

Temat: Wypożyczalnia samochodów miejskich (elektrycznych, hybrydowych, spalinowych).

Zespół: Wojciech Moczydłowski, Michał Kucharski, Hubert Fedorowicz,  
Michał Dąbrowski, Stanisław Czobot

## Projekt BD2

### Opis słowny

System tworzony jest dla firmy zajmującej się wypożyczaniem samochodów miejskich. Ma on na celu umożliwienie mieszkańcom łatwy dostęp do wypożyczenia samochodu. Potencjalny użytkownik ma mieć możliwość wypożyczenia dowolnego zaparkowanego samochodu i zwrotu w wybranym przez siebie miejscu i czasie. Mamy nadzieję, że część potencjalnych kupców samochodów zrezygnuje z zakupu na rzecz używania wyżej wymienionego systemu.

### Specyfikacja Wymagań

Id	Nazwa	Opis	Priorytet (1-10)
1	<b>Funkcjonalności dostępne dla klientów</b>		
1.1	Wyszukiwanie	Możliwość przeszukiwania dostępnych ofert wypożyczeń	10
1.2	Wypożyczanie	Możliwość wypożyczenia dostępnego samochodu	10
1.2	Zwrot	Możliwość zwrotu samochodu, gdy klient nie planuje już z niego korzystać	10

1.3	Reklamacja	Możliwość reklamacji wypożyczenia w razie napotkanych niedogodności	8
1.4	Lokalizacja klienta	Możliwość zlokalizowania klienta	8
1.6	Płatność	Możliwość płatności za wypożyczenie	10
1.7	Założenie konta	Możliwość założenia konta przez klienta	10
1.8	Historia klienta	Możliwość przeglądania swojej historii przez klienta	7
1.9	Ocena wypożyczenia	Możliwość oceny wypożyczenia	6
1.10	Ocena samochodu	Możliwość oceny wypożyczonego samochodu	6
<b>2</b>	<b>Funkcjonalności dostępne dla agentów</b>		
2.1	Dodanie oferty	Możliwość dodania do systemu oferty wypożyczenia samochodu	10
2.2	Przyjęcie wypożyczenia	Możliwość przypisania danego agenta do wypożyczenia	10
2.3	Zatwierdzenie reklamacji	Możliwość zatwierdzenia reklamacji danego wypożyczenia	8
2.5	Przyjęcie zwrotu	Możliwość zatwierdzenia zwrotu przez agenta	10

2.6	Lokalizacja samochodów	Możliwość odczytania lokalizacji wszystkich samochodów	8
2.7	Historia wypożyczeń	Możliwość przeglądania historii wypożyczeń	7
2.8	Rekomendacje	Możliwość wysyłania klientom odpowiednio dobranych dla nich ofert	5
<b>3</b>	<b>Wymagania niefunkcjonalne</b>		
3.1	Ilość użytkowników	System ma możliwość przechowywania do 200 000 kont użytkowników	6
3.2	Ilość ofert	System ma możliwość przechowywania do 20 000 ofert	7
3.3	Wyszukiwanie	System wyszukuje 10 potencjalnie najlepszych ofert w przeciągu do 0.5s a następnie 100 pozostałych do 2s	5
3.4	Historia	System prowadzi historię wszystkich wypożyczeń	8
3.5	Prawo	Prawo wynajmu oraz podatkowe jest zgodne z obowiązującym prawem w danych kraju do 12h po jego ogłoszeniu	5
3.6	Awaria	Podczas wystąpienia awarii system jest dostępny dla użytkowników w przeciągu 6h	8
3.7	Odpowiedź	Odpowiedzi systemu w 95% przypadków nie przekraczają 100 ms	6

3.8	Przeciążenie	System jest w stanie obsługiwać w jednym momencie do 5 000 użytkowników bez utraty wydajności. W razie ilości użytkowników większej niż 5 000 a mniejszej niż 20 000 system działa z 80% wydajnością	6
3.9	Rodo	System jest zgodny z prawem RODO. Jest ono aktualizowane do 12h od pojawienia się najnowszej aktualnej wersji.	7

## ETAP 2

### Instrukcja dotycząca generowania danych

Przed rozpoczęciem generowania danych przykładowych należy wykonać poniższe polecenia, po to aby stworzyć bazę danych wraz z wyzwalaczami oraz móc wykonywać funkcje i procedury

1. Wykonać wszystkie instrukcje znajdujące się w pliku BD2\_projekt\_create.sql
2. Skompilować dodatkowe typy danych znajdujące się w pliku generators\_decalration.sql na samej górze
3. Skompilować wszystkie procedury i funkcje znajdujące się w plikach functions.sql oraz generators\_declaration.sql
4. Skompilować triggery znajdujące się w pliku triggers.sql oprócz ostatnich dwóch (fill\_customer\_add\_date oraz fill\_opinion\_date)

### Testy

Przed rozpoczęciem testów, należy wygenerować dane za pomocą generatorów tak aby w systemie znajdowały się poniższe dane:

- a) firma, będąca właścicielem aplikacji - wywołanie procedury generate\_company()

- b) 200 000 kont użytkowników – wywołanie procedury `generate_customers(200000)`
- c) 50 agencji, każda po 20 pracowników, każdej agencji podlega 400 samochodów, a każdy samochód posiada 10 opinii na swój temat – wywołanie procedury `generate_agencies(50,20,400,10)`;
- d) 20 000 różnych opinii dotyczących dowolnych samochodów – wywołanie procedury `generate_opinions(5000)`

Wywołania podanych powyżej procedur znajdują się w pliku `generators.sql`

Podane liczby są przykładowe i można się posłużyć innymi, lecz w celu przetestowania wymagań нефункциональных należy zachować liczbę klientów 200 000 oraz zależność taką, że  $\text{liczba\_agencji} * \text{liczba\_samochodów\_na\_agencje} = 20\,000$ .

Poniżej znajdują się pozostałe scenariusze testów.

### **Scenariusze dotyczące klientów**

1. Wyszukanie ofert posługując się różnymi atrybutami pojazdu, takimi jak np. lokalizacja, model auta, marka samochodu czy też jego przebieg. Ponadto należy w tych testach sprawdzić w jakim czasie system wyszukuje 10 najlepiej dopasowanych ofert a w jakim 100
2. Wypożyczenie samochodu, zakończenie wypożyczenia, oraz ewentualne zgłoszenie reklamacji przez klienta
3. Dodanie i usunięcie nowego użytkownika bez użycia generatora
4. Dodanie opinii na temat wypożyczenia oraz na temat samochodu
5. Dodanie/usunięcie/aktualizacja danych użytkownika, takich jak : nick, hasło, email, imię, nazwisko, telefon

## **Scenariusze dotyczące agentów**

1. Dodanie/usunięcie/aktualizacja danych dotyczących danego pojazdu np. zmiana przebiegu, zmiana statusu auta, dodanie nowego samochodu do oferty, usunięcie samochodu z oferty
2. Dodanie/usunięcie/aktualizacja danych dotyczących wypożyczenia takich jak np. zmiana jego statusu, przypisanie klienta do danego wypożyczenia
3. Sprawdzenie czy samochód którego lokalizacja się aktualizuje jest obecnie wypożyczony
4. Sprawdzenie czy żaden klient nie ma w tym samym czasie wypożyczonego więcej niż jednego samochodu

Dla wszystkich powyższych scenariuszy należy sprawdzić ile zajmuje odpowiedź systemu na wykonane zadania ( oczekiwana to 100ms w 95% przypadków).