

Decide & Ride

W projekcie brali udział:

- Sebastian Rogowski (11303) – przygotowanie wstępnego architektury projektu. Development modelu Order, OrderDao oraz akcji związane ze zleceniami(dodać, usunąć), robienie review, mentoring oraz merge do repozytorium organizacji.
- Bartosz Strzemieczny (11224) – przygotowanie modelu Car, CarDao oraz akcji związane z samochodem(dodać, usunąć). Testowanie funkcjonalności oraz umieszczania komentarzy w kodzie.
- Agnieszka Skrzeczkowska (s11211) - przygotowanie modelu Customer, CustomerDao oraz akcji związane z klientem(dodać, usunąć), wstępna analiza programu oraz przygotowanie instrukcji.

Projekt polegał na stworzeniu wypożyczalni samochodów obsługującej podstawowe funkcjonalności takie jak:

- wypożyczanie i zwrot samochodu,
- rejestracja nowych klientów,
- rejestracja nowych samochodów,
- przechowywanie danych klientów, samochodów oraz wypożyczeń w bazie.

Podczas tworzenia analizy do projektu wzięliśmy pod uwagę możliwe błędy podczas każdego z ww. procesów celem zapewnienia naszemu programowi odpowiedniej jakości oraz dostosowania odpowiednich narzędzi do obsługi wszystkich procesów. W ten sposób na początku wykluczaliśmy szereg błędów, które mogłyby się pojawić podczas późniejszego użytkowania, w tym:

- brak możliwości dodania użytkowników o takich samych numerach dokumentów,
- brak możliwości usunięcia użytkownika z aktywnym wypożyczeniem,
- brak możliwości dodania samochodu o istniejącym numerze VIN,
- brak możliwości wypożyczenia wypożyczonych samochodów,
- brak możliwości usunięcia aktualnie wypożyczonego samochodu,
- w przypadku nie dokonania zamówienia przez użytkownika wypożyczany samochód będzie znajdował się na liście wszystkich samochodów, ale nie będzie możliwości go wypożyczyć.

Ponadto na ten moment wybraliśmy bazę danych Postgres, jako metodę przechowywania danych o użytkownikach, samochodach oraz wypożyczeniach, która jest dostępna przez

<https://www.elephantsql.com/> oraz skorzystaliśmy z Hibernate, który zabezpiecza wypożyczalnię przed ewentualnymi błędami wymienionymi powyżej.

Przykładowa instrukcja obsługi programu

1. Dodanie nowego samochodu

Użytkownik wybiera opcję 1 oraz uzupełnia wszystkie wymagane pola odpowiednimi wartościami.

W przypadku gdy samochód o takim vinNumber będzie w bazie, użytkownik zostanie poinformowany przez odpowiedni komunikat oraz będzie możliwość wprowadzić vinNumber jeszcze raz.

```
Enter number:
1 - add new car
2 - add new customer
3 - add new order
4 - get all cars
5 - get all customers
6 - get all orders
7 - delete car
8 - delete customer
9 - complete with order
0 - exit

Enter option: 1
Enter brand of car(string): Ferrari
Enter model of car(string): Portfino
Enter horse power(int): 600
Enter engine capacity(string): 3855
Enter production year(int): 2018
Enter vin number(string): 2GCEK13J581322529
Enter insurance date(YYYY-MM-DD): 2020-12-31
Enter technical inspection date(YYYY-MM-DD): 2020-12-31
Enter daily payment(double): 399.99
Enter rate(string): A+
```

Dane zostają uzupełnione, samochód zostaje dodany. Jest widoczny w bazie i może zostać wypożyczony

2. Dodanie nowego klienta

Użytkownik wybiera opcję 2 oraz uzupełnia wszystkie wymagane pola odpowiednimi wartościami.

W przypadku gdy klient o takim documentId lub drivingLicenceId będzie w bazie, użytkownik zostanie poinformowany przez odpowiedni komunikat oraz będzie możliwość wprowadzić documentId lub drivingLicenceId jeszcze raz.

```
Enter number:
1 - add new car
2 - add new customer
3 - add new order
4 - get all cars
5 - get all customers
6 - get all orders
7 - delete car
8 - delete customer
9 - complete with order
0 - exit

Enter option: 2
Enter new customer`s name: Jan Kolwaski
Enter document id: ABC123456
Enter expiration date of document(YYYY-MM-DD): 2030-01-10
Enter driver license id: 23456/20/1234
Enter expiration date of driving licence(YYYY-MM-DD): 2035-01-02
Enter birthday(YYYY-MM-DD): 2000-09-01
Enter address: ul. Kolorowa 15/54 00-125 Warszawa
Enter phone number: +48555072634
```

Dane zostają uzupełnione, klient zostaje dodany. Jest widoczny w bazie i może składać zamówienie.

3. Dodanie zlecenia

Użytkownik wybiera opcję 3 oraz podaje id klienta oraz id samochodu(podejrzeć id klienta i samochodu można kolejnymi akcjami).

W przypadku gdy klient o podanym ID lub samochód o podanym ID nie istnieje, użytkownik zostanie poinformowany przez odpowiedni komunikat oraz zlecenie się nie utworzy.

```
Enter number:
1 - add new car
2 - add new customer
3 - add new order
4 - get all cars
5 - get all customers
6 - get all orders
7 - delete car
8 - delete customer
9 - complete with order
0 - exit

Enter option: 3
Enter customer id: 5
Enter car id: 6
```

4. Wyświetlanie wszystkich samochodów

Użytkownik wybiera opcję 4, po tym zostaje wyświetlony sformatowany tekst ze wszystkimi samochodami.

```
Enter number:
1 - add new car
2 - add new customer
3 - add new order
4 - get all cars
5 - get all customers
6 - get all orders
7 - delete car
8 - delete customer
9 - complete with order
0 - exit

Enter option: 4
  id: 2
  brand: Toyota
  model: Yaris
  horsepower: 130
  engineCapacity: 1.5
  productionYear: 2015
  vinNumber: toyotaYarisVinNumber
  insuranceDate: 2020-12-30
  technicalInspectionDate: 2020-11-20
  dailyPayment: 60.00
  rate: C+

  id: 4
  brand: Toyota
  model: Carolla
  horsepower: 200
  engineCapacity: 2.0
  productionYear: 2016
```

5. Wyświetlanie wszystkich klientów

Użytkownik wybiera opcję 5, po tym zostaje wyświetlony sformatowany tekst ze wszystkimi klientami.

```
Enter number:
1 - add new car
2 - add new customer
3 - add new order
4 - get all cars
5 - get all customers
6 - get all orders
7 - delete car
8 - delete customer
9 - complete with order
0 - exit

Enter option: 5
  id: 8
  name: Jan Kolwaski
  documentId: ABC123456
  documentExpirationDate: 2030-01-10
  drivingLicenceId: 23456/20/1234
  drivingLicenceExpirationDate: 2035-01-02
  birthday: 2000-09-01
  address: ul. Kolorowa 15/54 00-125 Warszawa
  phoneNumber: +48555072634
```

6. Wyświetlanie wszystkich aktywnych zleceń

Użytkownik wybiera opcję 6, po tym zostaje wyświetlony sformatowany tekst ze wszystkimi aktywnymi zleceniami.

```
Enter number:
1 - add new car
2 - add new customer
3 - add new order
4 - get all cars
5 - get all customers
6 - get all orders
7 - delete car
8 - delete customer
9 - complete with order
0 - exit

Enter option: 6
    id: 11
    rentDate: 2020-06-02
    rentalCarId: 7
    customerId: 8
```

7. Usunięcie samochodu

Użytkownik wybiera opcję 7 oraz podaje id samochodu (id jest wyświetlane w opcji wyświetlania wszystkich samochodów).

```
Enter number:
1 - add new car
2 - add new customer
3 - add new order
4 - get all cars
5 - get all customers
6 - get all orders
7 - delete car
8 - delete customer
9 - complete with order
0 - exit

Enter option: 7
Enter car id: 2
```

Jeżeli dany samochód jeszcze nie został zwrócony użytkownik zostanie o tym poinformowany(**UWAGA** logi Hibernate można wyłączyć).

```
Enter number:
1 - add new car
2 - add new customer
3 - add new order
4 - get all cars
5 - get all customers
6 - get all orders
7 - delete car
8 - delete customer
9 - complete with order
0 - exit

Enter option: 7
Enter car id: 7
cze 03, 2020 12:02:05 AM org.hibernate.engine.jdbc.spi.SqlExceptionHelper logExceptions
WARN: SQL Error: 0, SQLState: 23503
cze 03, 2020 12:02:05 AM org.hibernate.engine.jdbc.spi.SqlExceptionHelper logExceptions
ERROR: ERROR: update or delete on table "cars" violates foreign key constraint "fkd2p23ixwrrt395glgi9nnbj23" on table "orders"
       Szczegóły: Key (id)=(7) is still referenced from table "orders".
cze 03, 2020 12:02:05 AM org.hibernate.engine.jdbc.batch.internal.AbstractBatchImpl release
INFO: HHH000010: On release of batch it still contained JDBC statements
The car is borrowed!(check car id)
```


Jeżeli nie będzie w bazie wpisu samochodu o takim carId, użytkownik zostanie o tym poinformowany.

```
Enter number:
1 - add new car
2 - add new customer
3 - add new order
4 - get all cars
5 - get all customers
6 - get all orders
7 - delete car
8 - delete customer
9 - complete with order
0 - exit

Enter option: 7
Enter car id: 1
The car is not in system!
```

8. Usunięcie klienta

Użytkownik wybiera opcję 8 oraz podaje id klienta (id jest wyświetlane w opcji wyświetlania wszystkich użytkowników).

```
Enter number:
1 - add new car
2 - add new customer
3 - add new order
4 - get all cars
5 - get all customers
6 - get all orders
7 - delete car
8 - delete customer
9 - complete with order
0 - exit

Enter option: 8
Enter customer id: 1
```

Jeżeli dany klient nie zakończył wszystkich samochodów użytkownik zostanie o tym poinformowany(**UWAGA** logi Hibernate można wyłączyć).

```
Enter number:
1 - add new car
2 - add new customer
3 - add new order
4 - get all cars
5 - get all customers
6 - get all orders
7 - delete car
8 - delete customer
9 - complete with order
0 - exit

Enter option: 8
Enter customer id: 8
cze 03, 2020 12:13:31 AM org.hibernate.engine.jdbc.spi.SqlExceptionHelper logExceptions
WARN: SQL Error: 0, SQLState: 23503
cze 03, 2020 12:13:31 AM org.hibernate.engine.jdbc.spi.SqlExceptionHelper logExceptions
ERROR: ERROR: update or delete on table "customers" violates foreign key constraint "fkpxtb8awmi0dk6smoh2vp1litg" on table "orders"
    Szczegóły: Key (id)=(8) is still referenced from table "orders".
cze 03, 2020 12:13:31 AM org.hibernate.engine.jdbc.batch.internal.AbstractBatchImpl release
INFO: HHH000010: On release of batch it still contained JDBC statements
The customer did not return car!
```

Jeżeli nie będzie w bazie wpisu klienta o takim customerId, użytkownik zostanie o tym poinformowany.

```
Enter number:
1 - add new car
2 - add new customer
3 - add new order
4 - get all cars
5 - get all customers
6 - get all orders
7 - delete car
8 - delete customer
9 - complete with order
0 - exit

Enter option: 8
Enter customer id: 1
Unknown customer!(check customer id)
```

9. Zakończenie zlecenia

Użytkownik wybiera opcję 9 oraz podaje id zlecenie (id jest wyświetlane w opcji wyświetlania wszystkich zleceń).

```
Enter number:
1 - add new car
2 - add new customer
3 - add new order
4 - get all cars
5 - get all customers
6 - get all orders
7 - delete car
8 - delete customer
9 - complete with order
0 - exit

Enter option: 9
Enter order id: 3
```

Jeżeli nie ma zlecenia o podanym orderId użytkownik zostanie o tym poinformowany.

```
Enter number:
1 - add new car
2 - add new customer
3 - add new order
4 - get all cars
5 - get all customers
6 - get all orders
7 - delete car
8 - delete customer
9 - complete with order
0 - exit

Enter option: 9
Enter order id: 1
There are no any order with that ID! Check order ID!
```

10. Zakończenie zlecenia

Żeby zakończyć działanie programu użytkownik powinien wybrać opcję 0.