

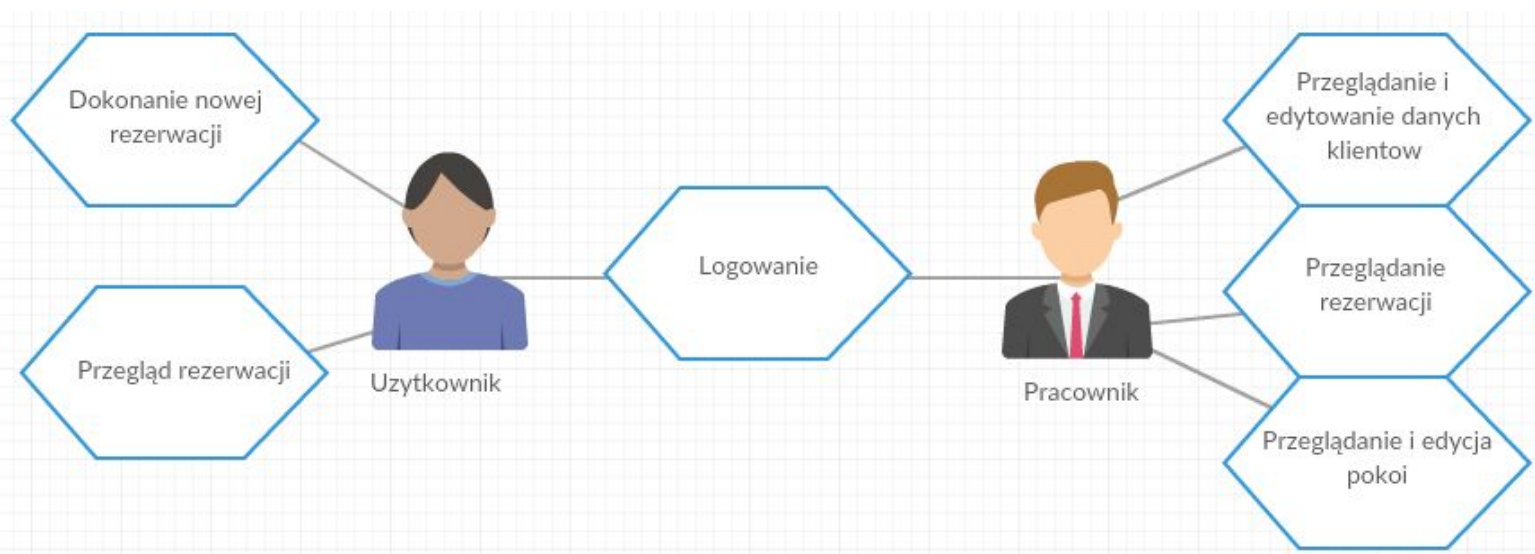
Projekt aplikacji bazodanowej przy użyciu WebForms oraz MVC

Temat 1: Rezerwacja pokoi w hotelu

1. Opis zadania

Program jest przeznaczony dla małego hotelu wypoczynkowego który ma 10-12 różnego typu(1os,2os ...). Za pomocą tego programu użytkownik może zarezerwować przez internet pokój odpowiadający jego wymogom. Każdy pokój jest w bazie danych. Rezerwacja polega na ustaleniu terminu wynajmowania pokoju przez klienta i nie może być on dostępny dla innych użytkowników. Każdy pokój jest w bazie danych. Kiedy pokój jest wolny trzeba dokonać rezerwacji wpisując dane użytkownika do bazy. Opłata rezerwacji nie dotyczy funkcji aplikacji. Baza musi zawierać dane mieszkań hotelu, datę rezerwacji tych mieszkań przez klientów, danych samych klientów (imię, nazwisko,telefon, e mail).

2. Kategorie użytkowników. Diagram przypadków użycia

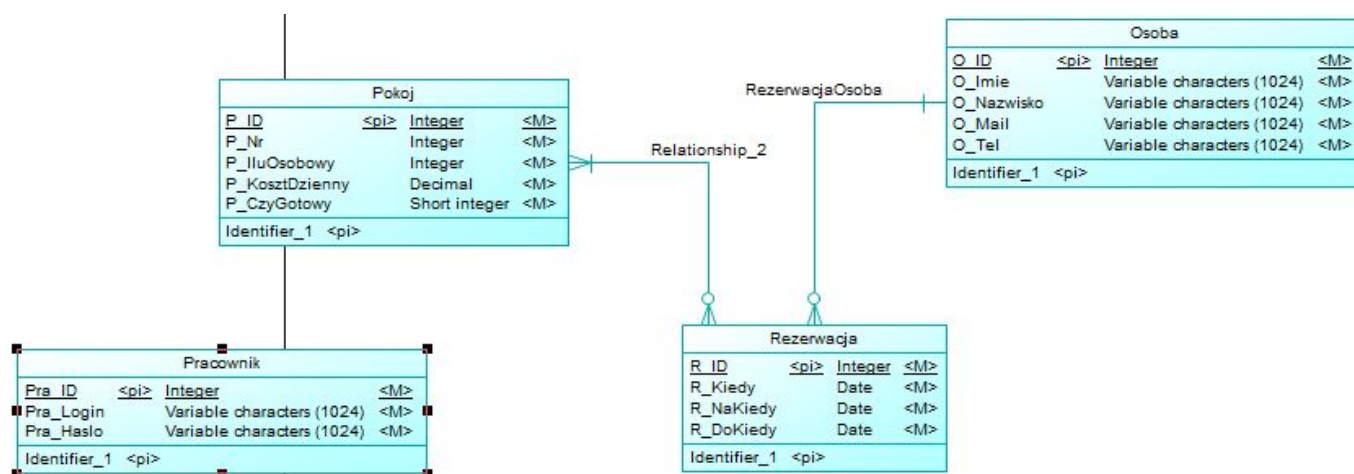


- a. **Klient**- Osoba rezerwująca pokoje w hotelu
 - Może dokonać rezerwacji
 - Może przeglądać swoje rezerwacje
 - Może zalogować się
- b. **Pracownik** - Osoba świadcząca usługi w ramach rezerwacji pokoi hotelowych na poziomie strony internetowej jak i np. recepcjonisty w hotelu
 - Może przeglądać i edytować dane klientów
 - Może przeglądać rezerwacje klientów
 - Może przeglądać i edytować informacje na temat pokoi

3. Słownik pojęć

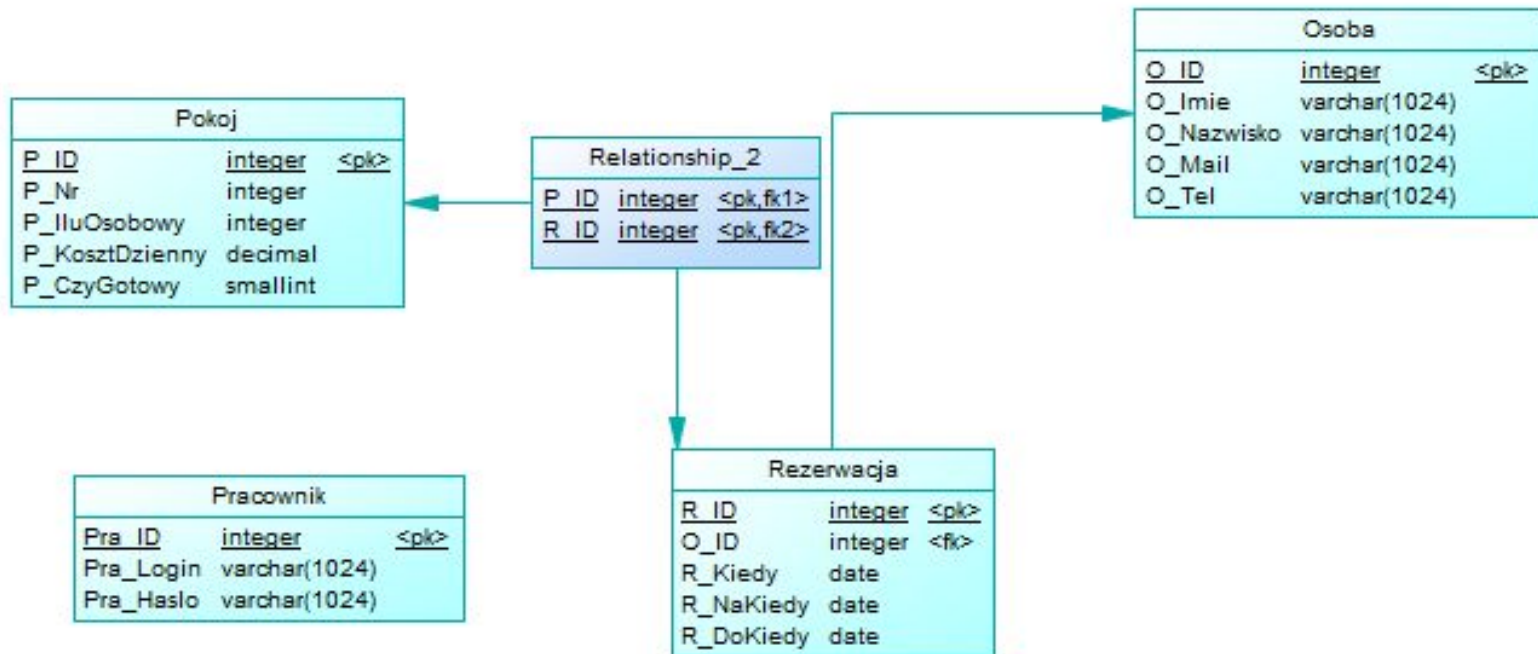
- a. **Osoba** - klient dokonujący rezerwacji w hotelu
- b. **Imię** - imię klienta dokonującego rezerwacji
- c. **Nazwisko** - nazwisko klienta dokonującego rezerwacji
- d. **Mail** - reprezentuje adres internetowej skrzynki pocztowej klienta
- e. **Tel** - reprezentuje numer kontaktowy do klienta
- f. **Pracownik** - Osoba świadcząca usługi w ramach rezerwacji pokoi hotelowych na poziomie strony internetowej jak i np. recepcjonisty w hotelu
- g. **Login** - reprezentuje login używany w procesie logowania przez pracownika
- h. **Hasło** - reprezentuje hasło używane w procesie logowania przez pracownika
- i. **Rezerwacja** - wynajęcie pokoju przez klienta na określony czas określonych pokoi
- j. **Kiedy** - reprezentuje datę dokonania rezerwacji przez klienta
- k. **NaKiedy** - reprezentuje datę od kiedy klient chce korzystać z pokoju w ramach rezerwacji
- l. **DoKiedy** - reprezentuje datę zakończenia korzystania z pokoju przez klienta
- m. **Pokoj** - jest to pomieszczenie lub zespół pomieszczeń wynajmowanych w ramach rezerwacji posiadających swój Nr.
- n. **Nr** - jest swoistego rodzaju identyfikatora fizycznego pomieszczenia.
- o. **IluOsobowy** - jest to typ pomieszczenia świadczący o ilości miejsc w pomieszczeniu
- p. **KosztDzienny** - reprezentuje koszt dzienny za zarezerwowany pokój
- q. **CzyGotowy** - Informuje czy pomieszczenie jest możliwe do wynajęcia (mogą trwać np prace remontowe)

4. Opis modelu konceptualnego



- Encja Osoba** - zawiera informacje na temat klienta dokonującego rezerwacji w hotelu
 - O_ID** - pole to jest identyfikatorem klienta
 - O_Imie** - imię klienta dokonującego rezerwacji
 - O_Nazwisko** - nazwisko klienta dokonującego rezerwacji
 - O_Mail** - reprezentuje adres internetowej skrzynki pocztowej klienta
 - O_Tel** - reprezentuje numer kontaktowy do klienta
- Encja Pracownik** - Przechowuje informację na temat pracownika i jest to osoba świadcząca usługi w ramach rezerwacji pokoi hotelowych na poziomie strony internetowej jak i np. recepcjonisty w hotelu
 - P_ID** - pole to jest identyfikatorem pracownika
 - Pra_Login** - reprezentuje login używany w procesie logowania przez pracownika
 - Pra_Haslo** - reprezentuje hasło używane w procesie logowania przez pracownika
- Encja Rezerwacja** - zawiera informacje na temat dokonanej przez użytkownika rezerwacji.
 - R_ID** - pole to jest identyfikatorem rezerwacji
 - R_Kiedy** - reprezentuje datę dokonania rezerwacji przez klienta
 - R_NaKiedy** - reprezentuje datę od kiedy klient chce korzystać z pokoju w ramach rezerwacji
 - R_DoKiedy** - reprezentuje datę zakończenia korzystania z pokoju przez klienta
- Encja Pokoj** - Przechowuje informacje na temat pokoi.
 - P_ID** - pole to jest identyfikatorem pracownika
 - P_Nr** - jest swoistego rodzaju identyfikatora fizycznego pomieszczenia.
 - P_IluOsobowy** - jest to typ pomieszczenia świadczący o ilości miejsc w pomieszczeniu
 - P_KosztDzienny** - reprezentuje koszt dzienny za rezerwowany pokój
 - P_CzyGotowy** - Informuje czy pomieszczenie jest możliwe do wynajęcia (mogą trwać np prace remontowe)

5. Opis modelu fizycznego



Na podstawie modelu konceptualnego został wygenerowany model fizyczny zawierający 4 tabel połączonych relacjami opisanymi w modelu konceptualnym. Relacja wiele do wielu, która nie jest fizycznie możliwa do zrealizowania, została rozwiązana za pomocą dodatkowej tabeli: Relationship_2. Relacje wiele do wielu są więc realizowane przy pomocy tabel pośredniczących.

6. Kod Sql do tworzenia tabel

```
/*=====*/
/* Database name: Database_1 */
/* DBMS name: Microsoft SQL Server 2014 */
/* Created on: 08.09.2017 20:23:46 */
/*=====*/
drop database Database_1
go
/*=====*/
/* Database: Database_1 */
/*=====*/
create database Database_1
go
use Database_1
go
/*=====*/
/* Table: Osoba */
/*=====*/
create table Osoba (
    O_ID int not null,
    O_Imie varchar(1024) not null,
    O_Nazwisko varchar(1024) not null,
    O_Mail varchar(1024) not null,
    O_Tel varchar(1024) not null,
    constraint PK_OSoba primary key (O_ID)
)
go
/*=====*/
/* Table: Pokoj */
/*=====*/
create table Pokoj (
    P_ID int not null,
    P_Nr int not null,
    P_IluOsobowy int not null,
    P_KosztDzienny decimal not null,
    P_CzyGotowy smallint not null,
    constraint PK_POKOJ primary key (P_ID)
)
go
/*=====*/
/* Table: Relationship_2 */
/*=====*/
create table Relationship_2 (
    P_ID int not null,
    R_ID int not null,
    constraint PK_RELATIONSHIP_2 primary key (P_ID, R_ID)
)
go
/*=====*/
/* Index: Relationship_2_FK */
/*=====*/
create nonclustered index Relationship_2_FK on Relationship_2 (P_ID ASC)
go
/*=====*/
/* Index: Relationship_3_FK */
/*=====*/
create nonclustered index Relationship_3_FK on Relationship_2 (R_ID ASC)
go
/*=====*/
/* Table: Rezerwacja */
/*=====*/
```

```

create table Rezerwacja (
    R_ID          int          not null,
    O_ID          int          not null,
    R_Kiedy       datetime     not null,
    R_NaKiedy     datetime     not null,
    R_DoKiedy     datetime     not null,
    constraint PK_REZERWACJA primary key (R_ID)
)
go
/*=====*/
/* Index: RezerwacjaOsoba_FK                      */
/*=====*/
create nonclustered index RezerwacjaOsoba_FK on Rezerwacja (O_ID ASC)
go
alter table Relationship_2
    add constraint FK_RELATION_RELATIONS_POKOJ foreign key (P_ID)
        references Pokoj (P_ID)
go
alter table Relationship_2
    add constraint FK_RELATION_RELATIONS_REZERWAC foreign key (R_ID)
        references Rezerwacja (R_ID)
go
alter table Rezerwacja
    add constraint FK_REZERWAC_REZERWACJ_OSOBA foreign key (O_ID)
        references Osoba (O_ID)
go
use Database_1
go
create table Pracownik (
    Pra_ID        int          not null,
    Pra_Login     varchar(1024) not null,
    Pra_Haslo     varchar(1024) not null,
    constraint PK_Pracownik primary key (Pra_ID)
)
go
insert into Pracownik(Pra_ID, Pra_Login,Pra_Haslo) values (0, 'Bartosz', 'Gronostaj')
go

```

7. Opis aplikacji WebForms

Aplikacja Web Forms została stworzona przy użyciu środowiska Visual Studio 2017 Enterprise Edition w języku C# przy użyciu Web Forms. Aplikacja ma za zadanie łączyć się z lokalną bazą SQL uruchomioną w środowisku Microsoft SQL Server 2016.

Zadania do zrealizowania przy tej aplikacji to:

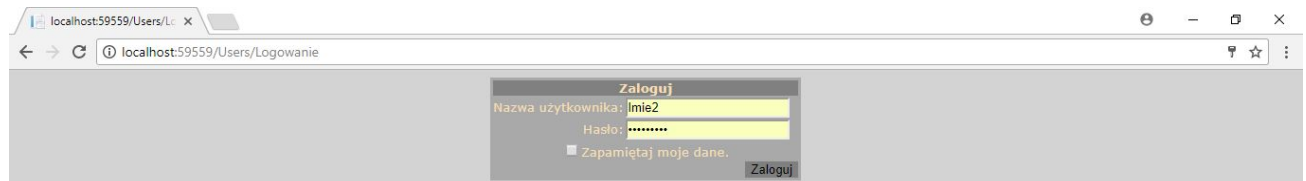
5-wykorzystanie mechanizmów personalizacji stron za pomocą kontrolki Web Parts

7-Wykorzystanie i opracowanie kontrolki użytkownika (rezerwacja pokoju - kalendarz)

8-wykorzystanie Entity Frameworks dla połączeń i dostępu do baz danych (przeglądanie danych klienta - panel Pracownika)

9 - Wykorzystanie usług WWW (logowanie użytkownika i sesja)

Okno Logowania

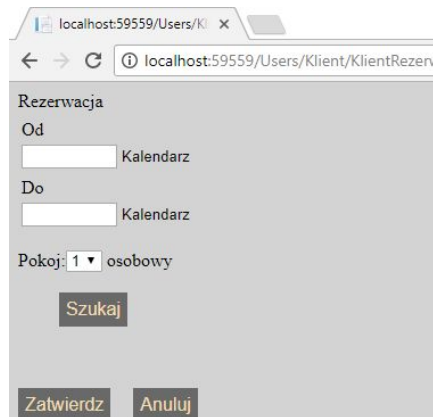


Okno Klienta



Od	Do	NrPokoju	KoszDzienny	IluOsobowy
06.09.2017	22.09.2017	3	3	8
10.10.2017	20.10.2017	6	5	4

Okno Rezerwacji dla Klienta



Okno Pracownika

localhost:59559/Users/P/ x

localhost:59559/Users/Pracownik/PracownikForm

Przeglądaj dane KlientówPrzeglądaj RezerwacjePrzeglądaj PokojeWyloguj

	O_ID	O_Imie	O_Nazwisko	O_Mail	O_Tel
Edytuj Usun Wybierz	1	Albert	Nazwisko1	mail1	000000000
Edytuj Usun Wybierz	2	Imie2	Nazwisko2	mail2	000000000
Edytuj Usun Wybierz	3	Imie3	Nazwisko3	mail3	000000000
Edytuj Usun Wybierz	4	Imie4	Nazwisko4	mail4	000000000
Edytuj Usun Wybierz	5	Imie5	Nazwisko5	mail5	000000000
Edytuj Usun Wybierz	6	Imie6	Nazwisko6	mail6	000000000
Edytuj Usun Wybierz	7	Imie7	Nazwisko7	mail7	000000000
Edytuj Usun Wybierz	8	Imie8	Nazwisko8	mail8	000000000
Edytuj Usun Wybierz	9	Imie9	Nazwisko9	mail9	000000000
Edytuj Usun Wybierz	10	Robert	Mucha	mr@gmail.com	79114422xx

8. Opis aplikacji MVC

Aplikacja MVC została stworzona przy użyciu środowiska Visual Studio 2017 Enterprise Edition w języku C#. Aplikacja ma za zadanie łączyć się z lokalną bazą SQL uruchomioną w środowisku Microsoft SQL Server 2016.

Dodatkowo aplikacja ma mieć podobną funkcjonalność co aplikacja Web Forms z tym wyjątkiem że tutaj sami decydujemy o użytych technologiach.

Okno Logowania



Logowanie

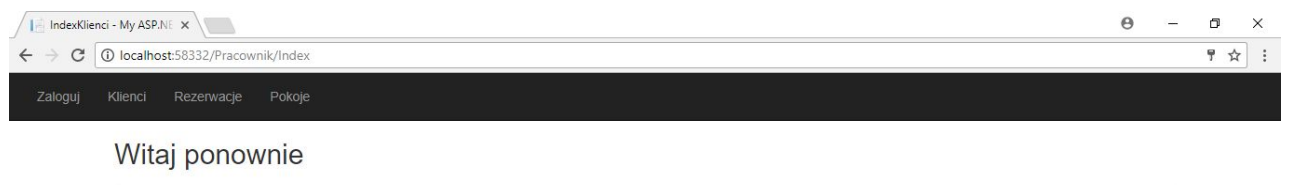
User name

Password

Remember on this computer ☐

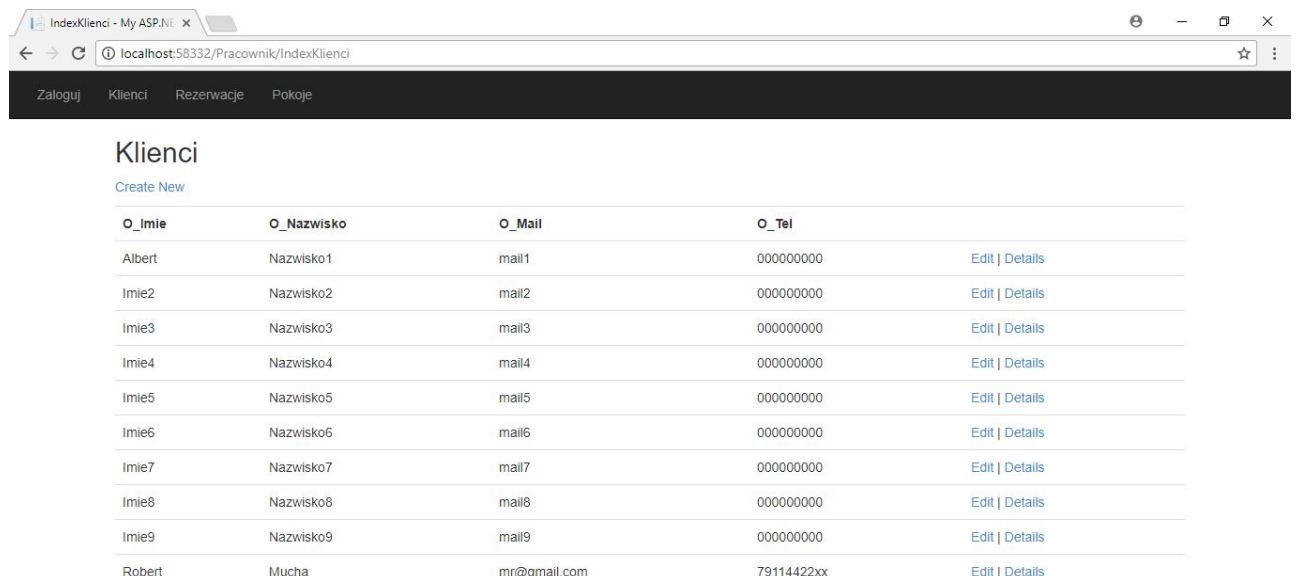
Login

Okno Pracownika



Witaj ponownie

Okno przeglądania tabeli Klienci



Klienci

[Create New](#)

O_Imie	O_Nazwisko	O_Mail	O_Tel	
Albert	Nazwisko1	mail1	000000000	Edit Details
Imie2	Nazwisko2	mail2	000000000	Edit Details
Imie3	Nazwisko3	mail3	000000000	Edit Details
Imie4	Nazwisko4	mail4	000000000	Edit Details
Imie5	Nazwisko5	mail5	000000000	Edit Details
Imie6	Nazwisko6	mail6	000000000	Edit Details
Imie7	Nazwisko7	mail7	000000000	Edit Details
Imie8	Nazwisko8	mail8	000000000	Edit Details
Imie9	Nazwisko9	mail9	000000000	Edit Details
Robert	Mucha	mr@gmail.com	79114422xx	Edit Details

Okno z tabelą Rezerwacje

IndexRezerwacje - My ASP.NET

localhost:58332/Pracownik/IndexRezerwacje

ZalogujKlienciRezerwacjePokoje

Rezerwacje

OdKiedy	DoKiedy	Imie	Nazwisko	PokojNr
06.09.2017	22.09.2017	Imie2	Nazwisko2	3
10.10.2017	20.10.2017	Imie2	Nazwisko2	6
03.09.2017	16.09.2017	Imie4	Nazwisko4	6
04.09.2017	18.09.2017	Imie5	Nazwisko5	4
08.09.2017	26.09.2017	Imie5	Nazwisko5	0
02.09.2017	14.09.2017	Imie7	Nazwisko7	5
09.09.2017	28.09.2017	Imie7	Nazwisko7	8
07.09.2017	24.09.2017	Imie8	Nazwisko8	1
10.09.2017	30.09.2017	Imie8	Nazwisko8	7
05.09.2017	20.09.2017	Imie9	Nazwisko9	9

Okno z tabelą Pokoje

IndexPokoje - My ASP.NET

localhost:58332/Pracownik/IndexPokoje

ZalogujKlienciRezerwacjePokoje

Pokoje

Create New

P_Nr	P_IluOsobowy	P_KosztDzienny	P_CzyGotowy	
0	2	6,00	<input checked="" type="checkbox"/>	Edit Details Delete
1	2	4,00	<input checked="" type="checkbox"/>	Edit Details Delete
3	8	3,00	<input type="checkbox"/>	Edit Details Delete
4	9	1,00	<input checked="" type="checkbox"/>	Edit Details Delete
5	5	0,00	<input type="checkbox"/>	Edit Details Delete
6	4	5,00	<input checked="" type="checkbox"/>	Edit Details Delete
7	1	7,00	<input checked="" type="checkbox"/>	Edit Details Delete
8	2	8,00	<input checked="" type="checkbox"/>	Edit Details Delete
9	6	2,00	<input type="checkbox"/>	Edit Details Delete

Okno Klienta(Twoje Rezerwacje)

Main - My ASP.NET App

localhost:58332/Klient/Index

ZalogujTwojeRezerwacjeNowa RezerwacjaPokoje

Twoje Rezerwacje

OdKiedy	DoKiedy	Imie	Nazwisko	PokojNr
06.09.2017	22.09.2017	Imie2	Nazwisko2	3
10.10.2017	20.10.2017	Imie2	Nazwisko2	6

Okno tworzenia rezerwacji

Rezerwacja - My ASP.NET

localhost:58332/Klient/Rezerwacje

ZalogujTwojeRezerwacjeNowa RezerwacjaPokoje

Rezerwacja

OdKiedy

DoKiedy

PokojNr

Create

[Back to List](#)

Okno z tabelą Pokoje

IndexPokoje - My ASP.NET

localhost:58332/Klient/Pokoje

ZalogujKlenciRezerwacjePokoje

Pokoje

P_Nr	P_IluzOsobowy	P_KosztDzienny
0	2	6,00
1	2	4,00
3	8	3,00
4	9	1,00
5	5	0,00
6	4	5,00
7	1	7,00
8	2	8,00
9	6	2,00

9. Wnioski

Celem niniejszego projektu było stworzenie bazy danych odpowiadającej tematowi projektu, oraz aplikacji które będą z nią współpracować. Jedną z nich miała być aplikacja wykonana przy użyciu WebForms z użyciem odpowiednich technologii zadanych wraz z otrzymaniem tematu pracy. Druga aplikacja miała zostać zrealizowana przy użyciu MVC już bez innych ograniczeń. Przy tworzeniu obu aplikacji należało skorzystać z ASP.NET. Projekt ten pozwolił zastosować zdobytą wiedzę teoretyczną i złożyć ją w jedną całość w praktyce. Dzięki technologii .NET w prosty sposób można było pobierać dane z bazy danych i przechowywać je w pamięci aplikacji. Przy tworzeniu aplikacji internetowej ASP.NET - WebForms użyty został GridView oraz SqlDataSource, dzięki czemu w prosty i zautomatyzowany sposób można było łączyć się z bazą używając poleceń sql. Również widoki zostały szybko dostosowane pod bazę danych, co umożliwiło zaoszczędzenie czasu, dzięki użyciu środowiska Microsoft Visual Studio. Jednakże ze względu na zadaną technologię został także użyty EntityFramework, który jest doskonałym rozwiązaniem dla osób które nie lubią bądź nie znają języka TSQL.

Ostatecznie uważam że znacznie lepiej jest wykorzystywać WebForms niż MVC ponieważ przy MVC należy dodatkowo znać się na tworzeniu stron przy użyciu HTML (cały View) w wyniku czego znajomość komponentów graficznych z WindowsForms, WPF czy WebForms na nic się nie przydaje i należy się uczyć od podstaw sposobu prezentacji graficznej danych. Tylko model oraz Controller opiera się na języku C#. Kolejnym plusem dla WebForms jest dostępny tryb Designera - możliwość podglądu wyglądu strony, oraz drag & drop czyli możliwość tworzenia aplikacji przy użyciu gotowych komponentów. Jednakże zaletą MVC jest rozdzielenie logiki biznesowej od widoku co mimo wszystko można wdrożyć także do WebForms(jako wzorzec projektowy).

Podsumowując raz jeszcze- w chwili gdybym miał wybór przy tworzeniu kolejnej aplikacji bez wątpienia wybrałbym WebForms zamiast MVC.