



Uniwersytet Rzeszowski
Kolegium Nauk Przyrodniczych
Instytut Informatyki

Praca projektowa Technologie Internetowe

NovoStudy

Prowadzący:

mgr inż. Wojciech Gałka

Autor:

Ewelina Kaniewska

nr albumu:

125129

Kierunek: Informatyka, grupa lab 2

Rzeszów 2024

1. Założenia i cele projektu

Projekt NovoStudy stanowi innowacyjną inicjatywę, która koncentruje się na stworzeniu aplikacji, mającej na celu ułatwienie procesu nauki poprzez wykorzystanie funkcjonalności takich jak lista zadań i timer oparty na technice pomodoro. Poniżej przedstawione są główne założenia oraz cele projektu:

- **Efektywność nauki**
NovoStudy dąży do zwiększenia efektywności procesu nauki poprzez skoncentrowanie uwagi użytkownika na bieżących zadaniach oraz wykorzystanie techniki pomodoro do optymalizacji czasu nauki.
- **Personalizacja**
Celem projektu jest dostarczenie użytkownikom narzędzi do tworzenia spersonalizowanej przestrzeni, co zwiększa ich zaangażowanie i motywację do nauki.
- **Intuicyjność obsługi**
Aplikacja została zaprojektowana z myślą o łatwości obsługi. Interfejs jest intuicyjny, a funkcjonalności, takie jak dodawanie zadań czy personalizacja tła, są dostępne w sposób przejrzysty.
- **Optymalizacja czasu**
Timer oparty na technice pomodoro pomaga użytkownikom efektywnie zarządzać czasem nauki, zapewniając okresy skupienia oraz krótkie przerwy, co sprzyja lepszej koncentracji.
- **Zachowanie postępów**
NovoStudy umożliwia śledzenie postępów w nauce, zarówno poprzez zaznaczanie ukończonych zadań, jak i przechowywanie zadań nowo dodanych.

Poprzez te założenia i cele, Projekt NovoStudy aspiruje do stworzenia kompleksowej, wszechstronnej aplikacji, wspierającej użytkowników w ich indywidualnych wysiłkach naukowych.

2. Działania podjęte w celu spełnienia założeń

- **Lista**
Jednym z kluczowych elementów aplikacji jest lista zadań. Projekt posiada klasę z metodami statycznymi odpowiedzialnymi za obsługę listy. Aplikacja umożliwia dodawanie zadań do listy, ukrywanie zadań wykonanych oraz ponowne ich pokazywanie. Do tworzenia zadań została wykorzystana metoda, która tworzy nowy element HTML i dodaje go do DOM, dzięki czemu można na nim wykonywać niezbędne operacje. Przy wpisywaniu tekstu nowego zadania pojawiał się problem w postaci stałego rozmiaru pola textarea. Aby rozwiązać ten problem zastosowano metodę, która dynamicznie zmienia rozmiar tego elementu w zależności od ilości zawartości. Dzięki temu zadania krótsze zajmują mniej miejsca na liście, a dłuższe odpowiednio więcej. Rozwiązanie to rozszerzono o zastosowanie metody, która za pomocą `window.addEventListener` sprawdza czy nie zmienia się rozmiar okna przeglądarki i dynamicznie reaguje na takie zmiany zmieniając wysokość pola z

tekstem zadania. Zaznaczanie zadań jako wykonane jest realizowane poprzez metodę ustawiającą styl przekreślenia dla tekstu w polu zadania. Takie zaznaczone zadania można ukryć po kliknięciu w przycisk Hide, który wywołuje metodę ustawiającą styl `display:none` dla zadań, które zostały zaznaczone jako wykonane. Kiedy opcja Hide jest włączona, kolejne zadania, które zostaną zaznaczone również będą ukrywane. Dzieje się tak dzięki zastosowaniu `eventListenera` dla każdego utworzonego checkboxa. Metoda Hide jest asynchroniczna i korzysta z instrukcji `await` w celu ustawienia 800ms czasu oczekiwania aż funkcja się wykona. Po naciśnięciu przycisku Show zaznaczone zadania ponownie pojawią się na liście, dzięki usunięciu stylu, który wcześniej je ukrywał. Dodane zadania przechowywane są w formacie JSON w local storage i są ładowane w obrębie jednej przeglądarki przy ponownym uruchomieniu aplikacji.

- Timer

Projekt posiada timer pozwalający na wprowadzanie czasu, który użytkownik chce poświęcić na daną sesję nauki. Po wprowadzeniu czasu i kliknięciu przycisku pola do wprowadzania staną się niemożliwe do edycji i rozpocznie się odliczanie. Odliczanie realizuje funkcja wywoływana cyklicznie co 1 sekundę, która po każdym wywołaniu aktualizuje stan licznika w przeglądarce. Funkcja uzupełnia zerami wartości mniejsze niż 10 w obu polach tak, aby zachować format czasu hh-mm. Po kliknięciu przycisku Pause funkcja zatrzymuje się, a do pół można ponownie wpisywać wartości. Przy pauzowaniu licznika pojawił się problem dodatkowych zer przy wartościach mniejszych od 10 po ponownym uruchomieniu np. wartość 09s była wyświetlana jako 009 i liczba zer zwiększała się z każdym naciśnięciem przycisków. Aby rozwiązać ten problem zastosowano konwersję z tekstu na typ liczbowy przed każdym odliczaniem, co spowodowało usunięcie zer na początku. Dopiero wtedy sprawdzany jest warunek dodający niezbędne zera. Wartości domyślne timera można ustawić w bocznym menu w zakładce Set default timer values. Po ustawieniu są one przechowywane w ciasteczkach przeglądarki i nie wygasają, dzięki temu po ponownym otwarciu przeglądarki ustawienia timera użytkownika są ładowane do aplikacji.

- Tło aplikacji

Na potrzeby zmiany tła została utworzona klasa Background zawierająca metody statyczne. Zmiana tła odbywa się poprzez wywołanie metody `SetBackground`, która wywołuje animację wygaszającą jedno tło i wyświetlającą inne tło utworzoną za pomocą metod `fadeOut` i `fadeIn`. Wygaszanie tła polega na stopniowym zmniejszaniu jego przezroczystości w zadanym czasie. Następnie jako nowe tło zostaje ustawione tło, do którego odnośnik został naciśnięty w menu i stopniowo zostaje zwiększana przezroczystość, żeby symulować gładkie przejście. Tło wybrane przez użytkownika jest zapisywane w local storage i jest ładowane po uruchomieniu aplikacji.

- Menu boczne

Do otwierania i zamykania menu służą dwie funkcje, które ustawiają w CSS styl menu odpowiadający za szerokość. Trochę inne podejście zostało zastosowane przy pokazywaniu i chowaniu zakładek w menu. Tam działa funkcja, która ustawia styl `display:none` i go usuwa. W zakładce z domyślnymi wartościami timera użytkownik może wpisać wartość jaka go interesuje i po kliknięciu przycisku wartości te są

zapisywane w ciasteczkach, gotowe do ponownego załadowania przy odświeżeniu strony. W drugiej zakładce po kliknięciu każdego z dostępnych przycisków wywołuje się metoda do zmiany tła dla tła, które jest przypisane do naciśniętego przycisku. Następnie tło zapisuje się w local storage i również jest gotowe do ponownego załadowania po odświeżeniu strony.

3. Testowanie

- **Użytkownik wprowadza niepoprawne dane**

Kiedy użytkownik próbuje wprowadzić inne dane niż liczby do pola formularza typu numerycznego w polu nie pojawia się żadna wartość. Aplikacja jest zabezpieczona przed tego typu przypadkami.

- **Niepoprawne korzystanie z przycisku**

Kiedy użytkownik próbuje nacisnąć przycisk rozpoczynający odliczanie w trakcie gdy odliczanie trwa, funkcja odpowiedzialna za timer się nie wykona.

-

4. Dokumentacja kodu źródłowego

<https://ziemto.github.io/TI-PROJECT/out/>

5. Źródła

<https://www.w3schools.com>

<https://developer.mozilla.org/en-US/>