

Tesina Tecnologie Web

Mattia Montanari

Ottobre 2024

Contents

1	Introduzione	2
1.1	Contesto e Scopo	2
1.2	Funzionalità Principali	3
2	Architettura del Sistema	4
2.1	Backend - Django	4
2.1.1	Componenti principali di Django	4
2.2	Frontend - HTML, Bootstrap e JavaScript	5
2.2.1	HTML	5
2.2.2	Bootstrap	5
2.2.3	JavaScript	5
3	UML	6
3.1	Use-Case Diagram	6
3.2	Class Diagram	7
3.3	Sequence Diagram	8
3.3.1	Prenotazione di un allenamento	8
3.3.2	Inserimento di una recensione	8
3.3.3	Richiesta Personal Traiener	9
3.4	State Diagram	9
3.4.1	Richiesta Personal Trainer	9
4	Immagini del sito	11
4.1	Homepage	11
4.2	Lista Allenatori	12
4.3	Prenotazione Allenamento	12
4.4	Recensione	13
4.5	Dashboard Amministratore	13
5	Conclusione	14
5.1	Difficoltà affrontate	14

Chapter 1

Introduzione

1.1 Contesto e Scopo

Il progetto di *Tecnologie web* nasce con l'obiettivo di creare un sito web per una palestra che consenta agli utenti di registrarsi, visualizzare personal trainer, prenotare sessioni di allenamento e lasciare recensioni. Questo documento descrive l'architettura del sistema e la progettazione mediante diagrammi UML.

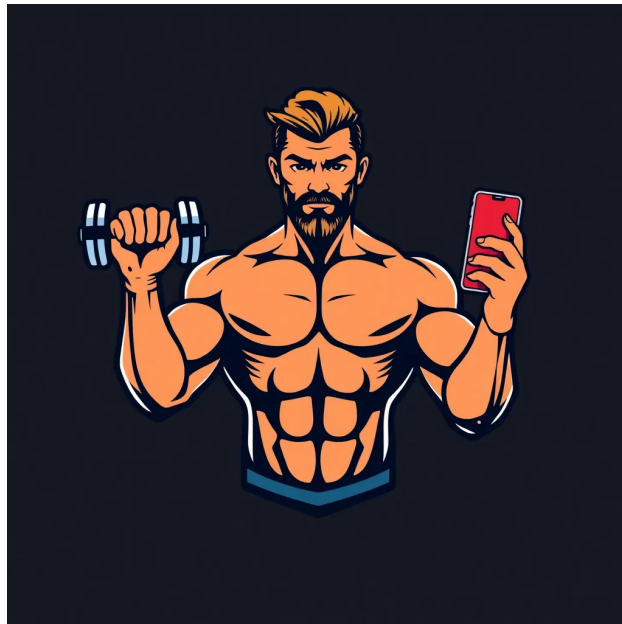


Figure 1.1: Logo Palestra

1.2 Funzionalità Principali

Le funzionalità principali del sistema includono:

- Registrazione e login degli utenti.
- Visualizzazione dei personal trainer disponibili.
- Prenotazione di sessioni di allenamento con personal trainer.
- Invio di recensioni e voti sui personal trainer.
- Gestione delle prenotazioni da parte dei personal trainer.

Chapter 2

Architettura del Sistema

2.1 Backend - Django

Il backend del sistema è realizzato utilizzando il framework *Django*, una piattaforma di sviluppo web basata su Python che permette di creare applicazioni in modo rapido e sicuro. Django segue il pattern *Model-View-Controller* (MVC), facilitando la separazione delle logiche di business, la gestione della presentazione dei dati e l'interazione con il database.

2.1.1 Componenti principali di Django

- **Model:** I modelli in Django rappresentano la struttura dei dati e sono strettamente collegati al database. I modelli definiti per questo progetto includono classi come *PersonalTrainer*, *RegistratoUtente*, *Recensione*, e *Prenotazione*, che corrispondono alle tabelle nel database.
- **View:** Le *View* in Django gestiscono la logica dell'applicazione. Ricevono le richieste HTTP dal client e restituiscono una risposta, che può essere una pagina web, dati in formato JSON o altro. Per esempio, le view per visualizzare la lista dei personal trainer o creare una prenotazione sono gestite qui.
- **Template:** I template rappresentano il livello di presentazione in Django. Sono utilizzati per generare pagine HTML dinamiche che vengono inviate al client. Attraverso i template, i dati provenienti dai modelli vengono mostrati all'utente.

2.2 Frontend - HTML, Bootstrap e JavaScript

Il frontend del sistema è realizzato utilizzando un insieme di tecnologie per garantire un'interfaccia utente responsiva, interattiva e intuitiva. Le tecnologie utilizzate sono:

2.2.1 HTML

HTML (HyperText Markup Language) è utilizzato per definire la struttura delle pagine web dell'applicazione. Gli elementi HTML organizzano il contenuto della pagina, come testi, immagini, pulsanti e form, consentendo agli utenti di interagire con il sito.

2.2.2 Bootstrap

Bootstrap è un framework CSS open-source che fornisce componenti predefiniti e stili per creare pagine web responsive e dall'aspetto professionale. Grazie a Bootstrap, l'interfaccia del sito si adatta automaticamente a diverse dimensioni di schermo (desktop, tablet, smartphone) senza la necessità di scrivere codice CSS personalizzato. Le principali componenti di Bootstrap utilizzate nel progetto includono:

- **Navbar:** Una barra di navigazione per muoversi tra le varie sezioni del sito.
- **Card:** Utilizzata per visualizzare le informazioni dei personal trainer in un layout ordinato e chiaro.
- **Form:** Utilizzato per l'inserimento di dati, come la prenotazione delle sessioni o l'invio di recensioni.

2.2.3 JavaScript

JavaScript è il linguaggio di scripting che gestisce l'interattività della pagina web. Nel progetto, JavaScript viene utilizzato per:

- Colorare in modo dinamico le stelle delle recensioni in base a quale stella viene cliccata.

Combinando HTML, Bootstrap e JavaScript, l'applicazione web è in grado di offrire un'esperienza utente fluida e piacevole, mantenendo al contempo un aspetto moderno e responsive.

Chapter 3

UML

Di seguito verranno riportati i principali diagrammi UML del progetto.

3.1 Use-Case Diagram

Nello Use-Case Diagram sono rappresentati i principali attori e le azioni che essi possono compiere.

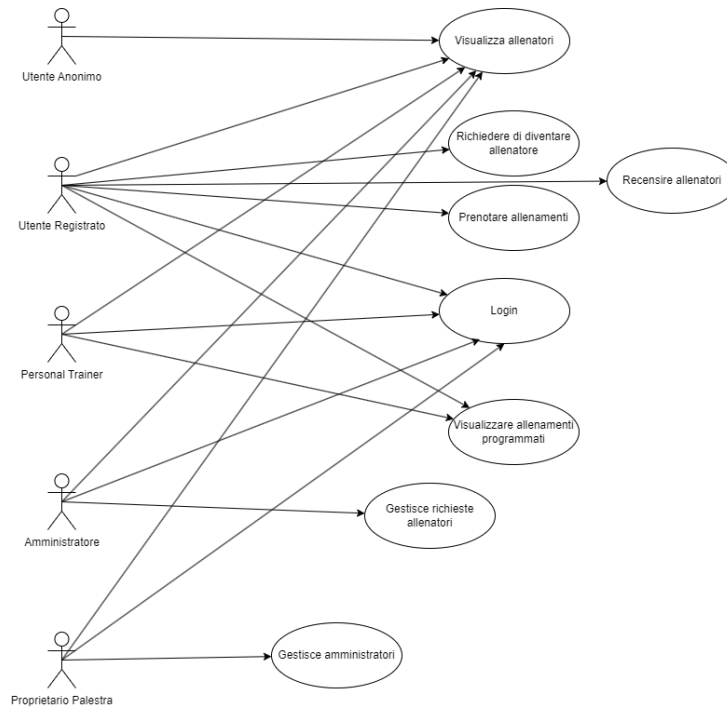


Figure 3.1: Use-Case Diagram

3.2 Class Diagram

Il Class Diagram rappresenta la struttura del sistema. Le classi principali sono:

- **User:** è il modello di utente predefinito che ci fornisce già implementato Django.
- **RegistratoUtente:** Rappresenta gli utenti registrati.
- **PersonalTrainer:** Rappresenta i personal trainer del sistema.
- **Recensione:** Gestisce le recensioni lasciate dagli utenti sui personal trainer.
- **Prenotazione:** Gestisce le prenotazioni effettuate dagli utenti per le sessioni di allenamento.
- **RichiestaPersonalTrainer:** Rapresenta la richiesta che un utente loggato invia per richiedere di diventare un Personal Trainer.

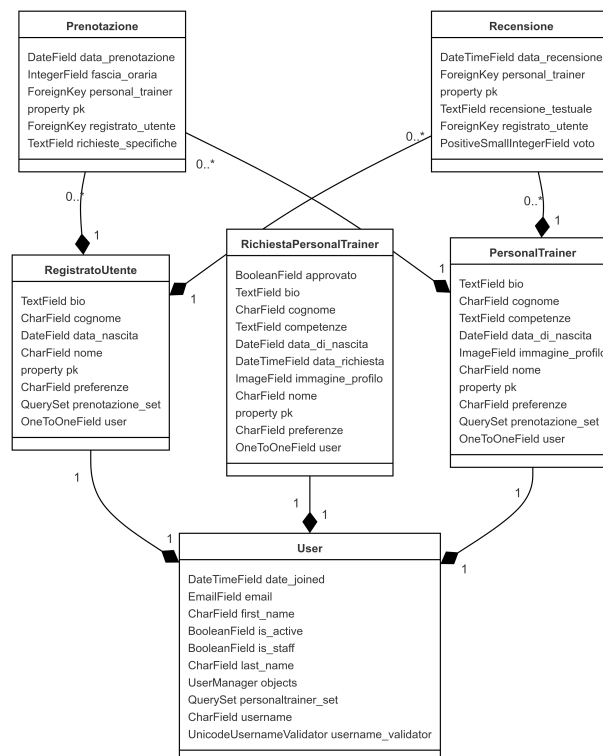


Figure 3.2: Class Diagram

3.3 Sequence Diagram

Il sequence diagram descrive in modo grafico come le varie classi interagiscono fra di loro e quali "informazioni" si scambiano per svolgere un determinato compito.

3.3.1 Prenotazione di un allenamento

Di seguito viene riportato il processo di prenotazione di un allenamento:

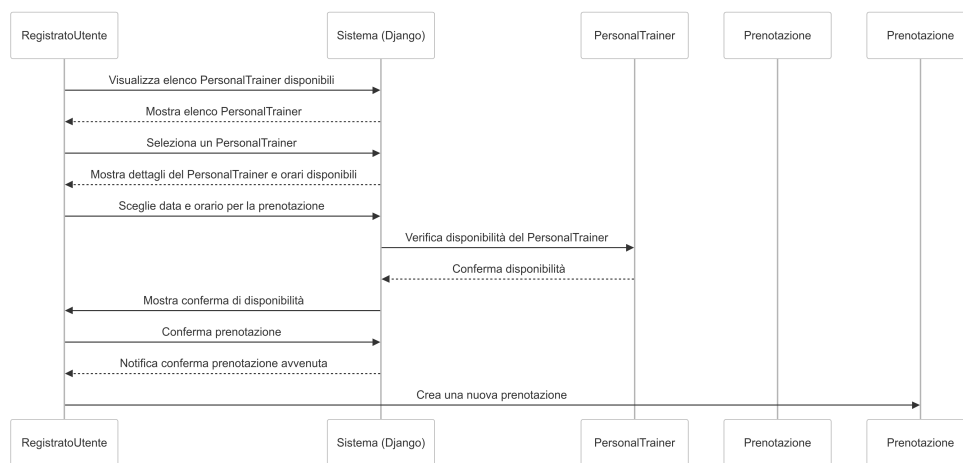


Figure 3.3: Sequence Diagram - Creazione di una Prenotazione

3.3.2 Inserimento di una recensione

Di seguito viene riportato il processo creazione di una recensione:

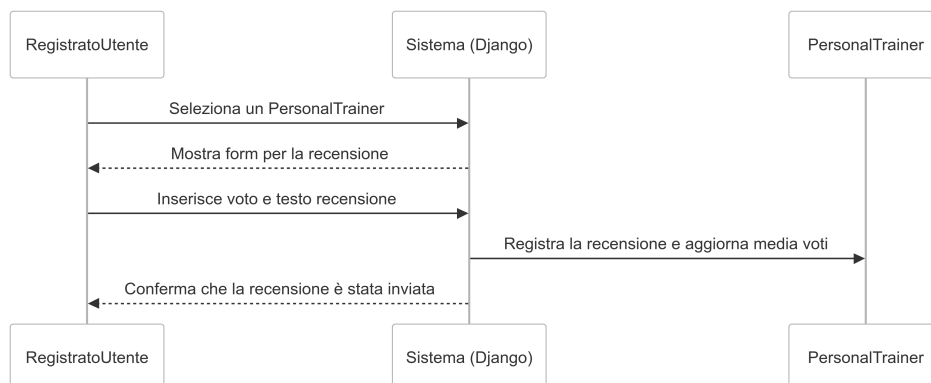


Figure 3.4: Sequence Diagram - Creazione di una Recensione

3.3.3 Richiesta Personal Traiener

Di seguito viene riportato il processo di invio di una richiesta per diventare Personal Trainer:

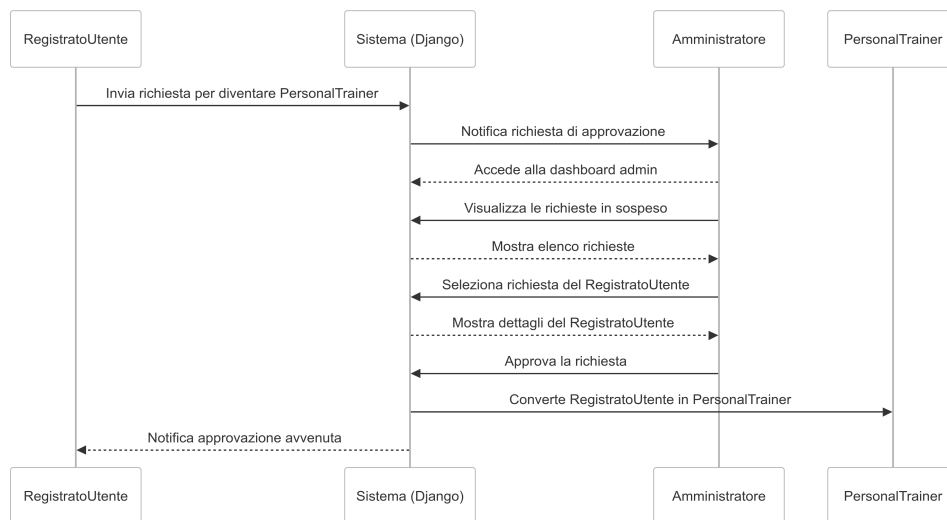


Figure 3.5: Sequence Diagram - Creazione di una Richiesta

3.4 State Diagram

Lo state diagram, mostra il comportamento di un sistema evidenziando i diversi stati che un oggetto attraversa durante il suo ciclo di vita e gli eventi che causano queste transizioni di stato

3.4.1 Richiesta Personal Trainer

Di seguito vengono riportati gli stati in cui una richiesta di diventare personal trainer si può trovare.

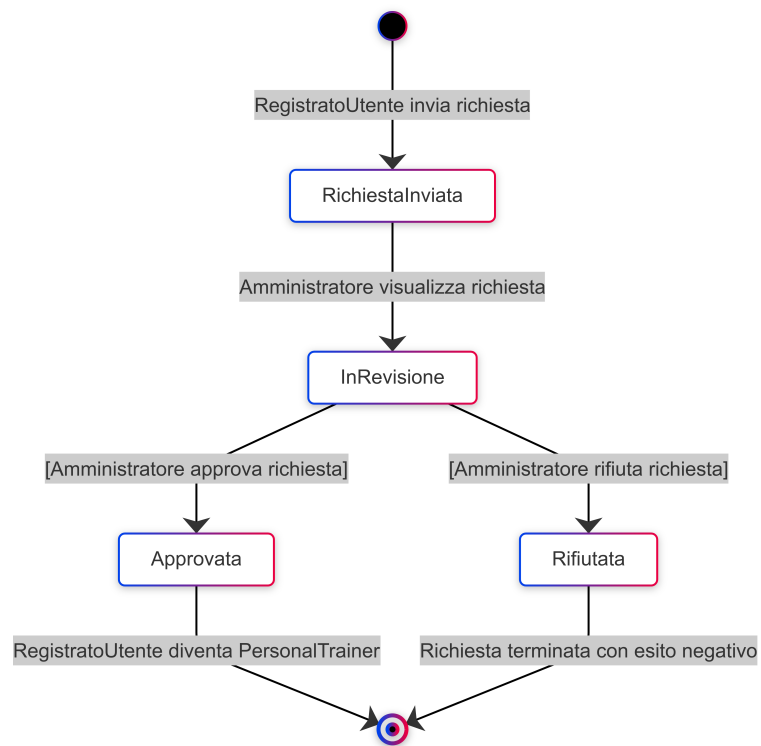


Figure 3.6: State Diagram - Richiesta Personal Trainer

Chapter 4

Immagini del sito

Ecco alcuni screenshot del sito.

4.1 Homepage

Di seguito viene riportata una immagine della Homepage:

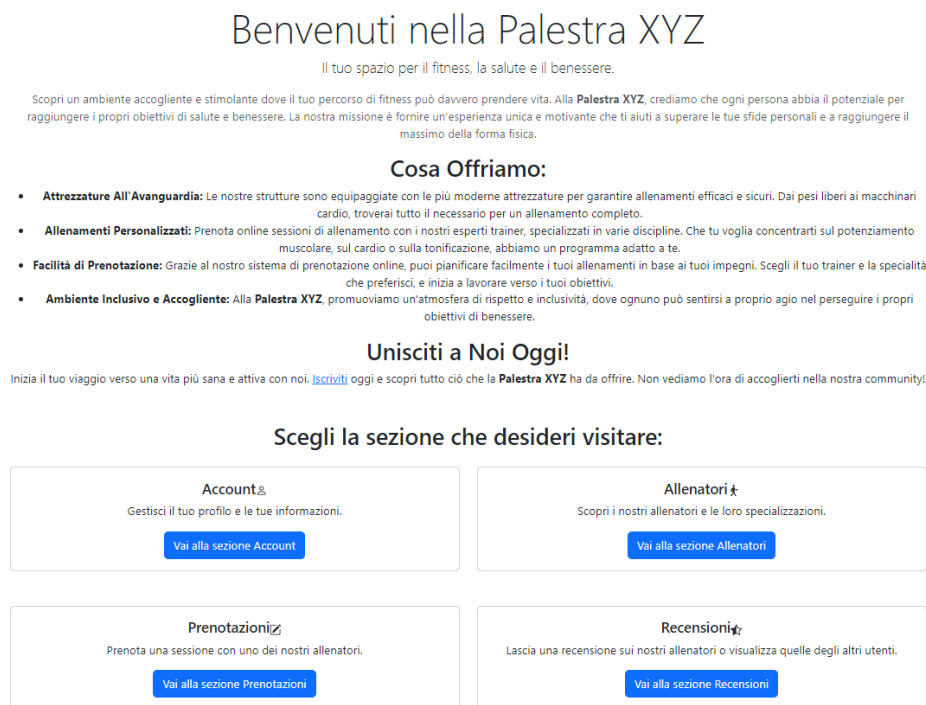


Figure 4.1: Homepage

4.2 Lista Allenatori

Di seguito viene riportata una immagine della lista degli allenatori:

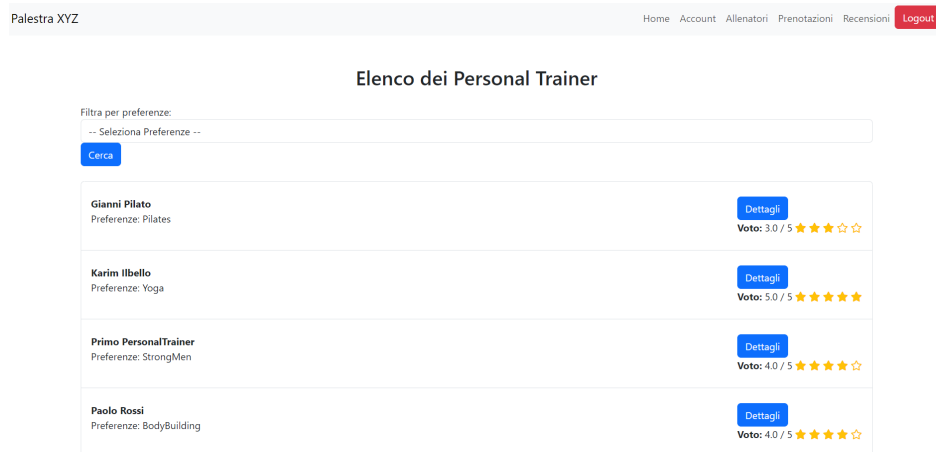


Figure 4.2: Lista Allenatori

4.3 Prenotazione Allenamento

Di seguito viene riportata una immagine della prenotazione di un allenamento:

The screenshot shows the 'Crea una Prenotazione' form. It includes a dropdown for 'Personal trainer:' with the selected value 'Kelly Montanari - Preferenza: Yoga'. Below this is a time range selector 'Fascia oraria:' set to '11:00 - 12:00'. The date selector 'Data prenotazione:' is set to 'October 11, 2024'. A text area for 'Richieste specifiche:' contains the text 'Vorrei che durante l'allenamento venga riprodotta musica rilassante'. A blue 'Prenota' button is positioned below the text area. At the bottom of the form, there are two links: 'Scopri tutti i nostri Personal Trainer disponibili nella palestra:' followed by a 'Lista Personal Trainer' button, and 'Verifica la disponibilità dei Personal Trainer per trovare il momento giusto per prenotare:' followed by a 'Verifica Disponibilità' button.

Figure 4.3: Prenotazione Allenamento

4.4 Recensione

Di seguito viene riportata una immagine dell'inserimento di una recensione:

Scrivi una recensione per Kelly Montanari

Recensione testuale:

Kelly è un'allenatrice eccezionale! Grazie alla sua professionalità e alla sua capacità di motivare, ho raggiunto risultati che non avrei mai immaginato. È sempre disponibile e attenta alle mie esigenze, creando allenamenti personalizzati e divertenti. La consiglio vivamente!

Voto



Invia Recensione

Annulla

Figure 4.4: Inserimento Recensione

4.5 Dashboard Amministratore

Di seguito viene riportata una immagine della dashboard dell'amministratore:

Dashboard Amministratore

Richieste per diventare Personal Trainer	
Michael Jordan	Dettagli

Personal Trainers	
Primo PersonalTrainer	Elimina
Primo PersonalTrainer	Elimina
Karim Ilbello	Elimina
Pippone Baudone	Elimina
Kelly Montanari	Elimina
Pinco Pallino	Elimina

Figure 4.5: Dashboard Amministratore

Chapter 5

Conclusione

In questa tesina abbiamo descritto l'architettura e la progettazione del progetto di *Tecnologie Web*. I diagrammi UML hanno illustrato le interazioni tra i vari componenti e il flusso di dati tra le entità principali del sistema.

5.1 Difficoltà affrontate

La principale difficoltà che ho riscontrato durante il progetto è stata associare l'utente fornito da Django ad un modello creato da me, per poter sfruttare la parte già implementata di *login* e *logout*.