IDSA – Inteligent Data Stock Analyzer.

## Developers :

1. CezaryWronka – data base behavior , task synchronization, chart’s
2. PiotrSacharuk – web data gathering, analyzing
3. PatrykJanota – csv reading, data presenting (main view), renta-wieczysta.

## Main Tasks (sorted with priority 1-5) TODO:

**All(1): CODE REFACTORING** - Spojnosc nazw: widok = to co widzimy bezposrednio w apps.[ Eventy – RaiseEventSTH] [Delegates - ViewName\_DelegateName] [Kolejnosc metod w klasach w sekcji Clean Code – **USE #REGION** **MANDATORY**]

Piotrek(1) : ! Do ROBOTY ! :D

Patryk(1): Addition on the csvEnum for BILANS AKTYWA PASYWA i CASH FLOW.

Patryk(1): Prepare Rzis and Bilans seperation for the database (Cezary part)

Patryk(2): Start scanner architecture module, and basic class implementation on the interface plugs ☺

Cezary(1): Reports (podzial w bazie na Bilans, RZiS, cala reszta)!!!!!

Cezary(2): Clean -> ServiceLocator -> dodac funkcje usuwania ze slownika. Clean bedzie szybszy, albo przerzucic się na **PRISM Service Locator**

Cezary(2): Spolki - nowe zakladki - new Tabbed View Provider(Internal) – zrobic tak, zeby nie bylo burdelu

Cezary(2): Chart Service Provider Interface(GetData(), Refresh(), Delegate do Company Changed?)

Cezary(2): Chart - Zaznaczasz kolumne, wyswietl wykres

Cezary(3): Dezaktywacja CsvUpload\_SaveDB

Cezary(3): Wykres funkcjonalnosc(wybierz max, ogranicz ilosc danych(raporty z 2 lat domyslnie), kolorystyka danych, ...)

Cezary(5): Add Companies/Reports w watkach(inny sposob blokowania aplikacji)

Cezary(5): Odpalenie bazy na innym watku.

Patryk(5): Filtr Box - Aktualizacja cofanie, jednorazowe wczytywanie z bazy(z poczatku, pozniej obiekty lokalnie - async)

Patryk(5): Transpozycja DataGrid View....

## Key Money MAKER features.

GREEN – Implemented into application.

1. **RentaWieczysta** (Calculated) – ver 0
2. DCF (Discounted Cash Flow Value) Wycena DCF. (Most hardcore to do, many external parameters… - very close to TV although DCF is good for new dynamic good potential clear business model like BENEFIT SYSTEM.)
3. Volumen Signal’s ( how to catch ? how to judge if is positive volume(positive information ) or negative(bad informations)
4. WEB Scanner - When quarter result comes out especially in hot season, there is a need for fast selection of companies that behave extra-ordinary. That with point (3) gives us possible investor’s money movment’s into this company, and other financial operators (for exampale TFI, OFE others)
5. Data Aquisitor – get the data from every possible source. PAP, Bankier.pl, Money.pl and other webpages.
6. **Chart Visualisation** – clean and good visualization of the result in the futer, process chart for some kind of external reports. Fast way of comparing more than two companys on the chart possibility to turning off and on the companies, and other LOOK AND FEEL features.
7. All in **scanner** –
   1. Terminal Value(Renta Wieczysta - wartosc teoretyczna spolki do generowania gotowki)
   2. Test TV versus Real Stock Price (Renta wieczysta kontra cenna gieldowa)
   3. Stable quarter growth on the last (4-8-12 quaters – parametrize)
   4. ROE / ROA best in the company enemys environment. (Parametrize)
   5. Other good quality financial parameters which determinize the good company value.

## ADDITIONAL FEATURES.

1. Chart visualization on financial data when user select rows, columns etc.
2. Automatic download of data like SUROWCE , WALUTY etc.
3. **TODO: feel free to add more**

### Luźne przemyślenia:

Mamy 3 typy wyświetlania dla głównego View:

View = Reports DataGrid + chart

View = Company Info

View = Chart

## Clean Code:

Ordering of code in classes(ReSharper does this automatically;) ):

1. Static Variables
2. Static Methods
3. Public Variables
4. Protected Variables
5. Private Variables (3-5 Fields and Props)
6. Constructors (Ctors)
7. Public Methods (Event Handlers at top of public methods)
8. Protected Methods
9. Private Methods (Database Connection, etc.)

The following rules:

* static before anything
* variables before constructors before methods (i consider constructors to be in the category of methods)
* public before protected before private

The idea is that you **define the object** (the data), before the **behaviors** (methods). Statics need to be separated because they aren't really part of the object, nor it's behavior.

# Chart flow in IDSA (by Patric):

             I think more important is coneception about how to do it flexible, for all possible views, that we will get into the further. For now, is easy to get dataTable from datagrid, presenter of view can provide this action, although we must be flexible and be able to visualize the data that user want to see.

             -> user choose the data on the view (Ebit, sales) -> advanced(multiselection EBIT,SALES at once).

             -> user confirms he want the chart by (submenu (rightmouseclick)

             -> presenter get the event action (eventagregator from prism ?)

             -> presenter knows that the user select on view this data by the event,

             -> presenter lunch the ChartServiceProvider (DataInput.)

             -> chart service provider do all the data stuff

             -> presenter informs view that chart is ready to display

             -> view (shows) the chart.

* + We need even more to work **on separate data.** (Czarek you now using the DataBase data not recalculated)

