**Politechnika Wrocławska**

**Wydział Informatyki i Telekomunikacji**

Kierunek: IST

**ZESPOŁOWE PRZEDSIĘWZIĘCIE INFORMATYCZNE**

**E-Wybory - System umożliwiający przeprowadzenie głosowania w wyborach**

Jan Jankowski

Michał Starba

Krzysztof Saar

Krzysztof Wróblewski

Opiekun pracy

**mgr inż. Dominika Dudziak-Gajowiak**

Słowa kluczowe: wybory, głosowanie, kandydat, wyborca, głos, aplikacja webowa

WROCŁAW (rok realizacji ZPI)

**Spis treści**

[DOKUMENTACJA PROJEKTOWA 3](#_Toc178333481)

[1. Wykaz symboli, oznaczeń i akronimów (opcja) 3](#_Toc178333482)

[2. Cel i zakres przedsięwzięcia 4](#_Toc178333483)

[3. Słownik pojęć (opcja) 5](#_Toc178333484)

[4. Stan wiedzy w obszarze przedsięwzięcia (opcja) 6](#_Toc178333485)

[5. Założenia wstępne 7](#_Toc178333486)

[6. Specyfikacja wymagań na produkt programowy 9](#_Toc178333487)

[7. Projekt produktu programowego 9](#_Toc178333488)

[8. Implementacja (opcja) 9](#_Toc178333489)

[9. Wyniki i analiza badań/ Demonstracja produktu programowego (w zależności od typu projektu) 9](#_Toc178333490)

# DOKUMENTACJA PROJEKTOWA

## 1. **Wykaz symboli, oznaczeń i akronimów (opcja)**

W dokumentacji zostały użyte następujące oznaczenia dot. publicznych instytucji:

* PKW – Państwowa Komisja Wyborcza
* PWr – Politechnika Wrocławska

Poza tym wykorzystane zostały również skróty technologiczne tj.:

* VM – Virtual Machine (maszyna wirtualna)
* API - Application Programming Interface (interfejs programowania aplikacji)
* 2FA – Two Factor Authentication(uwierzytelnianie dwuskładnikowe)
* JWT – JSON Web Token
* ORM – Object Relational Mapping (mapowanie obiektowo relacyjne)
* SSR – Server Side Rendering
* MVVM – Model-View-ViewModel
* LTS – Long Term Support(długoterminowe wsparcie)

## 2. **Cel i zakres przedsięwzięcia**

Celem naszego projektu jest przygotowanie oprogramowania, które umożliwi udział w wyborach bez potrzeby udawania się do lokalu wyborczego.

Aplikacja ma przyczynić się do usprawnienia procesu liczenia głosów oraz ograniczenia liczby błędów, co zwiększy efektywność całego procesu wyborczego. Ponadto udostępni ona informacje dot. wyborów w jednym miejscu zarówno na temat głosowania, jak i kandydatów. Użytkownicy będą mieli również wgląd do statystyk i wyników przeprowadzonych wyborów.

Oprogramowanie zapewni wysoki poziom bezpieczeństwa poprzez odpowiednio dobrane i użyte technologie. Obejmą one m.in.: szyfrowanie, ochronę tożsamości użytkowników oraz integralność oddawanych głosów.

## 3. **Słownik pojęć (opcja)**

* Okręg – jednostka terytorialna w strukturze wyborczej. Składa się z kilku mniejszych jednostek, zwanych obwodami.
* Obwód – teren, w którym wyborcy głosują w konkretnym lokalu wyborczym
* Użytkownik – osoba korzystająca z aplikacji. W zależności od przydzielonego typu posiada różne uprawnienia.
* Typ użytkownika - oznaczenie przypisane do użytkownika systemu, które definiuje, jakie działania może podejmować w aplikacji
* Grupa typów użytkownika – zbiór typów użytkownika mogący obejmować różne role, umożliwiając łączenie uprawnień i dostosowanie poziomu dostępu
* Wyborca – osoba uprawniona do udziału w wyborach. Jest to jeden z typów użytkownika, który jest przypisany każdemu w systemie
* Wybory – proces polegający na zorganizowaniu wydarzenia, które odbywa się w określonym czasie i ma ustalony rodzaj. W ramach wyborów mierzona jest frekwencja oraz wyniki głosowania.
* Rodzaj wyborów - kategoria, określająca ich cel i zakres. Każdy rodzaj wyborów ma swoją specyfikę, takie jak zasady głosowania i kategorie wybieranych przedstawicieli.
* Kandydat – osoba ubiegająca się o wybór na określone stanowisko publiczne w danych wyborach
* Partia – zbiór kandydatów, którzy działają wspólnie w wyborach
* Głos – wybór przypisany kandydatowi w danej instancji wyborów. Głosy oddane na kandydatów są sumowane, aby określić wyniki wyborów.

## 4. **Stan wiedzy w obszarze przedsięwzięcia (opcja)**

np. analiza istniejących rozwiązań z podsumowaniem, opis porównywanych metod

Na przestrzeni lat wdrażano wiele rozwiązań umożliwiających zdalne głosowanie. W niektórych państwach systemy te zyskały dużą popularność i stabilność, podczas gdy w innych natrafiły na wyzwania związane z bezpieczeństwem, frekwencją lub dostępnością.

* Estonia

|  |  |
| --- | --- |
| Zalety | Wady |
| Transparentny proces głosowania (kod klienta udostępniony publicznie) | Wysokie koszty wdrożenia i utrzymania infrastruktury |
| Możliwość wielokrotnego głosowania (ostatni głos się liczy) | Obawy o pogłębienie przepaści cyfrowej, |
| Zaawansowane mechanizmy bezpieczeństwa i liczne audyty |  |
| Dostępność ogólnokrajowa |  |

* Szwajcaria

|  |  |
| --- | --- |
| Zalety | Wady |
| Zdecentralizowany system, umożliwiający różnym kantonom wdrażanie własnych rozwiązań | Wciąż przeprowadzane pilotażowe testy systemów, ograniczona dostępność |
| Duże zaangażowanie w ochronę prywatności i bezpieczeństwa głosowania | Powolny rozwój ze względu na liczne ograniczenia |

* Francja

|  |  |
| --- | --- |
| Zalety | Wady |
| Umożliwienie głosowania obywatelom przebywającym poza granicami kraju | Ograniczony dostęp, system dostępny tylko w niektórych wyborach |
|  | Obawy o bezpieczeństwo wyborów, zawieszenie projektu |

* Rosja

|  |  |
| --- | --- |
| Zalety | Wady |
| Wykorzystanie platformy blockchain | Prywatne serwery, dostęp tylko dla wybranych urzędników – negacja korzyści blockchainu |
|  | Liczne wątpliwości co do rzetelności systemu |

## 5. **Założenia wstępne**

np. dobór technologii, przyjęte ograniczenia

W projekcie zdecydowaliśmy się użyć technologii spełniających nasze wymagania.

|  |  |  |
| --- | --- | --- |
| **Technologia** | **Tematyka** | **Powód wyboru** |
| MySQL InnoDB Cluster | Baza danych | Niezawodność i wydajność w obsłudze transakcji. Dzięki zastosowaniu klastrów możliwe jest zapewnienie ciągłej dostępności bazy danych oraz minimalizacji ryzyka utraty danych |
| .NET 8 Blazor Web App (Blazor Web Assembly + Blazor Server) | Backend/frontend | Umożliwia sprawne tworzenie wydajnych aplikacji z zachowaniem jednolitego języka programowania (C#) zarówno na froncie, jak i na serwerze, co ułatwia utrzymanie kodu |
| ASP.NET Core – WEB API + SSR | Backend | Pozwala na budowę skalowalnych i wydajnych API. SSR poprawia dostępność dla wyszukiwarek |
| ORM – Entity Framework | Backend | Przyspiesza proces tworzenia i zarządzania danymi w bazie |
| JWT | Backend | Pozwala na bezpieczne przesyłanie informacji o użytkownikach pomiędzy serwerem a klientem |
| MVVM | Frontend | Ułatwia zarządzanie kodem, testowanie i rozwijanie aplikacji, a także pozwala na jej lepsze dostosowanie do potrzeb użytkowników |
| Linux + .NET Runtime | Serwer aplikacji | Zapewnia stabilność, bezpieczeństwo oraz wydajność |
| 2FA | Zabezpieczenie | Zwiększa poziom bezpieczeństwa aplikacji, minimalizując ryzyko nieautoryzowanego dostępu do kont użytkowników |
| Ubuntu LTS na VM zahostowanej na PWr | Hosting | Zapewnia długoterminowe wsparcie i bezpieczeństwo. Hosting na PWr jest łatwo dostępny |

Do uwierzytelniania użytkowników zastosowaliśmy 2FA, choć planowaliśmy wykorzystać funkcjonalność Profilu Zaufanego oraz mObywatela. Jednakże odmówiono nam dostępu do wspomnianych narzędzi.

Zdecydowaliśmy, że konto użytkownika jest przypisane do jednej osoby. Dzięki temu każdy użytkownik może oddać tylko jeden głos w danych wyborach. Takie podejście minimalizuje redundancję danych i ułatwia zarządzanie przypisywaniem uprawnień użytkownikom. Dodatkowo zwiększa to bezpieczeństwo wyborów, ponieważ eliminuje ryzyko manipulacji wynikami.

## 6. **Specyfikacja wymagań na produkt programowy**

np. definicja wymagań funkcjonalnych/niefunkcjonalnych; możliwe formy: diagram wymagań, diagram przypadków użycia, lista historyjek (może być pogrupowana w epiki)

## W celu zapewnienia jasności i zrozumienia funkcji systemu, konieczne było zidentyfikowanie kluczowych wymagań, które zostały przestawione w formie diagramów przypadków użycia. Zastosowaliśmy strukturę podziału na epiki, gdzie każdy reprezentuje odrębną grupę funkcjonalności związanych z określonym typem użytkownika. W ten sposób klarowniej można zdefiniować rolę i możliwości każdego aktora w naszym systemie.

Poniżej znajduje się lista epików wraz z diagramami przypadków użycia :

Epik 1. Implementacja funkcjonalności użytkownika niezalogowanego

Obraz zawierający tekst, diagram, zrzut ekranu, krąg

Opis wygenerowany automatycznie

Epik 2. Implementacja weryfikacji użytkownika i podstawowych funkcji wyborcy

Obraz zawierający tekst, diagram, zrzut ekranu, krąg

Opis wygenerowany automatycznie

Epik 3. Implementacja funkcjonalności Państwowej Komisji Wyborczej

Obraz zawierający tekst, diagram, zrzut ekranu, pismo odręczne

Opis wygenerowany automatycznie

Epik 4. Implementacja funkcjonalności Urzędnika Wyborczego

Obraz zawierający tekst, diagram, pismo odręczne, rysowanie

Opis wygenerowany automatycznie

Epik 5. Implementacja funkcjonalności operatora w lokalu wyborczym

Obraz zawierający tekst, diagram, zrzut ekranu, krąg

Opis wygenerowany automatycznie

## 7. **Projekt produktu programowego**

np. opis decyzji architektonicznych, projekt architektury, bazy danych, zastosowane wzorce projektowe

## 8. **Implementacja (opcja)**

np. opis nietrywialnych rozwiązań implementacyjnych, w tym algorytmów (jeżeli ma zastosowanie)

## 9. **Wyniki i analiza badań/ Demonstracja produktu programowego (w zależności od typu projektu)**

* 1. np. Opis realizacji typowych zadań z podziałem na ich typy i/lub aktorów

## 