Obraz zawierający tekst, Czcionka, logo, design

Opis wygenerowany automatycznie

**Paweł Dziedzic**

Aplikacja internetowa do współdzielonego rysowania

Opiekun pracy:  
dr inż. prof. PRz Mariusz Borkowski

Rzeszów, 2024

[1. Wstęp 3](#_Toc167969959)

[2. Przygotowanie projektu 3](#_Toc167969960)

[3. Opis projektu 4](#_Toc167969961)

[3.1. Strona główna 4](#_Toc167969962)

[3.1.1. Okno „Create Room” 4](#_Toc167969963)

[3.1.2. Okno „Join Room” 5](#_Toc167969964)

[3.2. Tablica interaktywna 7](#_Toc167969965)

[3.2.1. Logika tablicy 7](#_Toc167969966)

[3.2.2. Logika Rysowania 8](#_Toc167969967)

[3.2.3. Logika przybliżenia i oddalania 10](#_Toc167969968)

[3.2.4. Logika cofania (ctrl+z) i przewijania do przodu (ctrl+y) 11](#_Toc167969969)

[3.2.5. Logika wczytania pokoju 12](#_Toc167969970)

[3.2.6. Logika zmiany koloru i grubości narzędzi. 12](#_Toc167969971)

[4. Podsumowanie 13](#_Toc167969972)

1. Wstęp

Celem niniejszego projektu było stworzenie zaawansowanej aplikacji internetowej do współdzielonego rysowania, wykorzystując nowoczesne technologie webowe, takie jak React. Aplikacja powstawała za pomocą oprogramowania Yarn stosowanego jako narzędzie do menadżera plików i zarządzania pakietami JavaScript. W projekcie wykorzystano również następujące biblioteki:

* Konva - biblioteka JavaScript stworzona do obsługi i manipulacji grafiką 2D w przeglądarce internetowej. Użyto jej głównie do utworzenia tablicy rysującej jak i samej logiki rysowania.
* Socket.io - biblioteka JavaScript, która umożliwia łatwą implementację komunikacji w czasie rzeczywistym między klientem a serwerem za pomocą WebSocketów.
* Router - biblioteka do obsługi routingu w aplikacjach React. Umożliwia tworzenie wielostronicowych aplikacji bez konieczności przeładowania strony, zarządzając nawigacją oraz renderowaniem odpowiednich komponentów w zależności od adresu URL.

1. Przygotowanie projektu

Na samym początku zainstalowano oprogramowanie yarn. Następnie utworzono projekt React z domyślnymi ustawieniami za pomocą oprogramowanie Vite, które pozwala na tworzenie szablonów aplikacji i uruchomienie lokalnego serwera. Poprzez środowisko Yarn zostały zaimplementowane do projektu wybrane biblioteki. Do uruchomienia utworzonego serwera użyto oprogramowania Node. Poniżej znajduje się sposób uruchomienia aplikacji.

* Uruchomienie serwera aplikacji



* Uruchomienie aplikacji



1. Opis projektu
   1. Strona główna

Na pierwszej stronie znajdują się dwa okna formularza. „Create Room” pozwala na utworzenie nowego pokoju do rysowania na podstawie generowanego kodu uuid. „Join room” pozwala na wejście do wybranego pokoju jeśli użytkownik posiada wybrany kod.

Obraz zawierający tekst, zrzut ekranu, Czcionka, diagram

Opis wygenerowany automatycznie

* + 1. Okno „Create Room”

W oknie „Create Room” użytkownik może wygenerować kod za pomocą przycisku „generate” który uruchamia wybraną funkcje:

Obraz zawierający zrzut ekranu, tekst, Czcionka

Opis wygenerowany automatycznie

Po wygnerowaniu kodu i wpisaniu nazwy użytkownika i wciśnięciu przycisku „create room” następuje przekierowanie na stronę z możliwością rysowania.

Obraz zawierający tekst, zrzut ekranu, oprogramowanie, System operacyjny

Opis wygenerowany automatycznie

Powyższy kod uruchamia się właśnie przy wciśnięciu przycisku. Generuje on użytkownika i wysyła na serwer zmienną pod którą znajdują się informacje o nim.

* + 1. Okno „Join Room”

Proces dołączenia do pokoju działa w podobny sposób jak generowanie. Różni się on jedynie implementacją wyjątku odpowiedzialnego za wykrycie błędu w przypadku dołączenia nie istniejącego pokoju. Za każdym dołączeniem użytkownika na stronę główną, pobierana jest z serwera lista zapamiętująca unikalne identyfikatory pokoi.

Obraz zawierający tekst, zrzut ekranu, Czcionka, linia

Opis wygenerowany automatycznie

Poniższy proces znajduje się po stronie klienta i nasłuchuje powyższego sygnału serwera.

Obraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznie

Następnie porównany jest wpisany identyfikator w pole tekstowe z tym znajdującym się w zapisanej tablicy. W przypadku błędnej identyfikacji zwracany jest błąd.

Obraz zawierający tekst, zrzut ekranu, oprogramowanie, Czcionka

Opis wygenerowany automatycznie

Obraz zawierający tekst, zrzut ekranu, Czcionka, numer

Opis wygenerowany automatycznie

* 1. Tablica interaktywna

Utworzona została nieskończona tablica do rysowania, którą można przesuwać, oddalać lub przybliżać. Posiada ona podstawowe funkcje wyboru grubości pędzla i koloru, oraz wyboru grubości gumki.

Obraz zawierający diagram, tekst, Wykres, linia

Opis wygenerowany automatycznie

* + 1. Logika tablicy

Tablica renderowana jest za pomocą bibiloteki Konva. Do obietku „Stage” odpowiadającym za tablice przekazywane są metody takie jak pisanie lub przybliżanie i oddalanie.

Obraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznie

Element „Layer”, grupuje kształty i działa jak warstwa na płótnie. W tym obiekcie mapowane są punkty w celu renderowania linii na płótnie. Linie są reprezentowane przez komponent biblioteki Konva „Line”. Obiekt posiada atrybuty wskazujące na położenie punktów rysujących, kolor i grubość pędzla lub gumki.

Obraz zawierający tekst, zrzut ekranu, oprogramowanie, wyświetlacz

Opis wygenerowany automatycznie

Poniżej znajduje się obiekt „Circle”, którego celem jest zwizualizowanie obszaru wymazywania przez gumkę elementów, które zostały narysowane.

Obraz zawierający szkic, Grafika liniowa, rysowanie, clipart

Opis wygenerowany automatycznie

* + 1. Logika Rysowania

Przy wyborze wybranej opcji do rysowania zmieniany jest status zmiennej mode.

Obraz zawierający tekst, Czcionka, zrzut ekranu, linia

Opis wygenerowany automatycznie

W chwili gdy użytkownik wciśnie lewy przycisk myszy sprawdzany jest typ elementu jaki wybrał. Następnie ustawiany jest stan „isDrawing” co na „true” co wskazuje że użytkownik zaczął rysowanie. Pobierana jest referencja do obiektu sceny, która jest przekazywana do funkcji pobierającej pozycję wskaźnika myszy względem tablicy. Na tej podstawie tworzona jest nowa linia i atrybutami takimi jak grubość ołówka lub gumki. Po tym aktualizowany jest stan tablicy zawierającej informacje na temat utworzonych linii.

Obraz zawierający tekst, zrzut ekranu, Czcionka, oprogramowanie

Opis wygenerowany automatycznie

Funkcja „getRelativePointerPosition” pozwala na zidentyfikowanie gdzie znajduje się kursor w trakcie wykonywania operacji rysowania.

Obraz zawierający tekst, zrzut ekranu, Czcionka, linia

Opis wygenerowany automatycznie

Poniższa funkcja jest wywoływana w trakcie ruchu myszą przy jednoczesnym trzymaniu lewego przycisku myszy. Pobierana jest pozycja kursora a następnie sprawdzany jest rodzaj narzędzia. W przypadku gumki ustawiana jest jej pozycja. W zmiennej „lastLine” zapisywana jest narysowana linia (czyli ta utworzona w funkcji „hadleMouseDown”). Do tej zmiennej będącej tablicą dodawane są nowe współrzędne. Ostatni element tablicy „drawingLines” jest zastępowany zaktualizowaną linią. Na koniec aktualizowana jest tablica „drawingLines” o ostatnią zaktualizowaną linię. Po wszystkim ostatnia narysowana linia jest wysyłana na serwer wraz z numerem identyfikacyjnym pokoju, aby zsynchronizować rysunek z innymi użytkownikami w tym samym pokoju.

Obraz zawierający tekst, zrzut ekranu, oprogramowanie, Czcionka

Opis wygenerowany automatycznie

Po opuszczeniu lewego przycisku myszy pobierana jest ostatnia linia i wysyłana na serwer. Następnie wyłączony zostaje tryb rysowania zmieniając status na false.

Obraz zawierający tekst, zrzut ekranu, Czcionka, linia

Opis wygenerowany automatycznie

* + 1. Logika przybliżenia i oddalania

Pierwsza linia poniższego kodu zapobiega przewijaniu strony w przeglądarce. Pobierana jest referencja do sceny, aktualna skala, pozycja kursora, współczynnik skali. Następnie obliczana jest nowa skala na podstawie kierunku przewijania. Obliczane jest względne położenie kursora względem sceny i aktualnej skali. Na tej podstawie ustawiana jest nowa skala dla sceny na obu osiach. Na koniec obliczana jest nowa pozycja sceny, która jest aktualizowana i następuje ponowne rysowanie elementów na tablicy.

Obraz zawierający tekst, zrzut ekranu

Opis wygenerowany automatycznie

* + 1. Logika cofania (ctrl+z) i przewijania do przodu (ctrl+y)

Gdy użytkownik postanowi skorzystać z cofnięcia lub przewinięcia zmian naniesionych na scenę, informacje o tym jest przesyłana na serwer wraz z numer identyfikacyjnym pokoju przypisanym pod użytkownika.

Obraz zawierający tekst, zrzut ekranu, Czcionka, oprogramowanie

Opis wygenerowany automatycznie

Po stronie serwera sprawdzana jest lista zawierająca informacje o narysowanych liniach danego pokoju. W razie gdy taki element w liście istnieje to jest on usuwany z od końca, a następnie dodawany do listy zapamiętującej cofniętą linię danego pokoju. Na koniec informacja jest rozsyłana do wszystkich użytkowników pokoju.

Obraz zawierający tekst, zrzut ekranu, oprogramowanie, Czcionka

Opis wygenerowany automatycznie

W przypadku przewijania do przodu sprawdzana jest lista elementów, które zostały cofnięte z listy „roomDrawings”. Ostatni element tej listy jest następnie usuwany z niej i przeniesiony na koniec listy początkowej z której to został na początku cofnięty. Na koniec informacja jest wysyłana do wszystkich użytkowników pokoju.

Obraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznie

* + 1. Logika wczytania pokoju

Gdy nowy użytkownik dołączy do stworzonego wcześniej pokoju to narysowane już wcześniej linie przez poprzednich użytkowników zostają odtworzone na tablicy nowego członka.

Obraz zawierający tekst, zrzut ekranu, Czcionka, linia

Opis wygenerowany automatycznie

* + 1. Logika zmiany koloru i grubości narzędzi.

Obraz zawierający tekst, zrzut ekranu, Czcionka, Jaskrawoniebieski

Opis wygenerowany automatycznie

Gdy użytkownik zmieni ustawienia dowolnego narzędzia to jego wartość jest aktualizowana przez poniższą funkcję.

Obraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznie

1. Podsumowanie

Projekt stworzenia w pełni funkcjonalnej aplikacji umożliwiającej jednoczesne pisanie w czasie rzeczywistym okazał się wyzwaniem. Kwestią pierwszorzędną w dalszym rozwoju aplikacji jest optymalizacja logiki pisania ze względu na wykorzystanie pamięci przez przechowywanie informacji na temat rysowania linii. Kolejną kwestią jest lepsze zabezpieczenie procesu dołączenia do pokoju i rozwiązanie wszelkich możliwych problemów z nimi występujących. Aktualnie po wyłączeniu serwera, wszystkie utworzone pokoje są usuwane z pamięci, dlatego kluczowym elementem jest zaimplementowanie bazy danych do projektu, która będzie przechowywać informacje na temat utworzonych pokoi.

Jak można zauważyć projekt wymaga sporo pracy nad nowymi funkcjonalnościami i naprawą widocznych problemów. Mimo to udało się utworzyć pewien zarys tego jak ta aplikacja będzie w przyszłości wyglądała. Ponadto podstawowe działanie aplikacji, czyli rysowanie w czasie rzeczywistym działa poprawnie i pozwala na podstawowe wykonywanie zamierzonych celów.