

STUDENCKA PRACOWNIA BAZ DANYCH

INSTYTUT INFORMATYKI  
UNIwersytetu Wrocławskiego

Łukasz CZAPLIŃSKI

---

Dokumentacja projektu

# Coffee Shop

DOKUMENTACJA OGÓLNA  
WERSJA 1.0

---

Wrocław 2014

Tabela 1. Historia zmian

Wersja	Data	Opis	Autor
1.0	2014-06-13	Powstanie dokumentu	Łukasz Czapliński

## Spis treści

<b>1. Wprowadzenie</b>	3
1.1. Cel dokumentu	3
<b>2. Zadania projektu</b>	3
<b>3. Model</b>	3
3.1. Model konceptualny	3
3.2. Model fizyczny	3
3.2.1. Dostawca	3
3.2.2. Klient	3
3.2.3. Właściciel	4
3.2.4. Typ produktu	4
3.2.5. Produkt	4
3.2.6. Zamówienie	4

## 1.1. Cel dokumentu

## 2. Zadania projektu

### 3. Model

### 3.1. Model konceptualny

Jest przedstawiony na rys. [1].

Można w nim wyróżnić 3 główne role: klient, dostawca i właściciel oraz 3 obiekty którymi operują: produkty, ich typy oraz zamówienia.

Klient składa zamówienia do właściciela na konkretne produkty.

Właściciel dodaje nowe produkty wybierając z oferty dostawców.

Dostawca rejestruje jakie typy produktów w jakiej cenie może zapewnić.

Zamawianie dostaw nie jest kontrolowane przez tą bazę danych – odbywa się pomiędzy właścicielem a dostawcą przez umówione kontakty – maile.

### 3.2. Model fizyczny

### 3.2.1. Dostawca

```
create sequence doid_seq;
create table dostawca (
  doid integer primary key default nextval( 'doid_seq' ),
  mail text unique check (mail ~ '([a-z]||[.,-]|\d)+)@(([a-z]||[_, -]||
);
```

### 3.2.2. Klient

```
drop table if exists kupujacy cascade;
drop sequence if exists kuid_seq cascade;

create sequence kuid_seq;
```

```

create table kupujacy (
    kuid integer primary key default nextval('kuid_seq'),
    mail text unique check (mail ~ '([a-z]|[_,-]|\d)+@(([a-z]|[_,-]|
);

```

### 3.2.3. Właściciel

```

create sequence wlid_seq;
create table wlasciciel (
    wlid integer primary key default nextval('wlid_seq'),
    mail text unique check (mail ~ '([a-z]|[_,-]|\d)+@(([a-z]|[_,-]|
    marza decimal(3,2) not null check ( marza < 1 and marza > 0)
);

```

### 3.2.4. Typ produktu

```

create sequence tpid_seq;
create table typ_produktu (
    tpid integer primary key default nextval('tpid_seq'),
    nazwa text ,
    opis text
);

```

### 3.2.5. Produkt

```

create sequence prid_seq;
create table produkt (
    prid integer primary key default nextval('prid_seq'),
    tpid integer not null references typ_produktu ,
    cena decimal(6,2) not null ,
    zaid integer null references zamowienie ,
    doid integer not null references dostawca ,
    wlid integer not null references wlasciciel
);

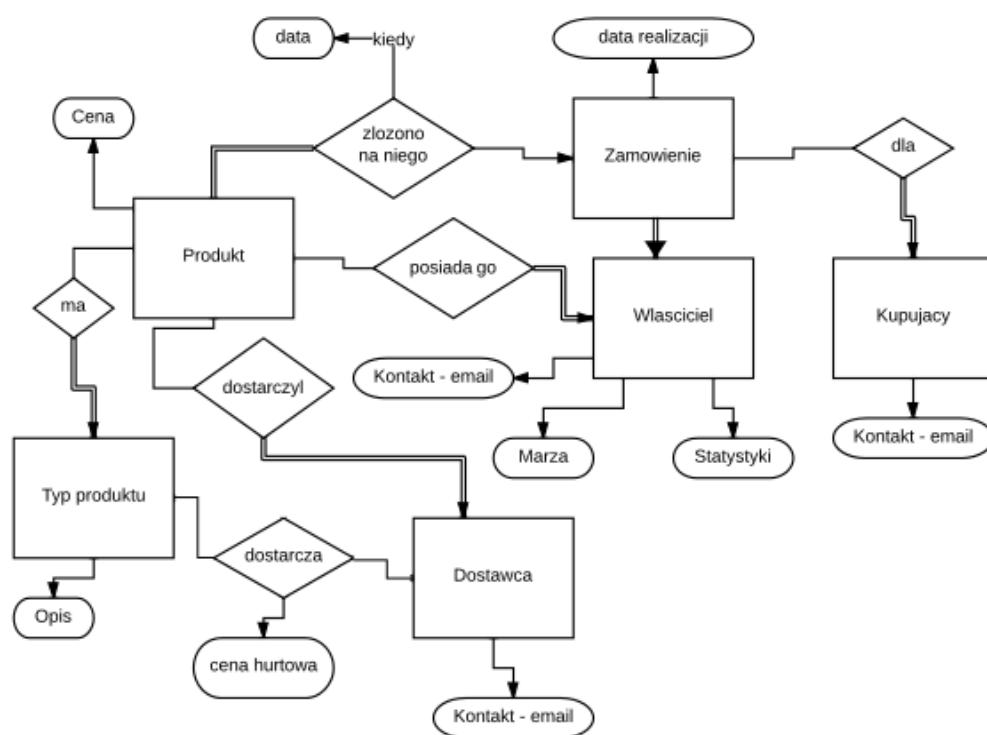
```

### 3.2.6. Zamówienie

```

create sequence zaid_seq;
create table zamowienie (
    zaid integer primary key default nextval('zaid_seq'),
    realizacja date null check (realizacja >= zlozenie),
    zlozenie date null ,
    wartosc integer not null default 0 ,
    kuid integer not null references kupujacy ,
    wlid integer not null references wlasciciel
);

```



Rysunek 1. Model konceptualny bazy danych