

Spis treści

I. Model konceptualny bazy danych	3
1. Opis „świata rzeczywistego”	3
2. Opracowanie wymagań funkcjonalnych i нефunkcjonalnych dla aplikacji bazodanowej:.....	5
3. Specyfikacja wymagań funkcjonalnych za pomocą diagramu przypadków użycia (łącznie z opisami poszczególnych przypadków użycia).....	7
4. Identyfikacja diagramu związków encji na podstawie analizy scenariuszy poszczególnych przypadków użycia (scenariusz, jako algorytm używający dane, które są źródłem identyfikacji encji)	8
.....	8
II. Projekt bazy danych	9
1. Analiza liczby instancji dla każdej encji	9
2. Analiza użycia identyfikująca podstawowe rodzaje transakcji: wstawianie, modyfikacja, usuwanie i wyszukiwanie oraz przemieszane ze sobą – określenie na tej podstawie zmienności zawartości poszczególnych tabel.....	10
3. Sformułowanie wymagań dotyczących dostępu – określenie częstości wykonania operacji na danych np. tworzenia raportów	12
4. Analiza integralności	14
5. Dostrajanie bazy danych pod względem wydajności (na podstawie punktów 1-4).....	16
III. Implementacja i testy bazy danych w wybranym systemie zarządzania bazą danych	18
1. Fizyczny projekt bazy danych:	18
2. Zbiór zapytań zoptymalizowanych	22
3. Polityka bezpieczeństwa.....	23

I. Model konceptualny bazy danych

1. Opis „świata rzeczywistego”

- Młodzieżowy Dom Kultury im. T. Sygietyńskiego znajduje się na ulicy Tysiąclecia 3 w Sieradzu. Jest to placówka oświatowo - kulturalna, która oferuje różnego rodzaju zajęcia w czasie pozaszkolnym dla dzieci i młodzieży. W budynku znajdują się pomieszczenia kulturalno – rozrywkowe, np. sala teatralno-kinowa, czytelnia, świetlica, sale dydaktyczne, a także sala gimnastyczna, w których można prowadzić pozaszkolną działalność i inicjatywy dydaktyczno – kulturalne (warsztaty kulturalne, przedstawienia, zabawy, zajęcia tematyczne).
- Podstawową misją MDK’u jest wychowanie i dydaktyka dzieci i młodzieży. Zaletą jest tutaj wzajemne partnerstwo wychowawców i młodzieży w zakresie wspólnych działań dydaktyczno – wychowawczych.
- Ośrodek tworzy indywidualne, grupowe i zbiorowe formy pracy wychowawczej. Swoje zadanie realizuje poprzez zapewnienie dzieciom i młodzieży możliwości atrakcyjnego spędzania wolnego czasu, a co za tym idzie pozwala na rozwijanie zainteresowań i uzdolnień młodzieży.
- Placówka zatrudnia kilkunastu dydaktyków posiadających potwierdzone osobiste predyspozycje i kompetencje poparte odpowiednim wykształceniem oraz przygotowaniem pedagogicznym i zawodowym do nauczania formalnego w oświacie.

W Młodzieżowym Domu Kultury działa wiele kółek zainteresowań oraz grup skupiających młodzież chcącą rozwijać swe uzdolnienia. W ośrodku działają następujące grupy zajęciowe:

- **taneczna** – zajęcia przebiegają w różnych formacjach i grupach tanecznych, tj.: taniec towarzyski, taniec nowoczesny, taniec ludowy, zajęcia rytmiczne. Aktywnie uczestnicząca młodzież może tutaj rozwijać swe zainteresowania i zdolności związane z tańcem i rytmiką
- **plastyczna** – umożliwia poznanie różnych dziedzin sztuk plastycznych: malarstwa, rzeźby, grafiki, ceramiki czy malarstwa na jedwabiu. Dodatkowo młodzież i dzieci mają możliwość poznania historii sztuki, przebywają kurs malarstwa czy grafiki, a także mogą rozwijać swoje zdolności pod czujnym okiem profesjonalistów z tych dziedzin
- **teatralna** – skupia osoby, które swe uzdolnienia i zainteresowania ulokowały w teatrze
- **modelarska** – skupia dzieci i młodzież zainteresowane sklejeniem modeli drewnianych (samolotów) oraz modeli sterowanych drogą radiową
- **muzyczna** – gromadzi dzieci i młodzież w różnych zespołach: gitarowych, keyboardowych, perkusyjnych, wokalnych. Uczestnicy swoje zainteresowania rozwijają z zakresu muzyki oraz śpiewu
- **gimnastyczna** – polepszanie koordynacji ciała oraz dynamiki ruchów
- **recytatorska** – recytowanie poezji, ćwiczenie dykcji, przygotowanie do konkursów recytatorskich
- **historyczna** – zapewnia właściwe przygotowanie do olimpiad przedmiotowych, zajmuje się organizacją widowisk historycznych

- **biologiczna** – zapewnia właściwe przygotowanie do olimpiad przedmiotowych, zajmuje się prezentacjami doświadczeń biologicznych
- **dziennikarska** – przygotowanie różnego rodzaju audycji radiowych
- **kulinarna** – warsztaty sztuki kulinarnej
- **językowa** – umożliwia naukę języków obcych (do wyboru jest nauka języka angielskiego oraz niemieckiego)

2. Opracowanie wymagań funkcjonalnych i нефunkcjonalnych dla aplikacji bazodanowej:

Uprawnienia dostępu do aplikacji posiadają pracownicy administracji oraz kadra dydaktyczna.

Założenia, co do wymagań funkcjonalnych obsługiwanych przez system:

- Przyjęcie przez system danych osobowych dzieci uczęszczających na zajęcia.

Funkcja ta umożliwia sprawdzenie czy w systemie znajdują się dana osoba oraz wprowadzenie do systemu nowego dziecka o danych osobowych: *imię, nazwisko, adres zameldowania, pesel, data urodzenia, nr telefonu, nr telefonu do prawnych opiekunów*. W razie potrzeby można modyfikować istniejące dane jak i usunąć „profil”.

- Dodanie dziecka do określonych zajęć.

Każda osoba posiadająca status ucznia będzie mogła zapisać się na dowolną liczbę zajęć, jak i wypisać się w razie potrzeby.

- Sprawdzanie obecności na zajęciach.

System będzie posiadał również funkcję dodawania obecności uczniów na zajęciach (widzianą „z zewnątrz” przez stronę web) co pozwoli rodzicom na swobodną kontrolę dziecka jak i również usunięcie uczęszczającego z zajęć w razie nadmiernych nie usprawiedliwionych nieobecności.

- Informacje o kadrze pedagogicznej.

W bazie danych znajdują się informacje jak i dane osobowe prowadzącej zajęcia kadry dydaktycznej. Możliwy jest wgląd w informacje jak i modyfikacja istniejących.

- Zamieszczanie udostępnionych materiałów dodatkowych do zajęć.

Kadra pedagogiczna ma możliwość zamieszczania dodatkowych materiałów do zajęć, które mogą służyć dzieciom to utrwalania wiedzy przyswojonej na zajęciach jak i samodzielnej pracy w domu.

- Harmonogram zajęć w postaci kalendarza.

Wszystkie zajęcia organizowane przez MDK będą umieszczone w szczegółowym harmonogramie czasu, który będzie umożliwiał jednorazową lub rekurencyjną zmianę terminu oraz sali.

- Dodanie zajęć nauczycielowi – dany nauczyciel może prowadzić różne zajęcia, np. Zajęcia ze śpiewu jak i nauka gry na flecie.

- Uczniowie zapisani na dane zajęcia – wyświetlenie danych osobowych uczniów uczęszczających na dane zajęcia.

Założenia, co do wymagań niefunkcyjnych obsługiwanych przez system:

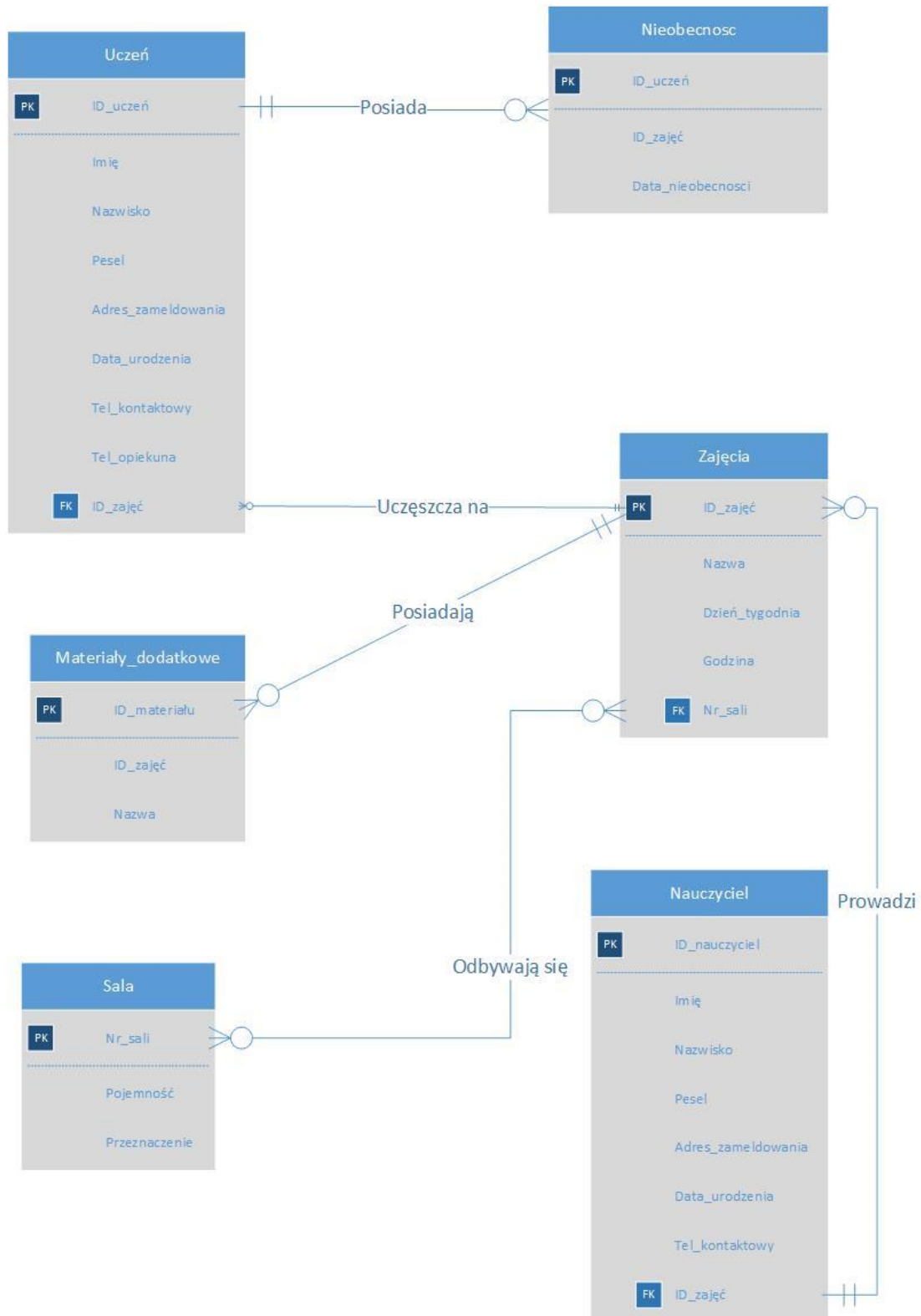
- Osoba zapisująca się do bazy MDK'u musi posiadać status ucznia bądź studenta.
- Uczeń jest zobowiązany do okazania zgody prawnego opiekuna na uczęszczanie w zajęciach.

3. Specyfikacja wymagań funkcjonalnych za pomocą diagramu przypadków użycia (łącznie z opisami poszczególnych przypadków użycia)



W powyższym diagramie przypadków użycia ukazaliśmy podstawowe funkcje jakie może wykonywać użytkownik systemu. W naszym przypadku jest to system zarządzania bazą danych w Młodzieżowym Domu Kultury, zaś użytkownikiem, który wykonuje różne czynności jest uczeń oraz nauczyciel.

4. Identyfikacja diagramu związków encji na podstawie analizy scenariuszy poszczególnych przypadków użycia (scenariusz, jako algorytm używający dane, które są źródłem identyfikacji encji)



II. Projekt bazy danych

1. Analiza liczby instancji dla każdej encji

Założenia wstępne co do liczby instancji dla każdej z encji:

- Zajęcia – w instytucji Młodzieżowego Domu Kultury jest 12 grup zajęciowych. Wstępnie przyjmujemy, że każde zajęcia dzielą się na dwie grupy zaawansowania oraz odbywają się 2 razy w tygodniu po 60 minut. Tabela zajęcia wstępnie będzie zawierała 24 rekordy. (12 grup * 2 grupy zaawansowania = 24 zajęcia w tygodniu)
- Uczni – przyjmujemy, że nasza instytucja Młodzieżowy Dom Kultury w ciągu roku (bo taka jest „ważność” przynależności ucznia w systemie) zawiera około 500 uczniów. Po tym czasie dane te przekazywane są do archiwizacji.
- Nauczyciel – w naszym systemie wstępna ilość kadry pedagogicznej wynosi 20 osób.
- Obecność – encja ta będzie najbardziej „wykorzystywaną” encją w systemie. W prosty sposób można obliczyć jej liczbę instancji:
$$\text{Średnia liczba instancji obecności} = 500 \text{ uczniów} * 2 \text{ zajęcia w tygodniu} * 50 \text{ tygodni w roku} = 50\,000 \text{ rekordów.}$$
- Sala – liczba instancji dla tej encji uzależniona jest od liczby sal dostępnych w budynku Młodzieżowego domu kultury, wstępnie przyjmujemy ok. 10 sal.
- Materiały dodatkowe – podejrzewamy, że encja ta będzie obciążona w niskim stopniu, oznacza to sporadyczne umieszczenie tam materiałów, ok. 10 MB danych materiałów na miesiąc.

2. Analiza użycia identyfikująca podstawowe rodzaje transakcji: wstawianie, modyfikacja, usuwanie i wyszukiwanie oraz przemieszane ze sobą – określenie na tej podstawie zmienności zawartości poszczególnych tabel

Na każdej z encji będą wykonywane pewne transakcje, które w pewnym stopniu obciążą przepustowość naszej bazy danych. Poniżej znajduje się wstępna deklaracja dostępnych funkcji wykonywanych na poszczególnych encjach:

I. Zajęcia:

- Wyszukaj informacje o zajęciach
- Dodaj zajęcia
- Modyfikuj zajęcia:
 - a. zmień ID zajęć
 - b. zmień nazwę
 - c. zmień termin odbywania się zajęć
 - d. dodaj ID materiału do zajęć
 - e. zmień salę, w której zajęcia się odbywają
- Usuń wybrane zajęcia

II. Uczeń :

- Wyszukaj informację o uczniu
- Dodaj ucznia
- Modyfikuj dane ucznia – zmiana: ID, imienia, nazwiska, itp.
- Usuń ucznia

III. Nauczyciel:

- Wyszukaj informację o nauczycielu
- Dodaj nauczyciela
- Modyfikuj dane nauczyciela
- Przypisz nauczyciela do grupy zajęciowej
- Usuń nauczyciela

IV. Obecność:

- zaznacz obecność
- odznacz obecność
- przeglądaj listę obecności
- generuj czasowy raport obecności

V. Sala:

- dodaj salę
- usuń salę
- modyfikuj dane sali

VI. Materiały dodatkowe:

- dodaj materiał
- usuń materiał

3. Sformułowanie wymagań dotyczących dostępu – określenie częstości wykonania operacji na danych np. tworzenia raportów

Dostęp do opracowywanego przez nas systemu Ewidencji uczniów Młodzieżowego Domu Kultury będą mieli głównie pracownicy administracji oraz dyrekcja, do kilku encji dostęp będzie miała również kadra nauczycieli. Określenie wymagań dotyczących dostępu oraz częstości wykonania operacji na danych przyczyni się w dużym stopniu to późniejszego dostrojenia bazy danych pod względem wydajności.

Przyjmijmy następujące grupy dostępowe:

1. Pracownicy administracji oraz dyrekcja
2. Kadra nauczycieli
3. Uczniowie

Poniżej w tabeli znajduje się określenie sformułowania wymagań dotyczących dostępu oraz częstości wykonania operacji na danych:

Encja	Grupa dostępu	Częstotliwość wykonywania transakcji	Częstotliwość w liczbach (założenia)
Zajęcia	1	Encja „Zajęcia” będzie znacznie obciążona w początkowym okresie zapisów, kiedy to tworzone są grupy zajęciowe, na które jest zapotrzebowanie oraz następnie do danych grup zajęciowych przypisywani są dani uczniowie.	W pierwszym miesiącu zajęć – 20 transakcji na dzień; W kolejnych miesiącach – 5 transakcji na tydzień.
Uczeń	1	Do tabeli „Uczeń” dane będą dodawane w dużej mierze wyłącznie w okresie zapisów – wrzesień oraz październik. W tym czasie encja ta będzie najbardziej obciążona.	W pierwszym miesiącu zajęć – 30 transakcji na dzień; W kolejnych miesiącach – 2 transakcje na tydzień.

		W późniejszym okresie transakcje na tej instancji będą wykonywane sporadycznie.	
Nauczyciel	1	Transakcje na tej encji będą wykonywane bardzo rzadko. Kadra pedagogiczna nie zmienia się często, a nowi nauczyciele dodawani są sporadycznie.	Operacje na ten encji będą wykonywane jeden lub dwa razy w roku.
Obecność	1, 2	Tabela obecność jest to tzw. Wąskie gardło naszego systemu, bowiem na niej wykonywane będą transakcje z największą częstotl., około 200 transakcji dziennie. Fakt, który działa na korzyść systemu jest taki, iż zmiana w tabeli „Obecność” nie powoduje zmian w pozostałych tabelach.	Operacje na ten encji będą wykonywane około 200 razy dziennie.
Sala	1	Podobnie do encji „Nauczyciel”, transakcje na tej encji będą wykonywane bardzo rzadko. Sala dodawana jest jedynie raz, następnie możliwa jest tylko modyfikacja jej nazwy, ID lub przeznaczenia.	Operacje na ten encji będą wykonywane jeden lub dwa razy w roku.
Materiały dodatkowe	1, 2, 3	Encja ta wykorzystywana będzie bardzo rzadko, ze względu na charakter większości zajęć. Jednak dostęp do niej mają wszystkie grupy dostępowe, co oznacza większą liczbę transakcji przez ciągły okres trwania semestru bądź roku szkolnego.	Operacje na ten encji będą wykonywane około 2 razy na tydzień

4. Analiza integralności

System tworzony przez nas bazy danych będzie posiadał odpowiednie mechanizmy zabezpieczające przed skutkami przypadkowych błędów logicznych. Ów system zapewni także możliwość sprawdzania i ewentualnej korekty wprowadzanych danych oraz będzie posiadał mechanizmy zapewniające prawidłowe przetwarzanie danych.

Analiza integralności w naszym przypadku polega na rozważeniu jak poszczególne operacje wykonywane na danych w jednych tabelach będą wpływać na dane znajdujące się w innych tabelach. Opis co powinno się stać z danymi w naszej bazie danych w wyniku wykonania jakiejś operacji przedstawia poniższa tabela:

Tabela	Operacja	Wpływ wykonania operacji na inne tabele
Zajęcia	Wysz. informację o zajęciach	-
	Dodaj zajęcia	W tabeli <i>Uczeń</i> oraz <i>Materiały Dodatkowe</i> pojawiają się nowe rekordy z ID dodanych zajęć
	Modyfikuj zajęcia	Modyfikacja poszczególnych atrybutów w tabeli <i>Zajęcia</i> będzie oznaczać modyfikacje w tabelach: <i>Uczeń</i> , <i>Materiały_dodatkowe</i> , <i>Sala</i> , <i>Nauczyciel</i>
	Usuń zajęcia	Zostaną usunięte (bądź przeniesione do archiwum) wszelkie materiały dodatkowe dołączone do zajęć. Uczniowie i nauczyciele zostaną wypisani z usuniętych zajęć. Zostanie „zwolniona” sala oraz termin zajęć
Uczeń	Wysz. informację o uczniu	-
	Dodaj ucznia	W tabeli <i>Obecność</i> pojawia się nowy rekord z ID dodanego ucznia

	Modyfikuj dane ucznia	Przy zmianie ID ucznia w tabeli <i>Uczeń</i> zmieni się ID ucznia w tabeli <i>Obecność</i>
	Usuń ucznia	Usunięcie ucznia oznacza usunięcie rekordów w tabeli <i>Uczeń</i> oraz <i>Obecność</i>
Nauczyciel	Wysz. infor. o nauczycielu	-
	Dodaj nauczyciela	-
	Modyfikuj dane nauczyciela	-
	Przypisz do grupy zajęciowej	W tabeli <i>Zajęcia</i> zostanie dodany nowy rekord
	Usuń nauczyciela	Usunięcie nauczyciela spowoduje, że do prowadzonych przez niego zajęć będzie trzeba przypisać nowego nauczyciela
Nieobecność	Zaznacz obecność	-
	Odznacz obecność	-
	Przeglądaj listę obecności	-
	Generuj czasowy raport obecności	-
Sala	Dodaj salę	-
	Usuń salę	Jeżeli do sali przypisane były zajęcia, powinna być możliwość przypisania nowego pomieszczenia do poszczególnych zajęć
	Modyfikuj dane sali	Jeśli zostanie zmodyfikowany nr sali, to ta zmiana zostanie propagowana do tabeli <i>Zajęcia</i>
Materiały Dodatkowe	Dodaj materiał	Materiał zostanie przypisany do konkretnych zajęć
	Usuń materiał	Materiał zostanie „odłączony” od konkretnych zajęć

Legenda:

„-” - oznacza że dana operacja nie ma wpływu na dane w pozostałych tabelach

5. Dostrajanie bazy danych pod względem wydajności (na podstawie punktów 1-4)

W projektowaniu bazy danych należy przede wszystkim zastanowić się, czy zamodelowany przez nas system będzie (po zaimplementowaniu) działał z zadowalającą nas wydajnością. Jeśli po zaimplementowaniu zorientujemy się iż pewne operacje na obiektach przechowujących dane, np. tabelach (modyfikacja, zapytania) wykonują się zbyt długo należy powrócić do stworzonego modelu bazy i wykonać odpowiednie czynności w celu poprawienia wydajności. W przypadku tworzonej przez nas bazy danych, aby „dostroić” ją pod względem wydajności można wykonać następujące czynności:

- W naszym projekcie do sprawdzania obecności służy tabela *Obecność*, która mówi nam czy dany uczeń obecny był na konkretnych zajęciach. Alternatywnym rozwiązaniem byłaby tabela *Nieobecność*, która mówiłaby nam, którzy uczniowie byli nieobecni na konkretnych zajęciach. Tabela ta zawierałaby znacznie mniej rekordów niż tabela *Obecność*, ponieważ zakładamy, że zdecydowanie więcej uczniów chodzi na zajęcia, niż je opuszcza. Dlatego podczas implementacji modelu systemu bazy danych (w przypadku gdyby wydajność okazała się nie zadowalająca) rezerwujemy możliwość zmiany pierwotnego rozwiązania na rozwiązanie alternatywne.
- Tabela *Uczniowie* oraz *Obecność* będą najbardziej rozbudowanymi tabelami w naszym systemie (będą posiadały dużą liczbę rekordów, spowodowaną dużą liczbą uczniów). W takim wypadku najlepiej zastosować indeksowanie kluczy głównych wyżej wymienionych tabel (w naszej sytuacji będzie to indeksowanie jednego klucza). Dzięki takiemu zabiegowi zapytania skierowane do owych tabel będą

zwracały wyniki znacznie szybciej, niż gdyby nie użyte zostało indeksowanie.

- Dzięki użyciu więzów integralności nie będzie można tak zmodyfikować danych by straciły one spójność:
 - każda tabela posiada swój klucz główny, który nakładany jest na kolumnę przechowującą dane jednoznacznie określające pojedynczy wiersz:

Tabela	Klucz główny (primary key)
Uczeń	ID_uczeń
Nieobecność	ID_uczeń
Materiały_dodatkowe	ID_materiału
Zajęcia	ID_zajęć
Nauczyciel	ID_nauczyciel
Sala	Nr_sali

- niektóre atrybuty w tabeli będą posiadać wartość *NOT NULL* (przy kluczach głównych wartość ta jest wstawiana domyślnie), która zapobiega wstawieniu *NULL'a* do kolumny; tymi atrybutami będą:

Tabela	Atrybuty NOT NULL
Uczeń	imię, nazwisko, pesel, adres_zameldowania, data_urodzenia, tel_kontaktowy
Nauczyciel	imię, nazwisko, pesel, adres_zameldowania, data_urodzenia, tel_kontaktowy
Materiały_dodatkowe	nazwa
Zajęcia	nazwa, dzień_tygodnia, godzina

- niektóre tabele będą posiadać klucz obcy służący do zdefiniowania relacji pomiędzy tabelami – rekord w tabeli

podrzędnej zawsze będzie miał swojego odpowiednika w tabeli nadrzędnej:

Tabela	Klucz obcy (foreign key)
Uczeń	ID_zajęć
Nauczyciel	ID_zajęć
Zajęcia	ID_materiału, Nr_sali

- niektóre atrybuty będą posiadały funkcję CHECK, która zapewnia, że wartość wstawiana do kolumny spełnia wymagany warunek logiczny (np. pesel musi posiadać określoną ilość cyfr)
- Zostanie przeprowadzona denormalizacja umożliwiająca przyspieszone wykonywanie zapytań

III. Implementacja i testy bazy danych w wybranym systemie zarządzania bazą danych

1. Fizyczny projekt bazy danych:

W etapie I projektu utworzono diagram związków encji, który jest podstawą implementacji bazy danych. Wybrano środowisko phpMyAdmin (narzędzie służące do zarządzania bazą danych MySQL napisane w języku PHP).

W pierwszym kroku utworzono bazę danych, następnie dodano do niej odpowiednie tabele, które zostały wypełnione przykładowymi rekordami. Podpunkty 1.1 oraz 1.2 przedstawiają szczegółową implementację dwóch

pierwszych kroków, w podpunkcie 1.3 zawarto sposób dodania po jednym rekordzie do każdej z tabel.

1.1. *Utworzenie bazy danych*

```
CREATE DATABASE `sklep`
```

1.2. *Utworzenie tabel*

1.2.1. Utworzenie tabeli „uczen”

```
CREATE TABLE `uczen` (  
  `ID_uczen` int (11) NOT NULL AUTO_INCREMENT,  
  `imie` varchar (20) NOT NULL,  
  `nazwisko` varchar (40) NOT NULL,  
  `pesel` bigint(40) NOT NULL,  
  `kod_pocztowy` varchar (6) NOT NULL,  
  `miejscowosc` varchar (20) NOT NULL,  
  `ulica` varchar (40) NOT NULL,  
  `nr_domu` varchar (5) NOT NULL,  
  `nr_mieszkania` varchar (5) ,  
  `data_urodzenia` datetime NOT NULL,  
  `tel_kontaktowy` bigint (20) NOT NULL,  
  `tel_opiekuna` bigint (20) NOT NULL,  
  `id_zajec` int (11),  
  PRIMARY KEY (`ID_uczen`)  
)
```

1.2.2. Utworzenie tabeli „nauczyciel”

```
CREATE TABLE `nauczyciel` (  
  `ID_nauczyciel` int (11) NOT NULL AUTO_INCREMENT,  
  `imie` varchar (20) NOT NULL,  
  `nazwisko` varchar (40) NOT NULL,  
  `pesel` bigint(40) NOT NULL,  
  `kod_pocztowy` varchar (6) NOT NULL,  
  `miejscowosc` varchar (20) NOT NULL,
```

```

`ulica` varchar (40) NOT NULL,
`nr_domu` varchar (5) NOT NULL,
`nr_mieszkania` varchar (5) ,
`data_urodzenia` datetime NOT NULL,
'tel_kontaktowy' bigint (20) NOT NULL,
'tel_opiekuna' bigint (20) NOT NULL,
'id_zajec' int (11),
PRIMARY KEY (`ID_nauczyciel`)
)

```

1.2.3. Utworzenie tabeli „zajecia”

```

CREATE TABLE `zajecia` (
`ID_zajec` int (11) NOT NULL AUTO_INCREMENT,
`nazwa` varchar (20) NOT NULL,
`dzien_tygodnia` varchar (15) NOT NULL,
`godzina` time NOT NULL,
`nr_sali` int (11),
PRIMARY KEY (`ID_zajec`)
)

```

1.2.4. Utworzenie tabeli „sala”

```

CREATE TABLE `sala` (
`nr_sali` int (11) NOT NULL AUTO_INCREMENT,
`pojemnosc` int (11) NOT NULL,
`przeznaczenie` varchar (500) NOT NULL,
PRIMARY KEY (`nr_sali`)
)

```

1.2.5. Utworzenie tabeli „nieobecnosci”

```

CREATE TABLE `nieobecnosci` (
`ID_nieobecnosci` int (11) NOT NULL AUTO_INCREMENT,
`ID_uczen` int (11) NOT NULL,
`ID_zajec` varchar (500) NOT NULL,

```

```
`data_nieobecnosci` date NOT NULL,  
PRIMARY KEY (`ID_nieobecnosci`))
```

1.2.6. Utworzenie tabeli „materialydotatkowe”

```
CREATE TABLE `materialydotatkowe` (  
`ID_materialu` int (11) NOT NULL AUTO_INCREMENT,  
`ID_zajec` int (11),  
`nazwa_materialu` varchar (100) NOT NULL,  
PRIMARY KEY (`ID_nieobecnosci`)  
)
```

1.2.7. Utworzenie tabeli „zapisanyzajecia”

```
CREATE TABLE `zapisanyzajecia` (  
`ID` int (11) NOT NULL AUTO_INCREMENT,  
`ID_zajec` int (11),  
`ID_uczen` int (11),  
PRIMARY KEY (`ID`)  
)
```

1.3. Dodanie przykładowych rekordów do każdej z tabel

1.3.1. Dodanie rekordu do tabeli „uczen”

```
INSERT INTO `mdk`.`uczen` (`ID_uczen`, `Imie`, `Nazwisko`, `Pesel`, `Kod_pocztowy`, `Mia  
sto`, `Ulica`, `Nr_domu`, `Nr_mieszkania`, `Data_urodzenia`, `Tel_kontaktowy`, `Tel_opieku  
na`, `ID_zajec`) VALUES (NULL, 'Damian', 'Damianowski', '93010112345', '98-  
200', 'Sieradz', 'Daszynskiego', '6', '12', '1993-01-  
01', '605123456', '507123456', '11');
```

1.3.2. Dodanie rekordu do tabeli „nauczyciel”

```
INSERT INTO `mdk`.`nauczyciel` (`ID_nauczyciel`, `Imie`, `Nazwisko`, `Pesel`, `Kod_poczt  
owy`, `Miejscowosc`, `Ulica`, `Numer_domu`, `Numer_mieszkania`, `Data_urodzenia`, `Tel_ko
```

```
ntaktowy`, `ID_zajec`) VALUES (NULL, 'Piotr', 'Piotrowski', '800505123456', '98-210', 'Biskupice', 'Sadowa', '4a', NULL, '1980-05-05', '509123456', '11');
```

1.3.3. Dodanie rekordu do tabeli „sala”

```
INSERT INTO `mdk`.`sala` (`Nr_sali`, `Pojemnosc`, `Przeznaczenie`) VALUES ('12', '15', 'Plastyka');
```

1.3.4. Dodanie rekordu do tabeli „zajecia”

```
INSERT INTO `mdk`.`zajecia` (`ID_zajec`, `Nazwa`, `Dzien_tygodnia`, `Godzina`, `Nr_sali`) VALUES (NULL, 'Plastyka', 'Wtorek', '16:30:00', '12');
```

1.3.5. Dodanie rekordu do tabeli „nieobecncosc”

```
INSERT INTO `mdk`.`nieobecncosc` (`ID_nieobecncosci`, `ID_uczen`, `ID_zajec`, `Data_nieobecncosci`) VALUES (NULL, '1', '11', '2015-01-02');
```

1.3.6. Dodanie rekordu do tabeli „materialydotatkowe”

```
INSERT INTO `mdk1`.`materialydotatkowe` (`ID_materialu`, `ID_zajec`, `Nazwa_materialu`) VALUES (NULL, '11', 'Przykładowe obrazy');
```

2. Zbiór zapytań zoptymalizowanych

Wykonano szereg zapytań do bazy danych. Zapytania odzwierciedlają operację wykonywane podczas korzystania z bazy danych. Przykładowe operacje i odpowiadające im zapytania:

- Dodanie nowego ucznia (przedstawione w podpunkcie 1.3.1)
- Usunięcie ucznia o indeksie 1:

```
DELETE FROM `mdk1`.`uczen` WHERE `uczen`.`ID_uczen` = 1
```

- Wyświetlenie numeru Sali, w której odbywają się zajęcia: plastyka

```
SELECT nazwa, nr_sali FROM zajecia WHERE nazwa = 'Plastyka'
```

Rezultat takiego zapytania jest następujący:

nazwa	nr_sali
Plastyka	12

- Wyświetlanie uczniów zapisanych na zajęcia o nazwie „Kolo historyczne”

```
Select u.ID_uczen, u.Imie, u.Nazwisko FROM uczen u join zapisanyzajecia z on z.ID_uczen = u.ID_uczen where z.ID_zajec = (Select id_zajec FROM zajecia where nazwa = 'Kolo historyczne')
```

Rezultat takiego zapytania jest następujący:

ID_uczen	Imie	Nazwisko
4	Karol	Damianski
5	Piotr	Piotrowski

Dla sprawdzenia wydajności bazy danych zmierzono czas wyszukiwania rekordu bazy danych w tabeli „uczen” przy różnej ilości dodanych uczniów:

Zapytanie:

```
SELECT * FROM `uczen` where imie = 'czas'
```

- 500 uczniów

✓ Pokazano wiersze 0 - 0 (1 ogółem, Wykonanie zapytania trwało 0.0010 sekund(y).)

```
.....  
SELECT * FROM `uczen` where imie = 'czas'
```

+ Opcje

ID_uczen	Imie	Nazwisko	Pesel	Kod_poc
920757	czas	Test	123	123




Edytuj Kopij Usuń

- 400 000 uczniów

✓ Pokazano wiersze 0 - 0 (1 ogółem, Wykonanie zapytania trwało 1.1571 sekund(y).)

```
SELECT * FROM `uczen` where imie = 'czas'
```

+ Opcje







	ID_uczen	Imie	Nazwisko	Pesel	Kod_pocz
<input type="checkbox"/>   	360796	czas	Test	123	123

- 800 000 uczniów

✓ Pokazano wiersze 0 - 1 (2 ogółem, Wykonanie zapytania trwało 1.6531 sekund(y).)

```
SELECT * FROM `uczen` where imie = 'czas'
```

+ Opcje

	ID_uczen	Imie	Nazwisko	Pesel	Kod_pocz
<input type="checkbox"/>   	360796	czas	Test	123	123
<input type="checkbox"/>   	530113	czas	Test	123	123

Na podstawie powyższego testu można stwierdzić, że nawet dla 800 000 uczniów wyszukanie jednego z nich trwa mniej niż 2 sekundy, co jest dobrym czasem, biorąc pod uwagę, że założenia co do rozmiaru maksymalnej liczby uczniów w projektowanym systemie wynosi: 500 osób.

3. Polityka bezpieczeństwa

Jest to pojęcie opisujące założenia w firmie dotyczące zabezpieczania i przechowywania informacji – w naszym przypadku bazy danych oraz w przyszłości aplikacji.

Politykę bezpieczeństwa placówki „Młodzieżowy Dom Kultury” można podzielić na dwie części:

1. Zabezpieczenia konieczne do ochrony bazy danych „Młodzieżowego Domu Kultury”:

- **Zasada minimalnych uprawnień** – w naszym systemie zastosowano system uprawnień, w którym każdy z użytkowników systemu (administrator, nauczyciele, uczniowie) powinien mieć dostęp tylko tam, gdzie jest to wymagane
- **Kopie zapasowe** – system bazodanowy powinien na bieżąco archiwizować ważne dane, np. dane uczniów, dane nauczycieli
- **Logowanie do systemu** – użytkownik loguje się do systemu (podając unikalny login oraz hasło) za pomocą aplikacji bazodanowej lub strony internetowej (w przyszłości). Każde konto założone w systemie posiada przydzielone uprawnienia, mówiące nam, co dany użytkownik może robić, a czego nie
- **Organizacja haseł** – hasła, które będą używane do logowania do systemu bazodanowego powinny być okresowo zmieniane, zaszyfrowane oraz trudne do odgadnięcia. Dla potrzeb naszej bazy danych przyjęto, że hasło musi składać się z minimum 10 znaków i zawierać co najmniej: jedną wielką literę, jedną cyfrę, jeden znak specjalny. Hasło powinno być różne od pięciu uprzednio używanych haseł
- **Zasada wielowarstwowych zabezpieczeń** – system bazodanowy „Młodzieżowego Domu Kultury” powinien być chroniony równolegle na wielu poziomach. Zapewnia to pełniejszą oraz skuteczniejszą ochronę danych osobowych. Przykładowo: w celu ochrony przed

wirusami i atakami hakerskimi stosuje się równolegle wiele technik: oprogramowanie antywirusowe, systemy oraz urządzenia pełniące rolę firewall'i, odpowiednią konfigurację systemu aktualizacji Windows

- **Zabezpieczenie stacji roboczych** – stacje robocze, z których korzystają nauczyciele oraz pracownicy MDK'u powinny zostać zabezpieczone przed nieautoryzowanym dostępem osób trzecich

2. Inne zabezpieczenia (niekoniecznie systemowe), które mogą zostać zaimplementowane do ochrony baz danych:

- **Weryfikacja przestrzegania polityki bezpieczeństwa** – zarząd sprawujący nadzór nad bazą danych okresowo wykonuje wewnętrzny lub zewnętrzny audyt bezpieczeństwa mający na celu wykrycie ewentualnych uchybień w realizacji założeń polityki bezpieczeństwa
- **Zasada ograniczania dostępu** – domyślnym uprawnieniem w systemach bazodanowych powinno być zabronienie dostępu. Dopiero w przypadku zaistnienia odpowiedniej potrzeby lub sytuacji np. awans pracownika, administrator IT przyznaje stosowne uprawnienia
- **Dostęp do danych poufnych** – jeśli stacja PC jest komputerem przenośnym to musi być ona dodatkowo zabezpieczona np. posiadać zaszyfrowany dysk twardy; dostęp do danych poufnych powinien odbywać się z wykorzystaniem kanału szyfrowanego np. VPN
- **Odpowiedzialność pracowników za dane poufne i dostępne** – każdy pracownik pracujący z bazą danych jest odpowiedzialny za utrzymanie w tajemnicy danych poufnych, do których dostęp został mu

powierzony, a także danych dostępowych (haseł, kluczy software'owych

- **Edukacja pracowników w zakresie bezpieczeństwa** – przedsiębiorstwo zarządzające systemem bazodanowym powinna dbać o edukację pracowników w zakresie bezpieczeństwa baz danych. Może to czynić organizując różnego rodzaju szkolenia dla pracowników
- **Transport danych poufnych** – zabronione jest przenoszenie niezabezpieczonych danych poufnych (np. pendrive, płyta CD) poza teren przedsiębiorstwa