**Dokumentacja projektu Lotogeneria**

**Wprowadzenie**

Lotogeneria to strona internetowa, która pozwala użytkownikom na generowanie liczb do różnych loterii liczbowych dostępnych w Polsce. Użytkownik będzie mógł dostosować generowane liczby do swoich upodobań i zgodnie z podanymi wytycznymi. Strona będzie składała się z czterech podstron: Strona główna, Lotto, EuroJackpot i miniLotto.

**Technologie**

* Vite.js
* React.js
* TailwindCSS
* Material UI
* Styled Components

**Architektura projektu**

1. **Strona główna** (Home)
   * Krótki opis aplikacji
   * Przyciski nawigacji do podstron z loteriami
2. **Lotto**
   * Formularz konfiguracji liczb (ilość, zakres)
   * Generator liczb
   * Wyświetlenie wygenerowanych liczb
3. **EuroJackpot**
   * Formularz konfiguracji liczb (ilość, zakres)
   * Generator liczb
   * Wyświetlenie wygenerowanych liczb
4. **miniLotto**
   * Formularz konfiguracji liczb (ilość, zakres)
   * Generator liczb
   * Wyświetlenie wygenerowanych liczb

**Proces implementacji**

1. **Przygotowanie projektu**
   * Utwórz nowy projekt Vite.js z szablonem React
   * Zainstaluj TailwindCSS, Material UI, Styled Components i potrzebne zależności
   * Skonfiguruj Vite.js, aby obsługiwać TailwindCSS i Styled Components
2. **Struktura folderów**
   * Utwórz foldery dla komponentów, stron, stylów i kontekstu aplikacji
   * Przykładowa struktura folderów:

src/

components/

pages/

Home/

Lotto/

EuroJackpot/

miniLotto/

styles/

context/

1. **Tworzenie komponentów**
   * Stwórz komponent **Navbar**, który zawiera przyciski nawigacji do podstron z loteriami
   * Stwórz komponent **NumberGenerator**, który będzie generował liczby według konfiguracji
   * Stwórz komponent **NumberDisplay**, który będzie wyświetlał wygenerowane liczby
2. **Tworzenie stron**
   * Stwórz strony: **Home**, **Lotto**, **EuroJackpot** i **miniLotto**
   * Na każdej stronie z loterią dodaj formularz konfiguracji, generator liczb i wyświetlenie liczb
   * W formularzu konfiguracji umożliw użytkownikowi wybór ilości liczb oraz zakresu
   * Zaimplementuj logikę generatora liczb w komponencie **NumberGenerator**
   * Przekazuj wygenerowane liczby do komponentu **NumberDisplay**
3. **Nawigacja**
   * Utwórz Routing dla aplikacji, aby umożliwić przełączanie mi

* Utwórz Routing dla aplikacji, aby umożliwić przełączanie między stronami
* Zainstaluj **react-router-dom**
* Zaktualizuj komponent **App** dodając router i komponenty ścieżek (**Route**)
* Zdefiniuj ścieżki dla stron: **Home**, **Lotto**, **EuroJackpot** i **miniLotto**
* Zaktualizuj komponent **Navbar**, dodając linki nawigacyjne do odpowiednich stron (**Link** z **react-router-dom**)

1. **Stylowanie**
   * Utwórz plik konfiguracyjny TailwindCSS i dostosuj do potrzeb projektu
   * Stwórz globalne style dla aplikacji
   * Zastosuj style z TailwindCSS i Material UI do komponentów i stron
   * Użyj Styled Components do tworzenia stylów, gdy jest to potrzebne
2. **Stan aplikacji**
   * Jeśli potrzebujesz zarządzać stanem aplikacji, użyj kontekstu React lub Redux
   * Utwórz foldery i pliki niezbędne do zarządzania stanem
   * Przykład: używając kontekstu React, utwórz plik **LotteryContext.js** w folderze **context**
3. **Testowanie**
   * Przeprowadź testy jednostkowe i integracyjne, używając narzędzi takich jak **Jest** i **React Testing Library**
   * Utwórz foldery i pliki niezbędne do testów
   * Napisz testy sprawdzające poprawność działania generatora liczb oraz poprawność wyświetlania liczb na stronach
4. **Optymalizacja**
   * Przeprowadź analizę wydajności aplikacji, używając narzędzi takich jak Lighthouse
   * Zoptymalizuj ładowanie zasobów, kodu i czas wykonywania
5. **Wdrożenie**
   * Wybierz hosting i domenę dla aplikacji
   * Skonfiguruj środowisko produkcyjne, używając Vite.js
   * Wdroż aplikację na serwerze i przetestuj jej działanie

**Podsumowanie**

Aplikacja Lotogeneria pozwala użytkownikom na generowanie liczb do różnych loterii liczbowych dostępnych w Polsce, zgodnie z podanymi przez nich wytycznymi. Projekt został zbudowany z wykorzystaniem Vite.js, React.js, TailwindCSS, Material UI oraz Styled Components. Projekt obejmuje tworzenie komponentów, stron, nawigacji, stylowania, zarządzania stanem, testowania, optymalizacji oraz wdrożenia.