

Specyfikacja wymagań dla systemu rekrutacji na studia

1. Charakterystyka systemu

1.1 Nazwa skrócona: SRS

1.2 Nazwa pełna: System Rekrutacji na Studia

1.3 Opis:

Niniejszy dokument opisuje wymagania systemowe dla Systemu Rekrutacji na Studia. System ten ma umożliwić elektroniczne wypełnianie zgłoszeń rekrutacyjnych oraz przysyłanie dokumentów przez kandydatów. Celem systemu jest gromadzenie danych potrzebnych do rekrutacji studentów, a także pomoc kandydatom oraz pracownikom uczelni w sprawnym przeprowadzeniu procesu rekrutacji.

2. Prawa autorskie

2.1 Autor: Michał Machnik

2.2 Warunki licencyjne: licencja MIT (Open Source)

3. Specyfikacja wymagań

3.1 Wymagania funkcjonalne:

I. Id: FR1

Nazwa: Lista kierunków

Opis: System musi umożliwiać kandydatom przeglądanie listy dostępnych kierunków studiów wraz z ich programami.

Priorytet: Wysoki

II. Id: FR2

Nazwa: Elektroniczne zgłoszenia i przysyłanie dokumentów.

Opis: System musi umożliwiać kandydatom na studia elektroniczne wypełnianie i przysyłanie swoich zgłoszeń rekrutacyjnych oraz wymaganych dokumentów.

Priorytet: Wysoki.

III. Id: FR3

Nazwa: Opłaty rekrutacyjne.

Opis: System musi umożliwiać kandydatom na studia przysyłanie opłat rekrutacyjnych.

Priorytet: Wysoki

IV. Id: FR4

Nazwa: Wgląd w listę zgłoszeń

Opis: System musi umożliwiać wgląd dla administratorów w listę zgłoszeń rekrutacyjnych i przestanych dokumentów.

Priorytet: Wysoki

- V. Id: FR5
Nazwa: Przyznawanie punktów rekrutacyjnych.
Opis: System musi umożliwiać administratorom przyznawanie i podliczanie punktów rekrutacyjnych kandydatów na podstawie zgłoszeń i dokumentów.
Priorytet: Wysoki
- VI. Id: FR6
Nazwa: Wybór kandydatów
Opis: System musi umożliwiać administratorom wybór kandydatów na podstawie wyników oceny zgłoszeń rekrutacyjnych i przyznanych punktów.
Priorytet: Wysoki
- VII. Id: FR7
Nazwa: Wprowadzanie wyników rekrutacji
Opis: System musi umożliwiać administratorom wprowadzenie wyników rekrutacji.
Priorytet: Wysoki
- VIII. Id: FR8
Nazwa: Wyświetlanie wyników rekrutacji
Opis: System musi umożliwiać kandydatom wyświetlanie wyników rekrutacji, w których bierze udział.
Priorytet: Wysoki
- IX. Id: FR9
Nazwa: Przypisanie programów studiów
Opis: System musi umożliwiać przypisanie wybranym kandydatom do odpowiednich programów studiów i semestrów.
Priorytet: Średni
- X. Id: FR10
Nazwa: Zarządzanie danymi osobowymi
Opis: System musi umożliwiać zarządzanie danymi osobowymi studentów
Priorytet: Średni

3.2 Wymagania нефункциональные:

- I. Id: NFR1
Nazwa: Dostępność systemu
Opis: System musi być dostępny przez przeglądarki internetowe z komputerów i urządzeń mobilnych
Priorytet: Wysoki
- II. Id: NFR2
Nazwa: Łatwość obsługi
Opis: System musi umożliwiać łatwe przeglądanie i wyszukiwanie informacji, aby umożliwić użytkownikom łatwe korzystanie z niego bez specjalistycznej wiedzy informatycznej.
Priorytet: Średni

- III. Id: NFR3
Nazwa: Bezpieczeństwo systemu
Opis: System musi być bezpieczny i chronić poufne informacje, takie jak dane osobowe kandydatów i studentów oraz informacje dotyczące ich dokumentów rekrutacyjnych.
Priorytet: Wysoki
- IV. Id: NFR4
Nazwa: Stabilność systemu
Opis: System musi być stabilny i niezawodny, aby zapewnić ciągłość procesu rekrutacji.
Priorytet: Średni
- V. Id: NFR5
Nazwa: Wydajność i szybkość systemu
Opis: System musi spełniać wymagania dotyczące wydajności i szybkości odpowiedzi, aby umożliwić szybką i skuteczną obsługę zgłoszeń aplikacji i innych zadań.
Priorytet: Średni
- VI. Id: NFR6
Nazwa: Integracja z innymi systemami
Opis: System musi umożliwiać łatwą integrację z innymi systemami, takimi jak systemy płatnicze i systemy uczelni.
Priorytet: Średni

4. Architektura systemu

4.1 Wymagania architektoniczne dotyczące architektury rozwoju:

- a) System musi być oparty na architekturze wielowarstwowej (np. Model-Widok-Kontroler lub podobnej).
- b) Aplikacja internetowa powinna być napisana w języku JavaScript oraz używać frameworka, takiego jak React lub Angular.
- c) System powinien korzystać z bazy danych relacyjnej, takiej jak np. MySQL.

4.2 Wymagania architektoniczne dotyczące architektury uruchomieniowej:

- a) System musi działać na serwerach z systemem operacyjnym Linux lub Windows.
- b) Aplikacja internetowa musi działać na serwerze WWW z zainstalowanym oprogramowaniem serwera aplikacji (np. Node.js lub PHP).
- c) System musi umożliwiać bezpieczne połączenia z serwerem poprzez protokół HTTPS.