|  |  |
| --- | --- |
| Projekt **Technologie obiektowe**  Wydział Elektrotechniki Automatyki i Informatyki  Politechnika Świętokrzyska | |
| Studia: **Stacjonarne II stopnia** | Kierunek: **Informatyka** |
| Grupa: **1ID22A** | **Autor:** Adam Słaby |
| **Dokumentacja techniczna:**  Modelowanie bazy danych | |

Spis treści

[1. Cel projektu 3](#_Toc72698521)

[2. Wstęp teoretyczny 3](#_Toc72698522)

[2.1 Użyte technologie 3](#_Toc72698523)

[2.2 Relacja jeden do jednego 4](#_Toc72698524)

[2.3 Relacja jeden do wielu 4](#_Toc72698525)

[2.4 Relacja wiele do wielu 5](#_Toc72698526)

[2.5 Dziedziczenie 5](#_Toc72698527)

# Cel projektu

Celem projektu było stworzenie aplikacji internetowej do projektowania diagramów ERD oraz do generowania kodu Sql na podstawie stworzonego diagramu. Projekt powinien wspierać takie relacje jak: jeden do jednego, jeden do wielu, wiele do wielu oraz relację dziedziczenia.

# Wstęp teoretyczny

## Użyte technologie

**Angular** - jest to framework front-endowy, który służy do tworzenia aplikacji SPA, czyli Single Page Application. Są to aplikacje składające się z jednej strony, gdzie zmieniają się tylko wewnętrzne komponenty, a nie ładowana jest cała strona od nowa. Jest on napisany w języku Typescript przez firmę Google. Dzięki temu frameworkowi można nie tylko tworzyć aplikacje webowe, ale także czyste strony internetowe, aplikacje mobilne oraz aplikacje desktopowe. Przykładem aplikacji, gdzie możemy zobaczyć jego zastosowanie jest np. główna strona internetowa firmy UPS czy też w głównym serwisie webowym firmy AT&T. Jego zaletami są: wysoka wydajność, szybkość działania, jasna i dobrze określona struktura budowy programu oraz szeroka baza dodatkowych komponentów.

**Bootstrap** jest to framework warstwy wizualnej, który został napisany przez programistów Twittera oraz wydany na licencji MIT. Pozwala on na tworzenie tak zwanych responsywnych stron internetowych, czyli takich, które automatycznie się dostosowują do wielkości ekranu. Był on także użyty jako podstawa do wielu systemów projektowych takich jak Argon, BLK, czy Material Design for Bootstrap 4.

Zalety Bootstrapa to:

* szybsze tworzenie funkcjonalnych interfejsów, stron i aplikacji internetowych,
* możliwość tworzenia stron niezależnych od przeglądarek internetowych z uwagi na to, że niweluje on drobne różnice w interpretowaniu stylów css przez te przeglądarki,
* łatwość w uzyskaniu wsparcia technicznego z powodu dużej popularności tego frameworka.

**MxGraph** jest to biblioteka do tworzenia diagramów w języku JavaScript, która umożliwia szybkie tworzenie interaktywnych wykresów i aplikacji do tworzenia diagramów, które działają natywnie w każdej większej przeglądarce obsługiwanej przez jej dostawcę.

**TypeScript** jest nadzbiorem JavaScript, co oznacza, że ​​zawiera wszystkie funkcje JavaScript. Każdy poprawny program napisany w języku JavaScript będzie również działał zgodnie z oczekiwaniami w języku TypeScript. W rzeczywistości TypeScript kompiluje się po prostu do zwykłego JavaScript. TypeScript oferuje nam większą kontrolę nad naszym kodem za pomocą adnotacji, typów, interfejsów i klas. Język ten został stworzony przez Microsoft i wydany w 2012 roku, po dwóch latach rozwoju. Został stworzony, aby umożliwić opcjonalne statyczne sprawdzanie typu, co byłoby szczególnie przydatne przy tworzeniu aplikacji na dużą skalę.

## Relacja jeden do jednego

Relacja jeden do jednego między tabelami A oraz B występuje wówczas, gdy każdemu rekordowi z tabeli A jest przyporządkowany tylko jeden rekord z tabeli B i na odwrót. Taki rodzaj relacji bardzo rzadko występuje w rzeczywistości. Jest ona stosowana np. wtedy, gdy zbiór dodatkowych atrybutów jest określony tylko dla wąskiego podzbioru wierszy w tabeli podstawowej.

Przykład takiej relacji w praktyce jest widoczny poniżej.



## Relacja jeden do wielu

Relacja jeden do wielu jest jedną z najczęściej używanych przy projektowaniu bazy danych. Występuje ona dla tabel A oraz B wówczas, gdy pojedynczemu rekordowi z tabeli A jest przyporządkowany jeden lub wiele rekordów z tabeli B, natomiast pojedynczemu rekordowi z tabeli B jest przyporządkowany dokładnie jeden rekord z tabeli A. Taka relacja występuje np. pomiędzy wydawcą a książką. Taka sytuacja oznacza, że wydawca może wydać wiele książek ale książka może być wydana tylko przez jednego wydawcę. W praktyce należy umieścić klucz obcy po stronie wiele, który będzie się odnosić do klucza głównego lub unikatowego po stronie jeden.

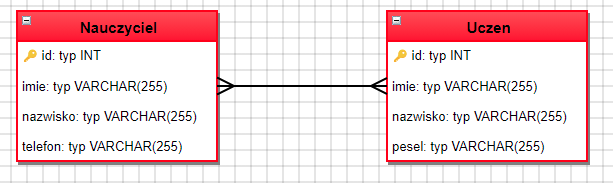
Poniżej znajduje się przykład użycia relacji w praktyce.



## Relacja wiele do wielu

Relacja wiele do wielu występuje pomiędzy tabelami A oraz B, gdy pojedynczemu rekordowi z tabeli A jest przyporządkowany jeden lub wielu rekordów z tabeli B, i na odwrót. Tego typu powiązanie będzie występować np. pomiędzy uczniami i nauczycielami, ponieważ jeden uczeń będzie nauczany przez wielu nauczycieli oraz jeden nauczyciel może kształcić wielu uczniów. W praktyce taki rodzaj relacji wymaga utworzenia tabeli dodatkowej, której klucz główny będzie się składał z dwóch kluczy obcych, gdzie jeden z nich będzie się odwoływał do tabeli A natomiast drugi do tabeli B.

Przykład zastosowania tej relacji znajduje się poniżej.



## Dziedziczenie

Dziedziczenie jest relacją występującą w modelu obiektowym i zachodzi między klasami, wtedy, gdy dana klasa (klasa pochodna) korzysta w swej definicji z definicji innej klasy (klasy bazowej)‏. W relacyjnych bazach danych nie występuje takie powiązanie pomiędzy tabelami jak dziedziczenie, gdyż model relacyjny jest uboższy od modelu obiektowego. Pomimo tego, że takie powiązanie nie występuje w relacjach pomiędzy tabelami, można je zasymulować korzystając z trzech strategii: SINGLE TABLE, JOINED, TABLE\_PER\_CLASS;

Przykład relacji dziedziczenia znajduje się poniżej.

