

Założenia projektu zaliczeniowego

(dotyczy grup zajęciowych dra Bartosza Bębla)

Przedmiotem projektu zaliczeniowego drugiej części zajęć laboratoryjnych z przedmiotu Systemy Baz Danych II będzie wykonanie prostej aplikacji, umożliwiającej definiowanie, przeglądanie i edytowanie danych, przechowywanych w bazie danych. Przed zaimplementowaniem aplikacji konieczne będzie zaprojektowanie i zaimplementowanie schematu relacyjnego, z którym będzie współpracowała aplikacja.

Projekt ma być realizowany w grupach 2-osobowych. W przypadku grupy jednoosobowej wymagania (np. liczba relacji w schemacie) zostają zmniejszone o (mniej więcej) połowę. W sytuacji grupy większej niż 2-osobowa wymagania ulegają zwiększeniu.

Tematykę (modelowaną rzeczywistość) projektu grupa wymyśla samodzielnie. Ma być ona na tyle pojemna, aby spełnić wymagania, zawarte w punktach dot. wymagań dotyczących schematu relacyjnego.

Dodatkowo: w modelowanej rzeczywistości określić operację, której jednoczesne (współbieżne) wykonanie przez kilka równoległe działających instancji aplikacji bez zastosowania dodatkowych mechanizmów zarządzania współbieżnością może prowadzić do niespójności (np. do wystąpienia anomalii) w danych. Następnie taka operacja ma zostać w aplikacji zaimplementowana w taki sposób, aby uniemożliwić wystąpienie ew. niespójności (np. przez zastosowanie odpowiedniego poziomu izolacji, zastosowanie niestandardowych blokad, itd.).

Wymagania dotyczące schematu relacyjnego

1. Rzeczywistość, wspierana przez aplikację, ma zostać zamodelowana przy zastosowaniu diagramu związków encji. Diagram ma składać się z min. 10 encji, połączonych związkami.
2. Diagram związków encji ma zostać przekształcony w schemat relacyjny. Oprócz relacji, implementujących encje, schemat ma wykorzystywać:
 - sekwencje w sytuacji, gdy kolumny, będące kluczami podstawowymi relacji, będą liczbami (lub alternatywne metody automatycznego generowania wartości dla sztucznych kluczy liczbowych),
 - indeksy wspomagające wyszukiwanie danych w relacjach oraz wykonywanie innych operacji (np. sortowania, grupowania, itd.),
 - min. dwa podprogramy składowane w bazie danych: jedna funkcja i jedna procedura, wykonujące wskazane operacje (np. wyliczanie wartości złożonych wyrażeń); podprogramy mogą zostać połączone w jeden pakiet; podprogramy mają być wywoływane z aplikacji.
3. Brak ograniczeń co do systemu zarządzania bazą danych, w której będą przechowywane obiekty bazodanowe (dopuszczalne są dowolne produkty

bazodanowe pod warunkiem, że będzie w nich możliwe zdefiniowanie obiektów wymienionych w wymaganiach, np. podprogramów składowanych).

4. Zadania projektowe dotyczące schematu relacyjnego są następujące:
 - zaprezentowanie krótkiego opisu funkcjonalności aplikacji i uzyskanie aprobaty prowadzącego: **do 4 listopada 2016 r.**,
 - zaprojektowanie diagramu związków encji, przedstawienie projektu prowadzącemu: **do 2 grudnia 2016 r.**,
 - po akceptacji diagramu związków encji przez prowadzącego prezentacja gotowego schematu relacyjnego wraz z obiektami dodatkowymi (sekwencjami, indeksami, podprogramami składowanymi): **do 16 grudnia 2016 r.**

Wymagania dotyczące aplikacji

1. Aplikacja ma składać się z kilku ekranów (formularzy ekranowych), za pomocą których użytkownik będzie mógł realizować operacje na danych wszystkich relacji ze schematu relacyjnego, na którym działa aplikacja.
2. W formularzu ekranowym aplikacji mają zostać zaimplementowane podstawowe operacje, dostępne dla użytkownika aplikacji:
 - wyświetlanie i przeglądanie danych,
 - wyszukiwanie danych – możliwość wprowadzania wzorca poszukiwanej informacji,
 - dodawanie danych,
 - modyfikacja danych,
 - usuwanie danych.
3. Formularz ekranowy ma wspomagać użytkownika w procesie definiowania danych podpowiadając użytkownikowi wartości dostępne dla określonego pola. Np. przy edycji danych pracownika (tabela PRACOWNICY) dostarczenie listy zespołów (z tabeli ZESPOLY), do których pracownik może zostać przypisany, listy etatów, na których pracownik może zostać zatrudniony, itd.
4. Formularz ekranowy ma kontrolować poprawność edytowanych przez użytkownika danych z punktu widzenia ograniczeń integralnościowych, zdefiniowanych w bazie danych, takich jak:
 - kontrola obecności określonych danych (np. wymuszenie podania nazwiska przy definiowaniu danych nowego pracownika),
 - kontrola wartości wpisywanych w polach (np. sprawdzanie zgodności wpisanej do pola wartości z wartościami dopuszczalnymi dla kolumny relacji, z którą pole jest skojarzone),
 - kontrola unikalności rekordów (klucz podstawowy i klucz unikalny),
 - kontrola usuwania rekordów (klucz obcy),
 - itd.

5. Przy definiowaniu nowych danych formularz ma korzystać z mechanizmów systemu bazy danych, wspomagających generowanie wartości dla kolumn numerycznych będących kluczami podstawowymi (czyli np. z sekwencji w SZDB Oracle).
6. Utworzone w bazie danych podprogramy składowane mają być wywoływane z chociaż jednego formularza ekranowego aplikacji (w przypadku funkcji jej wynik ma zostać zaprezentowany w formularzu).
7. Nie ma żadnych ograniczeń jeśli chodzi o wykorzystywane przy budowaniu aplikacji technologie (może to być aplikacja webowa, mobilna, desktopowa, ...).
8. Zadania projektowe dotyczące aplikacji są następujące:
 - zaprezentowanie krótkiego opisu funkcjonalności aplikacji i uzyskanie aprobaty prowadzącego: **do 4 listopada 2016 r.**,
 - prezentacja opisu ekranów aplikacji: **do 16 grudnia 2016 r.**,
 - prezentacja gotowych aplikacji i uzyskanie oceny projektu: **do 31 stycznia 2017 r.**

Harmonogram prac

do 4 listopada 2016 r.,

Zaprezentowanie krótkiego opisu funkcjonalności aplikacji i uzyskanie aprobaty prowadzącego.

do 2 grudnia 2016 r.

Zaprojektowanie diagramu związków encji, przedstawienie projektu diagramu prowadzącemu.

do 16 grudnia 2016 r.

Prezentacja gotowego schematu relacyjnego wraz z obiektami dodatkowymi (sekwencjami, indeksami, podprogramami składowanymi, itd.).

Prezentacja opisu ekranów aplikacji.

do 31 stycznia 2017 r.

Prezentacja gotowych aplikacji i uzyskanie oceny projektu.

Uwaga!

1. Opóźnienie w stosunku do założonego harmonogramu skutkuje przypisaniem 1 punktu karnego za każdy rozpoczęty tydzień spóźnienia.
2. Każdy błąd, który pojawi się przy prezentacji gotowej aplikacji lub odstępstwo od którejkolwiek z ww. zasad powoduje przypisanie 1 punktu karnego (przy czym wielokrotne występowanie w aplikacji tego samego błędu skutkuje przypisaniem liczby punktów równej liczbie wystąpień tego błędu).
3. Każde dwa punkty karne przekładają się na obniżenie oceny z projektu o 0,5 stopnia.