

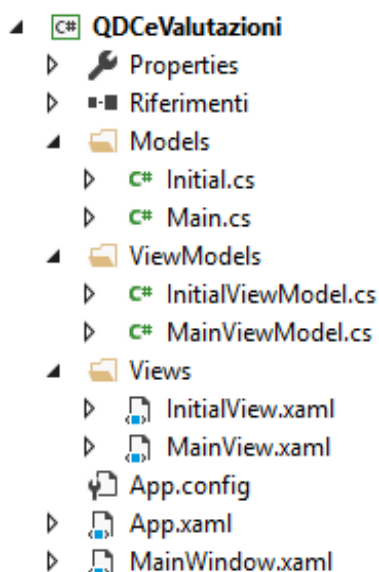
Diario di lavoro

Luogo	SAM Trevano
Data	20.09.2019

Lavori svolti

Oggi ho potuto iniziare la fase di implementazione del progetto. Ho creato una soluzione vuota chiamandola GestioneQDC. All'interno ho creato il progetto QDCeValutazioni che conterrà le varie classi che andranno a comporre il progetto. Il progetto è di tipo WPF. Ho creato un secondo progetto, questa volta di tipo Class Library e l'ho chiamato QDCeValutazioni.DA.

All'interno della libreria di classi QDCeValutazioni.DA ho creato una classe QDC. Ho creato questa classe tramite il programma MenuSAMT. In questo modo sono state create la struttura di cartelle e i vari file in modo automatico. Per la precisione è stata creata la directory Models con all'interno la classe QDC.cs. La cartella Models conterrà tutte le classi del progetto. Per ora contiene la classe QDC.cs e serve per definire il costruttore della classe, gli attributi (variabili o costanti) e i metodi che deve gestire. Questa classe contiene il costruttore e un id. Servirà come Super-classe perché sarà ereditata da tutte le altre classi che verranno create in futuro. È stata creata anche la directory Services che contiene l'interfaccia IDataRepository che contiene dei metodi base (come insert e delete) che vengono ereditati dall'interfaccia IQDCDataRepository che si occupa di gestire quei metodi.



All'interno del progetto WPF ho creato una classe Main, che sarà la classe "di partenza" del progetto e rappresenterà la base di tutte le maschere.

Ho creato poi una seconda classe, chiamata Initial, che sarà la classe della maschera iniziale (default). Questa classe gestirà la selezione della scelta tra Crea QDC e Seleziona QDC.

Creando una nuova classe anche in questo caso viene creata la struttura MVVM adeguata con le cartelle Models, ViewModels e Views.

In MainViewModels.cs viene aggiunto in automatico il property changed per notificare quando avviene un evento:

```
public event PropertyChangedEventHandler PropertyChanged;
private Main model;
```

In MainView.xaml vengono aggiunte in automatico le proprietà xmlns per gestire i riferimenti.

Infine ho creato la parte grafica della view iniziale.

Problemi riscontrati e soluzioni adottate

Quando viene generata la struttura MVVM e vengono assegnate le varie dipendenze e i riferimenti, nelle View il riferimento al ViewModel non funziona. Sostanzialmente non riesce a trovare il riferimento al file ViewModel.cs. Inoltre quando viene salvato e chiuso il progetto esso viene salvato correttamente. Quando però viene riaperto mancano nel progetto WPF le cartelle Models, Views e ViewModels, anche se in realtà nella cartella del progetto esistono ancora. Nel progetto DA invece vengono caricate correttamente le cartelle Models e Services.

Punto della situazione rispetto alla pianificazione

Ho invertito la creazione del database con la creazione della view iniziale, perché questa view per funzionare non necessita nessun dato.

Programma di massima per la prossima giornata di lavoro

La prossima lezione finirò di lavorare alla view iniziale e cercherò di risolvere gli errori legati alla struttura MVVM e al caricamento dei file quando viene riaperto il progetto. Se avanzasse tempo comincerò a lavorare sul database.