



**Politechnika Krakowska
im. Tadeusza Kościuszki**

Wydział Fizyki, Matematyki i Informatyki



Kamil Szczepański

Nr albumu: 114012

**Aplikacja internetowa wspomagająca
zakup/sprzedaż pojazdu wraz z modułem
publicystycznym**

**Web application that helps to buy/sell vehicle with
additional journalistic module**

**Praca inżynierska
na kierunku INFORMATYKA**

Praca wykonana pod kierunkiem
dra inż. Jerzego Białasa

Uzgodniona ocena:

.....
podpisy promotora i recenzenta

Kraków 2018

Spis treści

1. Wstęp	1
1.1. Wprowadzenie do tematu	2
1.2. Cele i założenia	2
1.3. Przegląd zawartości pracy	4
2.2. Identyfikacja aktorów	8
2.3. Wymagania funkcjonalne	9
2.4. Wymagania нефункционалне	12
2.5. Diagramy języka UML	12
2.6. Struktura bazy danych	15
3. Implementacja i wdrożenie systemu	17
3.1. Przegląd popularnych technologii do tworzenia aplikacji internetowych	17
3.2. Wybór technologii i narzędzi do realizacji systemu	18
3.2.1. Front-end	18
3.2.2. Back-end	19
3.3. Architektura systemu	20
3.4. Elementy składowe aplikacji	20
3.5. Szczegóły techniczne	32
3.6. Aspekty bezpieczeństwa	33
4. Testowanie	35
4.1. Testy interfejsu	35
4.2. Testy funkcjonalności	36
5. Rozwój aplikacji	38
5.1. Forum	38
5.2. Obsługa różnych rodzajów ogłoszeń	38
5.3. Dostosowanie do innych przeglądarek	38
5.4. Inne	39
6. Podsumowanie	40
7. Bibliografia	41

1. Wstęp

Rozdział ten poświęcony jest ogólnemu przedstawieniu stworzonej aplikacji. Opisane tu jest dlaczego autor zdecydował się na stworzenie systemu do handlu samochodami oraz pokaże w nim cele i założenia jakie wyznaczył sobie przed realizacją projektu.

1.1. Wprowadzenie do tematu

W dobie rozwoju oraz coraz większego dostępu do sieci internetowej, aplikacje funkcjonujące w niej stają się coraz bardziej rozbudowane, powstają nowe technologie upraszczające tworzenie szybkich i funkcjonalnych serwisów, a te coraz mocniej wypierają programy desktopowe. Nieco inaczej wygląda to w przypadku smartfonów, jednak i tutaj możemy zobaczyć, że aplikacje *webowe* coraz częściej można uruchomić w wersji mobilnej. Obecne narzędzia pozwalają na łatwe tworzenie responsywnych stron internetowych, czyli takich które dopasowują się do wielkości ekranu na jakim zostają wyświetlone. Nie tylko ułatwia to ich obsługę, ale również upraszcza proces “budowania” takich systemów.

W związku z tymi cechami autor postanowił podjąć się zaprojektowania i implementacji aplikacji internetowej, która będzie łączyła prostotę użytkowania z pełnią dostosowywania się do ekranu urządzenia na którym zostanie ona uruchomiona.

Tematyką, której dotyczy aplikacja zostały zaś samochody. Wzięło się to z zainteresowania twórcy tzw. “czterema kółkami” oraz z doświadczenia jakie posiada w procesie wystawiania ogłoszeń w kategorii motoryzacji, a także wyszukiwania pojazdów spełniających rozmaite kryteria, dla znajomych i rodziny. Wielokrotnie zdarzało się mu asystować podczas dodawania ogłoszeń na różnych serwisach ponieważ ludzie starsi nie zawsze radzili sobie z kolejnymi krokami narzuconymi przez te aplikacje.

W związku z tym postanowiono na jak największą prostotę w użytkowaniu tego serwisu. Starano się aby dostępne było tylko to co jest potrzebne do przeglądania, wyszukiwania i sprzedawania samochodów oraz kontaktu ze sprzedającym, czy “obserwowania” interesujących użytkownika ogłoszeń.

Nie jest to jednak tylko aplikacja z ogłoszeniami. Zamysłem było połączenie takiego serwisu z serwisem publicystycznym. Te dwa elementy mogłyby funkcjonować niezależnie

od siebie, chodziło o umożliwienie użytkownikom łatwego poruszania się pomiędzy nimi np. w temacie określonego modelu auta.

Nazwą tej aplikacji jest skrót “RCS”, który oznacza *Random Car Site*. Wymyślony został przy tworzeniu pewnego projektu, który to mógłby zostać określony mianem prototypu tytułowej aplikacji.

1.2. Cele i założenia

Głównymi celami przyświecającymi tworzeniu tej aplikacji była łatwość w użytkowaniu, nie zależnie od umiejętności posiadanych przez klientów, a także integracja dwóch podstawowych modułów - ogłoszeniowego i publicystycznego.

Pierwszy z modułów miał umożliwiać szybkie wystawianie ogłoszeń samochodów. Dodajemy podstawowe szczegóły, krótki opis, przynajmniej jedno zdjęcie i gotowe. Następnie inni użytkownicy, poszukujący dla siebie idealnego pojazdu mieli dostać opcje pozwalające na łatwe wyszukanie odpowiednich ofert z całej ich bazy. Odbywać się to miało zarówno za pomocą wyboru kolejnych cech charakterystycznych z listy, ale również przy pomocy słów kluczowych. Często wpisanie kilku wyrazów może sprawić, że odfiltrowanie interesujących wyników zajmie mniej czasu i będzie prostsze.

Oczywiście użytkownicy potrzebowaliby sposobu komunikacji ze sobą. Nie zalogowani klienci tzw. “goście” mieliby do dyspozycji numery telefonów do sprzedających oraz ewentualnie inne dane kontaktowe podawane w opisach aukcji. Dla zalogowanych zaplanowano dodatkową formę komunikacji osadzoną wewnątrz aplikacji. Chodzi o możliwość wysyłania wiadomości tekstowych między kupującymi, a sprzedającymi.

Dodatkową bardzo przydatną opcją w tego typu aplikacjach jest możliwość dodawania ogłoszeń do obserwowanych bądź ulubionych. Upraszcza to odnajdywanie ofert którymi użytkownicy są szczególnie zainteresowani. Zdecydowano się więc uwzględnić tę opcję w tworzonym systemie. Ważnym elementem jest także możliwość edycji oraz oczywiście usuwania dodanych ogłoszeń. Dlatego i tych funkcji nie można było pominąć.

Odnosnie drugiego modułu to tworzeniem zawartych w nim treści zajmować mieliby się specjaliści użytkownicy. Tylko oni mieliby dostęp do kreatora nowych artykułów, a także do zarządzania nimi. Uwzględniono możliwość zostawiania komentarzy pod tekstami - oczywiście tylko dla zalogowanych użytkowników. Z kolei wszyscy chętni mogliby

przeglądać artykuły oraz wyszukiwać te, które ich interesują np. dotyczą ulubionego samochodu.

Na czym więc miała polegać integracja tych dwóch modułów? Otóż w założeniu zaledwie jednym kliknięciem można by się przenosić między artykułami i ogłoszeniami dotyczącymi tego samego modelu auta. Zamysł był taki, że ktoś zainteresowany danym pojazdem np. po przeczytaniu jego recenzji, mógłby od razu przejść do ogłoszeń tego właśnie samochodu. Działałoby to również w drugą stronę. Przykładowo z poziomu ogłoszenia auta *xyz* użytkownik mógłby przenieść się do porównania go z autem *abc*, gdzie mogłoby się okazać, że to drugie bardziej odpowiada zainteresowanemu i natychmiast, tzn. kolejnym kliknięciem, mógłby znaleźć się w sekcji ogłoszeń modelu *abc*.

1.3. Przegląd zawartości pracy

Kolejnymi rozdziałami po wstępie są:

Projekt systemu - w rozdziale tym opisano, jak wyglądają konkurencyjne rozwiązania w temacie ogłoszeń motoryzacyjnych. Przedstawiono aktorów, czyli typy użytkowników jacy mogą korzystać z tworzonej aplikacji wraz z wypisaniem funkcji dostępnych dla określonych grup. Opisano także wymagania funkcjonalne i нефункционалне systemu. Zaprezentowano przykładowe diagramy UML - diagramy przypadków użycia. Na końcu z kolei pokazano schemat używanej bazy danych.

Implementacja i wdrożenie systemu - rozdział zawierający opis typowych technologii używanych do tworzenia aplikacji internetowych oraz tych, które wybrano do implementacji projektu. Przedstawiona zostanie architektura systemu. Następnie pokazane zostaną wszystkie składowe elementy wraz z opisem ich funkcjonowania. Na koniec zajmę się ukazaniem zastosowanych zabezpieczeń.

Testowanie - rozdział ten poświęcony jest na opis testów, jakim podlegał projekt po ukończeniu, jak i w trakcie tworzenia. Zarysowany jest tu podział na testy interfejsu oraz działania poszczególnych modułów i funkcjonalności.

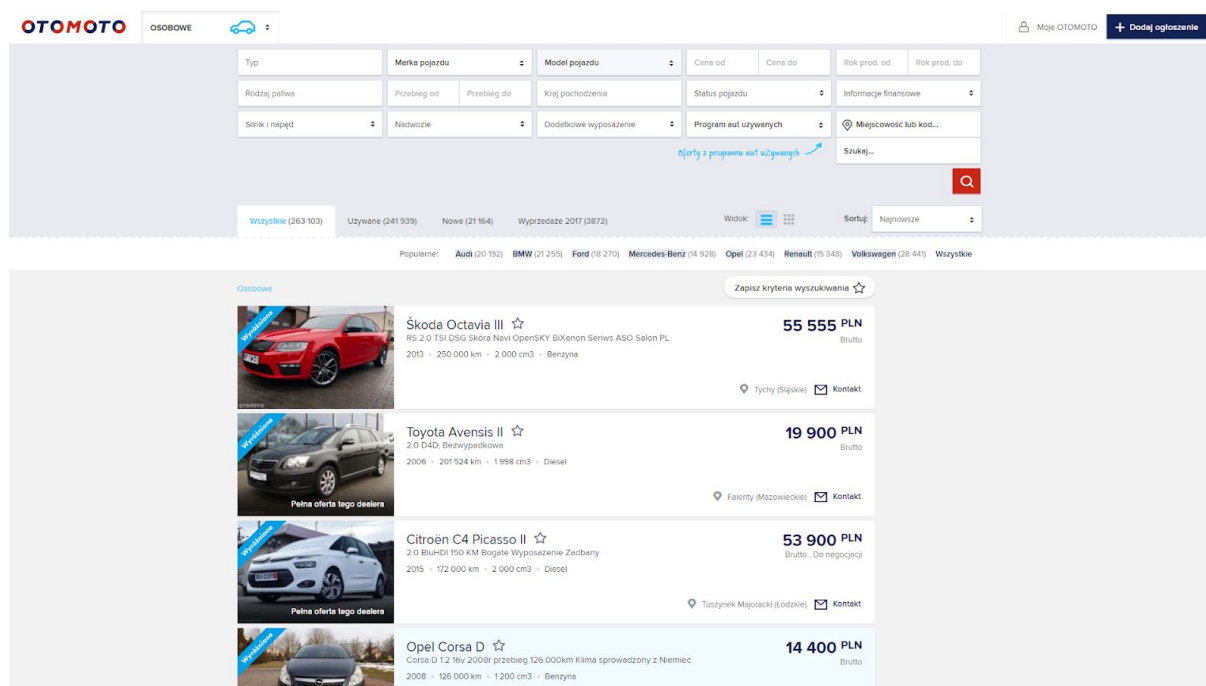
Rozwój aplikacji - Rozdział poświęcony opisowi pomysłów na przyszły rozwój aplikacji. Prezentacja dodatkowych modułów oraz technicznych ulepszeń do sytemu, jakie mogłyby wzbogacić i ulepszyć korzystanie z niego.

Na końcu znajdują się zaś podsumowanie i bibliografia.

2. Projekt systemu

2.1. Rozpoznanie istniejących rozwiązań

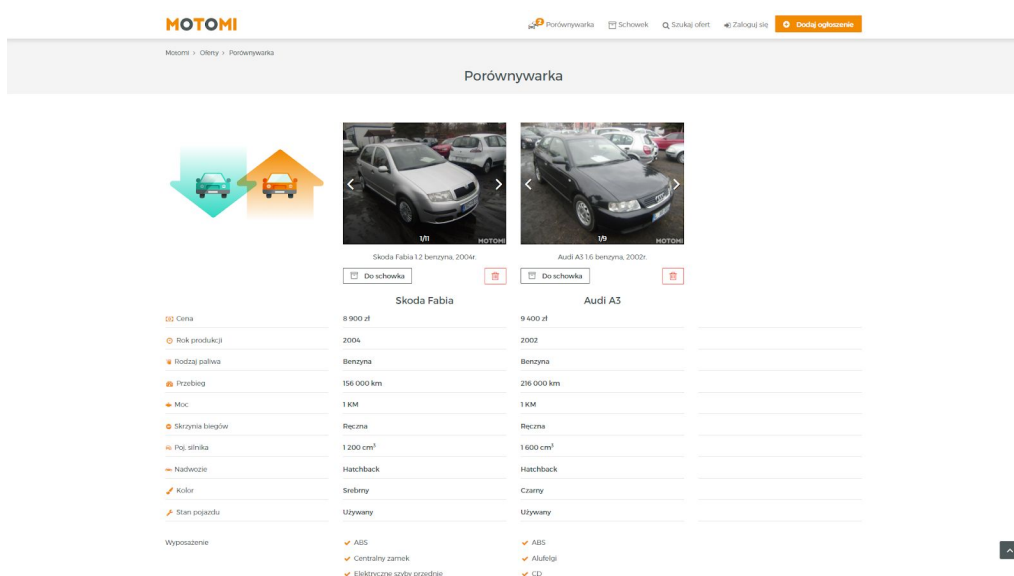
OTOMOTO.PL - jeden z najpopularniejszych, jak nie najpopularniejszy serwis do wystawiania i przeglądania ogłoszeń. Oprócz samochodów osobowych można tam znaleźć motocykle, samochody ciężarowe, maszyny rolnicze i budowlane oraz części do rozmaitych pojazdów. Posiada szeroki zakres opcji umożliwiających odfiltrowanie interesujących użytkownika ogłoszeń. Umożliwia kontakt pomiędzy kupującym i sprzedającym poprzez wbudowany system przysyłania wiadomości. Za dodatkową opłatą można wypromować tam swoje ogłoszenie. Aplikacja ta umożliwia również dwa rodzaje kont użytkowników - indywidualne oraz dealera.



Rysunek 2.1: zrzut przedstawiający wygląd listy ogłoszeń oraz formularza opcji filtrowania w aplikacji OTOMOTO

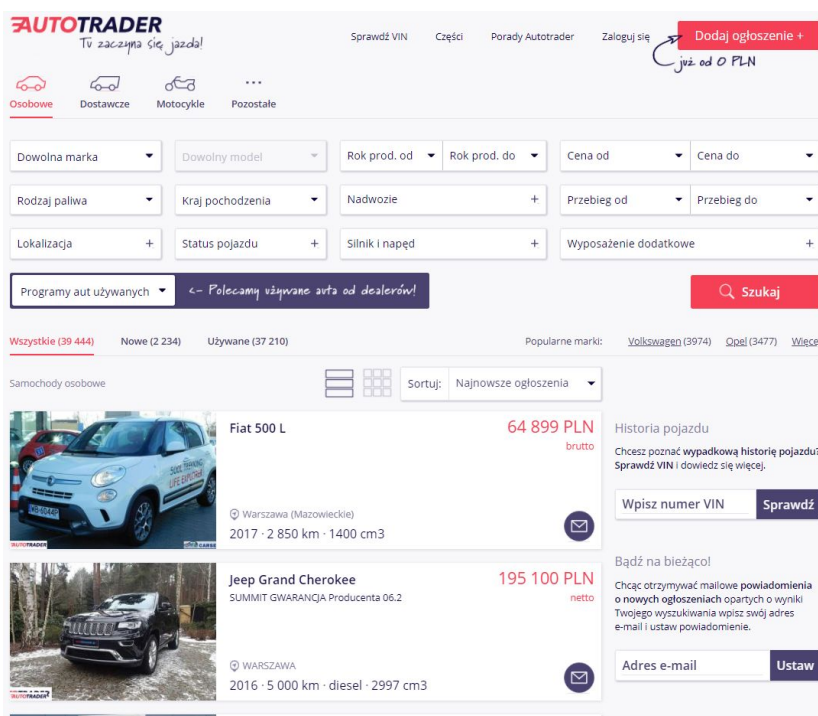
MOTOMI.PL - serwis zajmujący się wyłącznie ogłoszeniami samochodów osobowych. Podobnie jak inne aplikacje tego typu pozwala na filtrowanie interesujących ofert, tekstowy kontakt ze sprzedającymi oraz możliwość dodawania ogłoszeń do schowka. Z wyróżniających funkcjonalności serwis ten posiada porównywarke danych technicznych do czterech pojazdów na raz oraz współpracuje z AutoCentrum, dzięki czemu w ogłoszeniach można znaleźć opinie i raporty ekspertów odnośnie danego modelu. Dodatkowo istnieje tutaj

MOTOMI magazyn, czyli serwis z artykułami, poradami i nowościami ze świata motoryzacji. Nie jest on jednak bezpośrednio związany z wystawianymi ogłoszeniami.



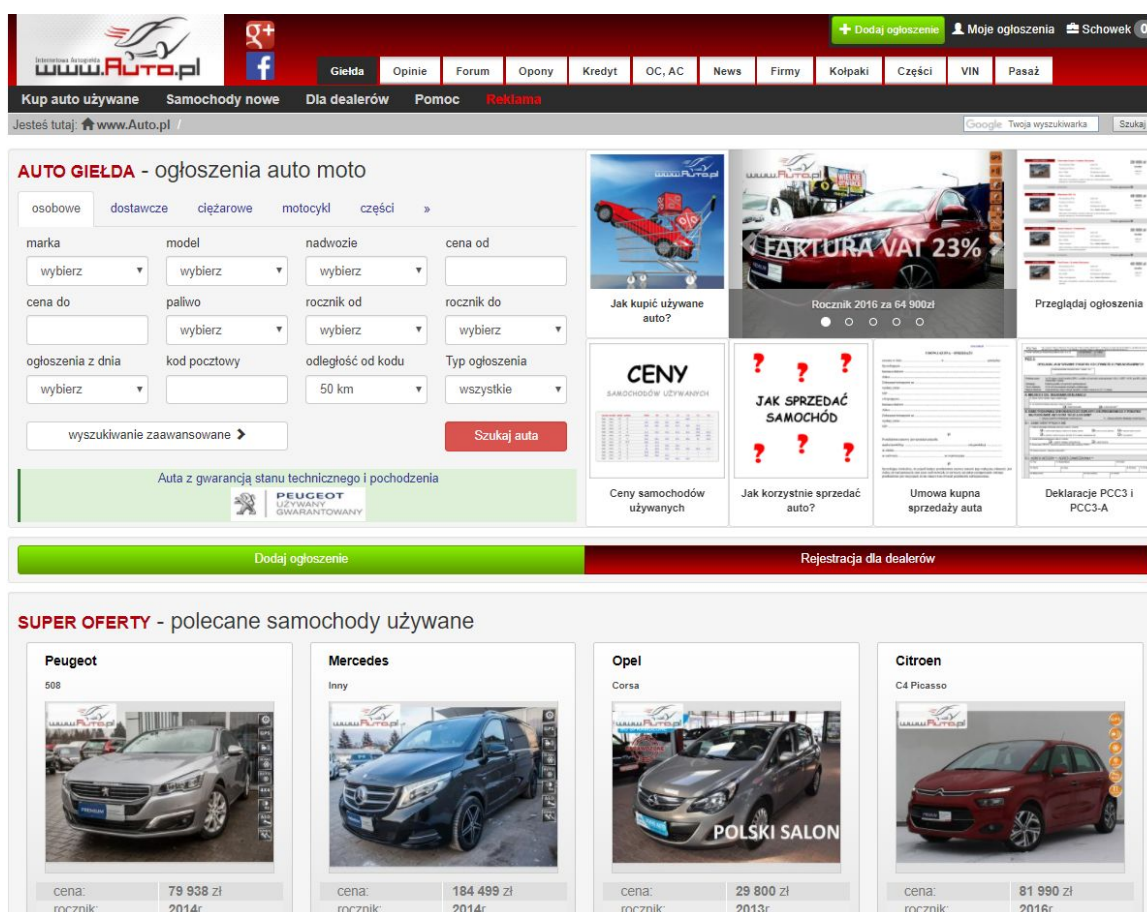
Rysunek 2.2: zrzut przedstawiający wygląd porównywarki aut w aplikacji MOTOMI

AUTOTRADER.PL - znajdziemy tutaj różne ogłoszenia motoryzacyjne, więc poza standardowymi samochodami osobowymi wystawiane tu są ciężarówki, autobusy, czy jachty. Pozostałe funkcje nie odbiegają od konkurencji. Podobnie jak w powyższym przykładzie aplikacja ta zawiera moduł z artykułami i poradami, lecz również nie jest on bezpośrednio powiązany z wystawianymi pojazdami.



Rysunek 2.3: zrzut wyglądu listy ogłoszeń i dostępnych filtrów na AUTOTRADER.PL

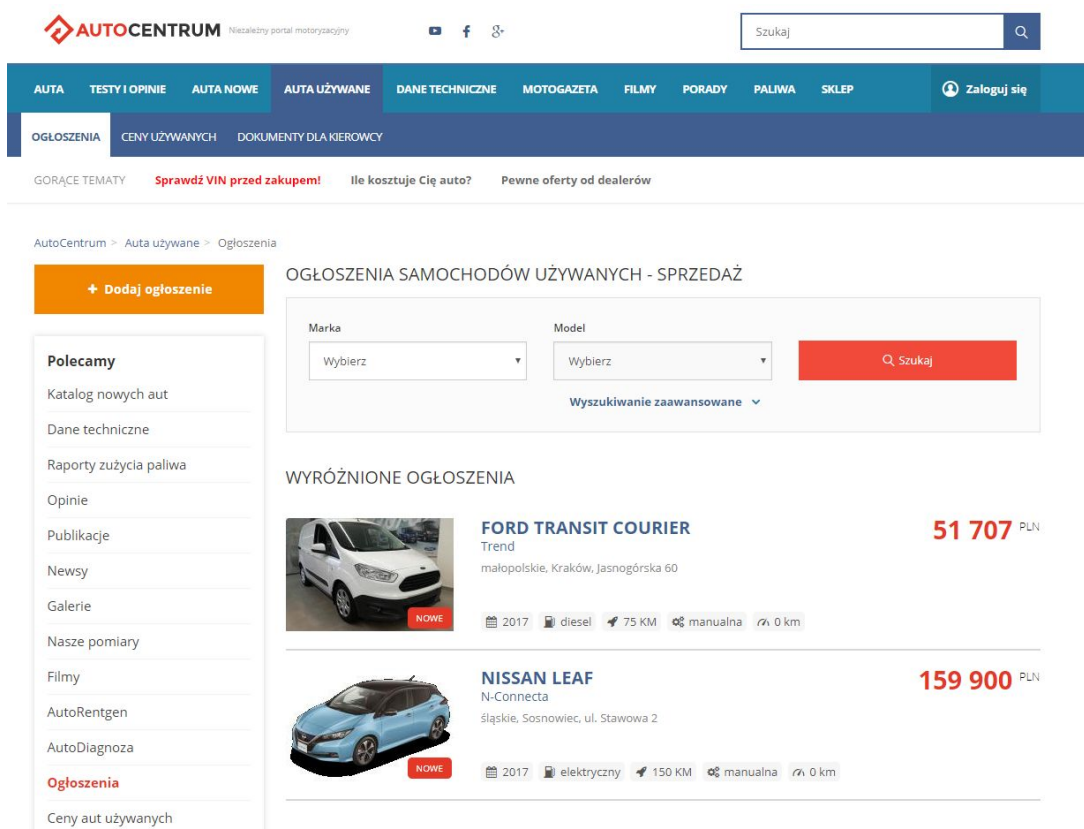
AUTO.PL - bardzo rozbudowana aplikacja z mnóstwem opcji. Oprócz ogłoszeń z samochodami - osobowymi, ciężarowymi i dostawczymi - znajdziemy tu ogłoszenia z oponami i innymi częściami, porównywarke cen, opinie użytkowników o poszczególnych modelach oraz forum. Tym razem jednak poszczególne elementy są ze sobą zintegrowane. Przykładowo z poziomu ogłoszenia danego auta możemy zobaczyć jego opinie, porównać je z podobnymi samochodami, znaleźć pasujące opony/inne części, czy zobaczyć wątki na forum odnośnie tego właśnie modelu. Jedyne zastrzeżenia można mieć do layoutu serwisu. W niektórych miejscach bywa on nie do końca czytelny.



Rysunek 2.3: zrzut prezentujący wygląd strony głównej aplikacji AUTO.PL

AUTOCENTRUM.PL - również bardzo rozbudowana aplikacja. Tym razem zajmująca się jedynie samochodami osobowymi. Poza ogłoszeniami zawiera testy, publikacje, newsy, rozmaite pomiary oraz filmy. A to wszystko z podziałem na marki, gdzie dodatkowo znajdziemy szczegółowe informacje o każdym modelu i jego wersji. Znajdziemy tutaj także średnie ceny paliw z podziałem na województwa. Większość tych modułów dostępna jest z

poziomu ogłoszeń, tak więc łatwo można dowiedzieć się wielu informacji o danym aucie bez dłuższych poszukiwań.



Rysunek 2.5: zrzut strony z ogłoszeniami aut używanych w AUTOCENTRUM.PL

Inne serwisy z ogłoszeniami - prawie każdy większy serwis o tematyce ogłoszeniowej zawiera kategorię motoryzacyjną. Mam tu na myśli takie aplikacje, jak OLX, GUMTREE, ALLEGRO, czy GRATKA. Zwykle takie kategorie różnią się jedynie opcjami przy dodawaniu ogłoszeń oraz ich wyszukiwaniu.

2.2. Identyfikacja aktorów

W tytułowej aplikacji rozróżnić można cztery typy aktorów. Rozróżniane są oni poprzez typ konta oraz fakt czy to konto w ogóle posiadają.

GOŚĆ - podstawowy typ użytkownika. Ten status nie wymaga posiadania konta w systemie, a przez to uniemożliwia korzystanie z jakichkolwiek funkcji wymagających pobierania/zapisywania danych z/do bazy SQL. W związku z tym jest to dostęp bierny. Gość może przeglądać i wyszukiwać ogłoszenia, ma dostęp do wszystkich rodzajów artykułów, a także może czytać komentarze pod tymi właśnie.

UŻYTKOWNIK ZALOGOWANY - jest to użytkownik z aktywnym dostępem do aplikacji. Status ten wymaga posiadania konta w systemie. Oprócz wszystkich możliwości, jakie ma gość, użytkownik zalogowany dostaje opcję dodawania ogłoszeń (razem z ich edycją i usuwaniem), dodawania komentarzy pod artykułami (z możliwością ich usunięcia), dodawania ogłoszeń innych użytkowników do obserwowanych oraz kontaktu z innymi użytkownikami poprzez wiadomości w aplikacji.

UŻYTKOWNIK SPECJALNY - tego typu użytkownik ma pełny zakres możliwości, tak jak normalny zalogowany użytkownik. Dodatkową opcją wynikającą z posiadania specjalnego konta jest możliwość dodawania, edycji oraz usuwania artykułów w module publicystycznym. Tego typu konta nie można stworzyć samodzielnie. Status “specjalny” nadawany jest przez administratora systemu. Celem takiego rozwiązania jest zapobieganie sytuacji, w których losowi użytkownicy mogliby dodawać artykuły dowolnej treści. Dlatego też jedynie zaufane osoby mogą posiadać tego typu konto.

ADMINISTRATOR - osoba taka posiada dostęp do bazy danych, dzięki czemu może nadzorować dodawane do aplikacji treści oraz odpowiednio reagować w niechcianych sytuacjach. Jest ona również odpowiedzialna za “mianowanie” użytkowników do posiadania konta specjalnego. Zajmuje się także kontaktem z użytkownikami oraz pomocą w razie potrzeby.

2.3. Wymagania funkcjonalne

Wyświetlanie listy ogłoszeń - możliwość wyświetlania listy dostępnych w bazie ogłoszeń w sposób uszeregowany np. według daty dodania. Zawarcie podstawowych informacji przy kolejnych elementach listy, tak aby zainteresowani mieli wgląd w najważniejsze dane techniczne oraz cenę bez wchodzenia w szczegółowy opis.

Wyświetlanie listy artykułów - możliwość wyświetlania listy dostępnych w bazie artykułów w sposób uszeregowany np. według daty dodania. Zawarcie tytułu, głównego zdjęcia i krótkiego opisu przy kolejnych elementach listy. Filtrowanie artykułów ze względu na ich typ.

Wyświetlanie artykułów - przekazanie treści artykułu, zawartych w nim zdjęć, ewentualnych komentarzy pod nim, tabelki z danymi technicznymi pojazdu (w określonych typach artykułu np. test, porównanie) w sposób czytelny.

Wyświetlanie ogłoszeń - przekazanie opisu, danych technicznych, zdjęć oraz powiązanych materiałów publicystycznych, tak aby każde dodawane w aplikacji ogłoszenie było przedstawione w sposób czytelny.

Filtrowanie ogłoszeń - umożliwienie wyboru różnych opcji oraz wpisania wartości danych technicznych, tak aby wyświetlić listę tylko tych ogłoszeń, które spełniają określone warunki.

Wyszukiwanie ogłoszeń oraz artykułów na podstawie słów kluczowych - możliwość filtrowania listy ogłoszeń, jak i listy artykułów poprzez wpisywanie słów kluczowych, takich jak marka, model, rok. Umożliwienie podawania wielu słów do jednorazowego przeszukania bazy.

Wyświetlanie najnowszych ogłoszeń i artykułów na stronie głównej - zachęcanie użytkowników aplikacji do przeglądania najnowszych zamieszczonych w niej treści w widocznym miejscu na stronie głównej.

Umieszczanie odnośników do powiązanych treści - na stronach artykułów oraz ogłoszeń mają znajdować się miniatury powiązanych z nimi odpowiednio ogłoszeń i artykułów np. dotyczących opisywanego modelu auta. W przypadku dużej ilości powiązanych treści ma istnieć możliwość szybkiego wyświetlania listy wszystkich tego typu ogłoszeń lub artykułów.

Tworzenie konta - chętni użytkownicy mają mieć możliwość utworzenia konta w aplikacji, co zwiększa dla nich liczbę dostępnych w systemie opcji. Do rejestracji potrzebny ma być adres e-mail, imię/nazwa oraz podstawowe dane kontaktowe (miasto, numer telefonu).

Dodawanie ogłoszeń - możliwość ogłaszania samochodu na sprzedaż. Przy dodawaniu określa się parametry pojazdu, szczegóły związane z kupnem (faktura, leasing), opis oraz zdjęcia.

Edytowanie ogłoszeń - użytkownicy mają mieć możliwość edycji parametrów określonych w wystawionych przez nich ogłoszeniach, zmiany zdjęć, czy opisu.

Usuwanie ogłoszeń - użytkownicy mają mieć możliwość usuwania swoich ogłoszeń w dowolnym momencie i bez podania przyczyny.

Obserwowanie ogłoszeń - użytkownicy mają mieć możliwość oznaczania interesujących ich ogłoszeń, co ma umożliwiać łatwy i szybki dostęp do nich w każdym momencie.

Przesyłanie wiadomości między kupującym, a sprzedającym - możliwość wysyłania wiadomości tekstowych do sprzedających, na które ci będą mogli odpowiedzieć.

Dodawanie artykułów - publikacja treści w kilku kategoriach (nowości, testy, porównania, porady oraz wspomnienia historyczne). Zależnie od kategorii dodawane mają być różne elementy np. tabelka z parametrami przy porównywaniu aut lub przy testach. Funkcja dostępna jedynie dla wybranych użytkowników.

Edytowanie artykułów - możliwość edycji treści artykułów, zmiany zdjęć i parametrów w tabelkach. Dostępne jedynie dla twórcy artykułu.

Usuwanie artykułów - możliwość usunięcia artykułu w dowolnym momencie i bez podania przyczyny. Dostępne jedynie dla twórcy artykułu.

Komentowanie artykułów - zalogowani użytkownicy mają mieć możliwość wyrażania swoich opinii odnośnie artykułów w formie komentarzy. Mają mieć również możliwość usuwania ich w dowolnym momencie.

Resetowanie hasła - użytkownicy którzy nie mogą się zalogować z powodu nie pamiętania hasła powinni mieć możliwość wygenerowania nowego i otrzymania go drogą mailową.

2.4. Wymagania niefunkcjonalne

Aplikacja stworzona ma być z wykorzystaniem popularnych technologii. Ważną rzeczą jest otwartość na dodatkowe moduły i elementy, które mogłyby pojawić się w przyszłości.

Aplikacja powinna być responsywna i możliwa do wygodnego używania niezależnie od rozmiaru ekranu oraz urządzenia. Ważna jest odpowiednia wielkość elementów interfejsu graficznego oraz ich odpowiednie rozłożenie tak aby korzystanie z systemu było przyjemne i nie stwarzało niepotrzebnych komplikacji. Kolorystyka powinna być nowoczesna i nie przesadnie jaskrawa. Interfejs powinien mieć ujednolicony wygląd tzn. wzór przycisków i użyte barwy powinny być jednakowe w całej aplikacji.

Ważne jest aby jednocześnie na ekranie nie znajdowało się zbyt wiele informacji oraz elementów. Poszczególne moduły mają być łatwe w użytkowaniu, a znalezienie odpowiedniej opcji ma być proste i nie ma zajmować przesadnie dużo czasu.

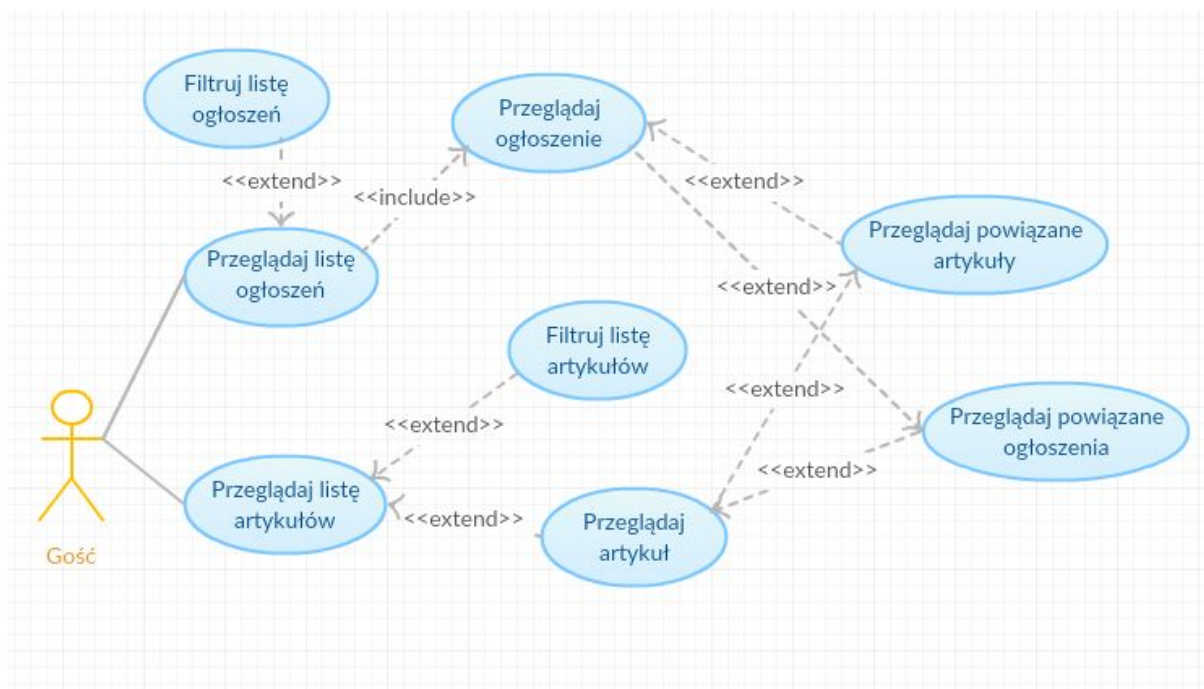
Mimo, że na początku liczba użytkowników aplikacji nie będzie znaczna to i tak musi być ona przygotowana na przechowywanie dużych ilości danych. Zastosowana baza danych powinna bez problemów radzić sobie z tym problemem. Ponadto serwer ma być odpowiednio przygotowany na przyjęcie dużej, domyślnie stale rosnącej, liczby klientów, aby nie pojawiły się sytuacje z brakiem dostępu do aplikacji lub długim czasem oczekiwania na odpowiedź. Pod uwagę trzeba również wziąć zapewnienie odpowiednio dużej przestrzeni dyskowej na przechowywanie dodawanych przez użytkowników zdjęć.

2.5. Diagramy języka UML

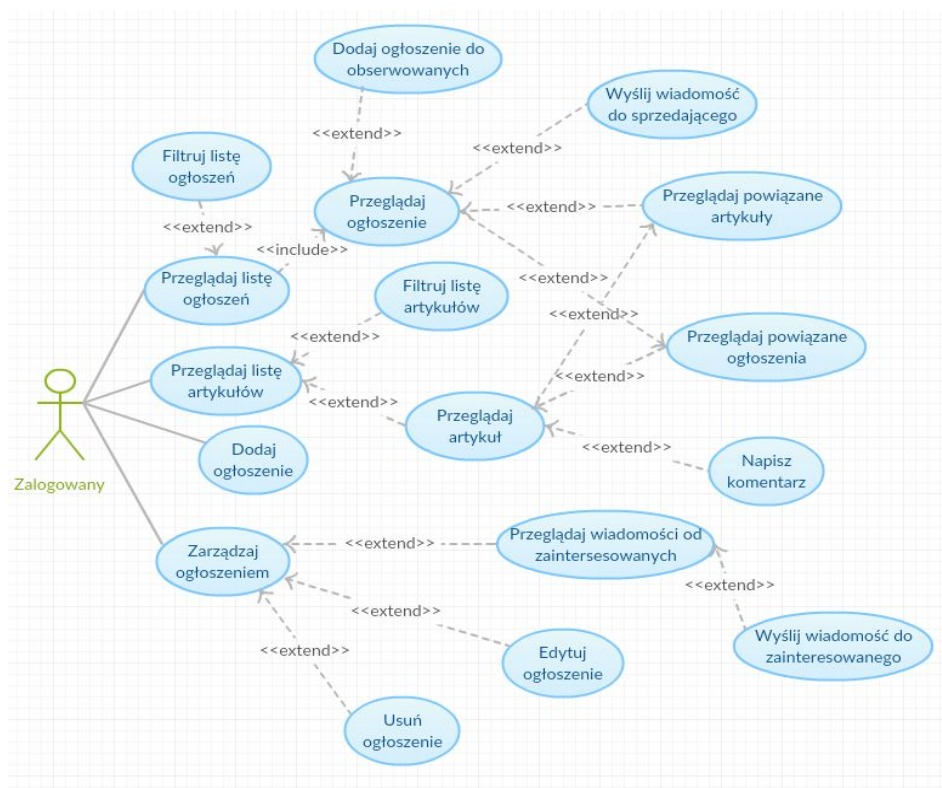
UML jest językiem modelowania służącym do graficznego przedstawienia systemu. To przedstawienie ma formę diagramów, przy czym rozróżniane są różne ich wersje. Tutaj przedstawione zostaną diagramy przypadków użycia.

Tego typu diagramy służą przedstawieniu funkcjonalności systemu. Graficznie reprezentowane są w nich własności w formie, w jakiej są one widoczne z zewnątrz przez

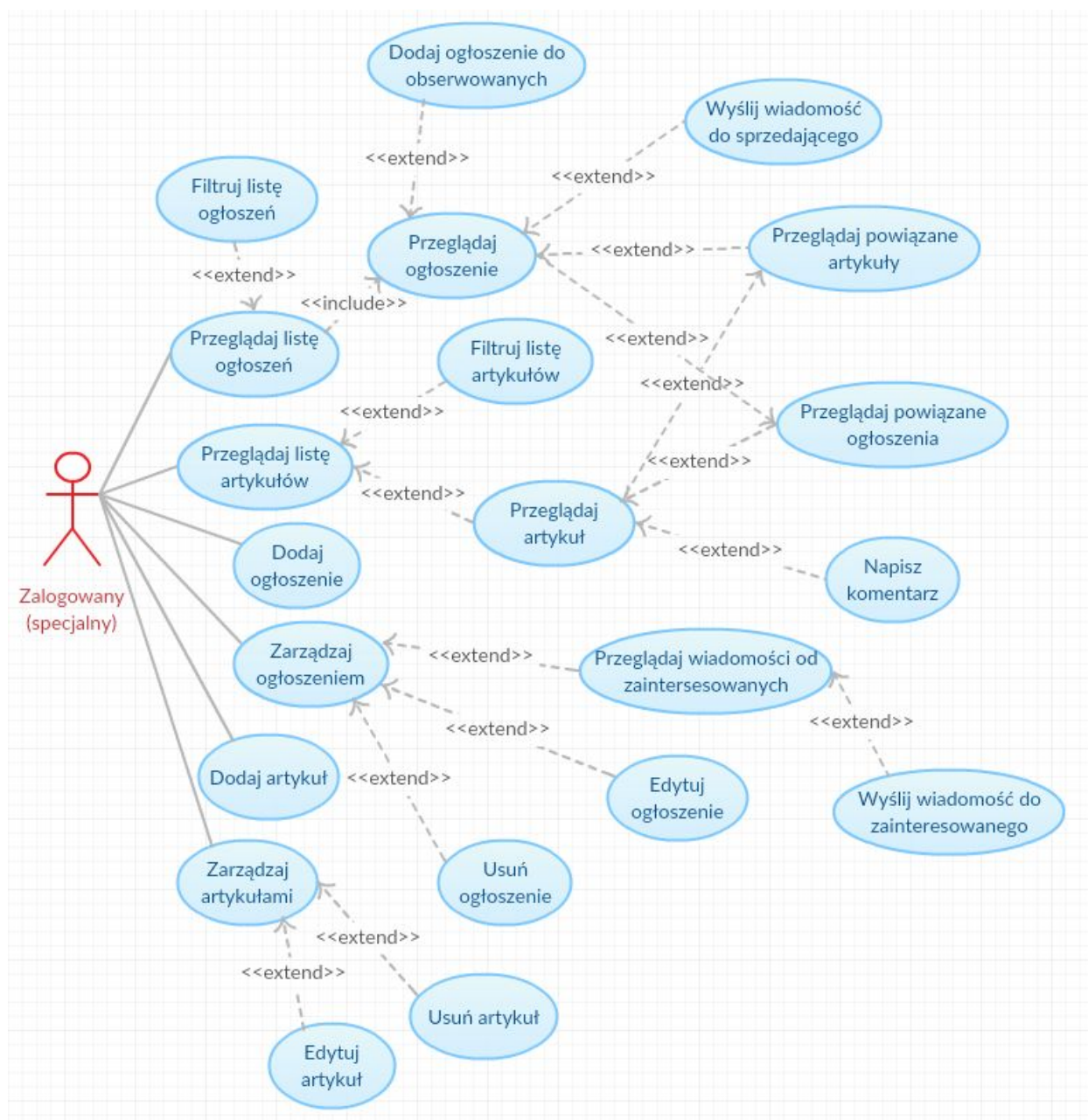
użytkowników aplikacji. Pomocne są przy implementowaniu wymagań funkcjonalnych w systemie.



Rysunek 2.6: diagram przypadków użycia dla użytkownika niezalogowanego (gościa)



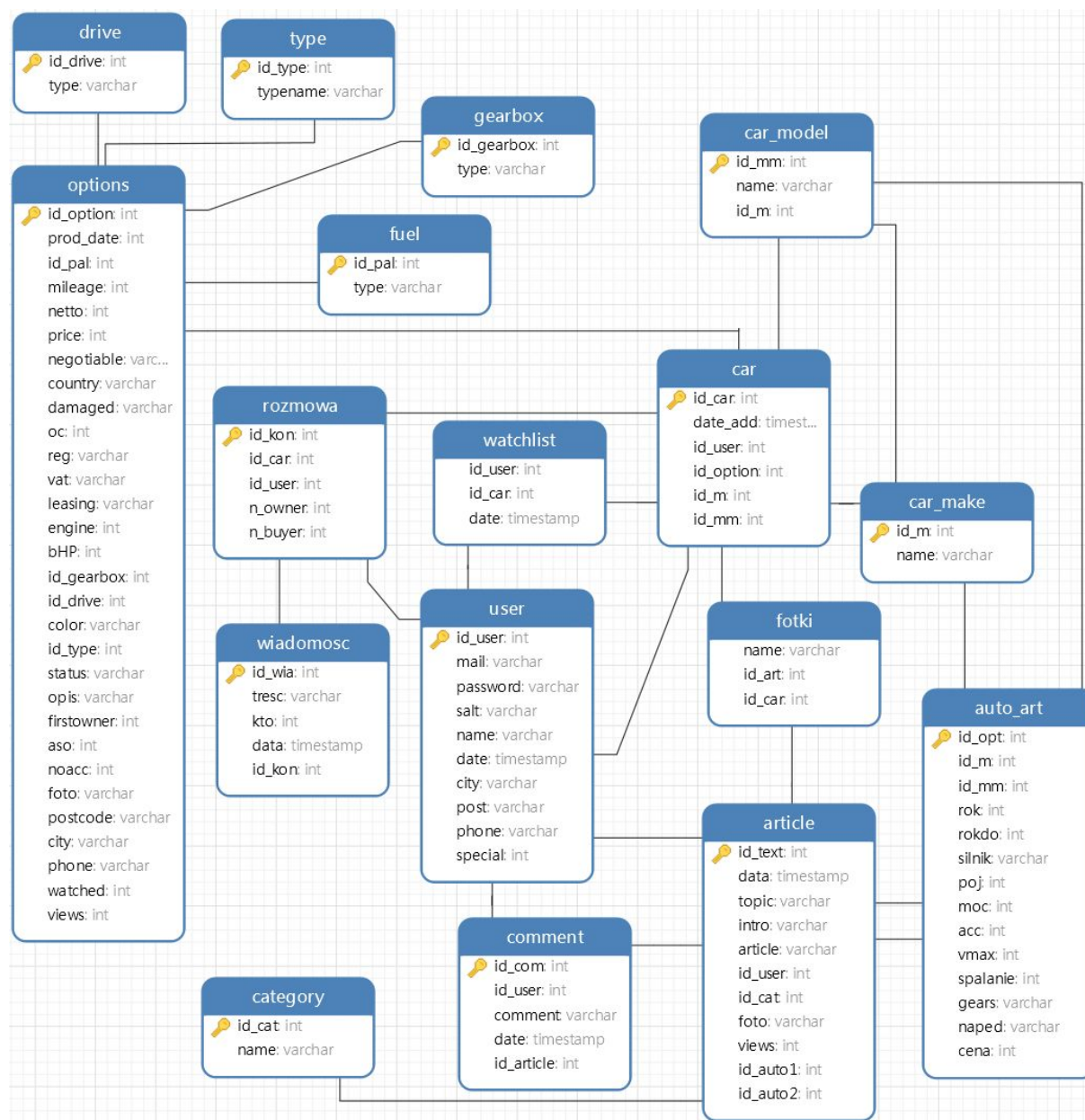
Rysunek 2.7: diagram przypadków użycia dla użytkownika zalogowanego



Rysunek 2.8: diagram przypadków użycia dla użytkownika specjalnego

2.6. Struktura bazy danych

Przedstawiona jest tu struktura bazy MySQL w formie diagramów ER, wraz z opisem poszczególnych tabel.



Rysunek 2.9: diagramy ER

user - zawiera dane użytkownika, które ten podaje przy tworzeniu nowego konta. Dodatkowo znajduje się tutaj flaga 'special' oznaczająca czy jest to konto zwykłe, czy specjalne.

car - tabela łącząca ogłoszenie z użytkownikiem, które je dodał, a także z parametrami ogłoszonego auta oraz marką i modelem.

car_make - zawiera dostępne marki samochodów.

car_model - zawiera dostępne modele samochodów przyporządkowane do odpowiadających marek.

options - znajdują się tutaj parametry techniczne ogłaszanego pojazdu oraz statystyki dotyczące wyświetleń/obserwacji.

drive - zawiera dostępne typy napędów, jakie mogą mieć wystawiane w ogłoszeniach samochody

type - zawiera dostępne typy aut, jakie można ogłaszać.

gearbox - zawiera rodzaje skrzyni biegów, jakie mogą mieć wystawiane w ogłoszeniach samochody.

fuel - zawiera rodzaje paliwa, z jakich mogą korzystać wystawiane w ogłoszeniach samochody.

article - znajdują się tu treści oraz szczegóły dodawanych artykułów - kto dodał, z jakiej są kategorii, ilość wyświetleń oraz ewentualne połączenia z tabelą przechowującą dane o opisywanych pojazdach (zależy od kategorii).

auto_art - zawiera dane o opisywanych w artykułach samochodach. Ilość przechowywanych rekordów zależy od kategorii artykułu.

comment - przechowuje komentarze do artykułów wraz z informacją o ich autorach, datach dodania i numerach artykułów.

category - przechowuje dostępne kategorie artykułów.

fotki - zawiera zdjęcia przypisane artykułom i ogłoszeniom.

rozmowa - przechowuje listę konwersacji pomiędzy kupującymi i sprzedającymi. Dodatkowo zawiera flagi określające, czy dany użytkownik nie wysłał nowej wiadomości.

wiadomosc - zawiera pojedyncze wiadomości tekstowe wraz z określeniem jakiego ogłoszenia one dotyczą oraz który użytkownik je wysłał.

watchlist - łączy użytkowników z ogłoszeniami, które dodali do obserwowanych.

3. Implementacja i wdrożenie systemu

3.1. Przegląd popularnych technologii do tworzenia aplikacji internetowych

Java - obiektowy język programowania, który poprzez dostępność rozmaitych bibliotek może służyć do tworzenia zróżnicowanego oprogramowania. Zastosowanie odpowiednich technologii Javy może między innymi umożliwić budowanie aplikacji webowych. Do najczęściej wykorzystywanych należą Java Servlet API oraz JavaServer Pages. Serwlety to klasy działające po stronie serwera WWW. Są one niezależne od protokołów i platform. Rozszerzają możliwości modelu żądanie-odpowiedź [1]. Technologia JSP odpowiada zaś za szybkie i proste tworzenie dynamicznych dokumentów WWW. Odpowiada za tworzenie aplikacji sieciowych, które są niezależne od serwera i platformy [2].

AngularJS - framework bazujący na JavaScript, który rozszerza HTML o nowe atrybuty. Ma to na celu przystosowanie go do obsługi dynamicznych treści. Pozwala na wdrożenie wzorca MVC do aplikacji sieciowych, ułatwiając testowanie i rozwój [3].

ASP.NET - technologia będąca zbiorem narzędzi i bibliotek służących do tworzenia dynamicznych aplikacji internetowych na platformie .NET. Wykorzystuje programowanie zdarzeniowe, dzięki czemu kod aplikacji oddzielony jest od kodu prezentacji. Poprawia to czytelność oraz umożliwia wielokrotne wykorzystywanie tych samych fragmentów kodu do różnych zadań. Korzysta z języków C# oraz Visual Basic [4].

Django - jeden z najpopularniejszych frameworków napisanych w języku Python. Służy do szybkiego tworzenia bezpiecznych i łatwych do zarządzania aplikacji internetowych. Zawiera gotowe szablony, co upraszcza proces tworzenia elementów często występujących w kolejnych projektach. Dostarcza wszystkie części składowe potrzebne do tworzenia aplikacji. Korzysta z zasady “Don’t repeat yourself” ograniczając ilość kodu poprzez eliminację niepotrzebnej duplikacji. Django zapewnia także obsługę wielu platform i systemów operacyjnych [5].

Perl - język programowania wysokiego poziomu, mający szeroką skalę zastosowań. Cechuje się dużą praktycznością - łatwy w użytku, efektywny, nie jest jednak przesadnie minimalistyczny ani elegancki. Zapewnia olbrzymią kolekcję dodatkowych modułów rozszerzających funkcjonalność [6].

Ruby on Rails - stworzony w języku Ruby framework do szybkiego tworzenia aplikacji sieciowych. Do jego głównych założeń, oprócz prostoty użytkowania należy także przyjemność z pisania kodu. Podobnie, jak Django korzysta z reguły DRY (unikanie powtarzalności kodu), lecz dodatkowo zapewnia zasadę "Convention Over Configuration". Polega ona na używaniu gotowych, zalecanych wzorców ograniczając w ten sposób konfigurację do minimum. Obsługuje wtyczki pozwalające na szybkie rozszerzenie funkcji tworzonej aplikacji [7].

3.2. Wybór technologii i narzędzi do realizacji systemu

3.2.1. Front-end

HTML - język opisowy określający zawartość strony przy użyciu znaczników. Mogą nimi być elementy tekstowe, nagłówki, listy, tabele, hiperłącza, przyciski, obrazki, elementy służące do wpisywania/wybierania/zaznaczania itd. Określamy w nim także parametry strony np. tytuł, używany język, używane skrypty, czy style [8].

CSS - kaskadowe arkusze stylów. Wykorzystywane do modyfikacji i dostosowywania elementów uprzednio zdefiniowanych w HTML. Pozwalają na zmiany kolorów, rozmiarów, odpowiednie rozmieszczenie obiektów i wiele więcej. Pozwala także na dziedziczenie właściwości pomiędzy kolejnymi elementami.

Bootstrap - framework CSS, zawierający zestaw narzędzi oraz gotowych elementów do stylizacji interfejsu graficznego stron oraz aplikacji webowych. Jedną z ważniejszych cech jakie dostarcza jest struktura dwunastu kolumn. Nie tylko pozwala to w łatwy sposób rozmieścić poszczególne obiekty, ale również sprawia że aplikacja jest responsywna - wszystko zachowuje swoje miejsce niezależnie od używanego urządzenia (istnieje również opcja określenia położenia i rozmiaru dla określonych wielkości ekranu) [9].

JavaScript - język skryptowy, który najczęściej stosowany jest na stronach/aplikacjach internetowych. Najpopularniejszym zastosowaniem jest oprogramowywanie reakcji na zdarzenia np. kliknięcie przycisku, czy wpisanie odpowiedniej wartości w formularzu, lub dynamiczne wczytywanie danych. Służy również do tworzenia złożonych efektów wizualnych [10].

AJAX + jQuery - technologie pozwalające na tworzenie aplikacji internetowych, w których komunikacja na linii klient - serwer odbywa się asynchronicznie. Dzięki temu interakcja odbywa się bez przeładowywania całej strony. Pozwala to na dynamiczne pobieranie i wysyłanie danych, co często uprzyjemnia korzystanie z takiej aplikacji [11]. Sam AJAX nie jest jednak najwygodniejszy w obsłudze, dlatego często dodaje się bibliotekę jQuery ułatwiającą pracę z tą technologią [12].

3.2.2. Back-end

Apache HTTP Server - najpopularniejszy na rynku otwarty serwer HTTP. Oferuje wsparcie dla wielu systemów operacyjnych oraz bezpieczeństwo, skalowalność i wielowątkowość. Można rozszerzać jego funkcje o moduły np. `mod_authz_host` do uwierzytelniania, czy `mod_ssl` do SSL. Wraz z językiem PHP oraz bazą MySQL tworzy platformę AMP [13].

PHP - język skryptowy o budowie zbliżonej do języków C oraz C++, jednak w pewnym stopniu uproszczony. Służy do generowania stron internetowych i budowania aplikacji webowych w czasie rzeczywistym, a także do rozszerzania ich funkcjonalności. Może być także stosowany do przetwarzania danych z wiersza poleceń oraz tworzenia programów działających w trybie graficznym (po użyciu odpowiednich bibliotek i rozszerzeń). Ma wbudowaną obsługę wielu typów baz danych co ułatwia operacje na przechowywanych danych [14]. Jego działanie jest szybkie, a możliwości na tyle szerokie, że wykorzystywany jest zarówno w małych projektach, jak i w najpopularniejszych aplikacjach na świecie [15].

MySQL - system zarządzania relacyjnymi bazami danych (RDBMS). Dostępny jest na wszystkie popularne platformy (dostępność kodu źródłowego pozwala na rozszerzanie go na inne systemy). Jego głównymi cechami są wysoka wydajność, łatwość użycia oraz

skalowalność. Korzysta z nieco zmienionej wersji SQL'a. Dane przechowywane są w formie rekordów w odpowiednich tabelach. Bardzo dobrze radzi sobie z bardzo dużymi bazami. Otwartość licencji sprawia, że można dopasować go do swoich potrzeb [16].

3.3. Architektura systemu

System utworzony jest w architekturze klient - serwer. Komunikacja następuje poprzez model żądanie - odpowiedź. Wszystkie dane przechowywane są po stronie serwera i dostarczane do klienta w momencie żądania usługi. Do najważniejszych typów tego typu architektur zaliczamy architektury warstwowe. Zarysowany jest w nich podział na trzy typy komponentów: interfejs użytkownika, logikę przetwarzającą oraz bazę danych. Zależnie od umiejscowienia logiki przetwarzającej następuje podział na architekturę dwuwarstwową (logika podzielona pomiędzy dwie warstwy), trójwarstwową (każdy element ma osobną warstwę) oraz wielowarstwową (logika dzieli się na dwie warstwy) [17].

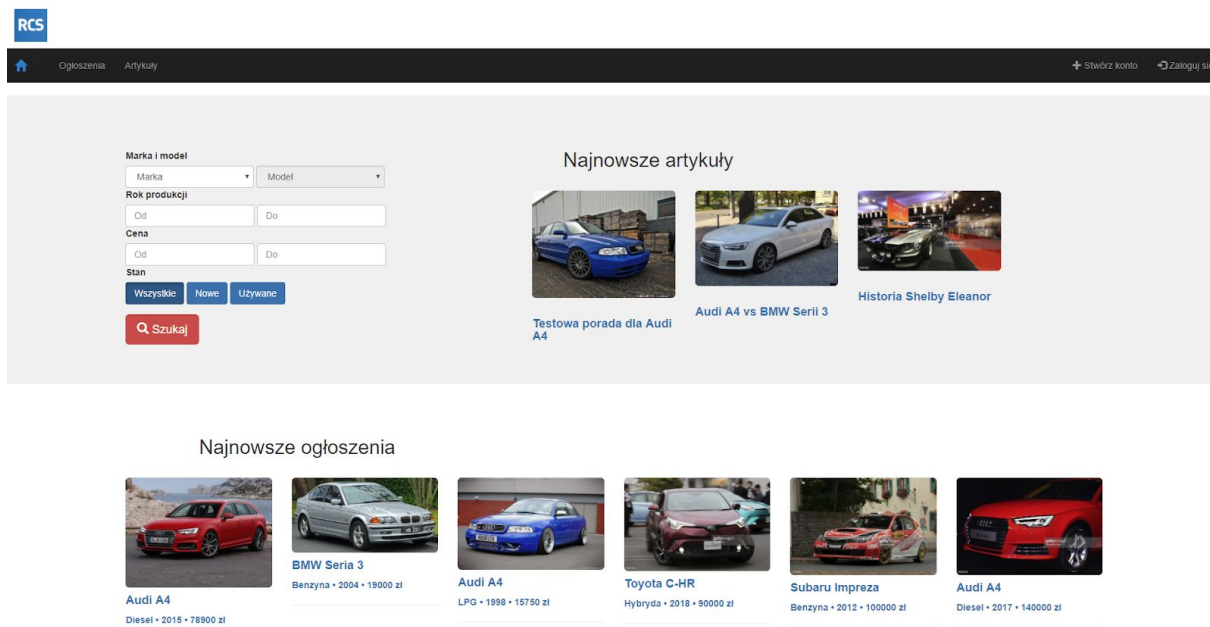
Tytułowa aplikacja utworzona jest w modelu architektury trójwarstwowej. Następuje w niej wyraźny podział na warstwę prezentacji (interfejs w jakim poruszają się użytkownicy), bazę danych (składowanie danych użytkowników oraz ogłoszeń i artykułów) i warstwę logiki (przetwarzanie żądań użytkowników oraz komunikacja pomiędzy pozostałymi warstwami) [18].

3.4. Elementy składowe aplikacji

Pasek nawigacji - element, który dostępny jest z każdego miejsca w aplikacji. Umożliwia szybkie poruszanie się pomiędzy modułami. Tam też znajdziemy opcje odpowiedzialne za logowanie, wylogowywanie oraz tworzenie nowego konta. Dzięki zastosowaniu pluginu do technologii Bootstrap o nazwie Affix [19] położenie paska jest zawsze u góry ekranu niezależnie od scrollowania. Opcje dostępne na tym pasku zależą od statusu użytkownika. W przypadku zalogowania umożliwia on zarządzanie treściami dostępnymi dla danego konta oraz przejście do kreatora nowych ogłoszeń. W przypadku, gdy szerokość ekranu ma mniej niż 768 pikseli dostępne opcje zostają ukryte i pokazują się jedynie po kliknięciu (zostaje jedynie przycisk 'home').

Strona główna - początkowy element aplikacji. Zawiera prosty formularz umożliwiający szybkie znalezienie interesujących ogłoszeń oraz miniatury pozwalające na dostęp do nowo

dodanych w aplikacji treści. Znajdziemy tutaj trzy najnowsze artykuły oraz sześć najświeższych ogłoszeń.



Rysunek 3.1: zrzut ekranu strony głównej

Ekran tworzenia konta - umożliwia stworzenie nowego konta do systemu. Znajdziemy tu, krótki opis nowych możliwości, jakie ta procedura zapewnia. Do stworzenia konta podajemy adres email, imię/nazwę, miasto, kod pocztowy, telefon oraz hasło (o minimalnej długości 6 znaków). Wykorzystywane są funkcje w języku JavaScript mające na celu sprawdzanie spełnienia określonych wymagań tj. długości hasła, poprawności hasła z jego powtórzeniem. Przy pomocy AJAX i jQuery dynamicznie sprawdzana jest możliwość utworzenia konta pod danym adresem email (oznaczane jest to za pomocą znaczków pochodzących z projektu Glyphicon [20]).



Rysunek 3.2: zrzut przedstawiający brak możliwości użycia przykładowego e-maila w systemie

```

function checkpasslen()
{
    var p = document.getElementById('password').value;
    if(p.length < 7 && p.length > 0)
    {
        alert("Hasło musi być dłuższe niż 6 znaków!");
        setTimeout("document.getElementById('password').focus()", 0);
    }
    if(p.length > 6)
        $("#cpasswrod").prop('disabled', false);
}

function checkpass()
{
    var p = document.getElementById('password').value;
    var c = document.getElementById('cpasswrod').value;
    if(p != c)
    {
        alert("Hasła się nie zgadzają")
        document.getElementById('cpasswrod').value = "";
    }
}

function check()
{
    $("#loaderIcon").show();
    jQuery.ajax({
        url: "checkdata.php",
        data: 'mail='+$("#mail").val(),
        type: "POST",
        success: function(data)
        {
            $("#user-availability-status").html(data);
        },
        error: function (){}
    });
}

```

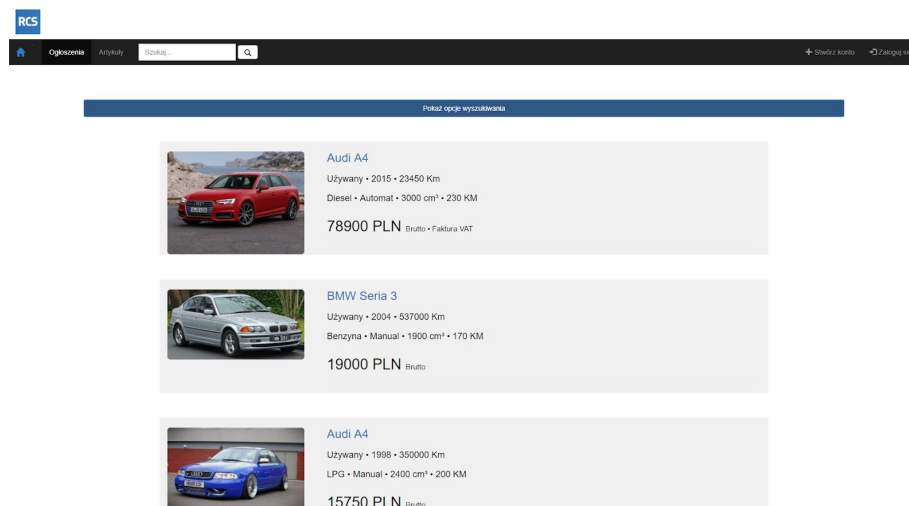
Rysunek 3.3: zrzut przedstawiający funkcje odpowiedzialne za sprawdzanie poprawności danych podawanych podczas tworzenia konta - długości hasła, zgodności hasła z jego powtórzeniem oraz dostępności danej nazwy.

Ekran logowania - po podaniu adresu email oraz hasła następuje sprawdzenie poprawności tych danych, a w przypadku powodzenia tworzona jest nowa sesja dla tego użytkownika.

Resetowanie hasła - opcja, dla użytkowników którzy z jakiegoś powodu nie pamiętają hasła do systemu. Po wprowadzeniu adresu email następuje wygenerowanie nowego hasła i przesłanie go drogą mailową.

Strona z listą ogłoszeń - domyślnie zawiera wszystkie ogłoszenia dostępne w aplikacji przedstawione w formie listy zawierającej miniaturę zdjęcia głównego oraz najważniejsze

informacje (marka, model, rok produkcji, przebieg, pojemność silnika, moc silnika, rodzaj skrzyni biegów, rodzaj paliwa używanego przez pojazd oraz cenę z określeniem, czy jest to cena netto, czy brutto wraz z ewentualnym oznaczeniem sprzedaży na fakturę).



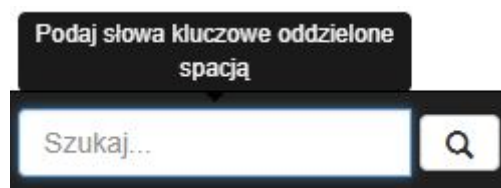
Rysunek 3.4: zrzut listy ogłoszeń

Filtr ogłoszeń - formularz zawierający wszystkie potrzebne opcje przy wyszukiwaniu pojazdów o określonych parametrach. Domyślnie jest on ukryty, aby nie zajmować niepotrzebnie dużo przestrzeni.

The screenshot shows a search filter form titled 'Pokaż opcje wyszukiwania'. It contains several input fields and dropdown menus for filtering car listings. The fields are arranged in a grid-like fashion. The first row includes 'Marka' (dropdown), 'Model' (dropdown), 'Rok(od)' (text input), and 'Rok(do)' (text input). The second row includes 'Cena(od)' (text input), 'Cena(do)' (text input), 'Stan' (dropdown), and 'Uszkodzone' (dropdown). The third row includes 'Typ' (dropdown), 'Paliwo' (dropdown), 'Skrzynia biegów' (dropdown), and 'Napęd' (dropdown). The fourth row includes 'Poj. silnika(od)' (text input), 'Poj. silnika(do)' (text input), 'Moc silnika(od)' (text input), and 'Moc silnika(do)' (text input). At the bottom left, there is a red button with a magnifying glass icon and the text 'Szukaj'.

Rysunek 3.5: zrzut przedstawiający dostępne opcje filtrowania

Wyszukiwarka słów kluczowych - znajduje się na pasku nawigacyjnym w panelu listy ogłoszeń oraz artykułów. Po najejchaniu myszą/kliknięciu pokazuje szczegóły używania. Użytkownicy mogą podać w tym miejscu dowolne słowa oddzielone znakami spacji, a aplikacja sprawdzi, czy w określonych danych (marka, model, kolor itp.) występują dane zwroty. Celem tego systemu jest szybsze filtrowanie ogłoszeń kiedy nie mamy zwięzłe określonych detali poszukiwanego pojazdu.

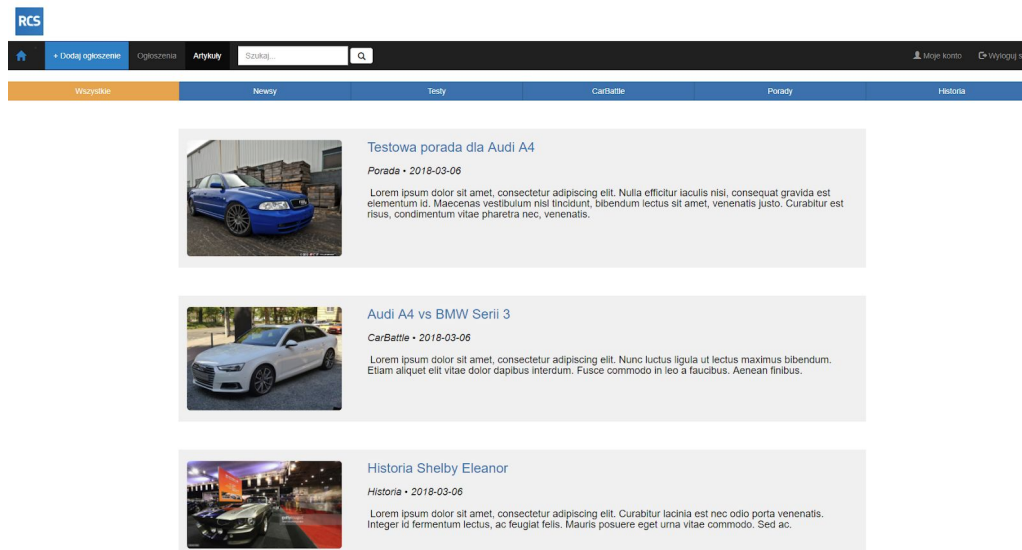


Rysunek 3.6: zrzut pokazujący wyszukiwarkę słów kluczowych od strony klienta

```
$szukaj = mysqli_real_escape_string($link, $_POST['search']);
$words = explode(' ', $szukaj);
$art = "SELECT id_text, topic, intro, foto, name, date(data) as dd FROM article, category WHERE article.id_cat = category.id_cat";
foreach ($words as $word)
    $art .= ' and concat(topic,intro) LIKE "%' . $word . '"';
if($cat != NULL)
    $art .= " and name = '$cat' ORDER BY data desc";
else
    $art .= ' ORDER BY data desc';
```

Rysunek 3.7: zrzut przedstawiający zapytanie do bazy danych w celu znalezienia artykułów zawierających podane słowa kluczowe

Strona z listą artykułów - zbliżona wyglądem do strony z listą ogłoszeń. Podobnie jak tam znajdziemy tu wyszukiwarkę słów kluczowych. Lista uporządkowana jest według daty dodania (najnowsze na górze). Zawiera miniatury głównych zdjęć, tematy, nazwę kategorii, datę dodania oraz intro artykułu (zwykle krótki opis treści). U góry znajdziemy przyciski odpowiedzialne za wyświetlanie wszystkich ogłoszeń lub tylko tych, które należą do określonej kategorii.



Rysunek 3.8: zrzut listy artykułów

Kreator nowego ogłoszenia - umożliwia dodanie nowego ogłoszenia po uprzednim podaniu jego specyfikacji. Na początku użytkownik wybiera markę i model z gotowej listy (przy pomocy AJAX i jQuery dynamicznie ładowana jest lista dostępnych modeli po wybraniu określonej marki). Następnie określa się stan techniczny, parametry, cenę z dodatkowymi

atrybutami (możliwa negocjacja, cena brutto/netto, możliwy leasing itp.), opis oraz zdjęcia. Na końcu wypisane zostają dane sprzedającego (pobrane z bazy), które można zmienić dla danego ogłoszenia.

Dodaj nowe ogłoszenie

Marka <input type="text" value="Marka"/>	Model <input type="text" value="Model"/>	Stan <input type="text"/>
---	---	------------------------------

Typ pojazdu <input type="text"/>	Uszkodzony <input type="text"/>	Rok produkcji <input type="text"/>
Paliwo <input type="text"/>	Skrzynia biegów <input type="text"/>	Przebieg - km <input type="text"/>
Napęd <input type="text"/>	Poj. silnika - cm ³ <input type="text"/>	Moc silnika - KM <input type="text"/>

Cena - zł <input type="text"/>	<input type="checkbox"/> Cena netto <input type="checkbox"/> Do negocjacji <input type="checkbox"/> Zarejestrowany	<input type="checkbox"/> Faktura VAT <input type="checkbox"/> Leasing <input type="checkbox"/> Ubezpieczony
-----------------------------------	--	---

Kolor <input type="text"/>	Kraj pochodzenia <input type="text"/>	<input type="checkbox"/> Pierwszy właściciel <input type="checkbox"/> Serwisowany w ASO <input type="checkbox"/> Bezwypadkowy
-------------------------------	--	---

Rysunek 3.9: zrzut pokazujący fragment formularza dodawania ogłoszenia

Kreator nowego artykułu - wygląda i działa podobnie do kreatora ogłoszeń. Opcja dostępna jedynie dla użytkowników o specjalnym statusie konta. Na pasku nawigacyjnym opcja “+ Dodaj artykuł” nie przenosi natychmiast do kreatora, lecz wymaga wyboru kategorii z listy typu *dropdown*. Zależy od tego bowiem liczba dostępnych opcji przy dodawaniu ogłoszenia. W przypadku typów *News* i *Historia* podaje się tytuł, intro (wstęp), treść artykułu i zdjęcia. Kategoria *Porada* rozszerza to o wybór marki i modelu, których dotyczy (z gotowych list) oraz przedziału lat, w których produkowano ten model. Kreator jest jeszcze bardziej rozbudowany w przypadku opcji *Test* oraz *CarBattle* (porównanie dwóch samochodów). Pojawiają się tutaj miejsca na specyfikację opisywanych aut (jedno w teście i dwa w carbattle).

```


$inc = 0;
foreach($_FILES["files"]["tmp_name"] as $key=>$tmp_name)
{
    $err= array();
    $files_name = $_FILES['files']['name'][$key];
    $files_size = $_FILES['files']['size'][$key];
    $files_tmp = $_FILES['files']['tmp_name'][$key];
    $files_type = $_FILES['files']['type'][$key];
    $files_ext = strtolower(end(explode('.',$_FILES['files']['name'][$key])));
    $extensions = array("jpeg","jpg","png");
    if(in_array($files_ext,$extensions)=== false)
    {
        $err[]="Można dodać jedynie pliki JPG, JPEG, i PNG";
    }
    if($files_size > 5242880)
    {
        $err[]='Rozmiar musi być mniejszy niż 5MB';
    }
    if(empty($err)===true)
    {
        $tmp = explode(".", $_FILES["files"]["name"][$key]);
        $fotka = round(microtime(true)) . $_id_user . $inc . '.' . end($tmp);
        move_uploaded_file($files_tmp,"res/".$fotka);
        chmod("res/".$files_name, 0777);
        chown("res/".$files_name, root);
        $i0=mysqli_query($link, "insert into fotki (name, id_car) values ('$fotka', '$idcar');");
    }
    else
    {
        print_r($errors);
    }
    $inc++;
}

```

Rysunek 3.10: zrzut fragmentu kodu odpowiedzialnego za dodawanie zdjęć na serwer, po uprzednim sprawdzeniu spełniania wymagań i zmianie nazwy na unikalną. Przetworzone zdjęcie trafia do odpowiedniego folderu, a do bazy zapisywana jest jego nazwa oraz identyfikator treści, do której należy

Strona ogłoszenia - dla wielu najważniejszy element aplikacji. Przechodzimy tu z listy ogłoszeń lub z powiązanego artykułu. Na tej stronie zamieszczone są dane sprzedawanego pojazdu: cena (z dodatkowymi opcjami o ile takie posiada), tabelka z parametrami, opis, zdjęcia, powiązane artykuły, data dodania, liczba wyświetleń oraz dodatkowe opcje. Do prezentacji zdjęć zastosowany został kolejny plugin frameworka Bootstrap, a mianowicie Carousel [21]. Jest to przeglądarka zdjęć umożliwiającą poruszanie się w dwóch kierunkach (prawo i lewo) w celu przewijania kolejnych fotografii. Kliknięcia aktualnie wyświetlanego zdjęcia spowoduje otworenie go w pełnym wymiarze w nowej karcie przeglądarki. Ponadto znajdziemy tu numer telefonu do sprzedającego, przycisk odpowiedzialny za dodanie ogłoszenia do obserwowanych (w celu późniejszego łatwego znalezienia), przycisk przenoszący do konwersacji ze sprzedającym lub wyświetlającym pole tekstowe (z użyciem pluginu Bootstrap o nazwie Modal [22]) do wysłania pierwszej wiadomości (w przypadku braku wcześniejszego kontaktu tym sposobem) oraz przycisk przenoszący do aplikacji Google Maps. W ostatnim przypadku generuje on adres URL wstawiając nazwę miasta i kod

pocztowy podane przy wystawianiu ogłoszenia i przenosi do widoku tej lokalizacji właśnie w wymienionej aplikacji Google. Kiedy oglądamy ogłoszenie które sami umieściliśmy w aplikacji, pokazuje się inny zestaw opcji. Są to: edycja, usunięcie oraz dostęp do wiadomości jakie otrzymaliśmy do tego ogłoszenia.



Typ: Kombi	Stan: Używany
Rok produkcji: 2015	Kraj pochodzenia: Belgia
Przebieg: 23450 km	Uszkodzony: Nie
Rodzaj paliwa: Diesel	Skrzynia biegów: Automat
Pojemność silnika: 3000 cm³	Moc: 230 KM
Napęd: 4x4 - stałe	Kolor: Czerwony
Ubezpieczony: Tak	Zarejestrowany: Tak
Pierwszy właściciel: Tak	Serwisowany w ASO: Tak
Bezwyypadkowy: Tak	Data dodania: 2018-03-06
Miasto: Polim	Wyświetleń: 6

Opis pojazdu

Rysunek 3.11: zrzut przedstawiający stronę ogłoszenia

Strona artykułu - zawiera artykuł przedstawiony w sposób przejrzysty i łatwy do czytania. Poza podstawowymi elementami, jak tytuł, wprowadzenie, data dodania, autor, i kategoria może przedstawiać także tabelkę z danymi pojazdu, w przypadku testu, a także tabelkę porównawczą parametrów, gdy tekst ma na celu porównanie dwóch aut. Pod artykułem znajduje się sekcja komentarzy. Zalogowani użytkownicy mogą dodawać oraz usuwać swoje opinie. Te są jednak widoczne także dla gości. W przypadku wyświetlania strony na dużym ekranie (powyżej 768 pikseli szerokości) zdjęcie główne widoczne jest pomiędzy wstępem, a faktycznym tekstem artykułu natomiast dodatkowe zdjęcia znajdują się obok, wzdłuż tekstu. Kiedy jednak ekran ma mniejszą wielkość wszystkie fotografie zostają zawarte w obiekcie typu Carousel, podobnie jak w przypadku stron ogłoszeń. Celem takiego zabiegu jest poprawienie czytelności i obsługi na mniejszych urządzeniach.

Audi A4 vs BMW Serii 3

CarBattle • dodane: 2018-03-06 • autor: Admin420

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nunc luctus ligula ut lectus maximus bibendum. Etiam aliquet elit vitae dolor dapibus interdum. Fusce commodo in leo a faucibus. Aenean finibus.



Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed fringilla ultrices tellus a consequat. Vivamus vestibulum commodo arcu, in imperdiet elit ornare nec. Aliquam elementum orci eu nulla sagittis, vitae dapibus lectus sagittis. Phasellus justo velit, ullamcorper eu sagittis vehicula, luctus in felis. Sed egestas mollis vestibulum. Quisque laoreet tellus diam, at sollicitudin enim ullamcorper in. Cras vel turpis nisi. Sed porttitor sed eros malesuada ornare. Morbi vel metus a enim pretium

Rysunek 3.12: zrzut “dużej” wersji strony artykułu

Porównanie		
	Audi A4	BMW Seria 3
Silnik	3.2 FSI	336d xDrive
Poj. silnika	3200 cm³	2993 cm³
Moc silnika	320 KM	313 KM
Przyspieszenie	6 s	5 s
Prędkość max.	250 km/h	250 km/h
Spalanie	8 l/100	6 l/100
Skrzynia biegów	6-biegowa, manualna	Automatyczna
Napęd	4x4	Tył
Cena	260000 zł	215000 zł

Rysunek 3.13: wygląd tabeli porównawczej w artykułach typu CarBattle

Audi A4 vs BMW Serii 3

CarBattle • dodane: 2018-03-06 • autor: Admin420

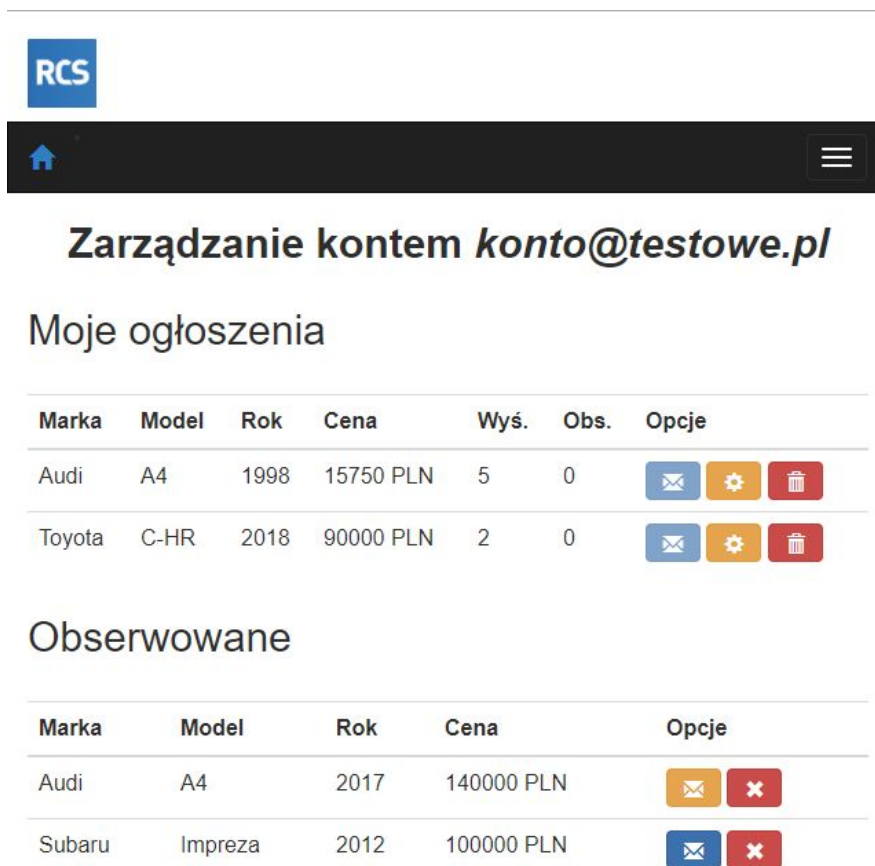
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nunc luctus ligula ut lectus maximus bibendum. Etiam aliquet elit vitae dolor dapibus interdum. Fusce commodo in leo a faucibus. Aenean finibus.



Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed fringilla ultrices tellus a consequat. Vivamus vestibulum commodo arcu, in imperdiet elit ornare nec. Aliquam elementum orci eu nulla sagittis, vitae dapibus lectus sagittis. Phasellus justo velit,

Rysunek 3.14: zrzut “małej” wersji strony artykułu

Strona zarządzania kontem - dla zalogowanych użytkowników pojawia się jako opcja “Moje konto” na pasku nawigacyjnym. Znajduje się tam tabele z listą wszystkich wystawionych z danego konta ogłoszeń i lista obserwowanych, a także lista artykułów w przypadku konta specjalnego. Każdy rekord z tych tabel posiada opcje, takie jak usuwanie/edycja wystawionego ogłoszenia, wysyłanie wiadomości, czy zaprzestanie obserwacji ogłoszenia innej osoby. Poza tym w tabelach mamy podstawowe szczegóły danych treści np. liczbę osób obserwujących nasze ogłoszenie. Kliknięcie na rekord tabeli odsyła bezpośrednio do strony danego ogłoszenia/artykułu. Dodatkowo przy pomocy JavaScript i CSS po najechaniu kursorem na dany rekord pojawia się miniatura głównego zdjęcia lub cały tytuł. Ma to pomóc w łatwej nawigacji i rozpoznaniu o jaką treść chodzi.



Rysunek 3.15: zrzut prezentujący zarządzanie kontem

Wysyłanie wiadomości - funkcja pozwalająca na szybką komunikację wewnątrz aplikacji. Mamy do niej dostęp zarówno z poziomu zarządzania kontem, jak i pojedynczego ogłoszenia. W przypadku, gdy w bazie danych nie ma zarejestrowanej konwersacji pomiędzy danymi użytkownikami, pierwszą wiadomość pisze się w obiekcie typu modal, pokazującym się po kliknięciu glyphiconu koperty (envelop). Wtedy najpierw rejestrowana jest nowa “rozmowa”, a później dodana wiadomość. Następne kliknięcie koperty przenosi już do ekranu konwersacji, gdzie można dodawać dalsze wiadomości. Wraz z zapisaniem tekstu do bazy zmienia się odpowiednia flaga, co powoduje że u adresata ikona z kopertą będzie “migać” informując o nowej wiadomości. Gdy sprzedawca otrzyma zapytania do jednego ogłoszenia od większej liczby osób to po kliknięciu koperty pokaże mu się modal z listą wszystkich prowadzonych rozmów, ilością wiadomości w każdej oraz datą ostatniej z nich. W rekordzie z konwersacją z nieprzeczytaną wiadomością znajdzie się ponadto migający napis “NOWE!!!”.

Od	Ostatnia wiadomość	Ilość wiadomości
Test54	2018-03-06 15:31:55	1
Test96	2018-03-06 15:32:42	2

Zamknij

Rysunek 3.16: zrzut obiektu modal z listą aktywnych rozmów

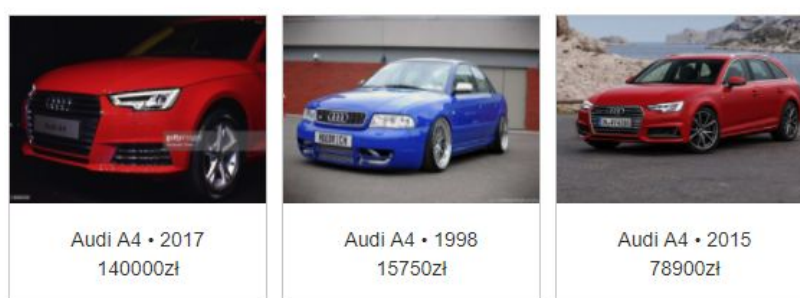


Rysunek 3.17: zrzut ekranu rozmowy

Powiązane treści - jest to element, który znajduje się zarówno na stronach ogłoszeń, jak i artykułów. W przypadku ogłoszeń wyświetlane są miniatury przenoszące do artykułów traktujących o przeglądany model. Jeżeli są to porady to auto musi być wyprodukowane w podanym w artykule przedziale lat. Ma to na celu zapobiegnięciu sytuacji gdy poruszany w tekście temat nie dotyczy starszych lub nowszych modeli. W przypadku testów i porównań przyjęty został dwuletni margines, jako że często w tym przedziale zmiany w pojazdach bywają kosmetyczne. W przypadku artykułów, powiązane treści to ogłoszenia aut, o których jest dany tekst. Zasada działania jest identyczna jak w odwrotnym przypadku. Jeżeli tego typu

treści jest więcej niż ustalony limit (domyślnie pięć) to pokazuje się przycisk przenoszący do stron z listą wszystkich dostępnych ogłoszeń/artykułów powiązanych z przeglądany temat. Zadaniem tej funkcji jest umożliwienie użytkownikom na łatwe zapoznanie się z materiałami publicystycznymi związanymi z wyszukiwanym autem, zapoznanie się z podobnymi modelami, czy zobaczenie opinii na jego temat. Podobnie z działaniem w drugą stronę. Moduł ten może również zostać łatwo rozszerzony w przyszłości na przykład o odnośniki do ewentualnego forum, czy też szybkie wyszukiwanie części do danego samochodu.

Powiązane ogłoszenia



Rysunek 3.15: zrzut widoku na przykładowe powiązane treści dla artykułu z poradami do Audi A4

3.5. Szczegóły techniczne

Od strony front-endu aplikacja była projektowana głównie z myślą o responsywności i robieniu pełnego użytku z dostępnej przestrzeni niezależnie od wielkości ekranu. Dzięki wykorzystaniu technologii Bootstrap było to proste w wykonaniu. Wszystkie elementy tworzone były w oparciu o zasadę 12 kolumn jaką cechuje się wymieniony framework. pozwoliło to zachować spójność i odpowiednie ułożenie poszczególnych elementów interfejsu względem siebie. Zupełnie inny wygląd aplikacji pojawia się na urządzeniach z szerokością ekranu poniżej 768 pikseli. Jak wspomniano przy opisie paska nawigacyjnego, ten element zostaje ukryty w takim przypadku, a dostęp do jego opcji otrzymujemy po użyciu przycisku z trzema poziomymi kreskami, który się w takiej sytuacji pojawia. Wiele obiektów zajmuje wtedy całą szerokość ekranu co ma ułatwić korzystanie z nich np. na telefonach. W systemie wielokrotnie i w różny sposób używane są ikony z pakietu Bootstrap tzw. Glyphicons. Polepszają czytelność niektórych obiektów poprzez oczywiste skojarzenia (koperta = wiadomości, kosz na śmieci = usuwanie itd.). Czasami są na tyle oczywiste, że nie

wymagają użycia dodatkowego tekstu obok. Opisy takich przycisków często umieszczono w formie tooltipów, czyli “chmurek” z tekstem. To rozwiązanie szczególnie przydatne okazało się przy liście ogłoszeń dostępnej z poziomu panelu *Moje konto*. Zamiast tekstu w pojawiają się tam miniatury zdjęć, więc dokładniej wiemy co znajduje się w treści ogłoszenia. Odnosnie kolorystyki całej aplikacji to postawiono na nowoczesność. Głównie jest to biel i szarość, pasek nawigacyjny oraz tekst są czarne, z kolei linki i przyciski w większości niebieskie. Jedynie te przyciski, które odpowiedzialne są za edycję oraz wysyłanie nowych wiadomości mają barwę żółtą, a za wysłanie formularzy czerwoną. Poprzez zastosowanie selektora *:hover* po najechaniu kursorem na dowolny przycisk lub link zmieni on kolor na turkusowy. Skorzystano z takiego rozwiązania ponieważ w wygląda bardzo dobrze oraz łatwo oznacza, które elementy aplikacji można kliknąć. W celu usprawnienia nawigacji oraz stworzenia ciekawych efektów zastosowano w projekcie trzy bardzo ciekawe pluginy Bootstrapa tzn. *affix*, *modal* i *carousel*. Będący zawsze u góry pasek nawigacyjny, wyskakujący dodatkowy fragment interfejsu oraz element robiący pokaz slajdów sprawiają, że aplikacja daje wrażenie mniej statycznej.

Od strony serwera, połączeniem z bazą danych oraz różnego rodzaju działaniami na danych zajmuje się PHP. Odpowiada on za wysyłanie zapytań do bazy MySQL, pobieranie i przetwarzanie odbieranych danych oraz zabezpieczanie systemu przed niepowołanymi działaniami np. poprzez funkcję *mysql_real_escape_string()*. Za jego pomocą następuje odpowiednie przetworzenie dodawanych do bazy zdjęć. Polega to na sprawdzeniu rozszerzenia oraz wielkości fotografii (przyjmowane są jedynie pliki *jpg*, *jpeg* i *png*, a maksymalny rozmiar to 5 MB. Dodatkowo zajmuje się zmianą nazw zdjęć, te bowiem muszą być unikalne. Składają się na nie: zaokrąglona wartość czasu - funkcja *microtime()*, połączona z id użytkownika oraz numerem kolejnego pliku. Poprzez PHP następuje również przesyłanie i odbieranie zmiennych umieszczanych w adresie URL.

3.6. Aspekty bezpieczeństwa

Aplikacje internetowe zawierające bazę danych oraz treści niedostępne dla standardowych użytkowników muszą być w odpowiedni sposób chronione. W tego typu systemach łatwo można utracić część lub nawet całość bazy lub ktoś może przejąć konto użytkownika.

Jednym z podstawowych rodzajów ataków jest SQL injection [23]. Polega on na wpisaniu kodu SQL w określonym miejscu aplikacji np. w formularzu. Standardowo tego typu dane nie są sprawdzane pod kątem zawartości. Wobec tego po użyciu konkretnego polecenia można narobić ogromnych szkód w systemie. Najprostszym przykładem jest usunięcie całej bazy danych lub zalogowanie się na konto administratora bez znajomości hasła. Z tego też powodu w aplikacji zastosowano funkcję *mysql_real_escape_string()*. Jest ona dedykowana do ochrony właśnie przed tego typu atakiem. Została użyta na każdej zmiennej, która następnie jest częścią zapytania SQL.

Kolejnym użytym zabezpieczeniem jest zdefiniowanie typu dokumentu na każdej stronie - `<!DOCTYPE html>`. Dzięki temu przeglądarka od razu wie z jakim typem dokumentu ma do czynienia. W innym przypadku mogłaby wyrenderować stronę w trybie kompatybilności z dawnymi konstrukcjami składniowymi co może prowadzić do wycieków danych lub ataków XSS (cross-site scripting) [24].

W aplikacji często korzysta się z Content Delivery Network, czyli pobierania zewnętrznych bibliotek i frameworków bezpośrednio przez przeglądarkę. W razie włamania na adres, z którego następuje pobieranie i podmiany plików, do aplikacji będą trafiały te właśnie pliki. W związku z tym użyty został mechanizm Subresource Integrity - SRI. Polega na dodaniu do tagów `<link>` oraz `<script>` atrybutu *integrity* przechowującego hash danego pliku. Jeśli jest on różny od tego z pobranego pliku to taki plik nie zostanie wykonany [24].

Ważne jest również odpowiednie zabezpieczenie haseł użytkowników, gdyż te mogą zostać “podслuchane”. W aplikacji hasło szyfrowane jest za pomocą funkcji skrótu SHA-1. Samo to jednak nie wystarczy, dlatego dodatkowo generowana jest tzw. sól. Jest to unikalna losowo generowana wartość, która również podlega hashowaniu, a następnie łączona jest z wcześniej zaszyfrowanym hasłem.

Dodatkowo na każdej podstronie wymagającej zalogowania/określonych uprawnień sprawdzane jest, czy dany użytkownik ma zapewniony do niej dostęp. Następuje to za pomocą zmiennych typu `$_SESSION` oraz zapytań do bazy danych. Działa to również w drugą stronę tzn. zalogowany użytkownik nie może przejść do ekranu tworzenia konta bez uprzedniego wylogowania.

4. Testowanie

4.1. Testy interfejsu

Podstawowym celem tych testów było określenie, poprawności wyświetlania poszczególnych elementów niezależnie od wielkości okna (ekranu). Dzięki uruchomieniu aplikacji na dużym wyświetlaczu można było powoli zmniejszać okno przeglądarki, dla każdej kolejnej podstrony. Dzięki temu sprawdzono, jak aplikacja zachowywała by się na różnych typach urządzeń. Na dużych ekranach (monitory komputerów stacjonarnych i laptopów) poszczególne obiekty oraz tekst były bardzo czytelne i nie nachodziły na siebie. Warto wspomnieć, że aplikacja testowana była na przeglądarce Google Chrome z domyślnym ustawieniem powiększenia - 100%.

Powolne zmniejszanie wielkości okna wpływało na rozmiar elementów zgodnie z założeniami. Tekst, w niektórych miejscach ulegał zawijaniu, aby nie nachodził na inne obiekty, z kolei pola tekstowe typu *input* czasami przestawały znajdować się w rzędach i rozmieszczały się jedno pod drugim. Nie do końca odpowiednio wyglądało zachowanie zdjęć na stronie głównej oraz na listach z ogłoszeniami/artykulami. Dzięki zastosowaniu klasy *img-responsive* frameworka Bootstrap, zmniejszały się one wraz z resztą obiektów, jednak następowało to trochę szybciej niż w innych przypadkach. Powodowało to sytuację, w której tuż przed przejściem systemu w tryb “mobilny” miniatury były zbyt małe w stosunku do reszty wyświetlanych obiektów. Właśnie ten wspomniany moment, gdy szerokość ekranu zbliża się do granicznych 768 pikseli sprawia najgorsze wrażenie i wymaga dodatkowej pracy.

Sytuacja zmienia się jednak po przekroczeniu tej wartości. Poszczególne elementy aplikacji takie jak pola tekstowe, rekordy w tabelach, czy tekst zajmują całą szerokość dostępnego ekranu. W związku z tym obiekty, które wcześniej rozmieszczone były obok siebie teraz znajdują się jeden pod drugim. Wszystko ma jednak odpowiedni położenie, a sama obsługa aplikacji staje się wygodniejsza niż w “pełnej” wersji. Nie przeprowadzono testów na urządzeniu mobilnym, jednak interfejs sprawia wrażenie łatwego do obsługi jedną ręką na małym ekranie.

4.2. Testy funkcjonalności

Na tym etapie testowania sprawdzano poprawne działanie aplikacji oraz zachowanie systemu w przypadku jego nieprawidłowego użycia.

Podczas tworzenia nowego konta sprawdzana jest dostępność danego adresu email. Jeśli użytkownik zignoruje otrzymaną informację to i tak nie będzie mógł utworzyć takiego konta. Ponadto przy wpisywaniu hasła, gdyby nie spełniało wymogu 6 znaków, pojawi się odpowiedni alert i nie zniknie aż do wpisania dłuższego hasła. Pole na powtórzenie wybranego hasła jest domyślnie zablokowane aż pierwszy raz wpisane hasło będzie spełniało ten warunek.

Za pomocą typu *email* w atrybucie *input* HTML, sprawdzane jest także czy został wpisany poprawny schematycznie adres mailowy. We wszystkich miejscach na podawanie liczby np. rok produkcji, pojemność silnika, zastosowano typ *number*. Zapobiega to podawaniu nieprawidłowych typów danych co mogłoby powodować błędy w zapytaniach SQL.

Przetestowano także możliwość użycia przycisku “wstecz” po wylogowaniu. Przenosi on na wcześniej odwiedzone podstrony jednak w trybie gościa (nie następuje powrót do zalogowanego konta).

Nie ma także możliwości znalezienia się nieuprawnionego użytkownika w miejscu, do którego nie powinien mieć dostępu. Przykładowo podanie adresu URL *usun.php?id=x*, gdzie *x* to id ogłoszenia nie wystawionego przez daną osobę nie spowoduje usunięcia tej treści.

Kolejnym etapem testów było sprawdzenie filtrów, zarówno w postaci formularza, jak i słów kluczowych. W pierwszym przypadku zaznaczanie/podawanie kolejnych parametrów poprawnie generowało listę ogłoszeń jedynie z ofertami spełniającymi określone warunki. Przy wpisywaniu słów kluczowych również wyświetlały się tylko ogłoszenia zawierające dane słowa w odpowiednich miejscach. W momencie gdy aplikacja nie zawiera żadnych ogłoszeń spełniających podawane wytyczne pojawia się odpowiedni komunikat oraz możliwość ponownego wyszukania lub wyświetlenia pełnej listy.

Następnie obiektem testów było usuwanie i edytowanie treści. Decydując się na usunięcie artykułu/ogłoszenia z bazy kasowane były wszystkie powiązane z nimi treści. Przykładowo razem z danymi ogłoszenia usuwane były wiadomości pomiędzy zainteresowanymi, a wystawiającym, lista rozmów, zdjęcia oraz wpisy w tablicy

“obserwowanych”. Jedyny problem stanowiły fotografie przechowywane w serwerowym katalogu. Nie zawsze bowiem następowało ich usunięcie, nawet pomimo zastosowania odpowiednich funkcji (*unlink()* w PHP [24]). Odnośnie edycji treści to opcja ta nie różni się specjalnie od dodawania nowych ogłoszeń i artykułów z tym, że pobierane z bazy i wyświetlane w polach tekstowych są aktualne wartości danych. Niezależnie od dokonanych zmian zapisanie edytowanej treści spowoduje aktualizację bazy danych.

W pozostałych aspektach nie stwierdzono żadnych problemów z użytkowaniem aplikacji.

5. Rozwój aplikacji

5.1. Forum

Ważnym aspektem serwisów, niezależnie od tematyki, jest możliwość stworzenia miejsca, gdzie użytkownicy mogliby “spotykać” się i prowadzić dyskusje na określone tematy. Mogłyby to być zarówno standardowe wymiany opinii na jakiś temat, prośby o pomoc od bardziej doświadczonych w temacie osób, czy przekazywanie własnych rozwiązań problemów. Istnieje wiele forów zrzeszających fanów motoryzacji, jednak zawsze warto mieć możliwość dostępu do takiej opcji w obrębie jednej aplikacji. Bardzo ciekawym dodatkiem do tego rozwiązania byłoby zwiększenie ilości i rodzajów powiązanych treści. Przykładowo z obrębu danego ogłoszenia istniałby dostęp do wątków na forum związanych z tym modelem, podobnie jak obecnie z powiązanymi artykułami.

5.2. Obsługa różnych rodzajów ogłoszeń

Obecnie aplikacja jest w pełni poświęcona samochodom osobowym. Należałoby poszerzyć zakres obsługiwanych ogłoszeń, jak i tematykę publikacji o motocykle, samochody ciężarowe, maszyny budowlane, sprzęt rolniczy oraz części samochodowe. Wymagałoby to odpowiedniego dostosowania filtrów, stron poszczególnych ofert, a przede wszystkim kreatora nowych ogłoszeń. Zwiększyłoby to zainteresowanie aplikacją oraz pozwoliło na kolejną rozbudowę łączenia się treści np. przejście do ekranu ofert z pasującymi oponami z poziomu ogłoszenia danego auta.

5.3. Dostosowanie do innych przeglądarek

Aplikacja od początku testowana była na przeglądarce Google Chrome i właśnie tam nie występują żadne problemy z interfejsem czy funkcjami. Dodatkowe testy przeprowadzone na przeglądarkach Internet Explorer, Microsoft Edge oraz Firefox, choć nie wykazały problemów z funkcjonowaniem, to niekiedy ujawniły niepoprawne wyświetlanie elementów interfejsu. Dodatkowo na Internet Explorer zdjęcia na miniaturach były nieostre co psuło wrażenia z użytkowania. Na rynku dostępnych jest wiele ofert przeglądarek i dostosowanie aplikacji do każdej z nich to wielkie wyzwanie, ale na tych najczęściej używanych powinna być ona wyświetlana w pełni poprawnie.

5.4. Inne

Dodatkową możliwością rozbudowy mogłoby być pobieranie ogłoszeń z innych aplikacji tego typu. Niektóre serwisy udostępniają swoje zasoby co pomogło by powiększyć dostępną bazę danych. Jest to funkcja przydatna szczególnie na początku istnienia takiego systemu ponieważ użytkownicy musieliby zapoznać się z daną aplikacją zanim zaczęliby dodawać swoje treści. Startowa dostępność bazy ogłoszeń mogłaby pomóc w uzyskaniu rozgłosu i zapewnieniu sobie klientów.

Dodanie tłumaczeń na inne języki również mogłoby zwiększyć popularność aplikacji. W Polsce nie jest rzadkością kupowanie używanych pojazdów przez obcokrajowców, a dla nich serwis oferujący ojczysty język mógłby być bardziej zachęcający.

Następną możliwością byłoby dodanie do interfejsu graficznego większej ilości ciekawszych np. animowanych elementów. Przykładem są zastosowane pluginy frameworka Bootstrap. Użycie takiego typu elementów nie tylko poprawia ogólne wrażenia z użytkowania, ale również może ułatwić lub usprawnić korzystanie z określonej funkcjonalności.

6. Podsumowanie

Wszystkie cele i założenia postawione na początku tworzenia tej pracy inżynierskiej zostały zrealizowane. Stworzona aplikacja webowa pozwala na łatwe i szybkie dodawanie oraz przeszukiwanie ogłoszeń samochodów osobowych, ale również zawiera moduł publicystyczny, który nie jest całkowicie odrębny od reszty aplikacji. W projekcie zawarto wszystkie najpotrzebniejsze elementy serwisów tego typu wykonane w sposób ciekawy i z użyciem nowoczesnych, jak i starszych lecz wciąż popularnych technologii. Zastosowano w nim różnego rodzaju zabezpieczenia i ograniczenia, aby użytkownicy mieli bezpieczny dostęp do swoich treści i nie mogli doprowadzić do niepożądanych sytuacji w systemie.

Aplikacja umożliwia już pełne działanie jednak podczas jej implementacji nie uniknięto drobnych błędów. Również potencjalne możliwości rozwoju projektu wskazują na to, że włożona musi zostać dodatkowa praca, aby osiągnąć w pełni rozwinięty, nowoczesny i niebanalny serwis ogłoszeniowy, który będzie oferował jednak o wiele więcej.

7. Bibliografia

- [1] Oracle. *Java Technologies for Web Applications*. 2006. URL: <http://www.oracle.com/technetwork/articles/java/webapps-1-138794.html>
- [2] Autorzy Wikipedii. *Serwlet*. 2017. URL: <https://pl.wikipedia.org/wiki/Serwlet>
- [3] Autorzy Wikipedii. *AngularJS*. 2017. URL: <https://pl.wikipedia.org/wiki/AngularJS>
- [4] CentrumXP. *ASP.NET - co to właściwie jest?*. 2005. URL: <http://www.centrumxp.pl/dotNet/142,01-ASPNET-co-to-wlasciwie-jest.aspx>
- [5] MDN web docs. *Django introduction*. 2017. URL: <https://developer.mozilla.org/en-US/docs/Learn/Server-side/Django/Introduction>
- [6] Dokumentacja Perl. 2017. URL: <http://perldoc.perl.org/perlintro.html>
- [7] Autorzy Wikipedii. *Ruby on Rails*. 2017. URL: https://pl.wikipedia.org/wiki/Ruby_on_Rails
- [8] Mirosław Zelent. *Kurs HTML*. 2017. URL: <http://miroslawzelent.pl/kurs-html/>
- [9] Twitter. *Bootstrap*. Wer. 4.0.0. Lic. MIT. URL: <https://getbootstrap.com/>
- [10] Mirosław Zelent. *Kurs JavaScript*. 2017. URL: <http://miroslawzelent.pl/kurs-javascript/>
- [11] Autorzy Wikipedii. *AJAX*. 2017. URL: <https://pl.wikipedia.org/wiki/AJAX>
- [12] Marcin Domański, Szama Szamson. *Kurs JavaScript - AJAX w jQuery*. 2017. URL: <http://kursjs.pl/kurs/jquery/jquery-ajax.php>
- [13] Apache Software Foundation. *Apache HTTP Server*. Wer. 2.4.29. Lic. Apache License. URL: <https://httpd.apache.org/>
- [14] Leszek Krupiński. *Kurs PHP*. 2007. URL: <http://phpkurs.pl/podstawy/>
- [15] Autorzy Wikipedii. *PHP*. 2017. URL: <https://pl.wikipedia.org/wiki/PHP>
- [16] Oracle. *MySQL*. 2017. URL: <http://www.oracle.com/technetwork/database/mysql/index.html>
- [17] Łukasz Dembiński. *Architektury klient-serwer*. 2017. URL: <http://dembol.org/blog/projektowanie/katalog-wzorcow-architektonicznych/wzorcow-perspektywy-alokacji/architektury-rozproszone/architektury-klient-serwer/>
- [18] Autorzy Wikipedii. *Architektura trójwarstwowa*. 2017. URL: https://pl.wikipedia.org/wiki/Architektura_trójwarstwowa

- [19] w3schools. *Bootstrap Affix*. 2018. URL: https://www.w3schools.com/bootstrap/bootstrap_affix.asp
- [20] Jan Kovarik. *Sharp and clean symbols*. 2018. URL: <http://glyphicons.com/>
- [21] w3schools. *Bootstrap Carousel*. 2018. URL: https://www.w3schools.com/bootstrap/bootstrap_carousel.asp
- [22] w3schools. *Bootstrap Modals*. 2018. URL: https://www.w3schools.com/bootstrap/bootstrap_modal.asp
- [23] Daniel Frużyński. *Atak SQL Injection*. 2018. URL: http://www.poradnik-webmastera.com/artykuly/bazy_danych/sql_injection.php
- [24] Michał Bentkowski. *Jak w prosty sposób zwiększyć bezpieczeństwo aplikacji webowej*. 2016. URL: <https://sekurak.pl/jak-w-prosty-sposob-zwiekszyc-bezpieczenstwo-aplikacji-webowej/>
- [25] Dokumentacja PHP. 2018. URL: <http://php.net/manual/en/function.unlink.php>