UNIWERSYTET GDAŃSKI
Wydział Matematyki, Fizyki i Informatyki

Marcin Dawidowski
nr albumu: 231010

Let's Bid It – portal aukcyjny

Praca licencjacka na kierunku:
INFORMATYKA
Promotor:
dr Włodzimierz Bzyl

Gdańsk 2017

Streszczenie

W pracy przedstawiono rozwojową wersję aplikacji webowej „Let's Bid It" do tworzenia i publikowania aukcji oraz ogłoszeń.

W aplikacji zaimplementowano ich kategoryzację, wyszukiwanie, a także tworzenie kategorii w hierarchii.

Zaprojektwano widok strony głównej, który wyświetla losowe aukcje znajdujące się w portalu oraz panele administracyjny i użytkownika, które są widoczne po zalogowaniu się na konkretny typ konta. Każda z aukcji wyświetlana jest w kolejności rosnącej według czasu zakończenia.

Do tworzenia aplikacji wykorzystano takie technologie jak Ruby on Rails, Twitter Bootstrap, Plataformatec Devise, reCaptcha, CKEditor, jQuery Turbolinks, Chartkick, ActsAsTree.

Projekt wdrożony został na platformie `heroku.com` i jest dostępny pod adresem: `https://ror-aukcja.herokuapp.com/`.

Słowa kluczowe

ruby on rails, gem, heroku, auction, online shopping, bid

Spis treści

Wprowadzenie

Chęć stworzenia czegoś od podstaw oraz własne doświadczenie w korzystaniu z portali aukcyjnych sprawiło, że dla mnie jako programisty stworzenie takiej aplikacji, byłoby niezwykle satysfakcjonujące i pozwoliłoby rozwinąć umiejętności. Ponadto stworzenie takiego portalu może ułatwić mi w przyszłości implementacje różnego rodzaju sklepów internetowych i innych serwisów związanych z handlem przez Internet.

Mając więc gotowy pomysł na projekt pozostało mi jedynie wybrać odpowiednią technologię, która pomoże mi go zrealizować. W trakcie studiów poznałem kilka różnych języków programowania i frameworków, które sprawdziłyby się w tego typu projekcie. Po rozważeniu wszystkich za i przeciw, mój wybór padł na język Ruby, wspierany przez framework Rails, który udało mi się poznać już wcześniej. Największą zaletą wybranej przeze mnie technologi jest przede wszystkim fakt, iż Ruby on Rails posiada szeroką gamę dostępnych rozwiązań w postaci gemów, które w dużym stopniu ułatwiają pracę nad projektem. Kolejną kwestią jest fakt, że technologia ta wykorzystuje architekturę MVC, co znacznie ułatwia pracę. Poza tym Ruby on Rails świetnie współpracuje z platformami takimi jak Heroku, czy DigitalOcean.

Aplikacja Let's Bid It jest wersją rozwojową portalu aukcyjnego, dlatego też nie posiada jeszcze wszystkich funkcjonalności, które pozwoliłyby w pełni wykorzystać potencjał serwisu.

Spośród rzeczy, które udało się zrealizować, mogę wyróżnić przede wszystkim obsługę aukcji. Ponadto zaimplementowano ich kategoryzację przy pomocy gemu Act as tree. Kolejną rzeczą jest wdrożenie zaawansowanego edytora tekstowego CKEditor, dzięki któremu dodając aukcję, użytkownik może swobodnie manipulować jej opisem. Dodatkowo zaimplementowałem zabezpieczenie reCaptcha przy logowaniu i rejestracji, a także statystyki strony, które dostępne są dla administratorów. Ważnym elementem Let's Bid It jest również grafika dołączana do każdej aukcji. W związku z tym, że aplikacja znajduje się na platformie Heroku, głównym problemem było przetrzymywanie plików graficznych, które na Heroku usuwane są automatycznie.

Rozwiązaniem tego problemu okazało się skorzystanie z serwisu Cloudinary, który pozwala bezpłatnie przechowywać pliki w chmurze, a co najważniejsze, możliwe jest skonfigurowanie go tak, by współpracował z aplikacją napisaną w Ruby on Rails, w czym pomocny okazał się gem cloudinary.

Najważniejszą rzeczą, której nie udało się zrealizować jest wdrożenie systemu mającego na celu symulacje płatności internetowych, a także zaawansowana wyszukiwarka opierająca się na silniku Elasticsearch.

ROZDZIAŁ 1

Let's Bid It w praktyce

1.1. Porównanie z dostępnymi rozwiązaniami

1.1.1. Allegro

Jest to największa, a także najpopularniejsza platforma transakcyjna on-line w Polsce, oferująca setki produktów w niezwykle konkurencyjnych cenach. Serwis udostępnia możliwość zakupu natychmiastowego, licytacji, a także wystawiania ogłoszeń. Typ danej aukcji, zależy tylko i wyłącznie od jej właściciela.

Jeśli chodzi o wystawianie aukcji, to proces wygląda podobnie do Let's Bid It. Tak jak w mojej aplikacji użytkownik może skorzystać z zaawansowanego edytora tekstowego, dodaje zdjęcia i przyporządkowuje ją do konkretnej kategorii. Główną różnicą jest fakt, że Allegro pozwala na wystawienie przedmiotu z opcją "Kup Teraz", czego Let's Bid It nie ma.

1.1.2. OLX

Serwis ten jest jednym z najpopularniejszych portali ogłoszeniowych na świecie. Użytkownicy mogą udostępniać lokalne ogłoszenia dotyczące sprzedaży oraz usług.

Główną różnicą pomiędzy OLX, a Let's Bid It jest fakt, że wymieniony wyżej portal nie udostępnia funkcji licytacji towaru, natomiast moja aplikacja posiada ją, a także tak jak OLX daje użytkownikowi prawo wystawienia towaru, czy też usługi jako zwykłe ogłoszenie. Ponadto stworzony przeze mnie portal aukcyjny, dzięki zaawansowanemu edytorowi tekstowemu pozwala stworzyć użytkownikom dużo bardziej oryginalne opisy, co nie jest możliwe przy korzystaniu z wyżej wymienionego portalu.

1.2. Możliwości zastosowania praktycznego

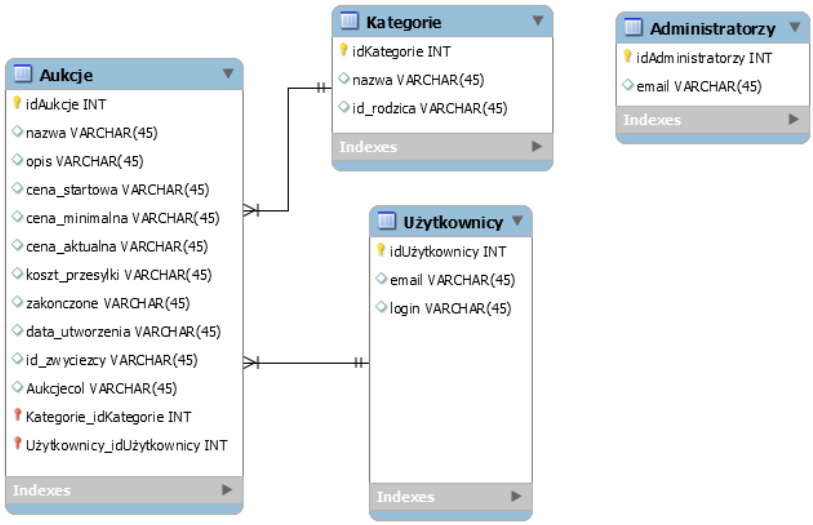
Głównym założeniem serwisu Let's Bid It jest przede wszystkim portal aukcyjny i jest to główna możliwość zastosowania napisanej przeze mnie aplikacji w praktyce. Poza tym portal ten użyty może zostać również do innych celów:

- Portal ogłoszeniowy - jest jedną z możliwości wykorzystania Let's Bid It. Potrzeby takiej aplikacji zaspokajają praktycznie wszystkie wdrożone w mojej aplikacji funkcjonalności, takie jak kategoryzacja, a także prosta obsługa dodanych ogłoszeń. Ponadto użytkownik takiego serwisu będzie miał do dyspozycji bardzo prosty interfejs ułatwiający poruszanie się po stronie, a także zaawansowany edytor, który pozwoli na to, by jego ogłoszenie wyglądało wyjątkowo.
- Sklep internetowy - jest to kolejna możliwa opcją zastosowania mojej aplikacji w praktyce, co można osiągnąć przy niedużym nakładzie pracy. Tak jak przy portalu ogłoszeniowym, dostępne są przede wszystkim kategoryzacja, a także obsługa dodawanych aukcji oraz zaawansowane narzędzia do zarządzania wyglądem aukcji.

ROZDZIAŁ 2

Projekt i analiza

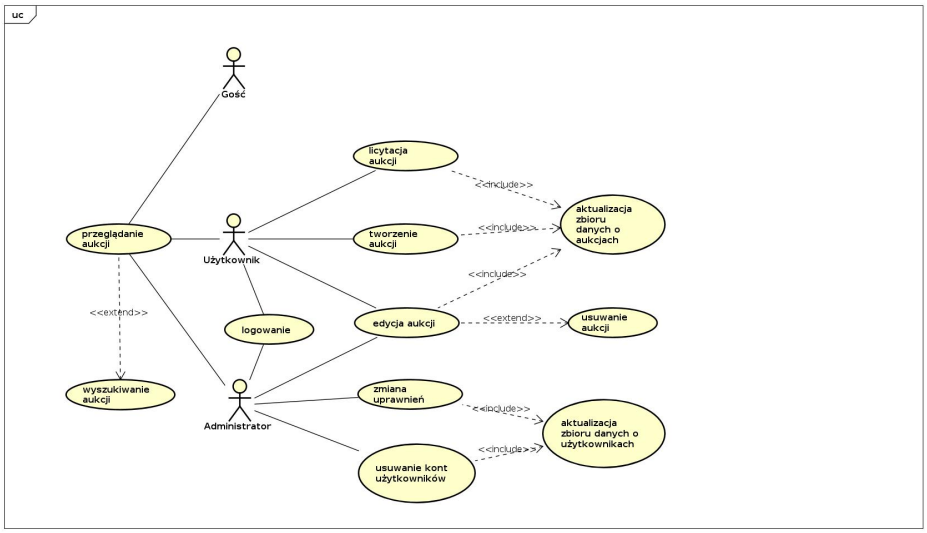
2.1. Diagram ERD



Rysunek 2.1. Diagram związków encji

Źródło: Opracowanie własne za pomocą MySQL Workbench

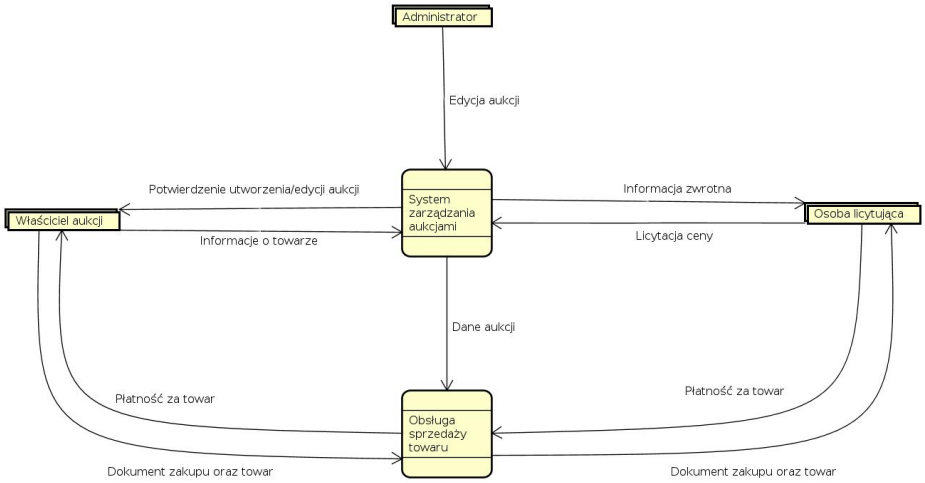
2.2. Aktorzy i przypadki użycia



Rysunek 2.2. Diagram Przypadków Użycia.

Źródło: Opracowanie własne za pomocą programu Astah [?].

2.3. Diagram przepływu danych



Rysunek 2.3. Diagram Przepływu Danych.

Źródło: Opracowanie własne za pomocą programu Astah [?].

ROZDZIAŁ 3

Implementacja

W celu stworzenia aplikacji, która ma w odpowiedni sposób realizować postawione przed nią założenia, potrzebne są odpowiednio dobrane technologie oraz narzędzia, dzięki którym uda się spełnić postawione przed nią wymagania. Dobór odpowiedniej technologii, a także frameworków jest kluczowy do skutecznej implementacji aplikacji.

3.1. Realizacja projektu

3.2. Ruby on Rails

Do stworzenia projektu wykorzystana została technologia Ruby on Rails bazująca na języku Ruby. W aplikacji wykorzystano Ruby w wersji 2.3, a także framework Rails w wersji 5.0.1. Całość stworzona została przy pomocy architektury MVC (ang. Model-View-Controller).

3.3. Twitter Bootstrap

Widoki utworzone zostały przy pomocy frameworka Twitter Bootstrap, który pozwala na proste tworzenie graficznego interfejsu stron internetowych z wykorzystaniem gotowych rozwiązań bazujących na językach HTML i CSS. Główną zaletą tego frameworka jest responsywność, czyli zapewnienie dobrego wyświetlania stron WWW na różnego typu urządzeniach.

3.4. Pozostałe rozwiązania użyte przy realizacji projektu

CKEditor - jest to wizualny edytor tekstowy języka HTML, który umożliwia użytkownikowi na wybranie między innymi konkretnej czcionki, jej rozmiaru, koloru liter, czy też

ich stylu. Poza tym użytkownik może również ustawić wyrównanie tekstu, a także możliwe jest wstawienie listy, tabeli, odnośników, a także obrazów.

ActsAsTree - gem ten wykorzystany został w celu stworzenia drzewa kategorii, które ułatwia w sposób zdecydowany nawigację po stronie, a także sprawia, że dane na stronie są uporządkowane. Sama implementacja przebiega w środowisku technologii Ruby on Rails, więc nie wymaga bezpośredniej instalacji.

reCaptcha - zabezpieczenie dzięki któremu możliwość wysyłania danych będą mieli jedynie sprawdzeni użytkonicy.

Zakończenie

W trakcie pracy nad projektem udało mi się zrealizować praktycznie wszystkie jego założenia. Praca ta pozwoliła mi nabyć nowe doświadczenia, a także poszerzyć posiadane umiejętności. Podczas realizacji aplikacji udało mi się wykorzystać niemal wszystkie umiejętności, które nabyłem w trakcie odbywania studiów.

DODATEK A

Płyta CD zawierająca kod projektu

Bibliografia

[1] David A. Black. Ruby. Przewodnik programisty wyd. II. HELION S.A. 2015.

[2] John Elder. Ruby on Rails - Tworzenie aplikacji WWW Codemy.com 2016.

[3] Oficjalna dokumentacja - Gem Devise
https://github.com/plataformatec/devise/ (dostęp 5.03.2017)

[4] Oficjalna dokumentacja - Gem ActsAsTree
https://github.com/amerine/acts_as_tree (dostęp 8.04.2017)

[5] Oficjalna dokumentacja - Gem reCAPTCHA
https://github.com/ambethia/recaptcha/ (dostęp 10.04.2017)

[6] Oficjalna dokumentacja - Gem CKEditor for Rails
https://github.com/galetahub/ckeditor/ (dostęp 10.04.2017)

[7] Oficjalna dokumentacja - Gem Chartkick
https://github.com/ankane/chartkick/ (dostęp 28.04.2017)

Oświadczenie

Ja, niżej podpisany(a) oświadczam, iż przedłożona praca dyplomowa została wykonana przeze mnie samodzielnie, nie narusza praw autorskich, interesów prawnych i materialnych innych osób.

.....	.....
data	podpis