

Diamond Cars

Dokumentacja projektowa

Programowanie portali społecznościowych



Wykonał: Mateusz Borkowski

# Opis projektu

Projekt zakłada stworzenie aplikacji webowej będącej połączeniem osób sprzedających auta,   
a osobami, które wyrażają chęć ich kupna. Dodatkowo został wprowadzony moduł forum, gdzie każdy zarejestrowany użytkownik może dodać pytanie związane z danym samochodem oraz komentować pytania innych użytkowników

Serwis udostępnia rejestrację użytkowników – Ci użytkownicy będą mogli wykonywać powyższą czynność oraz dodawać własne samochód na sprzedaż. Dodane przez użytkowników oferty będą znajdować się w zasięgu tylko danej aplikacji. Wszystkie powyższe funkcjonalności są zaadaptowane dla zarejestrowanego, a następnie zalogowanego użytkownika. Osoba niezalogowana może tylko czytać posty, nie ma możliwości ich komentowania, ani dodawania. Zarówno postów jak i ofert samochodowych również nie może dodawać. Oprócz oglądania ofert i postów każdy użytkownik, zalogowany lub nie może wyszukiwać samochody, które są zamieszczone na innych portalach aukcyjnych tj. otomoto.pl oraz olx.pl. Funkcja ta jest oparta na „scraperze”. Po wypełnieniu przez użytkownika danego formularza „scraper”-pajączek szuka na danych portalach ofert wg podanych przez użytkownika samochodów, a następnie wyświetla je na ekranie.

Społecznościowe aspekty serwisu, które pozwalają na interakcję pomiędzy użytkownikami   
i tworzonymi przez nich treściami to :

* dodawanie postów
* komentowanie postów
* dodawanie ofert z samochodami
* przeglądanie ofert z samochodami
* wyszukiwanie aut z powyższych portali
* profil użytkownika

# Technologie i narzędzia:

* Python, Django, Bootstrap, HTML, CSS, Beautiful Soup

# Architektura:

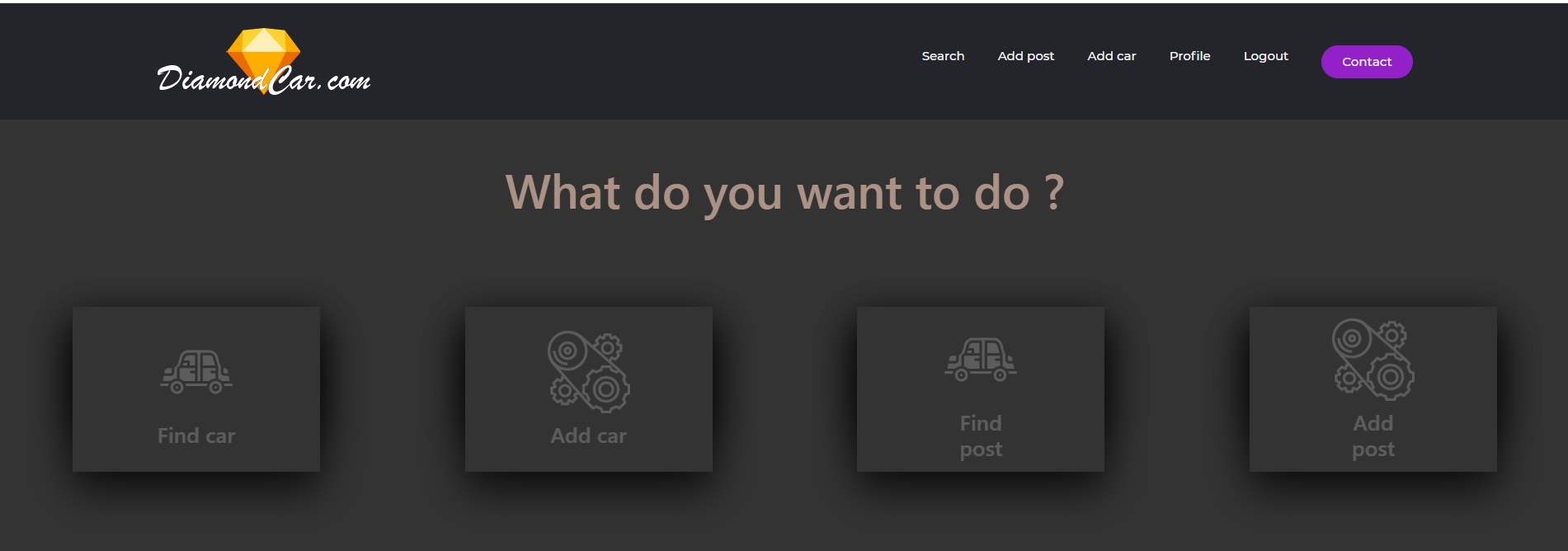
* REST

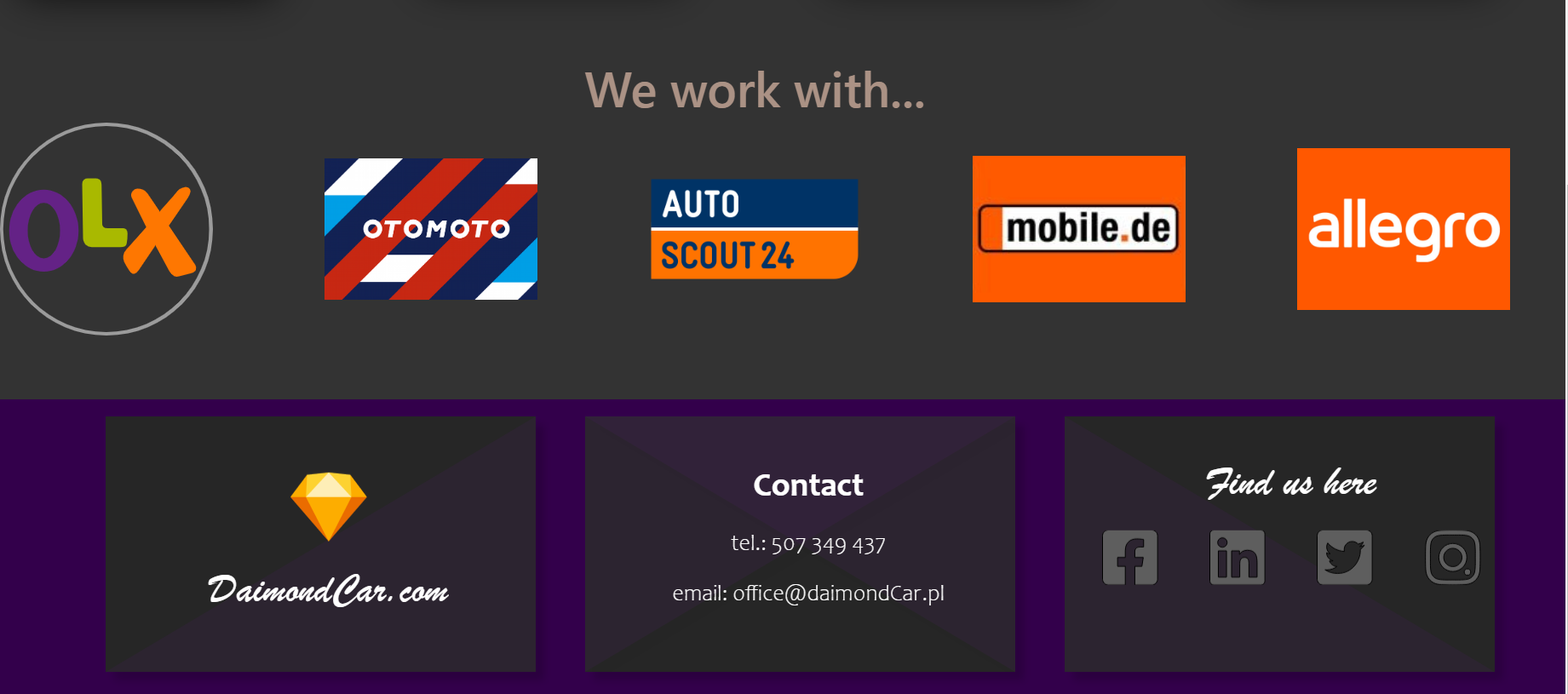
# Repozytorium projektu:

* <https://github.com>

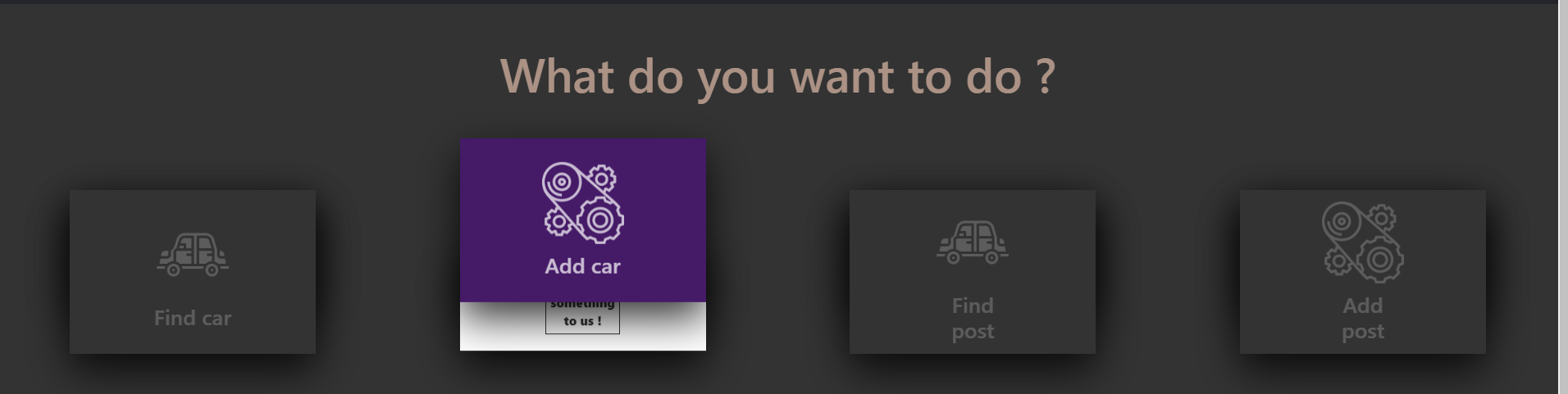
# Projekt interfejsu

Projekt interfejsu jest przejrzysty i smukły. Do tego projektu został wykorzystany Bootstrap. HTML, CSS. Główny interfejs przedstawia użytkownikowi 4 najważniejsze operacje tj.: dodawanie postów, szukanie postów, dodawanie samochodów oraz szukanie samochodów. Oczywiście jeśli niezalogowany użytkownik wybierze jedną z opcji, automatycznie zostanie przekierowany do formularza logowania. Dana osoba jeśli nie będzie jeszcze zarejestrowana, a serwis będzie chciał, aby się zalogowała to zaraz obok formularza logowania znajduje się odnośnik do formularza z możliwością rejestracji.

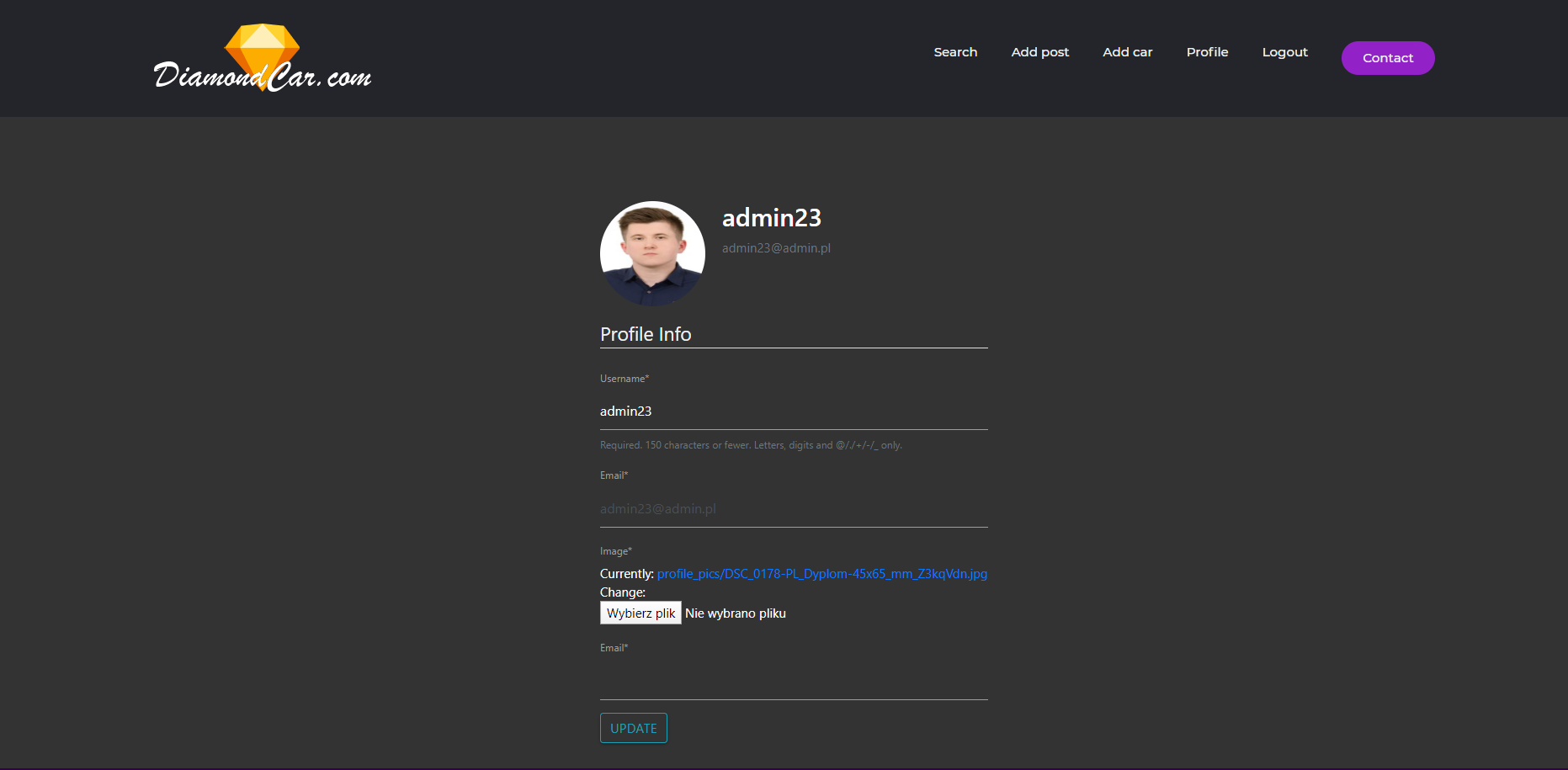




Po najechaniu przez kursor na dany kafelek, zmienia on kształt oraz kolor



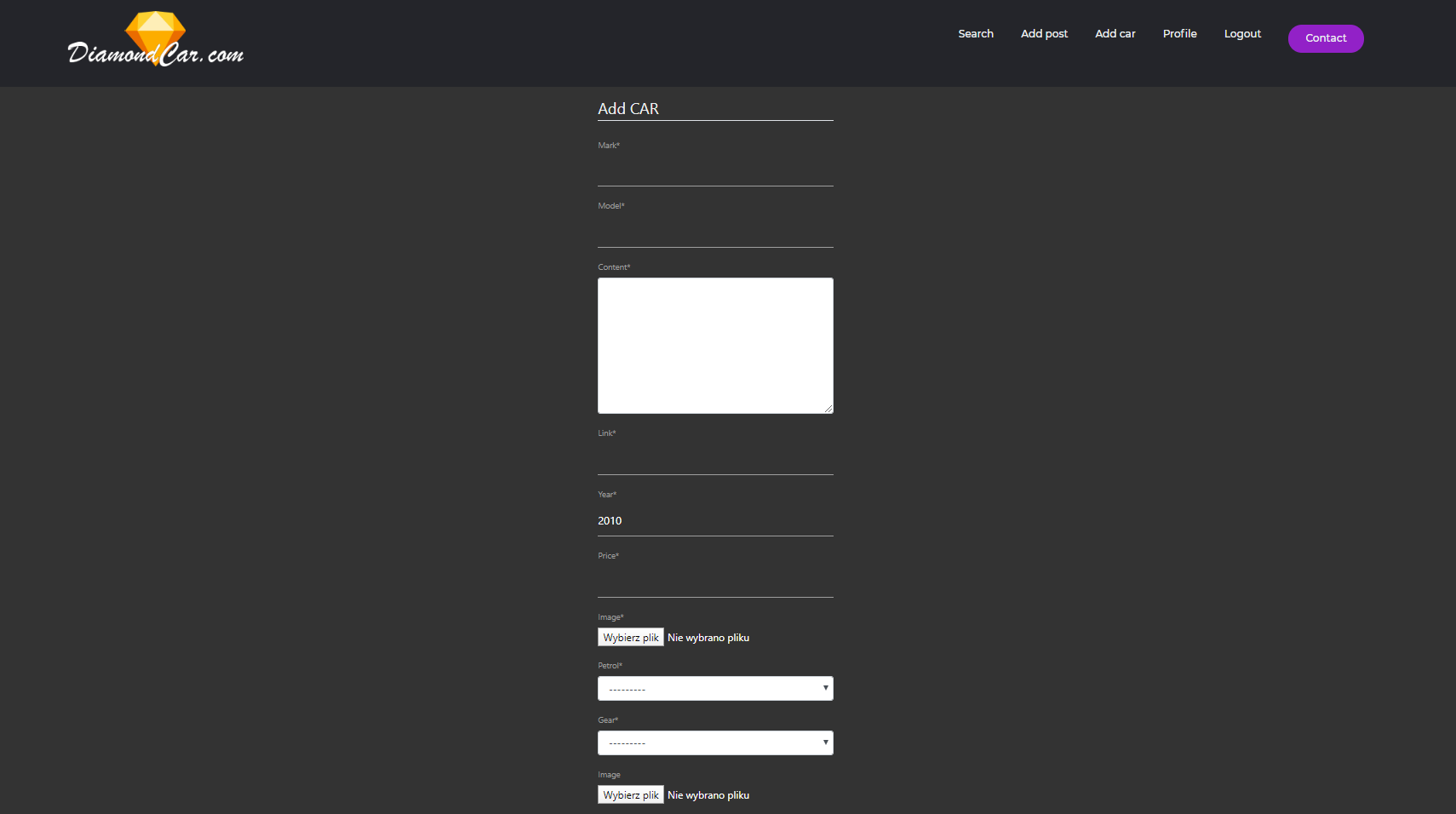
# Profil zalogowanego użytkownika:



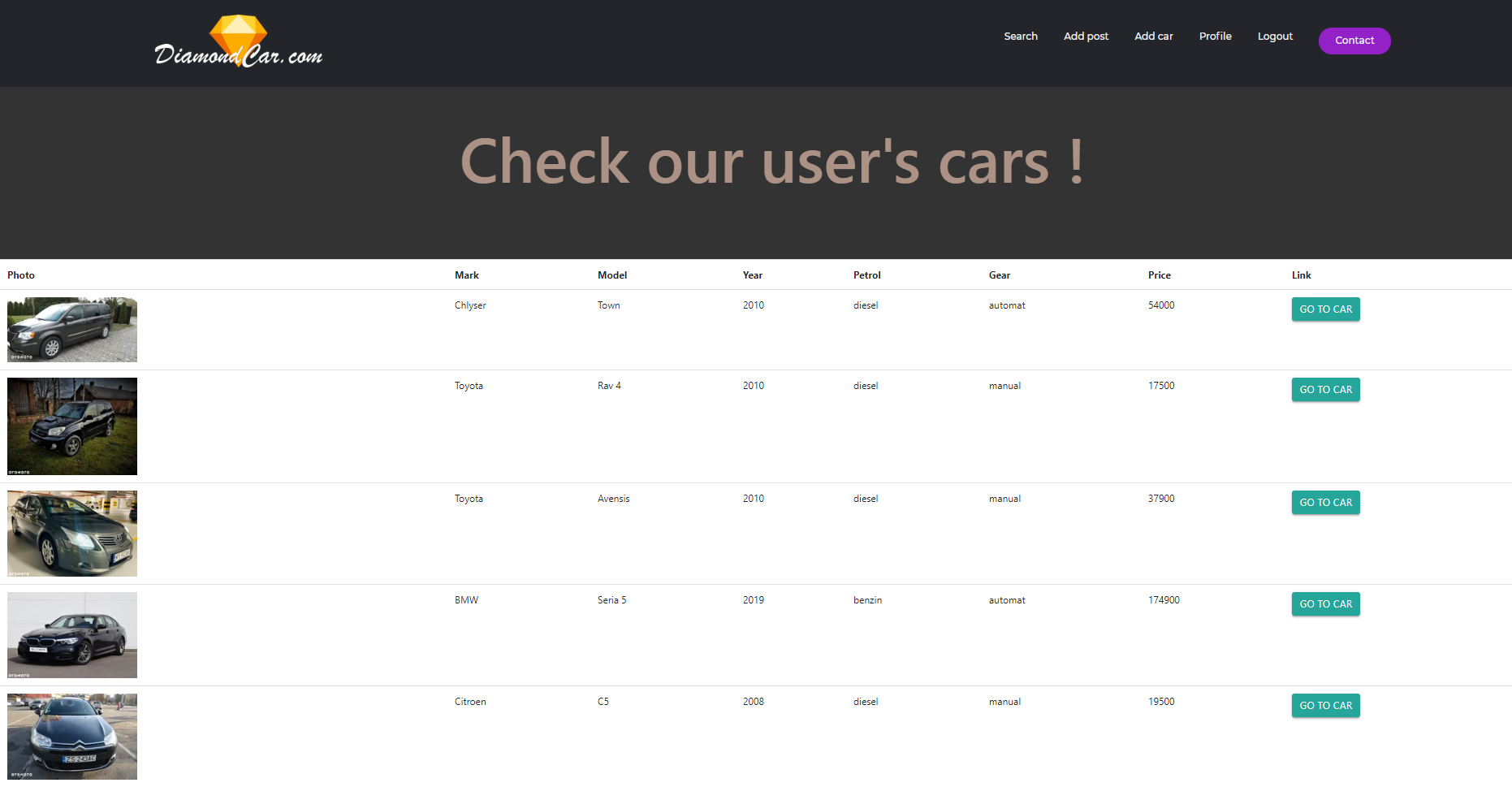
# Interfejs dodawania postu:

# 

# Interfejs dodawania postu samochodu:

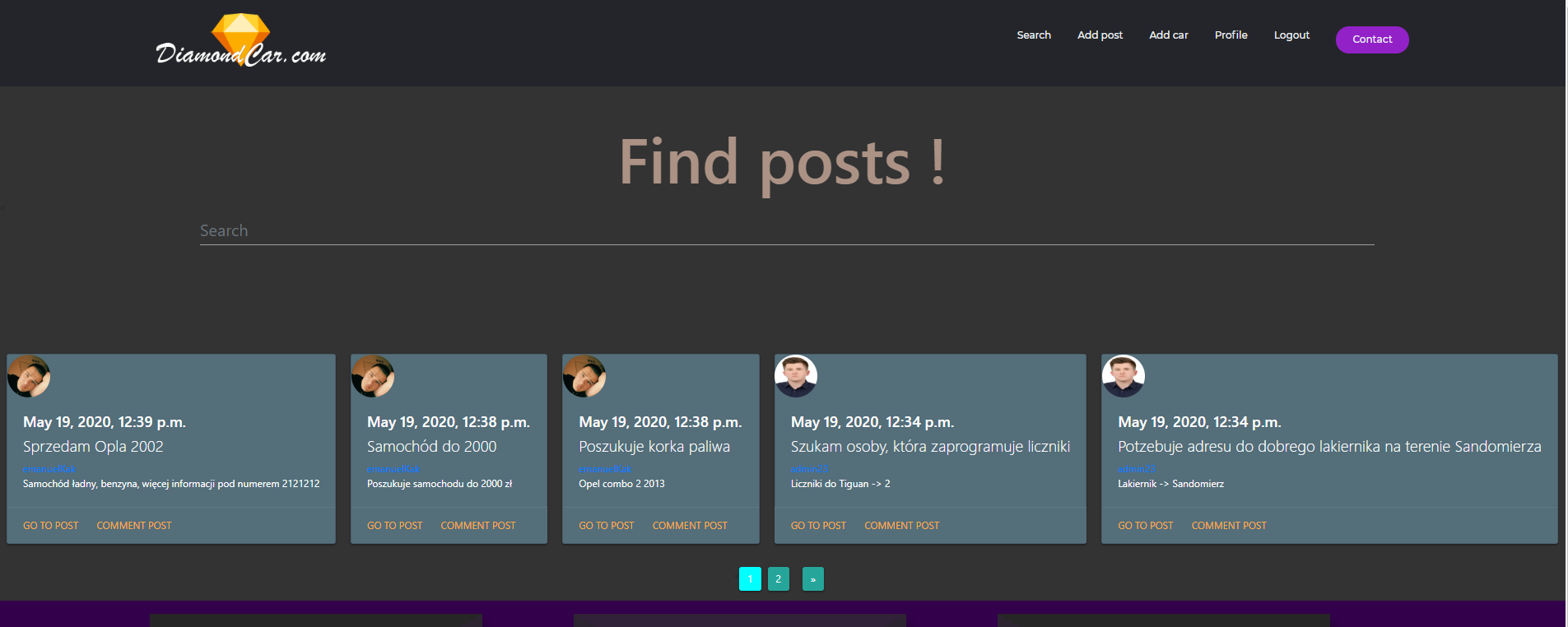


# Interfejs wszystkich dodanych samochodów przez użytkowników:



Każdy użytkownik może przypisać do danego auta link do aukcji z innego portalu z tym autem. Osoba zainteresowana dzięki temu może za pomocą kliknięcia w przycisk do tej oferty przejść.

# Interfejs wszystkich dodanych postów przez użytkowników:



Do każdego interfejsu związanego z szukanie postów czy aut zastosowana jest paginacja oraz moduł wyszukiwania. Wyświetlanie postów zaimplementowane została za pomocą ‘flex-boxa’. Dzięki temu wyświetlanie obiektów oraz działanie całej aplikacji jest w formule RWD – Relative Web Design.

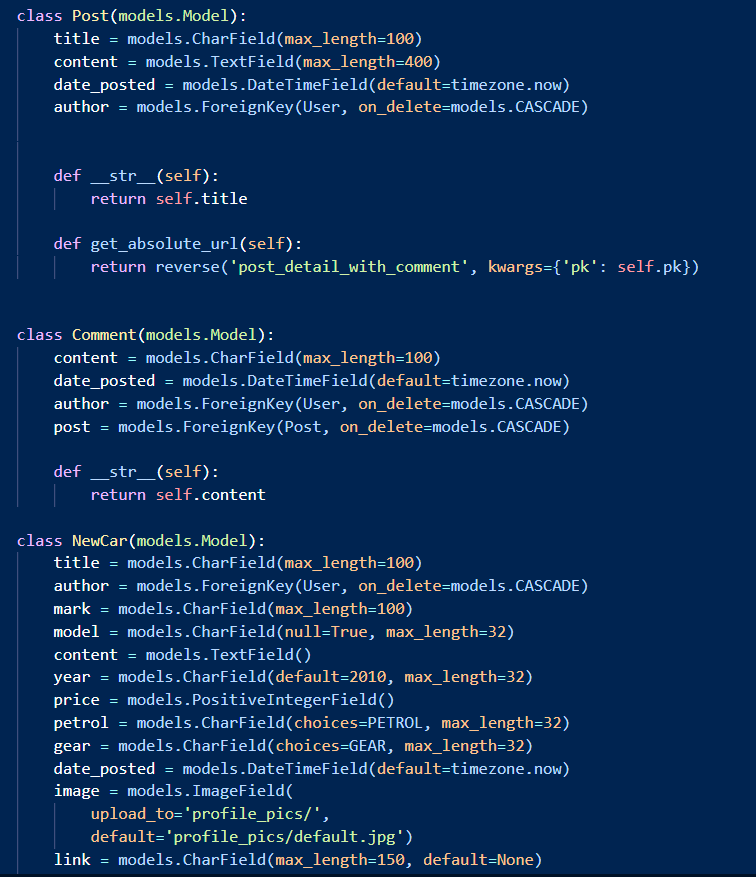


# Interfejs wnętrza postu oraz komentarzy do danego posta:

# 

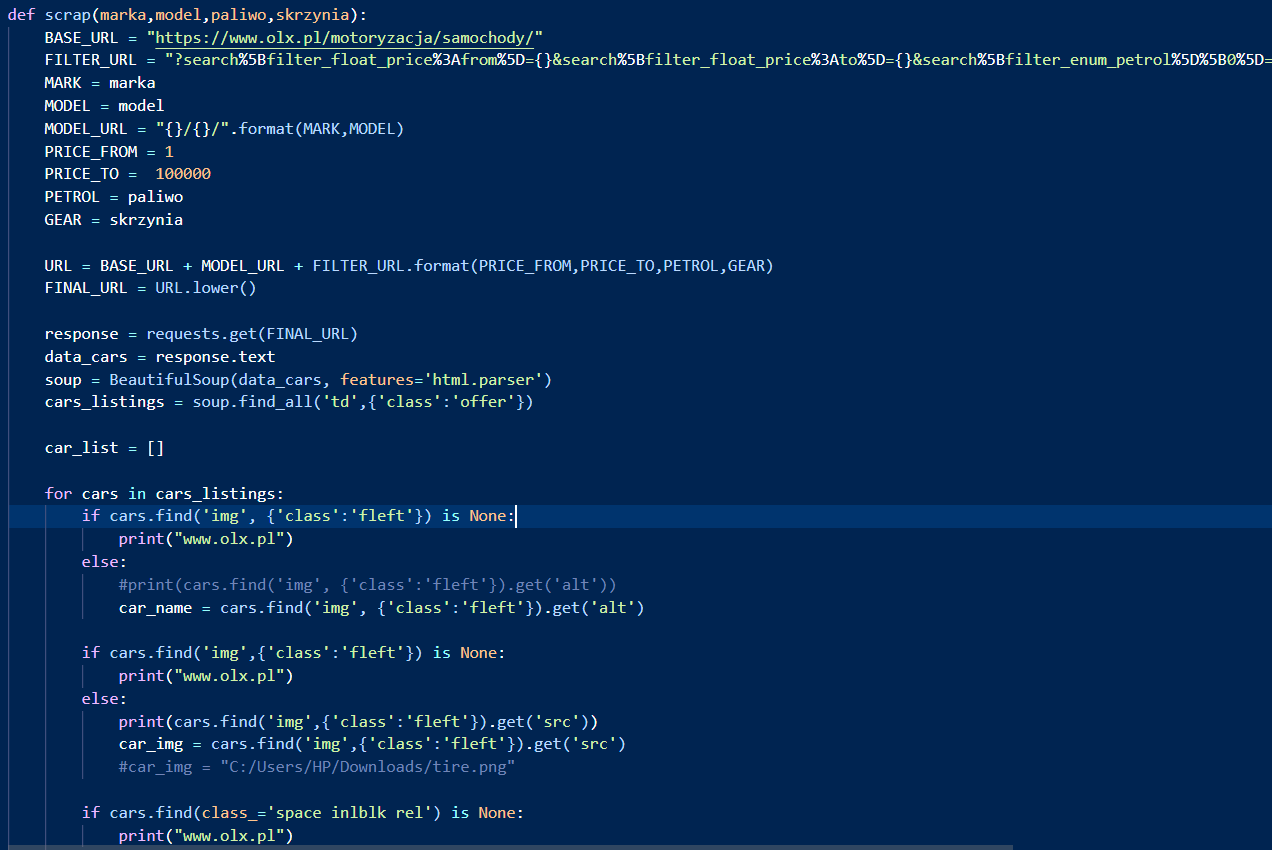
# Część biznesowa :

Modele obiektów zastosowane w aplikacji:



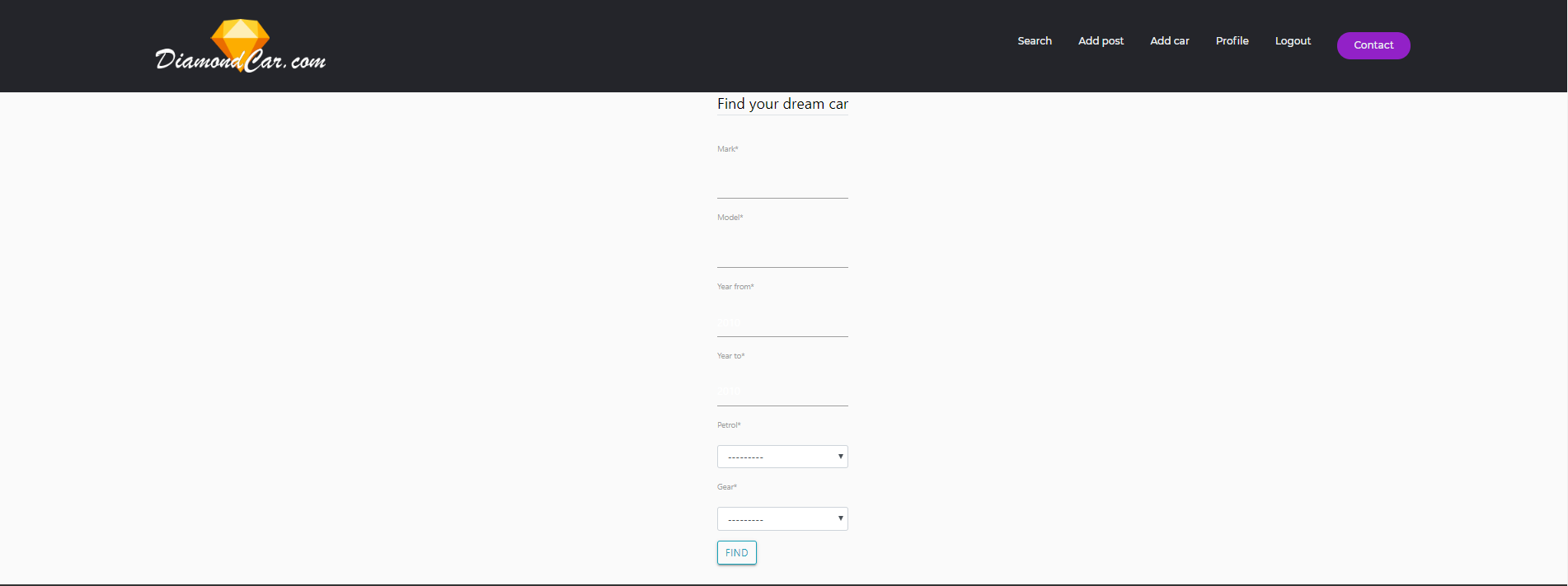
Powyższe modele odnoszą się do Postów, Komentarzy oraz dodawanych przez użytkowników samochodów.

# Scraper – otomoto – olx



Powyższy skrypt znajduje się w oddzielnym pliku oraz jest zaimportowany jako biblioteka do głównego pliku, określającego pracę serwisu.

Funkcja Scrap w danym skrypcie przyjmuje 4 argumenty, które są pobierane przez użytkownika   
z formularzu pod zakładką Search.



# Wynik SCRAPERA

Powyższy skrypt zbiera informacje z wysłanego przez niego URL do wszystkich



Widzimy wszystkie interesujące nas dane oraz odnośnik do danej aukcji, która znajduję się na innym portalu. Po wciśnięciu przycisku zostaniemy przeniesieni do tej samej oferty, lecz na innym serwisie, gdzie dowiemy się więcej na temat interesującego nas auta.

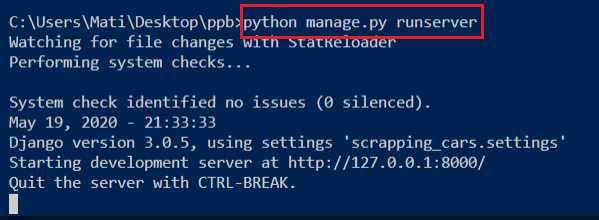
# Konfiguracja i uruchomienie projektu

## Sposób uruchomienia aplikacji

Aby uruchomić aplikację konieczne jest zainstalowanie wyżej określonych programów oraz środowiska uruchomieniowego. Do tego należy zainstalować środowisko Python >=3.6

Projekt był tworzony w Visual Studio Code, które jest jednym z lepszych środowisk dla taki projektów. Większość rzeczy, które są potrzebne instalują się same.

Po zaimportowaniu projektu należy uruchomić poniższe komendy, a projekt zbuduje i uruchomi się sam.



# Wnioski:

Tworzenie portalu pod kątem gromadzenia społeczności wymaga wkładu pracy głównie w aspekcie społecznościowym. Samo tworzenie aplikacji nie jest wygórowanym wyzwaniem. Aby zatrzymać użytkownika, zainteresować go treścią i sprawić by chciał tworzyć społeczną treść trzeba czegoś więcej. Dlatego tez główną zaletą tego serwisu jest oszczędność czasu. Dzięki skryptowi, który przeszukuje nam wiele portali jednocześnie, co lepsze przeszukuje tak dużo za pomocą jednego formularza. Wszyscy znamy dzisiejszy problem ze znalezieniem dobrego i wymarzonego dla nas auta. Godziny przeszukiwań wielu portali, wypełnianie tych samych formularzy, przeglądanie mnóstwa ofert. Tutaj mamy wszystko w jednym miejscu.

Dzięki temu przedmiotowi i wykonanemu do jego zaliczenia projektowi z pewnością zdobyłem wiedzę o projektowaniu portali społecznościowych i będę ją wykorzystywać w pracy zawodowej.