

Diari di lavoro

Luogo	SAM Trevano
Data	21.02.2018

Lavori svolti / Informazioni

Oggi ho realizzato l'interfaccia per scegliere quale esercizi svolgere (Immagine presente alla seconda pagina).

Infine, ho fatto una 'riunione' con il professore Bernasconi per alcuni chiarimenti su come usare il Messenger: classe utile alla comunicazione delle schermate, in modo da avere varie interfacce nell'applicazione come nel mio caso.

Questi sono i bottoni per mi collegheranno alle scelte di goni esercizio di ogni categoria:

```
<Button x:Name="btnSimply" Content="ESERCIZI SEMPLICI" FontSize="30" Width="600" Height="100" Grid.Row="0" VerticalAlignment="Bottom"/>
<Button x:Name="btnMedium" Content="ESERCIZI MEDI" FontSize="30" Width="600" Height="100" Grid.Row="1" VerticalAlignment="Center"/>
<Button x:Name="btnHard" Content="ESERCIZI DIFFICILI" FontSize="30" Width="600" Height="100" Grid.Row="2" VerticalAlignment="Top"/>
```

Questi saranno legati ad altre 'Views' tramite metodi.

Negli ultimi momenti ho realizzato anche i metodi per lavorare sulla tabella degli allievi (presente nelle seguenti pagine).

Problemi riscontrati e soluzioni adottate

Il problema principalmente che ho avuto in questi giorni era quello che non riuscivo a salvarmi l'immagine scelta da un utente che si 'registra' nell'applicazione.

Per risolvere questo problema ho dovuto aggiungere questo parametro nella mia lista d'immagini in modo che venisse anche riconosciuta nel file dove eseguo il codice di aggiunta del allievo:

```
SelectedItem="{Binding Path=SelectedItem}"
```

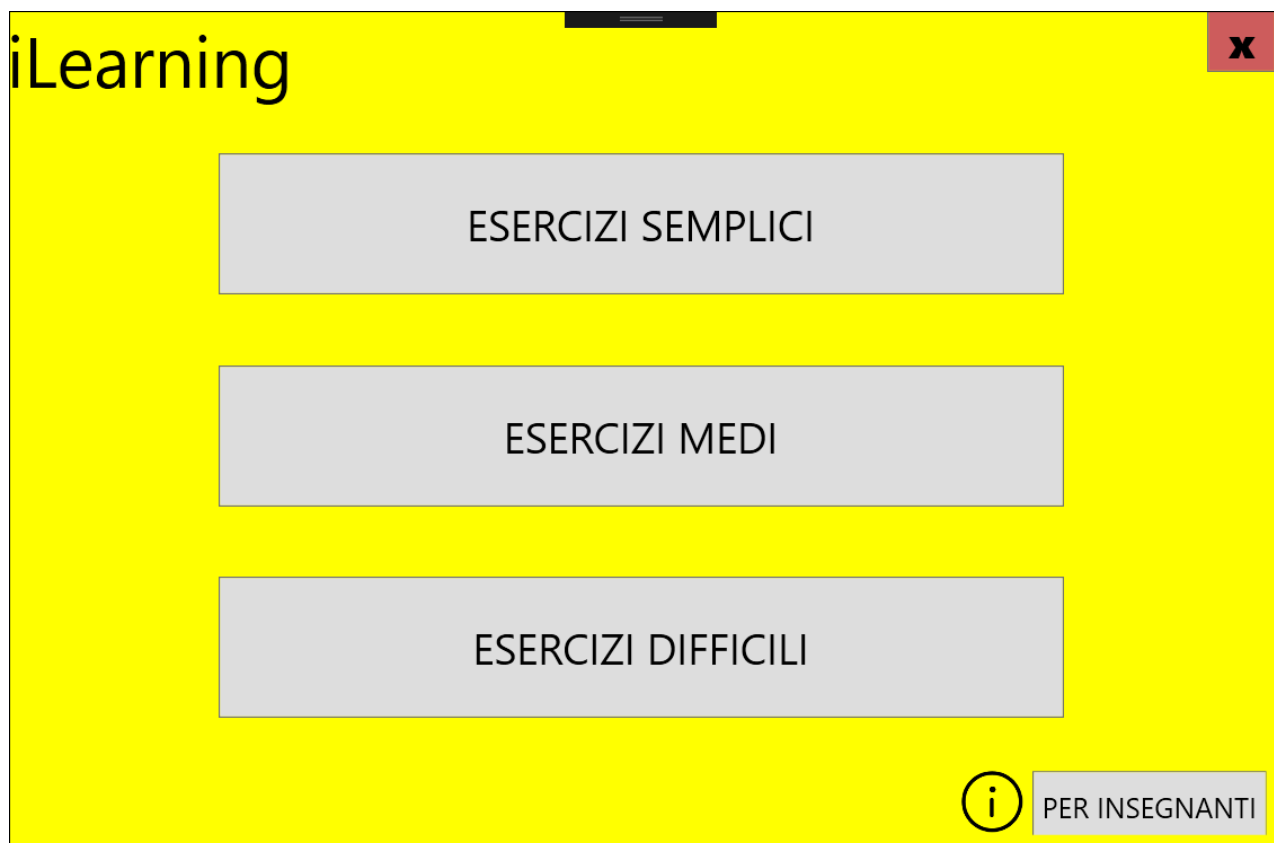
Questo l'ho messo in modo che l'oggetto selezionata venisse riconosciuto, in questo caso il percorso dell'immagine.

Punto della situazione rispetto alla pianificazione

Nonostante ho interrotto la realizzazione della interfacce non ho ancora riscontrato ritardi

Programma di massima per la prossima giornata di lavoro

Continuare la realizzazione delle interfacce.



Schermata per scelta categoria esercizi.

Codice per interagire con la tabella degli allievi:

```
2 riferimenti | 0 modifiche | 0 autori, 0 modifiche
public void Delete(Allievo a)
{
    db.Allievi.Remove(a);
}

4 riferimenti | 0 modifiche | 0 autori, 0 modifiche
public Allievo Get(int id)
{
    return db.Allievi.Where(c => c.AllievoId == id).FirstOrDefault();
}

4 riferimenti | 0 modifiche | 0 autori, 0 modifiche
public IEnumerable<Allievo> Get()
{
    return db.Allievi;
}

2 riferimenti | 0 modifiche | 0 autori, 0 modifiche
public void Insert(Allievo a)
{
    int newId = db.Immagini.Max(c => c.ImageId) + 1;
    a.AllievoId = newId;
    db.Allievi.Add(a);
}
```

Nome progetto:

2/3

ApplicazioneDidattica

```
2 riferimenti | 0 modifiche | 0 autori, 0 modifiche
public void Insert(Allievo a)
{
    int newId = db.Immagini.Max(c => c.ImageId) + 1;
    a.AllievoId = newId;
    db.Allievi.Add(a);
}

2 riferimenti | 0 modifiche | 0 autori, 0 modifiche
public void Update(Allievo a)
{
    // sfrutto il Get della classe
    Allievo allievoToUpdate = Get(a.AllievoId);
    if (allievoToUpdate != null)
        allievoToUpdate = a;
}
```

I metodi/operazioni sono molto comuni.