

Sprawozdanie z Projektu: Biuro podróży

1. Opis Projektu

Projekt przedstawia implementację obiektowej bazy danych dla biura podróży. System obiektowej bazy danych dla biura podróży, umożliwiający kompleksowe zarządzanie ofertami wakacyjnymi, rezerwacjami, oraz obsługą klientów.

1.1. Założenia Ogólne

1. Cel biznesowy systemu:

- Umożliwienie użytkownikom przeglądania i rezerwacji ofert wakacyjnych
- Zarządzanie bazą hoteli i ofert przez administratorów
- Zbieranie opinii i ocen od klientów
- Obsługa promocji i rabatów
- Zarządzanie rezerwacjami i płatnościami

2. Główne funkcjonalności:

- System rezerwacji z kontrolą dostępności terminów
- System ocen i recenzji hoteli
- Zarządzanie promocjami i cenami dynamicznymi
- Kategoryzacja ofert (rodzinne, dla dorosłych, all-inclusive)
- Obsługa wielu lokalizacji i regionów

3. Założenia dotyczące użytkowników:

- Różne poziomy dostępu (administrator, klient, użytkownik regularny)
- Możliwość oceniania tylko po zakończonym pobycie
- Weryfikacja konfliktów w rezerwacjach
- Historia rezerwacji i ocen dla każdego użytkownika

2. Realizacja Założeń Projektowych

2.1. Typy Obiektowe

W projekcie zaimplementowano następujące typy obiektowe:

- `Adresy_typ` - reprezentacja adresu
- `Ocena_typ` - struktura oceny
- `Hotel_typ` - informacje o hotelu
- `Uzytkownik_typ` - dane użytkownika
- `OcenaHoteli_typ` - powiązanie oceny z hotelem i użytkownikiem

- `Atrakcja_typ` - opis atrakcji
- `ListaAtrakcje_typ` - kolekcja atrakcji
- `Kategorie_typ` - kategorie ofert
- `OfertyWakacyjne_typ` - szczegóły oferty wakacyjnej
- `Promotions_typ` - informacje o promocjach
- `Rezerwacja_typ` - dane rezerwacji

2.2. Relacje Obiektowe

Zaimplementowano następujące powiązania między obiektami:

- Referencje między ofertami a hotelami (REF Hotel_typ)
- Referencje między rezerwacjami a użytkownikami (REF Uzytkownik_typ)
- Referencje między ocenami a hotelami (REF Hotel_typ)
- Kolekcje atrakcji w ofertach (NESTED TABLE ListaAtrakcje_typ)

2.3. Logika Biznesowa

2.3.1. Pakiety

Zaimplementowano następujące pakiety:

- `pkg_travel_search` - wyszukiwanie ofert
- `pkg_travel_display` - wyświetlanie wyników wyszukiwania
- `pkg_reservation_management` - zarządzanie rezerwacjami

2.3.2. Wyzwalacze

- `trg_check_ocena_hoteli` - weryfikacja uprawnień do oceny hotelu
- `trg_update_price_after_promotion` - aktualizacja cen po dodaniu promocji
- `trg_restore_price_after_promotion_delete` - przywracanie cen po usunięciu promocji
- `trg_set_original_price` - zapisywanie oryginalnej ceny
- `trg_check_reservation_conflict` - sprawdzanie konfliktów rezerwacji

2.4. Widoki

Utworzono widoki ułatwiające dostęp do danych:

- `v_oferty_details` - szczegóły ofert
- `v_hotel_sales_stats` - statystyki sprzedaży hoteli
- `v_user_activity` - aktywność użytkowników
- `v_reservation_details` - szczegóły rezerwacji

- `v_available_offers` - dostępne oferty

3. Role Użytkowników

3.1. Regularny_Uzytkownik

Uprawnienia:

- Przeglądanie ofert
- Dodawanie rezerwacji
- Dodawanie ocen hoteli
- Przeglądanie hoteli i kategorii

3.2. Klient

Uprawnienia:

- Modyfikacja ofert wakacyjnych
- Zarządzanie promocjami
- Przeglądanie danych

3.3. Administrator

Uprawnienia:

- Pełny dostęp do wszystkich tabel
- Zarządzanie użytkownikami
- Tworzenie i modyfikacja struktur

4. Ograniczenia i Walidacje

4.0. Główne Założenia Walidacyjne

1. Integralność danych:

- Każdy hotel musi mieć przypisaną lokalizację
- Każda oferta musi mieć określony termin i cenę
- Wszystkie referencje między obiektami muszą być poprawne
- System automatycznie zarządza cenami przy promocjach

2. Zasady biznesowe:

- Jedna osoba nie może mieć dwóch nakładających się rezerwacji
- Ocenę hoteli tylko przez zweryfikowanych gości
- Promocje nie mogą obniżyć ceny poniżej kosztów własnych
- Automatyczna aktualizacja dostępności ofert

- Automatyczna aktualizacja dostępności miejsc po anulacji rezerwacji
 - Nie można utworzyć rezerwacji bez przypisanego klienta
3. **Zarządzanie czasem:**
- Data zakończenia pobytu musi być późniejsza niż data rozpoczęcia
 - Promocje mają określony czas trwania
 - System śledzi terminy rezerwacji i automatycznie aktualizuje ich status
 - Historia rezerwacji jest przechowywana dla celów analitycznych
 - Wycieczka nie może mieć daty zakończenia wcześniejszej niż data rozpoczęcia
 -

4.1. Ograniczenia Biznesowe

- Ocena hotelu możliwa tylko po zakończonej rezerwacji
- Brak możliwości nakładających się rezerwacji dla jednego użytkownika
- Walidacja dat rozpoczęcia i zakończenia oferty
- Kontrola wartości ocen (1-5)
- Kontrola poprawności cen (większe od 0)

5. Szczegółowa Realizacja Założeń Funkcjonalnych

5.1. Zarządzanie Ofertami

1. **Kategoryzacja ofert:**
- Podział na oferty rodzinne i dla dorosłych
 - Oznaczenie ofert all-inclusive
 - Możliwość oznaczania ofert jako "last minute"
 - Elastyczny system kategoryzacji
2. **Zarządzanie cenami:**
- Automatyczna obsługa promocji
 - Przechowywanie historii cen
 - Możliwość definiowania różnych typów rabatów
 - Kontrola minimalnych cen
3. **Atrakcje i dodatki:**
- Możliwość dodawania listy atrakcji do oferty
 - Elastyczne zarządzanie dodatkowymi usługami
 - Grupowanie atrakcji według kategorii
 - Możliwość modyfikacji listy atrakcji

5.2. System Rezerwacji

1. **Proces rezerwacji:**
- Weryfikacja dostępności terminów

- Kontrola konfliktów rezerwacji
 - Automatyczne przeliczanie cen z uwzględnieniem promocji
 - Status rezerwacji i jego aktualizacje
2. **Obsługa użytkowników:**
- Przechowywanie historii rezerwacji
 - System ocen i komentarzy
 - Profile użytkowników z preferencjami
 - Różne poziomy uprawnień

5.3. Raportowanie i Analityka

1. **Widoki analityczne:**
- Statystyki sprzedaży
 - Analiza popularności ofert
 - Raporty rezerwacji
 - Zestawienia ocen hoteli

6. Podsumowanie Zrealizowanych Założeń

1. Implementacja typów obiektowych ✓
2. Utworzenie tabel obiektowych ✓
3. Zastosowanie referencji między obiektami ✓
4. Implementacja kolekcji ✓
5. Implementacja logiki biznesowej w pakietach ✓
6. Utworzenie wyzwalaczy dla automatyzacji procesów ✓
7. Implementacja widoków dla ułatwienia dostępu do danych ✓
8. Zdefiniowanie ról i uprawnień użytkowników ✓
9. Implementacja walidacji i ograniczeń ✓
10. Obsługa błędów i wyjątków ✓