

Dokumentacja projektu

Wypożyczalnia krótkoterminowa samochodów osobowych

Autorzy:

Bartosz Kustra

Tomasz Nadrowski

Adrian Osajda

Michał Rudawski

Wrocław, 2020/2021

Spis treści

1. Wstęp.....	3
2. Charakterystyka projektu.....	3
3. Funkcjonalności czynne	4
4. Funkcjonalności bierne	9
5. Projekt.....	13
6. Zgodność wersji systemu oraz aplikacji	14
7. Testy systemu	15
8. Dokumentacja użytkownika.....	17
9. Bibliografia	19
Dodatek 1:.....	20

1. Wstęp

Wybór tematu projektu nie był łatwy, zważywszy na nieograniczoną liczbę możliwości. Ostateczna tematyka została sprecyzowana i dotyczy ona wypożyczalni krótkoterminowej samochodów osobowych.

Obecnie w dobie pandemii COVID-19 wiele osób, nie będących w stanie pracować zdalnie, musi wciąż w przemieszczać się do stacjonarnego miejsca pracy (biura, placówki etc.). Nie wszystkie z tych osób posiadały wcześniej własny środek transportu, a wiedzą powszechną jest, iż korzystanie z usług komunikacji miejskiej nie jest obecnie najbezpieczniejszą z alternatyw. Tutaj z pomocą powinny przyjść wypożyczalnie samochodów miejskich – klient po opłaceniu takiego pojazdu zyskuje czas potrzebny na znalezienie, zakup własnego samochodu, nie tracąc przy tym możliwości bezpiecznego (dla siebie i bliskich), sprawnego przemieszczania się.

Dane napływające przez ostatnie miesiące z tego sektora finansowego zdają się być nieubłagane. Jak tłumaczy w swoim artykule Polski Związek Wynajmu i Leasingu Pojazdów, ilość wypożyczanych długoterminowo samochodów spadła w znaczący sposób w trakcie i po „lockdownie”. Istotną przewagą wynajmu krótko- nad długoterminowym jest fakt mniejszych zobowiązań finansowych, co powinno szybciej przyciągnąć wcześniej utraconą klientelę.

Wcześniej opisana sytuacja rynkowa powinna, wbrew pozorom, ułatwić rozpoczęcie działalności wypożyczalni krótkoterminowej aut w większych miastach. Salony samochodowe chcąc zredukować straty, byłyby skłonne do sprzedaży floty po znacznie bardziej korzystnych cenach. Środki przeznaczone na inwestycję w biznes mogłyby więc być lepiej zagospodarowane w strukturę informatyczną całego przedsięwzięcia.

Wynajem krótkoterminowy według Polskiego Związku Wynajmu i Leasingu Pojazdów obejmuje umowę zawieraną na maksymalny okres: 30 dni. Taka właśnie możliwość świadczona jest przez analizowaną w tym projekcie firmę.

2. Charakterystyka projektu

Projekt dotyczy połączenia kilku aspektów, często wykorzystywanych obecnie technologii w celu zbudowania kompletnego systemu, który składałby się z kluczowej dla reszty elementów bazy danych, składającej się z wielu tabel, połączonych ze sobą pewnymi relacjami. Oprócz wspomnianych zbiorów informacji, w celu graficznej reprezentacji niezbędne było stworzenie odpowiedniego środowiska front-endowego, które w tym przypadku reprezentowane jest przez stronę internetową.

Głównym celem projektu jest umożliwienie użytkownikowi dokonania wynajmu krótkoterminowego wybranego przezeń samochodu osobowego. Istnieje jednak wiele udogodnień, dzięki którym możliwe jest sprecyzowanie wymagań dotyczących pojazdu silnikowego. Oprócz wyświetlenia na stronie internetowej wszystkich dostępnych samochodów, jedną z funkcjonalności jest ustawienie konkretnych filtrów z obszaru:

- Segmentu,
- Typu nadwozia,

- Przekładni skrzyni biegów,
- Rodzaju paliwa,
- Ilości miejsc pasażerskich.

Ponadto każdy z samochodów znajdujących się we flocie zawiera swoje szczegółowe informacje, które mogą zostać wyświetlone po wejściu w profil auta.

Po zapoznaniu się ze wszystkimi informacjami o pojeździe, oraz opcjonalne udanie się na stronę autocentrum.pl w celu poszerzenia wiedzy, możliwe jest przejście do arkusza wynajmu. Zaznacza się w nim interesujący klienta typ wypożyczenia wraz z wszystkimi niezbędnymi datami, typ ubezpieczenia (podstawowy, rozszerzony oraz rozszerzony eu) oraz wybiera z listy opcjonalne udogodnienia/poszerzenia informacji o wynajmie, takie jak: chęć dostarczenia wraz z samochodem fotelika dziecięcego, chęć używania pojazdu poza granicami Polski oraz chęć korzystania z pojazdu nie tylko przez wynajmującego.

Opcje są czytelnie opisane, tak by uniknąć nieporozumień oraz zrealizowane są za pomocą „match-box’ów”, które są zaznaczane lub odznaczane przez klienta. Po zaznaczeniu wszystkich pożądaných opcji, możliwe jest sfinalizowanie rezerwacji, poprzez potwierdzenia widocznej na ekranie ceny całkowitej wszystkich świadczonych usług. Uiszczenie opłaty możliwe jest do zrealizowania na kilka sposobów, które również widoczne są na ekranie monitora klienta.

3. Funkcjonalności czynne

Funkcjonalności czynne oraz główne założenia projektu opisane zostały w formie tabelarycznej, zawierającej typowe, przewidziane sposoby wykorzystywania dostarczonego oprogramowania oraz bazy danych.

Nazwa	Wyświetlanie listy pojazdów i ich parametrów
Podmiot/y	<ul style="list-style-type: none"> • Klient, • Administrator.
Założenia	<ul style="list-style-type: none"> • Dostępne w wypożyczalni samochody wymagają prawa jazdy kategorii B lub B+x, • Baza samochodów obejmuje pojazdy zasilane: <ul style="list-style-type: none"> ○ Silnikami benzynowymi, ○ Silnikami Diesla, ○ Silnikami hybrydowymi.
Inicjacja	<ul style="list-style-type: none"> • Uruchomienie strony, • Uruchomienie podstrony z dostępnymi samochodami, • Dalsze uruchamianie wybranych opcji.
Przebieg domyślny	<ol style="list-style-type: none"> 1. Wyświetlone zostają wszystkie samochody dostępne we flocie (niezależnie od statusu), 2. Po wybraniu konkretnego pojazdu możliwe jest odczytanie bardziej szczegółowych danych technicznych takich jak: <ol style="list-style-type: none"> 2.1. Rodzaj paliwa,

	2.2. Typ klimatyzacji, 2.3. Liczba drzwi, 2.4. Rodzaj napędu (przedni, tylny, 4x4), 2.5. Rodzaj przekładni zmiany biegów, 2.6. Średnie spalanie paliwa w warunkach: miejskich oraz na trasie, 2.7. Liczba poduszek powietrznych (zaliczane są również poduszki typu: kurtyna powietrzna), 2.8. Pojemność bagażnika (wyrażona w litrach), 2.9. Pojemność silnika (wyrażona w centymetrach sześciennych), 2.10. Moc silnika (wyrażona w koniach mechanicznych), 3. „Profil” każdego samochodu zawiera odnośnik/link wskazujący na odpowiedni mu wpis na stronie autocentrum.pl, 4. Każdy z samochodów posiada widoczny na stronie numer VIN.
Przebieg alternatywny	---
Uchybienia	<ul style="list-style-type: none"> Możliwa jest sytuacja, w której w bazie danych nie znajduje się odpowiadający użytkownikowi pojazd.
Wynik	Lista i parametry techniczne samochodów widoczne dla użytkownika.
Powiązania	<ul style="list-style-type: none"> Wypożyczenie samochodu.

Nazwa	Wyświetlanie listy pojazdów dostępnych w ramach czasowych z możliwością filtrowania
Podmiot/y	<ul style="list-style-type: none"> Klient, Administrator.
Założenia	<ul style="list-style-type: none"> Założenia dotyczące scenariusza pokrywają się z zawartymi w „Wyświetlanie listy pojazdów i ich parametrów”.
Inicjacja	<ul style="list-style-type: none"> Uruchomienie strony, Uruchomienie podstrony służącej do zamawiania pojazdów, Dalsze uruchamianie wybranych opcji.
Przebieg domyślny	<ol style="list-style-type: none"> Wybór (z widocznego na stronie dostępnego menu) daty chęci rozpoczęcia i zakończenia wynajmu krótkoterminowego, Po przejściu do opcji „szukaj samochodu” wyświetlona zostaje lista dostępnych w ustalonym wcześniej czasie pojazdów, Możliwa jest filtracja pojazdów na podstawie parametrów: <ol style="list-style-type: none"> Segment samochodu, Typ nadwozia, Rodzaj przekładni zmiany biegów, Rodzaj paliwa, Liczba miejsc przewidziana w samochodzie (ustalana na podstawie danych zawartych w dowodzie rejestracyjnym). Przy każdym z pojazdów widoczna jest wysokość kaucji (wyrażana w PLN),

	5. Przy każdym z pojazdów widoczna jest wysokość ceny wynajmu za jeden dzień (wyrażana w PLN).
Przebieg alternatywny	---
Uchybienia	<ul style="list-style-type: none"> Możliwa jest sytuacja, w której w bazie danych nie znajduje się odpowiadający użytkownikowi pojazd.
Wynik	Klient wyświetla odpowiadające mu pojazdy, dzięki możliwości filtracji.
Powiązania	<ul style="list-style-type: none"> Wypożyczenie samochodu.

Nazwa	Wypożyczenie samochodu (wypożyczenie door-to-door)
Podmiot/y	<ul style="list-style-type: none"> Klient, Administrator.
Założenia	<ul style="list-style-type: none"> Osoba chcąc wypożyczyć pojazd wykonuje wszystkie czynności własnoręcznie, bądź prosi o wykonanie ich przez administratora serwisu.
Inicjacja	<ul style="list-style-type: none"> Uruchomienie strony, Uruchomienie formularza wypożyczenia samochodu.
Przebieg domyślny	<ol style="list-style-type: none"> Wybór (z widocznego na stronie dostępnego menu) daty chęci rozpoczęcia i zakończenia wynajmu krótkoterminowego, Po przejściu do opcji „szukaj samochodu” wyświetlona zostaje lista dostępnych w ustalonym wcześniej czasie pojazdów, Wybranie auta naciśnięciem opcji „Więcej” znajdującym się przy interesującym klienta modelu, Uruchomienie formularza wypożyczenia samochodu przyciskiem „Wynajmij”, Wybranie spośród dostępnych opcji pakietu ubezpieczenia: <ol style="list-style-type: none"> Pakiet Podstawowy, Pakiet Rozszerzony, Pakiet Rozszerzony EU. Opcjonalne wybranie usług/udogodnień dodatkowych: <ol style="list-style-type: none"> Fotelik dziecięcy, Chęć użytkowania pojazdu poza granicami Polski, Chęć użytkowania pojazdu przez więcej niż jednego kierowcę. Odczytanie ceny za wynajem – za jeden dzień użytkowania (z uwzględnieniem pakietu ubezpieczenia oraz udogodnień), Przejdźcie do sekcji uzupełniania formularza poprzez naciśnięcie przycisku „Dalej”, Wyświetlenie wymaganych do uzupełnienia pól: <ol style="list-style-type: none"> Imię, Nazwisko, Adres e-mail, Numer telefonu kontaktowego, Wiek,

	<p>9.6. Płeć,</p> <p>9.7. Wybór kategorii prawa jazdy z opcji:</p> <p>9.7.1. Kategoria B,</p> <p>9.7.2. Kategoria B+E.</p> <p>9.8. Liczba pełnych lat posiadania prawa jazdy.</p> <p>10. W przypadku chęci skorzystania z opcji door-to-door, należy zaznaczyć odpowiednią opcję w formularzu oraz uzupełnić następujące pola dotyczące adresu, na który należałoby przetransportować wybrany samochód (opcja uwzględnia również odbiór):</p> <p>10.1. Nazwa ulicy,</p> <p>10.2. Numer bloku/domu,</p> <p>10.3. Numer mieszkania,</p> <p>10.4. Kod pocztowy,</p> <p>10.5. Miasto.</p> <p>11. Należy zapoznać się klauzulą dotyczącą RODO,</p> <p>12. Należy zapoznać się z treścią oraz zaakceptować warunki:</p> <p>12.1. Poświadczenie korzystania z pojazdu wyłącznie osób o wieku przekraczającym 21 lat,</p> <p>12.2. Poświadczenie posiadania przez innych użytkowników uprawnień do kierowania pojazdami przez minimum jeden rok kalendarzowy,</p> <p>12.3. Akceptacja regulaminu wypożyczalni dostępnego do wglądu pod wskazanym linkiem.</p> <p>13. Zatwierdzenie uzupełnionych pól,</p> <p>14. Przejście do płatności:</p> <p>14.1. Możliwość opłacenia krótkoterminowego wynajmu poprzez przelew (numer konta Wypożyczalni oraz informacja odnośnie do tytułu przelewu widoczne na stronie) oraz wysłanie potwierdzenia płatności na adres e-mail biura obsługi klienta Wypożyczalni (e-mail oraz format tytułu podany na stronie) – data otrzymania maila najpóźniej do 30 minut przed rozpoczęciem domniemanego wypożyczenia,</p> <p>14.2. Możliwość opłacenia krótkoterminowego wynajmu poprzez transakcję gotówkową do 30 minut przed rozpoczęciem domniemanego wypożyczenia, w oddziale firmy,</p> <p>14.3. Informacja o karze finansowej w przypadku niewiszczenia płatności w wyżej wspomnianym terminie.</p> <p>15. Zmiana statusu samochodu na „zajęty” / „wypożyczony”,</p> <p>16. Przypisanie samochodu do wypożyczającego w bazie danych,</p> <p>17. Generowanie pliku w formacie PDF, zawierającego szczegółowe informacje dotyczące zamówienia oraz wysłanie go na wskazanego maila. Zawarty jest w nim również numer zamówienia.</p>
--	--

Przebieg alternatywny	<ul style="list-style-type: none"> Możliwość przeprowadzenia procesu wypożyczenia w oddziale firmy, Możliwość przeprowadzenia procesu wypożyczenia przy pomocy pracownika biura obsługi klienta (Pracownik uzupełnia formularz).
Uchybienia	<ul style="list-style-type: none"> Niepoprawne lub niepełne uzupełnienie formularza wypożyczenia (następuje wtedy niemożność przejścia do dalszych części procesu wypożyczania i wyświetlony zostaje odpowiedni komunikat).
Wynik	Zawarcie umowy pomiędzy klientem a Wypożyczalnią.
Powiązania	---

Nazwa	Wypadek/Kolizja
Podmiot/y	<ul style="list-style-type: none"> Użytkownik, Winny (opcjonalnie, jeśli winnym nie jest użytkownik).
Założenia	<ul style="list-style-type: none"> Co najmniej jedna ze stron uczestniczących w kolizji lub wypadku posiada zaakceptowane przez pracownika biura obsługi klienta konto, Samochód Winnego posiada ważną polisę OC, Każdy z pojazdów Wypożyczalni posiada ważną polisę OC, Na miejscu kolizji lub wypadku policja ustala sprawcę zdarzenia drogowego.
Inicjacja	<ul style="list-style-type: none"> Uruchomienie strony, Uruchomienie formularza zgłoszenia awarii.
Przebieg domyślny	<ol style="list-style-type: none"> W wyświetlonym na podstronie polu, należy umieścić numer zamówienia generowany w pliku PDF, wysyłanym na skrzynkę mailową po dokonaniu złożenia zamówienia, Do zgłoszenia należy przypisać poglądowy opis zdarzenia, w którym ucierpiał pojazd wypożyczalni, Możliwość przeprowadzenia procesu zgłoszenia kolizji/wypadku z udziałem samochodu Wypożyczalni w oddziale firmy, Możliwość przeprowadzenia procesu zgłoszenia kolizji/wypadku z udziałem samochodu Wypożyczalni przy pomocy pracownika biura obsługi klienta (Pracownik uzupełnia formularz).
Przebieg alternatywny	----
Uchybienia	<ul style="list-style-type: none"> W formularzu podane zostają nieprawidłowe wartości – formularz nie zostaje przyjęty, W przypadku niewielkiej szkody parkingowej, niepotrzebne jest wypełnianie formularza. W takiej sytuacji niezwracana jest opłacona wcześniej kaucja, W przypadku niezgłoszenia kolizji lub wypadku przez jakąkolwiek ze stron, dochodzi do złożenia wniosku w sądzie rejonowym najbliższym Wypożyczalni.

Wynik	<ul style="list-style-type: none"> • Przyjęcie formularza kolizji lub wypadku, • Zmiana statusu samochodu na „uszkodzony”, • Ewentualny brak zwrotu kaucji.
Powiązania	---

Oprócz wymienionych powyżej scenariuszy użycia, możliwe jest również skorzystanie z panelu administratora, który dostępny jest po odpowiednim zalogowaniu się, wymagającym uwierzytelnienia.

Gdy administrator wejdzie na odpowiednią mu stronę, możliwe jest z jej poziomu przeglądanie wszystkich informacji dostępnych dla klienta, lecz w odmienionej formie. Poszerzona jest ona o możliwość ingerencji w zawartość. Dostępne są następujące podstrony:

- Rezerwacje: Widoczne są wszystkie rezerwacje, których dokonano w wypożyczalni samochodów, w schemacie opisanym w tabeli *Orders*. Możliwe jest dodanie rekordu tabeli, poprzez wyświetlany na stronie formularz.
- Klienci: Widoczni są wszyscy klienci, którzy skorzystali z usług wypożyczalni, w schemacie opisanym w tabeli *Users*. Administrator ma możliwość dodania profilu klienckiego, podając odpowiednie dane wymagane w wyświetlonym formularzu.
- Flota: Wyświetlane są wszystkie dostępne w wypożyczalni samochody – zarówno niewypożyczone, jak i te aktualnie użytkowane przez klientów. Schemat widocznych informacji odpowiada temu, który tworzy tabelę *Cars*. Administrator może również edytować każdy rekord wspomnianej tabeli. Po interakcji z przyciskiem „edytuj”, możliwa jest zmiana wartości poszczególnych pól, dzięki wyświetlanemu formularzowi. Ponadto administrator może również dodać pojazd do floty, wypełniając podobny do wspomnianego formularz.
- Cennik: Oczom ukazuje się tabela *Segment_price*, z której to możliwe są do odczytania informacje dotyczące relacji ceny za jeden dzień wynajmu oraz depozytu od segmentu do jakiego przypisany może być samochód. Przed administratorem otworem stoją również drzwi do edycji zamieszczonych w bazie danych wartości pieniężnych.
- Ubezpieczenia: Widoczne są wszystkie informacje o dostępnych dla klientów pakietów ubezpieczeniowych (Podstawowy, Rozszerzony, Rozszerzony EU). Możliwe jest przeczytanie opisu przypisanego do konkretnej opcji oraz odpowiadającą jej dodatkową cenę. Podobnie jak w przypadku wcześniej wymienianych podstron, tak i tutaj możliwe jest edytowanie poszczególnych rekordów. W przypadku wyrażenia takiej chęci, wyświetlony zostaje formularz, którego pola odpowiadają tym z tabeli *Insurance_packet* (oprócz możliwości zmiany numeru identyfikacyjnego pakietu ubezpieczenia).

4. Funkcjonalności bierne

Baza danych składa się z połączonych ze sobą tabel, zawierających istotne informacje dotyczące interesujących (z punktu widzenia architekta) obiektów. W tabelach odnoszących się do samochodów oraz ich szczegółowych parametrów, każdy pojazd jest reprezentowany przez osobny rekord, któremu przypisany został uniwersalny numer identyfikacyjny, dzięki

któremu w łatwy sposób możliwe jest, odnoszenie się do niego. Poszczególne tabele składające się na ogół omówione zostały w następnych akapitach.

Tabela *Cars*: zawiera najprostsze, najbardziej podstawowe informacje o danym samochodzie. Składają się na nią kolumny reprezentujące: numer identyfikacyjny, model, markę, numer identyfikacyjny segmentu (wprowadzona wartość musi pokrywać się z tymi, które dostępne są w słowniku), parametr informujący o stanie samochodu (wypożyczony/niewypożyczony), typ nadwozia (wprowadzona wartość musi pokrywać się z tymi, które dostępne są w słowniku), typ przekładni zmiany biegów (manualna, automatyczna lub automatyczna z możliwością ręcznej zmiany biegów), typ paliwa (wyłącznie benzyna lub diesel) oraz liczba miejsc pasażerskich, znajdujących się w dowodzie rejestracyjnym pojazdu.

Tabela *Cars_detail*: składa się z bardziej szczegółowych informacji dotyczących pojazdów będących rekordami tabeli *Cars*. Na wspomniane powyżej atrybuty szczegółowe, składają się: pokrywający się ze znajdującym się w tabeli *Cars* numer identyfikacyjny, średnie spalanie paliwa w mieście, średnie spalanie paliwa na trasie, pojemność bagażnika, liczba drzwi, typ klimatyzacji (automatyczna bądź manualna), pojemność silnika, moc silnika wyrażona w KM, typ napędu (przedni, tylny lub stały napęd na cztery koła – wartości wprowadzone przy tworzeniu obiektu, muszą pokrywać się z zwartymi w słowniku), liczba poduszek powietrznych, liczba elektrycznie sterowanych szyb, pole typu bool przekazujące informację o tym, czy w samochodzie lusterka regulowane są w sposób elektryczny, informacja mówiąca o podgrzewanych lusterkach (również typ bool), VIN samochodu, link do poglądowego zdjęcia pojazdu, interaktywny link przekierowujący na stronę autocentrum.pl z informacjami o danym modelu; pole zawierające krótki opis widocznego na ekranie monitora samochodu.

Tabela *Users*: przechowuje ona informacje podane w formularzu przez klienta. Składają się na nią następujące kolumny: numer identyfikacyjny klienta (automatycznie przypisywana kolejna liczba naturalna), imię nazwisko, adres kontaktowy e-mail, numer telefonu kontaktowego, wiek, płeć (wartość wprowadzona musi zgadzać się dostępnymi w słowniku: kobieta, mężczyzna), kategoria prawa jazdy oraz liczba reprezentująca liczbę lat, które upłynęły od momentu wejścia w posiadanie przez klienta prawa jazdy.

Tabela *Orders*: niejako zbiera informacje zawarte w tabelach *Cars* oraz *Users*, w celu utworzenia gotowej informacji o złożonym „zamówieniu”. Składają się na nie informacje takie jak: nadany automatycznie numer identyfikacyjny, pokrywający się z danymi z tabeli *Users* numer identyfikacyjny klienta, pokrywający się z danymi z tabeli *Cars* numer identyfikacyjny pojazdu, numer wypożyczenia (generowany dla klienta, tak by mógł on w łatwy sposób odnosić się do swojego wypożyczenia), data złożenia zamówienia, ilość dni wynajmu, data rozpoczęcia wynajmu, data zakończenia wynajmu, pole typu bool mówiące o chęci użytkowania samochodu poza granicami Polski, pole typu bool przedstawiające stan opłacenia usługi, informacja o wysokości należnej firmie zapłaty za wynajem pojazdu.

Kolejne umieszczone w projekcie tabele pełnią rolę uzupełniających dla wyżej wymienionych. Pierwszą z nich jest tabela *Door_to_door*, która zawiera informacje niezbędne w przypadku realizacji zamówienia z dostarczeniem wybranego pojazdu pod wskazany adres. Składają się na nią następujące pola: numer identyfikacyjny zamówienia zgodny ze znajdującym się w tabeli *Orders*, informacja o ulicy (20 znaków), numerze bloku (6 znaków), mieszkania (6 znaków), kod pocztowy (6 znaków) oraz miasto (20 znaków) reprezentowane

są przez zmienne typu char, dzięki czemu klienci nie są ograniczani (np. nietypowy numer bloku).

Tabela *Insurance_packet*: składa się z podstawowych informacji dotyczących opcji ubezpieczenia wynajmowanego samochodu, takich jak: numer identyfikacyjny pakietu ubezpieczenia (dostępne trzy opcje – trzy numery), odpowiadająca numerowi pakietu nazwa ubezpieczenia (1 – Podstawowy, 2 – Rozszerzony, 3 – Rozszerzony EU), opis pakietu ubezpieczenia, przypisana pakietowi cena, o którą powiększona zostanie ostateczna należność za wynajem.

Pozostając przy tematyce ubezpieczenia, wymieniona musi zostać tabela *Accidents*, która jak sama nazwa wskazuje dotyczy problemu zgłaszania ewentualnych uszkodzeń pojazdu, wynikających z kolizji lub innych form destrukcyjnego wpływania na pojazd. Proces zarządzania przyjmowaniem zgłoszeń o kolizjach ujęty został w dalszej części dokumentacji, dotyczącej czynnych funkcjonalności systemu. Tabela składa się natomiast z pól takich jak: identyfikator użytkownika zgłaszającego kolizję, data kolizji, opis zdarzenia (pole umożliwiające wpisanie tekstu) oraz datę wpłynięcia do bazy danych informacji o zgłoszeniu.

Tabela *Segment_price*: rozszerza pojęcie segmentu, przypisując do niego (powiązanie za pomocą numeru identyfikacyjnego segmentu, zgodnego z wartością pola umieszczonego w tabeli *Cars*) konkretne, przydatne z punktu widzenia architekta, parametry. Należą do nich: nazwa segmentu reprezentowana w klasyczny, znany większości sposób (zapis literowy), przypisana wysokość kwoty depozytu oraz podstawowa cena za wynajem za jeden dzień. Cena ta oczywiście poszerzana jest w przypadku chęci korzystania z innych udogodnień oferowanych na stronie.

Tabela *Additional-services*: zawiera informacje o numerze identyfikacyjnym zamówienia (numer musi pokrywać się z tym, który znajduje się w tabeli *Orders*), informacja o chęci wyjazdu wypożyczonym samochodem poza granice Polski, numer identyfikacyjny pakietu ubezpieczenia (wybrana opcja z tabeli *Insurance_packet*), chęć wyposażenia samochodu w fotelik dziecięcy oraz informacja o chęci użytkowania wypożyczonego samochodu przez więcej niż jedną osobę.

Tabela *Prices_for_additional_services* przypisuje do zamówienia, do którego odnosi się za pomocą numeru identyfikacyjnego, ceny poszczególnych parametrów, zawartych we wcześniej wspomnianych tabelach.

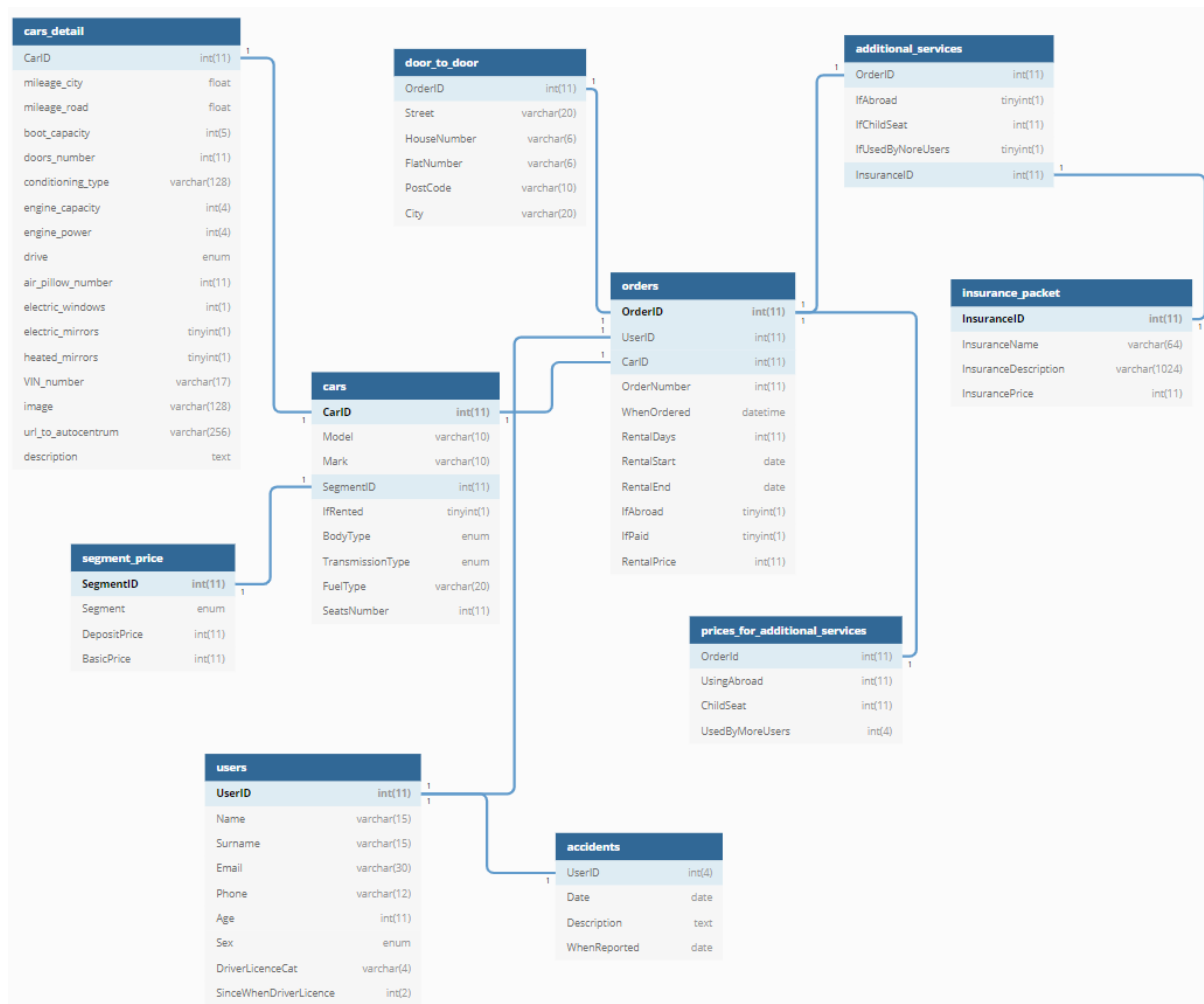
Zaimplementowane zostały również dodatkowe tabele pomagające określić odpowiednie uprawnienia dla administracji oraz użytkowników. Zapobiega to sytuacji, w której użytkownik chcący jedynie wypożyczyć pojazd, mógł realnie wpłynąć na strukturę bazy danych lub zmienić dane z poszczególnych rekordów tabel opisanych powyżej. Do grupy tabel pomagających w odpowiednim zarządzaniu autoryzacją należą: *auth_user* (informacje o użytkowniku i jego uprawnieniach: zwykły klient lub administrator) oraz *auth_group*, *auth_user_groups*, *auth_group_permissions*, *auth_permission*, *auth_user_user_permissions*, których wspólnym efektem jest przypisanie konkretnym użytkownikom odpowiadających mu uprawnień pochodzących z predefiniowanych tabel. Tabele te są ze sobą w odpowiednich relacjach, dzięki czemu możliwe jest odczytanie identyfikatorów poszczególnych czynności, jakich mogą dopuszczać się obsługujące stronę osoby.

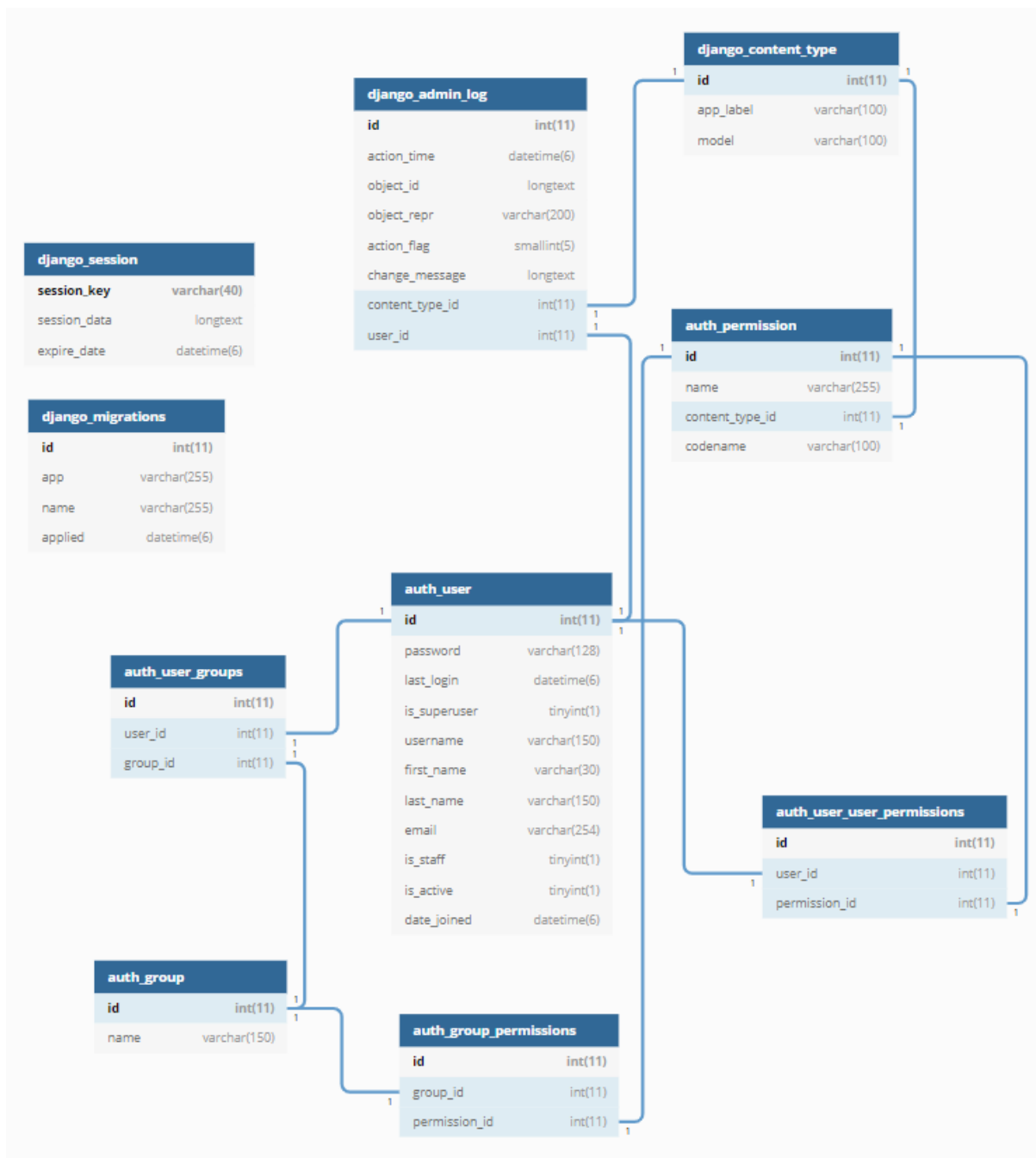
Zbliżoną rolę w połączeniu bazy danych oraz systemu strony internetowej stanowią również tabele wspomagające działanie *Django*: *django_session*, *django_migrations*,

django_admin_log, *django_content_type*. Dzięki nim możliwe jest na przykład: zapisywanie danych o zmianie zawartości poprzez stronę oraz pozyskiwanie informacji o ustanowionych sesjach połączenia.

Wszystkie powyższe informacje zostały zapisane w formie graficznej, tak by zrozumienie zachodzących relacji było łatwiejsze. Grafiki dostępne są poniżej. Ponadto jako *Dodatek 1* załączony do dokumentacji został pseudo-kod, który umożliwił sformułowanie obrazów, generowanych za pomocą środowiska dbdiagram.io.

Całość podzielona została na dwa segmenty, które różnią się pełnioną funkcją. W pierwszej z nich umieszczone zostały relacje tabel odpowiadających za gromadzenie danych dotyczących pojazdów, klientów, zamówień. Druga część dotyczy natomiast funkcjonalności ograniczeń dostępu według uprawnień, rejestracji ustanowionych sesji oraz innych, niezbędnych do poprawnego funkcjonowania połączenia na linii strona internetowa – baza danych.





5. Projekt

Omawiany projekt składa się z mnogiej liczby aplikacji oraz frameworków, dzięki którym zawarte wewnątrz dane możliwe były do przedstawienia w czytelnej dla użytkownika formie graficznej. U podstaw całości stoi język programowania Python, dzięki któremu możliwe było wykorzystywanie odpowiednich skryptów, służących do edycji w Django. Wspomniany framework pozwolił bowiem połączyć zawarte w bazie danych informacje z aplikacją internetową (wprawdzie działającej na adresie loopback).

Baza danych stawiana była w oparciu o MySQL. Wszystkie więc niezbędne komendy, wywołujące pożądane akcje na poszczególnych tabelach, pisane były w ramach przypisanej zeń składni. Bliźniaczo wyglądało zagadnienie uzupełniania kolejnych rekordów bazy danych.

Pomimo dołączonego podczas instalacji MySQL interfejsu, do projektu wybrane zostało inne kompatybilne środowisko Xampp. Udostępnia ono czytelną i przejrzystą możliwość analizy wprowadzonych do bazy danych informacji oraz weryfikację poprawności skonstruowanych tabel. Za pomocą odpowiedniej zakładki, w oknie tekstowym możliwe jest edytowanie już istniejących oraz wprowadzanie nowych danych za pomocą składni MySQL.

Strona internetowa napisana została z wykorzystaniem języka HTML, dzięki czemu użytkownik może samodzielnie poruszać się po stronie, wybierając pożądane opcje oraz wyświetlać dodatkowe informacje. Funkcjonalności poszerzające jakość doznań zostały zaimplementowane przy użyciu języka JavaScript. Skrypty zapewniają między innymi możliwość wyświetlania interaktywnego kalendarza, na którym klient może wybrać datę rozpoczęcia oraz zakończenia wypożyczenia samochodu osobowego.

W celu zapewnienia najwyższej jakości doznań wizualnych, w trakcie odwiedzania strony internetowej wypożyczalni, urządzenie z którego uruchamiana jest ów strona powinna posiadać pobrane odpowiednie grafiki, dla poszczególnych modeli samochodów. Każdy bowiem model ma przypisaną do siebie w bazie danych grafikę, prezentującą w określony sposób jego sylwetkę, tak by klient niezaznajomiony z poszczególnymi modelami różnych marek, był w stanie ocenić wizualny aspekt swojego potencjalnego wyboru.

Warto podkreślić, iż wszystkie najważniejsze funkcjonalności obsługiwane mogą być przez administratora w przeznaczonym dla niego specjalnym panelu. Istnieje możliwość dodawania oraz edytowania rekordów tabel oraz rezerwowania w imieniu klienta (po uzyskaniu danych) samochodów dostępnych w wypożyczalni. Ponadto w przypadku nieoczekiwanych wydarzeń, administrator może również ręcznie zmienić status dostępności (pole bool tabeli *Cars*) wyznaczonego pojazdu.

6. Zgodność wersji systemu oraz aplikacji

Projekt tworzony był za pośrednictwem wielu urządzeń multimedialnych, należało więc starannie zadbać o odpowiednie zaktualizowanie wszystkich stanowisk pracy. Jednym z kluczowych elementów było dobranie odpowiednich wersji poszczególnych aplikacji, frameworków oraz standardów języków programowania, tak by w trakcie tworzenia projektu uniknąć niechcianych problemów z konfiguracjami. Poniżej zamieszczona została lista niezbędnych elementów oprogramowania wraz z informacją o rekomendowanej jej wersji. Należy pamiętać, iż w przypadku dostępności nowszej wersji danego software'u, cały projekt powinien być w stanie uruchomić się poprawnie, niewykluczone są jednak drobne komplikacje.

- Python – wersja: 3.8.0. Oprócz konkretnej wersji najważniejsza jest zgodność z python3.
- Narzędzie „pip”, służące do odpowiedniej instalacji pakietów oprogramowania: wersja 20.2.3
- Django – wersja: 2.2
- MySQL – wersja 8.0
- Xampp – wersja 3.2.4

- jQuery – wersja 3.5.1
- HTML5
- CSS – wersja 2.1

Oprócz wyżej wymienionych aplikacji, rekomendowane jest użytkowanie najnowszej możliwej wersji Windows 10 oraz instalowanie najnowszej aktualizacji zapory ogniowej. W przeciwnym wypadku, możliwe jest napotkanie niedogodności.

7. Testy systemu

Jednym z najważniejszych zagadnień dotyczących projektowania każdego systemu informatycznego, nie tylko baz danych jak w tym przypadku, jest faza testowania funkcjonalności. Odpowiednie sprawdzenie poprawności działania pożądaných funkcji, jest kluczowe, mając przed sobą chęć dostarczenia rzetelnego oprogramowania.

W przypadku omawianego projektu, najbardziej wydajną metodą testowania poprawnego jego funkcjonowania, było stosowanie odpowiednich poleceń składni MySQL, które pozwalały sprawdzić poprawność zaimplementowanych struktur – tabel oraz danych – poszczególnych rekordów tychże tabel. Lecz przed przystąpieniem do „filtrowania” wyników, należało uzupełnić odpowiednimi rekordami poszczególne tabele. Możliwe było do zrealizowania za pomocą polecenia INSERT. Przykładowe zastosowanie widoczne poniżej:

```
INSERT INTO `cars` (`CarID`, `Model`, `Mark`, `SegmentID`, `IfRented`, `BodyType`, `TransmissionType`, `FuelType`, `SeatsNumber`) VALUES (99, 'Multipla', 'Fiat', 2, 0, 'HATCHBACK', 'MANUAL', 'BENZYNA', 6)
```

Po wprowadzeniu powyższego polecenia, możliwa była dwustopniowa weryfikacja poprawności działania: pierwszym krokiem było wizualne sprawdzenie rekordu w tabeli, po zastosowaniu polecenia:

```
SELECT * FROM `cars`
```

CarID	Model	Mark	SegmentID	IfRented	BodyType	TransmissionType	FuelType	SeatsNumber
26	Insignia	Opel	5	0	HATCHBACK	AUTOMAT	BENZYNA	5
27	Fiesta	Ford	2	0	HATCHBACK	MANUAL	BENZYNA	5
28	Octavia	Skoda	4	0	COMBI	MANUAL	BENZYNA	5
29	Mustang	Ford	8	0	COUPE	MANUAL	BENZYNA	4
30	Megane	Renault	3	0	HATCHBACK	MANUAL	BENZYNA	5
31	Fabia	Skoda	2	0	COMBI	MANUAL	BENZYNA	5
32	Fiesta	Ford	2	0	HATCHBACK	MANUAL	DIESEL	5
33	Yaris	Toyota	2	0	HATCHBACK	MANUAL	BENZYNA	5
34	S90	Volvo	6	0	LIMUZyna	AUTOMAT	DIESEL	5
35	Focus	Ford	3	0	SEDAN	MANUAL	BENZYNA	5
36	A8	Audi	7	0	LIMUZyna	AUTOMAT	BENZYNA	5
37	Seria 3	BMW	5	0	SEDAN	SEMIMANUAL	BENZYNA	5
38	Corvette	Chevrolet	8	0	CABRIO	AUTOMAT	BENZYNA	2
99	Multipla	Fiat	2	0	HATCHBACK	MANUAL	BENZYNA	6

Wprowadzony Fiat Multipla znajduje się na ostatniej pozycji w tabeli z numerem identyfikacyjnym 99. Druga faza testu, polegać mogła więc na zastosowaniu polecenia:

```
SELECT * FROM `cars` WHERE carID = 99
```

Którego efektem było wyświetlenie pojedynczego rekordu:

CarID	Model	Mark	SegmentID	IfRented	BodyType	TransmissionType	FuelType	SeatsNumber
99	Multipla	Fiat	2	0	HATCHBACK	MANUAL	BENZYNĄ	6

W celu usunięcia z tabeli dowolnego pojazdu, należało zastosować polecenie DELETE. W tym przypadku komenda ta została wykorzystana do zlikwidowania wcześniej wprowadzonego (przykładowego) rekordu:

```
DELETE FROM `cars` WHERE carID = 99
```

Obecnie w tabeli *Cars* niemożliwe było już odnalezienie Fiata Multipli. Zobrazowane zostały zatem podstawowe działania związane z bazą danych. Lecz oprócz nich, w celu jak najlepszego działania całego systemu wypożyczalni potrzebne było zaimplementowanie możliwości filtrowania samochodów, według przedstawionych przez potencjalnego klienta oczekiwań. Na potrzeby przykładu przedstawione zostanie zestawienie samochodów klasy kompaktowej z silnikiem Diesla, posiadających manualną przekładnię skrzyni biegów:

```
SELECT * FROM `cars` WHERE segmentID = 3 AND FuelType = 'DIESEL'  
AND TransmissionType = 'MANUAL'
```

Faktyczny efekt powyższych wymagań został przedstawiony poniżej:

CarID	Model	Mark	SegmentID	IfRented	BodyType	TransmissionType	FuelType	SeatsNumber
18	Astra	Opel	3	0	HATCHBACK	MANUAL	DIESEL	5
22	i30	Hyundai	3	0	COMBI	MANUAL	DIESEL	5

Powyższy oraz inne filtry zostały zaimplementowane na stronie internetowej dzięki skryptom Python, które dopasowywały odpowiednie filtry i zapytania MySQL, umożliwiające ich wyświetlanie.

Powyższe przykłady odnoszą się jedynie do jednej tabeli *Cars*, warto jednak pamiętać, iż odpowiednie filtrowanie oraz inne akcje wymagane były w użyciu dla wszystkich pozostałych obszarów bazy danych. Ich testowanie przebiegało podobnie do opisywanych powyżej procesów.

8. Dokumentacja użytkownika

Perspektywa użytkownika jest w znacznym stopniu odmienna od tej, którą widzi projektant systemu. Przydatny zatem mógłby okazać się instruktaż obsługi strony internetowej, będącej obrazem funkcjonalności bazy danych, tak by każda osoba była w stanie sprawnie się po niej poruszać. Przedstawiona instrukcja będzie dotyczyć najbardziej powszechnego (w trakcie tworzenia projektu) systemu operacyjnego Windows 10.

W celu uruchomienia serwera, odpowiedzialnego za wyświetlenie zawartości, należy w pierwszej kolejności dokonać weryfikacji zgodności wersji oprogramowania z tymi, które wymienione zostały we wcześniejszym rozdziale (rozdział 6.), tak by uniknąć niepowodzenia na tak wczesnym etapie. Obecnie by móc cieszyć się pełnym zakresem dostępnych możliwości projektu, należy pobrać wszystkie niezbędne pliki. Gdy wszystkie, wspomniane wymagania zostały spełnione, możliwe jest wydanie polecenia w interpreterze poleceń Windows: `cmd.exe`. Szybki dostęp do wymienionej aplikacji możliwy jest po zastosowaniu skrótu klawiszowego: `win + r` oraz wpisaniu w pole tekstowe „`cmd`”. W wyświetlonym na ekranie oknie dialogowym należy przejść do profilu użytkownika, w którego plikach znajdują się te wymagane. Następnie należy aktywować środowisko Django zawierające pliki projektowe poleceniem: `<nazwa_projektu>\Scripts\activate`. Kolejnym krokiem jest przejście do głównego katalogu projektu nazywanego *testsroda*, po czym aktywować serwer za pomocą polecenia: `python manage.py runserver`. Po wykonaniu tych czynności możliwe jest otwarcie strony internetowej znajdującej się pod adresem wyświetlonym na konsoli `cmd`: `http://127.0.0.1:8000`.

Znajdując się na stronie internetowej możliwe jest przejście do zewnętrznych stron internetowych, które mogłyby służyć marketingowi firmy: Facebook, Instagram, Twitter oraz serwis YouTube. Główna jednak treść ukryta jest pod zakładkami widocznymi w górnej części: Zamów, Flota, FAQ, Kontakt oraz Zgłoś Awarię. Dwie przedostatnie spełniają jedynie rolę informacyjną, zawarte są w nich często zadawane pytania oraz dane, dzięki którym możliwy jest kontakt z siedzibą firmy.

Zakładka Zamów, wyświetla użytkownikowi do dyspozycji prosty interfejs, w którym może on wprowadzić daty chęci rozpoczęcia wynajmu krótkoterminowego oraz jego zakończenia. Następnie wyświetlona zostaje lista dostępnych w tym przedziale czasowym pojazdów, na które dodatkowo można nałożyć filtry, opisane we wcześniejszej części dokumentacji. Dzięki temu zabiegowi umożliwione zostało potencjalnemu klientowi, dokładne dopasowanie konkretnego egzemplarza, do jego/jej upodobań. Cały interfejs filtrów został zrealizowany w intuicyjny sposób – jako zestaw możliwych do ustawiania (z rozwijanych list) pól. Klient może wybrać najlepszą jego zdaniem pozycję, po czym przejść do wypełnienia formularza potwierdzającego podpisanie umowy z firmą. Jest on uzupełniany według opisanego we wcześniejszej części schematu. Następnym krokiem – po zatwierdzeniu podanych danych – jest wybranie odpowiadającego klientowi pakietu ubezpieczenia oraz zaznaczenie pól odpowiadających za deklarację chęci wyjazdu za granicę Polski, użytkowania pojazdu przez osoby inne niż najemca, czy też dostarczenia fotelika dziecięcego na pokładzie wybranego pojazdu. Zaznaczanie odpowiednich opcji wiąże się ze wzrostem ceny podstawowej, która w czasie rzeczywistym jest powiększana o wartości wybranych przez

użytkownika dodatków. Kaucja, jest przypisana jest do konkretnych segmentów. W omawianej części procesu, widoczne jest również rozłożenie na poszczególne czynniki ceny końcowej, należnej do uiszczenia.

Jeśli klient zdecydowany jest na opcję wypożyczenia door-to-door, może uzupełnić dodatek do wcześniej opisywanego formularza. Znajdują się w nim informacje umożliwiające dostarczenie wybranego pojazdu przez pracownika wypożyczalni, pod wskazany adres. Gdy okres takiego wypożyczenia się zakończy – wprowadzony adres, będzie również miejscem odbioru pojazdu.

W przypadku podstrony Flota możliwe jest przeglądanie wszystkich dostępnych we flocie samochodów. Wyświetlane są wszystkie pojazdy oferowane przez wypożyczalnię i są uszeregowane według numerów identyfikacyjnych. Może to sugerować użytkownikowi, które z obserwowanych aut, jest „najnowsze” i może wykazać się najmniejszym zużyciem elementów wewnętrznych.

Po zakończonej procedurze uzupełnienia formularza wynajmu, na wskazany weń adres mailowy wysłane zostanie potwierdzenie w formie pliku w formacie PDF. Istotny jest on z punktu widzenia klienta, między innymi dlatego, iż znajdujący się na nim wygenerowany numer zamówienia, pozwoli na ewentualne wypełnienie formularza potwierdzenia kolizji, w której uszkodzony został pojazd wypożyczalni. Omawiana opcja dostępna jest po uruchomieniu zakładki Zgłoś Awarię.

Jeśli użytkownik strony, posiada odpowiednie uprawnienia i może przejść przez cały proces uwierzytelnienia systemowego, możliwe jest przejście do panelu administratora. W tym celu należy podać w wyszukiwarce adres z poprzednich kroków z dopiskiem /admin: <http://127.0.0.1:8000/admin>. Oczom ukazuje się wtedy kolejne menu dostępne, z którego możliwe jest przejście do sekcji: Rezerwacje, Klienci, Flota, Cennik, Ubezpieczenia oraz widnieje również opcja powrotu do strony przeznaczonej dla standardowych użytkowników. Działanie panelu administracyjnego szerzej opisane zostało w części funkcjonalności czynnych.

9. Bibliografia

<https://www.w3schools.com/>

<https://docs.djangoproject.com/en/2.2/>

<https://developer.mozilla.org/>

<https://www.mysql.com/>

<https://docs.python.org/3/>

<https://stackoverflow.com/>

Dodatek 1:

TABLE accidents

```
{
  UserID int(4) [NOT NULL]
  Date date [DEFAULT: NULL]
  Description text [DEFAULT: NULL]
  WhenReported date [DEFAULT: NULL]
}
```

TABLE additional_services

```
{
  OrderID int(11) [NOT NULL]
  IfAbroad tinyint(1) [DEFAULT: NULL]
  IfChildSeat int(11) [DEFAULT: NULL]
  IfUsedByNoreUsers tinyint(1) [DEFAULT: NULL]
  InsuranceID int(11) [DEFAULT: NULL]
}
```

TABLE auth_group

```
{
  id int(11) [pk, NOT NULL]
  name varchar(150) [NOT NULL]
}
```

TABLE auth_group_permissions

```
{
  id int(11) [pk, NOT NULL]
  group_id int(11) [NOT NULL]
  permission_id int(11) [NOT NULL]
}
```

TABLE auth_permission

```
{
  id int(11) [pk, NOT NULL]
  name varchar(255) [NOT NULL]
  content_type_id int(11) [NOT NULL]
  codename varchar(100) [NOT NULL]
}
```

TABLE auth_user

```
{
  id int(11) [pk, NOT NULL]
  password varchar(128) [NOT NULL]
  last_login datetime(6) [DEFAULT: NULL]
  is_superuser tinyint(1) [NOT NULL]
  username varchar(150) [NOT NULL]
  first_name varchar(30) [NOT NULL]
  last_name varchar(150) [NOT NULL]
  email varchar(254) [NOT NULL]
  is_staff tinyint(1) [NOT NULL]
  is_active tinyint(1) [NOT NULL]
}
```

```

    date_joined datetime(6) [NOT NULL]
}

```

```

TABLE auth_user_groups
{
    id int(11) [pk, NOT NULL]
    user_id int(11) [NOT NULL]
    group_id int(11) [NOT NULL]
}

```

```

TABLE auth_user_user_permissions
{
    id int(11) [pk, NOT NULL]
    user_id int(11) [NOT NULL]
    permission_id int(11) [NOT NULL]
}

```

```

TABLE cars
{
    CarID int(11) [pk, NOT NULL]
    Model varchar(10) [NOT NULL]
    Mark varchar(10) [NOT NULL]
    SegmentID int(11) [NOT NULL]
    IfRented tinyint(1) [NOT NULL, DEFAULT: 0]
    BodyType enum('COMBI','HATCHBACK','SEDAN','COUPE','CABRIO','LIMUZYNIA') [NOT NULL]
    TransmissionType enum('MANUAL','AUTOMAT','SEMIMANUAL') [NOT NULL]
    FuelType varchar(20) [NOT NULL]
    SeatsNumber int(11) [NOT NULL]
}

```

```

TABLE cars_detail
{
    CarID int(11) [DEFAULT: NULL]
    mileage_city float [DEFAULT: NULL]
    mileage_road float [DEFAULT: NULL]
    boot_capacity int(5) [DEFAULT: NULL]
    doors_number int(11) [DEFAULT: NULL]
    conditioning_type varchar(128) [DEFAULT: NULL]
    engine_capacity int(4) [DEFAULT: NULL]
    engine_power int(4) [DEFAULT: NULL]
    drive enum('FWD','RWD','AWD') [DEFAULT: NULL]
    air_pillow_number int(11) [DEFAULT: NULL]
    electric_windows int(1) [DEFAULT: NULL]
    electric_mirrors tinyint(1) [DEFAULT: NULL]
    heated_mirrors tinyint(1) [DEFAULT: NULL]
    VIN_number varchar(17) [DEFAULT: NULL]
    image varchar(128) [DEFAULT: NULL]
    url_to_autocentrum varchar(256) [DEFAULT: NULL]
    description text [DEFAULT: NULL]
}

```

```

TABLE django_admin_log
{

```

```

id int(11) [pk, NOT NULL]
action_time datetime(6) [NOT NULL]
object_id longtext [DEFAULT: NULL]
object_repr varchar(200) [NOT NULL]
action_flag smallint(5) [NOT NULL]
change_message longtext [NOT NULL]
content_type_id int(11) [DEFAULT: NULL]
user_id int(11) [NOT NULL]
}

```

TABLE django_content_type

```

{
id int(11) [pk, NOT NULL]
app_label varchar(100) [NOT NULL]
model varchar(100) [NOT NULL]
}

```

TABLE django_migrations

```

{
id int(11) [pk, NOT NULL]
app varchar(255) [NOT NULL]
name varchar(255) [NOT NULL]
applied datetime(6) [NOT NULL]
}

```

TABLE django_session

```

{
session_key varchar(40) [pk, NOT NULL]
session_data longtext [NOT NULL]
expire_date datetime(6) [NOT NULL]
}

```

TABLE door_to_door

```

{
OrderID int(11) [NOT NULL]
Street varchar(20) [DEFAULT: NULL]
HouseNumber varchar(6) [DEFAULT: NULL]
FlatNumber varchar(6) [DEFAULT: NULL]
PostCode varchar(10) [DEFAULT: NULL]
City varchar(20) [DEFAULT: NULL]
}

```

TABLE insurance_packet

```

{
InsuranceID int(11) [pk, NOT NULL]
InsuranceName varchar(64) [DEFAULT: NULL]
InsuranceDescription varchar(1024) [NOT NULL]
InsurancePrice int(11) [NOT NULL]
}

```

TABLE orders

```

{
OrderID int(11) [pk, NOT NULL]
}

```

```

UserID int(11) [NOT NULL]
CarID int(11) [NOT NULL]
OrderNumber int(11) [NOT NULL]
WhenOrdered datetime [NOT NULL]
RentalDays int(11) [NOT NULL]
RentalStart date [NOT NULL]
RentalEnd date [NOT NULL]
IfAbroad tinyint(1) [NOT NULL]
IfPaid tinyint(1) [NOT NULL]
RentalPrice int(11) [NOT NULL]
}

TABLE prices_for_additional_services
{
    OrderId int(11) [NOT NULL]
    UsingAbroad int(11) [DEFAULT: NULL]
    ChildSeat int(11) [DEFAULT: NULL]
    UsedByMoreUsers int(4) [DEFAULT: NULL]
}

TABLE segment_price
{
    SegmentID int(11) [pk, NOT NULL]
    Segment enum('A','B','C','C+','D','E','F','S') [NOT NULL]
    DepositPrice int(11) [DEFAULT: NULL]
    BasicPrice int(11) [NOT NULL]
}

TABLE users
{
    UserID int(11) [pk, NOT NULL]
    Name varchar(15) [NOT NULL]
    Surname varchar(15) [NOT NULL]
    Email varchar(30) [NOT NULL]
    Phone varchar(12) [NOT NULL]
    Age int(11) [NOT NULL]
    Sex enum('MAN','WOMAN') [DEFAULT: NULL]
    DriverLicenceCat varchar(4) [NOT NULL]
    SinceWhenDriverLicence int(2) [NOT NULL]
}

//-----

Ref: accidents.UserID - users.UserID

Ref: additional_services.OrderID - orders.OrderID

Ref: additional_services.InsuranceID - insurance_packet.InsuranceID

Ref: auth_group.id - auth_group_permissions.group_id

Ref: auth_group.id - auth_user_groups.group_id

```

Ref: auth_group_permissions.permission_id - auth_permission.id

Ref: auth_permission.id - auth_user_user_permissions.permission_id

Ref: auth_permission.content_type_id - django_content_type.id

Ref: auth_user.id - django_admin_log.user_id

Ref: auth_user.id - auth_user_user_permissions.user_id

Ref: auth_user.id - auth_user_groups.user_id

Ref: cars.SegmentID - segment_price.SegmentID

Ref: cars.CarID - cars_detail.CarID

Ref: cars.CarID - orders.CarID

Ref: django_admin_log.content_type_id - django_content_type.id

Ref: door_to_door.OrderID - orders.OrderID

Ref: orders.OrderID - prices_for_additional_services.OrderId

Ref: orders.UserID - users.UserID