

Modelowanie i Analiza Systemów Informatycznych

Spis treści

1. Wymagania użytkownika.....	3
2. Diagram przypadków użycia	3
3. Diagram klas – analityczny.....	5
4. Diagram klas – projektowy	6
5. Scenariusz przypadku użycia (jako tekst).....	7
6. Diagram aktywności dla przypadku użycia	9
7. Diagram stanu dla klasy	10
8. Diagram interakcji (sekwencji) dla przypadku użycia	10
9. Omówienie decyzji projektowych i skutków analizy dynamicznej	12

Wymagania użytkownika

Właściciel salonu Samochodowego określił swoje wymagania odnoszące się do systemu informatycznego:

1. W systemie mają być przechowywane dane takie jak: imię (nie dłuższe niż 10 znaków), nazwisko, oraz opcjonalnie nazwisko panińskie. Należy pamiętać o kolejności alfabetycznej przechowywania imienia.
2. Oprócz danych przechowywanych dla Osoby, należy przechowywać w systemie dane dla Klienta – data urodzenia oraz dane dla sprzedawcy – data zatrudnienia, data zwolnienia oraz listę języków obcych. Sprzedawca może też być klientem.
3. Dla Auta należy pamiętać unikalny vin, przebieg, jednakowy dla wszystkich aut, równy 0, koszt modyfikacji, cena obliczana na podstawie kosztu modyfikacji.
4. Ze względu na typ, auta dzielą się na auta standardowe, dla nich mamy pamiętać czy można zmodyfikować auta, oraz dzielą się na auta z modyfikacjami dla których pamiętamy listę modyfikacji. Auto standardowe może się stać autem z modyfikacjami, jeżeli Auto standardowe można zmodyfikować. Podział jest kompletny.
5. Należy pamiętać któremu klientowi zostało sprzedane auto z salonu. Klient może kupić wiele aut. Dla każdej transakcji należy pamiętać datę zakupu oraz opinie. Dla każdego zakupu opinia może być wystawiona tylko raz. Dla opinii pamiętamy komentarz.
6. Dla ubezpieczenia OC należy pamiętać zakres ubezpieczenia, a dla ubezpieczenia AC maksymalną kwotę pokrywającą szkodę. Ubezpieczenie zawiera datę początku ubezpieczenia, datę końca ubezpieczenia, numer umowy. Ubezpieczenie jest ubezpieczeniem OC i ubezpieczeniem AC. Ubezpieczenie jest przypisane do danego auta, natomiast auto może posiadać wiele ubezpieczeń.
7. Opis naprawy i unikalny numer naprawy pamiętamy dla Naprawy. Naprawa dotyczy danego auta, natomiast Auto może mieć wiele napraw. Na podstawie numeru naprawy można otrzymać odpowiadającą naprawę.
8. Należy zadbać o odpowiednią hermetyzację (enkapsulację) danych. Wszelkie dane powinny być zapisywane do pliku oraz odczytywane po uruchomieniu programu.
9. System powinien umożliwiać właścicielowi realizację m.in. następujących funkcjonalności:
 - wyświetlenie sprzedawców pracujących w salonie
 - obliczenie składki AC

Diagram przypadków użycia

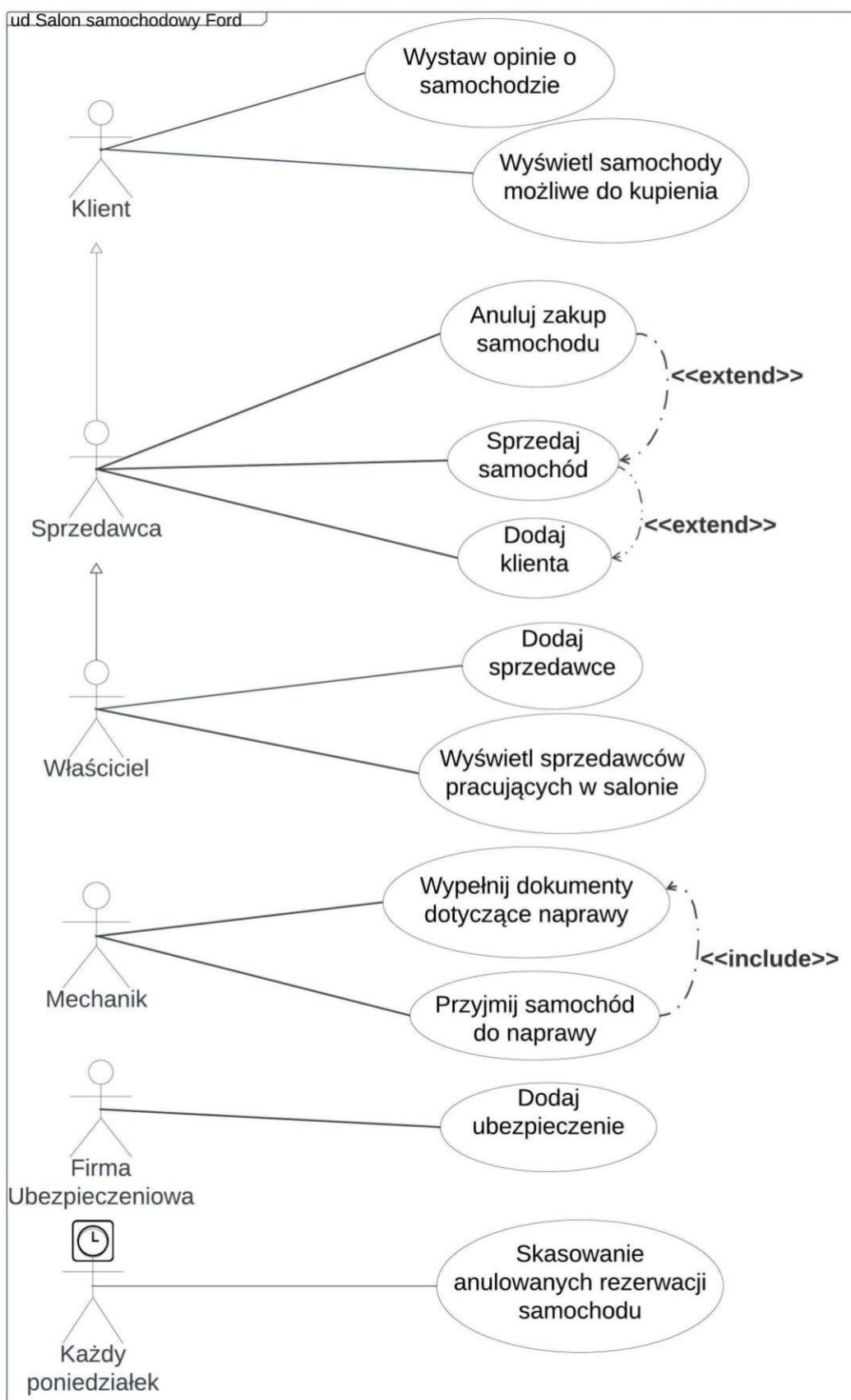


Diagram klas – analityczny

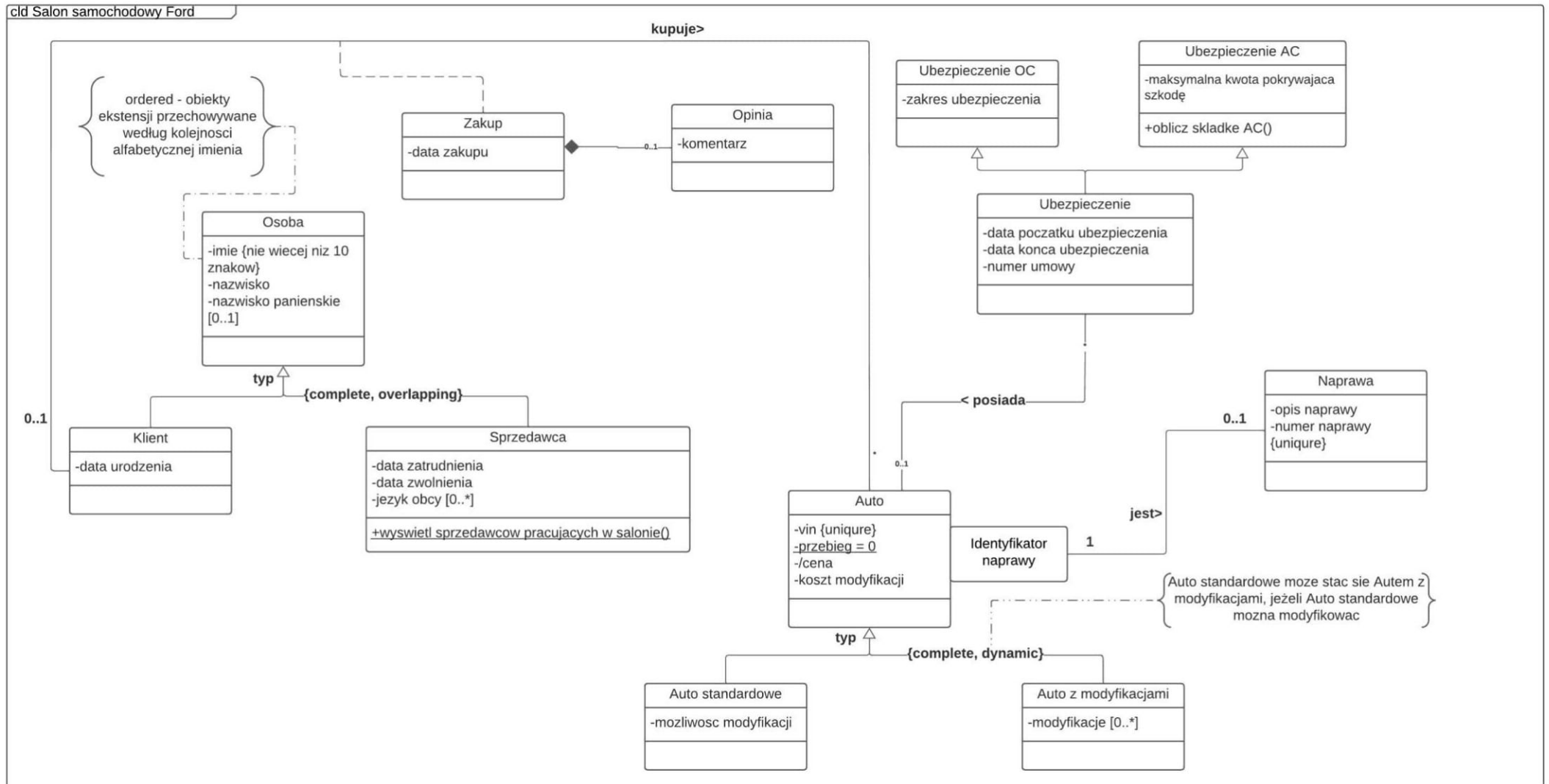
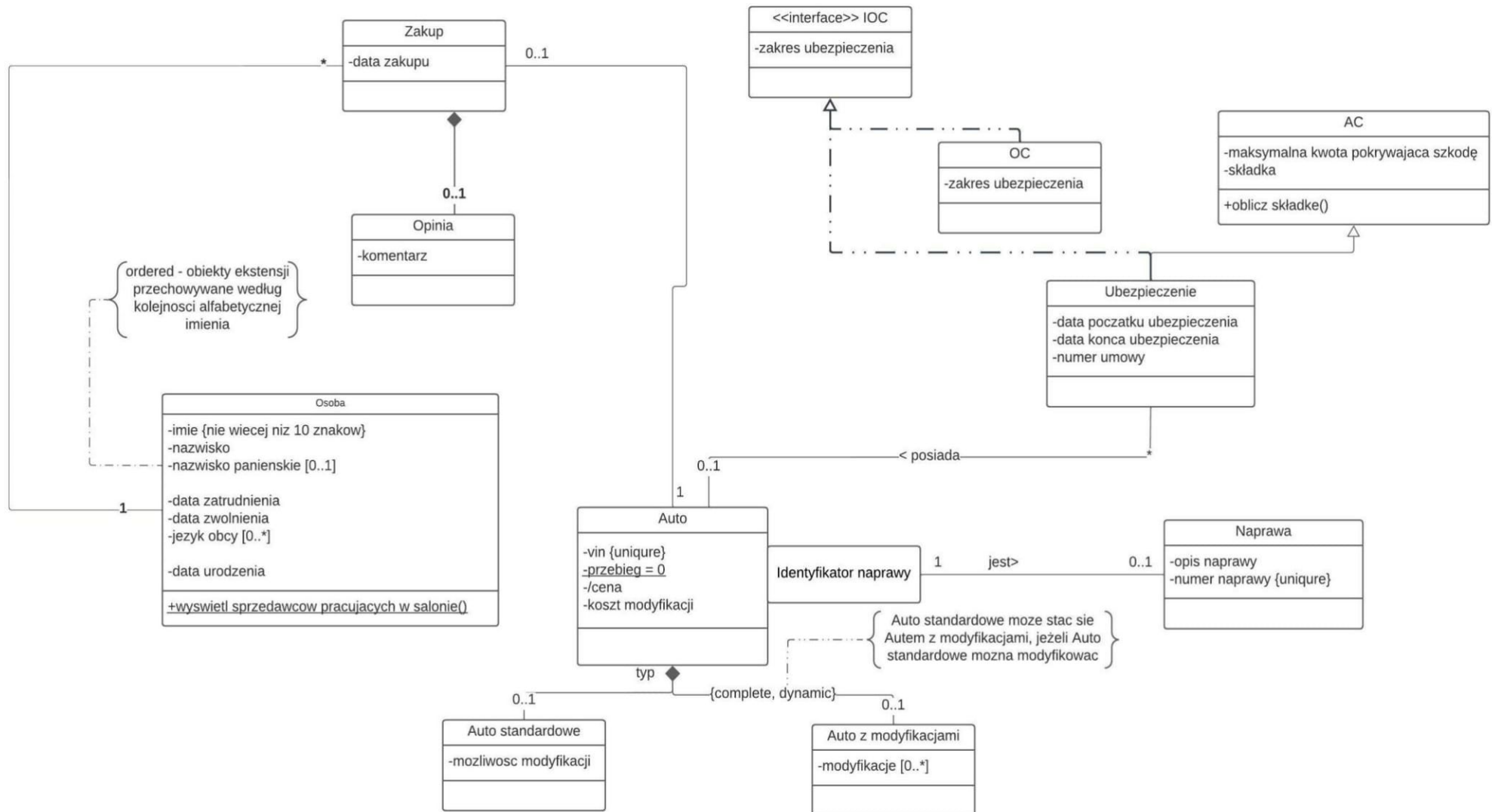


Diagram klas – projektowy



Scenariusz przypadku użycia: Sprzedaj samochód

Nazwa przypadku użycia	Sprzedaj samochód
Warunek początkowy	Istnieje co najmniej jeden samochód oraz co najmniej jeden klient w bazie.
Główny przepływ zdarzeń	<ol style="list-style-type: none">1. Aktor Sprzedawca uruchamia przypadek użycia.2. System wyświetla dostępne samochody możliwe do sprzedania. Aktor wybiera samochód.3. System wyświetla klientów znajdujących się w systemie. Aktor wybiera klienta.4. System prosi o uzupełnienie dokumentacji dotyczącą samochodu. Aktor wprowadza dane.5. System wyświetla warunki umowy. Aktor akceptuje umowę.6. System informuje aktora o pomyślnej sprzedaży samochodu klientowi.
Alternatywny przebieg zdarzeń	<p>2a. W systemie nie ma przypisanych żadnych samochodów możliwych do sprzedania. System wyświetla odpowiedni komunikat i kończy przypadek użycia.</p> <p>3b W systemie nie ma przypisanych żadnych klientów. System wyświetla odpowiedni komunikat i kończy przypadek użycia</p>

	5b Aktor nie akceptuje warunków umowy. System wyświetla odpowiedni komunikat i kończy przypadek użycia
Zakończenie	Zgodnie ze scenariuszem
Warunek końcowy	Zostaje odnotowana sprzedaż auta klientowi przez Aktora.

Diagram aktywności dla przypadku użycia: Sprzedaj samochód

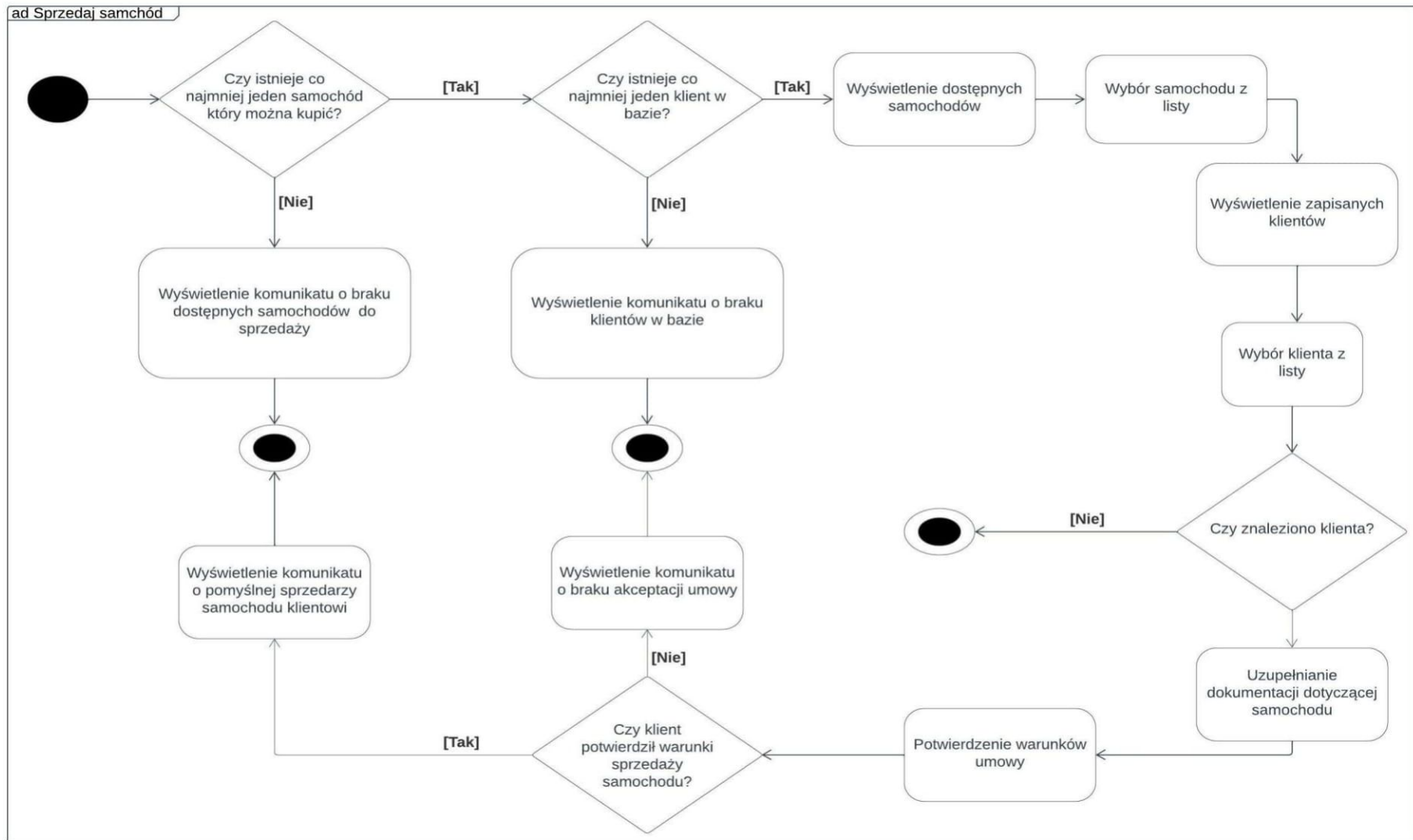


Diagram stanu dla klasy Samochód

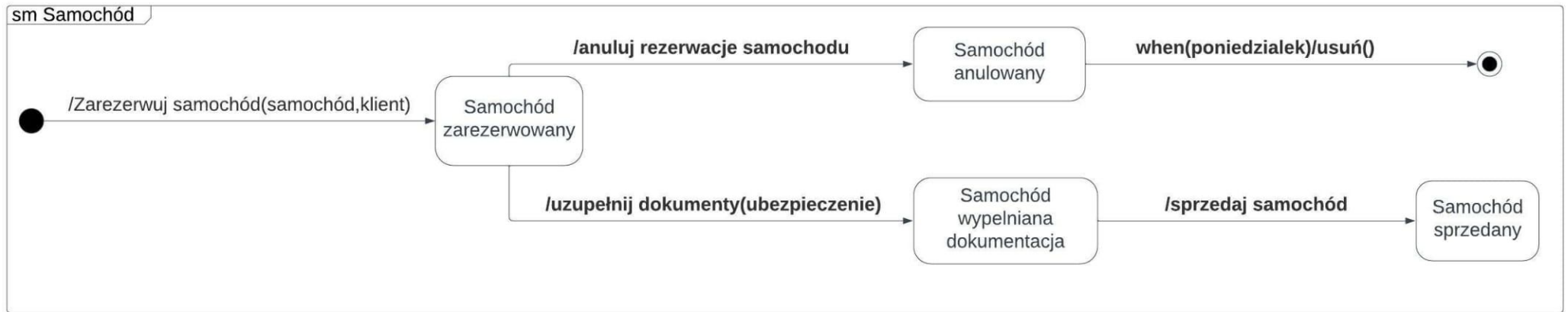
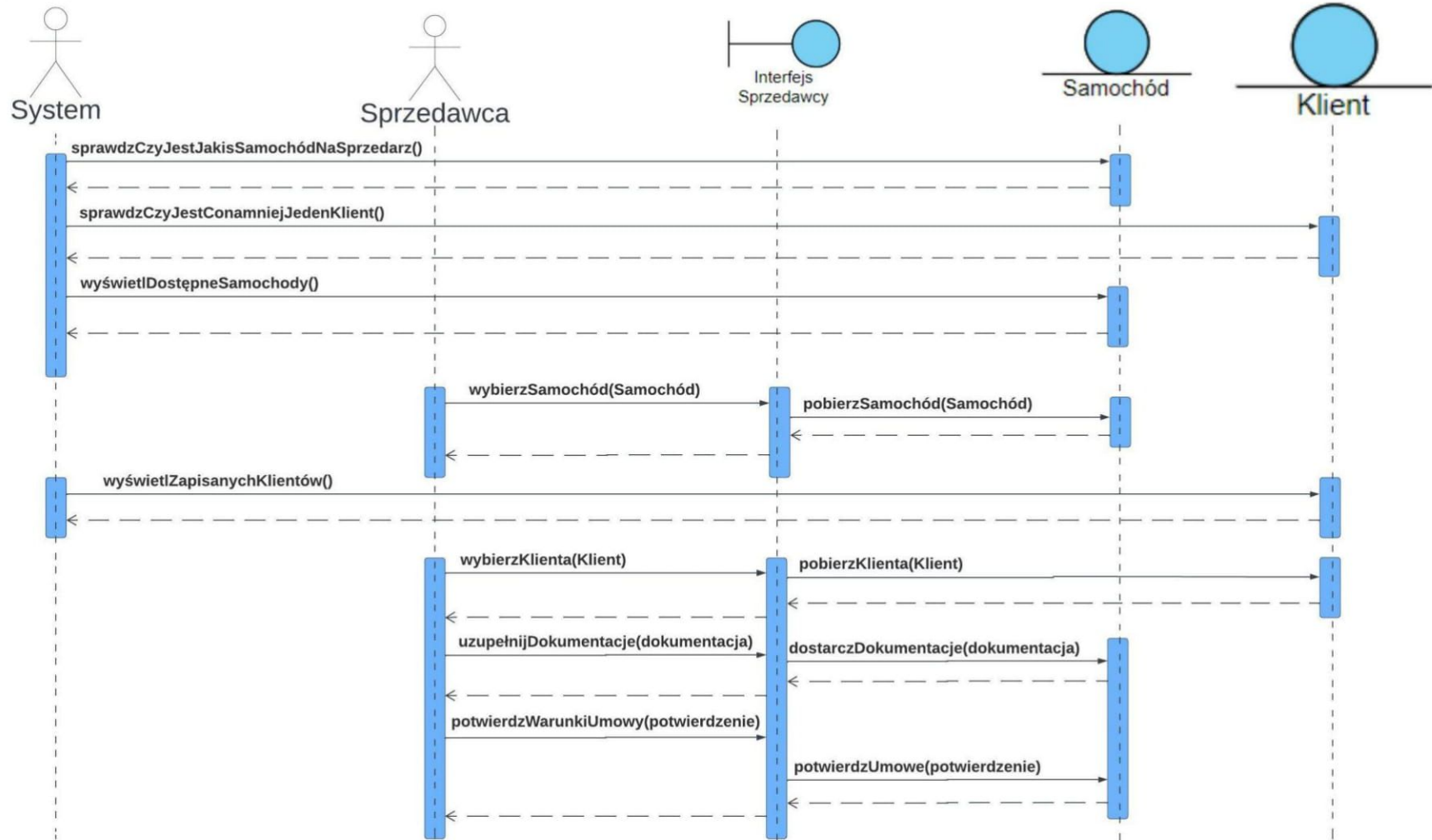


Diagram interakcji (sekwencji) dla przypadku użycia

sd.Sprzedaj_samochod



• **Omówienie decyzji projektowych i skutków analizy dynamicznej**

1. Dziedziczenie typu overlapping w klasach Klient i Sprzedawca będzie zastąpione przez spłaszczenie hierarchii, a pola oraz metoda w klasie Sprzedawca będą dodane do klasy Osoba.
2. Ograniczenie odrered zostało nałożone na klasę Osoba. Obiekty przechowywane w ekstensji, będą posortowane według kolejności alfabetycznej imienia.
3. Ograniczenie atrybutu nie więcej niż 10 znaków zostało nałożone na atrybut imię w klasie Osoba
4. W klasie Osoba został wprowadzony atrybut opcjonalny nazwisko panieńskie, ze względu na to, że przechowywanie jego wartości nie jest wymagane.
5. Atrybut pochodny cena w klasie Auto będzie wyliczany na podstawie kosztu modyfikacji. Podczas tworzenia obiektu Auto, należy odjąć od ceny koszt modyfikacji Auta.
6. Ograniczenie unigue nałożone na atrybut vin (klasa Osoba) zapewni jego unikalność. Unikalność w programie będzie realizowane za pomocą atrybutu klasowego HashSet przechowujący wszystkie dotychczasowe występujące wartości.
7. Atrybut klasowy przebieg w klasie Auto będzie zaimplementowany przy użyciu słowa kluczowego static, ze względu na to że jest taki sam dla wszystkich klas – równy 0.
8. Asocjacja z atrybutem – klasy Klient i Auto - została zrealizowana za pomocą klasy pośredniczącej Zakup
9. Dziedziczenie dyniamic – klasa Auto standardowe i klasa Auto z modyfikacjami (Auto standardowe może stać się Autem z modyfikacjami, jeżeli Auto standardowe można modyfikować), zostanie zrealizowane za pomocą dwóch kompozycji z ograniczeniem xor. Jeżeli warunek modyfikacji jest spełniony, zastępujemy stary obiekt Auto standardowe, nowym obiektem Autem z modyfikacjami. Jedynie informacje wspólne dla obu obiektów są zachowywane, natomiast informacja o możliwości modyfikacji zostaje zapomniana.
10. Atrybut powtarzalny w klasie sprzedawca oraz atrybut powtarzalny modyfikacja w klasie Auto z modyfikacjami będzie zaimplementowany za pomocą listy.
11. Klasą nadrzędną dla klasy Ubezpieczenie jest klasa Ubezpieczenie OC i Ubezpieczenie AC. Ze względu na to w języku java wielodziedziczenie nie występuje, dodano interfejs

IUbezpieczenie OC. Klasa Ubezpieczenie będzie dziedziczyć po klasie Ubezpieczenie AC oraz będzie implementować interfejs IUbezpieczenie OC, metodę dodaj zakres ubezpieczenia().

12. Aby na podstawie numeru naprawy można było otrzymać odpowiadającą naprawę zastosowano asocjacje kwalifikowaną pomiędzy klasami Auto i Naprawa. Dostęp do obiektu celowego Naprawa będzie odbywała się na podstawie unikatowego kwalifikatora: numer naprawy.
13. Hermetyzacja (enkapsulacja) danych, została zrealizowana poprzez wprowadzenie w każdej klasie pola prywatnego, natomiast dla metod zastosowano publiczny modyfikator dostępu (brak rozbudowanych metod).
14. Trwałość ekstensji będzie zapewniona dzięki użyciu mechanizmu serializacji, który pozwoli na binarny zapis obiektów całego drzewa obiektów.