TODO

done általánosságban MVVM

miért mvvm

osztály leírások

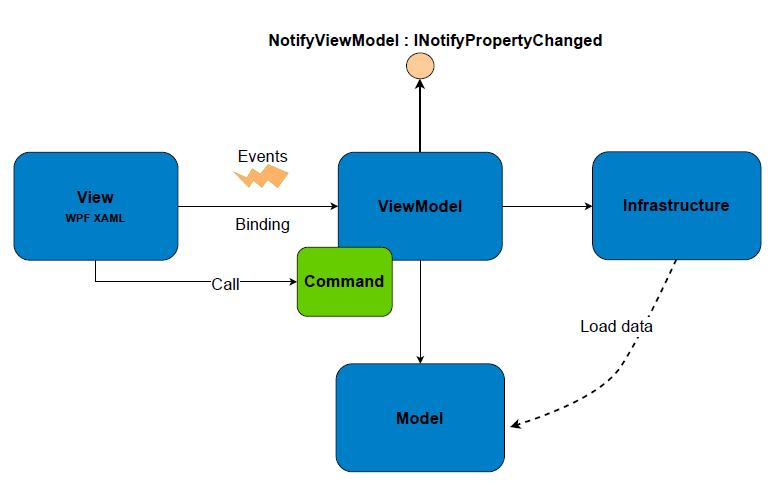
----

Mvvm: Model-View-ViewModel (MVVM) minta

Az MVVM a szoftferfejlesztésben használt minta, amely Microsoft által let kifejlesztve specifikusan, Presentation mintát alapúl véve specializálódott (állapotot és viselkedést reprezentálja függetlenül a felhasználói grafikus felülettől).

Az MVVM alapja a Model-View-Controller (MVC), olyan modern UI platformokra írányul, mint (WPF és Silverlight), amelyekben van UX fejlesztési lehetőség azaz más követelményei vannak, mint egy "hagyományosabb" felület fejlesztésének.[1]

Az MVVM az ügyfélalkalmazások létrehozásának egyik módja, amely kihasználja a WPF platform alapvető jellemzőit, lehetővé teszi az alkalmazás funkcióinak egyszerű tesztelését, és segíti a fejlesztőket és a tervezőknet a kevesebb műszaki nehézséggel történő együttmőködésben.



View: A nézetet az XAML határozza meg, és a kód mögött nem lehet logika. A nézetmodellhez csak az adatok kötésével kötődik, úgynevezett Data Binding (egy állandó kapcsolatot hoz ).

Model: A modell felelős az adatok olyan módon történő tárolásáért, amelyet a WPF könnyen felhasználhat. Megfelelően az INotifyPropertyChanged és / vagy INotifyCollectionChanged eseményeket kell végrehajtania.

ViewModel: A ViewModel egy specifikus modell kimondottan a nézethez az alkalmazásban, vagy mondhatjuk a nézet absztrakciójaként is. A nézet szempontjából releváns adatokat használja és a nézetek viselkedését, általában parancsokkal köti össze.

**[1] Mastering Windows Presentation Foundation**. Sheridan Yuen. February 17, 2017

--------------------------------------

Helper

GeneralCollection

Infrastructure

Message

Report

XmlHandéer

Model

Resource

View

ViewModel