

Zajęcia 0 - Zajęcia organizacyjne

- Zaproponowanie tematów do realizacji projektu.
 - W ramach każdego tematu studenci przeprowadzają rozmowy z prowadzącym celem wstępnego ustalenia jego zakresu.
 - Lista przykładowych tematów:
 - System zarządzania czasem pracy pracowników.
 - System obiegu dokumentów w przedsiębiorstwie.
 - System wsparcia pracy pogotowia ratunkowego.
 - System dokumentowania połączeń sieciowych w serwerowni.
 - System do ewidencjonowania środków trwałych w przedsiębiorstwie.
 - System wspierania organizacji konferencji.
 - System organizacji komunikacji miejskiej.
 - System wspierający zarządzanie cmentarzem.
 - System tworzenia ankiet użytkowników i gromadzenia odpowiedzi.
 - Symulator zdolności kredytowej oraz symulator planu kredytowego.
 - System organizujący terminy wykonywania przeglądów instalacji budowlanych.
 - System powiadomień o nadchodzących wydarzeniach kulturalnych.
 - System tworzenia narracyjnych gier RPG.
 - System wspierający planowanie posiłków.
 - System CMS umożliwiający tworzenie stron internetowych o dowolnej strukturze.
 - System tworzenia i składania wniosków o grant naukowy.
 - System wspierający organizację ochrony radiologicznej pracowników narażonych na promieniowanie jonizujące.
 - System wspomagający organizację transportu pacjentów pomiędzy szpitalami.
 - Baza danych dla działu kadrowo-płacowego.
 - Hipotetyczny system e-recepty.
 - System wspierający pracę zespołu programistów zgodny z metodyką SCRUM.
 - System wspierający ocenę narażenia pracowników na promieniowanie jonizujące.
 - System organizacji pracy salonu samochodowego.
 - System rozliczeń delegacji pracowników.
 - System zarządzania planem urlopowym pracowników.
 - Formularz PIT 11.
 - Stacja krwiodawstwa.
 - Więzienie.
 - Multikino.
 - Przychodnia lekarska.
 - System zarządzania miejscami w akademiku.
 - System zarządzania zgłoszeniami wspierający pracę helpdesków w przedsiębiorstwach.
-

PROJEKT RELACYJNEJ BAZY DANYCH

Zajęcia 1 - Faza koncepcyjna

1. Przedstawienie następujących informacji wynikających z analizy świata rzeczywistego:
 - **Streszczenie** - zarys wymagań projektu. Jakie są potrzeby informacyjne? Jakie czynności wyszukiwania (pytania) można wykonać za pomocą projektowanej bazy?
 - **Cele projektu** - cele projektu powinny być zapisane w formacie SMART¹, co oznacza, że muszą być konkretne, wymierne, osiągalne, realistyczne i określone w czasie.
 - **Zakres projektu** - co należy uwzględnić, a czego nie.
 - **Wymagania funkcjonalne**². Przedstawione wymagania mają uwzględnić mechanizm logowania użytkowników oraz pozwalać na przypisywanie im różnych poziomów dostępu (min. 3) np. admin, user, editor, guest etc.
2. Przygotowanie **diagramu obiektowo-związkowego**³. Na diagramie ma się znaleźć min. 15 różnych encji. Na tym etapie nie mogą występować encje asocjacyjne. Opracowany diagram ma uwzględniać przynajmniej jedną relację 1-N oraz przynajmniej jedną relację N-N.

Zajęcia 2 - Faza logiczna

1. **Definicja schematów relacji** na podstawie diagramu obiektowo-związkowego. Każdy schemat relacji powinien zawierać: atrybuty, zbiór funkcyjnych i wielowartościowych zależności między atrybutami. Schematy powinny odzwierciedlać elementy diagramu obiektowo-związkowego.
2. Normalizacja schematów do **III postaci normalnej**.
3. Wykonanie **diagramu relacji** za pomocą dowolnego pakietu wspomagającego projektowanie baz danych.

Zajęcia 3 - Faza fizyczna

1. Opracowanie **specyfikacji relacji** w formie skryptu **SQL DDL**. Omówienie przygotowanego skryptu i wykorzystanych instrukcji.
2. Wdrożenie projektu w wybranym **systemie zarządzania bazą danych** (np. PostgreSQL, MySQL)
3. Sformułowanie i implementacja **więzów integralności**. Omówienie typów **wywołań kaskadowych** i wskazanie ich w projekcie. Omówienie wykorzystania instrukcji **CHECK** w projekcie.

¹ <https://www.wrike.com/project-management-guide/faq/what-is-smart-in-project-management/>

² <https://qracorp.com/functional-vs-non-functional-requirements/>

³ <https://www.smartdraw.com/entity-relationship-diagram/>

Zajęcia 4 - Faza fizyczna

4. Napisać skrypt (w dowolnym języku programowania) wypełniający bazę danych losowymi danymi. W bazie ma się pojawić kilkanaście tysięcy rekordów w odpowiednio wybranych tabelach. W niektórych tabelach (np. zawierających listę ról użytkowników) rekordów może pojawić się odpowiednio mniej i mogą one zostać wprowadzone ręcznie. Za każdym uruchomieniem skryptu mają się generować różne dane. Skrypt ma obsługiwać powiązania pomiędzy encjami. Napisany skrypt może wykorzystywać bibliotekę typu ORM (np. SQLAlchemy, Hybernate, Ruby Object Mapper).

Zajęcia 5 - Faza fizyczna

1. **Projekt interfejsu systemu dostępowego do zaprojektowanej bazy.** Z wykorzystaniem dowolnego narzędzia do projektowania UX przygotować min.10 różnych, nietrywialnych widoków (formularz do podania loginu i hasła, bądź panele nie odzwierciedlające konkretnego zastosowania zaprojektowanej bazy danych są błędne).
2. Definicja **raportów i funkcji wyszukiwania**. Minimum 10 nietrywialnych i zróżnicowanych zapytań SQL, w tym zapytania o charakterze statystycznym. Zaprezentować działanie **funkcji agregujących**. UWAGA - zapytania mają być przemyślane i ekspresywne. Podmienianie warunków w klauzuli WHERE w tym samym zapytaniu lub obliczanie średniego numeru PESEL będzie skutkowało obniżeniem oceny końcowej.

Zajęcia 6 - Faza fizyczna

1. **Funkcja EXPLAIN.** Na przygotowanych zapytaniach omówić zawartość wyniku działania funkcji EXPLAIN i jej specyfiki wynikającej z wybranego systemu zarządzania bazą danych. **Jakie informacje są tam przedstawione? Jak je interpretować? Na czym polega optymalizacja zapytań?**
2. **Wybór i wprowadzanie indeksów⁴.** Na podstawie dokumentacji wybranego systemu zarządzania bazą danych do przygotowanych i wypełnionych tabel wprowadzić indeksy. **Jakie są wady i zalety poszczególnych typów indeksów i w jakich sytuacjach najlepiej ich używać?**
3. **Weryfikacja** działania zapytań z poprzednich zajęć w świetle wyników funkcji EXPLAIN przed i po wprowadzeniu indeksów. **Czy można inaczej zdefiniować zapytania?**

Zajęcia 7 - Faza fizyczna

1. Przygotowanie sprawozdania powykonawczego - min. 2 strony A4 konstruktywnych wniosków dotyczących realizacji projektu. Dokument (np. oparty o format **SWOT**) ma przedstawić krytyczną analizę przyjętych w początkowej fazie założeń i ograniczeń. Ponadto mają zostać przedstawionych kilka

⁴ <https://www.dbta.com/Columns/DBA-Corner/Top-10-Steps-to-Building-Useful-Database-Indexes-100498.aspx>
<https://www.essentialsql.com/what-is-a-database-index/>
<https://www.postgresql.org/docs/9.1/indexes.html>

kierunków rozwoju opracowanej bazy. UWAGA - wnioski takie jak „trzeba pracować systematycznie” lub „realizacja projektu była łatwa” są błędne.

2. Omówienie modyfikacji wprowadzanej do projektu. Modyfikacja ta ma wynikać np. ze zmienionych założeń lub nowych wymagań.

Zajęcia 8 - Faza logiczna

1. Aktualizacja diagramu obiektowo-związkowego do zmienionych wymagań.
2. Aktualizacja specyfikacji relacji. Weryfikacji normalizacji schematu.
3. Weryfikacja i aktualizacja więzów integralności.

Zajęcia 9 - Faza logiczna

1. Opracowanie skryptu **SQL DDL** wykorzystującego funkcję **ALTER** do przetworzenia struktury bazy danych przed modyfikacją do zgodnej z nowymi wymaganiami.
2. Aktualizacja raportów i funkcji wyszukiwania. Weryfikacja poprawności i spójności danych.

PROJEKT NIERELACYJNEJ BAZY DANYCH

Zajęcia 10 - Faza wstępna

1. Wybór i uzasadnienie technologii NoSQL do zrealizowania projektu bazy danych na ten sam temat co w przypadku relacyjnej bazy danych. Jakie są zalety wybranej bazy NoSQL? Jakie są wady wybranej bazy NoSQL?
2. Weryfikacja przyjętych założeń i ograniczeń.

Zajęcia 11 - Faza koncepcyjna i fizyczna

1. Definicja i wdrożenie struktur przechowywania danych w wybranej technologii nierelacyjnej. W jaki sposób definiowane są możliwe związki pomiędzy przechowywanymi danymi? Jakie wykorzystane mechanizmy zapewnienia spójności (jeśli istnieją)? Czym różni się paradygmat **BASE** od paradygmatu **ACID**?
2. Prezentacja przykładowych zapytań

Zajęcia 12 - Faza fizyczna

1. Napisać skrypt (w dowolnym języku programowania), analogiczny do skryptu z zajęć 4, wypełniający nierelacyjną bazę danych losowymi wartościami. W bazie ma się pojawić kilkanaście tysięcy wpisów w odpowiednio wybranych strukturach. Za każdym uruchomieniem skryptu mają się generować różne dane.
-

Zajęcia 13 - Faza fizyczna

1. Definicja raportów i funkcji wyszukiwania. Minimum 10 nietrywialnych i zróżnicowanych zapytań, w tym zapytania o charakterze statystycznym. Zaprezentować działanie funkcji agregujących (jeśli istnieją).
2. Analiza porównawcza z projektem relacyjnej bazy danych. Jakie są główne różnice w projektach zrealizowanych za pomocą bazy relacyjnej i nierelacyjnej? Czy ich zastosowania są tożsame i można je traktować wymiennie? W jakich zastosowaniach lepiej sprawdzają się bazy relacyjne, a w jakich wybrana technologia nierelacyjna? Czy pojawiła się konieczność zmiany założeń lub wybrane wymagania były niemożliwe do zrealizowania? W jaki sposób wybrana baza NoSQL różni się od bazy relacyjnej pod względem definiowania zapytań? Jakie są główne różnice w wydajności i jakie są ich przyczyny?