|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | | | | |
|  | |  | Wydział Informatyki i Zarządzania kierunek studiów: Informatyka  specjalność:  Praca dyplomowa - inżynierska  **Webowy system do video-rekrutacji pracowników**  Bartosz Domagała  słowa kluczowe:  system webowy  video  rekrutacja  krótkie streszczenie:  Praca opisuje problem jaki stanowi skuteczna rekrutacja pracownika na  stanowisko, oraz procedurę powstawania prototypu systemu webowego mającego  ułatwić selekcję kandydatów do zaproszenia na rozmowę kwalifikacyjną  ułatwić selekcję kandydatów do zaproszenia na rozmowę kwalifikacyjną  poprzez umożliwienie załączenia krótkiego video do każdego podania   |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | opiekun pracy  dyplomowej | .................................................. | | ....................... | | ....................... | | | *Tytuł/stopień naukowy/imię i nazwisko* | | *ocena* | | *podpis* | | | Ostateczna ocena za pracę dyplomową | | | | | | | | | Przewodniczący Komisji egzaminu dyplomowego | | | ..................................................  *Tytuł/stopień naukowy/imię i nazwisko* | | ....................... | | ....................... | | *ocena* | | *podpis* |   *Do celów archiwalnych pracę dyplomową zakwalifikowano do:\**   1. *kategorii A (akta wieczyste)* 2. *kategorii BE 50 (po 50 latach podlegające ekspertyzie)*   *\* niepotrzebne skreślić*   |  | | --- | | pieczątka wydziałowa | | |
|  |  | |  | Wrocław 2018 |

**Webowy system do video-rekrutacji pracowników**

# Streszczenie

Nagrania video mają coraz szersze zastosowanie we współczesnym świecie. W odpowiedzi na ten trend zaproponowane zostało zastosowanie nagrania jako materiału wspierającego lub nawet zastępującego klasyczne dokumenty składane podczas rekrutacji. Celem pracy było wytworzenie prototypu webowego systemu do video-rekrutacji pracowników. System ten daje możliwość zamieszczania ofert pracy oraz aplikowania na nie.

W ramach pracy opisano problem jaki stanowi skuteczna rekrutacja pracownika na stanowisko oraz procedurę powstawania prototypu systemu webowego. Cechą charakterystyczną wytworzonego systemu jest możliwość załączenia do każdego podania krótkiego filmu jako alternatywy listu motywacyjnego lub cv. To pozwala na łatwiejsze wyłonienie obiecujących kandydatów.

Ostatecznie przeanalizowane zostały dalsze możliwości rozwoju powstałego systemu z uwzględnieniem ograniczeń prawnych, które system musi spełnić aby został udostępniony publicznie.

**Web-based recruitment video system**

# Abstract

Video recordings are more and more widely used in modern world. In response to this trend using video as support or substitute for traditional recruitment documents has been proposed. Goal of this thesis was to create prototype of the web-based recruitment video system. This system offers the possibility of publishing job offers and applying for posted jobs.

During this thesis the problem of effective recruitment was described. Also the process of creating prototype of the web system was described. Characteristic feature of the system is that it allows for attachment of video file to a job application instead of letter of motivation or CV. This allows for easier selection of promising candidates.

Lastly opportunities for future development of created system was analysed. This analysis included legal restrictions that need to be fulfilled before system can be available publicly.

Spis treści

[Streszczenie 3](#_Toc504416517)

[Abstract 3](#_Toc504416518)

[Rozdział 1 - Wstęp 6](#_Toc504416519)

[Cel i zakres pracy: 6](#_Toc504416520)

[Realizacja: 6](#_Toc504416521)

[Rozdział 2 - Wprowadzenie 8](#_Toc504416522)

[2.1 Opis problemu: 8](#_Toc504416523)

[2.2 Propozycja rozwiązania: 8](#_Toc504416524)

[2.3 Użytkownicy 9](#_Toc504416525)

[2.4 Historyjki użytkownika 9](#_Toc504416526)

[Rozdział 3 – Projekt aplikacji 10](#_Toc504416527)

[3.1 Diagram 10](#_Toc504416528)

[3.2 Scenariusze 11](#_Toc504416529)

[3.3 Scenariusze logowania i rejestracji 17](#_Toc504416530)

[3.4 Projekt bazy danych 18](#_Toc504416531)

[3.4.1 Encje 18](#_Toc504416532)

[3.4.2 Diagram 19](#_Toc504416533)

[Rozdział 4 – Implementacja 20](#_Toc504416534)

[4.1 Kryteria wyboru technologii 20](#_Toc504416535)

[4.2 Zastosowane technologie 20](#_Toc504416536)

[4.3 Metodyka wytwarzania oprogramowania 21](#_Toc504416537)

[4.4 Interfejs użytkownika 21](#_Toc504416538)

[Rozdział 5 – Testowanie 25](#_Toc504416539)

[7.1 Testy 25](#_Toc504416540)

[7.2 Analiza wyników testów i naprawa błędów 25](#_Toc504416541)

[Podsumowanie i wnioski: 26](#_Toc504416542)

[Aspekt prawny 26](#_Toc504416543)

[Literatura 27](#_Toc504416544)

[Załączniki 27](#_Toc504416545)

Spis ilustracji

[Rysunek 1 - Diagram przypadków użycia 10](#_Toc504416546)

[Rysunek 2 - Diagram związków encji 19](#_Toc504416547)

[Rysunek 3 - formularz logowania 21](#_Toc504416548)

[Rysunek 4 - Formularz tworzenia/edycji oferty 22](#_Toc504416549)

[Rysunek 5 - Lista opublikowanych ofert 22](#_Toc504416550)

[Rysunek 6 - Widok oferty dla publikującego 23](#_Toc504416551)

[Rysunek 7 - Widok oferty dla aplikującego 23](#_Toc504416552)

[Rysunek 8 - Widok oferty przed złożeniem podania 24](#_Toc504416553)

# Rozdział 1 - Wstęp

Proces rekrutacji pracownika na stanowisko wiąże ze sobą problemy zebrania podań od wielu potencjalnych pracowników, oraz selekcji najbardziej obiecujących spośród nich na podstawie otrzymanych informacji. Na jedno stanowisko często aplikuje za dużo osób aby móc z każdym przeprowadzić rozmowę kwalifikacyjną. Klasycznym podejściem do problemu jest Curriculum Vitae i/lub list motywacyjny. Jednak ilość informacji uzyskanych z tych dokumentów jest ograniczona, zwłaszcza w przypadku gdy rekruter wymaga takich cech jak dobra prezencja lub dykcja. W pracy zaproponowano rozwiązanie problemu poprzez zastosowanie webowego systemu do video-rekrutacji. Jako system webowy ułatwi on zbieranie podań, a możliwość załączenia video do podania pozwoli pracodawcy łatwiej wybrać kandydatów, a składającemu podanie da dodatkową możliwość zainteresowania pracodawcy. W ramach pracy realizowany został prototyp, który nie jest publicznie dostępny z powodów prawnych[Aspekt prawny].

Ten temat został wybrany ze względu na to, że oferuje interesującą możliwość zastosowania nagrania video jako zastępnika lub materiału wspierającego podczas procesu rekrutacji nowego pracownika. System posiadający taką funkcjonalność z pewnością mógłby znaleźć zastosowanie w biznesie. Zainteresowane mogłyby być głównie branże takie jak marketing, w których jest wiele stanowisk na których bardzo ważna jest prezencja, dykcja czy postawa. Oczywiście inne branże też mogłyby na nim skorzystać w sytuacji gdy kandydatów jest zbyt wielu i trzeba ograniczyć liczbę rozmów kwalifikacyjnych.

## Cel i zakres pracy:

Celem pracy jest projekt i implementacja systemu webowego do video-rekrutacji pracowników. System ma umożliwiać umieszczanie ofert pracy, aplikowanie na zamieszczone oferty, oraz załączania plików video do podań. W celu realizacji zaproponowanego systemu konieczne jest:

* Przeprowadzenie przeglądu dostępnych technologii i wybrać te odpowiednie do problemu
* Projekt i implementacja bazy danych, w której przechowywane będą dane firm, potencjalnych pracowników, oferty, oraz podania
* Projekt i implementacja logiki biznesowej po stronie serwera
* Projekt i implementacja interfejsu użytkownika
* Testowanie systemu

## Realizacja:

W ramach realizacji zapoznano się z dostępnymi technologiami i zdecydowano na zastosowanie:

* Baza danych (dla prototypu) – MySQL
* Połączenie z bazą – Hibernate
* Logika biznesowa – Java + Spring
* Interfejs użytkownika – HTML, CSS, JavaScript, Thymeleaf

Realizację pracy rozpoczęto od zebrania historyjek użytkownika na podstawie których utworzono diagram przypadków użycia i ich scenariusze. Etap został bardziej szczegółowo opisany w rozdziale [Rozdział 3 – ].

Kolejnym krokiem było stworzenie projektu bazy danych zdolnej obsłużyć wszystkie historyjki użytkownika wybrane do realizacji w ramach pracy inżynierskiej. Etap został bardziej szczegółowo opisany w rozdziale [3.4 Projekt bazy danych]

Jako ostatni zaprojektowany został interfejs użytkownika. Na podstawie przypadków użycia stworzono szkic interfejsu.

Kolejnym krokiem była implementacja bazy danych oraz logiki biznesowej. W implementacji zastosowano metodykę iteracyjno-przyrostową.

Następnie dokonano implementacji interfejsu użytkownika.

Ostatnim etapem było testowanie uzyskanego systemu.

# Rozdział 2 - Wprowadzenie

## 2.1 Opis problemu:

Proces rekrutacji należy rozpatrywać z punktu widzenia dwóch grup użytkowników:

* Rekruter
* Kandydat

Rekruter:

Początkiem procesu jest przygotowanie oferty pracy oraz opublikowanie jej.  
Następnie rekruter oczekuje na podania dotyczące tej oferty. Gdy uzna, że liczba podań jest wystarczająca rekruter dokonuje selekcji kandydatów na podstawie otrzymanych zgłoszeń i rozpoczyna przygotowywanie rozmów kwalifikacyjnych. Jeśli kandydat lub kandydaci przejdą wszystkie etapy rekrutacji i zostaną zatrudnieni proces rekrutacji kończy się, a oferta zostaje zamknięta. W przeciwnym wypadku rekruter oczekuje na kolejne zgłoszenia, lub zmienia warunki oferty w celu przyciągnięcia większej liczby kandydatów. W przypadku zamknięcia pozycji z dowolnych przyczyn proces kończy się.

Dla rekrutera głównym problemem jest wybór spośród kandydatów osoby najbardziej odpowiedniej na stanowisko którego dotyczy oferta. Znaczna część osób zaproszonych na rozmowę okazuje się nie spełniać oczekiwań, co kosztuje pracodawcę czas i zasoby.

Kandydat:

Dla kandydata początkiem rekrutacji jest przeglądanie opublikowanych ofert pracy. W przypadku znalezienia interesującej oferty przygotowuje on podanie, a następnie dostarcza je pracodawcy. W ramach podania najczęściej załączane jest CV. Po dostarczeniu podania kandydatowi pozostaje oczekiwać na odpowiedź rekrutera. Jeśli dostarczone podanie przejdzie wstępną selekcję kandydat zostanie zaproszony na rozmowę, która stanowi kolejny etap rekrutacji. Dla kandydata rekrutacja kończy się w momencie gdy zostanie on zatrudniony lub jego podanie zostanie odrzucone.

Dla kandydata największym wyzwaniem jest stworzenie podania które wyróżni go spośród konkurencji. Jest to jednak trudne do wykonania w przypadku zwięzłej formy jaką jest CV, a jeśli list motywacyjny nie zainteresuje rekrutera to najprawdopodobniej nie zostanie przeczytany.

## 2.2 Propozycja rozwiązania:

W celu zaadresowania powyższych problemów oraz usprawnienia procesu zaproponowane zostało stworzenie webowego systemu do video-rekrutacji pracowników. Cechy systemu webowego rozwiązują wiele problemów związanych z rekrutacją.

Przede wszystkim jako system webowy umożliwia on dostęp za pośrednictwem klienta, w tym przypadku strony internetowej. Dzięki temu zarówno składanie podania jak i przeglądanie podań odbywa się w prosty sposób – bez konieczności wysyłania podania pocztą zwykłą lub elektroniczną.

Ponadto opcja załączania krótkiego filmu video do zgłoszenia ułatwi kandydatom wyróżnienie swojego zgłoszenia, a pracodawcy selekcję kandydatów.

## 2.3 Użytkownicy

Jak już napisano w procesie rekrutacji uczestniczą dwie grupy użytkowników:

* Rekruter – Osoba lub instytucja zamieszczająca ofertę w serwisie, a następnie oceniająca podania z nią związane
* Kandydat – Osoba poszukująca pracy, która przegląda oferty i składa podania na nie

## 2.4 Historyjki użytkownika

Dla wyznaczonych użytkowników zebrano następujące historyjki użytkownika:

1. Jako pracownik chcę założyć konto aby zapamiętać część danych
2. Jako pracownik chcę przeglądać oferty pracy aby znaleźć najbardziej odpowiednią
3. Jako pracownik chcę aplikować na stanowisko aby zdobyć posadę
4. Jako pracownik chcę załączyć film do podania aby przekazać dodatkowe informacje
5. Jako pracownik chcę dodawać załączniki do zgłoszenia aby spełnić wymogi oferty
6. Jako pracownik chcę by dane, które podaje były zabezpieczone, aby uniknąć ich kradzieży
7. Jako pracownik chcę otrzymywać powiadomienia o postępie podań aby być na bieżąco w przypadku ich akceptacji/odrzucenia
8. Jako rekruter chcę założyć konto, aby inni nie mogli dodawać ofert w moim imieniu
9. Jako rekruter chcę zamieszczać oferty pracy aby potencjalni pracownicy mogli na nie aplikować
10. Jako rekruter chcę móc edytować oferty pracy, aby odpowiadały one ewentualnym zmianom
11. Jako rekruter chcę podawać opis oferty aby doprecyzować otwartą pozycję
12. Jako rekruter chcę podawać wymagania do oferty aby uzyskać podania od odpowiednich ludzi
13. Jako rekruter chcę odrzucać podania aby dyskwalifikować kandydatów nie spełniających wymagań
14. Jako rekruter chcę przydzielać podaniom oznaczenia aby ułatwić selekcję w przypadku dużej liczby zgłoszeń
15. Jako rekruter chcę wyświetlać film załączony do zgłoszenia oraz CV, aby dokonać oceny kandydata

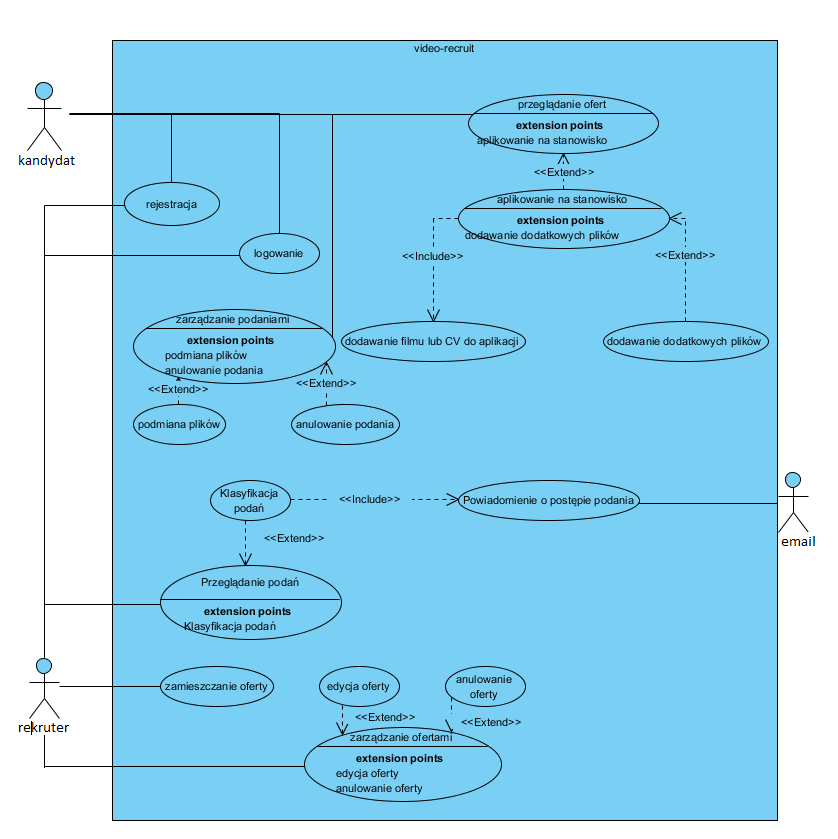
Wybrano formę historyjek użytkownika, ponieważ jest to wygodna i łatwa do zrozumienia forma zbierania wymagań. Starano się utrzymać historyjki niewielkich rozmiarów aby dało się je sprawnie zaimplementować.[3]

# Rozdział 3 – Projekt aplikacji

Następnie dokonano analizy zebranych historyjek użytkownika i na ich podstawie rozpoczęto projekt aplikacji. W tej części pracy zebrane zostały wymagania funkcjonalne zebrane zostały wymagania funkcjonalne w postaci przypadków użycia. Przypadki użycia zostały następnie opisane poprzez scenariusze uwzględniające scenariusze alternatywne. Ostatecznie na podstawie zaprojektowano schemat bazy danych umożliwiający ich obsługę.

## 3.1 Diagram

Na podstawie analizy zebranych historyjek użytkownika wytworzono następujący diagram przypadków użycia. [Rysunek 1 - Diagram przypadków użycia] Diagram został wykonany przy użyciu narzędzia Visual Paradigm[4]



Rysunek 1 - Diagram przypadków użycia

## 3.2 Scenariusze

|  |  |
| --- | --- |
| Przypadek użycia | Aplikowanie na stanowisko |
| Główny aktor | Kandydat |
| Krótki opis | Kandydat aplikuje na stanowisko |
| Warunki wstępne | Kandydat jest zalogowany |
| Scenariusz | |  |  |  | | --- | --- | --- | |  | Działanie aktora | Odpowiedź systemu | | 1 | Kandydat wchodzi w przeglądanie ofert |  | | 2 |  | System wyświetla dostępne oferty | | 3 | Kandydat wybiera interesującą go ofertę |  | | 4 |  | System wyświetla szczegóły oferty | | 5 | Kandydat wybiera opcję aplikowania na stanowisko |  | | 6 |  | System wyświetla formularz podania | | 7 | Kandydat załącza CV i video |  | | 8 | [Opcjonalnie] Kandydat załącza dodatkowe pliki |  | | 9 | Kandydat zatwierdza podanie |  | | 10 |  | System wyświetla potwierdzenie otrzymanego podania | |
| Warunki końcowe | Podanie zostało przyjęte i zapisane w bazie |
| Ścieżki alternatywne | 6a. Podanie na to stanowisko już zostało złożone – przejdź do PU”Modyfikacja Podania” krok 5 7a. Brak CV – system wyświetla powiadomienie 9a. Kandydat anuluje składanie podania - koniec |

|  |  |
| --- | --- |
| Przypadek użycia | Modyfikacja podania |
| Główny aktor | Kandydat |
| Krótki opis | Kandydat modyfikuje podanie, jeśli nie zostało jeszcze ocenione przez rekrutera |
| Warunki wstępne | Kandydat jest zalogowany |
| Scenariusz | |  |  |  | | --- | --- | --- | |  | Działanie aktora | Odpowiedź systemu | | 1 | Kandydat wchodzi w historię swoich podań |  | | 2 | Kandydat wybiera opcję modyfikacji |  | | 3 |  | System wyświetla listę podań, których status umożliwia modyfikację(czyli nie „zaakceptowana”, „odrzucona”) | | 4 | Kandydat wybiera podanie z listy |  | | 5 |  | System wyświetla szczegóły oferty i formularz podania | | 6 | Kandydat przesyła nowe pliki |  | | 7 | Kandydat zatwierdza podanie |  | | 8 |  | System wyświetla potwierdzenie dla klienta | |
| Warunki końcowe | Podanie zostało zmodyfikowane |
| Ścieżki alternatywne | 3a. Lista jest pusta – system wyświetla informację – koniec 6a. Brak CV – system wyświetla powiadomienie 7a. Kandydat nie zatwierdza modyfikacji - koniec |

|  |  |
| --- | --- |
| Przypadek użycia | Anulowanie podania |
| Główny aktor | Kandydat |
| Krótki opis | Kandydat anuluje podanie, jeśli nie zostało jeszcze ocenione przez rekrutera |
| Warunki wstępne | Kandydat jest zalogowany |
| Scenariusz | |  |  |  | | --- | --- | --- | |  | Działanie aktora | Odpowiedź systemu | | 1 | Kandydat wchodzi w historię swoich podań |  | | 2 | Kandydat wybiera opcję anulowania |  | | 3 |  | System wyświetla listę podań, których status umożliwia anulowanie(czyli nie „zaakceptowana”) | | 4 | Kandydat wybiera podanie z listy |  | | 5 |  | System wyświetla szczegóły oferty i formularz podania | | 6 | Kandydat anuluje podanie |  | | 7 |  | System pyta o potwierdzenie | | 8 | Kandydat potwierdza |  | | 9 |  | System wyświetla potwierdzenie dla klienta | |
| Warunki końcowe | Podanie zostało usunięte, pliki załączone do podania skasowane |
| Ścieżki alternatywne | 3a. lista jest pusta – system wyświetla informacje – koniec 8a. Kandydat nie potwierdza – powrót do 5 |

|  |  |
| --- | --- |
| Przypadek użycia | Zamieszczanie oferty |
| Główny aktor | Rekruter |
| Krótki opis | Rekruter zamieszcza ofertę pracy w systemie |
| Warunki wstępne | Rekruter jest zalogowany |
| Scenariusz | |  |  |  | | --- | --- | --- | |  | Działanie aktora | Odpowiedź systemu | | 1 | Rekruter wybiera opcję zamieszczania oferty |  | | 2 |  | System wyświetla formularz zamieszczania oferty | | 3 | Rekruter wypełnia formularz i zatwierdza ofertę |  | | 4 |  | System wyświetla potwierdzenie przyjęcia oferty powrót do 2 | |
| Warunki końcowe | Oferta została przyjęta |
| Ścieżki alternatywne | 3a. Brak wymaganych informacji – system wyświetla komunikat - powrót do 2 3b. oferta tej firmy o tej nazwie już istnieje – komunikat – powrót do 3 |

|  |  |
| --- | --- |
| Przypadek użycia | Modyfikacja oferty |
| Główny aktor | Rekruter |
| Krótki opis | Rekruter modyfikuje ofertę |
| Warunki wstępne | Rekruter jest zalogowany |
| Scenariusz | |  |  |  | | --- | --- | --- | |  | Działanie aktora | Odpowiedź systemu | | 1 | Rekruter wchodzi w listę swoich ofert |  | | 2 |  | System wyświetla listę ofert zamieszczonych przez firmę | | 3 | Rekruter wybiera ofertę do modyfikacji |  | | 4 |  | System wyświetla szczegóły oferty | | 5 | Rekruter wybiera opcję edycji |  | | 6 |  | System wyświetla wypełniony formularz zamieszczania oferty | | 7 | Rekruter wprowadza modyfikacje (nazwa i kraj są niezmienne) |  | | 8 | Rekruter zatwierdza ofertę |  | | 9 |  | System wyświetla potwierdzenie dla użytkownika | |
| Warunki końcowe | Oferta została zmodyfikowana |
| Ścieżki alternatywne | 2a. Lista jest pusta – koniec  8a. Rekruter nie zatwierdza modyfikacji - koniec |

|  |  |
| --- | --- |
| Przypadek użycia | Anulowanie oferty |
| Główny aktor | Rekruter |
| Krótki opis | Rekruter anuluje ofertę pracy w systemie |
| Warunki wstępne | Rekruter jest zalogowany |
| Scenariusz | |  |  |  | | --- | --- | --- | |  | Działanie aktora | Odpowiedź systemu | | 1 | Rekruter wchodzi w listę swoich ofert |  | | 2 |  | System wyświetla listę ofert zamieszczonych przez firmę | | 3 | Rekruter wybiera ofertę do anulowania |  | | 4 |  | System wyświetla szczegóły oferty | | 5 | Rekruter wybiera opcję anulowania |  | | 6 | Rekruter zatwierdza anulowanie oferty |  | | 7 |  | System anuluje oferte oraz wszystkie na nią podania | | 8 |  | System wyświetla potwierdzenie dla użytkownika, powrót do 2 | |
| Warunki końcowe | Oferta została usunięta, podania i pliki z nią skojarzone zostały usunięte |
| Ścieżki alternatywne | 2a. lista jest pusta – koniec 6a. Brak potwierdzenia – powrót do 4 |

|  |  |
| --- | --- |
| Przypadek użycia | Klasyfikacja podań |
| Główny aktor | Rekruter |
| Krótki opis | Rekruter klasyfikuje otrzymane podania |
| Warunki wstępne | Rekruter jest zalogowany |
| Scenariusz | |  |  |  | | --- | --- | --- | |  | Działanie aktora | Odpowiedź systemu | | 1 | Rekruter wchodzi w listę swoich ofert |  | | 2 |  | System wyświetla listę ofert zamieszczonych przez firmę | | 3 | Rekruter wybiera ofertę której podania chce przeglądac |  | | 4 | [Opcjonalne] Rekruter ustawia filtry dla statusu podania(domyślnie nie wyświetla odrzuconych podań) |  | | 5 |  | System wyświetla listę podań na wybrane stanowisko | | 6 | Rekruter wybiera podanie z listy |  | | 7 |  | System wyświetla szczegóły podania, umożliwia pobranie/wyświetlenie załączonych plików. | | 8 | Rekruter wybiera status podania |  | | 9 | Rekruter zatwierdza ustawione statusy |  | | 10 |  | System wyświetla potwierdzenie | |
| Warunki końcowe | Statusy podań zostały zmodyfikowane |
| Ścieżki alternatywne | 2a. lista jest pusta – koniec 6a. Brak potwierdzenia – powrót do 4 9a. Rekruter przechodzi do następnego podania – kroki 6 do 8 mogą być wykonywane w pętli. |

## 3.3 Scenariusze logowania i rejestracji

Scenariusze logowania i rejestracji zostały pominięte z powodu ich jednolitej formy na przestrzeni wielu systemów.

## 3.4 Projekt bazy danych

W tej części przedstawiony został projekt bazy danych. Przedstawione zostały encje oraz diagram związków między nimi.

### 3.4.1 Encje

Rozpoznano następujące encje konieczne w celu realizacji powyższych przypadków użycia:

1. Konto – Zawiera dane konieczne do dokonania autentykacji w tym adres e-mail na który zostało ono założone.
2. Rekruter – Zawiera dane firmy/osoby które będą kojarzone z publikowanymi ofertami
3. Kandydat – Zawiera dane szukającego pracy, które będą kojarzone ze składanymi podaniami.
4. Dziedzina – Zawiera listę unikalnych nazw dziedzin w celu łatwiejszego określenia kategorii zamieszczanej oferty
5. Oferta – Zawiera szczegóły oferty zamieszczonej przez rekrutera
6. Podanie – Zawiera szczegóły podania oraz odnośniki do plików, które zostały załączone.

Związki pomiędzy encjami przedstawia [Rysunek 2 - Diagram związków encji]

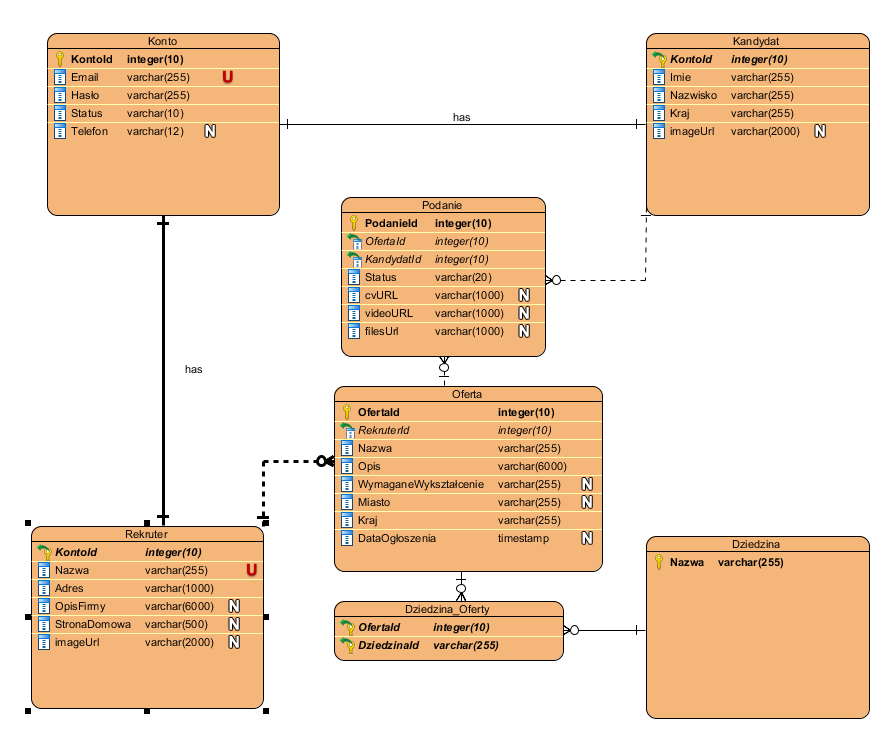
### 3.4.2 Diagram

Poniższy diagram przedstawia związki pomiędzy encjami, gdzie :

 oznacza relację 1 do 1

 oznacza relację wiele do 1

 oznacza relację 1 do wiele



Rysunek 2 - Diagram związków encji

# Rozdział 4 – Implementacja

Niniejszy rozdział przedstawia opis zastosowanych technologii wraz z uzasadnieniem ich wyboru. Przedstawiona również została metodyka pracy, którą zastosowano podczas implementacji.

## 4.1 Kryteria wyboru technologii

Wybór zastosowanych technologii odbył się ze względu na takie kryteria jak :

* Wydajność
* Cena
* Znajomość technologii
* Łatwość użycia

## 4.2 Zastosowane technologie

1. MySQL. Baza danych MySQL ma kilka przewag nad konkurentami. Jest darmowa, łatwa w instalacji oraz użyciu. Na potrzeby prototypu jest ona wystarczająca.

Podczas tworzenia bazy odwoływano się do dokumentacji dostępnej w sieci[5]

1. Hibernate – Framework, który pozwala na łatwiejsze pisanie aplikacji operujących na bazach danych. Jego podstawową funkcją jest translacja klas Javy do tabel w bazie danych. Dodatkowo jest on bardzo łatwy w zastosowaniu dla aplikacji pisanych w języku Java
2. Java została wybrana przede wszystkim ze względu na to, iż w tym języku mam największe doświadczenie. Dodatkowo aplikacja pisana w tym języku może być uruchomiona na serwerze niezależnie od zainstalowanego na nim systemu. Podczas implementacji odwoływano się do dokumentacji dostępnej w sieci[8]
3. Spring został wybrany jako popularny framework do tworzenia systemów webowych w języku Java. Dodatkowo pozostałe zastosowane technologie dobrze z nim współpracują. Podczas implementacji odwoływano się do dokumentacji dostępnej w sieci[6]
4. HTML, CSS, JavaScript– Zdecydowano, że interfejs użytkownika będzie zrealizowany jako strona internetowa. HTML + CSS + JavaScript są zastosowane jako podstawowe języki stosowane przy tworzeniu stron internetowych.
5. Thymeleaf – jest to silnik zajmujący się przetwarzaniem szablonów stron. Bardzo dobrze współpracuje z frameworkiem Spring i umożliwia łatwe przesyłanie danych z i do klienta. Podczas implementacji odwoływano się do dokumentacji dostępnej w sieci[7]
6. Podczas tworzenia stron zastosowano również biblioteki:
   1. Bootstrap
   2. JQuery
   3. Datatables[9]
   4. Select2[10]

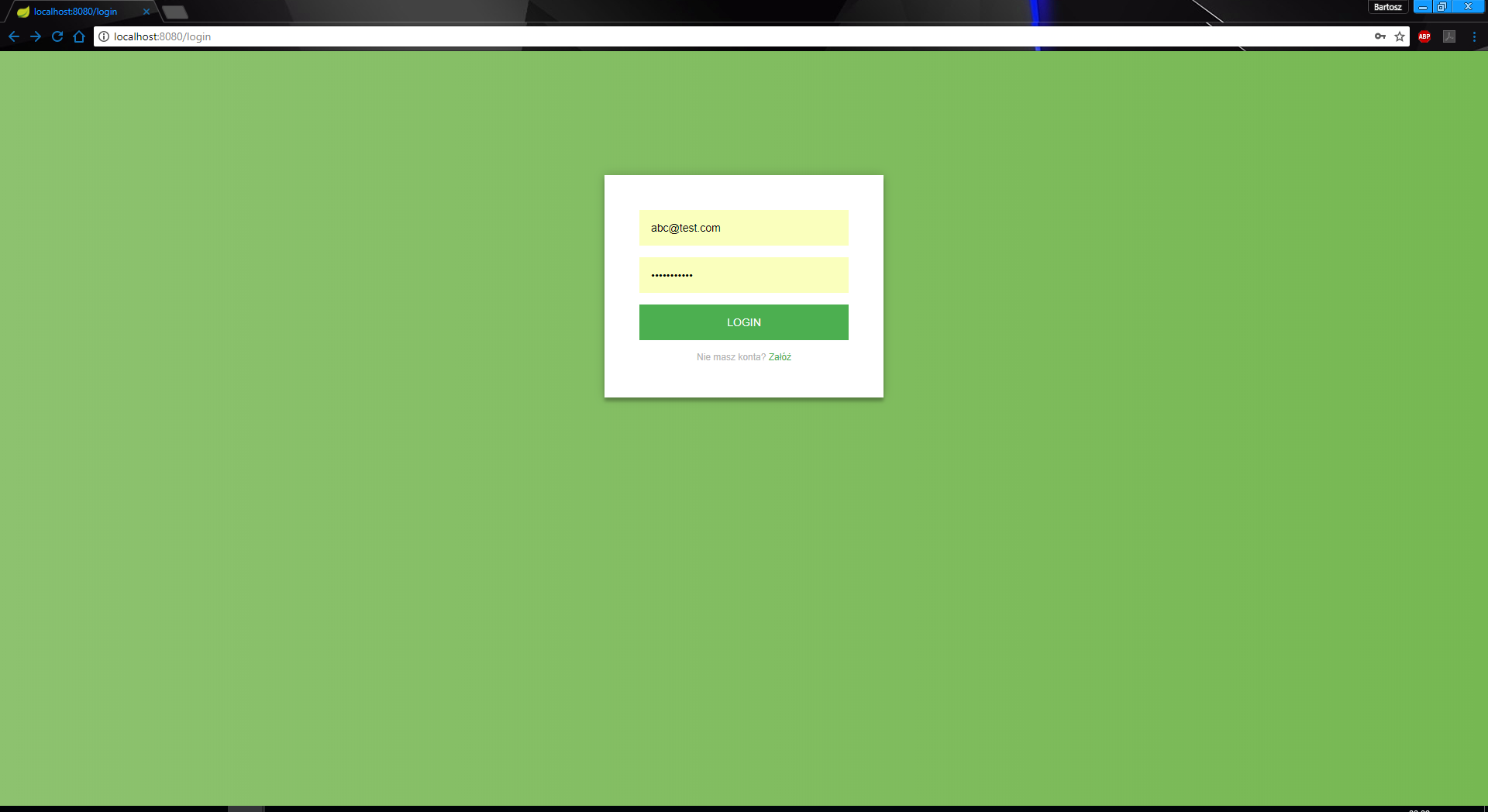
## 4.3 Metodyka wytwarzania oprogramowania

W pracy zdecydowano się na zastosowanie metodyki zwinnej ze względu na możliwość łatwej aktualizacji wymagań. Dodatkowo celem pracy jest stworzenie prototypu, który mógłby być w przyszłości rozwijany, a stopniowy rozwój działającego oprogramowania jest jednym z założeń Manifestu Agile. W całym procesie implementacji dołożono wszelkich starań, aby zachować czystość kodu, a tym samym ułatwić sobie dalszy rozwój systemu[1] W celu zapewnienia jakości kodu zastosowano w nim sprawdzone wzorce takie jak Model-View-Controller, który to wzorzec został częściowo narzucony przez wybrane technologie. W celu osłabienia zależności między klasami zastosowano wstrzykiwanie zależności.

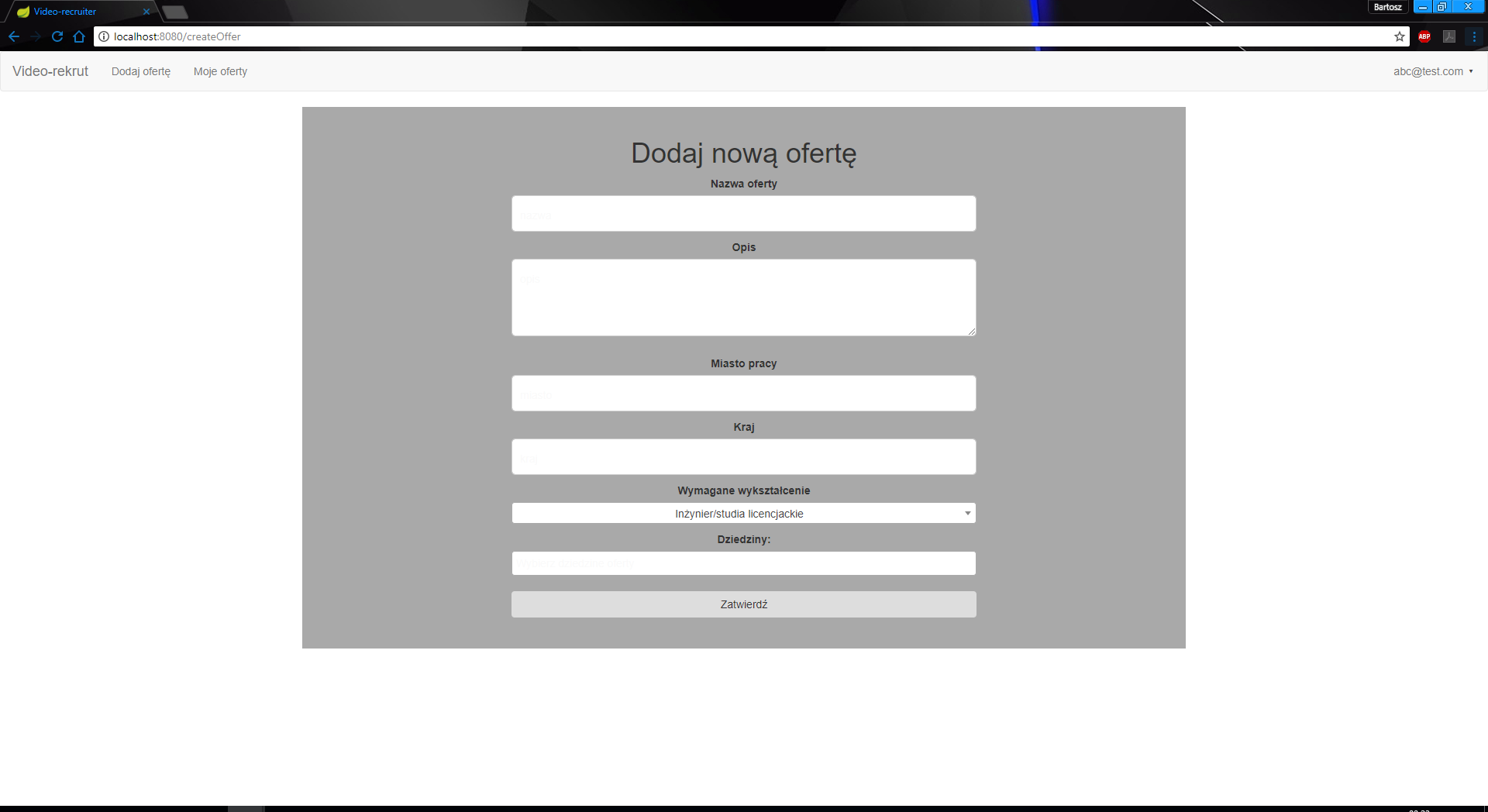
## 4.4 Interfejs użytkownika

W tej części pracy wytworzony został interfejs użytkownika. Funkcje udostępniane przez interfejs różnią w zależności od typu zalogowanego użytkownika. W sytuacji w której użytkownik zalogowany jest jako Rekruter interfejs udostępnia funkcje związane z tworzeniem, edycją i usuwaniem ofert, oraz przeglądaniem i klasyfikacją podań dotyczących opublikowanych z tego konta ofert. W przypadku, gdy użytkownik zalogowany jest jako Kandydat udostępniane są funkcje przeglądania ofert, oraz składania, edycji i anulowania podań. Niezalogowani użytkownicy mają dostęp tylko do strony głównej oraz do stron logowania/rejestracji. Interfejs został utworzony przy użyciu HTML5, CSS3, JavaScript oraz Thymeleaf. Podczas implementacji interfejsu wspierano się podręcznikiem „HTML5 i CSS3 standardy przyszłości”[1] oraz dokumentacją HTML5[11] i CSS3[12] Poniżej przedstawione zostały zrzuty ekranu przedstawiające przykładowe widoki interfejsu[Rysunek 3 – Rysunek] .

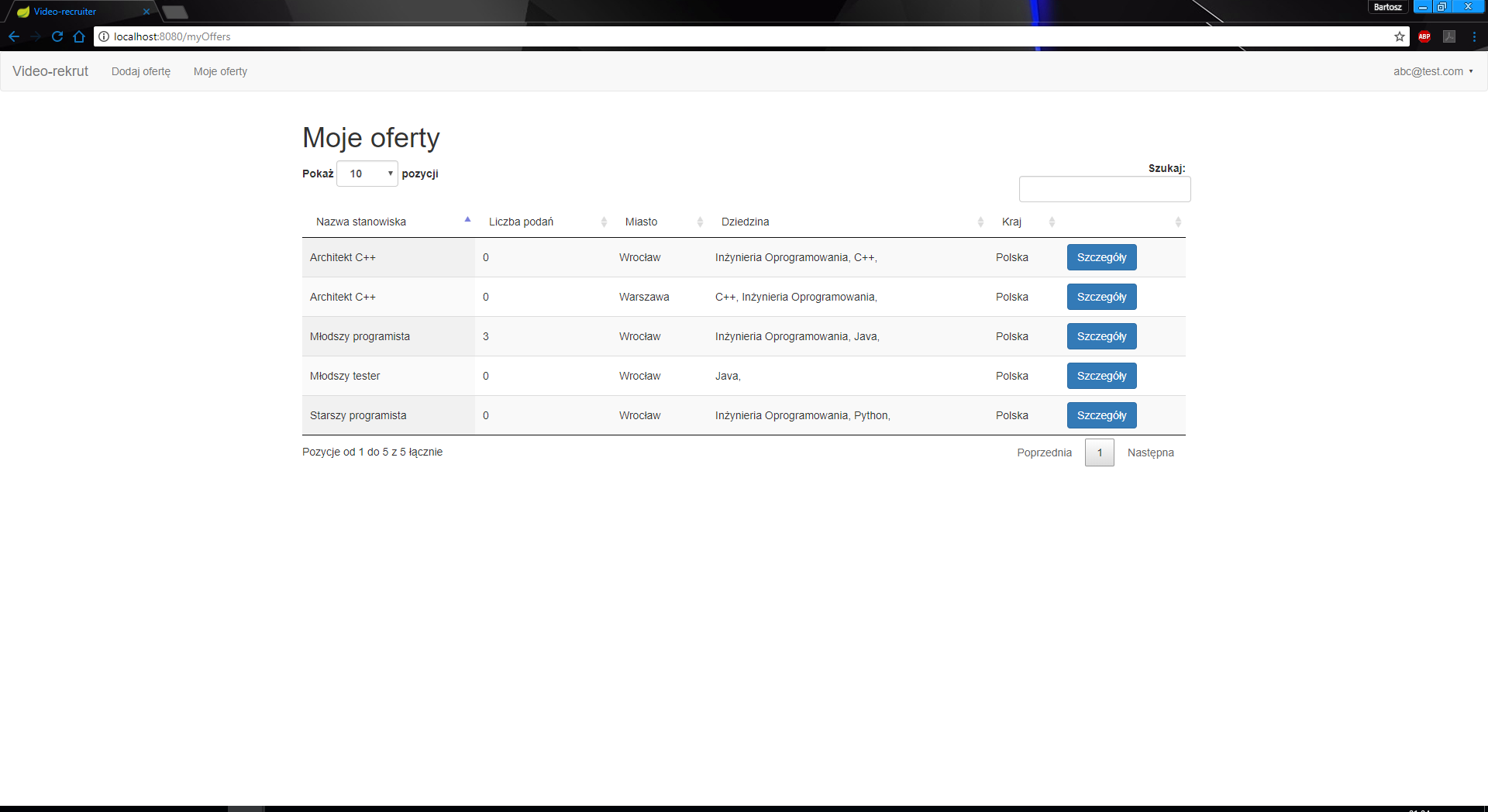
Tworząc strony logowania i rejestracji wzorowano się na darmowym szablonie utworzonym przez [Aigars Silkalns](https://codepen.io/colorlib) dostępnym na stronie: <https://colorlib.com/wp/html5-and-css3-login-forms>



Rysunek 3 - formularz logowania



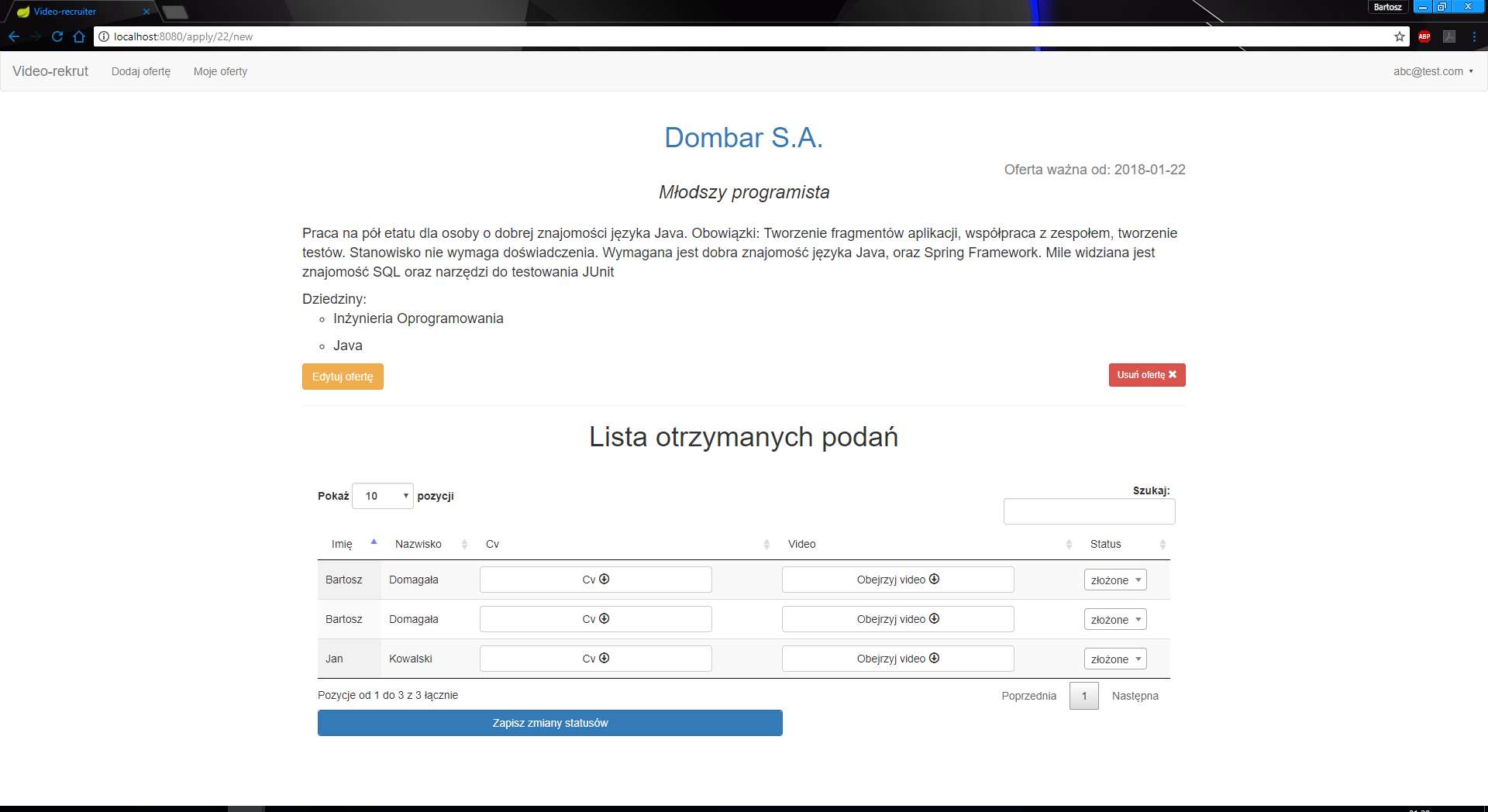
Rysunek 4 - Formularz tworzenia/edycji oferty



Rysunek 5 - Lista opublikowanych ofert

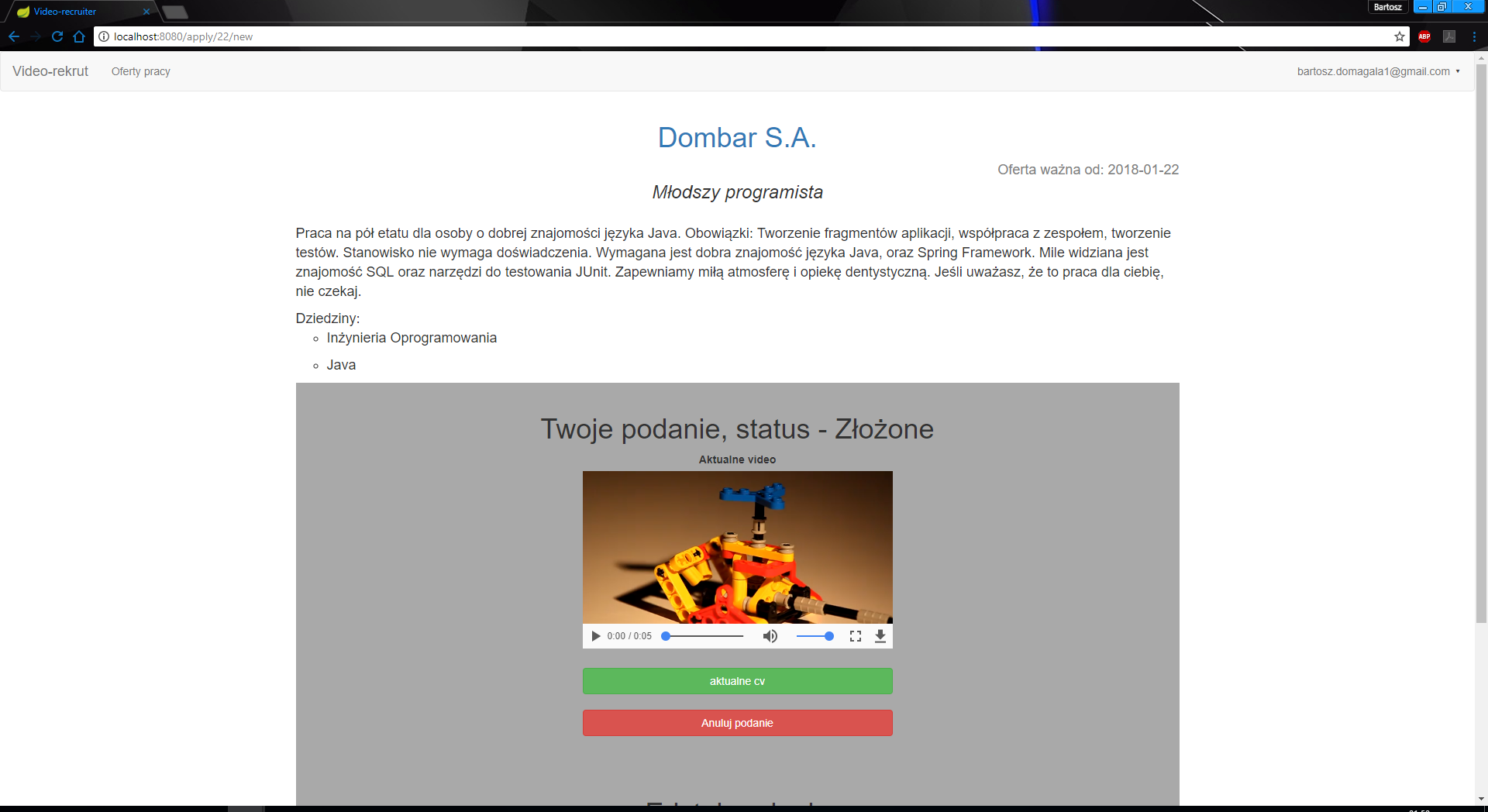
Widok przeglądania ofert pracy jest analogiczny do powyższego. Zmianie ulega jedynie kolumna „Liczba podań” która w przypadku przeglądania ofert wyświetla firmę/osobę która wystawiła dane ogłoszenie.

Podobnie jest w przypadku przeglądania podań przez kandydata. W przypadku przeglądania podań dodana jest dodatkowa kolumna określająca obecny status podania.

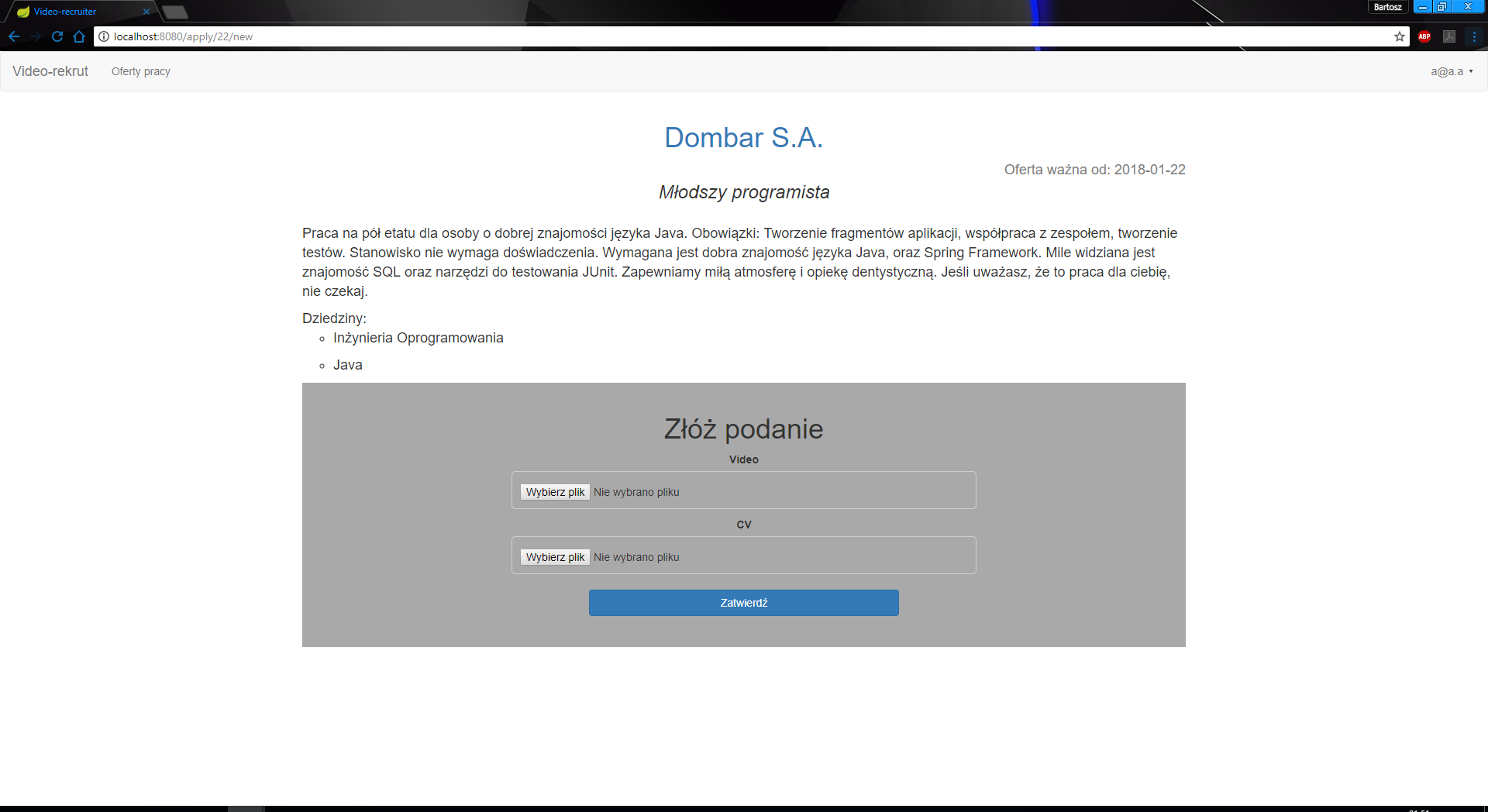


Rysunek 6 - Widok oferty dla publikującego

Jak widać powyżej na stronie oferty jej właściciel może przejść do edycji lub ją anulować. Jeśli na wyświetloną ofertę zostały już złożone podania może również pobrać cv, obejrzeć załączony film i dokonać klasyfikacji podań. W przypadku akceptacji lub odrzucenia podania kandydat zostanie o tym powiadomiony poprzez pocztę elektroniczną.



Rysunek 7 - Widok oferty dla aplikującego



Rysunek 8 - Widok oferty przed złożeniem podania

Dodatkowo utworzono również wiele komunikatów potwierdzających działanie lub informujących o niepowodzeniu. Nie zostały one przedstawione w tej pracy, gdyż nie są one samodzielnymi stronami, a jedynie nieznacznymi modyfikacjami już istniejących widoków.

# Rozdział 5 – Testowanie

Nie jest możliwe udowodnienie, że system będzie działał niezawodnie w każdej sytuacji, ponieważ logicznie niemożliwe jest udowodnienie nieistnienia błędu. Możliwe natomiast jest dokładne przetestowanie systemu, aby odnaleźć i wyeliminować jak największą część błędów (W wersji optymistycznej nie znaleźć żadnego pomimo dobrych testów)

W celu wyeliminowania błędów zastosowano testy na kilku poziomach.

## 7.1 Testy

Napisane zostały testy jednostkowe sprawdzające poprawność działania metod kluczowych dla poprawnego działania systemu. Następnie poprawiono ewentualne błędy.

Napisane zostały testy integracyjne sprawdzające poprawność działania operacji na bazie danych w celu zapewnienia niezawodności funkcji dokonujących trwałych zmian.

Ostatnim etapem testowania były testy manualne sprawdzające poprawność działania systemu zarówno dla głównych scenariuszy przypadków użycia jak i dla ścieżek pobocznych. Następnie system został udostępniony do testów kilku innym osobom niezwiązanym z programowaniem, aby zebrać opinie i uwagi od przeciętnego użytkownika.

## 7.2 Analiza wyników testów i naprawa błędów

Testowanie każdej funkcjonalności było wykonywane równolegle z implementacją dzięki czemu błędy w już zaimplementowanych funkcjonalnościach zostały szybko wyłapane i poprawione.

Podczas testowania pojawiło się wiele sugestii dotyczących układu interfejsu, oraz kilka błędów, które nie zostały pokryte testami automatycznymi. Dzięki tym bieżącym sugestiom możliwe było dostosowanie interfejsu do użytkownika.

# Podsumowanie i wnioski:

System do video-rekrutacji pracowników (robocza nazwa „video-rekrut”) został zaimplementowany i stanowi rozwiązanie rozpatrywanego problemu. Niestety w celu udostępnienia go publicznie wymagane jest jeszcze wiele pracy. Przede wszystkim konieczne jest zabezpieczenie systemu przed atakami mającymi na celu uszkodzenie systemu czy kradzież danych użytkowników. Dodatkowo w celu osiągnięcia konkurencyjności na rynku system musi posiadać więcej funkcji, które mogą być atrakcyjne dla użytkowników. Np. Rozbudowany profil Kandydata, który umożliwiłby znalezienie takiej osoby przez headhunterów.

Podsumowując przed zaproponowanym systemem stoją dwie potencjalne ścieżki rozwoju. Pierwszą z nich oraz mniej prawdopodobną jest znaczące rozbudowanie funkcjonalności i próba konkurowania na rynku z już ustalonymi portalami takimi jak LinkedIn oraz GoldenLine.

Drugą możliwością jest przejęcie systemu przez firmę i zastosowanie go jako wewnętrznego portalu do rekrutacji. Druga opcja była rozpatrywana już na początkowym etapie pracy, ale firma z którą prowadzono rozmowy nie wykazała zainteresowania.

W obu przypadkach konieczne będzie zapewnienie bezpieczeństwa danych i rozwój funkcjonalności systemu przed jego ewentualnym uruchomieniem.

## Aspekt prawny

Na chwilę obecną główną przeszkodą uniemożliwiającą udostępnienie systemu do użytku publicznego, jest nie spełnianie przez niego wymogów ustawy o ochronie danych osobowych państwa polskiego.

Zgodnie z prawem system informatyczny, który zbiera i przechowuje dane osobowe musi spełniać określone warunki:

* Jest zgodne z art. 23 ust 1 pkt 1-5 ustawy o ochronie danych osobowych.
* Obowiązek rejestracji zbierania danych osobowych został dopełniony
* Zastosowane zostały odpowiednie zabezpieczenia
* Dopełniono obowiązku informacyjnego ustanowionego w art. 24 ust. 1 oraz art. 25 ust. 1 ustawy o ochronie danych osobowych
* dokładał szczególnej staranności w celu ochrony interesów osób, których dane dotyczą,
* respektował prawa osób, których dane dotyczą

Spełnienie tych wymagań jest planowane w ramach ewentualnego dalszego rozwoju projektu.

W kwestii samego bezpieczeństwa zebranych danych wymagane są takie elementy zabezpieczeń jak zastosowanie bezpiecznego połączenia, czy szyfrowanie bazy danych.

# Literatura

1. Brian P. Hogan: HTML5 i CSS3 standardy przyszłości, Gliwice, Helion, 2011
2. Robert C. Martin. Clean Code: A Handbook of Agile Software Craftsmanship (1 ed.). Prentice Hall PTR, Upper Saddle River, NJ, USA. 2008.

1. <https://www.michalwolski.pl/2016/10/user-stories-historyjki-uzytkownika/>
2. https://www.visual-paradigm.com/tutorials/
3. https://dev.mysql.com/doc/
4. https://spring.io
5. <http://www.thymeleaf.org/>
6. <https://docs.oracle.com/javase/8/docs/>
7. <https://datatables.net/>
8. https://select2.org/
9. https://www.w3schools.com/TAgs/default.asp
10. <https://www.w3schools.com/cssref/default.asp>

# Załączniki

1. Płyta DVD z kodem źródłowym utworzonej aplikacji.