

Wydział:

Kierunek:

Specjalność:

Praca dyplomowa inżynierska

**Projekt i Implementacja Autorskiego Frameworka Front-endowego z Wykorzystaniem JavaScript i JSX**

Promotor: Autor:

**dr inż. Damian Raczyński Filip Sieniawski**

Spis treści

[Wstęp 3](#_Toc165296827)

[Definicje 5](#_Toc165296828)

[Założenia technologiczne 6](#_Toc165296829)

[Funkcje 6](#_Toc165296830)

# Wstęp

W dobie dynamicznie rozwijającego się sektora aplikacji internetowych, rośnie zapotrzebowanie na udoskonalenie technologii i narzędzi pracy dla programistów. Jednym z kluczowych obszarów, który w ostatnich latach przeszedł znaczącą transformację, jest tworzenie interfejsów użytkownika. Frameworki front-endowe, takie jak React, Angular czy Vue.js, stały się nieodłącznym elementem procesu tworzenia nowoczesnych aplikacji webowych. Stanowią one jednak zajmujące wiele miejsca pakiety obciążające system developera.

Niniejsza praca inżynierska stawia sobie za cel zaprojektowanie i implementację autorskiego frameworka front-endowego opartego na językach JavaScript i JSX. Celem nadrzędnym jest stworzenie narzędzia, które umożliwi programistom tworzenie zaawansowanych interfejsów użytkownika w sposób efektywny i intuicyjny jednocześnie eliminując wielkoskalowe środowiska programistyczne pośredniczące między programistą a tworzeniem kodu. Framework ten ma stanowić odpowiedź na rosnące zapotrzebowanie na szybkie tworzenie skalowalnych aplikacji webowych, przy jednoczesnym zachowaniu pełnej kontroli nad kodem i komponentami interfejsu.

Praca inżynierska obejmuje:

1. **Omówienie teoretycznych założeń i technologii**: Dokładne przedstawienie fundamentów teoretycznych oraz wybranych technologii, które zostaną wykorzystane do skonstruowania frameworka.
2. **Projektowanie architektury frameworka**: Szczegółowy proces projektowania architektury, uwzględniający jego główne funkcje, cechy i komponenty.
3. **Implementacja frameworka**: Praktyczna realizacja, obejmująca stworzenie przykładowych komponentów oraz mechanizmów interakcji z nimi.
4. **Testy i ewaluacja**: Przeprowadzenie testów wydajnościowych i porównawczych frameworka, w celu weryfikacji jego efektywności i funkcjonalności w kontekście istniejących rozwiązań.
5. **Zastosowania i korzyści**: Prezentacja potencjalnych zastosowań autorskiego frameworka front-endowego w praktyce, uwzględniając korzyści płynące z jego wykorzystania.

Opracowywany framework ma na celu nie tylko udostępnienie nowego narzędzia programistom, ale również wniesienie istotnego wkładu w rozwój dziedziny tworzenia aplikacji webowych. Projekt ten stanowić będzie platformę do eksploracji nowatorskich podejść i rozwiązań w zakresie budowy interfejsów użytkownika jednocześnie pozostawiając kod źródłowy w pełni otwarty, pozwalając innym programistom lub zespołom developerskim na dostosowywanie narzędzia do własnych potrzeb według uznania.

# Porównanie podobnych rozwiązań

1. Porównanie rozmiarów dostępnych rozwiązań w wersji developerskiej (kb)

\*Wykres nie uwzględnia kompilatora Babel

* Angular
* React
* Vue
* Lit Google
* MiniF.js

Tabela porównawcza

# Wykorzystane technologie

Js

Jsx

transpilator - babel

# Porównanie stron

# Porównanie przeglądarek

j-meter

„jest”

cypress

# Definicje

**Document Object Model (DOM)** - to reprezentacja struktury dokumentu HTML lub XML w formie hierarchicznego drzewa obiektów, które można modyfikować za pomocą skryptów JavaScript lub innych języków programowania. DOM definiuje sposób, w jaki przeglądarka internetowa interpretuje strukturę dokumentu i umożliwia interakcję z jej elementami.

W DOM każdy element dokumentu, taki jak nagłówek, paragraf, obraz czy formularz, jest reprezentowany jako obiekt, a relacje między nimi są reprezentowane poprzez zagnieżdżenie tych obiektów w formie drzewa. Dzięki temu, manipulując obiektami DOM za pomocą skryptów, można zmieniać zawartość, styl i właściwości elementów dokumentu, co umożliwia dynamiczną interakcję i aktualizację strony internetowej w czasie rzeczywistym

**JSX (JavaScript XML)** - Jest to składnia rozszerzająca język JavaScript, która umożliwia tworzenie struktur drzewa DOM za pomocą składni przypominającej HTML umożliwiającej wewnętrzne zagnieżdżanie skryptów i zmiennych JavaScript.

W praktyce, JSX pozwala na tworzenie elementów interfejsu użytkownika w sposób deklaratywny i czytelny, co znacznie ułatwia pracę programistom. Przykładowo, zamiast tworzyć elementy DOM za pomocą metod JavaScript, możemy użyć JSX do definiowania ich w sposób podobny do pisania HTML, co sprawia, że kod jest bardziej przejrzysty i łatwiejszy do zrozumienia. Jednocześnie, JSX po kompilacji zostaje przekształcony w wywołania funkcji JavaScript, co pozwala na dynamiczną generację interfejsu użytkownika w oparciu o dane i logikę aplikacji.

# Założenia technologiczne

Praca wykonywana jest w czystym języku JavaScript, oraz daje możliwość implementacjii stron w JSX co pozwala programiście na dynamiczne zarządzanie zmiennymi, generowanie zawartości na ich podstawie oraz łatwą komunikacją za pomocą api z serverem.

# Struktura Projektu i Funkcje