

Zadanie 3

Moja propozycja to utworzenie jednej tabeli użytkowników oraz wypełnianie odpowiednich pól zależnie od tego czy użytkownik zaznaczy że posiada firmę bądź nie. Pozwoli nam to na zapisywanie informacji o użytkowniku w jednym miejscu i dzięki temu nie będzie potrzebne zakładanie drugiej tabeli w której znajdowały by się informacje o firmie które mogły by być powtórzone i dostępne tylko i wyłącznie poprzez relacje z id.

Tabela "Użytkownicy":

Pole	Typ danych	Ograniczenia
ID	INT (klucz główny)	AUTO_INCREMENT
Typ użytkownika	ENUM('Osoba fizyczna', 'Firma')	NOT NULL
Imię	VARCHAR(255)	NOT NULL
Nazwa firmy	VARCHAR(255)	Jeśli "Typ użytkownika" = 'Firma'
Adres e-mail	VARCHAR(255)	UNIQUE, NOT NULL
Data urodzenia	DATE	Jeśli "Typ użytkownika" = 'Osoba fizyczna'
NIP	VARCHAR(10)	Jeśli "Typ użytkownika" = 'Firma'

Metody weryfikacji i ograniczenia:

- Unikalność adresu e-mail:** Poprzez zastosowanie ograniczenia UNIQUE na polu adresu e-mail, możemy mieć pewność że jeden email będzie przypisany tylko i wyłącznie raz.
- Warunkowe pola:** W zależności od wybranego typu użytkownika (osoba fizyczna lub firma), pola "Data urodzenia" lub "Nazwa firmy" oraz "NIP" są wymagane odpowiednio. Można to zaimplementować w kodzie tak aby dane zostały odpowiednio przesyłane poprzez wybranie bądź wypełnienie informacji przez użytkownika.
- Poprawność daty urodzenia:** W warstwie aplikacji można weryfikować, czy wprowadzona data urodzenia jest poprawna i nie przekracza obecnej daty.
- Poprawność adresu e-mail:** Można zastosować mechanizm walidacji adresu email poprzez wprowadzenie w aplikacji wyrażeń regularnych bądź odpowiednich funkcji w frameworkach.
- Poprawność NIP:** W przypadku pola NIP można zastosować algorytm walidacji numeru NIP, który sprawdzi, czy numer jest poprawny zgodnie z obowiązującymi zasadami.
- Ograniczenia ENUM:** Pole "Typ użytkownika" może być ograniczone do wybranych wartości (osoba fizyczna, firma) za pomocą typu wyliczeniowego (ENUM), co pomaga uniknąć wprowadzania nieprawidłowych danych.
- Zabezpieczenia przed SQL Injection:** Podczas tworzenia formularza i wprowadzania danych do bazy danych, należy pamiętać o zabezpieczeniach przed atakami SQL Injection. Wykorzystanie parametryzowanych zapytań SQL lub ORM zminimalizuje prawdopodobieństwo takich ataków.