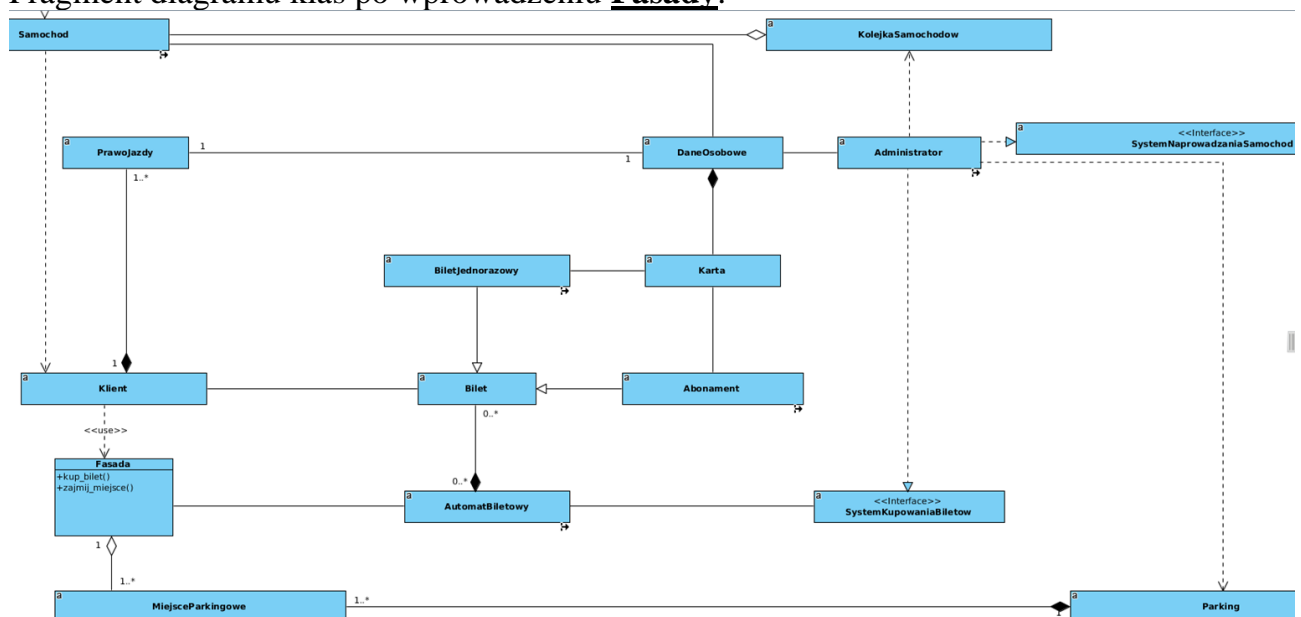
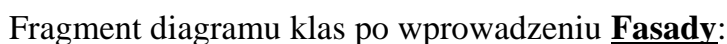
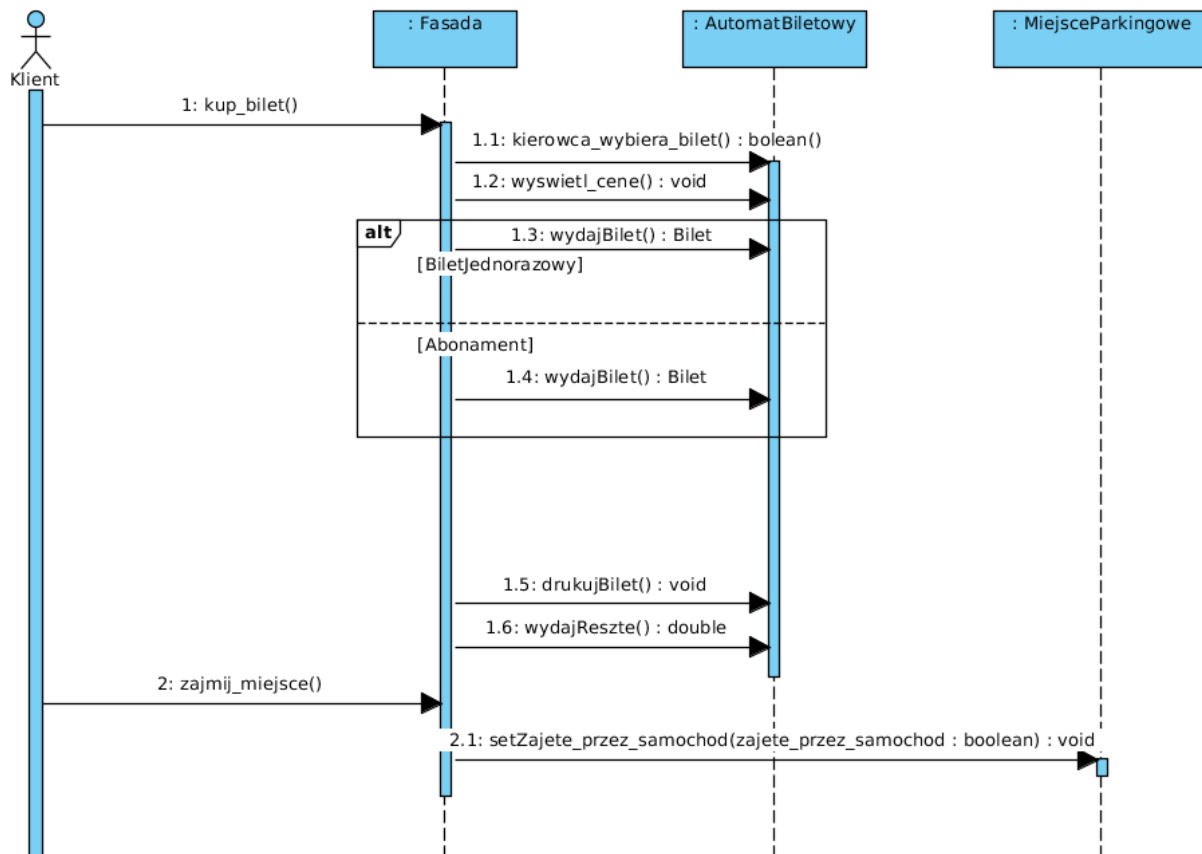


Czaja Hubert, Białek Tomasz, gr.1

Fragment diagramu klas przed wykorzystaniem **Fasady**:



Przebieg interakcji obiektów:

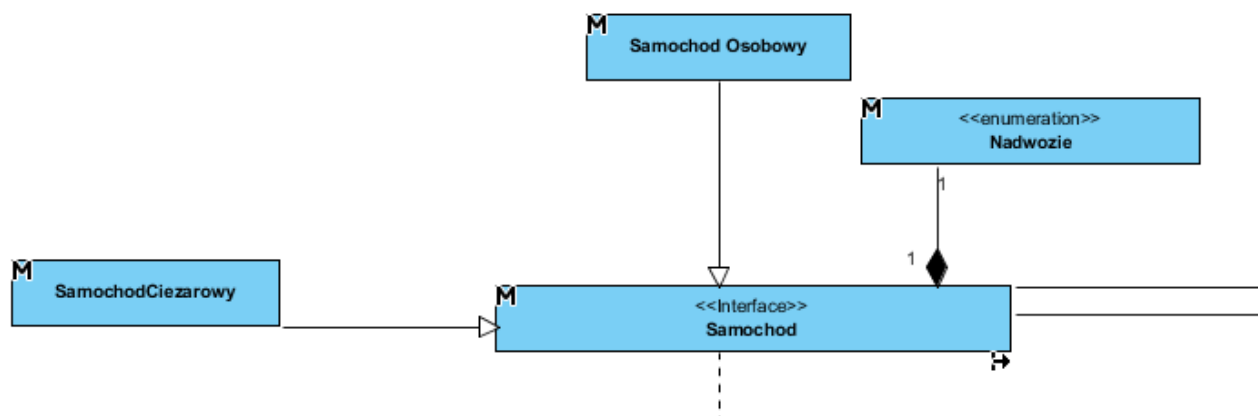


Enkapsulacja wiedzy na temat budowy systemu jest istotna w każdym projekcie. Klient nie powinien znać szczegółów implementacyjnych lub strukturalnych. Interakcja Klienta z systemem polega na wykorzystaniu **Fasady** jako interfejsu-łącznika.

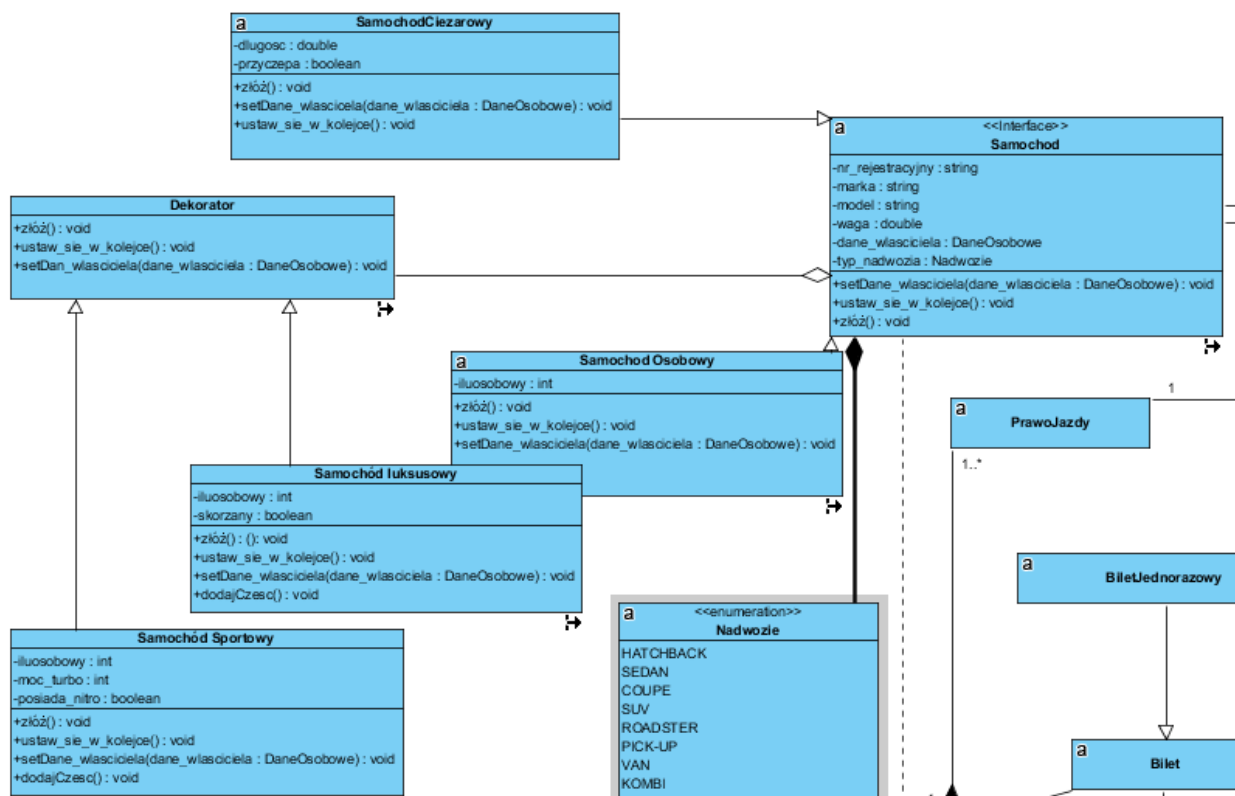
Wprowadzenie **Fasady** do projektu wydaje się obowiązkowe, aby interfejs użytkownika był prosty i przejrzysty, dodatkowo proces czy to tworzenia biletów czy obsługi miejsc parkingowych nie powinien być jawny. Początkowa wersja projektu uwzględniała wykorzystanie metod na automacie biletowym lub parkingu przez Klienta osobiście co jest znacznie mniej czytelne. Teraz korzysta on z metod „kup\_bilet()” oraz „zajmij\_miejsce()”. Podział aplikacji na warstwy ułatwia zatem niezależny rozwój klienta i złożonego systemu. Ponadto względem pierwszej wersji projektu dużo zmniejszyła się liczba zależności pomiędzy Klientem a elementami systemu.

Kolejnym zastosowanym wzorcem w projekcie parkingu jest dekorator. Wiadomym jest, że z parkingu korzystają różne typy samochodów, więc chcąc wprowadzić tę cechę do projektu wykorzystamy ten specyficzny wzorzec strukturalny. Z jego pomocą możemy wprowadzić relację pomiędzy specyficznymi rodzajami samochodów, a ogólnym – zwykłym typem.

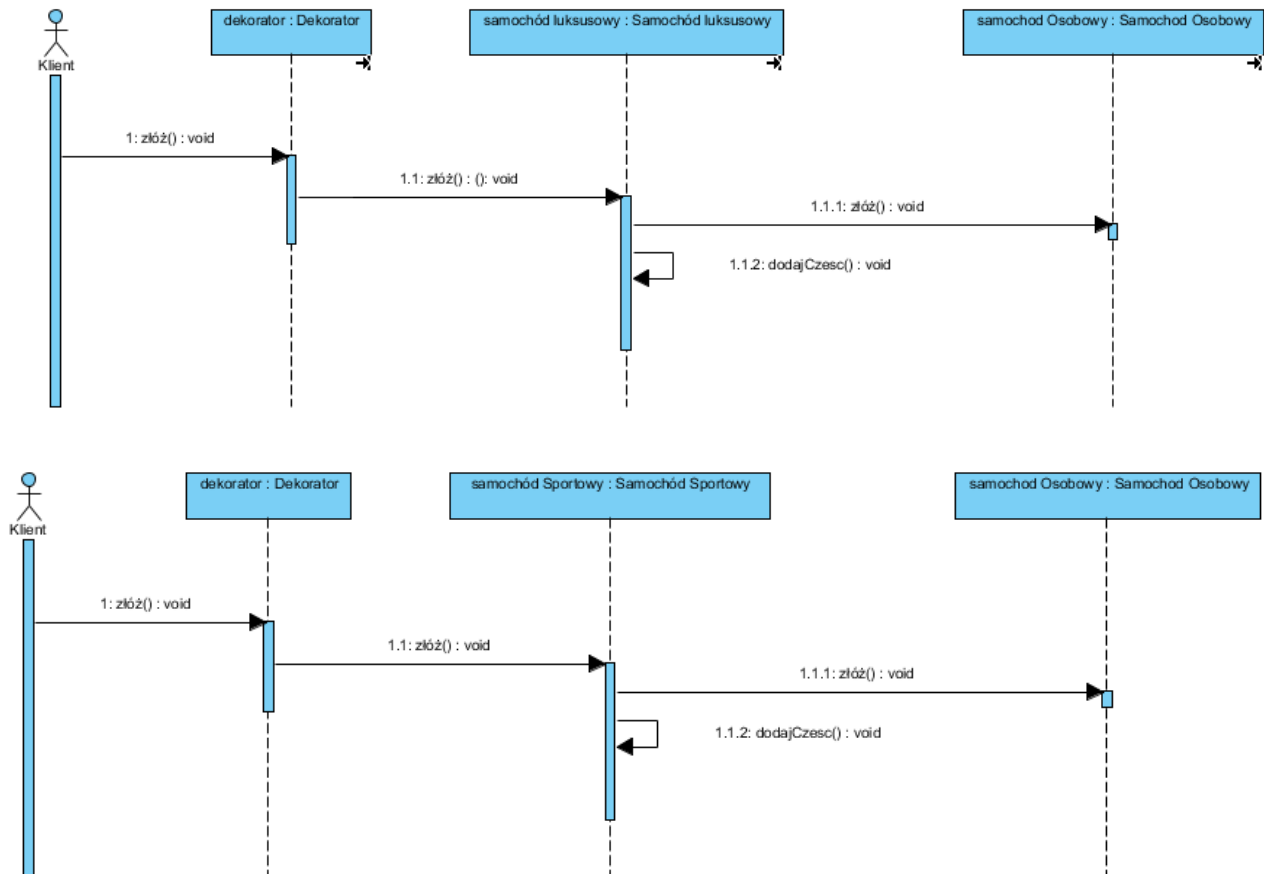
Fragment diagramu klas przed wykorzystaniem **Dekoratora**:



Fragment diagramu klas po wprowadzeniu **Dekoratora**:



## Przebieg interakcji obiektów:



Aby wprowadzić wzorec dekoratora potrzebujemy zestaw wymagań (interfejsu), który będzie implementowany przez klasę Samochód osobowy oraz dekoratora samochodów. Interfejs umożliwi wtedy tworzenie ogólnego typu „Samochód”. Aby stworzyć (przypisać definicję) specjalny samochód utworzymy instancję Dekorator, na którym z kolei wywołamy tworzenie obiektu specjalnego obiektu. W tym punkcie możliwe jest przypisanie do naszego samochodu jego unikalnych cech, których zwykły „Samochód Osobowy” nie posiada.

W poprzedniej wersji nie było możliwym tworzenie szczególnych typów samochodów, które posiadają cechy nieobecne w typowym aucie. Z pomocą dekoratora rozszerzamy możliwości definicji samego samochodu, przez co uzyskujemy większą unikalność w projekcie.