

Nazwa przedmiotu: Programowanie obiektowe (Java)

Temat projektu: Aplikacja do prowadzenia oraz zarządzania komisem samochodowym

Wiktoria Sikora, 2 rok studiów, grupa 2ID12B

Kielce, 08.06.2021

Politechnika Świętokrzyska
w Kielcach

Programowanie obiektowe (Java)

Temat: Aplikacja do prowadzenia oraz zarządzania komisem
samochodowym

Autor:
Wiktoria Sikora

Nazwa przedmiotu: Programowanie obiektowe (Java)

Temat projektu: Aplikacja do prowadzenia oraz zarządzania komisem samochodowym

Wiktoria Sikora, 2 rok studiów, grupa 2ID12B

Spis treści

1. Opis projektu	3
2. Technologie	3
3. Funkcjonalności projektu	3
4. Sposób uruchomienia aplikacji.....	4
5. Instrukcja obsługi.....	4
6. Struktura projektu	4
7. Testowanie	9

Nazwa przedmiotu: Programowanie obiektowe (Java)

Temat projektu: Aplikacja do prowadzenia oraz zarządzania komisem samochodowym

Wiktoria Sikora, 2 rok studiów, grupa 2ID12B

1. Opis projektu

Aplikacja ma za zadanie usprawnić działanie komis samochodowego. Dzięki niej możliwe jest przechowywanie wszystkich danych na temat samochodów i klientów w jednym miejscu. Funkcjonalność aplikacji została dokładnie opisana w punkcie 3.

2. Technologie

Do stworzenia projektu posłużyły następujące technologie:

- ✚ JDK 8 – oprogramowanie Java,
- ✚ Java FX – biblioteka do tworzenia interfejsu użytkownika,
- ✚ Maven – narzędzie do zarządzania procesem budowy projektu,
- ✚ Mongo Data Base – nierelacyjna baza danych tzw. noSQL.

Maven zależności:

- ✚ Lombok – pozwala na generowanie konstruktorów, getterów, seterów przy użyciu adnotacji co przekłada się na czystszy kod.
 - @AllArgsConstructor – konstruktor z wszystkimi parametrami
 - @NoArgsConstructor – konstruktor bezparametrowy
 - @Data – gettery i setery do wszystkich pól
- ✚ JUnit-jupiter – biblioteka pozwalająca na przygotowanie testów jednostkowych
- ✚ Mongo db driver – sterownik pozwalający na nawiązanie połączenia z bazą danych

3. Funkcjonalności projektu

Aplikacja została stworzona z myślą o wspieraniu komis samochodowego. Funkcjonalności, które zostały zaimplementowane:

- ✚ Lista dostępnych samochodów
- ✚ Lista sprzedanych samochodów
- ✚ Dodawanie nowych samochodów
- ✚ Edycja niesprzedanego samochodów
- ✚ Usunięcie niesprzedanego samochodu
- ✚ Sprzedaż samochodu
- ✚ Wyszukiwanie samochodów po:
 - Marce
 - Modelu
 - Roku produkcji
 - Przebiegu
 - Cenie
 - Skrzyni biegów
 - Paliwie
- ✚ Sortowanie wcześniej wyszukanych pojazdów po:
 - Marce
 - Cenie

Nazwa przedmiotu: Programowanie obiektowe (Java)

Temat projektu: Aplikacja do prowadzenia oraz zarządzania komisem samochodowym

Wiktoria Sikora, 2 rok studiów, grupa 2ID12B

- Roku produkcji
- Przebiegu

Możliwości sortowania:

- Rosnącego
- Malejącego

4. Sposób uruchomienia aplikacji

Aplikacja powinna zostać zainstalowana oraz przetestowana na komputerze klienta przez developera. Ostatecznie uruchomienie powinno odbywać się poprzez dwukrotne kliknięcie w przycisk komisStart.bat

Do działania aplikacji niezbędne jest zainstalowanie:

- Bazy danych MongoDB 4.4.6
- Sterowników jdk 1.8.0_241 zainstalowanych na ścieżce C:\Program Files\Java\jdk1.8.0_241
- Prawidłowe skonfigurowanie zmiennych środowiskowych

Aplikacja po prawidłowym uruchomieniu samoczynnie łączy się z bazą i tworzy kolekcję carService w której magazynowane są wszystkie informacje zamieszczone przy użyciu klienta GUI.

5. Instrukcja obsługi

Aplikacja jest stworzona w taki sposób by była jak najbardziej intuicyjna. Po otwarciu aplikacji możemy zobaczyć od lewego górnego rogu sortowanie. Sortować możemy po marce, cenie, roku produkcji oraz przebiegu i jednocześnie występuje możliwość sortowania rosnąco i malejąco. Kolejno mamy wyszukiwanie. Należy zaznaczyć po czym zamierzamy wyszukiwać, a następnie wpisujemy wartość jaka nas interesuje. Możemy wyszukiwać jednocześnie kilka parametrów np. kolor niebieski i markę samochodu Toyota. Po prawej stronie znajduje się menu. Wybierając przycisk „Nowy”, możemy dodać samochód do komis, pamiętając że wszystkie pola muszą być wypełnione. Kolejno występują „Edytuj”, „Usuń” i „Sprzedaj”, aby móc korzystać z tych opcji należy kliknąć na liście samochód, który chcemy kolejno edytować, usunąć czy sprzedać w innym wypadku nic się nie stanie. Jeśli wybierzemy opcję sprzedaj zostaniemy poproszeni o podanie danych o kliencie, który kupił samochód. Aby zobaczyć listę samochodów sprzedanych, powinniśmy wejść w „Sprzedane”. Również w zakładce „Sprzedane” znajdziemy koło samochodów klienta który kupił samochód.

6. Struktura projektu

Folder model - znajdują się w nim klasy reprezentujące byt rzeczywisty. Możemy wyróżnić:

Car – samochód

- Id

Nazwa przedmiotu: Programowanie obiektowe (Java)

Temat projektu: Aplikacja do prowadzenia oraz zarządzania komisem samochodowym

Wiktoria Sikora, 2 rok studiów, grupa 2ID12B

- Marka
- Model
- Typ nadwozia
- Pojemność silnika
- Rok produkcji
- Przebieg
- Rodzaj skrzyni biegów
- Numer vin
- Typ paliwa
- Kolor
- Kraj
- Klient

Customer – klient

- Imię
- Nazwisko
- Pesel
- Numer dowodu osobistego
- Adres

Address – adres

- Kraj
- Województwo
- Miasto
- Ulica
- Numer budynku

FuelType – rodzaj paliwa

- Diesel
- Benzywa
- Benzyna + LPG

Folder controller - znajdują się w nim klasy odpowiedzialne za wyświetlanie widoków aplikacji, możemy wyróżnić:

- MainViewController
- AddCarController
- EditCarController
- SellCarController
- SoldCarsController

MainViewController – Kontroler zarządzający widokiem mainView.fxml

Metody:

Nazwa przedmiotu: Programowanie obiektowe (Java)

Temat projektu: Aplikacja do prowadzenia oraz zarządzania komisem samochodowym

Wiktoria Sikora, 2 rok studiów, grupa 2ID12B

- Initialize – metoda uruchamiana automatycznie podczas włączenia widoku. Jej zadaniem jest zainicjalizowanie pól klasy oraz ustawienie tzw. ChangeListener.
- removeCar – metoda pozwalająca na usunięcie wskazanej pozycji z listy dostępnych aut.
- Search – metoda pozwalająca na precyzyjne wyszukanie spośród listy wszystkich dostępnych samochodów
- Sort – metoda pozwalająca na posortowanie wyników według wybranego kryterium
- newCar – metoda pozwalająca na przejście do widoku, w którym użytkownik może dodać nowy samochód
- editCar – metoda pozwalająca na przejście do widoku, w którym użytkownik może wprowadzić zmiany w już istniejącym samochodzie. Niezbędne jest wcześniejsze wybranie samochodu.
- Sell – metoda pozwalająca na przejście do widoku, w którym użytkownik może dokonać sprzedaży wskazanego samochodu. Niezbędne jest wprowadzenie danych klienta.
- Sold – metoda pozwalająca na przejście do widoku, w który użytkownikowi zostaje przedstawiona lista sprzedanych samochodów.

AddCarController – Kontroler zarządzający widokiem addCar.fxml

Metody:

- Initialize - metoda uruchamiana automatycznie podczas włączenia widoku. Jej zadaniem jest zainicjalizowanie pól klasy.
- Add – metoda pozwalająca na dodanie nowego samochodu do aplikacji po wcześniejszej prawidłowej walidacji wprowadzonych danych.
- ResponseMessage – metoda pozwalająca na wyświetlenie użytkownikowi informacji o ewentualnych błędach podczas wypełniania formularza.
- Back – metoda pozwalająca na powrót do głównego widoku aplikacji.

EditCarController – Kontroler zarządzający widokiem editCar.fxml

Metody:

- Initialize - metoda uruchamiana automatycznie podczas włączenia widoku. Jej zadaniem jest zainicjalizowanie pól klasy.
- Save – metoda pozwalająca na wprowadzenie zmian w wcześniej wybranym samochodzie. Wprowadzone dane muszą przejść pozytywną walidację.
- responseMessage - metoda pozwalająca na wyświetlenie użytkownikowi informacji o ewentualnych błędach podczas wypełniania formularza.
- Back - metoda pozwalająca na powrót do głównego widoku aplikacji.

SellCarController – Kontroler zarządzający widokiem sellCar.fxml

Metody:

Nazwa przedmiotu: Programowanie obiektowe (Java)

Temat projektu: Aplikacja do prowadzenia oraz zarządzania komisem samochodowym

Wiktoria Sikora, 2 rok studiów, grupa 2ID12B

- Initialize - metoda uruchamiana automatycznie podczas włączenia widoku. Jej zadaniem jest zainicjalizowanie pól klasy.
- Sell – metoda pozwalająca na sprzedanie samochodu poprzez przypisanie do niego użytkownika. Dane użytkownika muszą przejść prawidłową walidację.
- Back - metoda pozwalająca na powrót do głównego widoku aplikacji.

SoldCarsController – Kontroler zarządzający widokiem soldCars.fxml

Metody:

- Initialize - metoda uruchamiana automatycznie podczas włączenia widoku. Jej zadaniem jest zainicjalizowanie pól klasy.
- Back - metoda pozwalająca na powrót do głównego widoku aplikacji.

Folder service - znajdują się w nim klasy przechowujące główną logikę biznesową aplikacji, możemy wyróżnić:

- CarService
- DataBaseService

CarService – serwis zawierający metody umożliwiające manipulowanie danymi na typie Car

Metody:

- unsoldCars – metoda zwracająca listę dostępnych (niesprzedanych) samochodów.
- SoldCars – metoda zwracająca listę sprzedanych samochodów.
- Add – metoda pozwalająca na stworzenie nowego samochodu. Jeśli dane przejdą prawidłowo walidację zwracany jest obiekt Car w przeciwnym razie null.
- Edit – metoda pozwalająca na wprowadzenie zmian w wcześniej przekazanym aucie. Jeśli nowe dane przejdą prawidłowo walidację zwracany jest typ Car z nowymi wartościami, w przeciwnym razie null.
- Remove – metoda pozwalająca na usunięcie przekazanego pojazdu z bazy danych.
- Sell – metoda pozwalająca na sprzedanie pojazdu poprzez dodanie danych o kliencie do pojazdu. Jeśli przekazane dane przejdą prawidłowo walidację zwrócony zostaje typ Car w przeciwnym razie null.
- sortCars – metoda pozwalająca na posortowanie według podanych parametrów przekazanej listy samochodów. Jeśli przekazane parametry są null lub puste zwrócona zostaje pusta lista.
- isFieldValid – metoda pozwalająca na walidację danych. Jeśli któryś z rekordów jest null lub jest pusta metoda zwraca false. W przypadku poprawności wszystkich danych zwraca true.

Nazwa przedmiotu: Programowanie obiektowe (Java)

Temat projektu: Aplikacja do prowadzenia oraz zarządzania komisem samochodowym

Wiktoria Sikora, 2 rok studiów, grupa 2ID12B

- isManufactureYearValid – metoda pozwalająca na sprawdzenie poprawności wprowadzonego roku produkcji. Rok produkcji powinien mieścić się w przedziale od 1870 do 2021
- isMileageValid – metoda pozwalająca na sprawdzenie poprawności danych przebiegu. Przebieg powinien być większy niż 0.
- isPriceValid – metoda pozwalająca na sprawdzenie poprawności danych ceny. Cena powinna być większa niż 0.
- isVinValid – metoda pozwalająca na sprawdzenie poprawności numeru VIN. VIN powinien spełniać warunek długości (17 znaków), powinien składać się z znaków od a-z, A-Z, 0-9 z wyłączeniem liter i, l, o, O, q, Q

DataBaseService – serwis zawierający metody pozwalające na nawiązanie komunikacji z bazą danych.

Metody:

- Connect – pozwala na nawiązanie połączenia z bazą danych o nazwie carService
- dropDB – pozwala na usunięcie bazy danych o nazwie carService
- findAll – pozwala na pobranie z bazy wszystkich pozycji
- findById – pozwala na precyzyjne pobranie z bazy pozycji o danym id
- convertToCar – pozwala na zamianę pobranych danych z bazy na typ Car
- convertToCustomer – pozwala na zamianę pobranych danych z bazy na typ Customer
- convertToAddress – pozwala na zamianę pobranych pozycji z bazy na typ Address
- convertFuelType – pozwala na zamianę wartości pobranej z bazy danych (typ int) na typ FuelType
- insertCar – pozwala na zapisanie przekazanego typu Car w bazie danych
- update – pozwala na zaktualizowanie wartości w bazie danych przekazanego typu Car
- remove – pozwala na usunięcie z bazy przekazanego typu Car
- convertToCarData – pozwala na zamianę przekazanego typu Car na typ przechowywany w bazie danych
- convertToCustomerData – pozwala na zamianę przekazanego typu Customer na typ przechowywany w bazie danych
- convertToAddressData – pozwala na zamianę przekazanego typu Address na typ przechowywany w bazie danych

Klasa Main - Główna klasa pozwalająca na uruchomienie całej aplikacji.

Metody:

Start – metoda ustawiająca dane pierwszego uruchomienia. Wskazuje ścieżkę do pliku fxml o nazwie mainView.fxml, ustawia tytuł okna na „komis samochodowy” oraz reguluje wymiary okna aplikacji (szerokość 1000, wysokość 800).

Nazwa przedmiotu: Programowanie obiektowe (Java)

Temat projektu: Aplikacja do prowadzenia oraz zarządzania komisem samochodowym

Wiktoria Sikora, 2 rok studiów, grupa 2ID12B

Main – metoda uruchamiająca okno interfejsu użytkownika.

Folder fxml - Folder zawierający pliki fxml w których został zdefiniowany wygląd aplikacji. Możemy wyróżnić 5 widoków aplikacji:

- mainView – widok główny
- addCar – widok dodawania samochodu
- editCar – widok edycji samochodu
- sellCar – widok sprzedaży samochodu
- soldCars – widok sprzedanych samochodów

Folder test - folder, w którym została zdefiniowana klasa CarServiceTest. Klasa ta jest odpowiedzialna za testy jednostkowe aplikacji.

7. Testowanie

Ponieważ aplikacja nie jest dużym serwisem i nie będzie rozwijana zdecydowano się w całości przetestować system manualnie. To, co zostało sprawdzone to wszystkie dostępne formularze oraz walidacja danych wejściowych. Dobrze wykonane testy dowodzą, że stworzone controllery realizują swoje funkcje w sposób prawidłowy. Proces operacji na danych przebiega zgodnie z założeniami. Nie zdecydowano się na testy нефunkcjonalne. Ponieważ aplikacja działała na maszynie lokalnej, trudnym zadaniem było przetestowanie jej pod kątem wydajnościowym oraz bezpieczeństwa. Aplikacja posiada 23 testy jednostkowe które mają na celu sprawdzenie kluczowych mechanizmów działania aplikacji. Wszystkie testy dają wynik pozytywny. Podsumowując podczas testów nie znaleziono żadnych rażących błędów czy też błędów, które uniemożliwiałyby prawidłowe działanie aplikacji.