



Grimosport

2023

Biagio Grimalizzi

Matr. 140818

Unimore DIF, Ingegneria Informatica

Progetto di Tecnologie Web

Indice

Indice	1
Introduzione	2
Requisiti	3
Descrizione progetto	4
Utenti	4
Use Case Diagram	5
Database	6
Tecnologie usate e organizzazione logica	7
Backend	7
Frontend	8
Scelte fatte	9
Recommendation system	9
Test	10
Product	10
Order	10
Risultati	11
Problemi riscontrati	15

Introduzione

Grimosport è un'applicazione web stile e-commerce, l'idea è stata quella di un influencer che crea il suo brand di abbigliamento sportivo "grimosport" e crea un sito web in cui poter mostrare gli articoli e poterli acquistare.

L'applicazione offre di poter visualizzare e aggiungere/rimuovere al carrello gli articoli, e oltre ad registrarsi o accedere al proprio profilo, anche di simulare dei veri e propri pagamenti.

Ho deciso, per imparare qualcosa di nuovo e per divertimento, di effettuare il deploy dell'applicazione su un mio server e dominio che gestisco con un mio amico, quindi l'applicazione web è disponibile a questo [link](#).

Requisiti

L'applicazione deve essere in grado di:

- Creare e/o accedere al profilo personale
- Visualizzare gli articoli disponibili anche senza registrazione
- Mostrare agli utenti che hanno fatto l'accesso e almeno un ordine degli articoli consigliati tramite un algoritmo di recommendation system
- Gestire un carrello
 - Aggiungere prodotti al carrello
 - Rimuovere prodotti dal carrello
 - Visualizzare il carrello con i prodotti
- Gestire un ordine e mostrare una pagina di conferma se andato a buon fine
- Ricercare articoli mediante la barra di ricerca
- Moderazione:
 - Possibilità di aggiungere nuovi articoli da un profilo moderatore
 - Possibilità di vedere le statistiche di vendita

Descrizione progetto

Utenti

L'applicazione ha 4 tipologie di utente:

1. Utente non registrato/autenticato (anonimo)
2. Utente autenticato non moderatore
3. Utente autenticato moderatore
4. Utente autenticato admin ("superuser")

L'utente non registrato/autenticato può navigare il sito vedendo gli articoli disponibili, le ultime uscite, le categorie e può anche aggiungere o rimuovere articoli dal carrello.

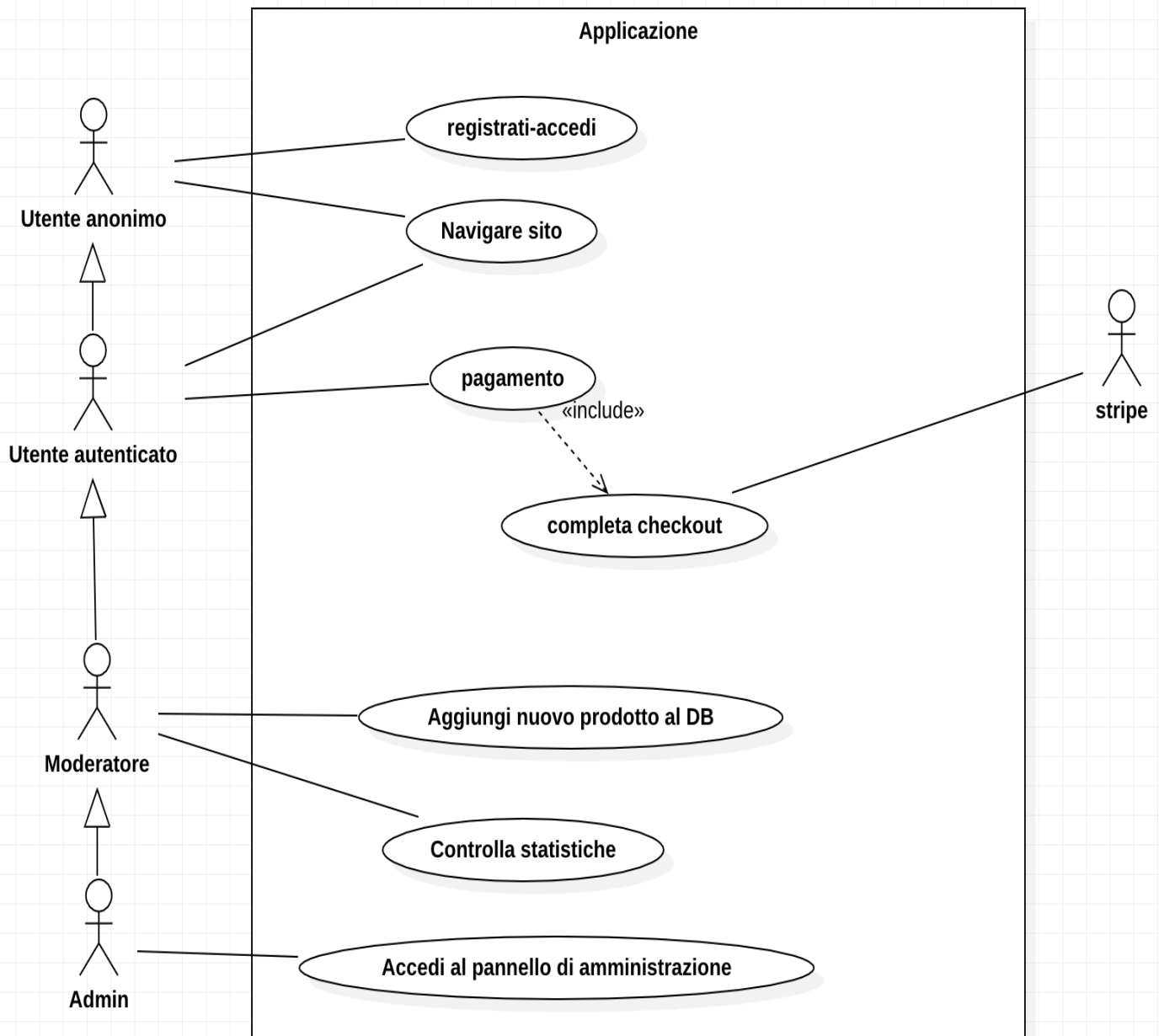
L'utente autenticato ma non moderatore può fare tutto ciò che fa l'utente anonimo, in più può procedere al checkout per pagare e vedere il proprio profilo con tutti gli ordini effettuati da lui.

L'utente autenticato moderatore può fare tutto ciò che fa l'utente autenticato non moderatore, in più può aggiungere dei nuovi prodotti compilando il form di aggiunta prodotto e controllare le statistiche di vendita dei prodotti.

L'utente autenticato admin può fare tutto ciò che fa l'utente autenticato moderatore, in più può accedere al pannello di amministrazione admin.

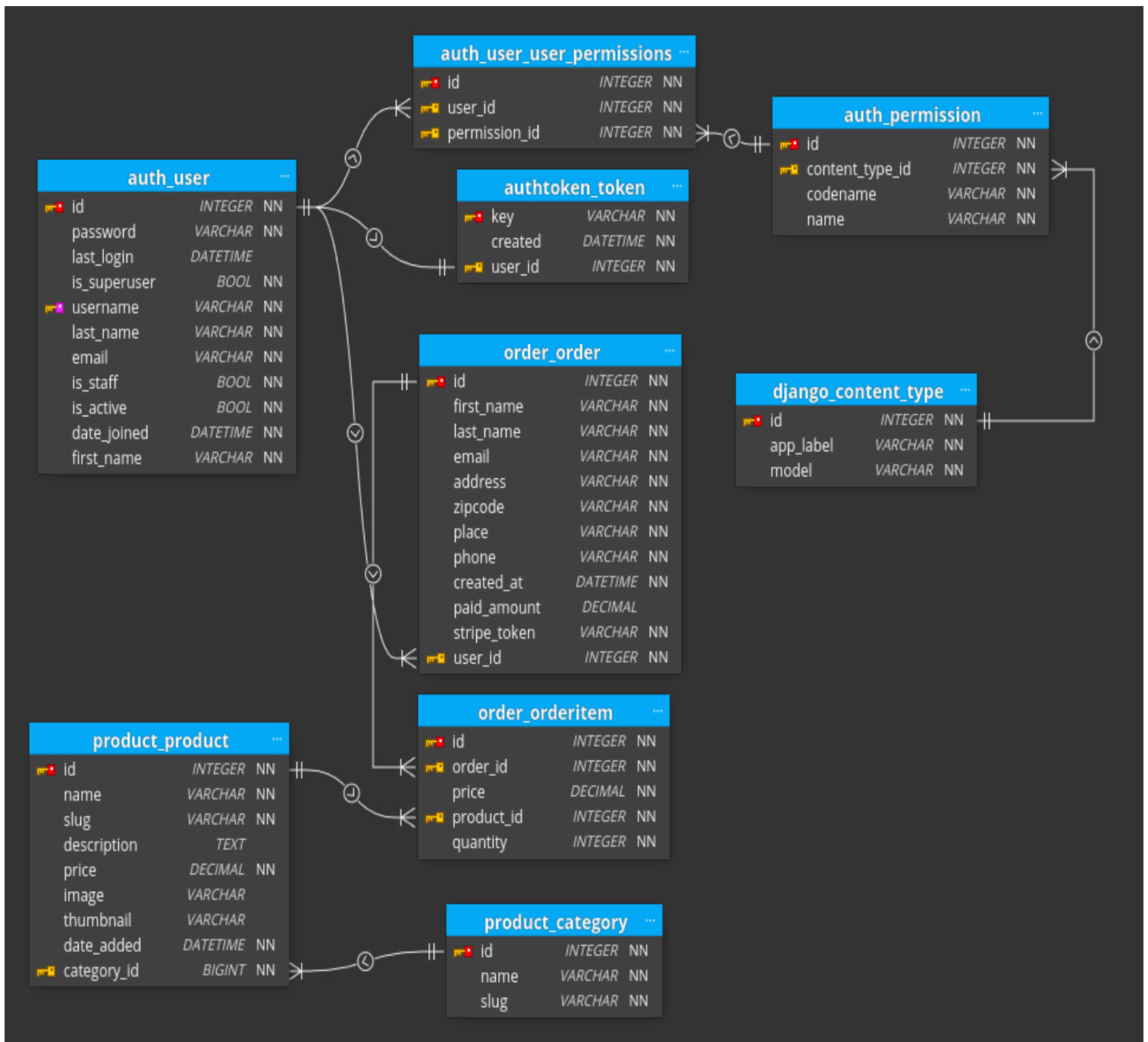
Nella pagina seguente mostro un USE CASE DIAGRAM (UML) che mostra le funzionalità per tipologia di utente.

Use Case Diagram



Database

Qui uno schema ER delle classi del database:



Tecnologie usate e organizzazione logica

Ho deciso, per rendere il progetto più professionale, di dividere il progetto in due parti, la parte di backend e quella di frontend.

Backend

Per il backend ho usato le seguenti tecnologie:

- django rest framework
- sqlite
- pillow
- django cors headers
- stripe

Django rest framework mi ha permesso di creare una vera e propria API RESTful.

Sqlite è una libreria in linguaggio C che implementa un motore di database SQL piccolo, veloce, autonomo, ad alta affidabilità e completo di funzionalità per questo l'ho scelto.

Pillow è una libreria python che ti permette di gestire le immagini.

Stripe invece è una piattaforma di pagamento che ti permette di accettare pagamenti online tramite carte di credito. Mi sono registrato e creato la mia key per simulare dei pagamenti, infatti si possono effettuare gli ordini sull'applicazione con una carta di credito fittizia che ti fornisce stripe, in modo tale da vedere gli ordini effettuati sul proprio profilo personale.

Frontend

Per il frontend ho usato le seguenti tecnologie:

- Vue.js
- Axios
- Bulma

Vue.js è un framework javascript accessibile, performante e versatile per la creazione di interfacce utente web.

Axios è un semplice client HTTP promise based per il browser. Axios fornisce una libreria semplice da usare in un piccolo pacchetto con un'interfaccia molto estensibile. L'ho usato per fare richieste HTTP.

Bulma è un framework gratuito e open source che fornisce componenti frontend pronti all'uso che puoi facilmente combinare per creare interfacce web reattive. L'ho usato per il css.

Scelte fatte

Recommendation system

Per il progetto ho deciso di implementare un semplice algoritmo di recommendation system che funziona nel seguente modo:

In base all'utente che ha effettuato l'accesso l'applicazione controlla gli ordini di quell'utente e conta in quale categoria ha acquistato più articoli, in base a questo l'app mostra nella home page gli ultimi prodotti usciti di codesta categoria.

Test

In questo capitolo si presentano e descrivono i test fatti sul progetto.

Product

Per la classe product ho testato le view esposte, in tutto ne sono 7 ma alcune si ripetono, per questo motivo ne elencherò solo 3 (le più importanti).

- **test_latest_product_list:** controlla se ritorna gli ultimi 4 prodotti del db e che lo status code sia 200.
- **test_search:** controlla la ricerca di prodotti.
- **test_set_product:** controlla se inserendo un nuovo prodotto nel db lo status code sia 201 e che effettivamente è stato aggiunto.

Order

Per la classe order ho testato le view esposte, in tutto ne sono 2.

- **test_get_order_list:** controllo se per un utente autenticato ritorna tutti gli ordini che ha fatto.
- **test_checkout:** è diviso in due
 - **test_checkout_with_valid_data:** controlla se inserendo dei valori esatti il checkout vada a buon fine e lo status code sia 201.
 - **test_checkout_with_invalid_data:** controlla se inserendo dei valori errati il checkout non vada a buon fine e lo status code sia 400

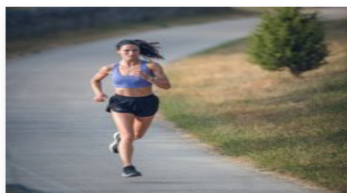
Risultati

Home page

GRIMOSPORT

Benvenuto nella home page di grimosport dove puoi trovare i nostri prodotti più recenti

Ultimi prodotti usciti



Pantaloncini donna estivi

€69,00

[Vedi dettagli](#)



Giubbino invernale running

€99,00

[Vedi dettagli](#)



Completo sportivo donna

€99,00

[Vedi dettagli](#)



Completo palestra uomo

€399,00

[Vedi dettagli](#)

Account non moderatore

Il mio account

Informazioni account

Ciao bea questi sono i tuoi ordini

[Logout](#)

I miei ordini

Riepilogo ordine

Prodotti

Prodotto	Prezzo	Quantità	Totale
Giubbino invernale running	99.00€	1	99.00€

Riepilogo ordine

Prodotti

Prodotto	Prezzo	Quantità	Totale
K-Way invernale	78.99€	1	78.99€

Riepilogo ordine

Prodotti

Prodotto	Prezzo	Quantità	Totale
----------	--------	----------	--------

Account moderatore

Questo è un account moderatore

[Logout](#)

[Vai al pannello di amministrazione](#)

Bentornat* admin, puoi vedere le statistiche di vendita cliccando il bottone qui sotto oppure compila tutti i campi per aggiungere un nuovo prodotto

[Vai alle statistiche](#)

Categoria:


Nome:

Slug:

Descrizione:

Prezzo:

Immagine:

 Scegli un'immagine

Nessun file selezionato

[Aggiungi prodotto](#)

Pagina visualizzazione carrello

GRIMOSPORT

Cosa stai cercando?

Chi siamo

EstateInverno

Profilo

Carrello (2)

Carrello

Prodotto	Prezzo	Quantità	Totale
Completo palestra uomo	399.00 €	1	399.00 €
K-Way invernale	78.99 €	1	78.99 €

Totale

477.99€, 2 prodotti nel carrello.

Procedi al checkout

Statistiche

GRIMOSPORT

Cosa stai cercando?

Chi siamo

EstateInverno

Profilo

Carrello (0)

Statistiche

Statistiche di vendita

Prodotto	Quantità totale di articoli di questo prodotto venduti	Quante volte il prodotto è stato in un ordine	Prezzo attuale del prodotto
Felpa grigia uomo	7	4	99.00 €
Completo palestra uomo	7	4	399.00 €
Completo invernale donna running	4	3	299.00 €
Giubbino invernale running	4	4	99.00 €
Prova-vestito	3	3	99.00 €
K-Way invernale	2	2	78.99 €
Felpa invernale donna	1	1	59.99 €
Completo running uomo	0	0	599.89 €
Reggiseno donna sportivo	0	0	49.99 €
Completo sportivo donna	0	0	99.00 €
Pantaloncini donna estivi	0	0	69.00 €

Pagina checkout

Checkout

Riepilogo ordine

Prodotto	Prezzo	Quantità	Totale
K-Way invernale	78.99€	1	78.99€
Totale		1	78.99€

Informazioni di spedizione

* Tutti i campi sono richiesti

Nome*

Indirizzo*

Cognome*

Città*

Email*

CAP*

Telefono*

Card number

MM / YY CVC

Paga

Problemi riscontrati

Il problema più importante e grande che ho riscontrato c'è stato quando ho implementato il modo di aggiungere dei nuovi prodotti al db per gli utenti moderatori, perchè dopo aver compilato il form e inviata la richiesta, al server arrivava un dizionario di stringhe che non sapeva gestire e quindi ritornava lo status code 400 (bad request).

Ho risolto questo problema usando la dictionary comprehension del python trasformando il dizionario di stringhe in uno con valori validi.

```
data = { a : (b[0] if isinstance(b, list) else b) for (a,b) in request.data.items() }
data['price'] = float(data['price'])
serializer = MyProductSerializer(data=data)
if serializer.is_valid():
    serializer.save()
    return Response(serializer.data, status=status.HTTP_201_CREATED)
else:
    return Response(serializer.errors, status=status.HTTP_400_BAD_REQUEST)
```

È stato interessante risolvere questo problema e scoprire di cosa è capace il python.

Oltre questo non sono stati riscontrati altri grossi problemi di sviluppo.