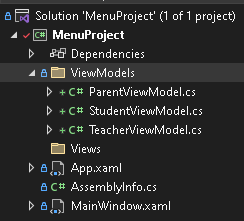
Menü készítése MVVM alapokon

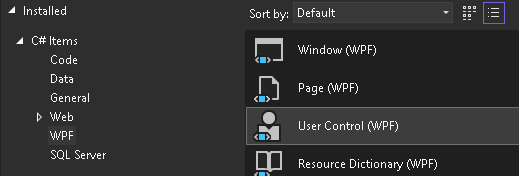
Készítsünk WPF projektet MenuProject néven.

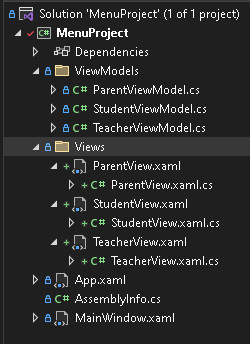
Hozzuk létre a ViewModel-eket:



A View rétegben UserControl oldalakat hozzunk létre!

[User Control in WPF (c-sharpcorner.com)](https://www.c-sharpcorner.com/UploadFile/mahesh/user-control-in-wpf/)

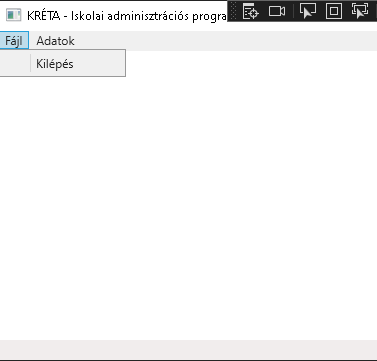




* MainWindow áthelyezése a Views mappába. App.xaml módosítása.

Helyezzük át a MainWindow-t is a View-mappába. Ügyeljünk a namespace módosításokra.

* MainWindowView és összekötése a MianWindow-al
* A MainWindow Grid felépítése
* A felső részbe kerül a menü
* Az ablak alsó részébe egy státusz sor



* Az ablak középső részét a tartalom tölti ki ami a ContentControl-ba kerül

[Switching between WPF XAML views using MVVM DataTemplate - technical-recipes.com](https://www.technical-recipes.com/2016/switching-between-wpf-xaml-views-using-mvvm-datatemplate/)

* A szükséges NuGet csomagok

<ItemGroup>

<PackageReference Include="CommunityToolkit.Mvvm" Version="8.2.1" />

<PackageReference Include="Extended.Wpf.Toolkit" Version="4.5.1" />

</ItemGroup>

* Elkészítüjük a BaseViewModelt-t, mivel az ObservableObject absztrakt osztály, nem lehet példányosítani



public class BaseViewModel : ObservableObject

{

}

* Az oldalnak van egy gyermek ablaka és a státuszsornak egy kezdő szövege, hogy lássuk, hogy működik
* A program indulása után egy „WelcomeView” oldal jelenik majd meg.
* A kunstruktorba létrehozzuk a nyitó oldalt, a xaml-ban pedig a Window.Resource-be megadjuk a ViewModelhez melyik View tartozik.
* Létrehozzuk az egyes menüpontokhoz a parancsokat
* A parancsokat már kötöttük a View-n a menüpontoknál, így csak megadjuk melyik ViewModel-hez melyik View tartozik

<Window.DataContext>

<viewmodel:MainWindowViewModel />

</Window.DataContext>

**<Window.Resources>**

<DataTemplate DataType="{x:Type viewmodel:WelcomeViewModel}">

<view:WelcomeView/>

</DataTemplate>

<DataTemplate DataType="{x:Type viewmodel:StudentViewModel}">

<view:StudentView/>

</DataTemplate>

<DataTemplate DataType="{x:Type viewmodel:TeacherViewModel}">

<view:TeacherView/>

</DataTemplate>

<DataTemplate DataType="{x:Type viewmodel:ParentViewModel}">

<view:ParentView/>

</DataTemplate>

**</Window.Resources>**

A program már működik.

Most beépítjük a diák adat felvételt a diák menüpontra:

