## 一、简介

本项目是对WPF程序的一个基本结构封装，提供一些基础的库和控件，以便以后能直接运用于项目。

采用技术：

EntityFramework, asp.net webapi,Unity,Automapper,WPF(MVVMLight)

**二、项目结构**

* DataModel： Entityframework实体模型
* Data.Access: 数据访问
* Business.Model: 业务逻辑模型
* Business:业务逻辑
* WebApi: 提供数据服务
* Presentation:表示层（WPF）
* Common:公共类库

**三、Common**

* Extensions:包括时间，字符串扩展类
* Http: http请求的封装
* Log: log4net的封装,记录异常日志
* Utiltiy:添加了md5加密类，Icon的处理
* Validators: WPF模型验证相关类定义

**四、Business**

* IBusiness:业务逻辑层操作的接口定义
* MapperProfile: 基于automapper的，从数据库实体到业务逻辑层模型的转换配置

**五、Business.Model**

* 包含业务逻辑模型和业务逻辑层返回对象等

**六、WebApi**

* DI:基于UnityIoc的依赖注入注册,注册了automapper及业务逻辑层接口和实现。
* App\_Start/JsonConfig:返回json的配置
* App\_Start/UnityConfig:配置UnityContainer
* Startup: 基于owin的token验证的封装。

**七、表现层（WPF）**

* Converters:数据绑定中值转换器的定义，已定义时间类型和字符串的转换。
* CustomControl:自定义控件文件夹。已定义DataGrid分页控件。
* Images:图片资源
* Message:Mvvmlight消息通知消息类封装文件夹，用于viewmodel跟窗口后台交互
* Service:封装一些类用于调用webapi获取数据。
* Style:系统所需wpf控件样式定义。已定义textbox和button基本样式。
* UserControl:用户控件定义。
* ViewModel:视图模型文件夹,其中ViewModelLocator类用于定义视图模型实例,以便在视图（view）中设置DataContext。
* Views:视图文件夹。目前已经实现一些demo。包括treeview, pdfViewer, Validation，数据绑定及值转换器，DateTemplate实例, DataGrid分页，DataGrid采用controltemplate重写外观等。