

**Akademia Górniczo-Hutnicza  
im Stanisława Staszica  
w Krakowie**

**Wydział Elektrotechniki, Automatyki Informatyki i Inżynierii  
Biomedycznej**

**Studia Podyplomowe Inżynieria Oprogramowania**

**Elementy Inżynierii Oprogramowania**

**Temat:  
Wypożyczalnia rowerów**

**Karol Brzozowski  
Gr 1.  
28.02.2014r.**

## **1. Opis Systemu**

W mieście znajduje się pewna ilość stacji rowerowych(będących tak naprawdę automatycznymi parkingami rowerowymi) w których możemy pobrać i oddać rower. Rowery nie są zwykłymi modelami, które możemy kupić w sklepie, ale muszą zostać wykonane na zamówienie operatora systemu. Najważniejszą ich cechą jest możliwość łatwego wypinania i oddawania do stacji - każdy operator stosuje swój własny system. Rower możemy pożyczyć z dowolnej stacji i oddać go również do dowolnej stacji. Specjalny skomputeryzowany system dba o autoryzację oraz rozliczenie czasu jazdy. Aby zachęcić użytkowników w większości systemów pierwsze pół godziny jazdy jest bezpłatne. Jeśli czas ten przekroczymy system pobierze od nas opłatę, która będzie rosła w miarę upływu czasu. Chodzi o to żeby rowery były cały czas w użyciu. Dzięki temu więcej osób może skorzystać z takiego transportu. Bardzo ważny jest sposób zapłaty za usługę, oraz problem minimalizowania strat związanych z uszkodzeniem i kradzieżą rowerów. Zazwyczaj jest to realizowane za pomocą karty magnetycznej, lub chipowej, która można "naładować" określoną kwotą pokrywającą kaucję oraz opłaty za przejazd.

## **2. Lista bodźców zewnętrznych**

Złożenie przez klienta wniosku rejestracyjnego w systemie i wyrobienie karty.

Wypożyczenie roweru.

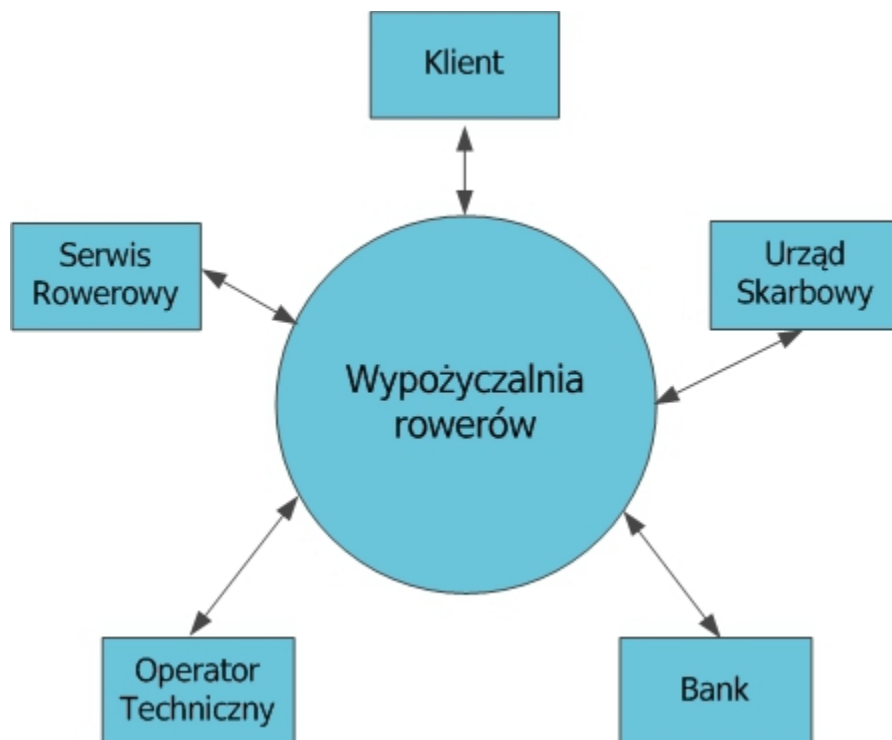
Zwrot roweru.

Doładowanie karty.

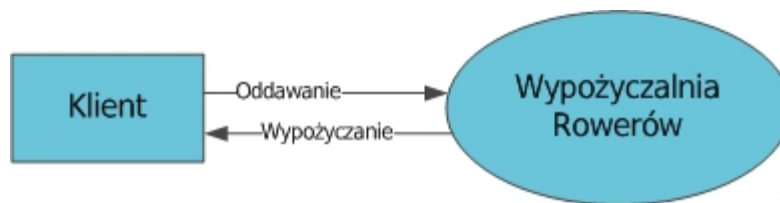
Zapłata za usługę

Wystawienie faktury

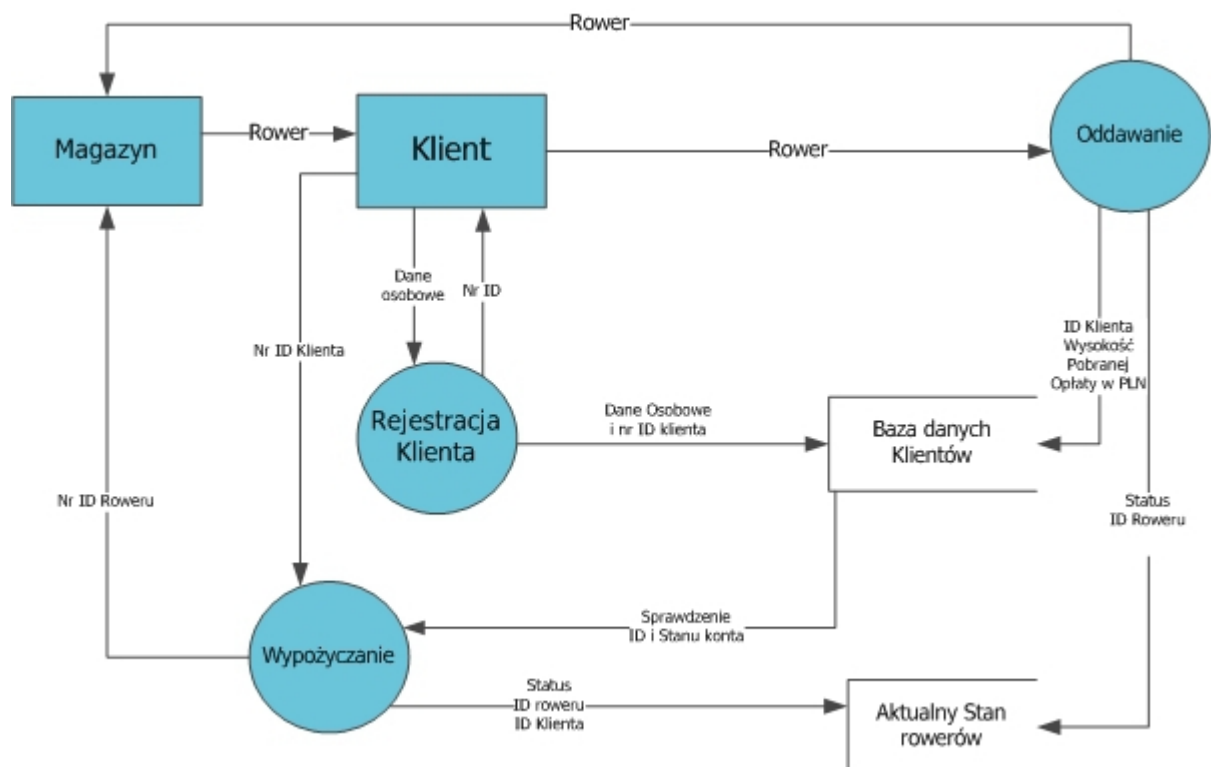
### 3. Diagram kontekstowy



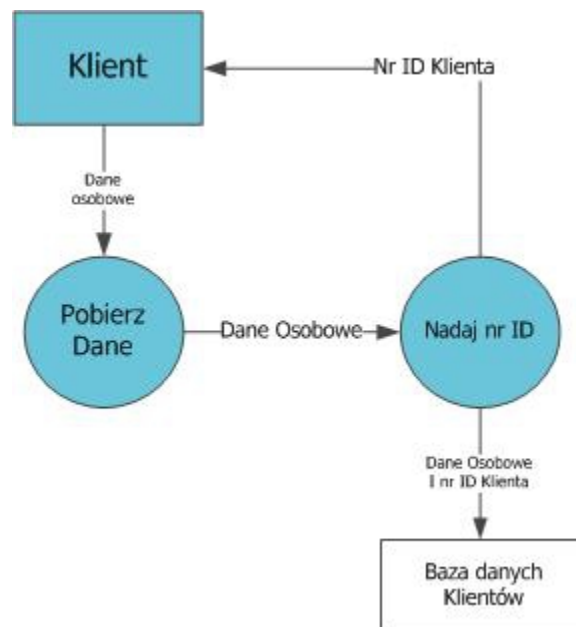
## 4. Model DFD



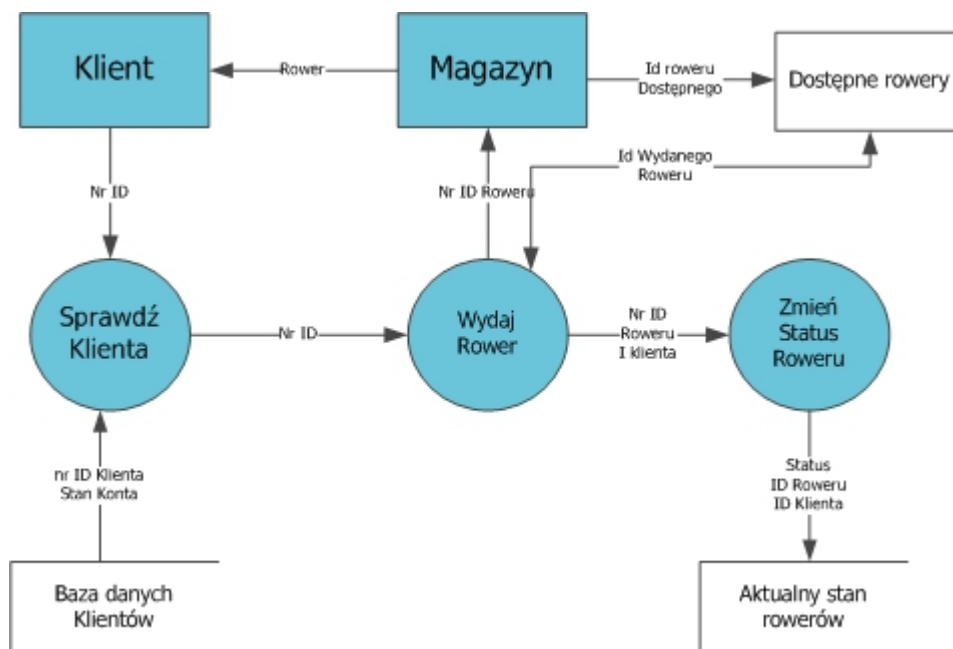
### 4.1 Dekompilacja



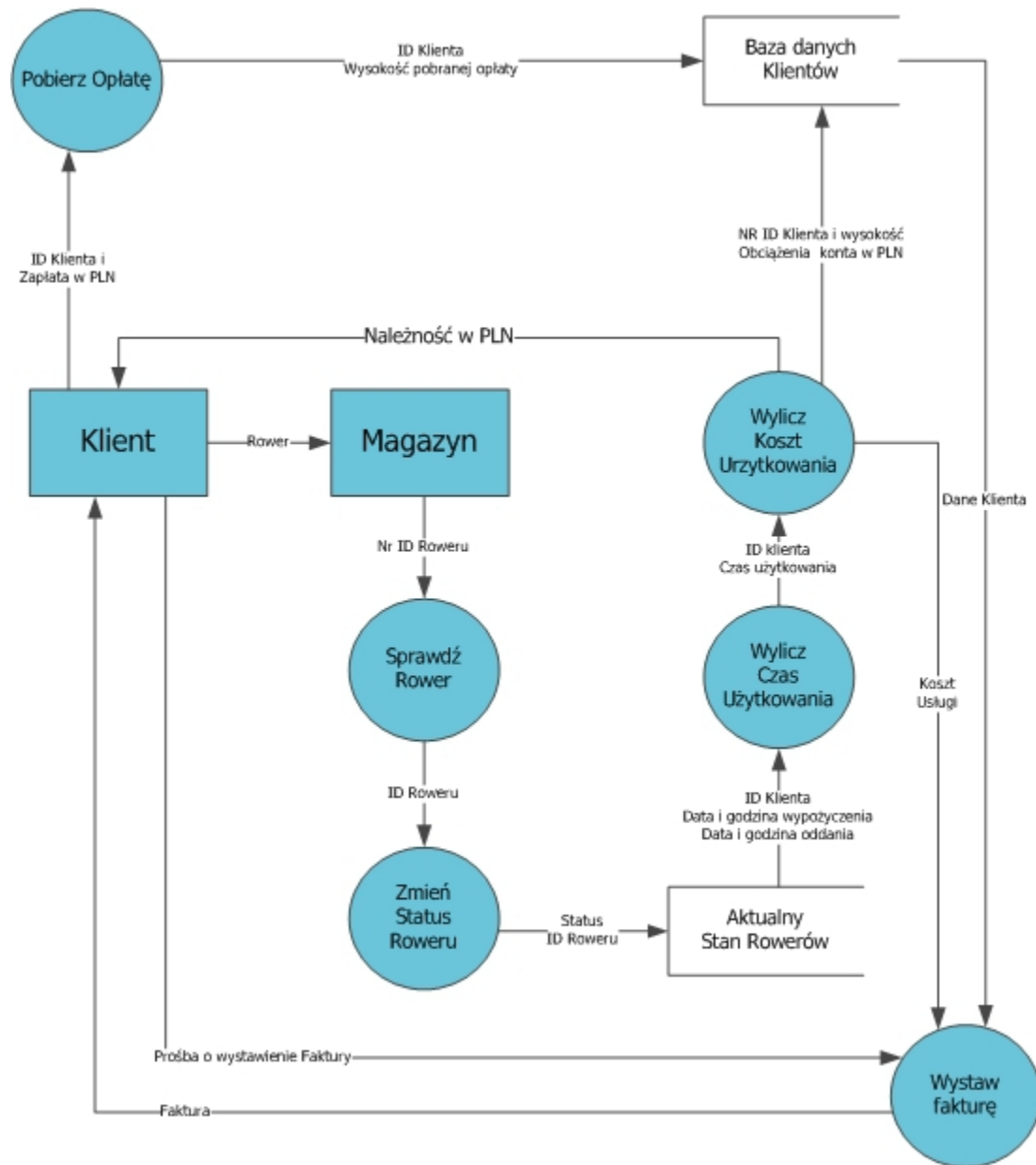
#### 4.1.1 Rejestracja Klienta



#### 4.1.2 Wypożyczenie



#### 4.1.3 Oddawanie



## 5. Specyfikacja procesów P-SPEC

### 5.1 Rejestracja Klienta

#### Specyfikacja procesu „Pobierz Dane”

**Wejście:** dane\_od\_klienta

**Wyjście** zaakceptowane\_dane

```
if( dane_od_klienta == TRUE){  
    zaakceptowane_dane = dane_od_klienta;  
}  
else{  
    generowanie komunikatu o błędzie;  
}
```

#### Specyfikacja procesu „Nadaj Numer”

**Wejście:** zaakceptowane\_dane

**Wyjście** nr\_id\_klienta, zaakceptowane\_dane

```
if( zaakceptowane_dane == TRUE){  
    generowanie: nr_id_klienta;  
}  
else{  
    generowanie komunikatu o błędzie;  
}
```

## 5.2 Wypożyczanie

### Specyfikacja procesu „Sprawdź Klienta”

**Wejście:** nr\_id\_klienta, stan\_rachunku\_klienta

**Wyjście** nr\_id\_klienta

```
if( nr_id_klienta == TRUE && stan_rachunku_klienta >= 0){  
    generowanie komunikatu o prawidłowości;  
    nr_id_klienta: udostępniony jest następnemu Procesowi;  
}  
  
else{  
  
    generowanie komunikatu o błędzie;  
}
```

### Specyfikacja procesu „Wydaj Rower”

**Wejście:** nr\_id\_klienta, nr\_id\_roweru, baza danych: dostępne\_rowery

**Wyjście** nr\_id\_klienta, nr\_id\_roweru

- 1.wyświetlenie\_dostępnych\_rowerów();
2. wybranie przez klienta roweru;
3. pobranie\_id\_roweru();
4. skasuj(); // skasowanie nr\_id\_roweru w bazie dostępne\_rowery
5. przekazanie nr\_id\_roweru do magazynu w celu wydania roweru,
6. przekazanie nr\_id\_klienta i nr\_id\_roweru następnemu procesowi



### Specyfikacja procesu „Zmień Status Roweru”

**Wejście:** nr\_id\_klienta, nr\_id\_roweru, data\_wypożyczenia, godzina\_wypożyczenia,

**Wyjście** nr\_id\_wypożyczenia, status

```
przypisz(nr_id_klienta, nr_id_roweru, data_wypożyczenia, godzina_wypożyczenia)
{
    nr_id = nr_id_klienta + nr_id_roweru + data_wypożyczenia + godzina_wypożyczenia
}
if( rower_wydany = True)
{
    status = 1;
    nr_id = nr_id + status;
}
nr_id = nr_id_wypożyczenia;
zapisz( nr_id_wypożyczenia); // zapisanie w bazie nr_id_wypożyczenia.
```

## 5.3 Oddawanie

### Specyfikacja procesu „Sprawdź Rower”

**Wejście:** nr\_id\_roweru,

**Wyjście** nr\_id\_roweru,

```
if( nr_id_roweru == TRUE){
    prześlij(nr_id_roweru); // przesyłanie numeru do kolejnej operacji.
}
else{
    generowanie komunikatu o niewłaściwym nr_id_roweru;
}
```

### Specyfikacja procesu „Zmień Status Roweru”

**Wejście:** nr\_id\_roweru, data\_oddania, godzina\_oddania, nr\_id\_wypożyczenia

**Wyjście** nr\_id\_oddania, status.

wyszukaj(nr\_id\_roweru); *// wyszukuję w bazie po nr\_id\_roweru, numeru przypisania o klienta.*

przypisz(nr\_id\_klienta, nr\_id\_roweru, data\_wypożyczenia, godzina\_wypożyczenia)  
}

nr\_id\_oddania = nr\_id\_roweru + data\_oddania + godzina\_oddania  
}

if( rower\_oddany == TRUE)

{

status = 0;

nr\_id\_oddania = nr\_id\_oddania + status;

}

zapisz( nr\_id\_oddania); *// zapisanie w bazie numeru nr\_id\_oddania;*

### Specyfikacja procesu „Wylicz Czas”

**Wejście:** nr\_id\_wypożyczenia, nr\_id\_oddania

**Wyjście** czas\_użytkowania, nr\_id\_klienta

wyszukaj\_nr\_id\_wypożyczenia(nr\_id\_oddania)

wyszukanie nr\_id\_wypożyczenia; *// na podstawie numeru oddania zostaje  
wyszukany nr\_id wypożyczenia.*

wylicz\_status(nr\_id\_oddania, nr\_id\_wypożyczenia)

generowanie : status\_oddania, status\_wypożyczenia;

if(status\_oddania == 0 && status\_wypożyczenie == 0){

generowanie\_id\_klienta(nr\_id\_wypożyczenie);

```
wylicz_czas(nr_id_wypożyczenia, nr_id_oddania)
    generowanie: czas_użytkowania;
przekazanie czas_użytkowania, nr_id_klienta do dalszych procesów;
}
else{
    generowanie komunikatu o błędzie;
}
```

### **Specyfikacja procesu „Wylicz koszt”**

**Wejście:** czas\_użytkowania, nr\_id\_klienta

**Wyjście** koszt\_użytkowania

```
wylicz_koszt(czas_użytkowania){
    generowanie: koszt_użytkowania;
}

obciążenie_rachunku(koszt_użytkowania, nr_id_klienta){
    stan_rachunku = stan_rachunku – koszt_użytkowania;
    zapisanie nowej wartości stan_rachunku przy nr_id_klienta
}

przeslij_do_klienta(koszt_użytkowania){
    przesyłanie koszt_użytkowania // przesłanie wysokości opłaty do klienta.
}
```

### **Specyfikacja procesu „Pobierz Opłatę”**

**Wejście:** nr\_id\_klienta, zapłata\_od\_klienta

**Wyjście** zapłata\_od\_klienta,

```
wpłata_na_rachunek(zapłata_od_klienta, nr_id_klienta){
    stan_rachunku = stan_rachunku + zapłata_od_klienta;
    zapisanie nowej wartości stan_rachunku przy nr_id_klienta
}
```

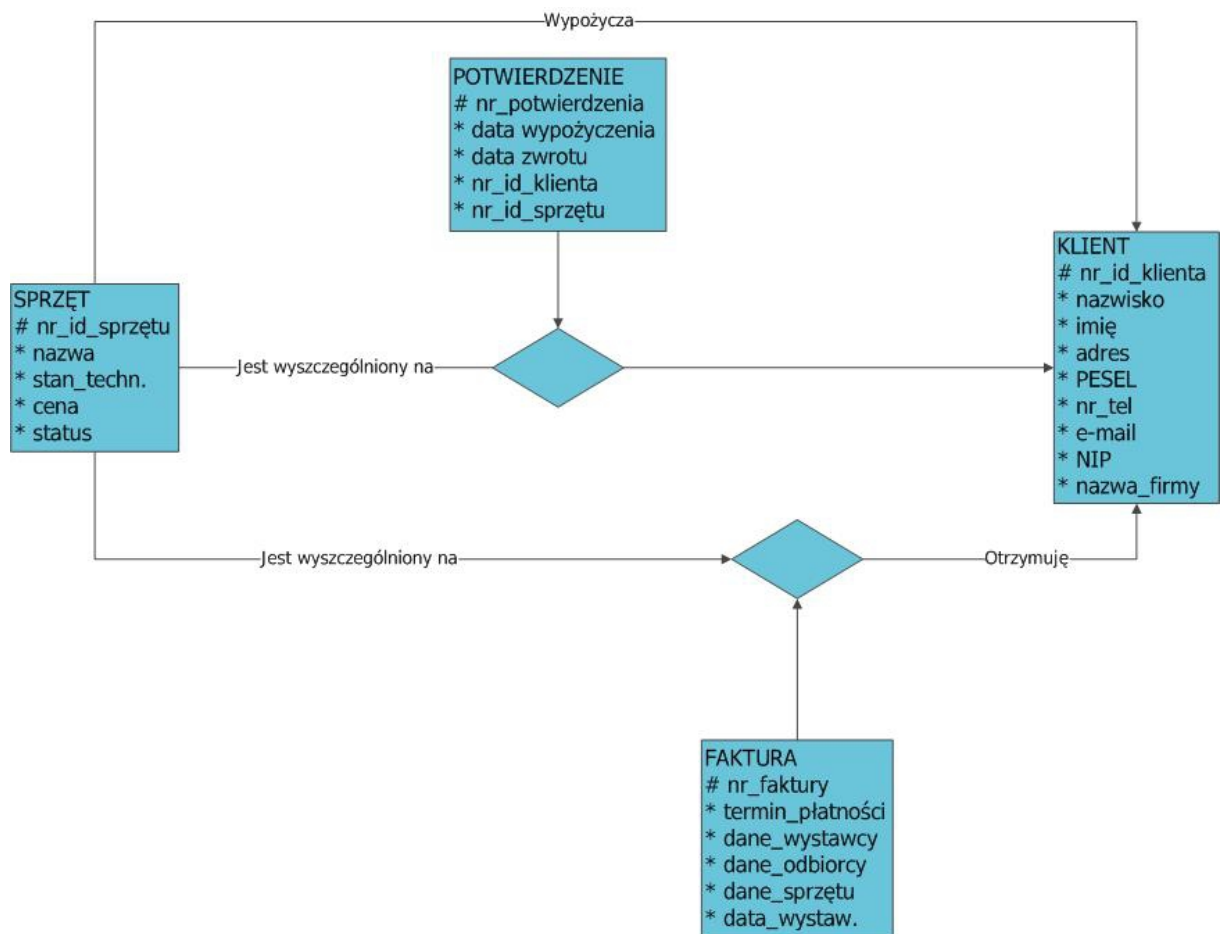
## Specyfikacja procesu „Wystaw Fakturę”

**Wejście:** koszt\_użytkowania, dane\_klienta, nr\_id\_klienta

**Wyjście** faktura

```
if(nr_id_klienta == TRUE){  
    wystaw_faktura(koszt_użytkowania, dane_klienta)  
    generowanie: faktura  
}  
else{  
    generowanie: komunikat o błędzie  
}
```

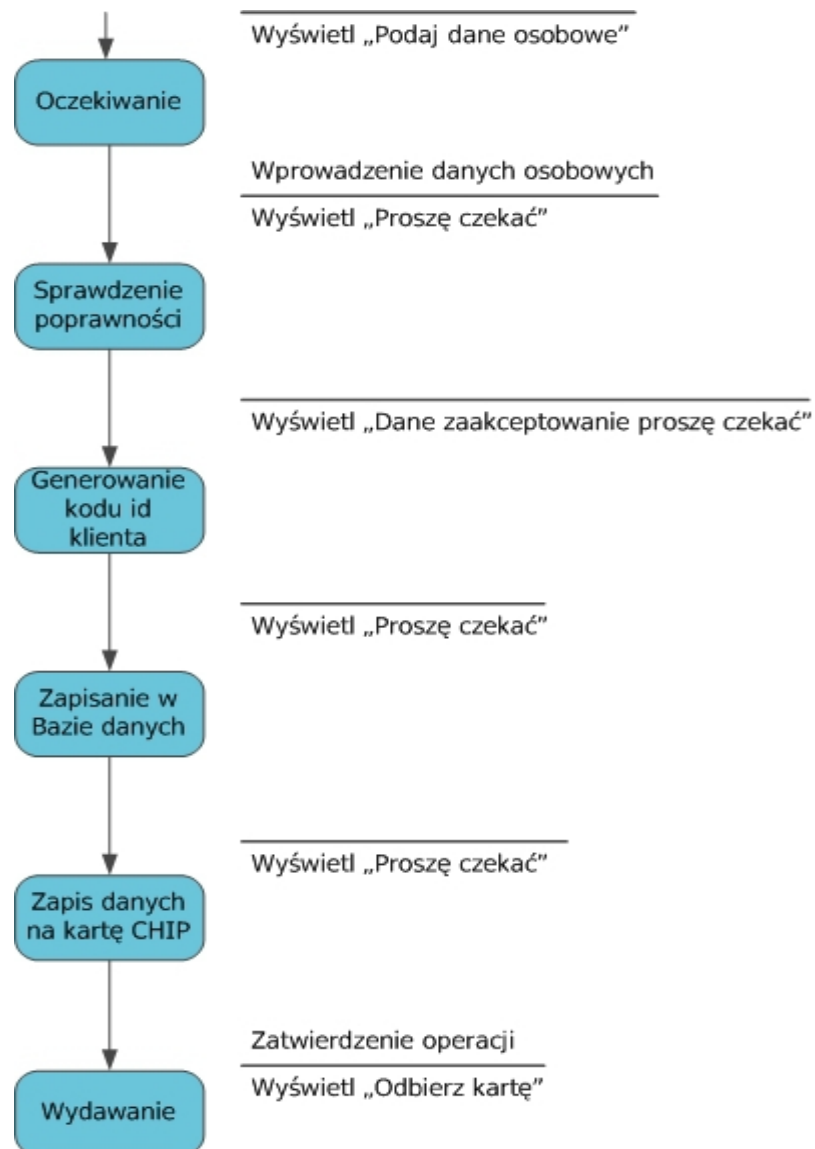
## 6. Diagram ERD związków encji



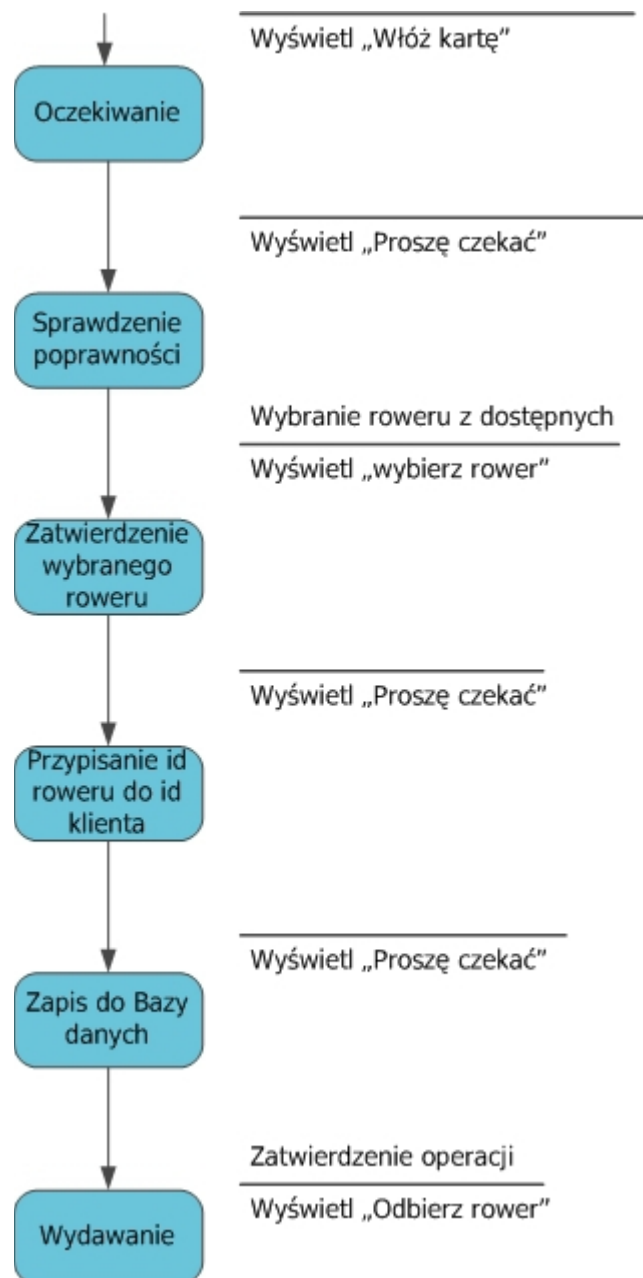
## 7. Diagram STD model zachowań



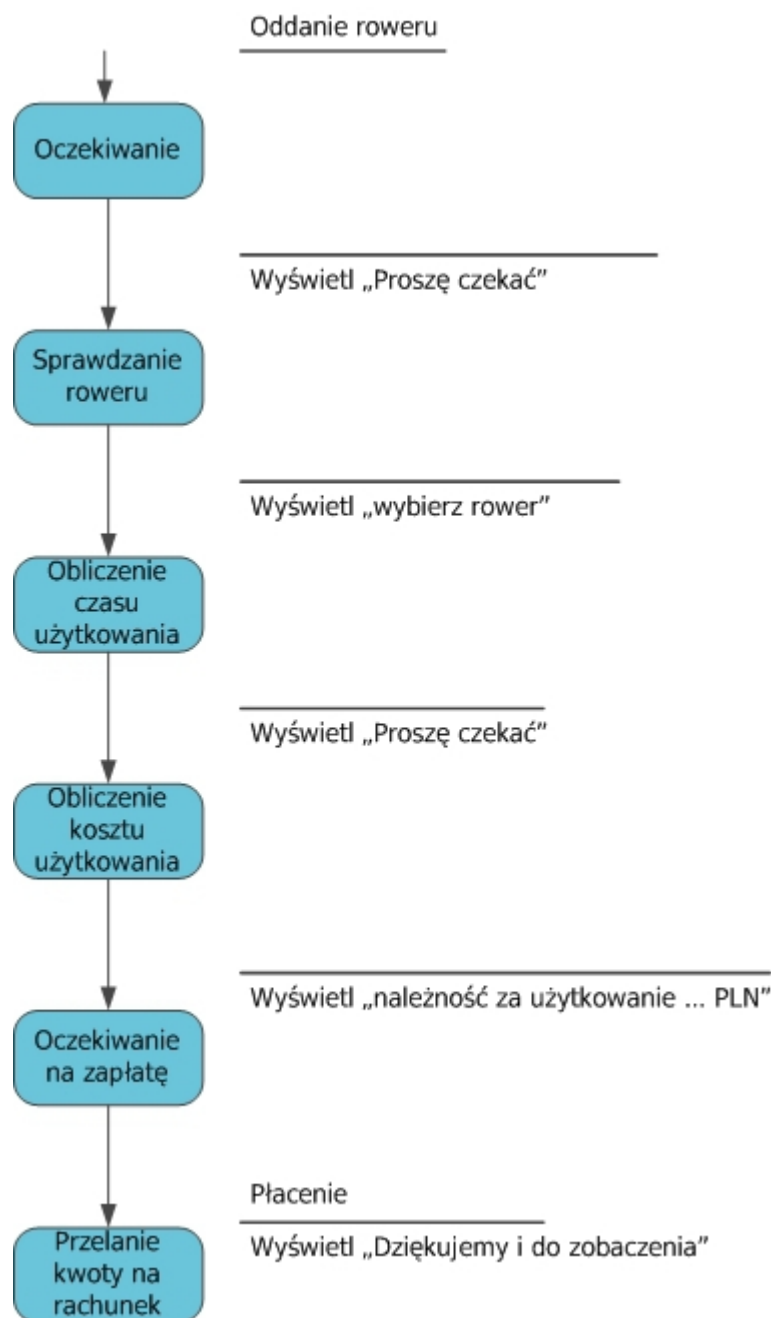
### 7.1 Model zachowań dla „Rejestracja Klienta”



## 7.2 Model zachowań dla „Wypożyczanie”



### 7.3 Model zachowań dla „Oddawanie”



## 8. DD - słownik danych Data Dictionary

**ROWER = nr\_id\_sprzętu + nazwa + stan\_techiczny + status + cena**

nr\_id\_sprzętu = {cyfra}

nazwa = {dozwolony znak}

stan\_techiczny = [dobry | zły]

status = [wypożyczony | wolny]

cena = cena za godzinę wypożyczenia {cyfra}

dozwolony\_znak = [ A – Z | a – z | 0 – 9 | - | | ]

liczba = [ 0 – 9 ]

**KLIENT = #nr\_id\_klienta + nazwisko + imię + adres + NIP + PESEL + nr\_tel  
+ e-mail + nazwa\_firmy**

# nr\_id\_klienta = {cyfra}

nazwisko = {litera}

imię = {litera}

adres = ulica + nr\_domu + nr\_mieszkania + kod + miejscowość

ulica = {dozwolony\_znak}

nr\_domu = {dozwolony\_znak}

nr\_mieszkania = {cyfra}

kod = {dozwolony\_znak2}

miejscowość {dozwolony\_znak}

PESEL = 11 znakowa {cyfra}

NIP = 12 znakowa {cyfra}

nr\_tel = {cyfra}

e-mail = {dozwolony\_znak}

nazwa\_firmy = {dozwolony\_znak}

cyfra = [0 – 9]

litera = [A – Z | a – z | |]

dozwolony\_znak = [A – Z | a – z | 0 – 9 | - | / | ]



**POTWIERDZENIE = #nr\_potwierdzenia + data\_wypożyczenia + data\_zwrotu +  
nr\_id\_klienta + nr\_id\_sprzętu**

#nr\_potwierdzenia = {dozwolony\_znak}

data\_wypożyczenia = {cyfra}

data\_zwrotu = {cyfra}

nr\_id\_klienta = {cyfra}

nr\_id\_sprzętu = {cyfra}

cyfra = [0 – 9]

dozwolony\_znak = [A – Z | a – z | 0 – 9 | - | / | . | |]

**FAKTURA = #nr\_faktury + termin\_płatności + dane\_wystawcy + dane\_klienta +  
dane\_sprzętu + data\_wystaw.**

# nr\_faktury = {dozwolony\_znak}

termin\_płatności = {cyfra2}

dane\_wystawcy = nazwa + adres + NIP + Regon

nazwa = {dozwolony\_znak}

adres = ulica + nr\_domu + nr\_mieszkania + kod + miejscowość

ulica = {dozwolony\_znak}

nr\_domu = {dozwolony\_znak}

nr\_mieszkania = {cyfra}

kod = {cyfra2}

miejscowość {dozwolony\_znak}

NIP = 12 znakowa {cyfra2}

Regon = 13 znakowa {cyfra}

dane\_odbiornicy = nazwisko + imię + adres + NIP

nazwisko = {litera}

imię = {litera}

adres = ulica + nr\_domu + nr\_mieszkania + kod + miejscowość

ulica = {dozwolony\_znak}

nr\_domu = {dozwolony\_znak}

nr\_mieszkania = {cyfra}  
 kod = {dozwolony\_znak2}  
 miejscowość {dozwolony\_znak}  
 dane\_sprzętu = nr\_id\_sprzętu + nazwa + stan\_tech. + cena + status  
 nr\_id\_sprzętu = {cyfra}  
 nazwa = {dozwolony znak}  
 stan\_techiczny = [dobry | zły]  
 status = [wypożyczony | wolny]  
 cena = cena za godzinę wypożyczenia {cyfra}  
 data\_wystawienia = {cyfra2}  
 cyfra = [0 – 9]  
 cyfra2 = [0 – 9 | / | .]  
 dozwolony\_znak = [A – Z | a – z | 0 – 9 | - | / | . | |]

#### **Dodatkowe użyte nazwy:**

**dane\_od\_klienta** = nazwisko + imię + adres + NIP + PESEL + nr\_tel  
 + e-mail + nazwa\_firmy  
**zaakceptowane\_dane** = TRUE/FALSE  
**status** = 0/1  
**stan\_rachunku\_klienta** = {cyfra od 0 – 9 || +/-}  
**nr\_id\_wypożyczenia** = nr\_id\_klienta + nr\_id\_roweru + data\_wypożyczenia +  
 godzina\_wypożyczenia + status  
**nr\_id\_oddania** = nr\_id\_roweru + data\_oddania + godzina\_oddania + status  
 czas\_użytkowania = (data\_oddania – data\_wypożyczenia)\*24 +  
 +(godzina\_oddania – godzina\_wypożyczenia)  
**koszt\_użytkowania** = czas\_użytkowania \* cena/h