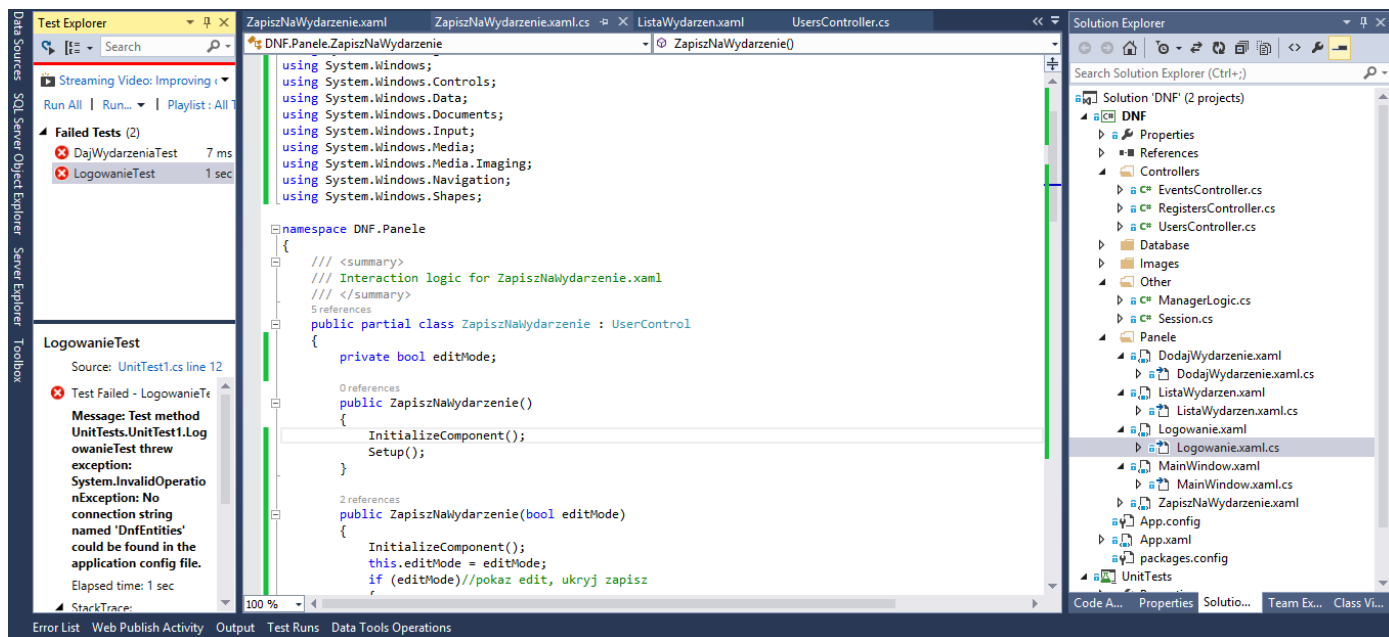


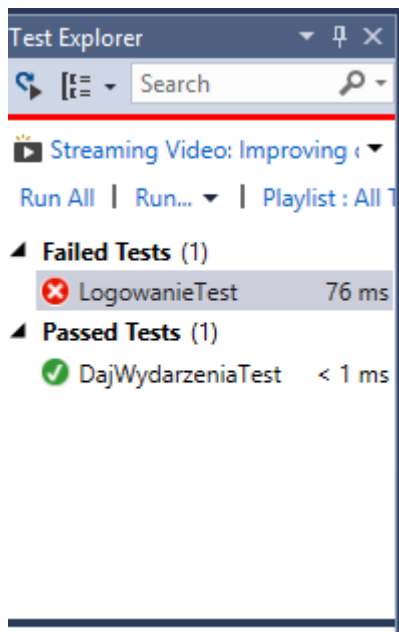
Dokumentacja

1. Testy jednostkowe
2. Implementacja przypadków alternatywnych
3. Wykaz bazy danych
4. Układ projektu
5. Szczegóły klas sterujących
6. Testy automatyczne
7. Metryki zaimplementowanej aplikacji

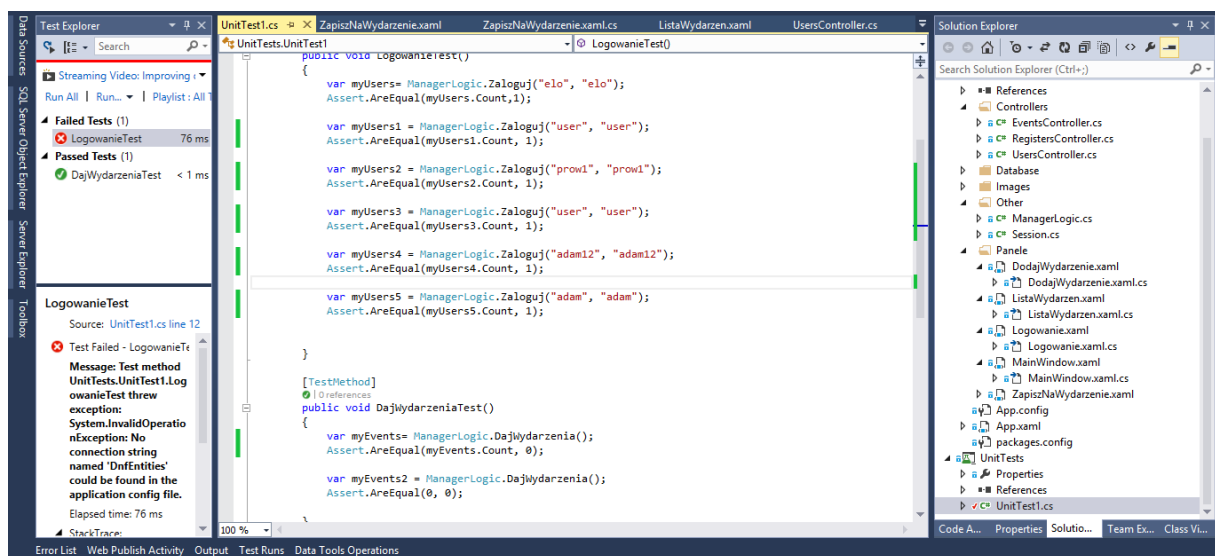
1. Testy jednostkowe

Wykonano testy 2 metod: Logowania do systemu a także otrzymywania wydarzeń.





Kod testów:



Testy przeprowadzono kilkakrotnie, wersja finalna aplikacji spełniała wymagane testy.

2. Implementacja przypadków alternatywnych

Przykłady implementacji przypadków alternatywnych:

```

ev.Nazwa = nazwaTb.Text;
ev.Opis = opisTbText.Text;
ev.Data = (DateTime)dataTb.SelectedDate;
ev.IloscWolnych = ev.IloscMiejsc = long.Parse(iloszMiejscTb.Text);
ev.KategoriaWiekowa = long.Parse(kategoriaWiekowaTb.Text);
ev.Miejsce = miejsceTb.Text;

bool infected = false;
foreach (var item in ManagerLogic.SzukajWydarzeniaProwadzacy( Session.zalogowanyUzytkownik.Login))
{
    if (item.Data == ev.Data && item.Id != ev.Id)
    {
        MessageBox.Show("Masz wydarzenie o tej porze!", "Błąd");
        infected = true;
        return;
    }
    if (item.Miejsce.Equals(ev.Miejsce) && item.Id != ev.Id)
    {
        MessageBox.Show("Sala jest zajęta!", "Błąd");
        infected = true;
        return;
    }
}

if (!infected)
{
    if (isEditMode)
    {
        ManagerLogic.EdytujWydarzenie(Session.aktualneWydarzenie.Id, ev);
        MessageBox.Show("Zaktualizowano wydarzenie", "Sukces");
    }
}

```

```

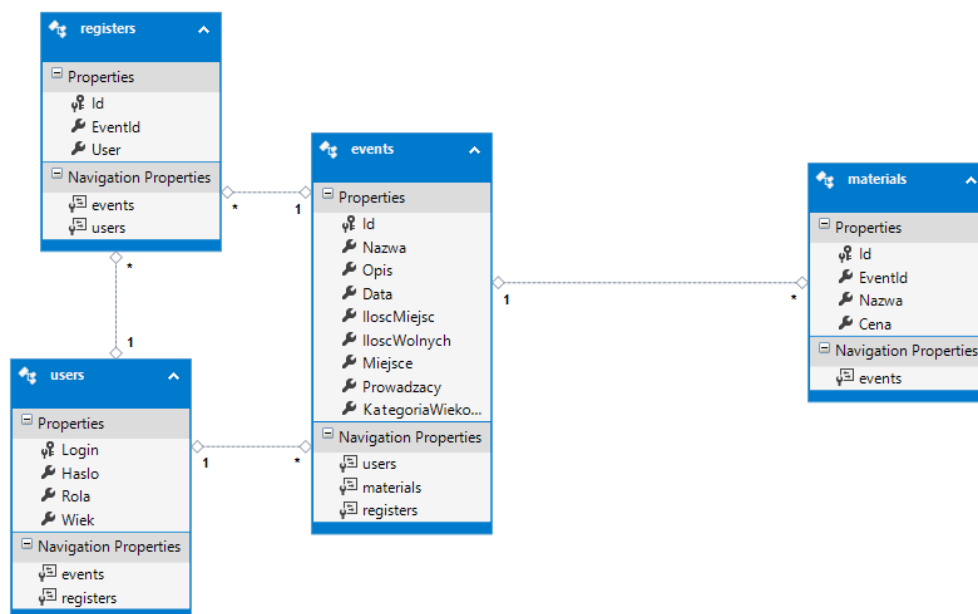
1 reference
private void zapiszBtn_Click(object sender, RoutedEventArgs e)
{
    if (Session.aktualneWydarzenie.IloscWolnych == 0)
        MessageBox.Show("brak wolnych miejsc!", "Błąd");
    if (Session.aktualneWydarzenie.KategoriaWiekowa > Session.zalogowanyUzytkownik.Wiek)
        MessageBox.Show("Zła kategoria wiekowa!", "Błąd");

    registers reg = new registers();
    reg.EventId = Session.aktualneWydarzenie.Id;
    reg.User = Session.zalogowanyUzytkownik.Login;
    ManagerLogic.Zarejestruj(reg);
    Session.aktualneWydarzenie.IloscWolnych -= 1;
    ManagerLogic.EdytujWydarzenie(Session.aktualneWydarzenie.Id, Session.aktualneWydarzenie);
}

```

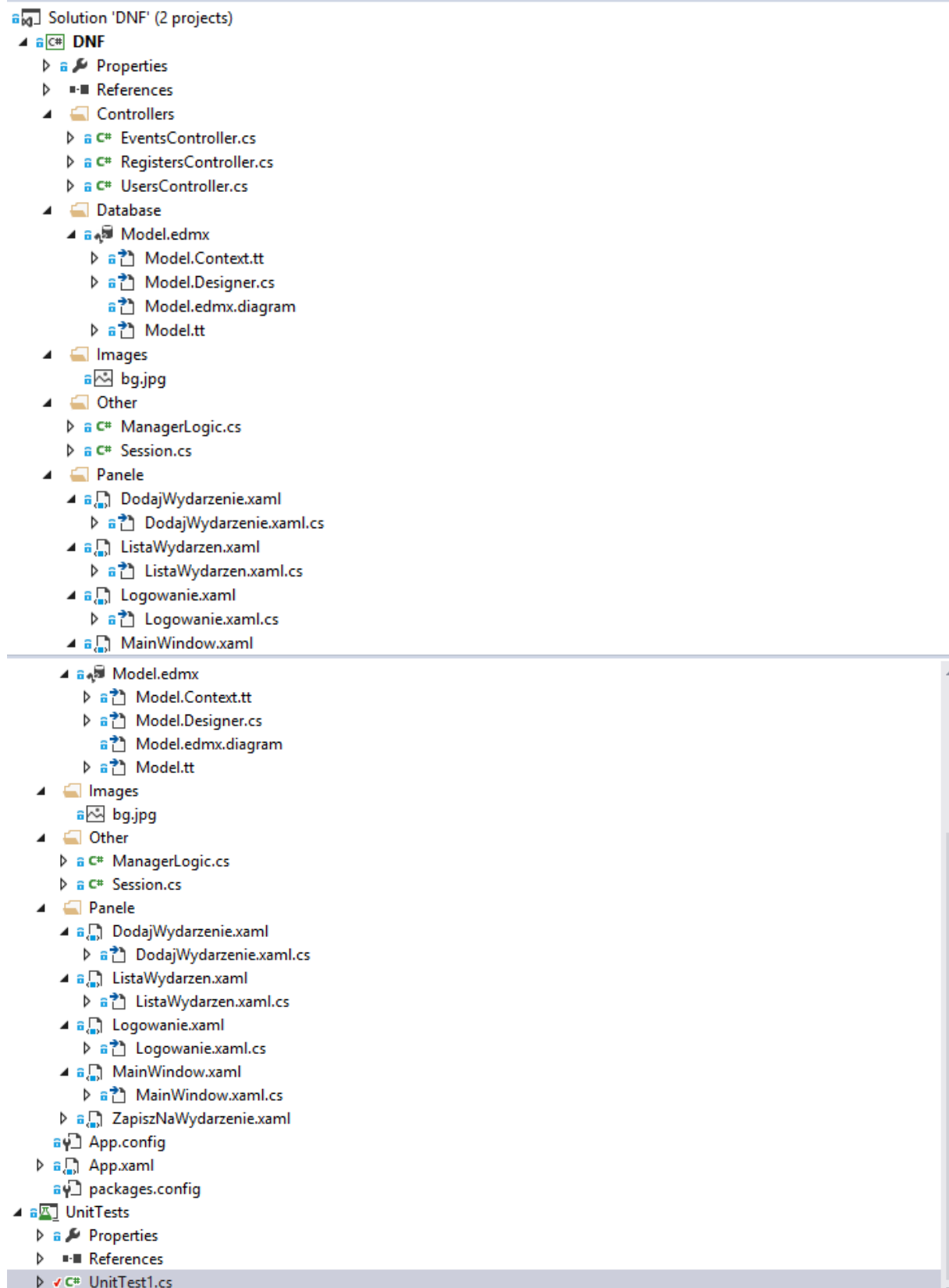
3. Zaimplementowana baza danych

Wykaz programu Visual Studio 2013

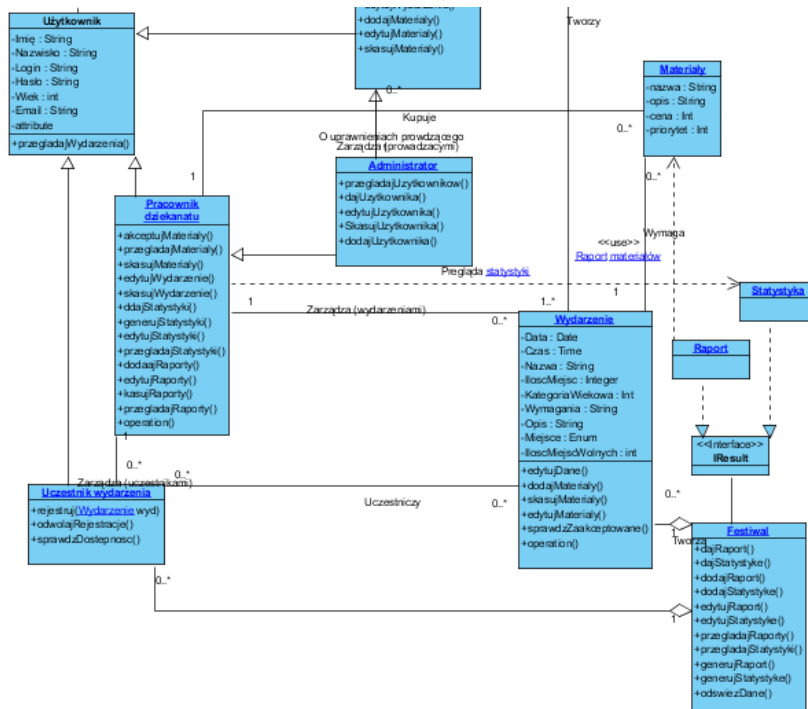


4. Układ projektu

Układ projektu zgodnie z modelem architektonicznym



5. Szczegóły klas sterujących



8. Metryki zaimplementowanej aplikacji

Code Metrics Results					
Filter: <input type="text" value="None"/>		Min: <input type="text" value=""/>	Max: <input type="text" value=""/>		
Hierarchy	Maintainabil...	Cyclomatic C...	Depth of Inhe...	Class Coupling	Lines of Code
▸ DNF (Debug)	80	151	9	67	326

Code Metrics Results						
Filter: None		Min:		Max:		
Hierarchy		Maintainabilit...	Cyclomatic C...	Depth of Inhe...	Class Coupling	Lines of Code
▾ [C#] DNF (Debug)		80	151	9	67	326
▸ {} DNF		87	11	9	12	17
▸ {} DNF.Controllers		68	26	1	23	100
▸ {} DNF.Database		93	70	2	11	74
▸ {} DNF.Other		92	12	1	7	25
▸ {} DNF.Panele		66	32	9	40	110