

# Dokumentacja projektu „System zarządzania hodowlą szczurów”

## Bazy Danych I

### Piotr Kumala

#### I. Projekt koncepcji, założenia

##### 1. Temat projektu:

System bazodanowy do zarządzania hodowlą szczurów. Pozwala pracownikom monitorować i zmieniać stan hodowli. Pozwala klientom hodowli składać zamówienia na szczury. Przed złożeniem zamówienia sprawdzana będzie wiedza klientów z zakresu znajomości hodowli szczurów.

##### 2. Analiza wymagań użytkownika:

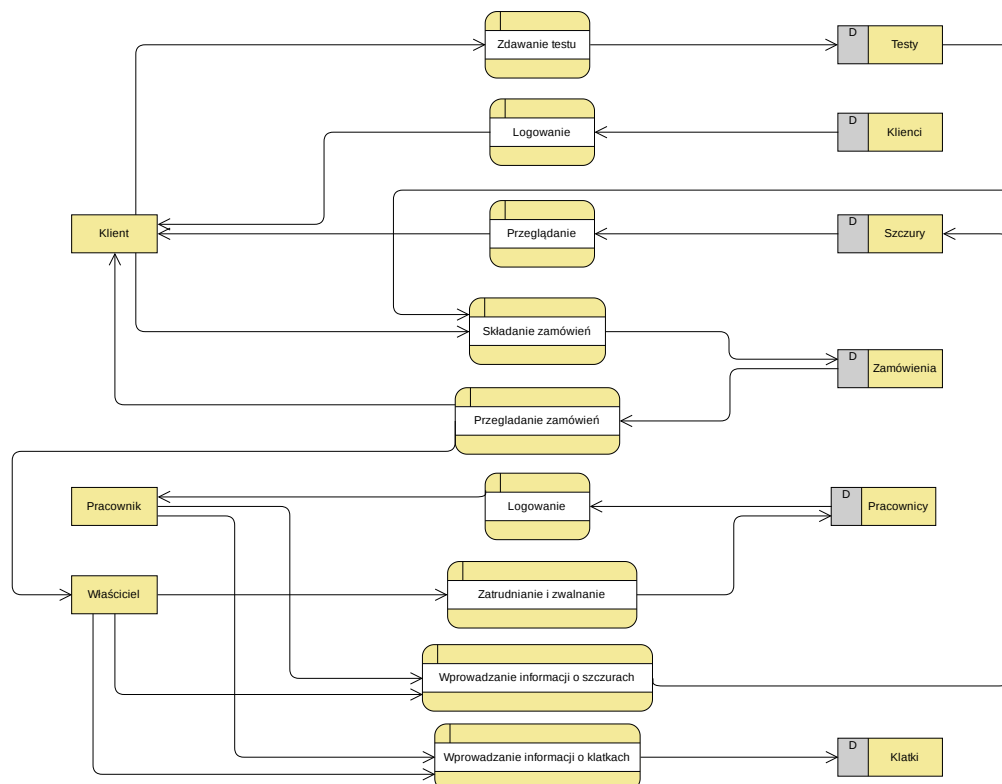
Baza danych musi przechowywać informacje o stanie hodowli, pozwalać na ich edytowanie. Powinna pozwalać użytkownikom na przeglądanie szczurów i składanie zamówień. Powinna być w stanie sprawdzać wiedzę klientów.

##### 3. Zaprojektowanie funkcji:

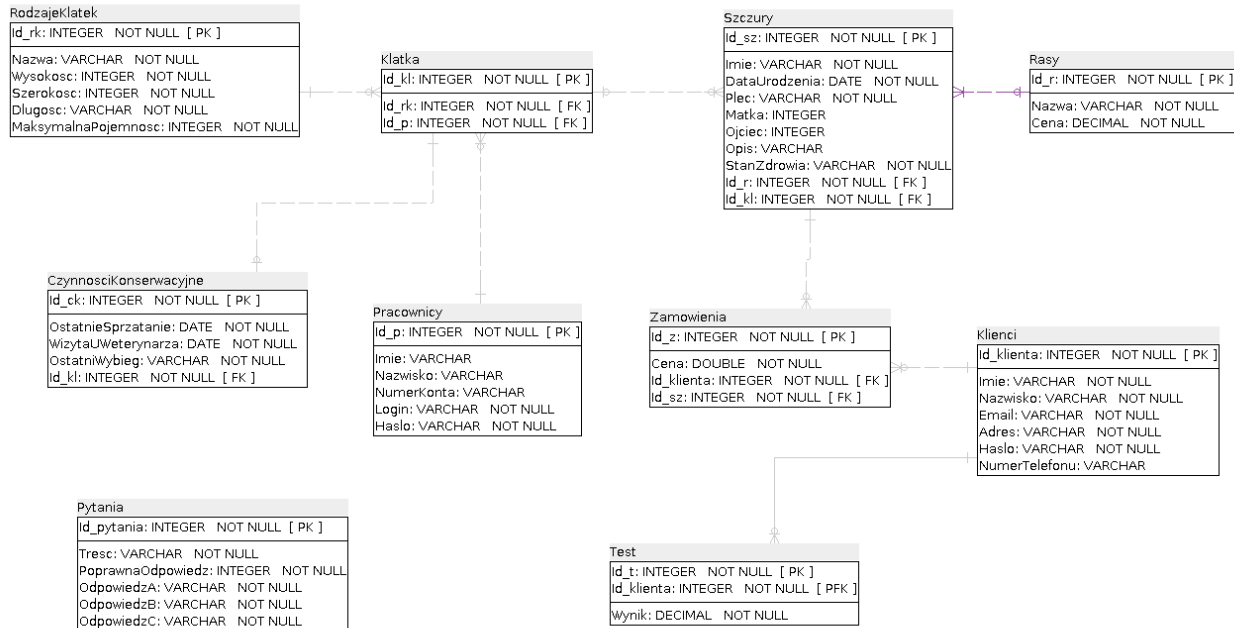
Pracownicy mogą przeglądać i edytować informacje o szczurach, klatkach, rozmieszczeniu szczurów w klatkach, czynnościach konserwacyjnych, które należy przeprowadzić w klatkach. Klienci powinni być w stanie składać zamówienia i zdawać test znajomości hodowli szczurów. Pracownicy mogą przeglądać zamówienia złożone przez klientów.

#### II. Projekt diagramów

##### 1. Diagram przepływu danych (DFD):



## 2. Diagram encji-relacji (ERD):



## III. Projekt logiczny:

### 1. Tabele:

Szczury:

Kolumny:

1. Id\_sz (INTEGER NOT NULL UNIQUE SERIAL)
2. Imie (VARCHAR NOT NULL)
3. DataUrodzenia (DATE NOT NULL)
4. Plec (VARCHAR NOT NULL)
5. Matka (INTEGER)
6. Ojciec (INTEGER)
7. Opis (VARCHAR)
8. StanZdrowia (VARCHAR NOT NULL)

Rasy:

Kolumny:

1. Id\_r (INTEGER NOT NULL UNIQUE SERIAL)
2. Nazwa (VARCHAR NOT NULL)
3. Cena (DECIMAL NOT NULL)

Klatka:

Kolumny:

1. Id\_kl (INTEGER NOT NULL UNIQUE SERIAL)

RodzajeKlatek:

Kolumny:

1. Id\_rk (INTEGER NOT NULL UNIQUE SERIAL)
2. Nazwa (VARCHAR NOT NULL)
3. Wysokosc (INTEGER NOT NULL)
4. Szerokosc (INTEGER NOT NULL)
5. Dlugosc (INTEGER NOT NULL)
6. MaksymalnaPojemnosc (INTEGER NOT NULL)

CzynnosciKonserwacyjne:

Kolumny:

1. Id\_ck (INTEGER NOT NULL UNIQUE SERIAL)
2. OstatnieSprzatanie (DATE NOT NULL)
3. WizytaUWeterynarza (DATE NOT NULL)
4. OstatniWybieg (DATE NOT NULL)

Pracownicy:

Kolumny:

1. Id\_p (INTEGER NOT NULL UNIQUE SERIAL)
2. Imie (VARCHAR NOT NULL)
3. Nazwisko (VARCHAR NOT NULL)
4. NumerKonta (VARCHAR)
5. Login (VARCHAR NOT NULL)
6. Haslo (VARCHAR NOT NULL)

Zamowienia:

Kolumny:

1. Id\_z (INTEGER NOT NULL UNIQUE SERIAL)
2. Cena(DOUBLE NOT NULL)

Klienci:

Kolumny:

1. Id\_klienta (INTEGER NOT NULL UNIQUE SERIAL)
2. Imie (VARCHAR NOT NULL)
3. Nazwisko (VARCHAR NOT NULL)
4. Email (VARCHAR NOT NULL)
5. Adres (VARCHAR NOT NULL)
6. Haslo (VARCHAR NOT NULL)
7. NumerTelefonu (VARCHAR)

Pytania:

Kolumny:

1. Id\_pytania (INTEGER NOT NULL UNIQUE SERIAL)
2. Tresc (VARCHAR NOT NULL)
3. PoprawnaOdpowiedz (INTEGER NOT NULL)
4. OdpowiedzA (VARCHAR NOT NULL)
5. OdpowiedzB (VARCHAR NOT NULL)
6. OdpowiedzC (VARCHAR NOT NULL)

Test:

Kolumny:

1. Id\_t (INTEGER NOT NULL UNIQUE SERIAL)
2. WYNIK (DECIMAL NOT NULL)

## **2. Relacje:**

- Rasy → Szczury (1 : N)
- RodzajeKlatek → Klatka (1 : N)
- CzynnościKonserwacyjne → Klatka (1 : 1)
- Pracownicy → Klatka (1 : N)
- Zamówienia → Szczury (1 : N)
- Klienci → Zamówienia (1 : N)
- Klienci → Test (1 : N)

## **IV. Projekt funkcjonalny:**

Do prezentacji danych stworzona została nakładka webowa. Stworzono formularze pozwalające na logowanie użytkowników i rejestrację nowych klientów. Formularze pozwalają na dodawanie nowych szczurów, edytowanie informacji o nich, usuwanie ich i składanie zamówień. Można również dodawać nowe klatki, nowe rodzaje klatek, usuwać je. Istnieje również możliwość dodawania i usuwania ras szczurów. Klient może również uzupełniać formularz testowy w celu zdania testu.

## **V. Dokumentacja:**

### **1. Wprowadzanie danych:**

Stworzony został skrypt populacyjny dodający dane testowe, dzięki którym nowy użytkownik będzie mógł się zapoznać z funkcjonalnościami aplikacji.

### **2. Dokumentacja użytkownika:**

Pracownik może dodawać i usuwać szczury, dodawać i usuwać klatki, rodzaje klatek, oglądać zamówienia złożone przez użytkowników. Klient po zarejestrowaniu się i zalogowaniu musi zdać test potwierdzający znajomość hodowli szczurów. Po zdaniu testu może zamawiać szczurki.

### **3. Dokumentacja techniczna:**

Aplikacja webowa jest napisana w języku PHP 7. W trakcie pisania aplikacji przyjęty został wzorzec projektowy MVC (Model – View – Controller). Dodawanie nowych funkcjonalności powinno odbywać się poprzez dodanie nowego kontrolera i modelu, zdefiniowanie ścieżki do tego nowego kontrolera w pliku index.php. Wszystkie operacje komunikujące się z bazą danych powinny odbywać się z poziomu modelu. Stworzone zostały klasy bazowe Controller i View, nowe kontrolery powinny dziedziczyć po klasie Controller. Klasa View ma stworzony mechanizm zaciągania szablonu, na podstawie którego tworzony będzie widok. Szablony powinny być przechowywane w folderze „templates” i posiadać rozszerzenie „.php”. W celu obsługi logowania użytkowników zaimplementowany został mechanizm sesji. Podczas logowania w sesji zapisywany jest numer identyfikujący użytkownika („Id”) oraz flaga „Type” określająca czy użytkownik jest klientem (jego wartość będzie wtedy wynosić „Klient”) czy pracownikiem hodowli (jej wartość będzie wtedy wynosić „Pracownik”).

