**MVVM Architecture**

A screenshot of a computer

Description automatically generatedA blue square with white text

Description automatically generated with medium confidence

- Model: Models encapsulate the application’s data and business logic, having absolutely no relation to visual aspects

(Engines)  
\*\* Models send **Notification** signals (can go up to View)

- View model: Contains all application logics (get data from Model, return data, communicate with UI – View)

- View: Receiving User inputs and display UI

\*\* VM communicates with View through **DataBinding**

* Databinding is connecting the “Data Consumer” – View to the “Data Provider” – View Model, such that any changes made to either source or display will be reflected to the other
* Windows Presentation Foundation (WPF) in .NET uses data binding by connecting the properties of target objects and data sources, including CLR, Language Integrated Query and XML objects. Data templates are also provided to control the presentation of data.

**Notifications**

1. **INotifyPropertyChanged**

Allow the Model to notify any property changes of a Model class to the View



A screen shot of a computer program

Description automatically generated with low confidence



A screen shot of a computer program

Description automatically generated with low confidence

*These can be bound directly to the UI (through xaml file!*

**Binding**

1. **ICommand** *(Button clickings* 🡪 *avoid using EventHandler directly coupled to the UI)*