Programowanie komponentowe

**Projekt – sprawozdanie**

# *Deska rozdzielcza samochodu osobowego*

Artur Madaj 224363

Wojciech Sowa 224429

**Informatyka, sem. IV, studia I st. stacjonarne**

**2019 / 2020**

# 1. Wstęp – krótki opis projektu, funkcji i jego zastosowań

Celem projektu było stworzenie programu symulującego podstawowe funkcjonalności samochodu osobowego. W związku z tym powstała aplikacja obsługuje takie zdarzenia jak:

- Przyspieszanie/hamowanie
- Dynamiczna aktualizacja prędkościomierza oraz obrotomierza
- Operowanie kierunkowskazami/światłami awaryjnymi
- Możliwość zmiany biegów z uwzględnieniem sprzęgła
- Możliwość operowania światłami
- Obsługa tempomatu
- Odtwarzanie utworów *.mp3
- Obsługa kontrolek informujących o stanie pojazdu
- Możliwość zmiany motywu programu
- Dynamiczna aktualizacja przebiegów pojazdu
- Synchronizacja z bazą danych
- Opcja zapisania ustawień

## 2. Najważniejsze klasy warstwy danych projektu

Klasa OperateOnDataBase - klasa odpowiedzialna za operacje związane z bazą danych, w naszym przypadku do bazy danych zapisujemy listę piosenek, która znajduje się w klasie ListOfSongs. Możliwości klasy to: wyświetlanie wszystkich/pojedynczego rekordu z bazy danych, dodawanie/usuwanie rekordu do/z bazy danych, aktualizowanie tytułu/wykonawcy/albumu utworu muzycznego. Link do klasy OperateOnDataBase w JavaDoc

Klasa OperateOnFiles - klasa odpowiedzialna za operacje zapisywania i odczytywania do/z plików XML. Operujemy na takich danych jak: przebiegi pojazdu, ustawienia interfejsu graficznego oraz lista piosenek (w przypadku braku dostępności bazy danych). Link do klasy OperateOnFiles w JavaDoc

Klasa Mileage - klasa odpowiedzialna za przechowywanie przebiegów pojazdu i dokonywania na nich podstawowych operacji, m.in. dodawanie do przebiegów przebytego dystansu. Link do klasy Mileage w JavaDoc

Klasa ListOfSongs - klasa odpowiedzialna za przechowywanie obiektów klasy Song w liście i przeprowadzanie podstawowych operacji: dodawania i usuwania z listy. Link do klasy ListOfSongs w JavaDoc

## 3. Najważniejsze klasy warstwy logiki

Klasa Gears - klasa odpowiedzialna za zarządzanie biegami w pojeździe oraz nakładanie ograniczeń na pewne sytuacje, które mogą w nim wystąpić (np. nie możemy jechać na pierwszym biegu szybciej niż 30 km/h). Link do klasy Gears w JavaDoc

Klasa Statistics - klasa odpowiedzialna za obliczanie podstawowych statystyk auta, takich jak: przebyty dystans, maksymalna prędkość, średnia prędkość i średnie spalanie. Link do klasy Statistics w JavaDoc

Klasa LightingSystem - klasa nadrzędna dla wszystkich świateł. Odpowiedzialna jest m.in. za zarządzanie nimi oraz wypisywanie stosownych informacji dotyczących aktualnego oświetlenia pojazdu. Link do klasy LightingSystem w JavaDoc

Klasy Brake, Accelerator oraz Clutch - klasy odpowiednich pedałów w samochodzie. Zawierają w sobie metody związane z dociskaniem/zwalnianiem pedału. Link do paczki zawierającej pedały w JavaDoc

Klasy SuchFileDoesNotExist oraz TooFastException - klasy odpowiedzialne za obsługę własnych wyjątków - błędnie podanej nazwy pliku oraz zbyt szybkiej jazdy. Link do paczki zawierającej wyjątki w JavaDoc

Klasa Console - klasa odpowiedzialna za obsługę programu w trybie konsolowym. Zawiera podstawowe operacje, takie jak: wypisanie utworów muzycznych, wyświetlenie statystyk oraz uruchomienie świateł awaryjnych. Link do klasy w Console JavaDoc

Klasa RunningTime - klasa odpowiedzialna za zarządzanie czasem w programie. Zawiera funkcjonalności, takie jak: obliczanie czasu działania silnika, operowanie zegarem znajdującym się na desce rozdzielczej. Link do klasy RunningTime w JavaDoc

## 4. Najważniejsze klasy warstwy interfejsu

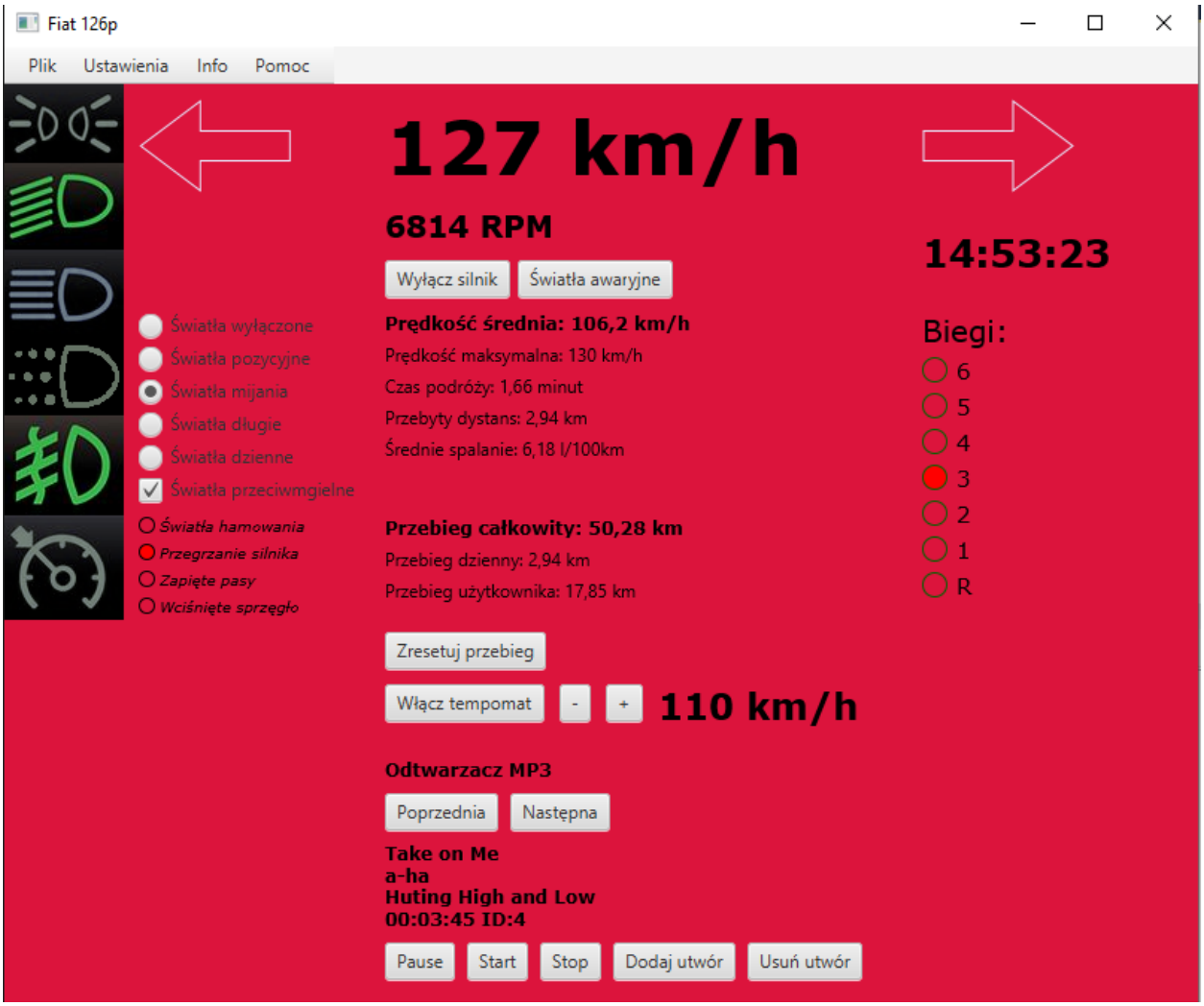
Klasy AlertBox, ConfirmBox oraz Infos - klasy odpowiedzialne za wyświetlanie komunikatów i informacji o aplikacji, takich jak: “Czy na pewno chcesz zakończyć działanie programu?” lub proste instrukcje obsługi programu. Link do przykładowej klasy AlertBox w JavaDoc

Klasy InsertingBox oraz Settings - klasy odpowiedzialne za wyświetlanie nowych okien i za wprowadzanie zmian w programie, takich jak zmiana koloru tła, czy dodanie nowej piosenki. Link do przykładowej klasy InsertingBox w JavaDoc

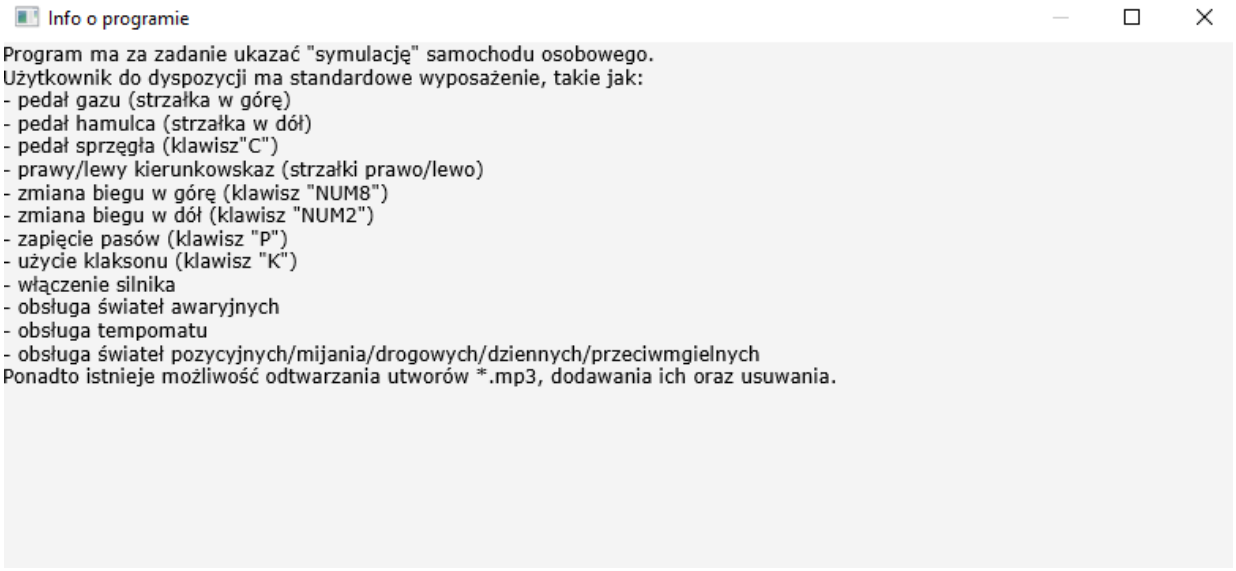
Klasa Gui - klasa odpowiedzialna za obsługę zdarzeń w graficznym interfejsie użytkownika. Pozwala ona również na prawidłowe rozmieszczenie różnorodnych, występujących w niej obiektów, takich jak: przyciski, podpisy. Link do klasy Gui w JavaDoc

## 5. Zrzuty ekranu najistotniejszych widoków działającej aplikacji

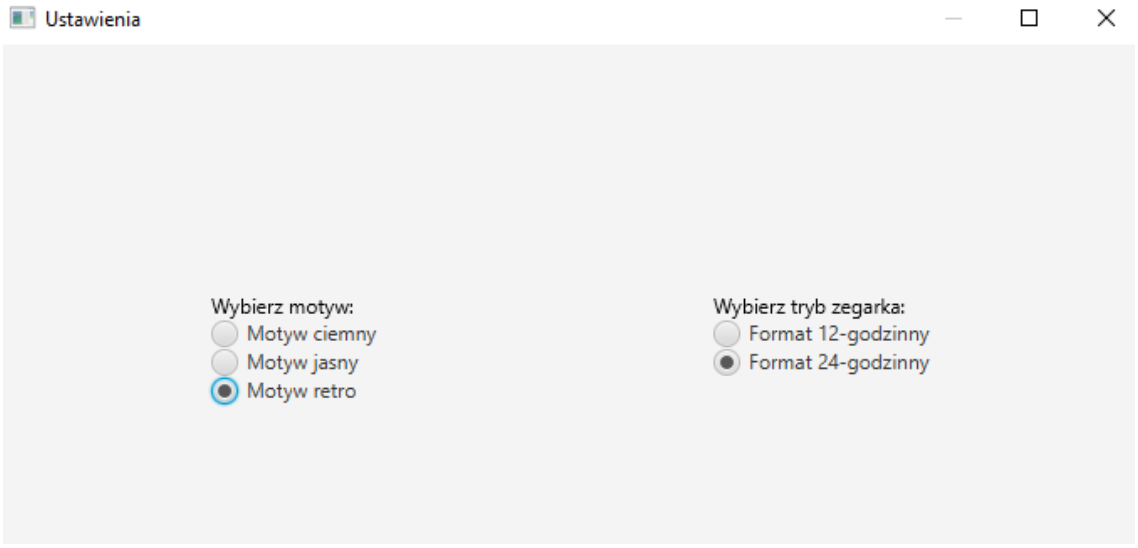
*5.1. Główne okno deski rozdzielczej samochodu osobowego*



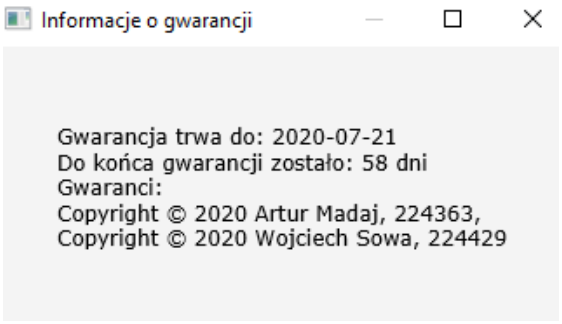
*5.2. Okienko ,,O programie", zawierające opis klawiszy funkcyjnych i przycisków obsługujących aplikację.*




*5.3. Okno dialogowe ,,ustawienia"*




5.4 Okno dialogowe ,,gwarancja”



## 6. Postać bazy danych (schemat tabel) z zapisanymi informacjami o zdarzeniach. (zrzut ekranu, fragment kodu/tekstu itp.)

Zrzut ekranu pokazujący tabelę utworów utworzoną w bazie danych:

| | tytul | wykonawca | album | czasTrwania | id |
|---|---|---|---|---|---|
| 1 | Make me Fade | Klej | Tuesdays | 00:04:39.0000000 | 1 |
| 2 | FML | K.Flay | FML | 00:03:28.0000000 | 3 |
| 3 | Take on Me | a-ha | Huting High and Low | 00:03:45.0000000 | 4 |
| 4 | Get Free | Lana Del Rey | Lust For Life | 00:05:34.0000000 | 5 |
| 5 | Astronomia | Vicetone & Tony Igy | Astronomia | 00:03:18.0000000 | 2 |

Do tabeli zapisywane są wartości znajdujące się w klasie ListOfSongs.
Do utworzenia i operowania na bazie danych wykorzystaliśmy program Microsoft SQL Server przy pomocy sterownika JDBC Driver.

Kod z pliku xml, do którego zapisywane są przebiegi:

```
<totalMileage>47.34513888888946</totalMileage>
<dailyMileage>13.971472222221665</dailyMileage>
<userMileage>14.911361111110635</userMileage>
<data>2020-05-22</data>
</warstwaDanych.Mileage>
```

Kod z pliku xml, do którego zapisywane są ustawienia kokpitu:

```
<darkOn>false</darkOn>
<lightOn>false</lightOn>
<retroOn>true</retroOn>
<englishFormatOn>false</englishFormatOn>
<normalFormatOn>true</normalFormatOn>
</warstwaInterfejsu.Settings>
```

## 7. Podsumowanie i ewentualne uwagi grupy nt. Projektu.

Program posiada wszystkie obligatoryjne funkcjonalności wymagane do zaliczenia projektu:

- Prędkościomierz
- Liczniki przebiegów
- Kontrolki kierunkowskazów i świateł
- Statystyki dotyczące podróży i pojazdu
- Okno dialogowe “Ustawienia”
- Obsługę podstawowych zachowań okna programu
- Zapis statystyk do pliku XML i bazy danych
- Obsługę poprzez klawiaturę i myszkę
- Tempomat
- Obsługę w trybie graficznym i tekstowym
- Podział na warstwy logiki, danych i interfejsu użytkownika

Ponadto zawiera w sobie dodatkowe elementy:

- Odtwarzacz utworów muzycznych *.mp3
- Wizualizacja operowania skrzynią biegów
- Obrotomierz
- Kontrolki odzwierciedlające stan pojazdu

Program powstał bez użycia dodatkowych narzędzi typu Scene Builder.