

System obsługi hotelu

Etap 0: Opis słowny projektu

Zadaniem systemu jest wspomaganie zarządzania rezerwacjami w hotelu. Główną funkcjonalnością jaką zapewnia projekt jest wspomaganie procesu składania rezerwacji w hotelu, obsługi klientów i automatyzacja procesu naliczania kosztów pobytu w hotelu.

Etap 1: Projekt bazy danych

Klient:

nazwa atrybutu,	typ danych / dziedzina	ograniczenia np. unikalny identyfikator, obligatoryjność	opis - co oznacza dany atrybut (jeśli nazwa atrybutu jasno o tym nie mówi)
ID_klienta	Numer	Obligatoryjny, unikalny identyfikator	-
Numer_dokumentu	Ciąg znaków	Obligatoryjny	Nr pesel ew. inny numer identyfikujący osobę np. nr paszportu.
Imię	Ciąg znaków	Obligatoryjny	-
Nazwisko	Ciąg znaków	Obligatoryjny	-
Numer_telefonu	Numer	Obligatoryjny	-
Kraj	Ciąg znaków	Opcjonalny	-
Miejscowość	Ciąg znaków	Opcjonalny	-
Kod_pocztowy	Ciąg znaków	Opcjonalny	-
Ulica	Ciąg znaków	Opcjonalny	-
Numer_ulicy	Ciąg znaków	Opcjonalny	-
Numer_mieszkania	Numer	Opcjonalny	-

Rezerwacja:

nazwa atrybutu,	typ danych / dziedzina	ograniczenia np. unikalny identyfikator, obligatoryjność	opis - co oznacza dany atrybut (jeśli nazwa atrybutu jasno o tym nie mówi)
ID_rezerwacji	Numer	Obligatoryjny, unikalny identyfikator	-
Data_założenia	Data	Obligatoryjny	-
Wartość	Numer	Obligatoryjny	Całkowity koszt rezerwacji.

Pozycja_rezerwacji:

nazwa atrybutu,	typ danych / dziedzina	ograniczenia np. unikalny identyfikator, obligatoryjność	opis - co oznacza dany atrybut (jeśli nazwa atrybutu jasno o tym nie mówi)
Od	Data	Obligatoryjny	Data rozpoczęcia pobytu.
Do	Data	Obligatoryjny	Data zakończenia pobytu.

Pokój:

nazwa atrybutu,	typ danych / dziedzina	ograniczenia np. unikalny identyfikator, obligatoryjność	opis - co oznacza dany atrybut (jeśli nazwa atrybutu jasno o tym nie mówi)
Numer_pokoju	Numer	Obligatoryjny, unikalny identyfikator	-
Numer_piętra	Numer	Obligatoryjny	-
Numer_telefonu	Numer	Obligatoryjny	-

Typ_pokoju:

nazwa atrybutu,	typ danych / dziedzina	ograniczenia np. unikalny identyfikator, obligatoryjność	opis - co oznacza dany atrybut (jeśli nazwa atrybutu jasno o tym nie mówi)
ID_typu_pokoju	Numer	Obligatoryjny, unikalny identyfikator	-
Cena	Numer	Obligatoryjny	Cena za jedną dobę hotelową.
Liczba_miejsc	Numer	Obligatoryjny	-
Liczba_łóżek	Numer	Obligatoryjny	-
Opis	Ciąg znaków	Opcjonalny	Opis pokoju.

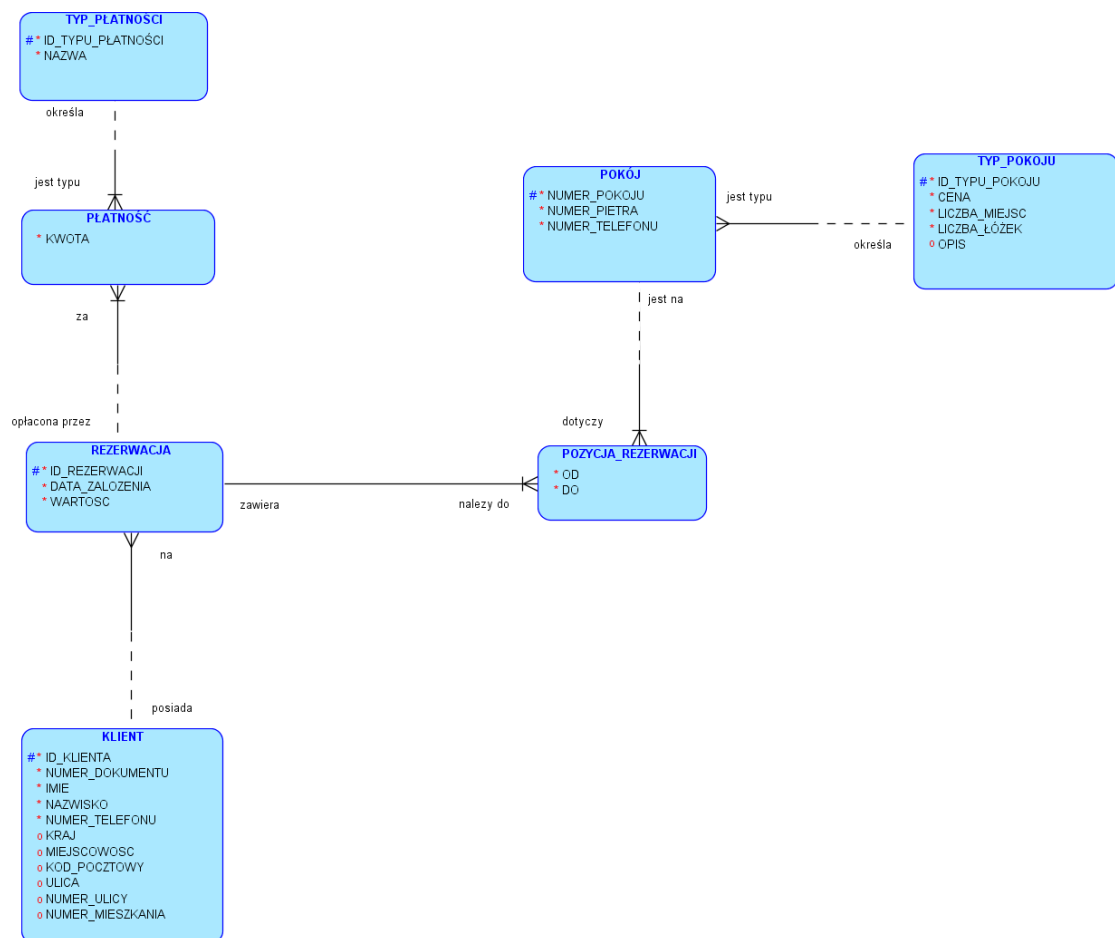
Płatność:

nazwa atrybutu,	typ danych / dziedzina	ograniczenia np. unikalny identyfikator, obligatoryjność	opis - co oznacza dany atrybut (jeśli nazwa atrybutu jasno o tym nie mówi)
Kwota	Numer	Obligatoryjny	-

Typ_płatności:

nazwa atrybutu,	typ danych / dziedzina	ograniczenia np. unikalny identyfikator, obligatoryjność	opis - co oznacza dany atrybut (jeśli nazwa atrybutu jasno o tym nie mówi)
ID_typu_płatności	Numer	Obligatoryjny, unikalny identyfikator	-
Nazwa	Ciąg znaków	Obligatoryjny	Nazwa typu płatności np. gotówka, przelew.

Diagram ER



Wymagania funkcjonalne i нефункционалне:

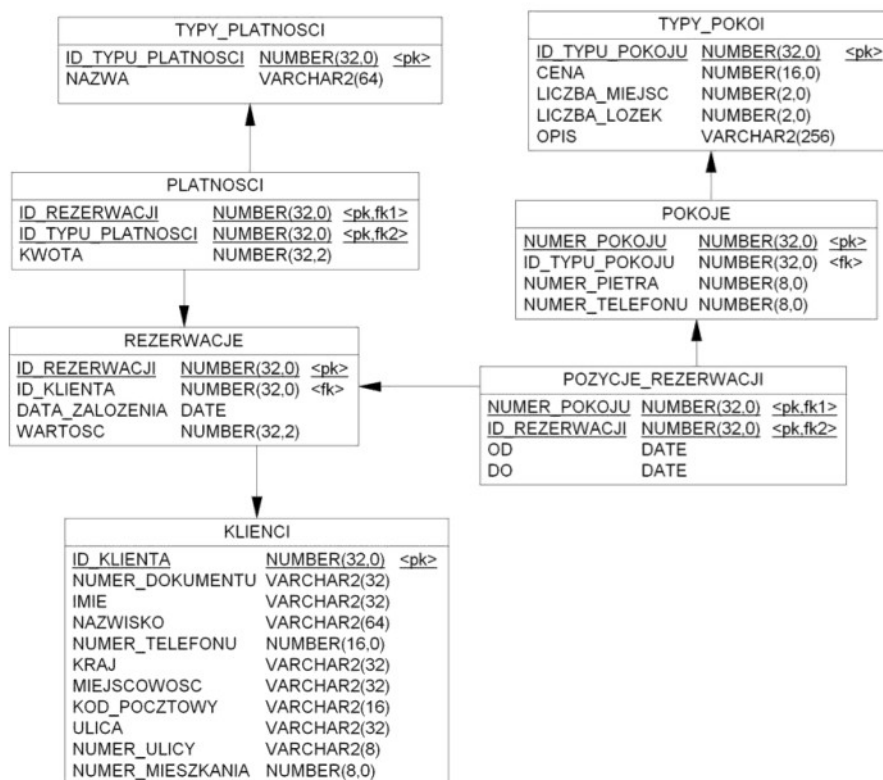
Wymagania funkcjonalne:

- Pracownik hotelu ma możliwość dodania nowej rezerwacji.
- Pracownik hotelu ma możliwość sprawdzenia stanu pokoi.
- Aktualizacja danych klienta (np. numer telefonu).

Wymagania нефункционалне:

- Należy przechowywać dane byłych klientów.
- System powinien być przenośny: możliwość uruchomienia na systemach Windows XP, Vista, 7 oraz Mac OS X.
- System powinien korzystać z bazy danych Oracle 11g.
- System musi umożliwiać przechowywanie danych 1000 pokoi.
- System powinien umożliwić przechowywanie informacji o 5000 klientach.
- System powinien umożliwiać przechowywanie informacji o 500000 rezerwacjach.

Diagram logiczny



Etap2: Projekt Aplikacji

Szkielet aplikacji:

Aplikacja jest podzielona na dwa pakiety: controller i view. Dwa oddzielne moduły realizują przeglądanie istniejących rezerwacji oraz dokonywanie rezerwacji.

Określenie sposobu realizacji wymuszanie integralności danych:

W definicji tabel używamy constraint'ów, które znajdują się w skrypcie DDL, natomiast programowe wymuszenie integralności danych polega na zmuszeniu użytkownika do wypełnienia wszystkich niezbędnych pól formularza.

Propozycje indeksów w bazie danych.

 -- DDL for Index PK_TYPU_POKOI

```

CREATE UNIQUE INDEX "BD2A14"."PK_TYPU_POKOI" ON
  "BD2A14"."TYPY_POKOI" ("ID_TYPU_POKOJU")
PCTFREE 10 INITRANS 2 MAXTRANS 255 NOLOGGING COMPUTE STATISTICS
STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS
  2147483645
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
TABLESPACE "BD2_STUD" ;
  
```

 -- DDL for Index PK_POZYCJE_REZERWACJI

```

CREATE UNIQUE INDEX "BD2A14"."PK_POZYCJE_REZERWACJI" ON
  "BD2A14"."POZYCJE_REZERWACJI" ("NUMER_POKOJU", "ID_REZERWACJI")
PCTFREE 10 INITRANS 2 MAXTRANS 255 NOLOGGING COMPUTE STATISTICS
STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS
  2147483645
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
TABLESPACE "BD2_STUD" ;

```

Projekt GUI dla wybranego modułu

Etap3: Implementacja

1) Kod po stronie serwera:

```

create or replace
TRIGGER "BASIC_CUSTOMER_INSERT" INSTEAD OF INSERT ON
  BASIC_CUSTOMERS
REFERENCING NEW AS new
FOR EACH ROW
DECLARE
  rowcnt number;

```

```

BEGIN
  SELECT COUNT(*) INTO rowcnt FROM KLIENCI WHERE NUMER_DOKUMENTU
    = :new.NUMER_DOKUMENTU;
  IF rowcnt = 0 THEN
    INSERT INTO KLIENCI VALUES(NULL, :new.NUMER_DOKUMENTU,
      :new.IMIE, :new.NAZWISKO, :new.NUMER_TELEFONU, NULL, NULL, NULL,
      NULL, NULL, NULL);
  ELSE
    SELECT COUNT(*) INTO rowcnt FROM KLIENCI WHERE NUMER_DOKUMENTU
      = :new.NUMER_DOKUMENTU and IMIE = :new.IMIE and NAZWISKO =
      :new.NAZWISKO;
    IF rowcnt = 0 THEN
      raise_application_error (-20999,'Customer credentials does not match. Document
      might be stolen.');
```

```

    ELSE
      UPDATE KLIENCI SET NUMER_TELEFONU =
      :new.NUMER_TELEFONU
      WHERE NUMER_DOKUMENTU = :new.NUMER_DOKUMENTU;
    END IF;
  END IF;
END IF;

END;
```

Trigger aktywuje się w momencie operacji insert na widoku BASIC_CUSTOMERS. Trigger sprawdza czy istnieje klient o podanym numerze dokumentu. Jeśli nie istnieje to dodaje nowego klienta do tabeli KLIENCI. Jeśli istnieje to sprawdza czy dane są poprawne a dokładnie czy zgadza się imię oraz nazwisko. Jeśli tak to zakładamy, że istnieje możliwość podania innego numeru telefonu niż, gdy klient po raz pierwszy odwiedził nasz hotel dlatego też, robimy operację update na kliencie zmieniając mu numer telefonu na nowy. Operacja wykonuje się nawet gdy numery telefonów się zgadzają. Jest to zrobione w ten sposób aby nie dodawać kolejnego zapytania SELECT do triggera. Warto również dodać, że trigger w przypadku gdy robimy INSERT powoduje wyzwolenie kolejnego triggera, który ustala sztuczny ID dla nowego klienta.

```

-----
-- DDL for Trigger RESERVATION_TRIGGER_VALUE
-----
```

```

  CREATE OR REPLACE TRIGGER "BD2A14"."RESERVATION_TRIGGER_VALUE"
  AFTER INSERT ON POZYCJE_REZERWACJI
  REFERENCING NEW AS new
  FOR EACH ROW
  DECLARE
    price number;
  BEGIN
    UPDATE REZERWACJE SET WARTOSC = WARTOSC + (SELECT CENA FROM
    POKOJE P, TYPY_POKOI T WHERE P.NUMER_POKOJU = :new.NUMER_POKOJU
    AND P.ID_TYPU_POKOJU = T.ID_TYPU_POKOJU)
    WHERE ID_REZERWACJI = :new.ID_REZERWACJI;
  END;
```

Ustala koszt rezerwacji.

```
ALTER PROCEDURE "BD2A14"."POKOJE_ILOSC_MIEJSC"(IN liczba_miejsc
    NUMBER(2,0))
RESULT (numer_pokoju NUMBER(32,0))
BEGIN

SELECT POKOJE.NUMER_POKOJU
FROM POKOJE, TYPY_POKOI
WHERE POKOJE.ID_TYPU_POKOJU = TYPY_POKOI.ID_TYPU_POKOJU
    AND TYPY_POKOI.LICZBA_MIEJSC = LICZBA_MIEJSC

END;
```

Procedura zwracająca numery pokoi, w których liczba miejsc jest równa zadanej.

```
ALTER FUNCTION "BD2A14"."KLIENCI_LICZBA_REZERWACJI"(IN id_klienta
    NUMBER(32,0))
RETURNS INTEGER
BEGIN

DECLARE "lr" INTEGER;

SET lr=(SELECT count(*)
FROM REZERWACJE
WHERE REZERWACJE.ID_KLIENTA=id_klienta);
RETURN lr;

END;
```

Funkcja zwracająca liczbę rezerwacji klienta o podanym ID.

WIDOK BASIC_CUSTOMERS

```
CREATE OR REPLACE FORCE VIEW "BD2A14"."BASIC_CUSTOMERS"
("NUMER_DOKUMENTU", "IMIE", "NAZWISKO", "NUMER_TELEFONU") AS
SELECT NUMER_DOKUMENTU, IMIE, NAZWISKO, NUMER_TELEFONU
FROM KLIENCI;
```

Jest to widok, który zawiera tylko podstawowe i najbardziej potrzebne informacje o klientach hotelu. Informacje to numer_dokumentu, imię, nazwisko oraz numer telefonu.

2) Kod po stronie klienta:

1. Zawiera transakcje dokonywania rezerwacji. Sprawdza poprawność danych klienta oraz czy jest możliwość wykonania rezerwacji. Jeśli warunki nie są spełnione transakcja zostaje anulowana.

2. Wyświetla rezerwacje w podanym okresie.