

# Inżynieria Oprogramowania

## 2. etap – sprawozdanie

*Wykonanie opisu biznesowego „świata rzeczywistego” projektowanego oprogramowania.  
Definicja wymagań funkcjonalnych i нефункциональных projektowanego oprogramowania.*

## **0. Temat projektu:**

Tematem projektu jest stworzenie oprogramowania...

## **1. Opis „świata rzeczywistego”:**

### **1.1. Opis zasobów ludzkich:**

- Pracownik techniczny serwisu może dodawać i usuwać świadczone usługi.
- Ponadto zarządza on utworzoną kartą klienta do którego dopisywane są świadczone usługi.
- Dla aktualnej karty klienta (tym samym jego rachunek) świadczone mu usługi są wyświetlane wraz z ich ceną.
- Na końcu umieszczone jest podsumowanie (suma) cen za wszystkie usługi.
- Pracownik serwisu może dowolnie edytować usługi oraz generować rachunki dla poszczególnych klientów.
- Nie może tych rachunków edytować.
- Zgłoszenie serwisu podlega kontroli (znaczniki wykonania danego etapu tj. wymiana oleju, myjnia etc.).
- Na każdym etapie serwisu powinna istnieć możliwość oznaczenia pojazdu, jako gotowego (tuż po etapie) poprzez wpisanie jego numeru rejestracyjnego.
- Dodane zgłoszenia serwisu można jedynie wstrzymać lub anulować.
- Każda usługa składa się z nazwy, ceny i kategorii, a także słownego opisu służącego do specyfikacji przez pracowników serwisu zakresu świadczonej usługi.
- Każda z usług będzie także posiadała nieedytowalny przez pracownika licznik, który sprawdza ile razy dana usługa została zakupiona.
- Całościowy serwis pojazdu jest priorytetowany.
- Rachunek dla klienta będzie rozpoznawany po numerze rejestracyjnym.
- Program będzie posiadał także możliwość generowania rachunków danego klienta lub historii ostatnich transakcji do pliku tekstowego, który będzie mógł być wydrukowany.
- System będzie możliwy do obsługi jedynie dla pracowników serwisu uprawnionych do korzystania z niego.

### **1.2. Przepisy:**

- Pracownik ponosi odpowiedzialność za poprawność wprowadzanych danych klienta oraz świadczonych mu usług.
- Aplikacja ma być maksymalnie odciążona ze zbędnych funkcjonalności, które generowałyby zbędne opóźnienia.

### **1.3. Dane techniczne:**

- Zakres usług dla danego klienta dodawany jest raz na początku tworzenia karty, edytowalna będzie jedynie notatka.
- System będzie stosowany na terenie serwisu samochodowego zlecającego wykonanie oprogramowania.
- Serwis samochodowy znajduje się na peryferiach miasta.
- Firma posiada kilkunastu pracowników wyspecjalizowanych do świadczenia usług.

- Pracownicy zostaną przeszkoleni w zakresie korzystania z oprogramowania.
- Oprogramowanie wykonane w technologii Java.

## **2. Wymagania:**

Na podstawie opisu świata rzeczywistego należało opracować wymagania funkcjonalne oraz wymagania niefunkcjonalne. Poniżej są one prezentowane:

### **2.1. Wymagania funkcjonalne:**

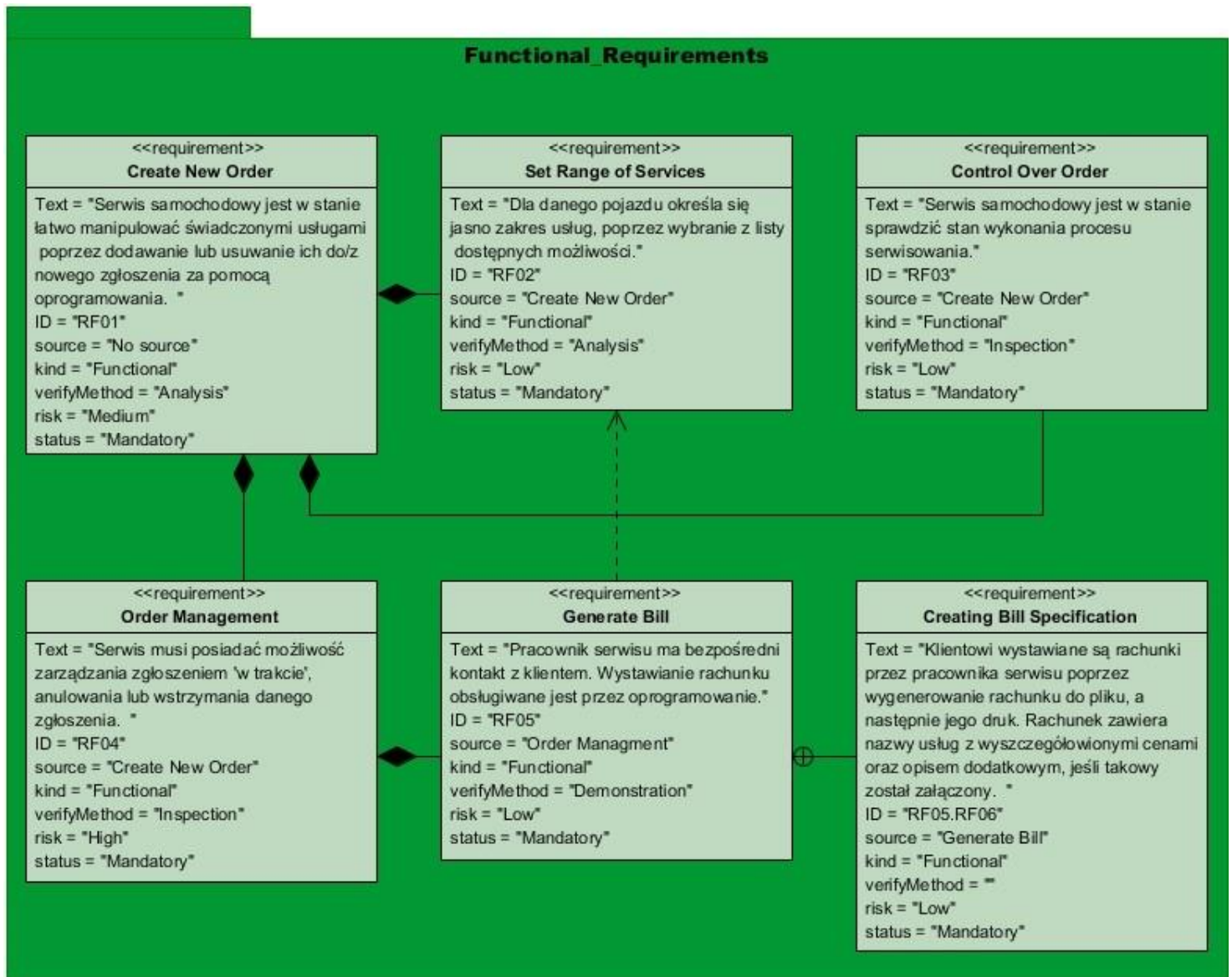
- Serwis samochodowy jest w stanie łatwo manipulować świadczonymi usługami poprzez dodawanie lub usuwanie ich do/z nowego zgłoszenia za pomocą oprogramowania.
- Dla danego pojazdu określa się jasno zakres usług, poprzez wybranie z listy dostępnych możliwości.
- Serwis samochodowy jest w stanie sprawdzić stan wykonania procesu serwisowania.
- Serwis musi posiadać możliwość zarządzania zgłoszeniem 'w trakcie', anulowania lub wstrzymania danego zgłoszenia.
- Pracownik serwisu ma bezpośredni kontakt z klientem. Wystawianie rachunku obsługiwane jest przez oprogramowanie.
- Klientowi wystawiane są rachunki przez pracownika serwisu poprzez wygenerowanie rachunku do pliku, a następnie jego druk. Rachunek zawiera nazwy usług z wyszczegółowionymi cenami oraz opisem dodatkowym, jeśli takowy został załączony.

### **2.2. Wymagania niefunkcjonalne:**

- Pracownik serwisu powinien zostać przeszkolony w zakresie wykorzystania oprogramowania.
- Pracownicy odpowiedzialni za oznaczanie pojazdu, jako gotowego na każdym etapie powinni posiadać podstawową wiedzę dot. obsługi komputera oraz sposobu oznaczania pojazdu, jako gotowe.
- System powinien pracować w popularnych systemach (LINUX, Windows).
- System powinien być elastyczny na rozwijanie o nowe funkcjonalności.
- Szata graficzna systemu powinna być czytelna i zrozumiała dla pracownika, natomiast drukowane rachunki czytelne dla klienta. Przy zachowaniu zasady minimalizmu.

### 3. Diagramy:

#### 3.1. Diagram wymagań funkcjonalnych:



### 3.2. Diagram wymagań niefunkcjonalnych:

