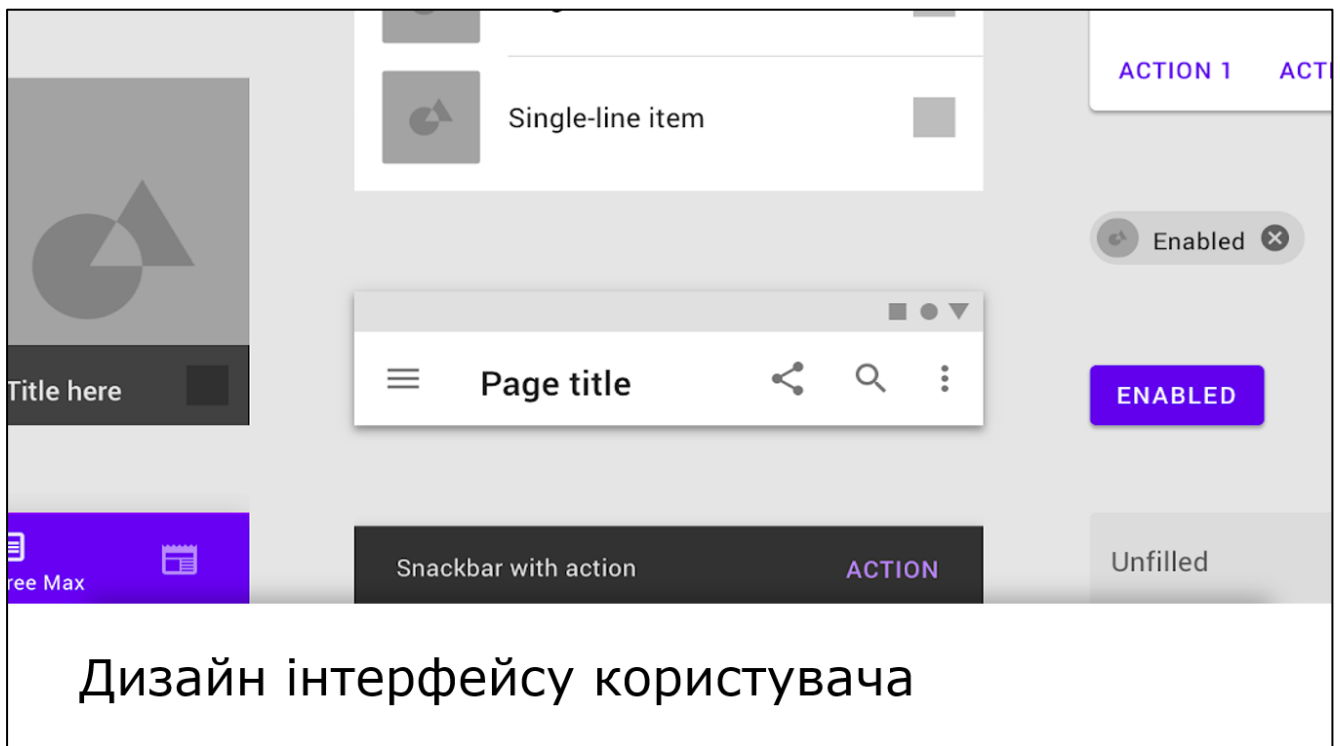
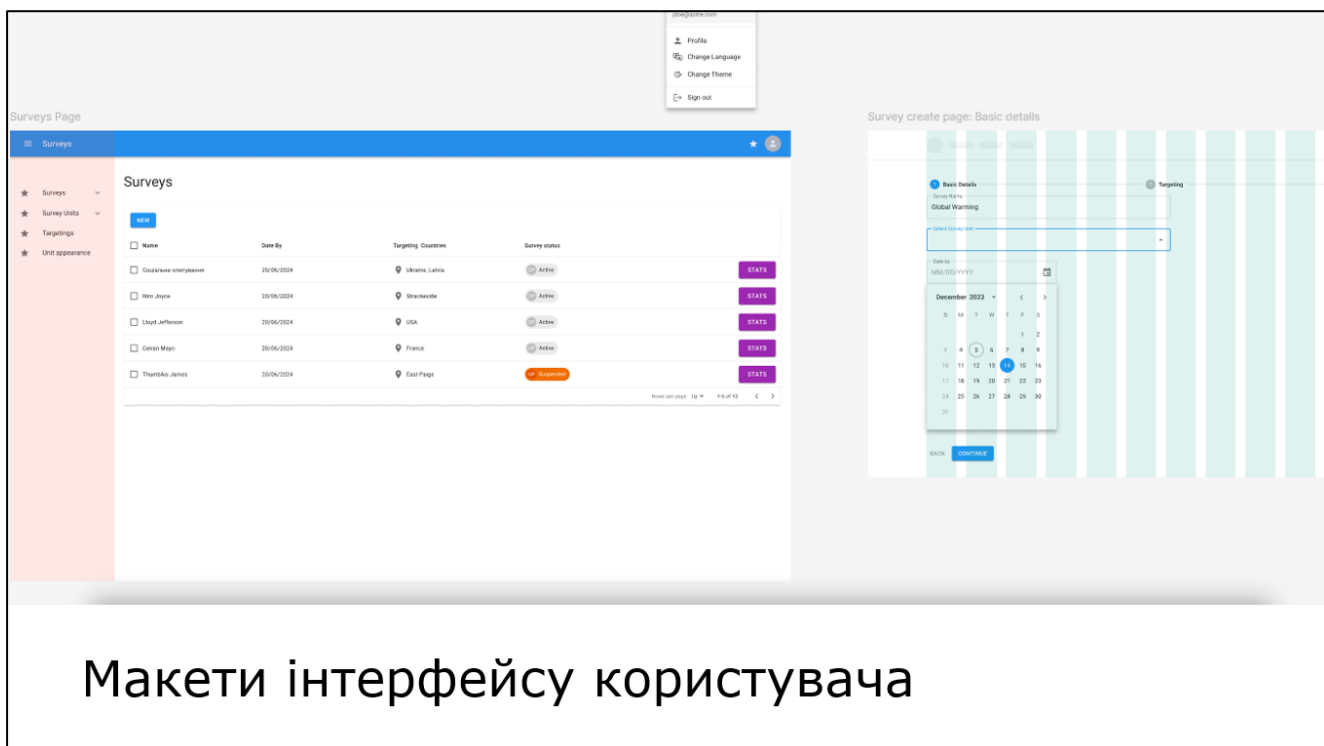
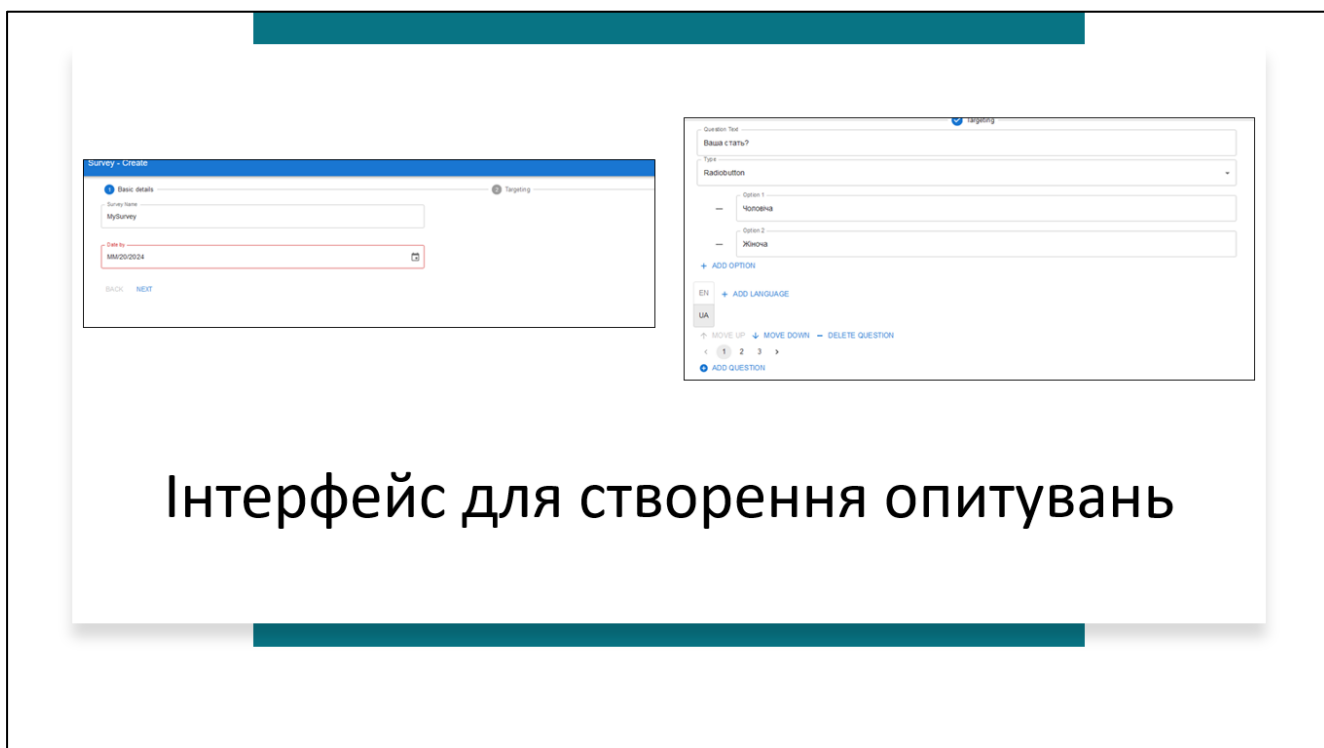


Рисунок Б.6 – Слайд №6



Макети інтерфейсу користувача

Рисунок Б.7 – Слайд №7



Інтерфейс для створення опитувань

Рисунок Б.8 – Слайд №8

Інтерфейс для перегляду даних

The screenshot shows a web application interface for managing survey units. At the top, there is a blue header bar with a menu icon, the text "Survey Units", and a user profile icon. Below the header, the main content area is titled "Survey Units" and contains a "NEW" button. The interface displays two data tables. The first table, "Survey Units", has columns for Unit Name, Appearance, Survey retake count, Max Surveys, and Actions. It contains one row with the following data: Unit Name: MyUnit, Appearance: MyAppearance, Survey retake count: 31, Max Surveys: 24, and a red "DELETE" button. The second table, "Surveys", has columns for Name, Targeting, Status, and Actions. It contains one row with the following data: Name: MySurvey, Targeting: MyTargeting (with a location pin icon), Status: Active (in a green circle), and a purple "STATS" button. At the bottom right of the tables, there is a pagination control showing "Rows per page: 5" and "1-1 of 1".

Рисунок Б.9 – Слайд №9

The screenshot shows a configuration interface for a survey template. It features several input fields: "Appearance Name" with the value "MyAppearance", "Template" with a dropdown menu set to "DefaultTemplate", "Color15" with a red color swatch and the value "#ff1216", and "Stroke15" with the value "20" and a "px" unit. Below these fields are "BACK" and "SAVE" buttons. A preview window titled "Survey Template" shows a question: "What is the highest mountain on the Earth?" with four radio button options: "Mount Everest", "Makalu", "Manaslu", and "Broad Peak". At the bottom of the preview window are "Prev" and "Next" buttons.

Інтерфейс для
налаштування
зовнішнього
вигляду


Рисунок Б.10 – Слайд №10

Тестування

| Назва тесту | Перегляд статистики по гендерам |
|-----------------------------|--|
| Опис тесту | Перевірка відображення статистики по гендерам по всіх опитуванням користувача. |
| Компонент системи | Компонент перегляду статистики |
| Кроки відтворення | 1. Авторизуватися як зареєстрований користувач. 2. Перейти на сторінку зі списком опитувань і обрати потрібне опитування. 3. Перейти на сторінку статистики опитування, натиснувши на кнопку "STATS". 4. Обрати потрібне питання зі списку наявних в опитування і групування по гендеру. |
| Очікуваний результат | Система відображає статистику по гендерам по всіх опитуванням користувача, включаючи кількість переглядів, кількість відповідей тощо. |
| Фактичний результат | Отримані коректні значення для статистики, але для перекладів варіантів відповідей – відображались зі зламаним кодуванням |
| Статус | Тест-кейс відтворений, баг виправлено |
| Коментар | Було виправлено кодування для бази даних програмної системи для потрібних полів, оскільки вони підтримували лише латинські символи кодування. |

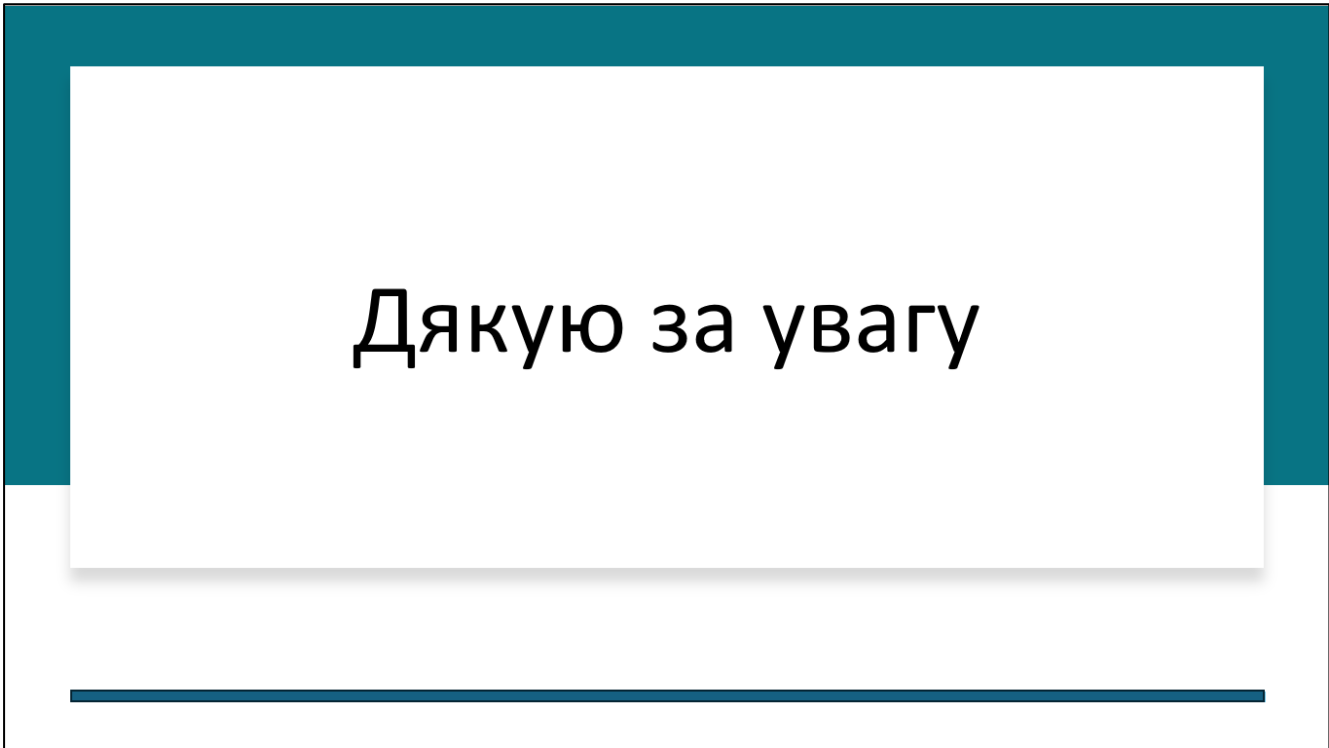
Рисунок Б.11 – Слайд №11

Висновки



- Розроблено клієнтську частину програмної системи що надає користувачам зручний інтерфейс для онлайн-опитувань.
- Спроектвано архітектуру клієнтської частини з використанням фреймворку ReactJS та мови програмування TypeScript
- Реалізовано основні функції системи, такі як авторизація, створення опитувань, їх груп, налаштування зовнішнього вигляду, створення параметрів таргетингу, перегляд статистики.

Рисунок Б.12 – Слайд №12



Дякую за увагу

Рисунок Б.13 – Слайд №13

ДОДАТОК В.
Специфікація програмного продукту

СПЕЦИФІКАЦІЯ ПЗ

До системи для проведення соціальних опитувань

Виконали:

Поронько Кирил Андрійович

Печій Ростислав Романович

Масенко Ярослав Володимирович

1. Вступ

1.1 Огляд продукту

Ціль цього документа полягає у створенні програмної системи для проведення опитувань «Pure Survey», яка дозволяє користувачам створювати різноманітні опитування, обираючи певні налаштування, з можливістю вбудовувати опитування на інші вебсайти, та переглядати статистику відповідей на опитування.

1.2 Межі

«Pure Survey» це програмна система, яка дозволяє користувачам створювати свої власні соціальні опитування, використовуючи гнучкі інструменти редагування.

Зареєстровані користувачі можуть оглянути та використовувати функціонал зі створення опитувань, змінювати відповідну аудиторію, на яку націлене опитування та їх редагувати.

Користувачі можуть бачити інформацію про створені опитування, групи в яких вони містяться, а з активною підпискою – також переглядати статистику обрану користувачем з певними відібраними даними по певних опитуваннях. Також експортувати створені опитування у вигляді скрипту для їх використання на різних веб-сайтах.

Додатково користувач може додати певні кошти на свій особистий рахунок профілю для публікації опитувань в мережі «Інтернет», відповідно використовуючи кошти за певну кількість отриманих відгуків.

1.3 Означення та аббревіатури

Таблиця 1 – Означення

| Термін | Означення |
|---------------------------|--|
| Таргетинг | Цільова аудиторія для якої буде доступне опитування. Залежить від країни користувача |
| Зареєстрований користувач | Користувач, який створив профіль у системі та може використовувати функції створення опитувань |
| Опитування | Елемент опитування створений юзером, залежний від налаштувань таргетингу. Опитування можна вбудовувати в інші вебсайти |

| | |
|------------------------------------|--|
| Група опитувань | Декілька однаково налаштованих опитувань в контексті зовнішнього вигляду та налаштувань правил опитування, які відрізняються таргетингом та вмістом питань |
| Front-end/Клієнт | Графічний інтерфейс, який відображається користувачеві за допомогою HTML, CSS і JavaScript, щоб користувачі могли переглядати дані та взаємодіяти з ними |
| Серверна частина/АПІ | Інтерфейс взаємодії між клієнтом і сервером, який також взаємодіє з базою даних та відповідає за базову частину програмної системи. |
| Високонавантажена серверна частина | Інтерфейс взаємодії між клієнтом і сервером, який також взаємодіє з базою даних та відповідає за опрацювання взаємодії з опитуваннями. |
| Темплейт | Зовнішній вигляд групи опитувань, задається у форматі html-коду. |

2. Загальний опис

2.1 Перспективи продукту

Програмна система складається з таких частин: веб-додаток користувача, веб-сервер для додатку, основний сервер бази даних для додатку, додаткова високонавантажена серверна частина і база даних для додаткової системи

Основна база даних міститься в СКБД MSSQL, яка містить в собі дані користувачів та опитувань. Користувач не може напряму комунікувати з сервером бази даних, але може використовувати основне АРІ системи.

При потребі даних клієнт формує запит і посилає його на АРІ веб-серверу. Даний веб-сервер інтерпретує запит і повертає помилку, якщо запит не може бути виконаний, інакше він формує та надсилає запит на сервер бази даних, або на інший додатковий веб-сервер відповідно до запиту користувача. База даних повертає помилку або дані, після чого веб-сервер перерозподіляє цю інформацію необхідним чином і надсилає клієнтові. Тільки клієнт відповідає за надання даних у графічному представленні.

Веб-сервер може додавати або змінювати дані під час обміну інформації. АРІ має бути прихована від користувачів, але доступна для інших частин системи і за її межами.

Додатковий веб-сервер використовує відповідно створену для нього базу даних і відповідає за внесення згенерованих подій під час проходження опитування і їх зберігання у додаткову базу даних, яка міститься СКБД Vertica. Також ця частина системи відповідальна за генерацію скриптів для вбудовування на інші веб-сайти.

2.2 Функції продукту

За допомогою клієнта зареєстровані користувачі зможуть створювати свої групи опитувань, налаштовуючи їхній зовнішній вигляд, правила проходження опитування, та створення одиниць опитування в групах, для яких будуть доступні налаштування контенту опитування та таргетингу. Опитування можна влаштувати в інші вебсайти, де й будуть проходити більшість проходжень опитувань. Також буде реалізовано функціонал оплати підписки задля можливості активації функцій системи, таких як вбудовування опитувань на інші веб-сайти.

Адміністратори зможуть створювати та змінювати темплейти для груп опитувань та налаштовувати систему.

2.3 Характеристики користувачів

У системі існує 3 типи користувачів: зареєстровані користувачі з підпискою і без неї та адміністратори.

Зареєстрованим користувачам без активної підписки доступний такий функціонал системи як: створення опитувань, груп опитувань, налаштування опитувань та зовнішній вигляд.

Зареєстрованим користувачам з активною підпискою доступний весь функціонал системи, таких як: створення опитувань та отримання скриптів для вбудовування в інші вебсайти, перегляд їх статистики.

Незареєстрованим користувачам доступні лише функції логіна та реєстрації.

Адміністратори зможуть оперувати темплейтами які будуть доступні в системі.

2.4 Загальні обмеження

Веб-програма клієнта обмежена веб-браузером, встановленим на пристрої користувача. Оскільки існує кілька веб-браузерів і різні пристрої з різною

роздільною здатністю екрана, інтерфейс, швидше за все, не буде однаковим для кожного з них.

Підключення до Інтернету також є обмеженням для програми. Програма підключається до веб-сервера через Інтернет, і немає способу отримати інформацію без зв'язку між клієнтом і сервером.

Сервер обмежений характеристиками системи баз даних і сервера баз даних. Коли є багато запитів, база даних може бути змушена поставити їх у чергу, що збільшує час, необхідний для отримання даних.

2.5 Припущення й залежності

Основне припущення полягає в тому, що більшість сучасних браузерів працюють однаково та підтримують більш-менш новітні технології, такі як HTML5. Якщо браузер не підтримує нові технології, інтерпретація коду та виконання інструкцій можуть призвести до помилок.

3. Конкретні вимоги

3.1 Вимоги до зовнішніх інтерфейсів

3.1.1 Інтерфейс користувача

Зареєстровані користувачі можуть створювати опитування та оплачувати підписку, після чого отримувати скрипти для вбудовування в інші вебсайти, переглядати їх статистику.

3.1.2 Апаратний інтерфейс

Для відображення інтерфейсу користувача до пристрою має бути підключений дисплей.

3.1.3 Програмний інтерфейс

Щоб мати доступ до веб-системи, користувач повинен використовувати програмне забезпечення для веб-перегляду. Він повинен підтримувати Javascript, HTML5, CSS3.

3.1.4 Комунікаційний протокол

Щоб користувач мав доступ до веб-системи, необхідно встановити підключення до Інтернету.

3.2 Функції продукту

3.2.1 Реєстрація

3.2.1.1 Опис

Неzareєстровані користувачі (відвідувачі) не мають дозволу на вхід до основного контенту системи, тому що це може призвести до хакерських атак. Щоб розпочати процес реєстрації, користувач повинен натиснути «створити обліковий запис». Це перемістить його до «реєстраційної форми».

3.2.1.2 Вхідні дані

Текст: Електронна пошта, пароль (8-50 символів, з перевіркою складності), ім'я(2-50 символів) прізвище (2-50 символів).

3.2.1.3 Обробка

При натисканні кнопки реєстрації клієнт надсилає запит із усіма вхідними даними на веб-сервер, який спілкується з API.

Веб-сервер перевіряє правильність логіна, пароля та електронної пошти. Якщо правильно – запитує сервер бази даних знайти користувачів із таким логіном або електронною поштою. Якщо знайдено – користувач залишається незареєстрованим і клієнт показує повідомлення з описаною причиною. В іншому випадку пароль буде хешовано, а всі дані, включаючи хеш перевірки, зберезуться у базі даних.

3.2.1.4 Вихідні дані

Якщо форма була заповнена успішно, після відправки клієнт видає повідомлення про те, що акаунт було успішно створено та перейде на сторінку логіна.

Якщо форма була заповнена невдало, перед формою з'являється повідомлення з відповідним текстом.

3.2.1.5 Обробка помилок

Повідомлення з помилкою з'явиться перед формою і не буде надіслано, якщо:

- неправильна електронна адреса;
- підтвердження електронною поштою не можна надіслати на адресу електронної пошти;
- користувач із цією електронною адресою вже зареєстрований;
- немає зв'язку з базою даних;
- невалідний пароль.

3.2.1.6 Обмеження

Неавторизовані/неzareєстровані користувачі.

3.2.2 Авторизація

3.2.2.1 Вступ

Неавторизовані користувачі (відвідувачі) можуть увійти, якщо вони мають зареєстрований профіль у системі. Це принесе їм привілеї. Щоб увійти, користувач повинен натиснути «увійти» в заголовку сторінки (в меню). Це перемістить його до «форми входу».

3.2.2.2 Вхідні дані

Текст: email (до 50 символів), пароль (8-50 символів).

3.2.2.3 Обробка

Веб-сервер хешує пароль і запитує базу даних для пошуку користувача з такою електронною поштою. Якщо знайдено та підтверджено користувача, у випадку якщо хешований пароль збігається – вхід виконано успішно. Це переносить дані користувача, що зберігаються в базі даних, до сеансу на сторони клієнта.

3.2.2.4 Вихідні дані

Якщо вхід виконано успішно, система вітає авторизованого користувача.

3.2.2.5 Обробка помилок

Якщо вхід не вдається, перед формою входу з'являється повідомлення з відповідним текстом.

3.2.2.6 Обмеження

Неавторизовані/незареєстровані користувачі.

3.2.2 Створення групи опитувань

3.2.2.1 Вступ

Користувачі зможуть створювати групу опитувань, яка є базою для кожного опитування, тобто для групи вказуються певні правила проходжень опитувань в цій групі, їх зовнішній вигляд, а саме: колір, розмір та шрифт. Після створення, користувачі можуть експортувати скрипт для вбудування опитування в інші вебсайти

3.2.2.2 Вхідні дані

Користувач заповнює форму створення групи опитувань, а саме такі налаштування як:

- дозвіл на повідомлення після завершення опитування;
- дозвіл на зникнення форми після проходження;

- кількість проходжень опитувань з одного пристрою за день;
- кількість проходження одного опитування одним пристроєм.

Також користувач обирає тип відображення, шаблон, задає певні параметри налаштування

3.2.2.3 Обробка

Після налаштування відображення опитування та правил проходження, юзер підтверджує форму. Після підтвердження форми клієнт відправляє запит на сервер зі створення групи опитувань та повертає статус створення та оновлює список груп опитувань з новоствореною групою.

3.2.2.4 Вихідні дані

Якщо якісь параметри було введено неправильно клієнт видає повідомлення про помилку. Якщо усе пройшло успішно – клієнт відображає нову групу опитувань у списку.

3.2.2.5 Обробка помилок

Якщо якісь параметри було введено неправильно клієнт видає помилку про те, що саме неправильно введено. Якщо було правильно введено усі дані, але була помилка на стороні сервера, то про це видається відповідна помилка. Якщо усе пройшло успішно – клієнт видає повідомлення про успішне створення та відображає нову групу опитувань у списку.

3.2.2.6 Обмеження

Доступно лише для зареєстрованих користувачів/

3.2.3 Перегляд набору опитувань

3.2.3.1 Вступ

Для зареєстрованого користувача доступна функція перегляду усіх груп опитувань, для цього на відповідній сторінці буде відображено список груп опитувань, які було створено юзером. Також користувач зможе переглянути вміст кожної групи опитувань натиснувши на певну групу та переглянути її дані: усі опитування групи, тип відображення, правила проходження тощо.

3.2.3.2 Вхідні дані

Для перегляду усіх груп опитувань користувач повинен перейти до відповідної сторінки усіх груп, щоб перейти до інформації певної групи опитувань потрібно натиснути на потрібний елемент групи опитувань, що направить користувача на сторінку інформації про конкретну групу опитувань.

3.2.3.3 Обробка

Після переходу на сторінку усіх груп опитувань, клієнт відправляє запит на отримання усіх груп опитувань поточного юзера, які відображає після отримання відповіді. Після натискання на конкретну групу опитувань, клієнт відправляє запит до сервера про отримання інформації про задану групу опитувань та відображає дані.

3.2.3.4 Вихідні дані

Усі дані про групи опитувань користувач бачить на екрані гаджета, а саме таку інформацію як: назва групи, тип відображення, кількість опитувань в ній.

Усі дані про конкретну групу опитувань користувач бачить на екрані гаджета, а саме таку інформацію як:

- правила проходження (дозвіл на повідомлення після завершення опитування, дозвіл на зникнення форми після проходження, кількість проходжень опитувань з одного пристрою за день, кількість проходження одного опитування одним пристроєм)
- тип відображення (колір, шрифт, розмір, місце розташування, шаблон)
- список опитувань у групі (назва та базова інформація про опитування)

3.2.3.5 Обробка помилок

Якщо в групі немає опитувань – повідомляємо користувача про це

3.2.3.6 Обмеження

Авторизовані користувачі.

3.2.4 Видалення набору опитувань

3.2.4.1 Вступ

Користувачу доступна функція видалення групи опитувань, у випадку, коли вона стає йому непотрібна.

3.2.4.2 Вхідні дані

Для видалення групи опитувань користувачу потрібно перейти до сторінки конкретної групи опитувань та натиснути на кнопку видалення, потім виконати підтвердження, натиснувши на відповідну кнопку.

3.2.4.3 Обробка

Після натискання користувачем на кнопку підтвердження видалення, клієнт виконує запит на видалення групи опитувань на сервер, після чого отримує результат запиту.

3.2.4.4 Вихідні дані

Кнопка видалення буде знаходитися на екрані користувача на сторінці конкретної групи опитувань. Повідомлення про успішне видалення чи помилку буде відображено модальним вікном на екрані користувача.

3.2.4.5 Обробка помилок

У випадку успішного видалення буде відображено відповідне повідомлення, у випадку помилки буде відображена причина помилки.

3.2.4.6 Обмеження

Авторизовані користувачі. Створена група опитувань для видалення

3.2.5 Редагування опитувань

3.2.5.1 Вступ

Користувачу доступна функція редагування опитування. Він може змінювати контент питання, їх переклад, варіанти відповіді та налаштування тагетингу.

3.2.5.2 Вхідні дані

Для редагування опитування, користувач має перейти на сторінку опитування та натиснути на кнопку редагування. Після цього змінити дані про таргетинг та питання і варіанти відповіді.

3.2.5.3 Обробка

Після відправки користувачем форми зі зміненими даними опитування, клієнт відправляє запит на сервер з оновлення даних про опитування, результат операції повертається на клієнт та залежно від результату відображається у користувача на екрані.

3.2.5.4 Вихідні дані

Якщо результат редагування успішний – система каже про успішну операцію користувачу

3.2.5.5 Обробка помилок

У випадку помилки при редагуванні, клієнт видасть помилку про це, відповідно до типу помилки, яку передає сервер.

3.2.5.6 Обмеження

Авторизований користувач, створене опитування для редагування.

3.2.6 Перегляд опитувань

3.2.6.1 Вступ

Користувач має можливість продивитися інформацію про опитування в групі, усі створені опитування та дані про конкретне опитування. У списку він має бачити таку інформацію, як: назва, кількість питань тощо. У деталізованих даних

про опитування, користувач має змогу передивитися список питань, їх переклад, таргетинг та дані про групу опитувань, в якій він знаходиться з даними про групу, таку як тип відображення, налаштування тощо.

3.2.6.2 Вхідні дані

Для перегляду списку опитувань, користувачу потрібно перейти на сторінку певної групи опитувань, щоб подивитися список опитувань у групі або на сторінку усіх опитувань, щоб переглянути усі створені. Щоб переглянути дані про конкретне опитування треба натиснути на певне опитування в списку, після чого користувач перейде на сторінку опитування.

3.2.6.3 Обробка

При переході користувача на сторінку усіх опитувань або опитувань у групі, клієнт відправляє відповідний запит на сервер, та, після отримання відповіді – відображає результат на сторінці. При переході до сторінки конкретного опитування, клієнт також посилає запит на АПІ, та відображає дані після отримання результату.

3.2.6.4 Вихідні дані

Усі дані про список опитувань чи дані про конкретне опитування будуть відображені на екрані користувача одразу по завантаженню відповідної сторінки.

3.2.6.5 Обробка помилок

У випадку помилки під час запиту на АПІ, на екрані користувача буде відображено відповідне повідомлення. Якщо у користувача немає опитувань, він не зможе перейти до сторінки конкретного опитування, а на сторінці списку опитувань він побачить повідомлення про відсутність опитувань.

3.2.6.6 Обмеження

Авторизований користувач.

3.2.7 Створення опитувань

3.2.7.1 Вступ

Користувачу доступна функція додавання опитування. Він може додати питання, їх переклад, варіанти відповіді, переклади, та налаштування тагетингу.

3.2.7.2 Вхідні дані

Для додавання опитування, користувач має перейти на сторінку усіх опитувань групи, в яку він хоче додати опитування та натиснути на кнопку «Додати». Після цього обрати дані про таргетинг, додати питання, варіанти відповіді, їх переклад.

3.2.7.3 Обробка

Після відправки користувачем форми з даними опитування, клієнт відправляє запит на сервер з додавання нових даних про опитування, результат операції повертається на клієнт та залежно від результату відображається у користувача на екрані.

3.2.7.4 Вихідні дані

Якщо результат додавання успішний – система каже про успішну операцію користувачу та у списку одразу з’явиться нове опитування.

3.2.7.5 Обробка помилок

У випадку помилки при редагуванні, клієнт видасть помилку про це, відповідно до типу помилки, яку передає сервер.

3.2.7.6 Обмеження

Авторизований користувач, доступний безкоштовний слот для створення опитування або куплена підписка.

3.2.8 Видалення опитувань

3.2.8.1 Вступ

Користувачу доступна функція видалення опитування, у випадку, коли воно стає йому непотрібне.

3.2.8.2 Вхідні дані

Для видалення опитування користувачу потрібно перейти до сторінки конкретного опитування та натиснути на кнопку видалення, потім виконати підтвердження, натиснувши на відповідну кнопку.

3.2.8.3 Обробка

Після натискання користувачем на кнопку підтвердження видалення, клієнт виконує запит на видалення опитування на сервер, після чого отримує результат запиту.

3.2.8.4 Вихідні дані

Кнопка видалення буде знаходитися на екрані користувача на сторінці конкретного опитування. Повідомлення про успішне видалення чи помилку буде відображено модальним вікном на екрані користувача.

3.2.8.5 Обробка помилок

У випадку успішного видалення буде відображено відповідне повідомлення, у випадку помилки буде відображена причина помилки.

3.2.8.6 Обмеження

Авторизовані користувачі. Створене опитування для видалення.

3.2.9 Додавання запитань до опитування

3.2.9.1 Вступ

Користувачу доступна функція додавання питання в опитування. Він може створити нове питання обравши варіанти відповіді та переклади для певних обраних мов.

3.2.9.2 Вхідні дані

Для додавання нового питання користувачу потрібно перейти до сторінки редагування певного опитування та натиснути на кнопку додавання нового питання. Після натискання на кнопку з'являється вікно створення нового питання, де користувач вписує текст питання та обирає відповідні варіанти відповідей та їх типи, а також написати текст питання на обраних мовах опитування.

3.2.9.3 Обробка

Після натискання кнопки збереження питання, клієнт виконує запит на додавання нового питання на сервер, після чого отримує результат запиту на додавання.

3.2.9.4 Вихідні дані

Кнопка додавання питання буде відображена на сторінці опитування. Після успішного додавання, користувач побачить повідомлення про це та також буде відображено щойно створене питання.

3.2.9.5 Обробка помилок

У випадку неуспішного додавання питання користувачеві буде відображена помилка з текстом причини результату запиту.

3.2.9.6 Обмеження

Авторизований користувач, створене опитування.

3.2.10 Видалення запитань з опитування

3.2.10.1 Вступ

Користувачу доступна функція видалення питання з опитування у разі його неактуальності.

3.2.10.2 Вхідні дані

Для видалення питання користувачу потрібно перейти до сторінки редагування певного опитування та натиснути на кнопку видалення питання. Після натискання на кнопку з'являється вікно підтвердження, де користувач підтверджує видалення.

3.2.10.3 Обробка

Після натискання кнопки підтвердження видалення, клієнт виконує запит на видалення питання у вигляді редагування опитування на сервер, після чого отримує результат запити.

3.2.10.4 Вихідні дані

Кнопка видалення буде відображена на сторінці редагування опитування біля кожного питання. Після видалення питання, користувач побачить повідомлення та питання зникне з опитування.

3.2.10.5 Обробка помилок

У разі помилки при запиті або неправильно введених даних користувач побачить відповідне повідомлення.

3.2.10.6 Обмеження

Авторизований користувач, створене опитування та питання в ньому.

3.2.11 Редагування запитань в опитуванні

3.2.11.1 Вступ

Користувачу доступна функція редагування питання в опитуваннях. Він може редагувати створене питання змінюючи варіанти відповіді та переклади для певних обраних мов.

3.2.11.2 Вхідні дані

Для редагування створеного питання користувачу потрібно перейти до сторінки редагування певного опитування та натиснути на редагування відповідного питання. Після натискання на кнопку з'являється вікно редагування питання, де користувач змінює текст питання та змінює, додає або видаляє відповідні варіанти відповідей і їх типи, а також змінити чи додати текст питання на обраних мовах опитування.

3.2.11.3 Обробка

Після натискання кнопки збереження питання, клієнт виконує запит на оновлення даниї певного питання на сервер, після чого отримує результат запити на зміну даних.

3.2.11.4 Вихідні дані

Кнопка редагування питання буде відображена на сторінці опитування біля певного питання. Після успішного редагування, користувач побачить повідомлення про це, і на сторінці опитування буде відображене щойно відредаговане питання .

3.2.11.5 Обробка помилок

У випадку неуспішного додавання питання користувачеві буде відображена помилка з текстом причини результату запиту

3.2.11.6 Обмеження

Авторизований користувач, створене опитування та питання в ньому.

3.2.12 Оплата підписки за користування послугами

3.2.12.1 Вступ

Щоб користуватися функціями активації опитувань на інших сайтах, користувачу потрібно купити підписку.

3.2.12.2 Вхідні дані

Щоб активувати підписку, користувачу потрібно натиснути на кнопку активації, відбувається редірект на вікно оплати PayPal, в якій користувач вводить свої дані для оплати та натискає на кнопку «сплатити».

3.2.12.3 Обробка

Після натискання кнопки оплати підписки користувача відбувається запит на створення підписки PayPal, після чого відбувається перехід на іншу сторінку з оплатою на PayPal систему і посилається запит на оплату підписки з відповідним станом оплати.

3.2.12.4 Вихідні дані

Після успішно проведеної оплати користувач може побачити стан активації підписки у своєму профілі з відповідною датою кінця статусу підписки.

3.2.12.5 Обробка помилок

У випадку неуспішної оплати, сторонній сервіс повідомляє користувача про результат оплати, при цьому на сайті буде відображена помилка про статус неуспішної оплати підписки.

3.2.12.6 Обмеження

Авторизований користувач, діюча картка PayPal.

3.2.13 Перегляд статистики по опитуванню

3.2.13.1 Вступ

Користувачу доступна функція перегляду статистики по обраному опитуванню. Він може переглядати та фільтрувати дані, отримані з відгуків клієнтів, використовуючи візуальні діаграми та засоби фільтрації.

3.2.13.2 Вхідні дані

Для перегляду статистики опитування користувачу потрібно перейти до сторінки статистики і натиснути на відповідну кнопку для перегляду статистики, після чого користувач буде переправлений на сторінку з відображеними діаграмами, які користувач може обрати, а також змінити поля та фільтрацію за допомогою відповідних засобів на сторінці.

3.2.13.3 Обробка

Після натискання кнопки на перегляд статистики, клієнт виконує запит на отримання даних по обраному питанню на високонавантажений сервер, після чого формуються відповідні дані для їх перегляду на сторінці.

3.2.13.4 Вихідні дані

Кнопка перегляду статистики опитування відображена на сторінці опитування біля основної інформації. Після успішного отримання даних, користувач побачить сторінку з відображеними даними у вигляді діаграм, і на цій ж сторінці користувач зможе відфільтрувати дані за допомогою наявних інструментів.

3.2.13.5 Обробка помилок

У випадку неуспішного отримання даних опитування користувачеві буде відображена помилка з текстом причини результату запиту

3.2.13.6 Обмеження

Авторизований користувач, опитування та відгуки на це опитування.

3.2.14 Перегляд таргетингів

3.2.14.1 Вступ

Користувач може переглядати усі створені таргетинги для опитувань двома способами: при створенні опитування, коли користувач обирає таргетинг, та окремо зайшовши на сторінку усіх таргетингів.

3.2.14.2 Вхідні дані

Перегляд таргетингів можливий після переходу на сторінку усіх таргетингів та під час створення опитування у вигляді форми.

3.2.14.3 Обробка

В обох випадках процес обробки однаковий, а саме: клієнт відправляє запит про усі створені таргетинги на сервер та, після отримання відповіді, відображає список таргетингів.

3.2.14.4 Вихідні дані

Список таргетингів буде відображено на екрані користувача під час створення опитування та після переходу на сторінку усіх таргетингів.

3.2.14.5 Обробка помилок

У випадку помилок клієнт повідомить користувача про те, що сталася помилка з текстом відповідним до типу помилки. Якщо не було створено жодних таргетингів, то про це буде повідомлено користувача.

3.2.14.6 Обмеження

Авторизований користувач

3.2.15 Створення таргетингів

3.2.15.1 Вступ

Для налаштування аудиторії, на яку буде націлено опитування, користувач зможе створювати таргетинги, обравши цільові країни.

3.2.15.2 Вхідні дані

Для створення таргетингу користувачу потрібно натиснути на відповідну кнопку та заповнити форму, де потрібно обрати цільові країни та придумати назву таргетингу.

3.2.15.3 Обробка

Під час підтвердження та відправки форми створення таргетингу, клієнт посилає запит на сервер, де створюється таргетинг, та відображає на клієнті результат залежно від відповіді.

3.2.15.4 Вихідні дані

У випадку успішного створення, про це буде повідомлено користувача.

3.2.15.5 Обробка помилок

Якщо під час створення було отримано помилку – вона відображається на екрані користувача.

3.2.15.6 Обмеження

Авторизований користувач.

3.3 Класи/Об'єкти

3.3.1 User

3.4.1.1 Атрибути

- Email – text
- Hashed password – text
- First name – text

- Last name – text
- Role – text

3.3.2 SurveyUnit

3.3.2.1 Атрибути

- Name – text
- One survey take per device – int
- Maximum surveys per device – int
- Message after no surveys – bool
- Hide after no surveys – bool

3.3.3 UnitAppearance

3.3.3.1 Атрибути

- Type – text
- Params – text
- State – text
- Name – text
- Template code – text
- Default params – text

3.3.4 Survey

3.3.4.1 Атрибути

- Name – text
- DateBy – text
- Params – text
- Countries – text
- QuestionType – text
- OrderNumber – int
- Translate – text

3.4 Нефункціональні вимоги

3.4.1 Продуктивність

Будь-яка сторінка повинна бути завантажена менш ніж за 5 секунд;

3.4.2 Надійність

Система не повинна виходити з ладу, коли користувач вводить неправильні параметри.

3.4.3 Доступність

Система повинна бути доступною для використання з будь-якого місця.

3.4.4 Безпека

- Конфіденційні дані повинні бути захищені;
- Паролі мають бути хешованими.

3.4.5 Придатність до обслуговування

Повинна бути панель адміністратора з реалізованими деякими функціональними вимогами.

3.4.6 Портативність

Сервіс має бути доступним для використання з мобільних браузерів.

3.5 Зворотні вимоги

Веб-сервіс не надає жодних додаткових конфіденційних даних користувача, окрім тих, які вводить користувач.

3.6 Обмеження дизайну

- всі користувацькі інтерфейси повинні бути перекладені українською мовою;
- колір # 2196F3 використовується для панелей (наприклад, головного меню);
- колір #FFFFFF для тла або тексту, якщо текст написаний на темних кольорах;
- колір #000000 для тексту, якщо він написаний на світлих кольорах;
- Pure Survey використовує шрифти: Impact для заголовків, Century Gothic для підзаголовків;
- дизайн має бути мінімалістичним.

3.7 Вимоги до логічної бази даних

Будемо використовувати MS SQL для основної функціональності системи,

оскільки важлива безпека та цілісність даних та СКБД Vertica для збереження даних статистики опитувань.