



**WYŻSZA SZKOŁA
INFORMATYKI i ZARZĄDZANIA**
z siedzibą w Rzeszowie

Projekt: Języki i technologie webowe

Krystian Labak, w68962

Spis treści

Sposób uruchomienia projektu	4
Wykorzystane technologie i biblioteki.....	4
Architektura aplikacji.....	4
Proces powstawania projektu	5
Harmonogram	5
Planowanie i estymacja zadań.....	5
Podział zadań w zespole	5
Dokumentacja użytkownika.....	6
Przykłady użycia głównych funkcjonalności (user stories)	7
User Story 1: Kalkulator.....	7
User Story 2: Chatbot	7
User Story 3: Ceny kryptowalut.....	7
Podsumowanie	8

Sposób uruchomienia projektu

Sposób uruchomienia pliku znajduje się w dołączonym do projektu pliku README.txt

Wykorzystane technologie i biblioteki

Do stworzenia projektu zostały wykorzystane technologie takie jak:

- Blazor – framework do budowy interaktywnych aplikacji webowych przy użyciu c#
- .NET 8.0 – platforma do tworzenia aplikacji webowych i usług backendowych
- Bootstrap – framework CSS do stylizacji

Architektura aplikacji

Architektura aplikacji składa się z:

1. Warstwy prezentacyjnej
 - Komponenty blazor – poszczególne strony (kalkulator, chatbot, ceny kryptowalut) zbudowane jako komponenty Blazor.
 - Routing – mechanizm nawigacji pomiędzy podstronami za pomocą systemu routingu w Blazor.
2. Warstwa logiki biznesowej
 - Serwisy – klasy odpowiadające za logikę kalkulatora, obsługę komunikacji w czasie rzeczywistym (chatbot) oraz pobieranie danych o kursach kryptowalut.
3. Warstwa dostępu do danych
 - API zewnętrzne – usługa dostarczająca aktualne kursy kryptowalut oraz usługę dla chatbota

Proces powstawania projektu

Harmonogram

Etap projektu	Data rozpoczęcia	Data zakończenia	Opis
Planowanie, analiza i projektowanie architektury	01.01.2025	13.01.2025	Rozplanowanie, analiza funkcjonalności, wstępna budowa projektu
Implementacja kalkulatora	16.01.2025	17.01.2025	Budowa logiki kalkulatora oraz interfejsu
Integracja API chatbota	20.01.2025	25.01.2025	Połączenie z API, budowa logiki oraz interfejsu
Integracja API kryptowalut	26.01.2025	30.01.2025	Połączenie z API, logiki oraz budowa interfejsu

Planowanie i estymacja zadań

Każde z zadań zostało oszacowane pod względem czasu realizacji na podstawie złożoności i zależności między modułami:

- Planowanie oraz powstawanie „fundamentów” – 12 dni
- Kalkulator – 1 dzień
- Chatbot – 5 dni
- Kryptowaluty – 4 dni

Podział zadań w zespole

Projekt został zrealizowany przez jednego programistę

Dokumentacja użytkownika

Aplikacja jest przeznaczona do użytku zarówno na komputerach, jak i urządzeniach mobilnych. Poniżej opisano główne funkcjonalności oraz sposób ich wykorzystania.

Instrukcje użytkowania:

- Opis: Kalkulator umożliwia wykonanie podstawowych obliczeń matematycznych.

Kroki:

- 1. Przejdź do zakładki „Kalkulator”
- 2. Wprowadź liczby oraz wybierz operator który chcesz wykonać (dodawanie, odejmowanie, mnożenie, dzielenie)
- Po wciśnięciu operacji, wynik wyświetli się automatycznie
- Opis: Chatbot umożliwia interakcję z systemem poprzez zadawanie pytań i otrzymywanie odpowiedzi w czasie rzeczywistym

Kroki:

- Przejdź do zakładki „Chatbot”
- Wpisz pytanie lub wiadomość w polu tekstowym
- Naciśnij przycisk Zapytaj, a wiadomość zostanie wyświetlona przez sztuczną inteligencję
- Opis: Podstrona pokazująca aktualne kursy kryptowalut

Kroki:

- Przejdź do zakładki „Kryptowaluty”
- Użytkownik może przeglądać aktualne kursy kryptowalut, dzięki zautomatyzowanemu systemowi pobierania danych
- Użytkownik może dodać do ulubionych wybraną kryptowalutę i dzięki temu może śledzić tylko ulubioną

Przykłady użycia głównych funkcjonalności (user stories)

User Story 1: Kalkulator

Użytkownik chce korzystać z kalkulatora i móc szybko obliczyć wynik operacji matematycznej.

- Scenariusz:
 1. Użytkownik uruchamia aplikację
 2. Wybiera zakładkę „Kalkulator”
 3. Wprowadza dwie liczby i system automatycznie wykonuje obliczenie w momencie kliknięcia operacji
- Kryteria akceptacji:
 - Kalkulator wykonuje operacje poprawnie.
 - Wynik jest wyświetlany w czytelny sposób.

User Story 2: Chatbot

Użytkownik chce zadać pytanie dla chatbota, aby uzyskać odpowiedź na zadane pytanie

- Scenariusz:
 1. Użytkownik uruchamia aplikację
 2. Wybiera zakładkę „Chatbot”
 3. Zadaje pytanie w polu tekstowym i wciska przycisk zapytaj
 4. Użytkownik dostaje szybką odpowiedź
- Kryteria akceptacji:
 - Chatbot odpowiada w sposób zrozumiały, oraz w poprawnym języku
 - Interfejs rozmowy jest intuicyjny

User Story 3: Ceny kryptowalut

Użytkownik chce śledzić aktualne ceny kryptowalut, aby podejmować poprawne decyzje inwestycyjne.

- Scenariusz:
 1. Użytkownik uruchamia aplikację
 2. Wybiera zakładkę „Kryptowaluty”
 3. System automatycznie pobiera aktualne kursy z zewnętrznego API
 4. Użytkownik ma możliwość dodania sobie konkretnej kryptowaluty do ulubionych
- Kryteria akceptacji:
 - Dane są aktualizowane w czasie rzeczywistym
 - Interfejs wyświetla informacje w sposób czytelny i jasny dla użytkownika

Podsumowanie

W dokumentacji została przedstawiona aplikacja wykonana w technologii Blazor. Omówiono wykorzystane technologie, architekturę aplikacji, wdrażania projektu oraz została przedstawiona instrukcja użytkownika wraz z przykładami user stories. Projekt jest gotowy do wszelkiej modyfikacji oraz edycji, jednakże w planach jest dalszy rozwój nad projektem, takim jak:

- Stworzenie okna do zalogowania się (w tym celu konieczne jest połączenie z bazą danych, w celu przechowywania użytkowników oraz ich zapisanych do ulubionych kryptowalut, lub historia rozmów z Chatbotem)
- Zabezpieczenia (aktualnie strona działa w trybie offline, zatem nie ma zagrożenia jeśli chodzi o cyberataki, jednakże będzie trzeba wprowadzić zabezpieczenia, jeśli podejmie się chęć udostępnienia projektu jako strony internetowej)
- Rekonstrukcja interfejsu (Aktualny interfejs jest prosty, jednak nie przyciąga użytkownika, konieczna byłaby modyfikacja CSS, by strona była bardziej zachęcająca)
- Rozbudowanie struktury (Dodanie większej liczby podstron, by strona była bardziej rozbudowana)