

# UNIVERSITÀ DEGLI STUDI DI MODENA E REGGIO EMILIA

DIPARTIMENTO DI INGEGNERIA "ENZO FERRARI"

Elaborato per l'insegnamento di Tecnologie Web

## Django Project: Tatum24

### Code Sharing Web Application

Matricola: [Cliccami per accedere al github!](#) Studente:  
**165238** **Gianmarco Michelini**

---

Anno Accademico 2023/2024

# Indice

<b>1</b>	<b>Introduzione</b>	<b>3</b>
1.1	Nascita del Progetto e Ispirazione . . . . .	3
1.2	Obiettivo e Funzionamento . . . . .	3
<b>2</b>	<b>Specifica dei requisiti</b>	<b>4</b>
2.1	Requisiti Funzionali . . . . .	4
2.2	Requisiti Non Funzionali . . . . .	6
2.2.1	Django Framework . . . . .	6
2.2.2	Database . . . . .	6
2.2.3	Tools e librerie esterne . . . . .	6
2.3	Requisiti Essenziali del Progetto . . . . .	6
<b>3</b>	<b>Design</b>	<b>7</b>
3.1	Organizzazione Logica . . . . .	7
3.2	Diagrammi UML . . . . .	8
<b>4</b>	<b>Testing e Risultati</b>	<b>10</b>
4.1	Codice Applicativo . . . . .	10
4.2	View . . . . .	10
4.3	Risultati Ottenuti . . . . .	11
<b>5</b>	<b>Conclusioni</b>	<b>14</b>
5.1	Problemi Riscontrati . . . . .	14
5.2	Considerazioni Personali . . . . .	14

# Capitolo 1

## Introduzione

### 1.1 Nascita del Progetto e Ispirazione

Tatum24 nasce dalla necessità di disporre di un repository per i frammenti di codice (snippets). L'utilità di tale meccanismo deriva dalla versatilità che il codice già scritto fornisce. L'ispirazione mi è stata offerta da applicazioni web come "[Stack Overflow](#)", che, sebbene concettualmente diversa, mi ha guidato nella realizzazione della mia applicazione Web.

### 1.2 Obiettivo e Funzionamento

Tatum24 è una piattaforma online per la gestione e la condivisione di snippet di codice. All'interno di questo sito gli utenti potranno trovare snippet di codice già scritti in grado di risolvere determinati problemi di programmazione.

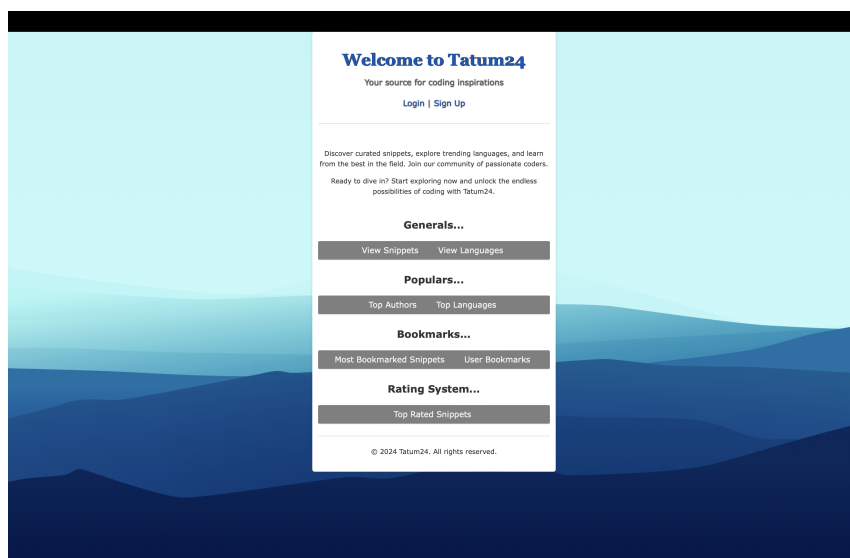


Figura 1.1: Tatum24: Home Page

# Capitolo 2

## Specifica dei requisiti

### 2.1 Requisiti Funzionali

I requisiti funzionali definiscono cosa deve fare il sistema. Qui sono riportati i principali requisiti funzionali non banali riguardanti Tatum24.

**Utente non registrato:**

- Visualizzare snippet di codice già scritti

**Utente registrato:**

- Visualizzare snippet di codice già scritti
- Creare e editare snippet di codice personali
- Gestire i commenti sotto i propri snippet
- Scrivere commenti per gli snippet
- Votare (like o dislike) gli snippet
- Scaricare gli snippet
- Visualizzare il proprio profilo
- Modificare il profilo
- Aggiungere (o eliminare) segnalibri per gli snippet

**Utente moderatore:**

Oltre alle funzionalità base di un utente registrato, l'utente moderatore è in grado di:

- Eliminare qualsiasi snippet
- Eliminare qualsiasi commento
- Modificare qualsiasi snippet

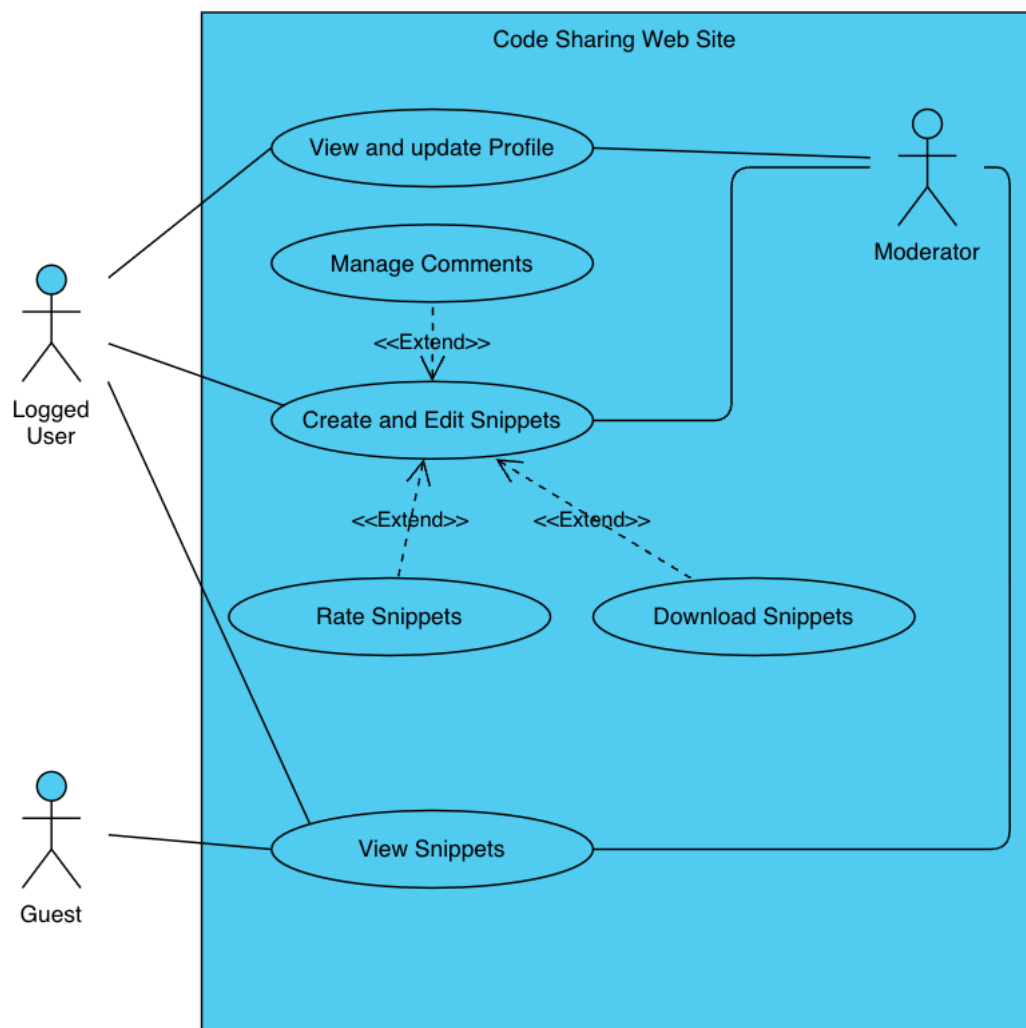


Figura 2.1: Tatum24: Use Case Diagram

## 2.2 Requisiti Non Funzionali

### 2.2.1 Django Framework

Utilizzare Django come framework per lo sviluppo di Tatum24 ha migliorato significativamente l'efficienza e la qualità del prodotto. Ciò è stato reso possibile grazie ad alcune caratteristiche fondamentali di Django: ORM integrato, l'interfaccia admin automatica, sistema di autenticazione e autorizzazione integrato, e l'aspetto legato alla sicurezza del sito Web.

### 2.2.2 Database

L'applicazione in termini di gestione degli accessi al database non presenta un'alta complessità, per questo motivo ho usufruito della semplicità di utilizzo, leggerezza e velocità di SQLite.

### 2.2.3 Tools e librerie esterne

Oltre a Django ho utilizzato delle librerie per migliorare le funzionalità dell'applicazione.

- Markdown: utilizzata per formattare il codice degli snippet.
- Pygments: utilizzata per evidenziare il codice degli snippet. Si evidenzia sulla base di un'analisi sintattica legata al linguaggio con cui è stato scritto il codice stesso.
- xhtml2pdf: utilizzata per generare pdf relativi a snippet di codice.
- Pillow: per la gestione delle immagini profilo.

## 2.3 Requisiti Essenziali del Progetto

Le informazioni relative ai requisiti essenziali per far funzionare il progetto sono reperibili all'interno del file `requirements.txt`, all'interno della root directory del github stesso.

[Cliccami per visualizzare i requisiti necessari](#)

Comando per scaricare e installare i requisiti del progetto: `pip3 install -r requirements.txt` (avendo già effettuato l'accesso al virtual environment con il comando `pipenv shell`).

Per avviare il server eseguire il comando: `python3 manage.py runserver`. Il server sarà accessibile dall'url: `localhost:8000`

# Capitolo 3

## Design

Tatum24 presenta un'architettura di design basata su un approccio orientato agli oggetti e modulare, capace di gestire la complessità del software e migliorare la struttura del progetto.

### 3.1 Organizzazione Logica

L'applicazione Web è stata suddivisa in 5 applicazioni fondamentali: tatum24 (la root), bookmarks, ratings, users e snippets. Il motivo per il quale ho deciso di dividere l'applicazione in tale maniera deriva dal fatto che avevo bisogno di separare i compiti all'interno dell'applicazione.

- Tatum24: è la root del progetto, all'interno della quale si hanno le informazioni più importanti del progetto e di gestione.
- Bookmarks: si occupa della gestione dei segnalibri relativi agli snippet
- Snippets: gestisce tutto ciò che concerne uno snippet di codice, a partire da come viene memorizzato alla visualizzazione effettiva nella pagina Web.
- Ratings: si occupa della gestione delle valutazioni fornite dagli utenti agli snippets.
- Users: contiene gli elementi fondamentali dedicati alla gestione degli utenti dell'applicazione

## 3.2 Diagrammi UML

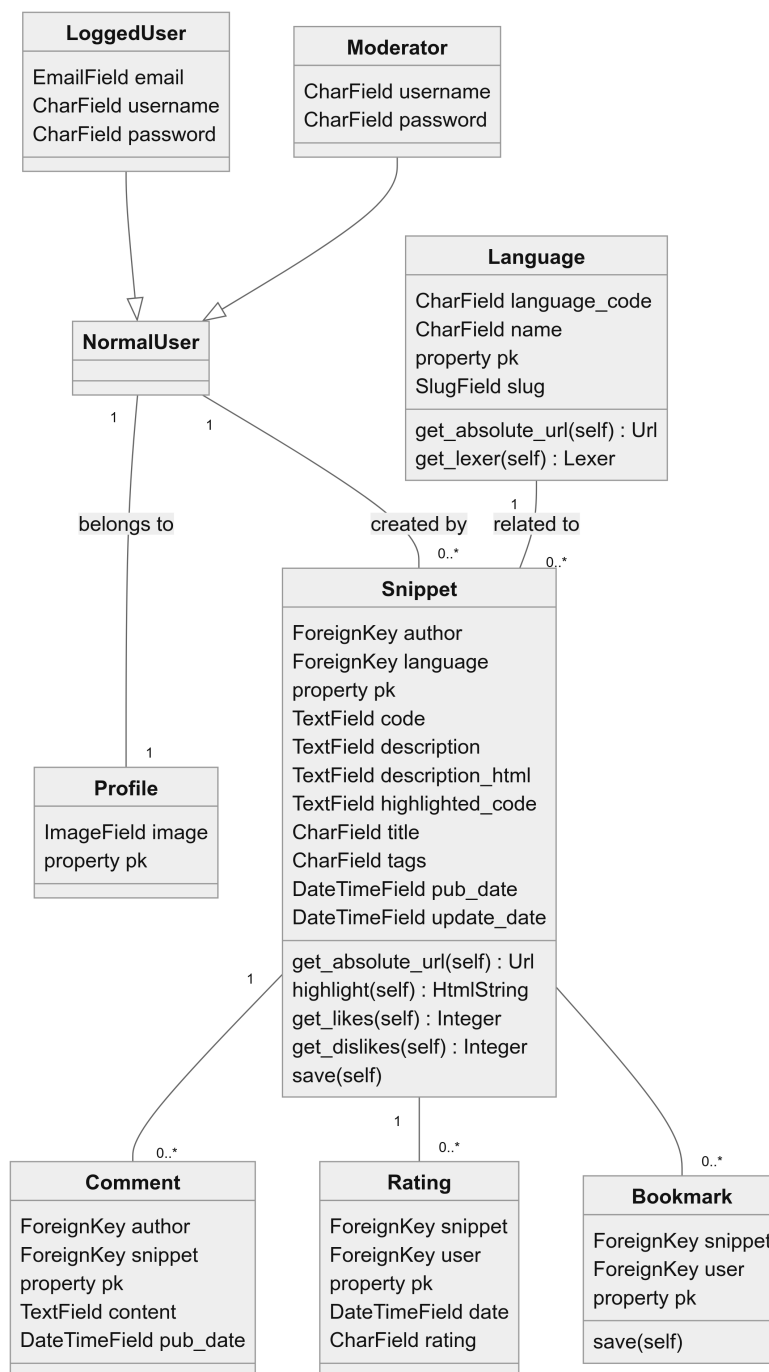


Figura 3.1: Tatum24: Class Diagram



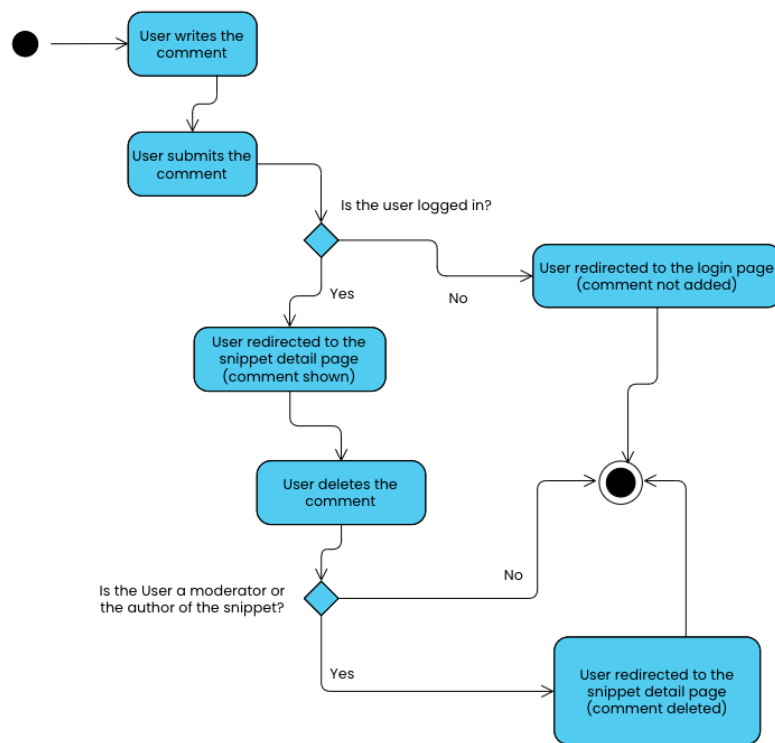


Figura 3.2: Tatum24: Activity Diagram, legato all'inserimento di un commento e successiva eliminazione

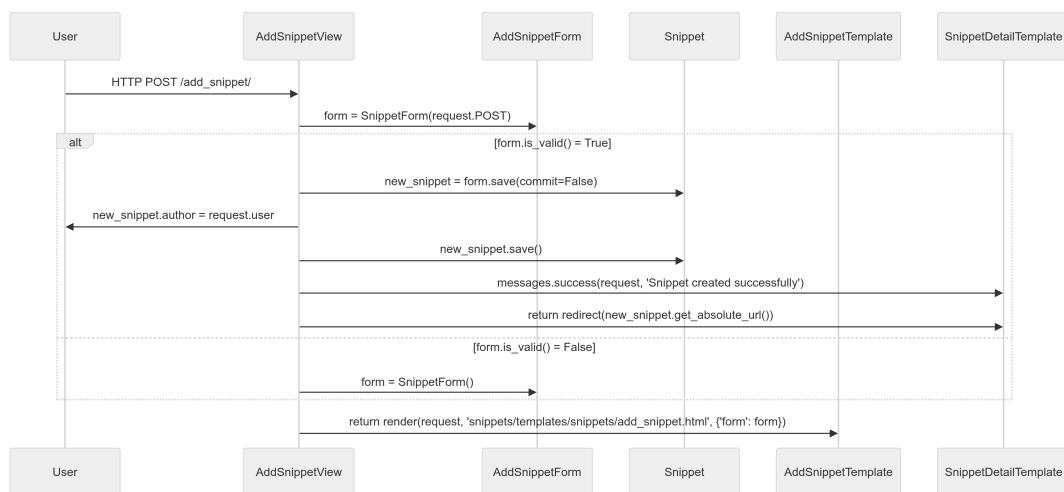


Figura 3.3: Tatum24: Sequence Diagram, legato all'operazione di aggiunta di uno snippet di codice

# Capitolo 4

## Testing e Risultati

I test sono stati pianificati e implementati sulla base della comprensione dei requisiti e identificazione delle funzionalità critiche, questo per quanto riguarda i commenti e la view relativa alla modifica di uno snippet di codice.

### 4.1 Codice Applicativo

`CommentTestCase`. In ottica black-box ho sviluppato dei test per i commenti che verificano: la corretta creazione, anche con diversi tipi di contenuto, l'aggiornamento e l'eliminazione.

- `test_create_comment_valid()`
- `test_create_comment_long_content()`
- `test_create_comment_unicode_content()`
- `test_update_comment()`
- `test_delete_comment()`
- `test_get_absolute_url()`

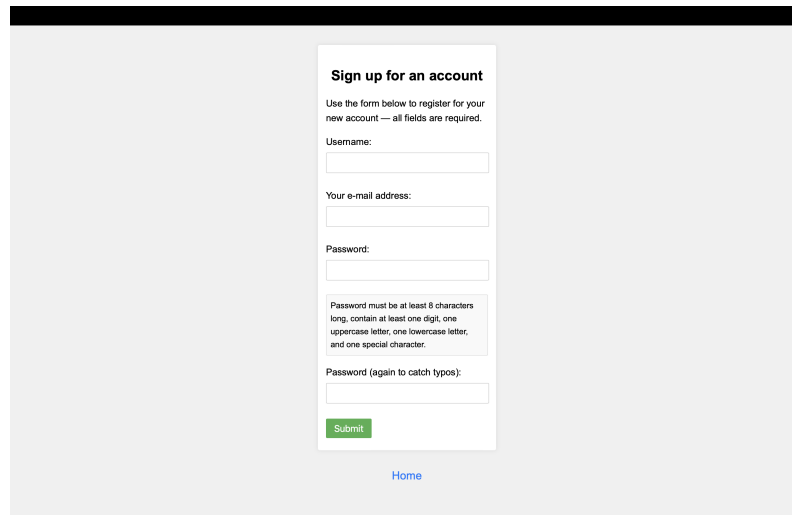
### 4.2 View

`EditSnippetViewTestCase`. Viene testata la view corrispondente alla modifica di uno snippet. In questo scenario bisogna assicurarsi della corretta implementazione dei permessi per gli utenti.

- `test_edit_snippet_author`
- `test_edit_snippet_moderator`
- `test_edit_snippet_no_permission`

## 4.3 Risultati Ottenuti

Di sotto vengono riportati alcune pagine importanti del sito Web.



**Sign up for an account**

Use the form below to register for your new account — all fields are required.

Username:

Your e-mail address:

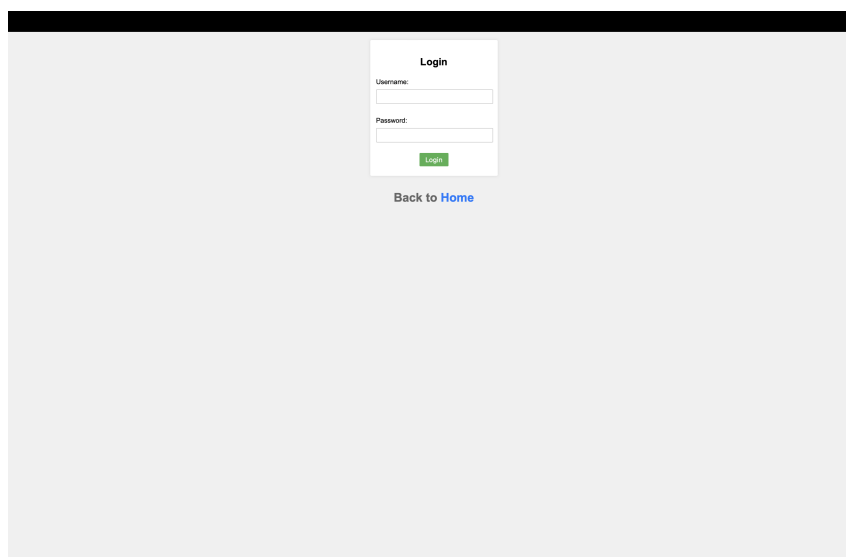
Password:

Password must be at least 8 characters long, contain at least one digit, one uppercase letter, one lowercase letter, and one special character.

Password (again to catch typos):

[Home](#)

Figura 4.1: Sign up



**Login**

Username:

Password:

[Back to Home](#)

Figura 4.2: Login

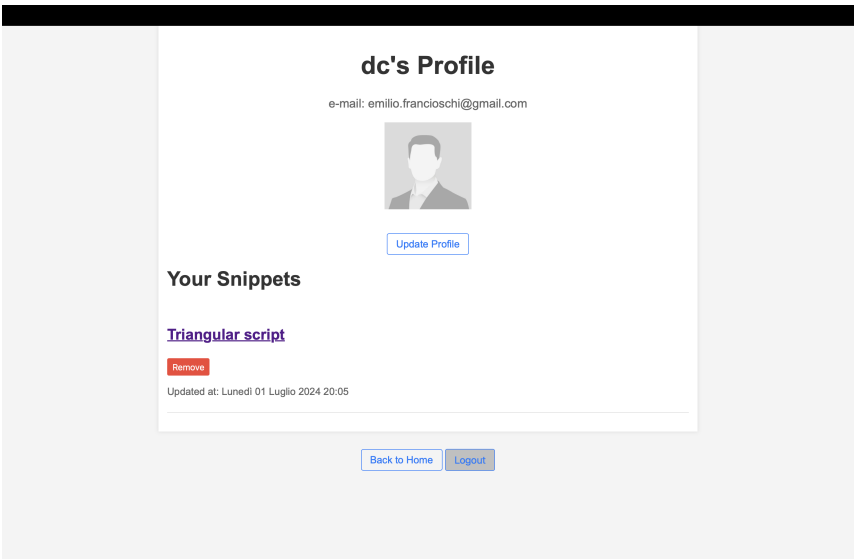


Figura 4.3: Profile

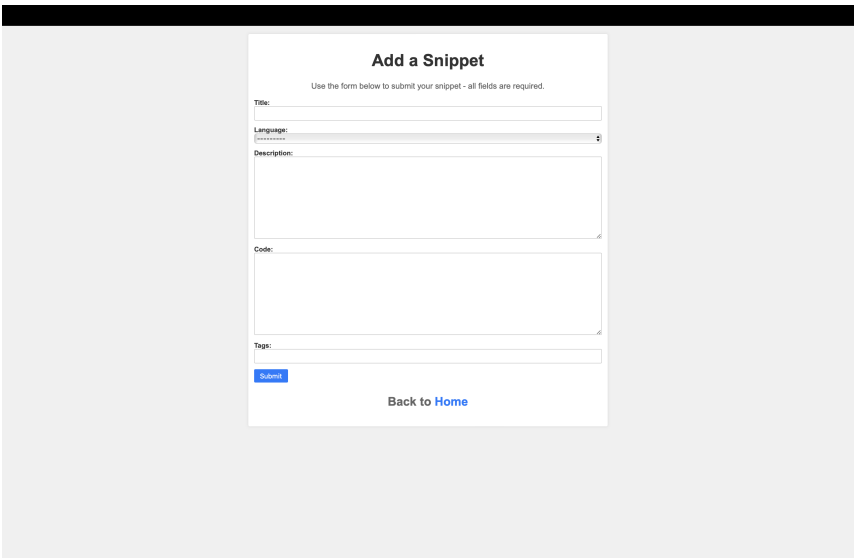


Figura 4.4: Add Snippet Form

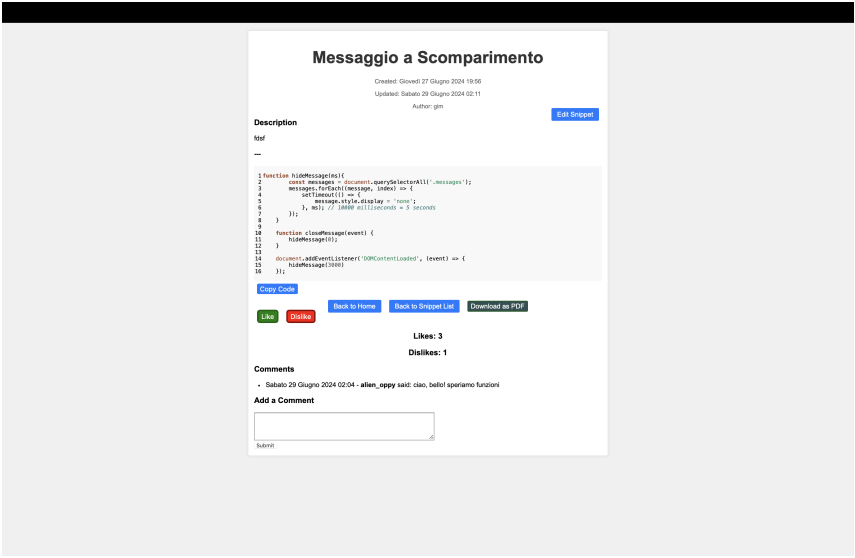


Figura 4.5: Snippet Detail

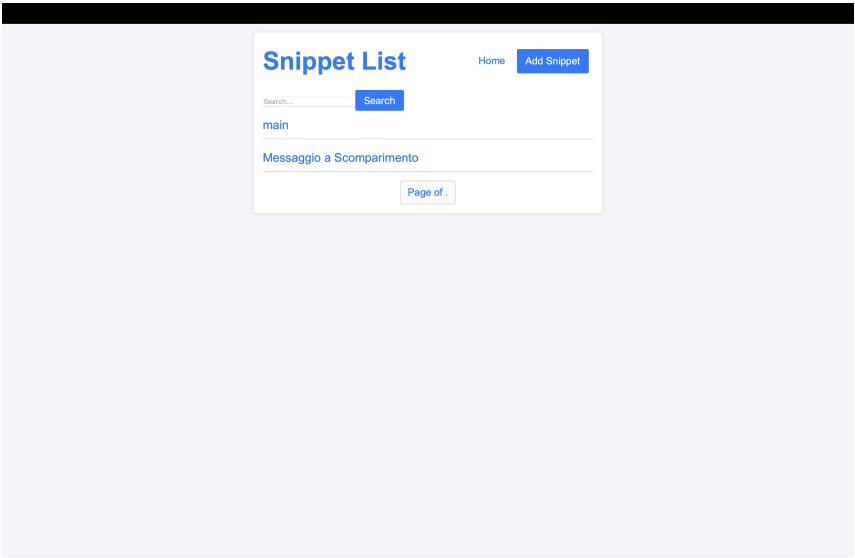


Figura 4.6: Snippet List

# Capitolo 5

## Conclusioni

L'applicazione, descritta all'interno di questo elaborato, ovviamente non è ultimata ma è solamente un punto di partenza per l'utilizzo di Django come framework per lo sviluppo di applicazioni Web. Difatti il progetto può essere, sicuramente, migliorato ed ampliato sviluppando ulteriori funzionalità caratteristiche di un sito Web dedicato al code sharing. Ad esempio, potrebbero essere sviluppati anche ulteriori meccanismi di sicurezza, ottimizzazione delle prestazioni e integrazione con altre piattaforme e servizi per offrire un'esperienza utente più completa e professionale.

### 5.1 Problemi Riscontrati

I principali problemi riscontrati riguardano l'utilizzo di Django e la programmazione con Python in generale. Django è un framework molto intuitivo ma non per questo è semplice da utilizzare, mi è capitato spesso che, nonostante abbia seguito con molta attenzione le lezioni del corso, non riuscissi a implementare le idee che avevo progettato. In particolare, nello sviluppo di query con l'ORM di Django. Non avendo frequentato corsi di pura programmazione riguardanti Python ho avuto numerose difficoltà nella gestione delle operazioni e nella ricerca di modi efficienti per implementarle.

### 5.2 Considerazioni Personali

Progettare e implementare Tatum24 mi ha permesso di affrontare il "nuovo" tema della programmazione Web, un elemento che ritengo fondamentale per i programmatori di oggi. Durante lo svolgimento dell'insegnamento ho avuto l'opportunità di esplorare diverse tecnologie e strumenti moderni, comprendendo le best practice per lo sviluppo di applicazioni web performanti e sicure. In particolare, ho avuto modo di lavorare con Django e varie librerie come Javascript o altri framework come Angular che, attraverso interfacce utente dinamiche e interattive, migliorano la user experience. Inoltre, ho acquisito familiarità con il backend development.