

SCELTE PROGETTUALI 2° PROGETTO: CODING TURTLES

Rota Leonardo (1086029)
Bonomelli Pietro (1087035)

Come scelta progettuale abbiamo deciso di seguire il caso A, ossia di ristrutturare il 1° progetto utilizzando Django e Bootstrap.

Il progetto è stato suddiviso in 6 app principali di Django:

- sale, che ha due sotto app:
 - prenotazioni
 - fasce Orarie
- clienti
- abbonamenti
- homepage

Le strutture del database sono in *sale/models.py* per le classi: Sala, Fascia Oraria, Prenotazione, Posto. Mentre *clienti/models.py* è stato utilizzato per le classi: Cliente, Abbonamento, SubAbbonamento.

- Ogni app ha un template (*app/templates/app/app.html*) che eredita *base.html* per diminuire la ripetizione di codice.
- Ogni App che mostra i dati (ossia tutte esclusa homepage) della palestra ha il file *static/app/app_logic.js* per la gestione dell'ordinamento delle colonne, mentre è stato utilizzato il file *app/views.py* per la gestione dei filtri (tramite una richiesta GET). Nel caso dell'app **sala** il file *views.py* è stato utilizzato anche per gestire le funzioni CRUD.

Gli **urls** del progetto sono stati gestiti in questo modo:

Progetto Django

```
|— path("")          -> homepage
|— path("clienti/")
|— path("abbonamenti/")
|— path("sale/")
|   |— sale.urls
|   |— prenotazioni.urls ( /sale/prenotazioni )
|   |— fasce_orarie.urls ( /sale/fasce_orarie )
```

Come si può notare dal diagramma, abbiamo mantenuto la struttura per la quale le pagine **prenotazioni** e **fasce_orarie** sono accessibili solo come sottopagine di **sale**.

La cartella **management**: contiene *popola_db.py*, necessario per il popolamento del database.

La cartella **static** contiene i contenuti statici del progetto, ossia i template per le pagine html, i vari fogli di stile css e le immagini.